Proceedings - Index of authors

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | R | S | T |V |
W | X | Y | Z

home
Contents
Index of authors

A

B

C

J

K

M

R

S

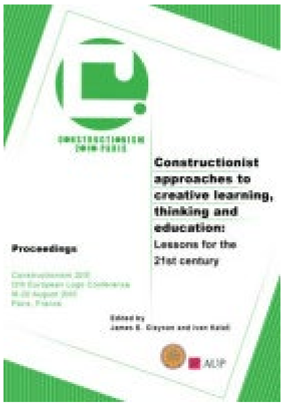
T

V

W

Constructionism 2010 • Paris

Constructionism 2010



Constructionism 2010
Constructionist approaches to creative learning, thinking and education: Lessons for the 21st century

Proceedings for Constructionism 2010
The 12th EuroLogo conference
16-20 August, 2010
Paris, France

ISBN 978-80-89186-65-5 (Proc)
ISBN 978-80-89186-66-2 (CD)

home
Contents
Index of authors

Associated conference websites

www.aup.edu/news/special_events/constructionism2010.htm
www.eurologo.org

Sponsors of the conference

Contents

Index of authors

Proceedings - Contents

Plenary Presentations

Papers

Workshops

Posters

Constructivism(s): Shared roots, crossed paths, multiple legacies

Edith K. Ackermann, *edith@media.mit.edu*
Design Lab, MIT School of Architecture / Media Lab (Swatch Lab), Cambridge MA, USA.

Abstract

This paper examines the shared roots and crossed paths between Jean Piaget's constructivism, what Seymour Paper refers to as "constructionism", and socio-cultural theories as epitomized by Lev Vygotsky. We do so in the light of more situated, pragmatic, and ecological approaches to human cognition. All these views are *developmental* (stressing the genesis children's interests and abilities over time), *experiential* (in the sense that knowledge is rooted in sensori-motor activity) and *interactionist* (people are seen as constructing their knowledge by transforming the world). Yet, the views also differ, each highlighting some aspects of how children grow and learn, while leaving other questions unanswered.

Piaget's main contribution was to flesh out what is common in children's ways of thinking at different stages of their cognitive development and, more important, how consistent, robust, and generally "adapted" their views are. The theory stresses the progressive de-contextualization of knowledge (from here-and-now to then-and-there) and identifies some of the hidden mechanisms (internal reorganizations) that drive human cognitive development. Papert, in contrast, stresses how individuals learn in context and how they use their own—and other people's—externalizations as objects to think with, especially as their convictions break down. His approach is more situated. Papert is particularly interested the role of new media in human learning. Both Papert and Vygotsky shed light on the articulations between direct and mediated experience (from action and tool-use to enactments, language, and symbol-use). Yet Vygotsky and the Russian school have paid much closer attention to the role of caring adults and peers in a child's initiation to her culture. They remind us that it takes a whole village to raise a child.

Integrating the views helps rethink how children come to make sense of their experiences, and how they find their own places—and voices—in the world. At once world-makers, world-readers, and dwellers in the world, human infants are granted from birth with the abilities to optimize exchanges with people and things by moving in and out of contexts, by shifting perspectives, and by switching roles or standpoint. They are extraordinary learners, and much can be learned from them. Lastly, while mostly inner-driven and curious, children need caring adults, secure grounds, and engaging peers and props to thrive and grow, Tools, media, and cultural artifacts are the tangible forms through which they explore their surrounds, express their thoughts, and share the fun with others—and the traces left by those who came before (cultural heritage) become a terrain for newcomers to create their paths.

Keywords

Constructivism, Piaget, Papert, Vygotsky, situated learning, embodied cognition, ecology of mind

Note

An earlier version of this paper appears in Ackermann. E. (2004) *Constructing knowledge and transforming the world.* In *A* learning zone of one's own: Sharing representations and flow in collaborative learning environments [M. Tokoro and L.Steels (Eds.). Amsterdam, Berlin, Oxford, Tokyo, Washington, DC. IOS Press. pp. 15-37.

Introduction

The beliefs we held about children's learning are deeply ingrained in our own convictions on what it means to be knowledgeable, intelligent, or literate and what it takes to become so. Whether tacit or explicitly stated, these convictions drive our attitudes and practices as parents, educators, and researchers. If we think, for example, that intelligence is innate and talents are given, we are likely to gear our interventions toward helping learners unfold their gifts; and we may do so at the cost of not giving a chance to those we think of as less "gifted". If we believe, on the other hand, that knowledge or intelligence are a reflection of a child's surrounds, then we may be quick to "pass on" our own truths and values. And we sometimes do so at the cost of ignoring a person's own ways of being, thinking, and relating to the world. And if we believe, as constructivists do, that knowledge is actively constructed through relating to others and transforming the world, then we may tend to step aside and just set the stage for kids to engage in hands-on explorations and creative activities that fuel the constructive process. We may do so at the cost of letting learners "rediscover the wheel" or drift endlessly when shortcuts could be welcome.

Clearly, there is nothing wrong in helping a gifted child unravel her talents, in telling pupils how we see the world, or in offering opportunities for students to discover things by themselves. Yet, the believe in either extreme fixity or malleability of mind can be a formula for disaster when world views are at odds, or when value systems clash. My own life-long interest in constructivism grows out of a personal belief that wherever diversity reigns, the mere transmission of traditional values won't suffice. That's when people(s), young and old, need to find their own paths, speak their voices, and bring their personal and collective experience to the world.

What unifies constructivists across the board, is the notion that children are active builders of their own cognitive tools, as well as of their external realities. In other words, knowledge and the world are both construed and interpreted through action, and mediated through tool- and symbol use. Each gains existence and form through the construction of the other. In Piaget's worlds: "intelligence organizes the world by organizing itself " (Piaget, 1937, p. 311). What's more, knowledge, to constructivists, is not a mere commodity to be transmitted—delivered at one end, encoded, retained, and re-applied at the other. Likewise, the world is not just sitting out there waiting to be uncovered, but gets progressively shaped and reshaped as people interact with it.

Most psychologists and educators of constructivist obedience indeed would agree that learning is less about acquiring information or transmitting existing ideas or values, than it is about individually and collectively imagining and creating a world in which it is worth living. In what follows, I present some aspects of Piaget's constructivist theory, and I contrast them with Papert's constructionism and Vygotsky's socio-constructivism. I highlight what each captures and leaves out, thus setting the stage for my own attempt at integrating the views. I conclude by reframing some of the constructivist/constructionist legacies through the lens of more pragmatic, "situated", and ecological approaches to human learning and development.

The "logic" behind the stages — Piaget, the rationalist

Piaget is best known for his stages, which offer a window into what children are generally up to and capable of at different levels of their cognitive development. While this is an important contribution, there is more to Piaget than his stages. Piaget has shown that children have their own views on things—which differ from those of adults—and that these views are extremely coherent and robust. They are stubborn, if you wish, i.e., not very easy to shake. Children, in other words, are not incomplete adults. Instead, their ways of thinking have a reason to be, and are mostly well suited to their current needs and possibilities. This is not to say that children's belief systems do not change through contact with people and things. The views are continually evolving. Yet knowledge, to Piaget, grows according to complex laws of self-organization, which operate in the background, and the function of which it is to ensure the viability of the organism. Thus, for a child—or an adult—to let go of her current 'theories' requires more than being exposed to a better theory. Conceptual changes in children (Carey, 1987), like theoretical breakthroughs in scientists (Kuhn, 1970), emerge as a result of people's action-in-the-world in conjunction with "hidden" regulatory processes at play to compensate for surface perturbations while, at the same time, not jeopardizing the inner equilibrium of the system as a whole.

From here-and-now to there-and-there…

Piaget's developmental theory emphasizes how children become progressively detached from the world of concrete objects and local contingencies, and gradually able to mentally manipulate symbolic objects within a realm of hypothetical worlds. The focus is on the construction of ever more stable and mobile knowledge structures, through which the growing child interprets and organizes the world, and expands her experiential field. Piaget's empirical studies shed light on the inner and outer conditions under which learners are likely to maintain or change their views of a phenomenon when interacting with it during a significant period of time.

The child that Piaget portrays in his theory is an idealized child (Aries, 1962). Often referred to as an *epistemic subject*, s/he is a representative of the most common way of thinking at a given level of development. And this "common way" is depicted as that of a young scientist mostly driven by the urge to bring some order into a bewildering and exciting world. Piaget's child is a young Robinson in the conquest of an uncharted territory. His conquest is somewhat solitary yet deeply engaging since the explorer himself is curious, inner-driven, and an independent agent. The ultimate goal beyond the journey itself is the joy of mastering the territory under exploration.

In essence, Piaget the rationalist portrays children's intellectual development as a progressive move away from intuitive toward logical thinking, from everyday cognition towards scientific reasoning. In his view, the path leading to higher forms of reasoning, or "formal operations", proceeds from local to general, from context-bound to context-free, from the concrete (tangible) to the abstract (mental). Accordingly, cognitive achievements are gauged against three major acts of distancing. 1. The ability to emerge from here-and-now contingencies, which are characteristic of practical intelligence; 2. The ability to extract knowledge from its 'substrate": i.e., specific uses, appearances, and material properties of things; and 3. The ability to act mentally on virtual worlds, i.e, carry out operations in the head instead of playing them out externally.

Wait wait, don't tell me…

The implications of Piaget's theory for education are profound, even if Piaget didn't think of himself as an educator. Let me mention three lessons that I learned from working with Piaget.

1. Teaching can never be direct, whether we like it or not! Children don't just take in what's being said. Instead, they interpret what they hear in the light of their knowledge and experience. A more radical formulation of lesson 1 would be to say that learning doesn't occur as a result of teaching or, in Piaget's own provocative terms: 'whatever you tell a child, you won't allow her to discover by herself'.

2. Knowledge is not information (a commodity to be delivered at one end and received, unchanged, at the other) but lessons from experience. To equate knowledge with information (using a computational metaphor) confuses matters. As Reddy and Lakoff put it, the "conduit metaphor" of human communication has been over-rated and time has come to move toward what Reddy coins "the tool-maker's paradigm:" a mutual desire to reach shared understanding and negotiate differences through co-creation, and design (Reddy. 1993, Lakoff, 1993).

3. A theory of learning that ignores resistances misses the point, and it is our view that most "misconception" models are doing just that! As we have seen, children have good reasons not to abandon their current views. And this is true no matter how relevant a proposed alternative may be. A good teacher, in this sense, is not a sage on the stage but a guide on the side. She helps learners explore, express, exchange—and ultimately expand— their views from within.

To conclude, while capturing what is common in children's ways of thinking at different developmental stages—and describing how this commonality evolves over time—Piaget’s theory tends to overlook the role of context, uses, and media, as well as the importance of individual preferences, or styles, in human learning and development. That’s where Papert’s “constructionism” comes in handy!

Media Matter— Papert, the intuitionist

If Piaget did not see himself as an educator, Papert, on the other hand, used what Piaget has taught us about children as his basis for rethinking education in the digital age. He coined his theory “constructionism”. In his words, “*Constructionism—the N word as opposed to the V word— shares contructivism’s view of learning as “building knowledge structures” through progressive internalization of actions... It then adds the idea that this happens especially felicitously in a context where the learner is consciously engaged in constructing a public entity, whether it’s a sand castle on the beach or a theory of the universe"* (Papert, 1991, p.1).

To Papert, externalizing—or projecting out—inner feelings and ideas is both the flip side and a pre-requisit to internalizing action. In expressing their thoughts, or giving them form, learners make their ideas tangible and shareable which, in turn, helps shape and sharpen the ideas. Externalizing is a key to communicating and to negotiating meaning. The cycle of self-directed learning, to Papert, is an iterative process by which learners invent for themselves the very tools and mediations that best support the exploration of things they most care about. Because of his focus on learning-through-making at a micro-genetic scale, Papert’s “constructionism” sheds light on how ideas get formed and transformed when expressed through different media, when actualized in particular contexts, and when worked out by individual minds. The emphasis has shifted from stages to styles, and from general laws of development to individuals’ conversation with their own representations, artifacts, or objects-to-think with.

Learning as design—a conversation with artifacts

Stressing the importance of external supports as a means to augment the unaided mind is not new. As will become clear in the next section, Vygotsky has spent his entire life studying the role of cultural mediations as a lever to expand learners' experience, broaden their horizon, and augment their potential. So have many other researchers in the socio-constructivist tradition (Leont'ev, 1932; Wertsch, 1991; Cole, 1996; Gee, 1992, 2004). The difference, as I see it, resides in: 1. The role such external aids are meant to play at higher levels of development. 2. The types of external aid or media studied (Papert focuses on digital media). and lastly, 3. The type of initiative the learner takes in the design of her own objects-to-think with.

More than Piaget, Papert stresses that “diving into” situations rather than looking at them from a distance, that connectedness rather than separation, are powerful means of gaining understanding. Becoming one with a phenomenon under study, in other words, is a key to learning. In Mindstorms, Papert (1980) states: “*A [robotic] turtle has a position and a heading. In this, it is like a person or an animal or a boat (p.55). Children can identify with the turtle and are thus able to bring their knowledge about their bodies and how they move into the work of formal geometry (...) Drawing a circle in turtle geometry is body syntonic in that the circle is firmly related to children’s sense of and knowledge about their own bodies. It is ego syntonic in that it is coherent with children’s sense of themselves* " (p.63).

Papert’s constructionism is more situated than Piaget’s, even if Papert himself doesn’t explicitly use the term when describing his enterprise. One of its contributions is to remind us that intelligence should be defined and studied in-situ; alas, that being intelligent means being grounded, connected, and sensitive to variations in the environment. To Papert, abstract or formal thinking may well be a powerful tool. Yet, it is not necessarily the most appropriate in all situations, and certainly not every one's cup of tea!

The child that Papert is interested in is more relational than Piaget's Robinson. S/he likes to get in touch with people and things. S/he resembles what Sherry Turkle described as a "soft" master (Turkle, 1984). Like Piaget's Robinson, s/he enjoys discovering novelties, yet more than him, she wants to be in the flow of things, and in tune with people. S/he feels at one with them. Like Robinson, she tries out things and is inquisitive. Unlike him, S/he is more of a conversationalist than a builder. She prefers sharing her experience in context, rather than telling what s/he learned in retrospect: She is a *reflective practitioner* in Schoen's sense (Schoen, 1983).

To Papert, like to Schoen, *learning is design* and design[ing] is an iterative process of *mindful concretization*, or materialization of ideas. It is the flipside of, and needed complement to, Piaget's abstractive reflection, or process of ideation. Without it no ideas would live very long!

Implications for education

To Papert, like to Piaget, better learning won't come from finding better ways for the teacher to instruct, but from giving the learner better opportunities to construct. In his view, students best learn when engaged over long periods of time in the construction of personally meaningful products - or products they truly care about. Open-ended design projects, under teacher's guidance, usually offer greater opportunities for students to actively engage, collaborate, and contribute. Creating better opportunities for learners to build has led Papert and his research team at MIT to design a variety of construction materials for children, as well as settings or learning environments in which such materials can be used (Harel and Papert, 1991)

To conclude, while Piaget best described the genesis of children's interests and abilities (ways of thinking) in terms of successive plateaus of equilibrium, Papert is most interested in the dynamics of change. He stresses the fragility of thought during transitional periods. His great contribution, as an educator, is to focus on how people think once their convictions break down, once alternative views sink in, once adjusting, stretching, and expanding current views becomes a necessity. Papert always points toward this fragility, contextuality, and flexibility of knowledge under construction. A strong believer in the idea that mistakes are a key to learning, and that especially children are experts at using the little the know as a lever to grow (learning to learn), Papert has spent much of his life creating technology-enhanced environments, or microworlds, in which learners are invited to mess around with otherwise risky ideas, on safe ground.

Born to bond—Vygotsky, the "socialite"

At the heart of Vygotsky's socio-constructivism lays a simple idea. From the day they are born, human infants learn, thrive, and grow in connection with others. We "are" because we relate. The theory stresses the importance of caring and knowledgeable adults and peers; and emphasizes the role of language and other cultural mediations as motors in human learning and development. In spite of his focus on culture as a *teaching tool*, Vygotsky sees a child's intellectual development as a constructive process. This is why, in our view, his socio-cultural approach is not fundamentally at odds with Piaget, Papert, Bruner, and other constructivists.

To Vygotsky, as well as activity- and socio-cultural theorists, the "social" has a primacy over the "individual" in a very special and important sense: Society is the bearer of a cultural heritage without which the development of individuals is simply unthinkable. Parents and other members of a social group form a living habitat to which all contribute yet that also reflects —and to an extent embodies— its own history and thus impacts, however indirectly, the generations to come. In other words, our cultural heritage is at once a niche and a medium. It is at once a "terrain," or stage for human experience, and a lens, or interpretive frame, at the disposal of the terrain's inhabitants.

Vygotsky introduced the concept of psychological tool to capture the idea that the cultural artifacts at our avail become part of our own "psychology" once we appropriate them. Psychological tools include: various systems for counting; mnemonic techniques; algebraic symbol systems; works of art; writing; schemes, diagrams, maps, and technical drawings; all sorts of conventional signs, and so on. (Vygotsky, 1982:137, cited in Cole & Wertsch, 1996).

It takes a village to raise a child

Piaget and Papert no doubt would agree that co-operating with others is co-constitutive of operating on one's own. Yet, Vygotsky put greater emphasis on how the presence of caring adults and peers can both cater and “speed up” a child’s self-directed learning, and how cultural artifacts are used, from the outset, to help mediate this process.

To Vygotsky, a person’s cognitive development proceeds outside-in, i.e., from other to self: “Every function in the child's development appears twice: first, on the social level, and later on the individual level; first, between people, and then inside the child” (Vygotsky, 1978:57 in Lock, 1989). Inter-personal relations, to him, are the precursors, and necessary conditions, for the emergence of individual/intra-mental processes: Youngsters first share their experience with others, before they become able to master and understand them for themselves. Their development proceeds from socio-centric to egocentric.

Vygotsky’s child, as I see it, may be more of an intelligent *trusting disciple*, in Harris’s sense (Harris, 2000), than an *autonomous agent* (Papert, 1980). While curious, active, inner-driven, and autonomous, s/he also trusts that others, more experienced, may tell her things that she cannot yet understand or maybe won't be able to experience directly. In other words, s/he knows that she can learn vicariously by listening to what others have to say about what interests her. The autonomous agent, in contrast, is not comfortable if he cannot check out for himself what others propose, at the cost – sometimes – of re-inventing the wheel! or forgetting that others have come before or even inspired her!

Implications for education

One of the key concepts in Vygotsky’s theory is the notion of "zone of proximal development" (ZPD). Much quoted and often misunderstood, the ZPD defines a potential area of expansion for individuals to overcome their limits, provided the social environment ”pitches in”. In other words, the zone of proximal development tells us “how far” individuals can stretch the envelope of what they know, when supported and guided by others. It is, again, through social interaction, that learners can mobilize, and best use, the resources at their avail.

To conclude, Vygotsky’s socio-cultural theory can be read as being in line or at odds with Piaget and Papert's contributions depending how much agency and autonomy we think each theory lends to individuals (children and adults) and/or their groups of affiliation (social actors) to re-invent their lives. In this respect, the term *enculturation*, used by socio-culturalists continues to feed polemics among scholars and may be usefully substituted by the twin-notions of object-[or environmental] affordances and personal appropriation. Likewise, the chicken-and-egg duality of inside-out versus outside-in seems less interesting than notion of co-evolution (present in Vygotsky and further developed by activity theorists): The idea that, from the outset, child, care-takers, and objects or tools form a mutually enriching triad that cannot be broken down.

Moving between worlds—Integrating the views

In The Evolving Self, Kegan defines human development as a lifelong attempt to resolve the unsolvable tension between *being embedded* in situations and *emerging from embeddedness* (Kegan, 1982). In a similar way, a person's cognitive or affective growth can be seen as a lifelong attempt to find a viable balance between fusion and separation, openness and closure, or in Piaget’s words, between *assimilation* and *accommodation*. Said otherwise, imposing one’s order upon things and looking at the unknown in terms of the familiar (assimilation) goes hand in hand with being sensitive to variations in the environment and letting go of previously held believes (accommodation). Any unbalance in favor of one or the other pole can lead to evolutionary unstable strategies, i.e, less viable ways of regulating exchanges with others and things, and compensating for surface perturbations. Our own attempts at integrating the views and coming to grips with the multiple legacies from the three forefathers can be summarized as follows:

- Along with Papert, we suggest that diving into the unknown, at the cost of experiencing a momentary sense of loss, is a crucial part of learning. Without immersion there is no empathy, and without empathy there is no way to feel for others, or grasp a situation from-within.

- Along with Piaget, we view separateness through progressive de-centering as a necessary step toward relating ever more intimately and sensitively to both people and things. In any situation, no matter how engaging, there comes a time when we need to remove ourselves and look at things from a distance. To advocate the importance of separateness does not preclude the value of being embedded in one's experience. It only suggests not to get locked in it forever.

- What Vygotsky adds to the equation is the notion that no human can be or grow without the presence and support of other people. Children are bound to their cultures because the people, places, and tools they interact with form the *holding structure* out of which they grow: their intelligence is collective because we are all in it together, whether by choice or by necessity!

Only when a learner actually moves in, around, and between worlds, by adopting different perspectives, or putting on different lenses, can a dialogue begin between initially fragmented, or partial, views. Indeed, how could anyone learn from experience as long as they are totally immersed or forever distant? Likewise, how could anyone know who they are (and what they are worth) if they are not “held” by others? As mentioned before, there are times when pushing back and extracting oneself from the deep waters becomes a necessity. And that's when a new cycle can begin, and the stage is set for new and deeper connectedness and understanding.

Making sense of the legacies

People spend a great deal of their time carving out their niches –virtual and physical– so that they fit their needs, support their purposes, and augment their potential. They build cities and homes, they invent computers and airplanes, and they create alphabets and geometries. People are also busy keeping track of their experience and leaving traces behind. They mark their grounds and they use the traces they leave behind as anchors to orient themselves. Newcomers to a culture are left to live with the marks traced by others.

In addition to being world-makers and leaving traces behind, people are also devoting much of their time ‘reading’ meaning into existing forms, be they their own or those produced by other. And they do so in creative ways. Readers, in other words, are in no ways passive consumers. Instead, they engage designed artifacts by reconstructing them through the lens of their interests and experiences. As Bordwell points out about film audiences: “The artwork sets limits on what the spectator does. But within these limits, the viewer literally recasts the play” (Bordwell, 1986, p. 30). Viewers impose their order by rearranging or replacing clues, by filling in blanks or ‘creating phantoms’, by ignoring clues, and by forcing causal-temporal connections. In Piaget’s parlance, they assimilate incoming signals (in this case, a narrative, which they interpret through the lens of previously constructed experience), and they accommodate their views only insofar some unexpected puzzlements or surprises are called upon by the materials.

- In his work, Piaget has extensively written about intelligence as adaptation, and adaptation as a viable balance between accommodation and assimilation. It is our contention, however, that Piaget, as a thinker, was himself more of an "assimilator" than an "accommodator". Hence his interest in children as "assimilators", or world-makers!

- Papert in contrast, has always shown a personal *penchant* for the meanderings of individual minds in context, especially as they navigate their ways through uncharted territory. His interest is in the navigator’s abilities to deal with unexpected obstacles, as she moves along. Hence his interest in children as "accommodators," and world-dwellers (one may say world-travelers)!

- Vygotsky, for his part, liked to think of youngsters and their groups of affiliation as intelligent listeners and creative team-players. His dilemma, as a scholar, was to reconcile the processes of individuation and enculturation, and to open up spaces for children and their caretakers to grow as autonomous agent and responsible citizen. Hence his interest in collective intelligence!

From interaction to co-evolution

As learners outgrow some of their previously held beliefs, they sometimes forget that "what they know depends on how came to know it!" (Watzlawick,1984, p.9). They then act *as-if* their construed "realities" had always been out there waiting to be uncovered, and they rely upon them and refer to them as tangible and shareable entities. While this cognitive amnesia poses a problem to constructivists, it holds the advantage of sharpening our sensitivity to "their" qualities independent of our immediate relation with them. Treating others and our own creations *as-if* they had an existence beyond our rapports with them (even if we know that we cannot know their whereabouts) and "celebrating them for what they are" (at the risk of over-interpreting)—is ultimately a viable mental attribution, provided we remember that the attribution itself is a construct. Its function is to elevate human transactions (between me/not me) beyond blind projections, or assimilation pure, with its unfortunate consequence: reducing anything that is other to a mirror-of-self (over-assimilation).

From a pragmatic-ecological standpoint, it seems essential for designers and educators to take responsibility for their offerings by not assuming –I caricature the constructivist's stance– that learners will use them as Rorschach stains anyway. Designers should acknowledge that their products will survive after them, and that it is ultimately the built artifact, rather than the builder's intentions, that becomes part of other people's cultural heritage. It is 'its' qualities that will persist and signal potential uses to newcomers who encounter it for the first time.

Clearly, designers cannot predict or be accountable for how their creations will be appropriated by others. What designers can, however, is be attentive to the idea that, once conceived, their creations are no longer a mere extension of themselves. Instead, they come to exist as separate entities, and an integral part of the cultural landscape in which other newcomers will live and grow. People read into artifacts because of who they are, but also because artifacts offer clues. Like archeological sites or eroded landscapes, they are marked by—and in this sense embody—the knowledge or collective experience that went into their being. The constructivist's nightmare may well come true! Yes, human-made artifacts can call upon certain experiences and uses, and discourage others. And they sometime impose their logic, much in the same way as a partner or conversationalist does. To deny the power of places and things to impact people can bread a culture of 'not caring'.

From an epistemological standpoint, it seems important for learning researchers and educators to rethink the role of accommodation in cognitive adaptation. To Piaget, we have seen, intelligence is adaptation, and adaptation is the ability to maintain the maximum of what is acquired while opening up to the maximum of novelty. In his words: "*Assimilation is by its very nature conservative, in the sense that its primary function is to make the unfamiliar familiar, to reduce the new to the old*" (Piaget, 1954, p. 352-353). Accommodation, by contrast, de-crystallizes existing schemes so that they fit our expectations. Its primary function is to make what is familiar unfamiliar again, and to question the old by listening to the new. Question is: what would Piaget's legacy be had he paid closer attention to the opportunistic nature of minds in context.

It is not exaggerated to say that human beings at once create their worlds, inhabit their creations, and become "inhabited" by them. In their imagination, fusion (becoming one) and separation (removing oneself) coexist, and both contribute to their personal and cognitive development. It is the dance that matters.

References

Ackermann. E. (2004) *Constructing knowledge and transforming the world.* In A learning zone of one's own: [M. Tokoro and L.Steels (Eds.). Oxford, Washington, DC. IOS Press. pp. 15-37

Ackermann, E. (1996) *Perspective-taking and object construction: Two keys to learning*. In Constructionism in Practice: Designing, Thinking, and learning in a Digital World (J. Kafai, & M. Resnick, Eds.). Mahwah, NJ: Lawrence Erlbaun, Publishers, pp. 25-37

Aries, P. (1962) *Centuries of childhood: A social history of family life*. New york: Vintage Books.

Bordwell, D. (1985) *The viewer's activity. In Narration in the fiction film*. Madison:University of Wisconsin Press, pp. 29-62

Carey, S. (1987). *Conceptual Change in Childhood*. Cambridge, MA: MIT Press.

Cole, M. (1996) *Culture in mind.* Cambridge, MA: Harvard University Press.

Gee, J. (1992) *The social mind: Language, ideology, and social practice*. Series in language and ideology. New York: Bergin & Garvey

Gee, J (2004) *Situated language and learning: A critique of traditional schooling*. London: Routledge

Harel, I. and Papert, S. (Eds) (1991). *Constructionism*. Norwood, NJ. Ablex Publishing Co.

Harris, P. (2000). The work of the imagination. Hoboken, NJ: Wiley Blackwell

Kegan, R. (1982) The Evolving Self. Cambridge, MA: Harvard University Press.

Kuhn, T. (1970) The structure of scientific revolutions. Chicago: University of Chicago Press.

Lakoff, G. (1993) *The contemporary theory of metaphor. In Metaphors and thought (Ed. A. Ortony). Cambridge University Press, pp. 202-252.*

Leont'ev, A.N. (1932) *Studies on the cultural development of the child*. In Journal of genetic psychology, 40, 52-83.

Papert, S. (1980) *Mindstorms: Children, Computers and Powerful Ideas*. NewYork: Basic books.

Piaget, J. (1977) *The Essential Piaget* (Gruber H. & Voneche, J. Eds). New York: Basic Books.

Piaget, J. (1954) *The origin of intelligence in children.* New York: International University Press.

Piaget, J. (1975) *L'equilibration des structires cognitives*. Paris: PUF

Reddy, M. (1993) *The conduit metaphor: A case of frame conflict in our language about language*. In Metaphor and thought (Ortony A, Ed.). Cambridge University Press, pp. .164-202.

Schön, D. (1983). *The Reflective practitioner: How professionals think in action*. New York: BasicBooks.

Suchman, L. (1987) *Plans and Situated Actions.* Cambridge, MA: Cambridge University Press.

Turkle, S. (1984) *The Second Self: Computers and the Human Spirit*. NY: Simon and Schuster.

Turkle, S. (1995) *Life on the Screen*. New York: Simon and Schuster.

Vygotsky, L.S. (1962) *Thought and Language*. Cambridge, MA: MIT Press.

Vygotsky, L.S. (1978) *Mind in Society*. Cambridge, MA: Harvard University Press

Watzlawick, P. (1984). *The invented reality*. New York: Norton.

Wertsch, J. (1991) *Voices of the Mind: A Socio-cultural approach to mediated action*. Cambridge, MA: Harvard University Press.

BULGARIAN MULTIMEDIA SIGN LANGUAGE DICTIONARY FOR CHILDREN

Professor PhD Tzanka Popzlateva, *popzlatevatz@abv.bg*
PhD Student Boyanna Aleksova, *boyana1980@abv.bg*
Assistant Professor Anna Trosheva, *trosheva@mail.bg*
Dept of Special education, Sofia University "St. Kliment Ohridski"

Ivailo Ivanov, *iivanov@fmi.uni-sofia.bg*
Assistant Professor Temenujka Zafirova-Malcheva, *tzafirova@fmi.uni-sofia.bg*
Oleg Konstantinov, *oleg@fmi.uni-sofia.bg*
Dept of Mathematics and Informatics, Sofia University "St. Kliment Ohridski"

Abstract

This report describes the problems in the education of deaf children and the role of sign language in communication and forming their own speech. The paper present the role of advanced computer technology and various multimedia tools in the learning process of children with impaired hearing, and the need to integrate these tools and resources in the process of learning. In this connection, the report gives main attention to *Bulgarian Multimedia Sign Language Dictionary for Children*, developed by a team of University of Sofia. This dictionary is designed specifically for children and combines the advantages of modern multimedia technologies and teaching methods to master sign language, which makes it a powerful educational tool.

Multimedia sign dictionary for children contains 2300 vocabulary units of child language. Each unit is represented by an interactive image, sign, one-handed dactyl, written and auditory modality. Special attention is paid to alternative versions of the Bulgarian alphabet – one-handed and two-handed dactyl and alphabet in pictures. Access to its content is provided by alphabetic, character and thematic catalogues and search tool for actual lexical unit.

In alphabetical catalogue for each letter of the alphabet is given a set of lexical units, beginning with this letter. Each lexical unit from the list is illustrated by a picture, video of the gesture, one-handed dactyl displayed the word letter by letter, pronunciation of the word represented by a sound file and games with the word.

In the thematic catalogue lexical items are grouped by themes, each of which is represented by the same attributes as in the alphabetic catalogue.

In character catalogue the letters of the alphabet are represented by a picture of object beginning with this letter, one-handed and two-handed dactyl, and touch gestures of the letter.

The dictionary has very good navigational structure it provides an opportunity to return to the main menu, return step back and exit the program at any time of work. Moreover, at each step in the process of work the user has access to help information.

Multimedia model of sign language dictionary allows creative use and development of new products to support language training and rehabilitation of children with atypical developmental.

Keywords

Deaf children, children with impaired hearing, sign language, multimedia sign dictionary, special needs education, Imagine.

Sign language for deaf people

Over the last decade sign language for deaf people has established itself as an alternative linguistic system that is comparable in capacity to meet the communicative, cognitive and personal needs with the auditory-oral language.

The problem with obsessing of language skills is particularly topical in congenital or early acquired deafness. Scientific studies show that deaf children of deaf parents (5-10% of the total population) learn sign language in a comparable mastery of oral language patterns. These children demonstrate significantly higher achievement in the management of oral and written language, higher level of general intellectual development and psycho-social functioning compared with deaf children in hearing families who do not have adequate communication tools. The compensatory potentials of this alternative language are already recognized, and their implementation still in preschool and early rehabilitation, in the communication of parents and children and in the use of general educational content.

Although the sign language of deaf people is recognized as their mother tongue, lack of literacy complicates its assimilation from the hearing people (parents, professionals and peers). Bulgaria is one of the first countries where more than 10 years has created an official (literary) sign language with 5 000 vocabulary units (Union of the Deaf in Bulgaria). In 2006, this dictionary has been expanded and reprinted. Unfortunately, it has not reported the specifics of child language, making it ineffective for purposes of rehabilitation and education.

Group of deaf children is heterogeneous and individual approach to them would indicate the presence of different opportunities to meet their special needs and abilities, especially in terms of integrated training and rehabilitation. The need to integrate alternative linguistic systems in children with various developmental disorders becomes increasingly inescapable fact.

This supposes serious research on both the vocabulary used in language development, language therapy and interpersonal communication and the ability to integrate advanced multimedia and computer technology for video and sound recordings and interactivity in the operation with symbolic characters presented in different language modalities.

Multimedia sign dictionary for children

Stages of development

Multimedia sign dictionary for children consists of 2300 vocabulary units of child language based on content analysis of existing image, gestures and language dictionaries for children (paper and software), programs and educational materials approved by the Ministry of education for the comprehensive and special schools.

The process of product development is realized on several major stages:

- Determination of missing in official Bulgarian sign language analogues of verbal units and study of their equivalents in the speech of deaf children with deaf parents and deaf adults;
- Video recording of 2 300 gestures demonstrated by 17 students from special schools in Sofia, Plovdiv and Targovishte, in accordance with the approved principles of Union of the Deaf in Bulgaria in formation of a sign;
- Video recording of hand alphabet – one-handed and two-handed alphabets (dactyl);
- Create a multimedia design of sign dictionary, which combines written, oral, dactyl, image and gestures sign presentation of each lexical unit;
- Program implementation using programming environment *Imagine*.

Software structure

The Multimedia sign dictionary is stored on DVD. It includes about 2300 lexical units. Access to its content is provided by alphabetic, character and thematic catalogues and search tool for actual lexical unit (Figure 1, Figure 2, Figure 3, Figure 4). Emphasis has placed on the therapeutic possibilities of the dictionary. Each unit is represented by an interactive image, sign, one-handed dactyl, written and auditory modality. Special attention is paid to alternative versions of the Bulgarian alphabet – one-handed and two-handed dactyl and alphabet in pictures.

Figure 1 Initial screen of Multimedia sign dictionary for children

Figure 2 Alphabetic catalogue

Figure 3 Thematic catalogue

Figure 4 Character catalogue

In alphabetical catalogue for each letter of the alphabet is given a set of lexical units, beginning with this letter (Figure 5).

Figure 5 Alphabetic catalogue – letter “A”

Each lexical unit from the list is illustrated by a:

- picture (Figure 6);
- video of the gesture demonstrated by the student comes from a special school (Figure 7);

- one-handed dactyl displayed the word letter by letter (Figure 8);
- pronunciation of the word represented by a sound file;
- games with the word.

Each of these options can be selected via a buttons located on the left side of the screen. The actual presentation of the word in appropriately way is displayed in the main part of the screen. At the bottom of screen has a bar with one-handed dactyl of letters constructing the word, the choice of one of them started in the main part of the screen video with one-handed dactyl of the corresponding letter.

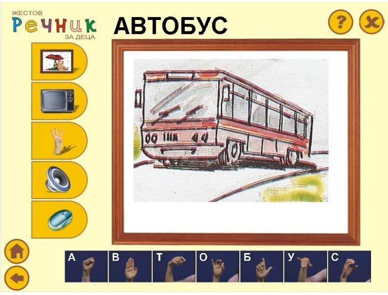


Figure 6 Picture

Figure 7 Video gesture

Figure 8 One-handed dactyl

In the thematic catalogue lexical items are grouped by themes, each of which is represented by the same attributes as in the alphabetic catalogue. There are differentiated seven main topics. They correspond to one of the widely implemented in Bulgaria training programs for preschool age which is developed by a scientific team from the Sofia University (Popzlateva, 1994). The topics are:

- *My World*, which includes words related to the description of body parts, toys, home, food and more (Figure 9);
- *Me and others* – words associated with joint activities in the family, kindergarten, friends and others (Figure 10);
- *Me and Nature* – animals, plants, fruit, vegetables, flowers, seasons, natural phenomena, etc. (Figure 11);
- *I celebrate* - birthdays, Christmas, New Year, Mom's Day, Easter, Holidays and more (Figure 13);
- *The world around me* – a city, village, vehicles, professions and other (Figure 15);

In the world of fairy tales - fairy, ethical rules, human qualities and other (Figure 14); *I learn and communicate* – learning activities and materials, tools and standards for communication, greetings and more (Figure 12).

Figure 9 My World

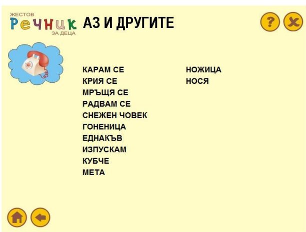


Figure 10 Me and others

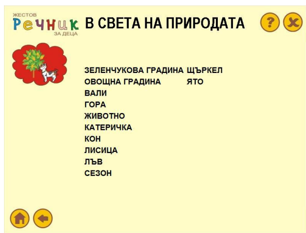


Figure 11 Me and Nature

Figure 12 I learn to communicate

Figure 13 I celebrate

Figure 14 In the world of fairy tales

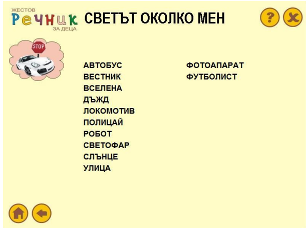


Figure 15 The world around me

Character catalogue of the dictionary is available by clicking the button "alphabet" from the initial screen. There the letters of the alphabet are represented by a picture of object beginning with this letter, one-handed and two-handed dactyl (Figure 16, Figure 17, Figure 18).

Figure 16 Picture

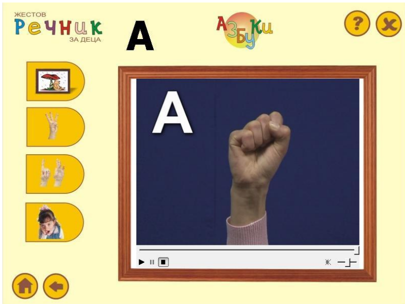


Figure 17 One-handed dactyl

Figure 18 Two-handed dactyl

The dictionary provides an opportunity to find a specific word with a specially designed search tool (Figure 19).

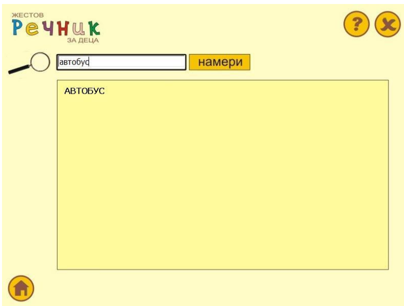


Figure 19 Search tool

In order to master the individual lexical units a separate module with gaming was developed. This module is for developing language skills and basic mental operations through game situations, which are differentiated by several criteria. The games for developing language skills are related between sound, label, sign and its picture. The games for the development of the basic mental operations are meant to identify similarities and differences, setting ratios (at least, most), according to an object or group of objects and environments.

Several types of games that differ in what must be pointed – a gesture or a picture, are developed.

In games to select the correct gesture child may be placed in three different situations - choose one of the displayed gestures corresponding to the showed picture (Figure 20), choose the gesture corresponding to the written word (Figure 21) or select the gesture corresponding to the word shown by one-handed dactyl (Figure 22).

In games to select the right image there are two situations – to choose one of the displayed pictures corresponding to the shown video sign (Figure 23) or selected picture corresponding to the word shown by one-handed dactyl (Figure 24).

Figure 20 Game show the sign – picture

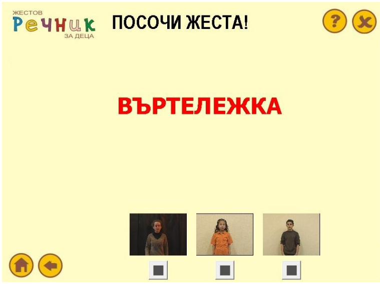


Figure 21 Game show the gesture – word

Figure 22 Game show the gesture – one-handed dactyl

Figure 23 Game show the picture – video gesture

Figure 24 Game show the picture – one-handed dactyl

Games are used to develop teaching module through which to manage their content and degree of difficulty. This allows developing a systematic program for every deaf child, according to his individual special educational needs.

From the methodological point of view, the use of the contents of the dictionary is recommended and should be done through teamwork between the child (children) and teacher (parent). This is of particular importance in the introduction to the letters and words of the thematic catalogs. In it the teacher should draw the attention of the child to the particularities of each gesture and its use. The teacher should ask the child several times repeated gestures shown on screen to assure the proper performance of his child. If this is not done, the student will remain only with the knowledge to understand the language, gestures, but not its use for expression. Two to three new gestures are necessary to make a return to a single gesture, which has already been seen. It requires implementation of the child and then to see the screen video.

When working with the games, the child can be left alone to determine to what extent that he/she understands the gestures. The screen shows only the correct answers and the total number of exercises that are played. At the end of each game, for developing language skills, the teacher can review a list of words that were misspelled by the child. This provides the teacher (parent) feedback on the gaps in language gestures.

The dictionary has very good navigational structure it provides an opportunity to return to the main menu, return step back and exit the program at any time of work. This is done by using the button "Menu", "Back" and "Exit". Moreover, at each step in the process of work the user has access to help information.

Technical characteristics of the captured gestures are based on the HD (High Definition) modern standards. All the requirements of professional cinematography and video editing work - quality lighting, digital media (DV CAM with High Definition recording), suitable optics, wireless microphones of high class, and use of advanced video and audio processing programs are met. The shooting of the individual signs units is made with children of different ages, which further complicates work.

The dictionary is designed to meet the special needs of preschool children with developmental problems and language communication. The modern information technologies, various media types and dedicated rehabilitation provide an effective access for parents and professionals to alternative means of communication and the enrichment of children's linguistic competence in daily life. The dictionary can be used both in special and integrated units of rehabilitation. Multimedia model of sign language dictionary allows creative use and development of new products to support language training and rehabilitation of children with developmental disorders.

Acknowledgement

The work presented in this paper was supported within the Research Project of Sofia University, Grand agreement №219 - 2009/2010 and №152 – 2010/2011 „Integration of the Sign Language in Psychosocial Rehabilitation of Atypical Children”

The work presented in this paper was partially supported within the project BG 051PO001-3.3.04/13 of the HR Development OP of the European Social Fund 2007-2013.

This paper was partially supported within the SISTER (Strengthening the IST Research Capacity of Sofia University, Grant agreement No 205030) project, funded by European Commission’s 7th Framework Programme FP7-REGPOT-2007-1.

References

Panev, V. (1995) Bulgarian sign language dictionary SGB, Sofia.

Bulgarian sign language dictionary Part II (2007), SGB, Sofia

Baltadjieva, M. Mitzeva (1994-1996). Textbook on speech development of schools for deaf children. Education, S.

Popzlateva, Tz. (2004) *Sign language – a unique natural experiment in neuropsychology.* - Special Education, № 4.

Popzlateva, Tz., Baltadjieva, A., Mitzeva, M. (1994). *Second grade reader schools for deaf children*. Education, S.

Popzlateva Tz., Baltadjieva, A., Mitzeva, M. (1996). *Third grade reader for schools for deaf children*. Education, S.

Roussinova, E., Gurov, D., Baeva, M., Gurova, V. (1993-1996). *Book for the child*. S.

Popzlateva Tz (2010). The Child Deafness – neurocognitive and neurolinguistic profiles. SU “St.Kl. Ohridsky” (in print).

Programming playfully for a real-life problem: conditional statements on the stage of Scratch

Rene Alimisi, *ralimisi@ioe.ac.uk*
Department of Preschool Education, University of Thessaly

Niall Winters, *n.winters@ioe.ac.uk*
London Knowledge Lab, Institute of Education, University of London

Abstract

The literature has shown that students have problems with certain programming concepts, including the concept of variable (Doukakis *et al*, 2007), conditional statements, repeat structures (DuBulay *et al*, 1989; Putman *et al*, 1989; Soloway and Spohrer, 1989). In this study, we present the design of an activity based upon the 'Kindergarten approach to learning' (Resnick, 2007b) and the 'thick view of authenticity' (Shaffer and Resnick, 1999) using Scratch programming environment. The study aims at exploring 12-13 years old students' understanding of conditional statements. A 'thick view of authenticity' is taken by design a learning activity that is based on a real- life scenario: the functionality of a lift. The ways in which the activity design supported students' understanding and the role that Scratch played in this are explored and discussed. The study found that the four novice programmers, who took part in the study, gained a good level of understanding of the way in which the conditional statements function. The paper details the importance of the interrelation of the real- life scenario and the functionality of Scratch programming environment in supporting the playful exploration of programming solutions and the learning process.

Keywords

Constructionism, Kindergarten approach, real life scenario, programming, playful learning, Scratch

Introduction

Programming is 'one of the most widely practiced instructional activities' (Lee and Lehrer, 1987), providing potential learning opportunities for students (Papert, 1993). Papert (1993, p.27) advocates that programming encourages students to explore, reflect upon and develop their own 'style of thinking'. Kahn (2004, n.p), taking into account Papert's work as well as other important studies in the area of computer programming, parallels the process of programming to 'a fertile ground for learning general thinking skills' such as 'problem decomposition, component composition, explicit representation, abstraction, debugging and thinking about thinking'.

However, engaging students in introductory programming is not straightforward (Guzdial, 2003). Studies have shown that novice programmers face difficulties in understanding basic programming structures (Doukakis *et al*, 2007; Soloway and Spohrer, 1989). As Pea (1986, p.25) posits, 'students have such pervasive conceptual misunderstandings as novice programmers that correct programs early in the learning process come as pleasant surprises'.

According to Doukakis *et al* (2007), 'conditional statements' are among the programming concepts that cause difficulties to students. A conditional statement is a structure comprising of commands. These commands will be executed upon evaluation of a TRUE/FALSE condition. If the specified condition is TRUE, a set of commands will be executed. Otherwise, if the specified condition is FALSE, another set of commands, possibly embodied in the 'else' part of the structure will be executed. For instance, if the *condition* evaluates to true, *statement_1* is executed; *statement_2* is executed only in case the *condition* evaluates to false (see figure 1).

```
if (condition)
   statement_1
[else
   statement_2]
```

Figure 1. Conditional statement

Taking into account that conditional statements are a fundamental concept in programming languages, it is important to look behind novice programmers' difficulties. Complex programming environments and traditional programming languages that are usually used for introductory programming courses (Pea, 1986) are often laid behind students' difficulties and confusion. DuBulay *et al* (1989), identify simplicity and visibility as crucial characteristics for programming languages for novices. The kind of the problems, that students are called to program for, plays also a significant role towards their understanding. Commonly, students are called to work on and to program for mathematical problems disassociated with real life situations (Tzimogiannis, 2005). However, it is of great importance to allow students to work on computational artefacts that are meaningful to them (Guzdial, 2003). Given that students often say that they feel fascinated to work with real-world problems (Mims, 2003), it is worth providing them with the opportunities to program for such problems.

This research focuses on novices' understanding of the programming concept of 'conditional statements'. It aims to engage students in a playful learning experience that draws upon the Kindergarten approach to learning and the thick view of authenticity (Resnick, 2007b; Shaffer and Resnick, 1999) using Scratch. A 'thick view of authenticity' is taken by design a learning activity that is based on a real- life scenario: the functionality of a lift.

This paper addresses two main issues. First, the learning activity was designed as a real life scenario; the way in which the use of the real life scenario supported students' understanding of the programming concept of conditional statement, is examined. Second, the way the programming learning environment of Scratch supported students' understanding of the programming concept of conditional statement is discussed.

Theoretical framework

Resnick (2008) claims that we should strive for a 'Creative Society'. The vision of the Creative Society lays emphasis on the ability to think creatively and such ability constitutes the key to success at both a personal and professional level (Resnick, 2007b). The design and development of new things are seen to play a fundamental role in the concept of creativity and to encourage the revising of the models that are already in one's mind. Two pedagogical approaches closely associated with the idea of creative thinking and capable of meeting the needs of the current society are the 'Kindergarten approach to learning' and the 'thick view of authenticity'.

The Kindergarten Approach to learning is based on the 'creative thinking spiral' (see figure 2) introduced by Resnick (2007a, p.18). The so- called 'creative thinking spiral' (Resnick, 2007a, p.18) is used to describe a process in which children '*imagine* what they want to do, *create* a project based on their ideas, *play* with their creations, *share* their ideas and creations with others, *reflect* on their experiences – all of which leads them to *imagine* new ideas and new projects'. The engagement in such a process is seen to encourage the development of creative thinking skills and purportedly to form the fundamental steps towards the Creative Society; or in Resnick's (2007a) words 'to sow the seeds for a more creative society'. Resnick aims at providing students with opportunities to learn through designing, creating, inventing and reflecting, and such an aim seems to have its roots in the Papertian 'powerful ideas'. (Resnick, 2008; Resnick and Silverman, 2005; Papert, 1993, p.4) To Papert these ideas 'can be used as tools to think with over a lifetime' and provide the leverage to help students 'make sense of the world' (Papert, 1980 cited in Resnick and Silverman, 2005, n.p).

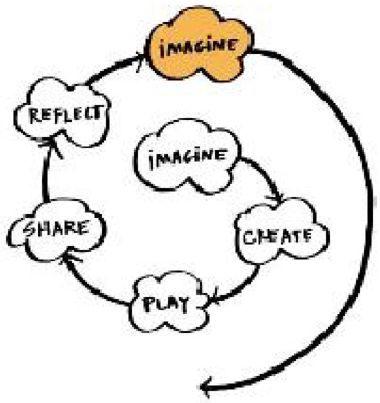


Figure 2. The creative thinking spiral (Resnick, 2007b)

Shaffer and Resnick (1999, p.195) introduce the 'thick view of authenticity', according to which authentic learning is personally meaningful to the learner (*personal authenticity*), is closely associated with the world outside the classroom (*real world authenticity*), offers the opportunity to 'think in the modes of a particular discipline' (*discipline authenticity*) and embodies means of assessment which reflect the learning process (*authentic assessment*). This learning approach encourages students to experiment with knowledge in context and allows them to make connections with the real world (Shaffer and Resnick, 1999). Taking into account the fact that students often say that they feel motivated to work with real-world problems (Mims, 2003), the 'thick view of authenticity' can be seen as a way of engaging students in meaningful activities.

The idea underpinning this study is to engage students with the programming concept of conditional statements though a framework that would allow real- life connections and a creative process as it is described through the 'creative thinking spiral' to occur. Moreover, it is critical to provide students with the opportunity to work on a programming environment which eliminates the complexity of traditional programming environments and supports the engagement with meaningful activities.

Scratch, 'a networked, media rich programming environment' built upon Logo and created by 'Lifelong Kindergarten Group at Mit Media Laboratory in collaboration with Yasmin Kafai's group at UCLA' (Maloney *et al*, 2008, p.367) was chosen as was seen as capable of engaging students in the different stages of the 'creative thinking spiral' and supporting the thick view of authenticity, which have both been seen to play a key role in the development of students as creative thinkers. Apart from primary reasons, pragmatic ones shaped also this decision. Firstly, programming in Scratch is simplified without degrading the mental process which underpins it. Secondly, the way in which blocks are joined together, eliminates the possibilities for syntactic errors to occur. Thus, students are allowed to focus on their projects without spending a considerable amount of time on syntactic issues.

Methodology

The research was carried out in a secondary school in South London that is specialist in Mathematics and Computing. It was gratifying to know that the ideas underpinning this study could be useful for the school and that future findings could be utilised towards meeting its needs. In particular, the attention paid (see figure 3) in the exploitation of real- life scenarios in the classroom as well as the focus on innovative Information Communication Technologies applications, made the school ideally suited to the purposes of the study.

> It is our intention to provide active lessons allowing students to not only learn different computer skills, but also understand how these skills can be used in other subjects and learning areas throughout the school.
> Practical "real life" scenarios are often used to highlight how these lifelong skills can be used and adapted for the workplace.
> Thursday, 30 April 2009 10:55

Figure 3. Screenshot from school's webpage

The project follows a case study approach and occurs in four stages. The first stage employs questionnaires and involves informal discussions which aim to outline students' pre-programming experience and to identify the four novice programmers, participants of the study. The four 12 to 13 years old students that were selected (Kevin, Timothy, Luke and Billy) shared two common characteristics. First, all of them had not been taught any programming language either during school or out of school hours. Second, all of them had previous experience in designing games and mainly animations by making use of particular software applications such as Pivot, Power Point and GameMaker. The question that arose was whether through Pivot, Power Point or GameMaker the students had been involved with programming concepts. Interestingly, it became clear during the informal discussion stage that their experience was not in any way associated with programming concepts and in particular with the programming concept of conditional statements. In fact, students' applications did not demand the use of significant programming concepts, such as conditional constructs. In addition, it is worth mentioning that the creation of the games and the animations occurred through a practice based on 'dragging and dropping' or 'clicking and selecting' pre- programmed behaviours. This conclusion was reached through the students' statements and confirmation from the class ICT teacher.

> *'I moved the person and I just clicked a button and saved it; and then I clicked another button; and it played it and made it move slowly' (Kevin on 'how he made an animated stickman on Pivot')*

The second stage consisted of a number of familiarization activities through which the four participants would have the opportunity to engage with Scratch at a level where they could carry out the main activity. The programming concepts and commands that were necessary for the accomplishment of the main activity constituted the target- content during the familiarization stage. The different programming concepts that would be employed in the main activity were introduced as ways to 'breathe life into the sprites'. Thus, blocks which resulted in making sprites move or display messages or reproduce sounds were presented first. Then, the introduction to the programming mechanisms for synchronizing and controlling the sprites followed. The programming concepts were introduced in the framework of a simple task (the movement of a sprite) which included several steps. In the framework of the familiarization stage, we worked with all the four students on one laptop. However, students were often moved to practice the newly introduced programming concepts individually.

The main activity is carried out in the third stage (for more details see the subsection below). The method of observation was used for gathering data during students' engagement in the main activity. During the stage of observation field notes were recorded. A flip camera and a digital audio recorder were also utilised in order to record parts of students' progress while they were developing their programming solutions. The recorder was also used in the fourth stage where semi- structured interviews with the four participants took place. The interviews aimed to explore students' perceptions of the programming concept as well as to achieve triangulation of the data.

Activity Design

According to the rationale, it was intended that the activity would draw upon a real- life scenario, the implementation of which to be based on the use of conditional statements. The real- life problem of the functionality of the lift was considered suitable in order for a programming solution to be drawn based on the programming concept of conditional statements.

Drawn upon Alexopoulou and Kynigos (2008) study, the idea of the half- baked approach was exploited as it could guarantee the frame needed and the context in which students could construct and explore the concept of conditional statements. The semi- finished nature of the approach could also guarantee that a level of freedom could be given to the students to develop their own thinking and their own programming solutions. In order to restrain students' cognitive load, it was opted to provide them with parts of code that was not relevant to the programming concept of conditional statements; students were free to experiment with this part of code in order to manage to compose a solution putting together the different parts of the 'puzzle' (see figure 12).

The main activity based on the scenario of the lift, consisted of two parts. The purpose of the first part of the activity was to snap together the appropriate blocks so that the lift will function according to the buttons pressed. The buttons '0', '1',' 2' (see figure 4) which represented the ground floor, the first floor and the second floor had already been programmed in the framework of the idea of the half- baked approach (see figures 5,6,7). In accordance to the activity: If button '0' is pressed, the lift will move to the ground floor and the message 'ground floor' sounds or appears. If the lift is already on the ground floor, the message 'already on ground floor' will be displayed. Similarly, buttons '1' and '2', if pressed, function following the same concept. *The solution addressed to the problem is represented in figure 8.*

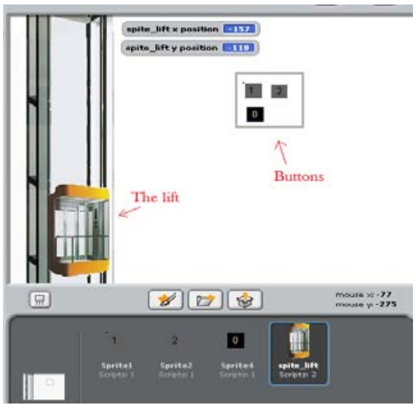


Figure 4. The components of the activity

Figure 5. Script for button ‘0’

Figure 6. Script for button ‘1’

Figure 7. Script for button ‘2’

The second part of the activity was more advanced as it has been designed in a way according to which students should use nested conditional constructs (see figure 11) in order to solve the given problem. This part was based on the first part of the activity which was further enriched with the so called ‘risk button’ (see figure 9).

The task for the students was to snap together the appropriate blocks in order to put the lift temporarily out of order only when the risk button is pressed and the risk exists. If the ‘risk button’ is pressed, the lift does not move and is temporarily unavailable (even though button 0, 1, 2 will be pressed). The message ‘lift is unavailable’ appears, followed by the recorded message ‘lift temporarily unavailable due to technical problems’. After a specified amount of time the lift becomes available and functions normally according to the buttons pressed.

The risk button had been programmed for the students and further explanations were given to each of them individually about the idea underpinning this script in order to ensure that students were aware of the way the variable ‘risk’ is used. Obviously, the variable risk is set to one for a specified amount of time when the button risk is pressed (see figure 10). After this amount of time the variable is set to zero which means that the risk does not exist (see figure 10). *Written instructions, closely associated with the concept of the two parts of the activity, were also delivered to students.*

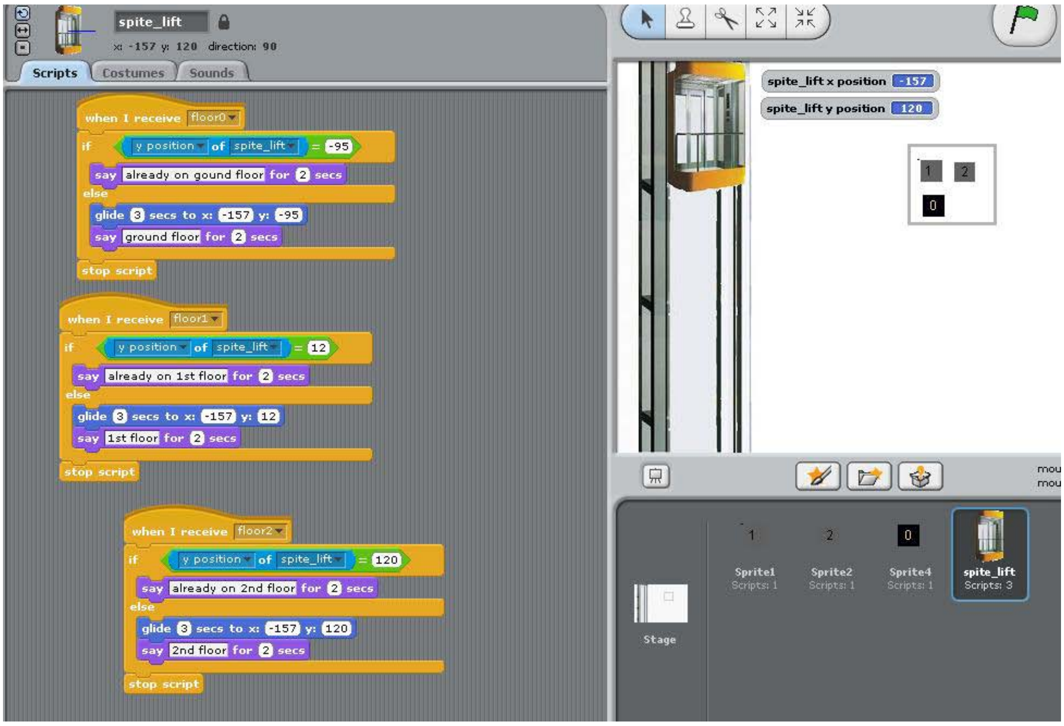


Figure 8. The solution to the first part of the activity

Figure 9. The risk button

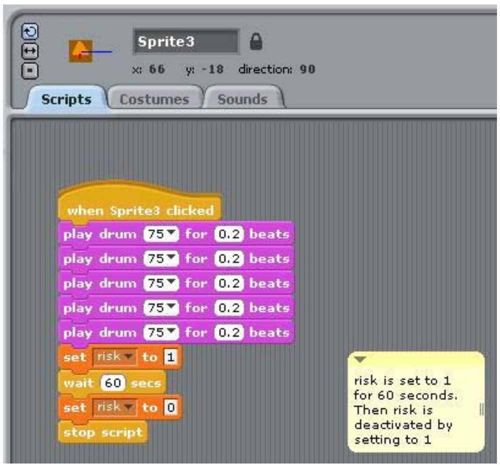


Figure 10. The script for the 'risk button'

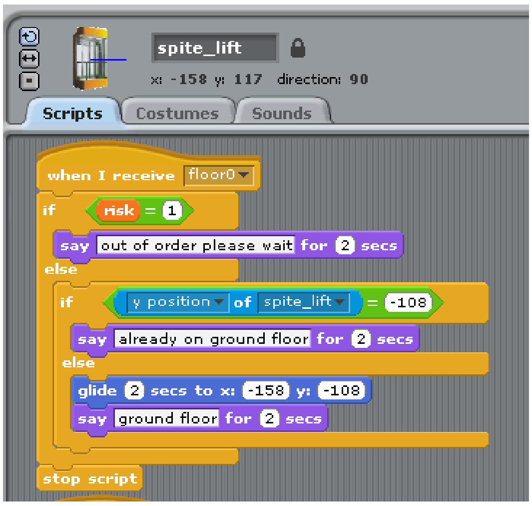


Figure 11. One of the 'nested if- else constructs' that students were called to implement

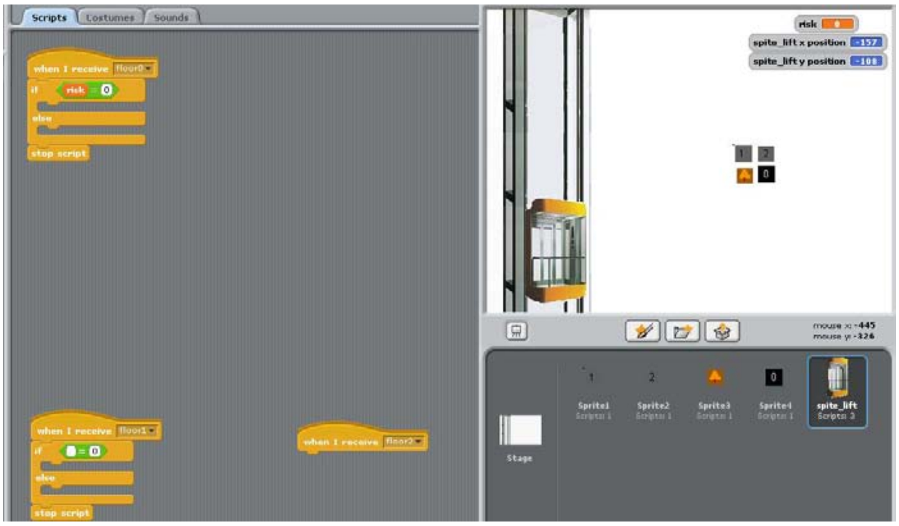


Figure 12.The half-baked approach for the second part of the activity

Findings

This section focuses on what the participants actually did and the way in which the Scratch learning experience supported the development of their ideas. The role that the features of simplicity and visibility, which were embodied in Scratch, played towards participants' engagement in the programming process is brought into focus. The role of the real-life scenario in the process of engaging in programming concepts is also discussed. The section ends with a discussion of how students' awareness of the way conditional statements function was emerged and shaped.

Programming in Scratch

The features of visibility and simplicity, (as introduced by DuBulay *et al* (1989)), which are embodied in the programming environment of Scratch, were critical for the students' engagement in the process of programming. In a way, the absence of concerns about 'syntactic issues/errors' and the simplicity underpinning the process of building a script (by snapping together different blocks) encouraged the participants to focus upon the implementation of the programming solution. Although the participants did not verbalise this explicitly, they at no time expressed that they encountered difficulties in using Scratch. Instead they focused on finding *'a solution that worked'* and '*understanding what each block does*'. The features of visibility and simplicity can be found lying in Guzdial's (2003) consideration, in accordance of which, it is of great significance to provide novice programmers with immediate feedback on their work, especially when the work is still a work in progress. Interestingly, Scratch provides this option to users and students unconsciously took advantage of this opportunity. In a way, the whole process of programming in Scratch was based on the feature of the 'immediate feedback' which supported students to test their solution and engaged them in a debugging procedure. This allowed them to reflect on the way the conditional statement operated, by checking the outcome when changes were made in the body of the conditional constructs. Discussions that took place with Luke and Kevin are detailed below.

> R: I noticed that while you were working you realised that the script did not work and you said to me 'Oh, this does not work'. How were you sure that your script did not work?
>
> L: Cos basically, I played it. No. First I made it and I played it and after I played it I realised that something was wrong with it. I could look back at everything; and I found the problem and I played it again [...]

Interestingly, the discussion exemplifies Guzdial' s (2003, p.19) point according to which when students are working on their script, they 'don't want or need to deal with subtle shades of correctness- they want it to be right or wrong, so that they can correct it and move on'.

In a similar way, the following episode with Kevin exemplifies the fact that the feature of the immediate feedback, as well as the simplicity underpinning the process of changing the script, played a significant role in the programming process and met his needs for solving the problem immediately.

> K: Miss?! Look! Let's go! [he presses the execution button]
>
> R: ...
>
> K: It's not doing anything! [disappointment and anger]
>
> R: Don't worry. Take your time...
>
> K: No, no! [he is looking his script again]
>
> R: ...
>
> K: Uhmm... that's why... I didn't add this; here is y-position not x. [he makes the needed corrections and moves on] Now, it must work!

The real life scenario

This section focuses on how the use of the real life scenario resulted in supporting student's understanding of programming concepts. First, it was observed that it was easier for the students to simulate the functionality of the lift due to the fact that they could establish connections with their experience of using it. Second, it was perceived that students had drawn upon the real- life scenario of the activity, developed it and even extended it. Interestingly, the extended scenario required the exploitation of the conditional statements and nested conditional constructs as well as other programming commands at a more advanced level. Both points are discussed further below.

As far as the first point is concerned, it is worth mentioning that almost all the students altered the messages that were supposed to be displayed on screen or to be heard. Initially this was considered to be the result of the playful nature of the learning experience. However, it then became clearer that in fact students' intention was to change the messages and the sounds, so as these corresponded to their real-life experiences. For instance, it was observed that Billy changed the message that was displayed on the screen when the risk button was pressed. When he was asked to explain his script, interestingly he mentioned:

> *'I just did it like the normal lift. If there is risk the message 'lift is temporarily out of order please wait for a member staff' sounds. And the lift is not moving. Like the normal lift'.*

The real- life scenario possibly encouraged the establishment of connections with their real- life experiences; this might allow students to engage more easily with the activity (without paying attention to the written instructions) making also clearer the rationale for using the conditional statements.

In relation to the second point, the students were seen to enter Resnick's creative thinking spiral for a second time through imagining a new idea which extended the given scenario of the initially given activity. Interestingly, through the implementation of the extended scenarios students' became involved deeper into the programming concepts. In fact, the extended scenario usually moved them to exploit the programming concepts introduced at a more advanced level or to explore new programming concepts altogether.

Timothy was the first student to explore and implement his own ideas. His idea was to extend the real-life scenario by adding a person on the lift. He focused on achieving synchronisation among the three objects (the lift, the man and the button pressed). Interestingly, the same tendency was perceived in the case of the other three students, who in the time left attempted to either to add more buttons to interface with the lift or to add a man in the scenario (*see figure 13*). The scenario was extended further using this concept: 'The man drives the car, gets out of the car and uses the lift'. To understand and use this concept, students used simple and nested conditional statements, new blocks from the 'palette' ('hide', 'show', 'rotate' commands) and the 'broadcast mechanism' in order to synchronise the different objects.

> R: What are you trying to do Luke?
>
> L: I'm just trying to make the lift shake a bit when there is risk.
>
> R: How will you do this?
>
> L: I don't know actually. Maybe I'll use these if-elses. Combined together. [He means nested 'if- else' constructs]. Miss how can I make it move slightly right and left quickly?
>
> R: What about using the 'move' block [...] ?

However, the contribution of the programming environment of Scratch should not be ignored. The students' engagement in the process of extending the real- life scenario is closely associated with the representational pluralism that the programming environment of Scratch supports. The idea of extending the real- life scenario would be impractical if the programming environment of Scratch did not provide users with the necessary tools (i.e. designing area ,range of sprites and blocks, area for recording messages) in order to breathe life into their ideas.

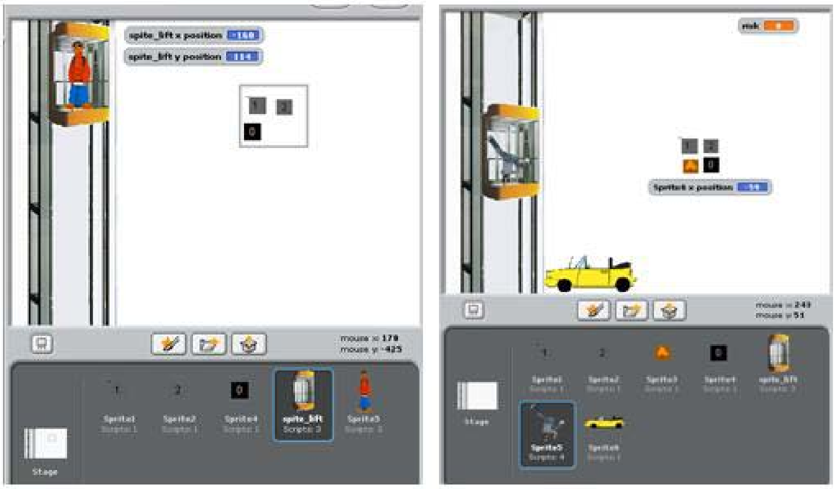


Figure 13. The extended scenarios

Students' perceptions

During the first stage of the study, it was critical to examine whether or not the students had come across a statement like the one given to them (see *figure 14*) because this would imply that they could be familiar with the concept of conditional statements. Interestingly, none of the four students were able to explain the given conditional statement of the questionnaire. Examining the ability of students to explain the given conditional construct after their engagement in the study was considered to be particularly interesting, as it would indicate their level of understanding.

It was observed that the students' explanations could mainly occur with references to Scratch. Students used the terminology used in Scratch or tried to correlate the given conditional statement (*see figure 14*) to the ones they had constructed previously in the framework of the activity. Presumably, this fact was expected, taking into consideration that this experience was the students' first time to engage in programming concepts. For instance, Luke was a typical case. He explained the conditional statement (see *figure 14*) by making use of the terminology embodied in the blocks/commands in Scratch (x-axis, x-position, y-position etc) and parallelizing it to a conditional construct which he had implemented during the activity.

```
If (x=0)
   walk();
else
   run()
```

Figure 14. The given statement

> R: [...]Can you now explain this statement [see figure 14]?
>
> L: Yeah! When on x-axis is x-position equals to zero [...] Look here Miss! [he shows to me a conditional statement that he had previously implemented on Scratch]. It is the same thing. Here [he is referred to a condition in his script where x equals to 39] it will not be 39, it will be zero. And it will walk. Otherwise the thing will run.

However, some other explanations were 'less associated' with the programming experience in Scratch. This is not to say that these explanations were completely disassociated with the activity; but rather to bring into focus the fact that students were seen to develop a more comprehensive understanding of the programming concept of conditional statements and to identify programming scripts behind everyday technologies. Previous experience seems to set a basis whereupon further connections with the wider application of the programming concept of conditional statements can be established. The explanations which were 'less associated' with Scratch were seen as the result of a fruitful interpretation of the new knowledge based on the existing experience. In a way, this process brings into focus the ideas that underpin the Piagietian constructivism, which argues that previous experience and knowledge in general affects the ways in which the construction of new knowledge occurs (Hewson, 1992).

Billys' explanation falls into this category. The fact that he had first come across a conditional statement like the given one on a webpage, which did not load normally, can be seen as a significant factor that activated the process of the de- contextualisation.

R: Initially you stated that you don't know what this statement does.

B: Yes. But I've seen this on a webpage. It didn't load properly.

R: Yes. I remember that you had mentioned this. Can you now explain what the statement does?

B: Basically, it is like here with the lift. If I press '1' the lift goes there [he shows his script]. It is pretty similar to that. If I press something and there is a problem, the page doesn't load properly and probably after refreshing it [noise] there is no problem and it loads properly. It is all programming and on the webpage

R: Interesting. So can you explain to me what this statement does?

B: Basically is a script. And if 'x=0' it will walk. If it is not, the object will run.

R: What will be the value of 'x' in order for the object to run?

B: Basically, it has to be greater than zero or lower than zero but it can't be zero coz the object then won't run.

However, whether closely associated with the experience in Scratch or 'less associated', all the students' explanations illustrated a growth in awareness about the programming concept of conditional statements. The growth in awareness was identified through a change from being unaware of the way the statement functioned to becoming more aware. The achievement of awareness was not a straightforward process for all the students. In whichever way, small or big there was some degree of awareness which was achieved after considering and reconsidering possible approaches, experimenting with multiple solutions and passing the different stages of Resnick's creative spiral.

Conclusion

This study presented how the use of Scratch in a scenario drawn upon real- life, supported the four participants' in developing their understanding of conditional statements. Although, it was a small- scale study and the findings raise new questions for exploration, it seemed that the programming environment of Scratch, as well as the real- life scenario of the activity, was capable of encouraging students' to understand conditional statements. However, there are additional elements that could have contributed to this; the way instructions were provided, the sense of the creation or the feeling of 'developing as a programmer', another designing of familiarization activities and the interrelations between these elements could be the basis whereupon further research could be conducted.

In a way this study could be seen as a small step towards the direction of encouraging students to experiment with knowledge in context, allowing them to make connections with the real world and enabling them to cope with problems creatively and playfully (Resnick, 2007a; Galarneau 2005; Nicaise *et al,* 2000; Siemens 2004).

From a student viewpoint, a learning experience -drawn upon the Kindergarten approach and the thick view of authenticity in Scratch- creates two significant opportunities. Firstly, it allows children to experiment with programming concepts and shape the idea of computational programming. In the framework of this process is likely general thinking skills as well as a general interest for the area of programming and computer science to be developed. Last, it allows students to implement their ideas. Through this process there is the potential for students to see themselves as producers and not merely as consumers of technological 'products'; however, the nature of such production is worth arousing one's interest. It is the type of producer students become that is central to the conception of the Creative Society.

References

Alexopoulou, E. and Kynigos, C. (2008), 'Half- baked games as a context for understanding conditional constructs' (in greek). Available at: http://www.etpe.gr/files/proceedings/21/1223368001_DIDINFO08_71_80.pdf

Doukakis D., Tsaganou G., Grigoriadou M.(2007), 'Using animated interactive analogies in teaching basic programming concepts and structures'. *Proceedings of the ACM Conference on the State of: Informatics Education Europe II*, Thessaloniki, Greece, 257-265.

DuBoulay, B. (1989), 'Some difficulties of learning to program', In: E. Soloway and J. C. Spohrer (eds), *Studying the Novice Programmer*, Hillsdale, NJ, Lawrence Erlbaum Associates , (pp.283-299).

Galarneau, L. (2005), 'Authentic Learning Experiences Through Play: Games, Simulations and the Construction of Knowledge'. Available at: http://www.digra.org/dl/db/06276.47486.pdf

Guzdial, M. (2003), 'Programming Environments for Novices'. Available at http://coweb.cc.gatech.edu/mediaComp-plan/uploads/37/novice-envs2.pdf

Hewson, P. W. (1992), 'Conceptual change in science teaching and teacher education'. National Center for Educational Research, Documentation, and Assessment, Madrid, Spain.

Kahn, K. (2004), 'ToonTalk- Steps Towards Ideal Computer- Based Learning Enviroments'. In: Mario Tokoro and Luc Steels (eds), *A learning Zone of One's Own: Sharing Representations and Flow in Collaborative Learning Enviroments*, Ios Pr Inc.

Lee, O. and Lehrer, R. (1987), 'Conjectures concerning the origins of misconceptions in LOGO'. *The annual meeting of the American educational research association (AERA 1987)*, Washington, DC.

Maloney, J., Peppler, K., Kafai, Y., Resnick, M., and Rusk, N. (2008), *'Programming by Choice: Urban Youth Learning Programming with Scratch'*. Available at: http://web.media.mit.edu/~mres/papers/sigcse-08.pdf

Mims, C. (2003), 'Authentic learning: A practical introduction and guide for implementation'. *The Meridian Journal, 6*(1).

Nicaise, M., Gibney, T. and Crane M. (2000), 'Toward an Understanding of Authentic Learning: Student Perceptions of an Authentic Classroom'. *Journal of Science Education and Technology*, 9(1), 79-94.

Papert, S.(1993). *Mindstorms: Children,Computers, and Powerful Ideas.* London: Basic Books, 2nd Edition.

Pea, R. D. (1986). 'Language-independent conceptual bugs in novice programming'. *Journal of Educational Computing Research*, 2(1), 25-36.

Putnam, R. T., Sleeman, D., Baxter, J., Kupsa, L. (1989), 'A summary of the misconceptions of high school BASIC programmers', In: E. Soloway and J. C. Spohrer (eds), *Studying the Novice Programmer*, Hillsdale, NJ, Lawrence Erlbaum Associates, (pp. 301-314).

Resnick, M. (2008), 'Falling in love with Seymour's ideas'. Available at: http://llk.media.mit.edu/papers/AERA-seymour-final.pdf

Resnick, M. (2007a), 'Sowing the Seeds for a More Creative Society'. *International Society for Technology in Education,* 18-22.

Resnick, M. (2007b), 'All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten'. *Proceedings of the 2007 Conference on Creativity and Cognition*, Washington DC, USA (pp. 1-6).

Resnick, M., and Silverman, B. (2005), 'Some Reflections on Designing Construction Kits for Kids'. *Proceedings of Interaction Design and Children conference*. Available at: http://llk.media.mit.edu/papers/IDC-2005.pdf

Soloway, E. and Spohrer, J. (eds) (1989), *Studying the Novice Programmer.* Hillsdale, NJ, Lawrence Erlbaum Associates.

Shaffer, D. W. and Resnick, M. (1999), '"Thick" authenticity: New media and authentic learning'. *Journal of Interactive Learning Research, 10*(2), 195-215.

Siemens, G. (2004), 'Connectivism: A learning theory for digital age', available at http://www.elearnspace.org/Articles/connectivism.htm

Tzimogiannis, A. (2005), A pedagogical context for teaching programming in Secondary Schools (in greek)*. Proceedings of the 3rd Panhellinic Conference in Didactics of Informatics*, Korinthos.

Introducing robotics to teachers and schools: experiences from the TERECoP project

Dimitris Alimisis, *alimisis@otenet.gr* School of Pedagogical & Technological Education, Greece

Javier Arlegui, *arleguip@unavarra.es* Public University of Navarra, Spain

Nello Fava, *favanello@museocivico.rovereto.tn.it* Town Museum of Rovereto, Italy

Stassini Frangou, *s_frangou@phs.uoa.gr* National and Kapodestrian UoA, Greece

Silviu Ionita, *silviu.ionita@upit.ro* University of Pitesti, Romania

Emanuele Menegatti, *emg@dei.unipd.it* Dept. of Information Engineering, Univ. of Padova, Italy

Stefano Monfalcon, *stefano.monfalcon@gmail.com* Town Museum of Rovereto, Italy

Michele Moro, *mike@dei.unipd.it* Dept. of Information Engineering, Univ. of Padova, Italy

Kyparissia Papanikolaou, *spap@di.uoa.gr* School of Pedagogical and Technological Education, Greece

Alfredo Pina, *pina@unavarra.es* Public University of Navarra, Spain

Abstract

This joint paper is based on cooperative work done by the TERECoP project partnership during the 3 years of the project (2006-09) and presents a constructivist methodology for teacher training in robotic technology and its implementation within the framework of teacher training courses. The courses gave teachers the chance to pass from their initial role of a trainee in robotics to that of a teacher planning activities in robotics for their pupils. Some indicative examples from teacher training courses and activities in school classes are presented and commented. Finally, conclusions and recommendations based on our 3-years experiences from the TERECoP project are attempted. The TERECoP partnership through this paper aspires to contribute to the further development of the dialogue within the broader European and international community of educational robotics under the light of constructionism theory.

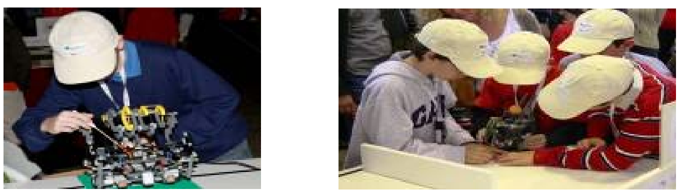


Figure. Greek pupils in action with Lego Mindstorms NXT robotics kit

Keywords

Educational Robotics; Teacher Training; Constructionist Learning

Introduction

It was 3 years ago in EUROLOGO 2007 conference, just some months after having started the TERECoP project, when we presented our project aims, aspirations and expectations emphasizing then that "TERECoP project's aim and ambition is to contribute a constructivist model of teacher training in these new (robotic) technologies… TERECoP project is expected to be a beneficial one for teachers both at national and European level enabling them to introduce robotics in their classrooms in a constructivist framework…" (Alimisis et al., 2007).

Now, a few months after the end of the project, this joint paper, based on the cooperative and shared work done by the TERECoP partnership during the 3 years of the project, comes to summarize our experiences from the implementation of pilot teacher training courses and interventions in school classes and to offer some of our conclusions drawn from the evaluation of those activities.

Theoretical background and methodology of the TERECoP Project

Research in the field of educational robotics has for years placed emphasis on the interplay between the invention of new technologies and the development of innovative ways of learning: new pedagogical ideas can lead to new technologies, and vice-versa (Martin et al 2000). Since the late 1960's, research has been developed for robotic construction kits for children focusing on the invention of construction kits and programming tools that children will find easy to understand and control, thus becoming active participants in their learning and creators of their own technological artefacts instead of being just users of devices that others have made for them (Martin et al., 2000).

Educational robotics has been introduced as a powerful, flexible teaching/learning tool stimulating learners to control the behavior of tangible models using specific programming languages (graphical or textual) and involving them actively in authentic problem-solving activities. This is the field where the European project "*Teacher Education on Robotics-Enhanced Constructivist Pedagogical Methods - TERECoP*" was activated during the years 2006-2009 with the participation of 8 European educational institutions from 6 European countries (www.terecop.eu) (Alimisis et al., 2007; Alimisis, 2008; Papanikolaou et al., 2008; Arlegui et al., 2008; Fava et al., 2009).

Over the last few years, several educational projects and initiatives have been developed in the field involving universities, schools or other educational and research institutions. A pluralism of thematic areas, educational objectives, learning approaches, topics and diverse audiences has been involved in past and current applications of robotics in the broader school settings. The TERECoP Project has aimed at the development of a design and implementation framework for activities advisable mainly for secondary school education related to programmable robotic constructions and based on learning methodologies inspired from constructivism (Piaget, 1974) and constructionism (Papert, 1992) theory. According to our view robotics projects and activities in school settings might be classified in two separate categories:

- Robotics as learning object: This first category includes educational activities where robotics is being studied as a subject on its own. It includes educational activities aimed at configuring a learning environment that will actively involve learners in the solution of authentic problems focusing on Robotics-related subjects, such as robot construction, robot programming and artificial intelligence.
- Robotics as learning tool: In the frame of this second category, robotics is proposed as a tool for teaching and learning other school subjects at different school levels. Robotics as learning tool is usually seen as an interdisciplinary, project-based learning activity drawing

mostly on Science, Maths, Informatics and Technology and offering major new benefits to education in general at all levels.

In the TERECoP project, a constructivist view for learning was adopted, whereby robotic technologies are not seen as mere tools, but rather as potential vehicles of new ways of thinking about teaching, learning and education at large. Learners, in a constructivist learning environment are invited to work on experiments and authentic problem-solving with selective use of available resources according to their own interests, research and learning strategies. They seek solutions to real world problems, based on a technological framework meant to engage students' curiosity and initiate motivation. The LEGO Mindstorms NXT system (http://www.legomindstorms.com) was selected among others since we found it appropriate to partner technology with the ideas of constructivism and costructionism. It offers building materials, sensors connecting a robot with the external environment and programming software with a simple graphical interface intended for the creation of robot behaviours. In addition to that, it is a mostly open platform allowing the development of a large variety of formal and informal robotics-oriented activities, as the huge number of initiatives and examples shown on the Web confirms (www.legoeducation.com).

Believing that the role of teacher is crucial for the successful introduction of technological and pedagogical innovations in classrooms, the TERECoP Project focused on the training of prospective and in-service teachers in the use of robotics technologies through courses implemented in each of the six participating countries, the evaluation of the training courses and the dissemination of the educational results at European level. As part of the TERECoP activities, we developed pilot training courses for in-service and student-teachers. The aim of the courses was to enable trainees to understand the pedagogical perspectives of educational robotics and to develop robotic activities within a constructivist and constructionist teaching and learning approach.

The idea of “learning by design” is central in our pedagogy supported by a project-based learning approach. The learning tasks of the course are organized as small or large scale robotics projects encouraging trainees to design and develop their own products. As Rusk et al (2008) point out, the way robotics is currently introduced in educational settings is unnecessarily narrow and suggest that designing activities, focused on *themes* and not just on *challenges,* helps to engage wide and diverse audiences in robotics. In accordance with this idea, the projects proposed in our methodology focus mostly on themes broad enough to give everyone freedom to work on a project according to their interests and are developed around open-ended problems engaging participants not only in “problem solving” but also in “problem finding” (Rusk et al., 2008).

The active involvement of the trainees in all the parts of the course was the second important aspect. A teacher training course can contribute to the professional development of teachers, by forming relations between teachers’ existing experiences and the proposed new educational technologies. So, from the beginning of the course, trainees were encouraged to express themselves and to participate in all activities of the course through discussions in small groups, presentations in plenary sessions and publications on their e-class. In this way, current ideas, beliefs and attitudes of the participants were made explicit and evaluated within the constructivist approach.

Throughout the course, trainees were working on their learning tasks independently. The role of the trainer was to facilitate the learning process by creating an interesting and stimulating learning environment: giving feedback at regular intervals, raising interesting questions, guiding the research concerned and synthesizing ideas. Trainees, on the other hand, were responsible for their work; they could follow their own path in their exploration and could develop their own ideas. They were supported in their work by useful resources, such as worksheets, representative examples and user guides. We were also aware that teachers need and appreciate support after the training phase in form of fora, Q&A sessions with the trainers and exchange of experiences during their implementations in classroom. In the pilot activities we also provided this support, when possible, with positive results.

Finally, the constructivist learning environment was based on cooperation. Social interaction within small groups generates a fruitful learning environment, where ideas are expressed, discussed and developed. So, the most of the learning tasks were performed by the trainees working in small groups.

Training teachers in educational robotics

Some indicative examples from the TERECoP training activities are presented shortly in the following sub-chapters just to shed light on the above mentioned theoretical ideas and methodology.

The training course held in Athens, Greece

The TERECoP group in Greece involved researchers and teachers from secondary and tertiary education. The central concept of the training course implemented in Greece was to build constructivist professional development sessions based on learning activities that teachers should be able to use in their own classrooms (Papanikolaou et al., 2008). To this end we used the methodology for designing robotics-enhanced activities for secondary school students which we had developed in previous stages of the TERECoP project (Frangou et al., 2008).

The course was held at the premises of the School of Pedagogical and Technological Education (ASPETE) in Athens, and was organized in 5 face to face meetings of six teaching periods each (5x6=30 teaching periods in total) during 3 Fridays/Saturdays afternoons. In this course 4 trainers and 23 trainees participated: 15 teachers in service (4 from primary education and 11 from secondary education) and 8 candidate teachers. During the course, trainees worked in a constructionist learning environment since they were actively engaged in hands-on activities, working in teams with peers. To enhance the sense of community and promote collaboration through the course an e-class was also maintained.

During the training course, trainees undertook multiple roles. They initially worked as students to familiarize themselves with materials and the programming environment, then they worked as teachers to reflect on the methodology for designing robotics-enhanced activities used in TERECoP and on the pedagogical implications of working with programmable robotic constructions in the classroom, and finally as designers constructing their own project. Finally six projects were developed by the teachers and evaluated by both trainers and trainees.

The evaluation of the course was based on the trainees' products through the course and mainly on the projects they developed, on diaries kept by the trainees during the course sessions, on questionnaires filled in by the trainees and on a semi-structured interview at the end of the course. Based on the evaluation results, we identified that the trainees:

- recognised their active participation in all the sessions of the course and their creative involvement even in the theoretical parts introducing constructivist and constructionist principles and the methodology for designing robotics-enhanced projects;
- very much liked the activity-orientation of the educational content;
- acknowledged the central role of the e-workspace during the face-to-face meetings and beyond them in enhancing social interaction and promoting a positive sense of community;
- acknowledged the potential of educational robotics as a teaching tool but also as a subject in different disciplines such as technology, informatics, and engineering;
- highly appreciated the opportunity to create their own projects.

Training experiences from Italy and Spain

In this section we present some relevant results from 3 actions (pilot training courses) that have been carried out in Italy and in Spain. These activities had a “second demonstrative part”, respectively the Discovery Film 2008 Exhibition in Rovereto (Italy) and the First Lego League (FLL) tournament in Pamplona (Spain). The first pilot course was organized in November/December 2007 in the Town Museum of Rovereto (TMR, http://www.museocivico.rovereto.tn.it) which is one of the Italian partners of the TERECoP project and an active and sound divulgation-center in Northern Italy. The course was attended by 15 teachers coming from schools of Trentino and Veneto. The second course was held in April/May 2008 in Pamplona (Spain), where several institutions collaborated to organise it: the Public University of Navarra, the Supporting Center for Teachers of Navarra and CEIN (a public company interested in promoting creativity and innovation among young students). This was a fine example of synergy among the school education system, the local university and an external actor. In both cases, the TERECoP partners from Univ. of Padova (Italy), Public Univ. of Navarra (Spain) and Town Museum of Rovereto acted as trainers.

A third teacher training activity was organized beyond the TERECoP framework but it followed mostly the same curriculum. It was held in October/November 2008 in Bolzano (Italy), hosted by a private senior secondary school that has already integrated robotics experiences within their school subjects (http://www.rainerum.it). It involved about 15 teachers (almost all in-service) from different levels and subjects of specialization. The TERECOP partners from Univ. of Padova and TMR acted as trainers and the local education office loaned some robotic kits to the trainees to stimulate their personal experiences and to enable them to design short school projects during the course period and after that as well. In the meantime additional support was provided by tutors through a web-based specific forum on Moodle platform.

An open activity was organised following the training period, where secondary level students with their teachers (some of them were trainees or tutors in the above mentioned courses) showed their robotics projects and how they were using robotics at their school. In May 2008 the Discovery Film Exhibition in Rovereto hosted an open day for Educational Robotics with stands of several school institutions, exhibitions, meetings, presentations and film showings, all about robotics. In November 2008 the Spanish TERECoP partners co-organised the FLL (http://www.firstlegoleague.org/) tournament in Pamplona as part of the Spanish national qualifications for the international FLL, an international competition for primary and junior secondary student teams supported by an experienced teacher. 15 teams from Navarra and 1 from Zaragoza participated in the first organisation of the event in Navarra. Every team had to design strategies and to build and program Lego robots to achieve pre-defined goals in order to get positive marks during the contest. Every year the contest focuses on a different real-world topic related to science. There was also a scientific project related to the topic of the year that the participating students were requested to develop and present.

Some details about the organization of the courses and their evaluation have already been reported (Arlegui et al., 2008; Arlegui and Pina, 2009). Some improvements and adaptations suggested by the previous experience coming from the course held in Rovereto were implemented in the course held in Navarra. The remarks made by the first group of teacher trainees, like suggestions to allocate longer time for the laboratory activities and to shorten the lectures about constructionism, were followed to improve the course. In the second implementation of the course a video conference system was used allowing partners from Italy to act as distance trainers.

The courses were aimed to achieve two main objectives:

- to assure scientific competences necessary for students to face the nowadays world challenges;

- to design activities and curricula able to adapt disciplinary structures to the learning dynamics.

During the design of the courses, we took particular care to:

- adequately combine inter-disciplinary groups of trainees,
- stress the importance of linguistic aspects during the teaching process (a natural language for group discussions versus the NXT-G formal language for programming);
- identify the trainees' and trainers' tasks during the training process;
- establish a progressive constructivist path for the training course within a Project-based Learning (PBL) approach.

We emphasised that the constructivist approach itself requires experimentations with open problems and that the project is an integrated sequence of problems (in a single purpose and context). So in our courses the constructivist path was achieved combining the following aspects:

- using a PBL methodology and the relevant tools developed during the TERECoP project;
- support of the capacity of abstraction for trainees: for example, some of the intermediate problems were leading to the construction of sub-commands corresponding to *MyBlocks* in NXT-G language.
- sessions organized in such a way that every group "had to construct something concrete" that might be an appropriate robot and/or a successful NXT-G program and to make a good demonstration of how they had worked offering ideas for suitable and affordable (for the other trainees and for their students) problems.

Therefore the work was oriented to build 'intelligent' machines to be controlled following all the steps of the construction, from the design to the realization, using a trial and error methodology, but with clear objectives. Moreover educational paths were designed to introduce robots in teaching scientific subjects, making the trainees confident with the constructivist learning following the aims of the TERECoP project. These aims could be summarized in the 'slogan': *"from robots as **learning objects** to robots as **learning tools**"*.

Training Experiences from Romania

In this section we present some results from the evaluation of the pilot training course that was provided for 15 trainees at the University of Pitesti, Romania. The group of trainees included 7 teachers in service and 8 students already enrolled in a pedagogical module (7 female, 8 male). The evaluation was carried out based on two structured questionnaires filled in by trainees at the end of the course (Part 1: Course design, content and pedagogical approaches and Part 2: Course delivery).

The analyses of trainees' answers showed that all of them considered that the educational module had enhanced the possibility of *acquiring new skills* to create models or to use simulations of science processes and also to use properly sensors and programming interfaces. The vast majority of the trainees considered that the course had enhanced their abilities for applying ICT tools in problem solving (80%), offered performing and accumulation of experiences (86%), methodological skills and realization of reasoning strategies (66%). All the trainees appreciated the applicability of the module in current science teaching as useful. Only few of them declared some difficulties that were related to programming and software development tools and (in one case) to putting together the mechanical parts of the robotic construction. The percent scorings in the second questionnaire highlighted the course was well delivered and well received by the trainees. The issues granted with high scores by the trainees at the end of the pilot course were trainer's enthusiasm (80%) and trainer's relationship with the group of trainees (86%). The support and the organization of the course were also appreciated as excellent (73%).

Generally, positive answers or high ranks were obtained from both questionnaires. The provided course revealed a high interest especially for in-service teachers that found in this a good reason to improve significantly their current pedagogical approaches, although some difficulties were observed to rethink the current topics in the actual curricula using robotics and particularly Lego Mindstorms kit from the constructivist point of view. By the student-teachers this course was perceived as a challenge and was appreciated as a good reason to reflect on the constructivism as a pedagogical strategy. Finally the training course seems that had a really quantitative and qualitative impact on trainees.

Introducing robotics in school classes

After the end of the training courses, our teachers-trainees were encouraged to implement similar robotics activities in their school classes following the proposed methodology. Although robotics is not included in the official curriculum of schools in our countries, which means that teachers should devise and undertake initiatives for introducing robotics in classes and to overcome obstacles related to the official curriculum, many of our trainees tried enthusiastically and succeeded in organising robotics activities in their classes either integrating them in the curriculum or working in after school hours. These implementations offered an evidence for a broader impact that our training courses had on the teachers and finally on their students. Some indicative examples from those implementations in school classes by our trainees are presented shortly below.

Case 1: Integration of Robotics in the curriculum

The following brief quotation from a report written by one of the trainees (teacher Gianfranco Festi, class of 5th year of the Industrial Technical Institute “G.Marconi” of Rovereto, Italy), who participated in the TERECoP course held in Rovereto (October-December 2008), shows a possible way to combine effectively robotics activities within the normal curriculum with public events beyond school.

“This was the third edition of our robotics-oriented project. Even if, the involved teachers are not always the same from year to year, the results were very encouraging and the collaboration with the Town Museum of Rovereto (TERECoP partner) allowed us to realize significant experiences. This year we can use this collaboration to open an innovative educational path. We will participate in several robotics events and exhibitions. The participation in public events is aimed to encourage our students to acquire interdisciplinary skills involving not only the technical subjects but also the humanities. In fact the students, working in small groups are requested to explain in details to visitors what they are showing using the most appropriate media. This turns to be a good experience also for teachers.
Starting form this year, the robotic subject is introduced in the normal curriculum of the technological secondary school. This has several objectives: introduction to programming; Introduction to technology; Introduction to robotics; development of software/hardware solutions in an interdisciplinary framework. Some laboratory hours will be reserved to prepare all the materials to be shown during the public events and namely the next "Discovery on film" exhibition in Rovereto. The necessary equipment, in addition to robots, will be primarily the one for the stand construction, such as posters, laptop and projector to display the multimedia materials. The activity of robotics will take place during the school year and culminate in participation in the event in late May 2010…”

Case 2: “Robo-poly”

“*Robo-poly*” is a project for introducing robotics in primary school designed by teachers who had attended the TERECoP training course in Athens (Terzidis et al., 2009). It includes a series of learning activities which are aimed to familiarize pupils with robotic systems and to allow them to acquire skills of construction and programming within a supervised environment where they can express themselves freely and enhance their creativity. The project was implemented in a year 4 class of the 2nd Primary School of Pallini, Greece, as an extra curriculum activity. The project was divided in three stages:

- Engagement: introduction to basic concepts;
- Experimentation: playning the Robo-poly game (to acquire basic skills);
- Creation: helping the “baby turtle” to reach the sea.

During the *engagement stage*, students were involved in introductory activities such as:

- discussion about what a robot is;
- assembly of the robotic device: students, in groups of 4-5 constructed a vehicle by using instruction sheets and teachers’ guidelines;
- basic movement and control commands: basic programming blocks were introduced by the teacher.

During the *experimentation stage* students were introduced to the basic concepts of programming in a playful manner. They were asked to play the game *Robo-poly* as reminiscent of the “*Monopoly*” game. The game was made up of 20 cards with tasks organized into three levels of difficulty. The goal of the game was for each team to gather 100 points. The goal could be achieved by using a variety of different card combinations. At the end of each activity the students presented their solution to the rest of the class.

Then, students were asked to give a solution to the open-ended problem: “Helping the baby turtle to reach the sea” (*creation stage*). According to this scenario students were asked to drive a just born baby *Karetta-Karetta* sea turtle (i.e. their robot) safely to the sea. The activities performed during that stage included: introduction of the scenario through a short story, creation of a poster based on the symbolic elements of the story, discussions, planning and solving the problem, presentations, and evaluation.

The familiarization and experimentation stages were aimed to familiarise students with robotics and programming and extend their abilities in dealing with this kind of technologies. The creation stage gave them the opportunity to utilize all skills and knowledge gained during the previous stages in a problem solving situation. The students were very enthusiastic and they participated creatively in all stages of this project. All groups worked and presented their work. The teachers tried to guide students in their exploration in a constructivist manner: asking questions, facilitating discussion between members of the group or between groups. At the end of the school year, in June 2009, the entire class referred to this experience with robotics as if not the best, one of the two most favoured activities.

Case 3: Motion and control

This project was implemented in a secondary school of Athens, Greece (3rd Gymnasium of Glygada) during March 2009. The project lasted for 12 teaching periods (45’) and it was developed as an interdisciplinary project combining Technology, Mathematics and Computer Science. It was the result of the cooperation of teachers from different disciplines: Maths, Computer Science, and Technology. One of the teachers had followed the TERECoP training course while the other two had attended a 6 hours seminar on educational robotics. The class teachers designed all the activities according to their curriculum objectives with some support and suggestions from a member of the Greek TERECoP team. The aim of the project was to familiarize students with robotic technologies (construction and programming skills) and to stimulate them to explore basic mathematics and science concepts about motion like displacement, circumference of a circle, length of an arc and proportional quantities. The main idea of the project was that students construct and program a vehicle which could move on a specific route.

Initially students decided on the path that their vehicle should follow and agreed on the themes/topics they were going to further explore (*engagement*). Then, students working with Lego Mindstorms constructed a robotic vehicle with several sensors able to move freely. Two guided experimentations aimed to help students to understand displacement and the changes in the direction of the vehicle (turns) (*exploration*). Students were encouraged to investigate the relation between the rotations of the motor and the change in direction of the vehicle to the right (angle of 90 degrees) (*investigation*). Students were also provided with a mock up of the vehicle route and they were asked to program their vehicle to move on that path. All groups used a light sensor to control the right turn of the robotic vehicle (*creation*).

During the engagement and exploration stages students worked with worksheets on guided activities. Specific questions guided their observations, measurements and calculations. All students performed very well in those tasks. Poor performances were observed in tasks looking for explanations requiring the use of concepts from mathematics like the formula of circumference. Students also performed quite well in the programming tasks (made their vehicle to move all along the proposed path). In the given time two groups out of six came up with a solution close to the initial plan agreed at the engagement stage. One group proposed a quite correct solution with a small error in the chosen turning parameters. Two groups did not take in consideration the physical characteristics of the construction (the position of the light sensor on the vehicle). The last group managed correctly each action of the final task (moving straight, turning to the right) but didn't manage to use these actions in a coherent manner.

Implementing robotics in classrooms was in the case of the 3rd Gymnasium of Glygada an innovation. Important factors that effectively supported that intervention were the positive attitude of the principal of the school, the collaboration between the teachers and the external support by a researcher (member of the TERECoP team) during the design and application phase.

Conclusions and recommendations

Robotics in Teacher Training: "teachers teach as they are taught, not as they are told to teach"

From the evaluation results it appears that our trainees appreciated the project-based learning method that they followed in their work and the exploration, experimentation and creation features included in that method. They appreciated also as "the best thing that happened to them during the course" the practical activities, the creation of engineering artifacts and their programming work ("*We experienced the joy of creation… when we built it up and it was operational*"). It appears, indeed, that they enjoyed their work with robots ("*When, ultimately, the robot moved along a square on the laboratory floor we rejoiced like young children*"). Their preference for practical work and their negative attitude towards "*theoretical presentations*" was also clear from the fact that they recorded among their negative experiences the cases where, because of lack of adequate training time, their practical work was substituted for theoretical discussion. Although the training methodology ensured trainees' active participation in practical activities, some of them requested even more practical activities and fewer presentations. Several trainees told us that the educator's axiom '*teachers teach as they are taught, not as they are told to teach*' emphasised by the trainers was really followed in the course. They admitted that they had a real experience of constructivism ("*It was for me a lesson of knowledge construction*", "*Constructivism was present all the time in the course*", "*this course was substantially different from the courses I had attended in the past*").

A practical difficulty encountered was the fact that the time available for face to face training sessions was usually limited and not enough for trainees to develop their team projects. They needed longer time to cooperate within their groups beyond the course sessions. It means that extra equipment (Lego Mindstorms sets) must be handed in each group for the necessary period of time and appropriate time arrangements must be anticipated in the schedule of the course that will allow the groups of trainees to work cooperatively at their own private time and place before they come back to the course to present their final products.

One of the main requests from the teachers concerned examples with a clear didactical content. In some cases we found that teachers were not interested in directions for building a new fancy robot or a robot with a cool behavior. They perceived this as out of the scope of their duty. They preferred to have a set of lab activities closely linked to their teaching subjects. We realized that designing such a set is not easy task and can be very tough for a teacher without previous educational robotics experience. Therefore, we tried to produce a variety of such experiences, and we intend to put as one of the main points of a future project the realization of a repository of several didactical experiences organized by curricular subject and learning objectives.

Another problem identified concerned the collaboration of teachers from different disciplines. Robotics projects are usually introduced as interdisciplinary activities. However, the cooperation between teachers from different disciplines is not usual in our school practice. We noticed that for example teachers of arts and humanities not only have difficulties in approaching robots as teaching tool, as one might expect, but also they have difficulty (or poor will) in collaborating on a project with the teachers of science or technology. Teachers seem also to worry about the management of big classes during the implementation of robotics-enhanced activities in school settings.

Robotics in School Classes: bringing the Logo turtle from the screen to the real world

Logo programming language and logo-based environments have been used for years as tools intended to introduce constructionism in classrooms. Our experiences emerged from the aforementioned cases and from several other similar activities carried out during the 3 years of the TERECoP Project (Alimisis, 2009), indicate that, educational robotics is a possible 'present & future' for constructionism theory and practice at the "beyond Logo" age. Educational robotics brings the Logo turtle from the screen to the real world and offers some key advantages: students can use robots not as a "virtual creatures" or as ready-made mechanical devices but they can build and program their own constructions and a wide variety of creative machines. Educational robotics might be seen as a vehicle of new ways of constructionist thinking and a vehicle driving to new paths in constructionist learning. It enables children (and educators!) to design their own instruments for meaningful investigations engaging them in new ways of learning in close connection with their interests and passions and providing a deeper and more concrete understanding of scientific ideas and a richer sense of the interplay between science and technology (Resnick, 1998). As a Greek teacher, who was supported by TERECoP to introduce robotics in his classroom, reported "*through these projects students get involved enthusiastically in learning activities that in traditional teaching situations would be boring for them. For example learning a mathematical formula to calculate an angle or a distance suddenly became an extremely interesting subject…*" (Giannakopoulos, 2009)

Our experience from the TERECoP activities in schools showed that when teachers are supported appropriately by trainers and their school authorities, they can find ways to introduce robotics in school classes as an interdisciplinary project. However, it is not always easy to integrate such approaches within the current school curriculum. This integration demands not only a lot of school time, but also changes in the official curriculum and a great investment in terms of the necessary equipment. So, a critical issue for integrating robotics-enhanced projects in the schools is how an interdisciplinary project may fit to the current school curricula and schedules. Interesting ideas were proposed for integrating educational robotics in schools such as working out interdisciplinary projects or research programs running out of the school everyday schedule in the form of after school classes.

As other teachers trained by TERECoP and involved in robotics activities in their class put it "*within the current curriculum, our students had not the time necessary to design, to experiment with, to program and to try their models…Due to the large number of students in one class, there* *was not enough room for all of them to work efficiently in groups. The teachers involved in this project were not willing to spend more time so as to experiment with the techniques or to use such activities in their everyday teaching practice. The majority of students could not find time available to participate in our project due to their busy afternoon schedule….We finally decided to form a group of 6 students who would work on robotics and eventually would participate in a robotics competition. We strongly believe that the competition motivated our students and stimulated their interest….Their parents' attitude towards the whole project was positive and supportive and that helped the team*" (Bakamitsou and Tsitsos, 2009). Once again, the combination of school activities with events happening beyond school seems to offer extra motives to students and makes their work more meaningful since the public event offers the opportunity for students and teachers to present their projects to a wider audience beyond the walls of their school class.

It is well known that in addition to the activities that take place in school settings, many other robotics events run in informal educational contexts, structured as competitions or exhibitions. The mission of the competitions is usually to engage young people in exciting mentor-based training that builds science, engineering and technology skills, inspire innovation, and foster self-confidence and communication skills. Robotics contests and the relevant project work appear as a very suitable platform to support team-based learning, which is often undervalued in the current school systems (Petrovič and Balogh, 2008).

In addition to teacher training courses and implementations in school classes, the TERECoP partnership organized successfully "European open days" on educational robotics (Venice 2008, Navarra 2009) and participated in exhibitions ("Discovery on film", Rovereto, 2007, 2008,2009). The European Open Days on Educational Robotics involved local educational authorities, schools, universities and companies and offered a forum of reflection about all the opportunities that robotics can offer for schools and students. During those "open days" teachers and students were provided with the opportunity to present and exchange experiences in a public place, to get in contact with associations and companies activated in educational robotics, to participate in round tables, to exhibit their models and to compete each other with their robots. We found very supportive for those activities the development of "local alliances" with institutions and companies involved in teacher training and in the promotion of creativity and innovation of the future citizens (Arlegui and Pina, 2009).

Our experiences from those open public activities showed that organizations external to the scholastic system like after school classes, clubs and centers equipped with robots, PCs, software and staff can stimulate and support teachers and students to develop their own personal or team robotics projects. These centers might be set in a network at regional or national level for an exchange of experiences and "know-how". All these promising initiatives like robotics competitions, exhibitions and "open days" complete the scenario of possibilities for the future years.

Epilogue

In addition to the aforementioned training and learning goals, the TERECoP partnership has been working the previous 3 years to build, through our robotics activities, communities of teachers activated in educational robotics at local, national and European level having achieved some remarkable results in Greece, in italy, in Spain and in the other participating countries and we intend to continue this work in the future. Finally, we aspire, exploiting the already gained rich experience, to continue to contribute to the further development of the dialogue within the broader European and international community of educational robotics, especially under the light of constructionism theory.

References

Alimisis, D. (ed.) (2009) *Teacher Education on Robotics-Enhanced Constructivist Pedagogical Methods*, School of Pedagogical and Technological Education, Athens,

Alimisis, D. (2008) Designing robotics-enhanced constructivist training for science and technology teachers: the TERECoP Project. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, AACE, Chesapeake, VA, pp. 288-293.

Alimisis, D., Moro, M., Arlegui, J., Pina, A., Frangou, S., Papanikolaou, K. (2007) Robotics & Constructivism in Education: the TERECoP project. In Ivan Kalas (ed.), *Proceedings of the 11th European Logo Conference*, Comenius University, Bratislava.

Arlegui , J. and Pina, A. (2009) Teacher training in the scientific field through robotic activities, in *Book of Abstracts of the Conference "Lessons Learnt from the Terecop Project and New Pathways into Educational Robotics across Europe",* ASPETE, Athens, www.terecop.eu

Arlegui, J. Fava, N. Menegatti, E. Monfalcon, S. Moro, M. Pina A. (2008) Robotics at primary and secondary education levels: technology, methodology, curriculum and science. In *Proceedings of the 3rd International Conference ISSEP, Informatics in secondary schools,* Toruń, Poland.

Bakamitsou T. and Tsitsos, V. (2009) Development of learning strategies focusing on the construction of mathematical meanings using contemporary new technologies and innovations of tangible materials, in *Book of Abstracts of the Conference "Lessons Learnt from the Terecop Project and New Pathways into Educational Robotics across Europe",* ASPETE, Athens, *www.terecop.eu*

Fava, N., Monfalcon, S., Moro, M. Menegatti, E. Arlegui, J., Pina, A. (2009) Teacher training in the scientific field through Robotics activities: some experiences from Italy & Spain, *Proceedings of INTED2009 Conference*, Valencia, Spain, 747-756.

Frangou S., Papanikolaou K., Aravecchia L., Montel L., Ionita S., Arlegui J., Pina A., Menegatti E., Moro, M., Monfalcon S., Fava N., Pagello I. (2008) Representative examples of implementing educational robotics in school based on the constructivist approach In *Proceedings of the SIMPAR 2008 conference/Workshop "Teaching with robotics: didactic approaches and experiences",* University of Padova, 2008

Giannakopoulos, N. (2009) Experiences from WRO 2009 competition and verifications about the robotics incorporation in the school, in *Book of Abstracts of the Conference "Lessons Learnt from the Terecop Project and New Pathways into Educational Robotics across Europe",* ASPETE, Athens, *www.terecop.eu*

Martin, F., Mikhak, B., Resnick, M., Silverman, B. and Berg, R. (2000) To Mindstorms and Beyond: Evolution of a Construction Kit for Magical Machines, Morgan Kaufmann Series in Interactive Technologies, *Robots for kids: exploring new technologies for learning*, 9 - 33

Papanikolaou K., Frangou S., Alimisis D. (2007), "Developing a framework for the design and implementation of activities with programmed robotic devices: The TERECoP Project. In *Proceedings of the 4th National Conference of ICT in Education,* Syros, , Vol. 2, pp. 604-612 (in Greek)

Papanikolaou, K., Frangou, S., Alimisis, D. (2008) Teachers as designers of robotics-enhanced projects: the TERECoP course in Greece, in *Proceedings of the SIMPAR 2008 conference/Workshop "Teaching with robotics: didactic approaches and experiences",* University of Padova, pp.100-111.

Papert, S. (1992) *The Children's Machine*. N.Y: Basic Books.

Petrovič, P., Balogh, R. (2008). Educational Robotics Initiatives in Slovakia, in *Proceedings of the SIMPAR 2008 conference/Workshop "Teaching with robotics: didactic approaches and experiences",* University of Padova, pp. 122-131

Piaget, J. (1974) *To understand is to invent.* N.Y.: Basic Books.

Resnick, M. (1998) Technologies for Lifelong Kindergarten, *Educational Technology Research & Development,* 46(4).

Rusk, N. Resnick M., Berg, R., Pezalla-Granlund M. (2008) New Pathways into Robotics: Strategies for Broadening Participation, J Sci Educ Technol. 17: 59–69

Terzidis, S., Goumenakis, G., Spyratou, E. (2009) An example of ICT- enhanced teaching and learning through the implementation of robotic systems in Primary School, in *Book of Abstracts of the Conference "Lessons Learnt from the Terecop Project and New Pathways into Educational Robotics across Europe",* ASPETE, Athens, www.terecop.eu

Acknowledgement and disclaimer: This paper is based on work done in the frame of the project TERECoP co-funded by the European Commission-European Programme Socrates/ Comenius /Action 2.1, Agreement No 128959-CP-1-2006-1-GR-COMENIUS-C21 2006 – 2518 / 001 – 001 SO2. We thank all the teachers and students who participated in the activities reported by this paper. This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Discussing about IBSE, Constructivism and Robotics in (and out of the) Schools

Javier Arlegui, *arleguip@unavarra.es,*
Dept of Psychology and Pedagogy, Public University of Navarra, Spain

G. Barbara Demo, *barbara@di.unito.it*
Dept of Informatics, University of Torino, Italy

Michele Moro, *mike@dei.unipd.it*
Dept of Information Engineering, University of Padova, Italy

Alfredo Pina, *pina@unavarra.es*
Dept. of Mathematics and Informatics, Public University of Navarra, Spain

Abstract

When we look for “Inquiry Based Science Education” (IBSE) in a search engine we find more than 2.000.000 items. The IBSE we refer to is, at the same time, a Problem Based Learning (PBL) approach enriched with experimental activities (using technological tools) and a constructivist learning method. It is consistent with our overall conception of the science and technology school activity, an activity of problem solving by building and using models through hands-on-experiences that introduce and motivate concepts present in school curricula and that are performed by pupils using technology. Here we concentrate on programmable mini robot and describe activities for a constructivist IBSE introduction to concepts from standard school curricula by programming mini robots. We propose in this paper an innovative vision to use IBSE, PBL, and constructivist strategies to design up-to-date and effective (regular) educational paths involving technological artifacts (like mini-robots) and linking the schools with cultural and scientific institutions (out of the schools) to promote the establishing and the application of new knowledge.

Figure 1. At the Educational centers: Trainers, Trainees and Pupils

Keywords

IBSE, PBL, Programming mini Robots, Constructivism/Constructionism, ICT, Science & Society

Introduction

Since its publication in 2007, Rocard's Report is a mandatory reference for European projects with educational impact (Rocard et al, 2007). Rocard and his co-authors emphasize the Inquiry Based Science Education (IBSE) approach as a guideline to be followed in projects for the future life in our schools from the very first years in order to trigger science attitudes from the very young age. In this paper we discuss relationships among IBSE, Constructivism/Constructionism and the use of technology in schools, particularly referring to authors' experiences in 'robotic-oriented' programming activities they have carried out during last three years with pupils from primary to high schools in Italy and in Spain (figure 1).

If we have a look to this conference call-for-paper, where the idea of constructionism is described, we find the following sentence (about the students): "They also need opportunities to actively explore and experiment with new concepts and materials on their own, to test and extend their understanding by designing and constructing sharable artifacts". In fact for us, Constructionism may be seen as a methodology to produce empirical facts by designing, assembling, planning and drawing, with pencils on paper, paths for the robots, in short concretely working. Observing the fact that current technological tools & devices are attractive to schoolchildren and are used by them normally more easily than by adults, technology shall be employed more often than it is nowadays to produce facts (or data) and in more original ways than it has already been used.

More specifically, here we suggest the use of autonomous mini robots because they are themselves technological devices for producing facts when students assemble different robots, then design different behaviors for their robots, scenarios where they have to move and finally implement programs discussing and comparing robots movements and behaviors with their classmates. Robots and programs are artifacts designed and then assembled or implemented by groups of schoolchildren. When a program is run also the behavior the robot shows during the program execution is an artifact. All these artifacts produce data that are collected by pupils, shared and discussed among them.

Activities based on technology implement an IBSE approach where pupils do not only collect facts from the experience but also produce new facts or can produce other new facts by changing data and/or achievements from these activities: i.e. when they are authors of their experiences. When we work concepts in standard school curricula using an IBSE approach we implement a learner centred education because pupils are the guided designers and authors of experiments where they produce facts, collect them and derive from them a concept relevant for their standard school curriculum.

For each activity the teacher gives pupils a problem to solve having a didactical objective. Each group of pupils decides how to assemble its robot, designs a behavior for that robot and develops its own program to implement the decided behavior that, with the assembled robot, is a solution to the given problem. The teacher follows discussions within each group and among groups of pupils monitoring them toward his/her didactical objective.

An adequate sequence of problems, designed to follow a constructivist strategy, can be used to organize courses. The didactical value of traditional formal teaching/learning may be expanded to include non-formal and even informal activities that can be carried out within the school or out of the school (figure 2). This obliges us to make serious reflections about the roles in the Educational issues of both officially educational institutions and cultural and scientific institutions and the society in general.

In the 2nd chapter we present a theoretical framework where our approach could fit. The 3rd makes some reflections about the interactions between IBSE, PBL and technology. In the 4th we describe some concrete activities where we propose using programmable robots in education according to the constructionist approach. The 5th chapter relates our educational approach with an education based on competences and with a social and professional framework, the European Qualifications Framework. In the last chapter some conclusions.

Figure 2. Some external activities:Venice Open day 2008, Discovery 2008, FLL 2008

Pedagogical concepts about IBSE

This chapter defines inquiry-based learning as introduced by Yves Chevallard (Chevallard, 1999). With respect to other definitions Chevallard addresses what IBSE is, i.e. which are its peculiar elements and characteristics, rather than how it can be put into practice during school activities as done for example in Andee Rubin (1993).

Linking facts to laws

In experimental sciences the IBSE methodology can be considered as constructing “praxeological” units, whose meaning is explained by Yves Chevallard as follows: ‘A praxeology shows the linking structure between a class of facts of a physical phenomenon and its “technical law” (or a technical model) that “explains” these facts. It also shows the linking between a class of technical laws (from similar phenomena) and its corresponding “technological law” that “explains” them, building in this way a hierarchically structured “theory” about the phenomena’ (see figure 3).

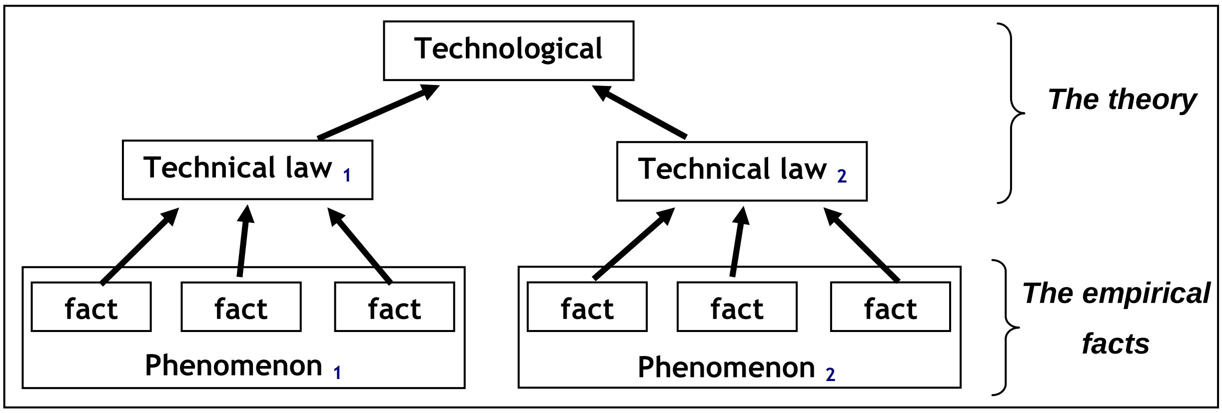


Figure 3. A praxeology structure from the facts to the theory

The “empirical facts” are the factual level (related with “doing”) and the structured law in the “theory” is the level of formalization (related with “knowledge”), and this praxeology structure ensures the link between the two. When, in a school class or laboratory, we build a praxeology structure from the facts to the theory (in the bottom – up sense of the diagram in figure 3) we are working in an inquiry based approach.

All this suggests a didactical strategy for the teaching and learning of a science praxeology at school:

1. Formulating an every day problem and reformulating it as a “scientific” (phenomenon) problem
2. Constructing the model:
    a. Collecting experiences about the problem (“interaction” activities by the pupils)
    b. Modelling the experience to build up a technical law (“formulation” and “verification” activities by the pupils)
3. Using the model:
    a. Using the technical law to solve problems
    b. Looking for the limits of the model, extending the initial problem to one more general and applying again the IBSE process to reach a technological law.

We can see that the above IBSE learning way is, at the same time, a problem based learning (PBL) approach and a true constructivist learning method, and it is according with our overall conception of the science and technology school activity as an activity of “problem solving by building and using models”.

IBSE, Constructionism and Technology

As we have said constructionism is based on giving students “opportunities to test and extend their understanding by designing and constructing sharable artifacts” through which they can actively explore and experiment with new concepts and materials on their own. Constructionist IBSE approach in school subjects is based on letting pupils design and construct the empirical facts. Technology can play an important role in these experiences. Software tools have been successfully used to produce facts (or data), for example by simulating real world events. Autonomous mini robots for some aspects allow us to go a step further because they are themselves technological devices for producing facts when students assemble different robots. Then when students design different behaviors for their robots and implement corresponding programs they have another chance of producing facts related to a concept. Both robots and programs are artifacts designed and then assembled or implemented by groups of schoolchildren and allowing experiences shared among pupils.

In the field of inquiry-based science teaching research the French Academy of Sciences has initiated the Pollen project in the frame of its international cooperation. Pollen is inspired from the previous and well known “Mains à la pâte” project, also French, and “invented” by the Nobel Prize Georges Charpak. Both projects promote science teaching renovation in primary schools based on the inquiry approach. For this, Pollen aimed at creating a sustainable framework for science education through a child-centred approach. Activities developed within the Pollen project on light, sound, temperature or other topics can be found at www.pollen-europa.net.

We share the same aims of “Mains à la pâte” and Pollen but projected towards a broad use of robotics and other technologically advanced tools. Our contribution concerns investigating the kinds of activities students can develop using technology as the main tool they employ for building their own knowledge. The fact that the learner is able to build his/her own knowledge is one of the inquiry-based science education main issues. In our everyday life we can observe how young people are uninhibited and easy to get involved in activities using technological devices. Our research concerns taking advantage in schools of this confidence in order to make pupils discover concepts from different disciplines during and by means of activities they design, discuss, implement, verify working with a group of classmates and then show and discuss with the rest of their class. Technology has been positively used to present in more attractive ways educational concepts: for example by asking students to build hypertexts (or podcasts or other types of document) describing history events or other subjects. It has an unreachable success in simulating events, in recording and helping the analysis of natural and unnatural events, for making visible and manifest biological, chemical processes. Our investigations concern giving to the pupils opportunities to be authors of activities where technological devices are used; in this way empirical facts are produced and collected in order to derive/work concepts from school standard curricula.

More specifically in this paper the authors analyse different research works they have carried out using programmable mini robots in schools. One of these aspects concerns the programming language offered to schoolchildren. Sometimes a textual Logo-like language is introduced to children in different layers of primitives corresponding to new types of activities the teacher suggests for introducing new concepts from the standard curriculum of her/his pupils. Some times we use iconic languages, but always we see the need of establishing different layers of “complexity” (gain through the creation of procedures or primitives).

Indeed our first and main goal with educational robotics is inquiry based education in primary schools or for even younger pupils (De Michele, 2008). We also observe that teaching programming is a sort of side effect contributing to introduce principles of Computer Science in education according to the 2006 ACM curricula.

A constructivistic IBSE using programmable robots

In Rocard’s 2007 Report we read: “By definition, inquiry is the intentional process of diagnosing problems, critiquing experiments, and distinguishing alternatives, planning investigations, researching conjectures, searching for information, constructing models, debating with peers, and forming coherent arguments (Linn et al., 2004). In mathematics teaching, the education community often refers to Problem-Based Learning (PBL) rather than to Inquiry Based Science Education. In fact, mathematics education may easily use a problem based approach while, in many cases, the use of experiments is more difficult.” In activities where pupils program mini robots, experiments on mathematics concepts are also possible as we sketch in this chapter. We also emphasize the constructionistic methodology suggested in approaching robot programming activities in schools.

Using natural and programming languages at different ages/levels

In our activities “not yet writing” children use the BeeBot mini robot by TTS (described at http://www.tts-group.co.uk/Bee-Bot). Children can move the BeeBot (back, forward, left, right) by pushing buttons on its back. Didactical goals to reach are counting, comparing quantities, problem solving and finding one or more strategies to make the BeeBot reach a given goal position. BeeBot is also quite used in schools where teachers are somehow frightened of ‘programming through a computer’. We try to overcome this fear by offering the programming language and the Integrated Development Environment specifically conceived for pupils and teachers described in (Demo, 2009). The textual and Logo-like programming language, called NQCBaby, can be used for programming the RCX and NXT Lego bricks. Educational itineraries introduce the programming language by means of different layers as described in (Demo, 2007). Commands of the first language level, called NQCBaby0, correspond to the BeeBot buttons commands that pupils have used before writing, i.e. in k-2, or, in higher school grades, if robot activities begin later. Each next layer is introduced for making possible conceiving and implementing activities having new educational goals with respect to previous activities or for a constructive introduction of some programming commands or concepts. As an example, NQCBaby2 is NQCBaby1 plus commands allowing pupils to make experience of synchronous activities the robot can perform such as playing a song and/or rotating a palette and/or switching off/on a light while moving.

NQCBaby1 = NQCBaby0 ∪ {forward(n), backward(n), left(n), right(n)}

NQCBaby2 = NQCBaby1 ∪ {play(..), switch-on(light), switch-off(light), clock-rotate(n), counterclock(n)}

When the touch sensor is introduced we need commands to specify where it is connected and to check it. The language becomes NQCBaby3 containing primitives for moving while paying attention whether an obstacle to the robot movement has been found.

NQCBaby3 = NQCBaby2 ∪ {port-1 is touch, forward-always, repeat-always, if-touches, end-if}

In our educational itineraries a further step is covered, i.e. a next layer of the language is introduced, to allow pupils designing new kinds of robot behaviors and teachers conceive each activity, even competitions, as having a specific didactical goal synchronized with her/his pupils curriculum.

Past experiences have interested particularly mathematics teachers. In primary schools, activities have been developed concerning direct and inverse proportions, fractions, geometrical shapes. Also discussions on which of the proposed solutions to a given problem was more general have been carried out in a fifth grade. In junior secondary schools algebraic expressions have been introduced and motivated as a way to express the length of the path (or part of it) a robot covers while running a given program (Demo, 2010).

Procedures or block construction: pure constructivism/constructionism

One of the crucial aspects of the constructivist/constructionist approach is the tendency of assuming a new cognitive equilibrium, with positive balance, after a phase of assimilation of new knowledge stimulated by a problem (or a simple question) followed by a phase of accommodation concerning the conceptual structuring of the new information. These processes are often accompanied by the building of abstractions able to generalize from the specific details of the addressed problem. Dealing with programming (in particular robot oriented programming), these abstractions have a linguistic counterparts represented by a parameterised function or method. This requires that the teacher, after the resolution of a specific problem, helps the student to formalize the general aspects of the problem recognizing its parametric entities and transforming functionally the solution; in fact they are creating procedures or primitives that will be re-used to construct new primitives. More details on our experience on this matter can be found in (Arlegui et al, 2009).

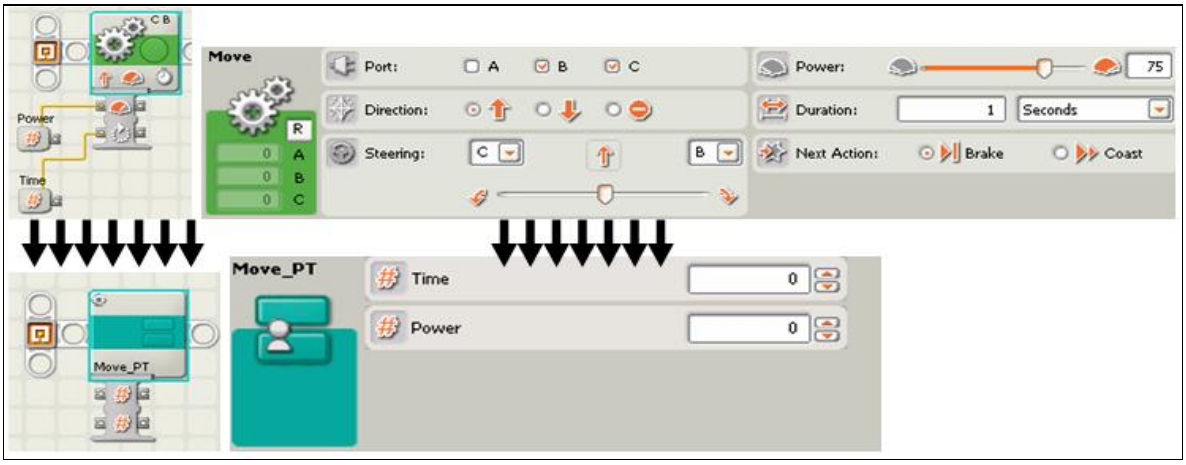


Figure 4. An example where we "simplify" the Move block using a new procedure Move_PT, which is a "rewriting" of the Move block but with only 2 parameters (Power and Time). We are constructing new vocabulary (new blocks) that can be re-used later, for example to define a Move_VT (velocity, t).

For example, in the iconic NXT-G language this corresponds to define a *MyBlock* sub-command, whereas in the textual NXC language this means to define a traditional function with formal parameters. A typical problem of this class is move the robot on a straight line for a given time at a given Power (of the motors). Figure 4 shows how to construct a “new block” able to be re-used.

Finding “real problems” to solve when following at the same time regular curricula

At the secondary junior level students have more awareness of the real world and its challenges. Some real problems and situations offer several cues for proposing, with suitable simplifications, effective robotic-enhanced experiences. For example the control of a supervised metro line can be reduced to a very simple motion planning problem, i.e. a robot moving on a straight line and forced to stop on positions with a fixed distance between one another. After having solved this problem, starting from a basic knowledge of the involved motion parameters (speed, time and distance), the problem can be generalized with not uniformly spaced metro stations.

At the secondary senior level educational robotics can explicit all its potential provided the robotic platform can ensure a sufficient precision, both mechanical and for the controlling program. In spite of its apparent, relative simplicity, Mindstorms NXT provide a good motion control and a complete series of commands able to get enough reliable data from the environment through its sensors. Some briefly explained examples will convince the reader of the wideness of possibilities.

When studying the pivoting of a robot with two wheels and one motor per wheel, when you move in one direction only one of the two motors, i.e. the pivoting is around the still wheel, the student must infer the angle performed by the robot from the rotational angle of the moved wheel. This requires the application of simple relations based on measuring angles in radians, a concept that can be more easily understood when applied to such a practical problem.

If you mount the sonar sensor on a motor, you can rotate the sensor to evaluate distance and angles on a plane. In figure 5 this layout is used to indirectly calculate the distance separating two objects: the distances *d1* and *d2* are measured by the sonar whereas the angle α is given by the encoder integrated in the servo-motor differentiating the angular positions where the distances are evaluated. Now applying the cosine theorem the student can obtain the unknown distance d ($d^2 = d1^2 + d2^2 - 2 \cdot d1 \cdot d2 \cdot \cos\alpha$).

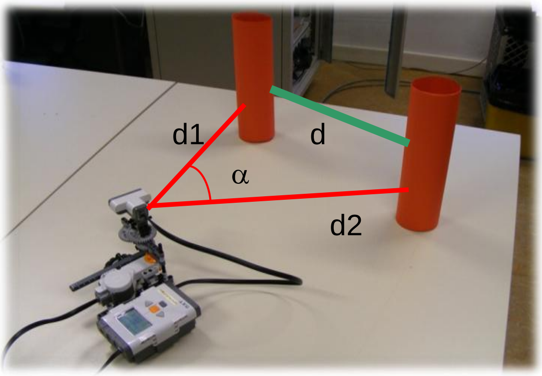


Figure 5. An application of the cosine theorem

The self-positioning problem is a well known problem in autonomous robotics. It can effectively be simplified with the following layout: the robot is initially put in an unknown position in front of, and not so far from, two objects; these latter two are in known 2-D cartesian positions. Measured the two distance d1 and d2 from each object and the robot using the sonar sensor, the student can argue that the position of the robot is for sure in one of the two intersection of the two circles centered in each object position and having respectively d1 and d2 as their radius (a simple heuristic helps to recognize which one of the two is correct). The intersections can be calculated applying a not trivial system of two second order equations.

If you make the robot emit a fixed frequency periodic sound and pass in front of a microphone connected to a PC for analysing the perceived sound, you can experiment the effects of the Doppler phenomenon. The difference of frequencies when the robot is approaching the microphone and when it is going away from it results limited to some Hertz, therefore the analysis must be performed with suitable tools on the PC but it can give a sufficiently precise estimation of the robot speed (in formula, $f_{perceived} = f_{emitted} \cdot v_{soundInAir} / (v_{soundInAir} - v_{robot})$).

PBL/IBSE, constructivism, competences, European framework

In this chapter we would like to relate the concepts of constructionism and IBSE with an aspect that has recently assumed more and more a strategic meaning in the actualization of curricula, namely the so-called 'competences'. This term has been used in the literature with a wide spectrum of meanings depending on the context. From all these different definitions it may be argued that the competence is an integration of knowledge, practical ability, meta-cognitive and methodological capabilities, personal and social capacities. More recently the term has been related to a unique view of the human potential when stressed by real problems. In this view the emphasis is given to the effective mobilization of the person in front of problems: therefore, a traditional set of “discipline-oriented competences” is now enriched by set of “generic competences” giving the measure of a 'know how to act' that involves the whole personal sphere of knowledge and skills.

In a competences-centred teaching/learning the transmission of knowledge is substituted by offering students opportunities to solve problems together with the assumption of duties and independent initiatives. This implies to design inter-disciplinary work units where different competences (both specific and generic) are worked. The main change is the way of how organizing the entire teaching sphere. Teachers also need to mature in the awareness that their discipline promote the construction of competences.

The competence does not exist until it is practiced in a meaningful context. Even if you can get a better application of knowledge in a real context, and from this point of view, the combined school-work initiatives provide conditions particularly effective; nevertheless laboratorial activities at school can boost the students’ autonomy in order to cope with complex situations.

This view is also consistent with the more recent guidelines issued by the Council and European Parliament. The conclusions to the work of the Lisbon European Parliament in 2000 show 3 main objectives: the definition of key competences, the raising of educational levels and the enlargement of lifelong education, the recognition of non formal and informal learning in the formal ones. The recommendation (Recommendation 2004) points out the necessity to recognize both formal and informal learning because they both contribute to build a strong competence. In (Recommendation 2006) the eight key competences (Communication in the mother tongue, Communication in foreign languages, Mathematical competence and basic competences in science and technology, Digital competence, Learning to learn, Social and civic competences, Sense of initiative and entrepreneurship, Cultural awareness and expression) are officially declared as key aspects of the European citizenship.

Even the successive European Qualifications Framework (EQF) (EQF, 2008) emphasizes the need to promote the validation of non formal and informal learning, especially for students who manifest difficulties applying traditional evaluation criteria. It is remarkable that in this recommendation, along with definitions of knowledge and skill, the competence is seen as a proven ability to use knowledge and practical skills in operational situations, but also personal, social and methodological abilities, and with the application of personal responsibility and autonomy. As you can see this is a definition that includes a significant ethical value: in fact responsibility and autonomy give substance to an active citizenship and to social inclusion.

Teaching-Learning processes designed for acquiring competences have a very close relationship with the constructivist/constructionist approach. Indeed laboratorial activities, autonomy, accountability, creation of artifacts, inductive and collaborative learning, are perfectly in accordance with the competence-oriented view as ‘how to act knowing’. The implementation of such approach through IBSE explorations in coherent sequence of problems to be solved put learning activities strictly in connection with the real world and its complexity and variability, though usually in simulated environments.

When robotics is concerned, we should add the conclusion that it promotes and stimulates any fundamental competence thanks to aspects like multidisciplinary, attractiveness, building of ‘intelligent’ artifacts, interesting links to literature and cinema, etc.. Once again, the emphasis shifts to a thoughtful design of teaching units, within which the construction of the robot and its programming are only instrumental aspects necessary to achieve educational purposes related to the competences the teacher wants to stimulate.

Conclusions

Figure 6 summarizes all the ideas expressed in this paper and specifically shows that, in spite of the augmented role of students in all the active phases of the PBL/IBSE, the teacher maintains a crucial, central position as facilitator and mediator.

We believe that IBSE, PBL, constructivism/Constructionism and technological artifacts can be used to promote a scientific way of thinking among the students/pupils with new, relevant potentialities. This can be done by teachers within the schools and out of the schools (formal and informal learning). They need for that a double support.

First of all we have to set up an adequate teacher training plan (as it is being done in the case of robotics activities) within the schools (in our case trainers from the university are teaching trainees, the teachers at the school, in order to help them to integrate the design of robotic-enhanced experiences in their institutional curriculum and to teach this to the pupils).

Then once the teachers are confident with the tools they can use and with the methodology they can apply, it is the moment to link the school activities with real problems from the real world. At this point the society, through institutions like Public (Science related) Museums or other Public Institutions that promote creativity and innovation among young students, can “give” to the school real problems with real data to be solved, and at the same time the school can “transform” these problems in order to produce the necessary didactical approach (the didactic transposition). These two “bi-directional” steps could enrich both roles; the school gains “real problems from out of the school” to be used with a Constructivist Project Based Learning strategy using IBSE methodologies; and the Public institutions like Museums or others gain a more didactic way of showing/describing/sharing their “contents”.

We think that this is the way we can educate our future (XXI century) citizens, linking schools (at any level) and society, in harmony with the European Qualifications Framework.

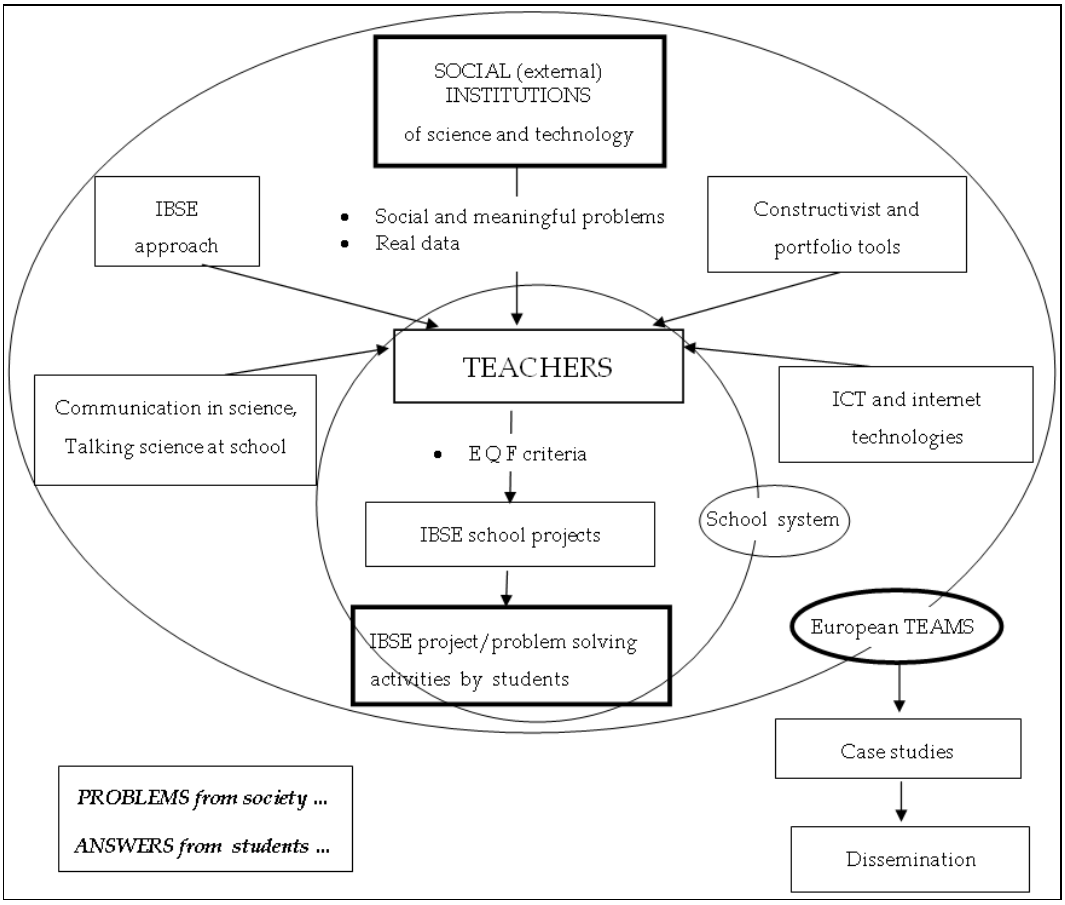


Figure 6. Linking Educational institutions and Social Cultural & Scientific institutions for Educational purposes to provide, share and disseminate "real problems" to be solved using a constructivist IBSE by our Teachers & Students

References

Arlegui, J. Fava, N. Menegatti, E. Monfalcon, S. Moro, M. Pina A. (2009) Teacher training in the Scientific field through robotics activities: some experiences from Italy & Spain. In *Proceedings of INTED 2009, International Technology, Education & development Conference,* Valencia, Spain.

Chevallard, Y. (1999) L'analyse des pratiques enseignantes en théorie anthropologique du didactique, *Recherches en Didactique des Mathématiques, Vol 19, nº 2, pp 221-266.*

Demichele, M.S., Demo, G.B., Siega, S. (2008) "A Piedmont SchoolNet for a K-12 Mini-Robots Programming Project: Experiences in Primary Schools", Proc. TERECoP Workshop "Teaching with robotics, Conference SIMPAR 2008, Venice, November 2008.

Demo, G.B. and Marcianò, G. (2007) "Contributing to the Development of Linguistic and Logical Abilities through Robotics", Proc. EuroLogo 2007 Conf., August 2007, p.46.

Demo, G.B., "Robot Programming Integrated in a Junior High School Curriculum", Proc. Informatics Education Europe IV Conf., Freiburg, Nov.5-6, pp 56-63, 2009.

Lajoie, S. P., & Derry, S. J. (1993). *Computers as cognitive tools*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Nickerson, R. S. (1988). Technology in education in 2020: Thinking about the not-distant future. In R. S. Nickerson & P. P. Zodhiates (Eds.), Technology in education: Looking toward 2020. Hillsdale, NJ: Lawrence Erlbaum Associates.

Rocard. M. et al. (2007) Science Education now: a renewed pedagogy for the future of Europe, EUR22845, ISBN – 978-92-79-05659-8, European Communities.

Rubin, A. (1993) Video laboratories: Tools for scientific investigation. *Communications of the ACM*, 36(5), pp 64-65.

http://www.griffith.edu.au/__data/assets/pdf_file/0009/168192/IBSE-PLP-FlyerRegistration.pdf

[Recommendation 2004] The Council conclusions on common European principles for the identification and validation of non-formal and informal learning of 28 May 2004.

[Recommendation 2006] RECOMMENDATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 18 December 2006 on key competences for lifelong learning

[EQF] THE EUROPEAN QUALIFICATIONS FRAMEWORK FOR LIFELONG LEARNING (EQF), 2008.

Creative Scratch Robots for Under $30

Liu Huan – Computer Science, MIT
Aditi Dugar – Mechanical Engineering, Cornell University
Jonathan Heckerman, Carolyn Corbet , Jeronimo Arranda – Intern Developers
Build-It-Yourself, 269 Pearl Street, Cambridge, MA 02139,
Contact: John Galinato, 617-875-1960, john@build-it-yourself.com

Introductory description and overall goals:

Would you like to build playful robots for under $30 dollars? Robots that dance, play music, tell jokes, throw candy and more.

Figure 1 – Scratch robots made from premium quality junk

Build-It-Yourself aims to make engaging, playful robotics accessible to all. Premium quality junk is used to build creative, robotic sculptures that quickly attract attention.

Method:

Build-It-Yourself Scratch robots have the following components:

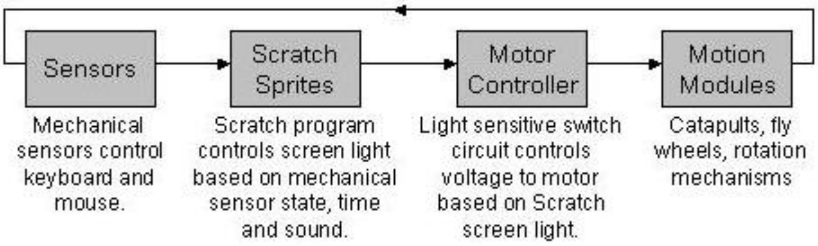


Figure 2 – Components for Scratch Robots under $30

Expected Outcomes:

Instructions to build sensors, motor controller and motion modules are posted online.

Live Webcast workshops are run by MIT, Harvard and Cornell engineering students.

Invention Universe enables members to share creations online.

www.build-it-yourself.com

KEY WORDS:

Scratch; robotics; project-based-learning; construction-system

For reference only

Scratch Robotics for Under $30 (Workshop proposal extended description)

Workshop parameters:

1. Participants will learn how to create engaging robots from 'premium quality junk,' $30 worth of Radio Shack electronic parts and Scratch (freeware from MIT).
2. Directed to education publishers, educators and kids 8 – 18.
3. Up to twelve (12) participants.
4. Two (2) hours (or more)

Equipment required:

1. Screen projector

The Demand:

There is a strong demand for robotics programs.

1. Many are predicting that robotics will be the next technology to significantly change the world. (Bill Gates, Scientific American, Jan 2007)
2. Robotic projects engage kids in a way that sets the stage for multi-discipline learning.
3. Robotics can stir the imagination of almost anyone (young, old, boy, girl, rich, poor) unlike other complex technologies.
4. There are so many 'black boxes' in our lives that sometimes, we fail to appreciate the basic art and science that underlies how things work.

The Problem:

How can we engage as many kids as possible in robotics? Many kids have access to computers but cannot afford expensive robotic construction kits. Even when such kits are purchased, they may not have long product lives. Not only may expensive parts become obsolete, but also teachers may need to be re-trained with new lesson materials.

The Mission:

Build-It-Yourself aims to make engaging, playful robotics accessible to all.

1. Webcast workshops are lead by engineering students from leading universities.
2. Instructions and videos are posted online.
3. Members can share their creations online in Invention Universe.

Figure 3 – Scratch Robots Construction System

The Build-It-Yourself robotic construction system is inexpensive, transparent and extremely flexible. Build-It-Yourself robots have the following components:

1. Almost any program on any device that has a screen display (the controller)
2. Sensor modules built from common craft materials can detect a coin or movement. The sensor modules are mechanically linked to the keyboard and/or a mouse where they can be interpreted by a program that can control the screen display.
3. A simple, electronic breadboard circuit detects light intensity on an LCD or LED screen and drives a geared motor. Scratch or almost any program controls the computer screen display based on keyboard, mouse, sound and time inputs.
4. Motion modules, also built from common craft materials, can make a robot dance, throw candy, and beat drums.
5. Speech recognition and audio clips, inherent in Scratch, can add an exciting dimension.
6. Premium quality junk (which kids can collect from a recycle bin, toy collection, or coffee shop) is used to build creative, robotic sculptures that quickly attract attention.

Project Management Site:

www.build-it-yourself.com/biy-projects/proj-scratch-robotics/index-scratch-robotics.html

The PowerPoint presentations we use to introduce kids to these life skills are posted at:

www.build-it-yourself.com/support/support-projectware/biy-projectware-presentations.html

The Build-It-Yourself mission is to inspire and guide kids to use technology creatively and constructively. In the process, we introduce multi-discipline, 21st century skills including collaboration, computer programming, modular construction, the art of presentation, and an appreciation for the social consequences of technology.

Creative Scratch Robots for Under $30

Liu Huan – Computer Science, MIT
Aditi Dugar – Mechanical Engineering, Cornell University
Jonathan Heckerman, Carolyn Corbet , Jeronimo Arranda – Intern Developers
Build-It-Yourself, 269 Pearl Street, Cambridge, MA 02139,
Contact: John Galinato, 617-875-1960, john@build-it-yourself.com

Would you like to build playful robots for under $30 dollars? Robots that dance, play music, tell jokes, throw candy and more?

Figure 1 – Scratch robots made from premium quality junk

Build-It-Yourself aims to make engaging, playful robotics accessible to all. Premium quality junk is used to build creative, robotic sculptures that quickly attract attention.

Build-It-Yourself Scratch robots have the following components:

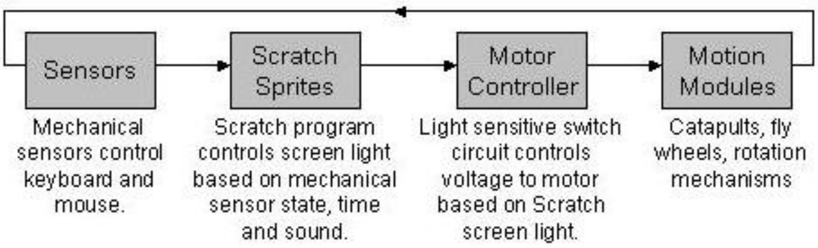


Figure 2 – Components for Scratch Robots under $30

Instructions to build sensors, motor controller and motion modules are posted online.

Live Webcast workshops are run by MIT, Harvard and Cornell engineering students.

Invention Universe enables members to share creations online.

www.build-it-yourself.com

Keywords:
Scratch; robotics; project-based-learning; construction-system

Constructionism, Complex Thinking and Emergent Learning: Preschool Children Designing and Programming

Eleonora Badilla-Saxe *eleonora.badilla@ucr.ac.cr*
Academic Evaluation, University of Costa Rica

Abstract

In this short paper I report, from three interacting perspectives, a research experience about learning in a constructionist environment. From one of the perspectives, preschool children learn while they collaborate in designing relevant contexts (microworlds) and constructing and programming behaviours for physical models, all using reusable materials. On another, two groups of my graduate students from the School of Teacher Education at the University of Costa Rica learn while planning a constructionist approach to support children in the construction and programming of their microworlds and creatures; identifying, as well, evidences of their learning according to the official preschool curriculum in Costa Rica. From yet another perspective, while supporting my students with their research project and challenging them to go beyond what has been planned, I learn about complex thinking, and learning as an emergent phenomenon that transcends what has been foreseen in the curriculum.

Keywords

constructionism, microworlds, objects-to-think-with, preschool, robotics, design, emergent learning, complex and eco-systemic thinking.

Chapter I Conceptual Framework

I.1 On Constructionism

Constructionism is Seymour Papert's pedagogical proposal, derived from the Constructivist theory of Jean Piaget. Papert's central idea is that learners should play an active role in their learning. Therefore, the purpose of education would be to empower them so that they may assume this leading role, designing their own projects and building their knowledge. In that regard, he says: *"…the best learning will not come from finding the best ways in which teachers can instruct, but from providing the students with the best opportunities for them to construct"* (Fabel, 1990). This is the premise that will rule the learning process from the constructionist approach, which supposes that people possess a natural ability to learn through experience, creating mental structures that organize and combine the information and experiences acquired through daily life. In Papert's opinion, knowledge construction often occurs in a especially fruitful manner when the person learning is consciously involved in a more public construction, that is, a construction that can be exhibited, discussed, examined, proved, or admired. This public construction ranges from a sand castle or a Lego house, to a webpage or a computer program (Fabel, 1990.). It is in this sense that Papert warns that it is not enough to suggest to students they take charge of their learning by assuming an active role, but rather that the learning environment and tools made available to the learners are fundamental. That is to say, society and culture are largely responsible, for they must provide the resources needed to learn. In his opinion, computers are particularly powerful tools, since he starts from the hypothesis that much of what we now consider too formal or abstract to be understood at early ages, will be learned more easily when the learners perform within a computerized rich world. That is why he focuses on the process of inventing objects-to-think-with within a new kind of learning environment, which supposes the interaction between children and computers. Thus, he affirms that *"… we can free ourselves from the superficial and pragmatic considerations that once ruled with regard to what knowledge should be learned and at what age"* (Papert, 1987, p.69). Papert is especially interested in the role played by physical objects on thought development, which is why he believes that an object-to-think-with may be used by the learner to think about other things, while reflecting on the construction of the object. He affirms that we create our knowledge of the world by creating objects, experimenting with them, modifying them, and studying how they operate. In keeping with Piaget, Papert considers that the learning process cannot be removed from the lesson itself. In this sense, the objects-to-think-with cannot be removed from the learning process itself, nor from the content learnt, thus becoming an inherent part of knowledge construction. Regarding the environment where knowledge is constructed supported by the objects-to-think-with, he developed and coined –together with Marvin Minsky– the concept of "microworld", as a model to create representations of an immediate reality over a subject matter, which will be refined or polished by the students, emerging from a starting point that allows them to create their own "extensions". In that sense, a microworld constitutes a tiny *constructionist* world, wherein the learner can explore choices, prove hypothesis, and discover facts that are true in relation to such world. It differs from simulation in the fact that the microworld is a real world (even if virtual), instead of just a simulation of another world.

I.2 On Complex Thinking

As the sociologist Edgar Morin (2008) points out, the kind of thought in which knowledge is fragmented, compartmentalized, monodisciplinary and quantified leads to a blind intelligence, insofar as the normal human aptitude to connect knowledge is sacrificed to the no less normal aptitude for differentiating it. He alternatively propounds complex thinking as a method to know and know the knowing process. Complex thinking is the organizer of the organization we use to represent the world; it is part of our thoughts, our ideas, and our scientific theories. In Morin's opinion, (2004) complex epistemology is the knowledge of knowledge, and knowledge is a "spiral adventure" that has a starting point but no end, tirelessly performing concentric circles. However, it is important to clarify that complexity does not lead to the elimination of simplicity, nor does it reach *completion.* Regarding simplicity, Morin says: *"… while simplifying thinking disintegrates the complexity of what is real, complex thinking integrates, as far as possible, the simplifying ways of thinking; rejecting, nevertheless, the mutilating, reductionist, one-dimensional and finally blinding consequences of a simplification deemed a reflection of what could be real in reality"* (Morin, 2006). And concerning *completion,* he says*: "Certainly, the ambition of complex thinking is to account for the articulations between disciplinary dominions disjoined by dispersing thinking (one of the main features of simplifying thinking); the latter isolates what it separates, and hides everything that rebinds, interacts, interferes. In this sense, complex thinking aspires to multidimensional knowledge. But knowing, from the start, that complex knowledge is impossible: one of the axioms of complexity is the impossibility, even theoretical, of an omni-science"* (Morin, 2006).

In that sense, we may explain to ourselves complex thinking based on the etymology of the term "complexus", understood as that which is woven together or conjointly interwoven, and in that context, the emergence or the emergent of the interrelations between parts and properties will become relevant to diverse authors and in different fields. In the Wikipedia Free Encyclopedia[1], we find that emergence refers to those properties or processes of a system not reducible to the properties or processes of its constituent parts. It is closely related to the concepts of self-organization and supervenience, and defined in opposition to the concepts of reductionism and dualism. One of the most common examples of an emergent phenomenon is the mind: it emerges from the interaction distributed among diverse neuronal, corporal and environmental processes, but it cannot be reduced to any of the components that participate in the processes. From the epistemological perspective, emergence refers to the impossibility of the observer to predict the appearance of new properties on the system in question. Enrique Margery (2007) affirms that the emergent is an unexpected answer or reaction, not anticipated, produced as a result of the interaction between the parts of a whole.

For our purposes, we accept that learning is an emergent phenomenon. It arises from the interaction distributed among different processes –neuronal, corporal, emotional and environmental– but it cannot be reduced to any of the components that take part in the processes. In that context, we should understand that most learning is unexpected, often impossible to foresee, and that the contents of those learning are both simple and complex, although the complex ones are not mere aggregates of the former.

[1] http://es.wikipedia.org/wiki/Emergencia_(filosof%C3%ADa) recovered on June 29, 2007

We acknowledge some reference points posed by the Brazilian researcher Maria Candida Moraes (2008), which guide our steps towards understanding learning in constructionist environments as an emergent phenomenon, unforeseeable and unrepeatable. In Moraes' opinion:

- Complexity is dynamic and therefore, processual. Dynamic processes are unforeseeable and creative, with the ability to go beyond the known or foreseen horizon.
- Each experience is unique, does not repeat itself and nontransferable, since time does not act retrospectively over matter, and neither does the present over the past. This understanding warns us about the importance of being aware of the important moments in life, both in the personal sense as in regard to knowledge and learning.
- Phenomena are multidimensional, and perceiving their multi-causality and multiplicity of effects is necessary to gain a more appropriate understanding. To think in a complex way is to understand relations, connections and links.
- There is no single objective reality independent from that which is observed, but rather multiple realities, and which of those realities will be revealed depends entirely on the observer.
- There are important and different types of knowledge. The interpretations of each individual concerning reality are different.

Chapter II Learning in a Constructionist Environment: Preschool Children Designing and Programming Creatures

The research experience "**Learning in a Constructionist Environment: preschool children designing and programming creatures"** was carried out at the University of Costa Rica, in 2005-2008. The process was developed in two interrelated stages, as the final project of graduate students to apply for the Licenciate degree on Preschool Education. The project was a reflective practice, where the students-researchers viewed themselves as part of the process, particularly reflecting upon their own learning process. In my case, as researcher director of both stages, I also considered myself as part of the process, reflecting as well upon my own learning process; thus, I present my own conclusions in this report, concerning the experiences I lived with my students and the preschool children.

A Constructionist methodology: In the first two years of the research, a group of students from the Preschool Education degree course designed and tested a constructionist methodology, so that 5 and 6-year-old children would learn in a creative and meaningful way, using physical-digital materials to build programmable creatures. The building of these programmable creatures became a means for learning, and not the end itself. The activity of the children was conducted in the context of a preschool education classroom. The children called their learning activity ***El divertidor***, "The Entertainer". Using cardboard boxes and disposable materials, they assembled, inside their regular classroom, what we call (based on Seymour Papert)[2] a microworld: a haunted house, a castle, a space ship…

---

[2] Seewww.papert.org, recovered on August 1, 2008.

And for that context, they designed, built and programmed a creature: a guardian monster, a flame thrower dragon, or a Martian detector (an object-to-think-with). They only used the digital materials strictly needed, making the most out of plenty of disposable and easy-to-get material. Collaborative work and family involvement were promoted. The constructionist methodological proposal to ensure a learning environment where preschool children would learn how to build programmable creatures (objects-to-think-with) within a microworld context, included the following components, not performed sequentially but recurrently.

1. **Social Interaction**

Figure 1
Social interaction must be an important part of a preschool constructionist learning environment

2. **Constructing a physical microworld**

 *Figure 2*	 *Figure 3*
 *Figure 4*	*Using a large cardboard box and colored papers, watercolors and other building materials, the children designed and created their microworlds: a castle, a bus, and a haunted house.*

3. Predicting the creature's behavior with "natural language"

 *Figure 5*	*"There is a monster in the haunted house. If someone enters, the monster spins many times to frighten him/her."*

4. Exploration of physical-digital material

Figure 6

The children explored physical-digital materials created by the teachers. They discovered the digital materials that make programming feasible.

5. Programming the creature by way of artificial language

 *Figure 7*	 *Figure 8*
Exploration of language programming icons, using printed cards	*Programming the "monster" in the computer using programming language*

6. Comparing natural and artificial languages: verification

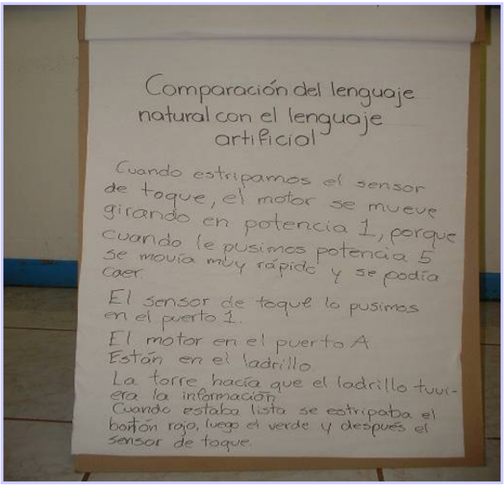


Figure 9
Comparing languages

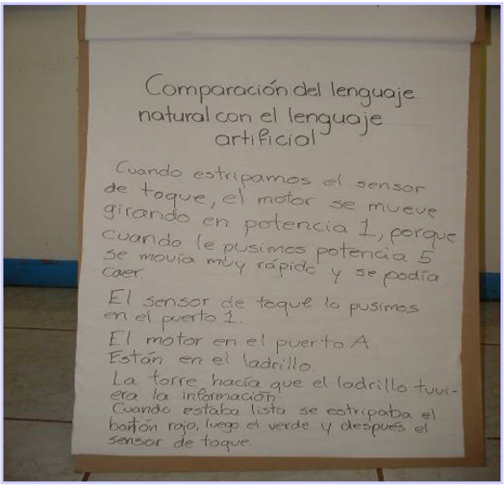


Figure 10
Comparing languages

During this stage, and in a process of self-assessment, the children verified whether the programming of their creature by means of digital language corresponded to the behavior they had foreseen with natural language.

7. Exploration of the creature's behavior in the microworld

Figure 11
The creature in the microworld

8. Reconstruction of the process

Figure 12
What happened? How did we do it?

The children immersed themselves in the microworld, in order to export the behavior of the creature they had designed and programmed. In other words, they became part of the microworld. They had the chance to test their programming, assess whether it answered their expectations, and return to the programming process in order to modify it.

For instance, the programming of the "monster" actually worked, since the moment the light sensor was activated, the motor started and the creature revolved. But it did so too fast, and was finally thrown into the air. The children had to go back to the programming stage in order to modify it, identify what criteria should be changed, test their new programming and repeat the process, until they were satisfied with the monster's behavior. Self-assessment was encouraged, both in the stage of exploration as in the reconstruction process.

With regard to this first stage of the research, and to complement the conclusions of the students-researchers concerning the process with the children, I can ascertain that:

By definition, learning is an emergent phenomenon, and according to Maria Candida Moraes, the learning environment should be: *"… enjoyable, rich in meaningful and off-key elements, and capable of rescuing the joy and pleasure of learning…* It must promote *the construction of knowledge, the creation of identities, and the development of cultural and social practices, holding complexity as one of its main foundations."*

And with regard to the digital technologies, I agree with Seymour Papert when he says: *"When we think about technology in education, we must not expect it to have an effect. We should rather reflect on the opportunity offered by technology to rethink what learning is, to rethink education."*

Emergent Learning: During the last two years of the research, another group of students-researchers of the degree course mentioned above put into practice the design of a learning environment, enjoyable and rich in meaningful and off-key elements, as well as the methodological proposal previously described, while trying to identify, at the same time, the mathematical skills and social conducts displayed by the children as they designed, assembled, programmed, tested, corrected, explored, discussed, built, played, thought, doubted, reflected and had fun. This time, the students suggested this assessment (besides the self-assessment that the children would do later on), since they deemed important to prove that, by using these new technologies, the children could also learn the contents included on the official curriculum for the preschool level in Costa Rica. This second part was conducted at a public kindergarten located in a semi-rural area, throughout five learning sessiones.

On these five sessions, the 5 and 6-year-old children showed, in July (in half the time scheduled by the country's official school calendar), 53 of the 68 mathematical skills anticipated by the Syllabus for the Transition Cycle (preschool) enforced by the *Ministerio de Educación Pública* MEP (Ministry of Public Education), in Costa Rica for the entire school year. In the social sphere, the children showed social conducts predicted by the aforementioned Syllabus: cooperation, expression of likes and preferences, and tolerance to frustration, among others.

Nevertheless, I had suggested to the students to not limit themselves to observe whether the children showed the mathematical skills and social conducts included in the official Syllabus, but to observe signs of other learning as well.

Observing the children work, and using regular evaluation instruments for preschooler, they identified, among others, the following:

- **Solving exercises and problems within time limits** (included on the Syllabus for 5th grade);
- **Time estimation and measurement, concept of movement, implementation of concepts such as speed, velocity, distance, movement and force** (7th grade);
- **Power** (10th grade).

In the social sphere, they showed social conducts not included in the official curriculum for preschool levels, for instance:

- **Working as a team;**
- **Collaborating on the development of a project;**
- **Respecting the choice of the majority.**

Chapter III Lessons Learned

To complement the conclusions issued by the students-researchers, I came to the following conclusions:

- Being that learning is an emergent phenomenon, it is important to bear in mind that the learners will construct other knowledge not necessarily related to the purpose of the curricular design;
- Being that learning is an emergent phenomenon, it is important to bear in mind that the students will express simple and complex knowledge, and that the complex will not be mere aggregates of the former; therefore, it is important to explain the relations and connections that exist amongst what was learnt;
- In a constructionist environment, rich in meaningful experiences and where the students are allowed to take an active role in their own learning process, unforeseen knowledge will emerge; therefore, teachers and researchers should be aware of this reality, in order to detect and record (not necessarily grade) the knowledge resulting from the multiple interactions that intercross within a learning environment.
- Being that the circumstances for knowledge verification are not replicable, and that the different interpretations of the subjects on one and the same reality should be taken into account, it is important to understand that, in each situation, different knowledge will emerge.
- If a constructionist environment is allowed, wherein learners take charge of their own process, the use of digital materials in learning promotes the building of concepts at earlier ages.
- From the perspective of education in general, and training of teachers in particular, these conclusions have enormous implications concerning the way we view our students; the manner in which we should help teachers so that they really “see” their students; the approach used to design curricula and study programs. And above all, it forces us to rethink our beliefs concerning learning and assessment of learning.

References

Avendaño, Mario et al, (sf) **Planeamiento didáctico, mediación pedagógica y evaluación de los aprendizajes para la atención de la diversidad: un enfoque intergral**, Ministerio de Educación Pública, San José, Costa Rica

Badilla-Saxe, Eleonora, 2006, **Proyecto de Trabajo Final de Graducación: Interacción, preescolares, átomos y bits**, Facultad de Educación, Univesidad de Costa Rica

Carvajal, Chacón, Enríquez, Rojas y Sánchez, 2005, **Interacción preescolares, átomos y bits: Propuesta para el diseño de un ambiente de aprendizaje para niños y niñas de 5 y 6 años propicio para la construcción del conocimiento a partir de la interacción con materiales físicos y físico-digitales,** Facultad de Educación, Universidad de Costa Rica

Computer Clubhouses http://www.computerclubhouse.org/ recuperada el 01 de agosto de 2008.

Enciclopedia Wikipedia
http://es.wikipedia.org/wiki/Emergencia_(filosof%C3%ADa) recuperada el 29 junio 2008
http://es.wikipedia.org/wiki/Educaci%C3%B3n#Evaluaci.C3.B3n recuperada el 04 de agosto de 2008

Fernández, Obando, Rodríguez, Salazar y Trejos, 2008**, Experiencia Didáctica para la evaluación de conocimiento Matemático y de Relaciones Sociales de niños entre los 5 y 6 años a partir de la interacción con materiales físico y físico-digitales**, Universidad de Costa Rica

Flavel, Aaron, 1990, **Construccionismo**, Fundación Omar Dengo, traducción libre Eleonora Badilla-Saxe.

Goleman, Daniel, 2008, La inteligencia emocional, en Enciclopedia Wikipedia
http://es.wikipedia.org/wiki/Educaci%C3%B3n#Evaluaci.C3.B3n recuperada el 04 de agosto de 2008

Margery Bertoglia, Enrique, 2007, **Las Organizaciones del Siglo XXI**: En Conversatorios sobre Pensamiento Complejo, Vicerrectoría de Docencia, Universidad de Costa Rica

Moraes, María Cándida, 2008, (sp) **Dimensoes motodológicas e suas implicacoes educacionais**

Papert, Seymour, www.papert.org Recuperada el 10 de agosto de 2008

Zamora Zumbado, Denia, 2001, **La evaluación de la niña y el niño en edad preescolar,** EUNED, San José, Costa Rica

Turtle Geometry on a Sphere: a Projected Future for Constructionism

Michael MacFerrin, Brian Hallesy, Garrett Shulman, James Bailey, Neal Robbins, Brandon Shelton, and Michael Eisenberg, *duck@cs.colorado.edu*
Dept. of Computer Science, University of Colorado, Boulder CO USA

Abstract

Historically, the educational philosophy of constructionism has been founded upon the idea that children learn by designing and creating *public, shared* artifacts. Typically, the techniques for making artifacts "public" or "sharable" have involved (e.g.) sending files via email or posting creations (such as graphics or music) on social networks or public websites. Recently, however, another important technique has emerged for sharing certain types of graphical projects– namely, by making use of innovative projection systems to display those projects on large, widely visible, and often unorthodox surfaces. This paper describes a project in which an interactive Logo-style turtle has been implemented on the giant "Science on a Sphere" projected display installed at our University planetarium. Science on a Sphere is a display technology available at numerous museums around the world, and is typically employed to present pre-made animations (e.g., of global weather patterns); but by allowing children to write graphical programs for display on the sphere, programming projects take on an aspect of public performance that is largely absent on smaller computer screens. Moreover, the fascinating mathematical ideas underlying spherical geometry can be naturally represented and explored on this surface. This paper discusses the current implementation of our *Turtle on a Sphere* system and presents a variety of geometric examples making use of the system. We conclude with a somewhat wider-ranging discussion of the potential role of novel projection technologies in constructionist education.

Figure 1. A spherical turtle-drawn flower.

Keywords

Spherical geometry, turtle graphics, projection technology, constructionism.

Introduction: Toward a New Era of Display

In what is likely the most commonly cited description of "constructionism", Papert [9] writes:

> Constructionism–the N word as opposed to the V word–shares constructivism's connotation of learning as "building knowledge structures" irrespective of the circumstances of the learning. It then adds the idea that this happens especially felicitously in a context where the learner is consciously engaged in constructing a public entity, whether it's a sand castle on the beach or a theory of the universe. [9, p. 1]

As Papert notes, *public* or *shared* presentation of one's creations is arguably a central element of constructionism's philosophy of learning by design. It enables learning communities of students to observe and build off one another's work; it gives students the opportunity both for hearing critiques of their own work and critiquing the work of others; and it endows student creations with important elements of "social capital", in that creations may be traded, given as gifts, combined into larger collaborative entities, and so forth. (Cf. also the excellent discussion of these themes in the introductory section of Noss and Hoyles [8].)

Figure 1. A spherical turtle-drawn "flower" pattern.

In our own Craft Technology Laboratory at the University of Colorado, this element of public creation is often reflected in projects centered upon physical artifacts (such as polyhedral paper sculptures, popup cards, or programmable clothing, to name just a few [2]). Tangible creations of this sort lend themselves to certain highly natural–often, almost unconscious–types of sharing, performance, and presentation. In contrast, screen-based artifacts–of the sort typical of "traditional" Logo turtle graphics–tend, historically, to be viewed on small, individual computers. As such a traditional turtle project is viewed by at most a few people at a time, and "sharing" tends to take place in more indirect ways. A student might (e.g.) send a file to a friend via email, or a group of students might, over time, observe a running program over the Web; but in general, running a computer program is not seen as an act of *performance*, done for a sizable audience. The public, shared aspect of such projects is thus necessarily muted.

This paper argues that novel technologies for projection enable us to rethink the element of performance in children's programming. As an example, we describe an innovative software project, *Turtle on a Sphere*, in which turtle graphics can be displayed interactively on a giant museum-based spherical display. Turtle on a Sphere illustrates the ways in which recent developments in projection technology can vastly expand the notion of "sharing" and "presenting" computational artifacts. At the same time–and not entirely coincidentally–Turtle on a

Sphere is a natural means for the introduction of important ideas in non-Euclidean (in this case, spherical) geometry. Turtle on a Sphere projects–such as the multicolored flower shown in Figure 1–are both highly visible public decorations, and encounters with challenging mathematics.

The remainder of this paper focuses on the Turtle on a Sphere project as a single instance of a larger technological development in children's programming. In the following (second) section, we discuss the current implementation of the project and its combination of programming language and graphical user interface. The third section presents several sample Turtle on a Sphere projects, and explains the basic concepts of spherical geometry that underlie those projects. In the fourth and final section, we use the Turtle on a Sphere system as a springboard for a more wide-ranging discussion of related and potential future projects integrating novel projection technologies with children's programming and construction.

Turtle on a Sphere: a Brief Description of the System

The Turtle on a Sphere system is, essentially, an implementation of a "classic" Logo-style turtle intended for use with the "Science on a Sphere" (SoS) display system. Before turning to the implementation of our software, it is worth spending some time on the SoS system itself. The display was developed at the National Oceanic and Atmospheric Administration (NOAA) by Dr. Alexander MacDonald as an educational device for viewing computer-generated graphics on a sphere. As the NOAA website [7] explains:

> Science On a Sphere® is a large visualization system that uses computers and video projectors to display animated data onto the outside of a sphere. Said another way, SOS is an animated globe that can show dynamic, animated images of the atmosphere, oceans, and land of a planet. NOAA primarily uses SOS as an education and outreach tool to describe the environmental processes of Earth. [7]

Many museums and institutions around the globe now include the SoS display–including (among others) the Lawrence Hall of Science in Berkeley, the Bishop Museum in Honolulu, the Minnesota Museum of Science in St. Paul, and the Centre National d'Etudes Spatiales (CNES) in Strasbourg. (See [6] for a more extensive list; our project was conducted with the display at our University's Fiske Planetarium in Boulder, Colorado.) The scale of the display–six feet in diameter–and the high resolution of its graphics are both remarkable, as can be seen in the photograph in Figure 1. In practice, the display is implemented with a set of four video projectors positioned around the spherical surface; these projectors are controlled via a software system that translates portions of the spherical display into individual commands for each projector. The net effect, then, is for the four individual projections to combine seamlessly into a single unified spherical animation. In principle, the same sort of technique could be used to project animations onto surfaces such as a cube, cylinder, or cone; we will return to this idea toward the end of the paper.

By searching for photographs or videos on the Web, one can see an extensive variety of Science on a Sphere projects; generally, these projects involve the presentation of "pre-canned" animations or graphics–e.g., an animated weather map, or a display of the surface of the moon. Such projects are unquestionably both gorgeous and educationally worthwhile, but in these cases the students' role is simply to watch and admire the work of (usually unseen) professionals. In contrast, the intent of our Turtle on a Sphere project is to give youngsters a chance to write interactive programs for the sphere, creating aesthetically appealing designs, sharing those designs with viewers, and experimenting directly with the mathematical ideas of spherical geometry.

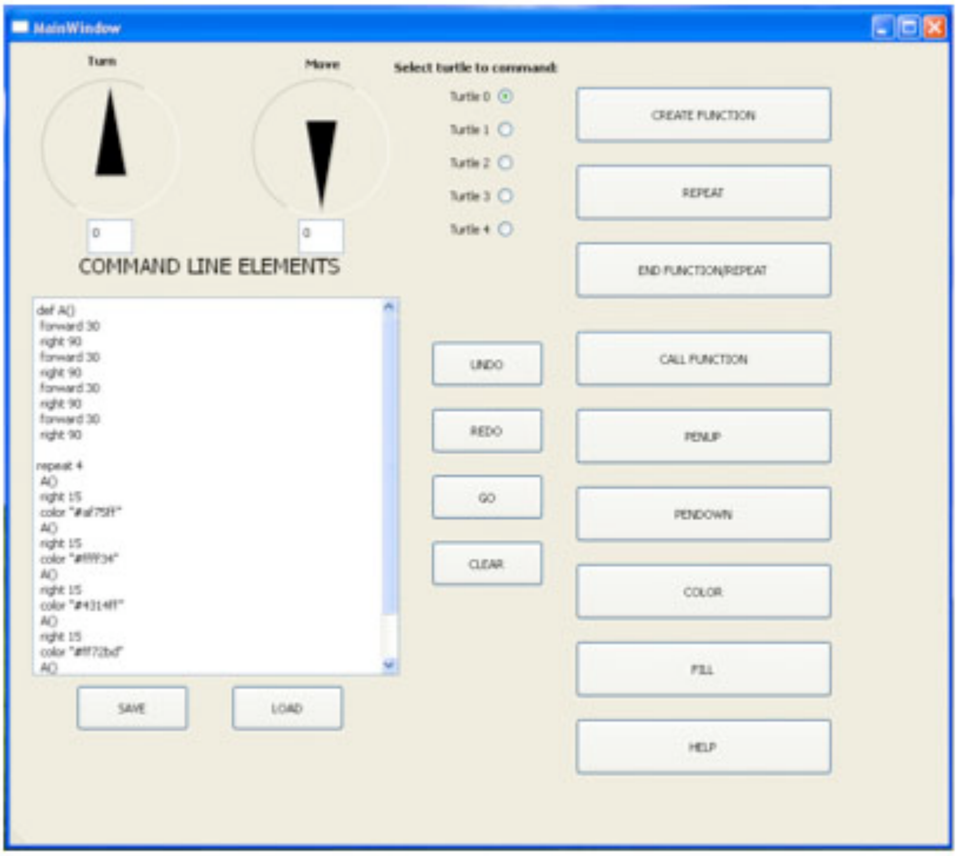


Figure 2. The graphical user interface for the Turtle on a Sphere system. Toward the left, the text window allows for the direct creation of a spherical turtle-walk program. The set of push-buttons at right can be used to insert textual elements into the program. The dials at the top are used to create "forward" and "right" commands directly, by hand. Other elements of the interface are described in the accompanying text. This figure shows a program in the process of being created; the current program is visible in the text window.

The Turtle on a Sphere system is a software-based "front end" to the Science on a Sphere display that allows students to create and observe turtle walks on the giant surface. We will discuss the spherical geometry behind these programs in the following section, but for now it is simply worth noting that the Logo "forward" command, on the sphere, causes the turtle to walk in the arc of a great circle, and that the default measure of the circumference of the sphere is 360. Thus, telling the turtle to move "forward 360" from any starting position and heading whatever on the sphere will cause the turtle to move all the way around the sphere and to finish in the same state that it started. Other turtle commands (right, penup, pendown, and color-changing commands) work in much the same way that they do on the plane.

Figure 2 shows the graphical user interface for the Turtle on a Sphere system, and also provides an overview of the basic functionality of the system. The major elements of the interface consist of a program text window (toward the left), in which turtle commands can be entered either via direct textual insertion, or interactively via selection of buttons and sliders. The set of buttons toward the right of the interface allow the user to input often-used language commands into the text window: the user can create a new function, start a "repeat" command, pick up or put down the turtle's pen, change the color, and so forth. The dials at the top of the screen allow the user to select a parameter for the turtle's forward and right commands, and to insert the appropriate command into the text window.

There are still other features of the Turtle on a Sphere interface worth noting here. The four smaller rectangular buttons toward the center of the screen allow the user to undo or redo an editing command, to run a program on the sphere (the "Go" button), or to reset the sphere to a blank state and clear the program window (the "Clear" button). The five radio buttons toward the upper center of the interface allow the user to associate a given program with any one of five distinct spherical turtles; thus, by selecting a given turtle and then pressing "Go", the same program may be run for distinct turtles in succession. Finally, the "Save" and "Load" buttons toward the bottom left allow the user to save and reload files in the text window.

In sum, then, the graphical interface contains the essential elements for a wide variety of programming projects using the SoS display. The Figure 2 interface runs on a tablet computer; but our current system also has a "pure" language interface through which more complex programs may be entered via keyboard.

Explorations in Spherical Turtle Geometry: a Sampler

The previous section introduced the basic interface for the Turtle on a Sphere system. In this section, we show how the system may be used to explore important ideas in non-Euclidean geometry. (A good mathematical introduction to these concepts may be found in Abelson and diSessa's book *Turtle Geometry* [1, Chapter 5].)

As an initial experiment in spherical turtle geometry, we can make use of several intersecting great circles to create the pattern shown in Figure 3.

Figure 3. A pattern created by three great circles.

The program that generates this pattern can be written directly:

```
forward 360     ; the turtle moves all the way around the globe,
right 90        ; turns right,
forward 360     ; moves all the way around the globe again,
forward 90      ; goes forward 1/4 around the globe,
right 90        ; turns right,
forward 360     ; and makes the third and final great circle
```

Several points are worth noting about the simple pattern of Figure 3. First, we note that the pattern divides the globe into eight identical spherical triangles, one of which is prominent toward the front of the display in the photograph. Each of these triangles is an equilateral triangle (each side is a quarter of a full great circle arc), and each triangle contains three right angles. In the plane, a triangle with three right angles is an impossibility; but on the sphere, the interior angle sum of a triangle depends on its enclosed area. Thus, a tiny equilateral triangle on the sphere has interior angles just a little above the planar value of 60 degrees, for a total just greater than 180; but the much larger triangles formed by the Figure 3 pattern each cover one-eighth of the globe, and consequently (see the discussions in [1, 4]) each of these triangles has interior angles that total 270 degrees.

The pattern in Figure 3 is in fact the projection onto the sphere of an inscribed octahedron (the second shape in the top row of Figure 4). To put it another way: suppose we were to place a large six-foot-diameter octahedron inside the giant sphere so that each of its six vertices just touch the interior of the sphere. A light bulb at the very center of the sphere would then project the octahedron's edges onto the surface in the same pattern shown in the photograph. This observation leads us to try creating patterns on the sphere resulting from the projections of still other inscribed shapes. Figure 4 shows the five so-called "Platonic", or regular solids, whose vertices are all surrounded by the same number of identical regular polygons; and Figure 5 shows the projections of the remaining four Platonic solids (the tetrahedron, cube, icosahedron, and dodecahedron) onto the sphere.

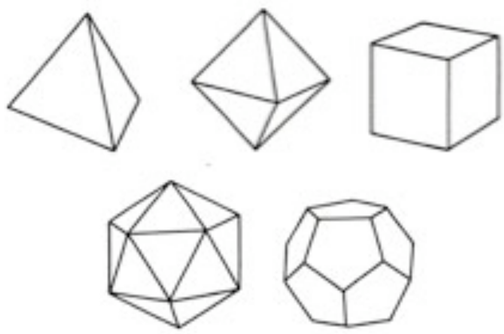


Figure 4. The five Platonic solids. Top row: tetrahedron (composed of four equilateral triangles), octahedron (eight equilateral triangles), cube (six squares). Bottom row: icosahedron (twenty equilateral triangles) and dodecahedron (twelve regular pentagons).

A little reflection on the patterns in Figure 5 will reveal still other interesting deviations from planar (Euclidean) geometry. For example, both the tetrahedral pattern at upper left and icosahedral pattern at lower left contain equilateral triangles (four and twenty, respectively). The large triangles of the tetrahedral projection each have three interior angles of 120 degrees each–that is, they each have an angle sum of 360 degrees (instead of the planar 180). The icosahedral triangles, which are much smaller in area, each have three interior angles of 72 degrees each, for an interior angle sum of 216; inspection of the figure shows that these smaller triangles look a good deal more "planar", or perhaps less "spherical", than those of the octahedron and tetrahedron. For the sake of completeness, it is also perhaps worth noting that the interior angles of the "spherical squares" in the cube are 120 degrees (instead of the planar 90); and those of the pentagons in the dodecahedron are also 120 degrees (instead of the planar 108).

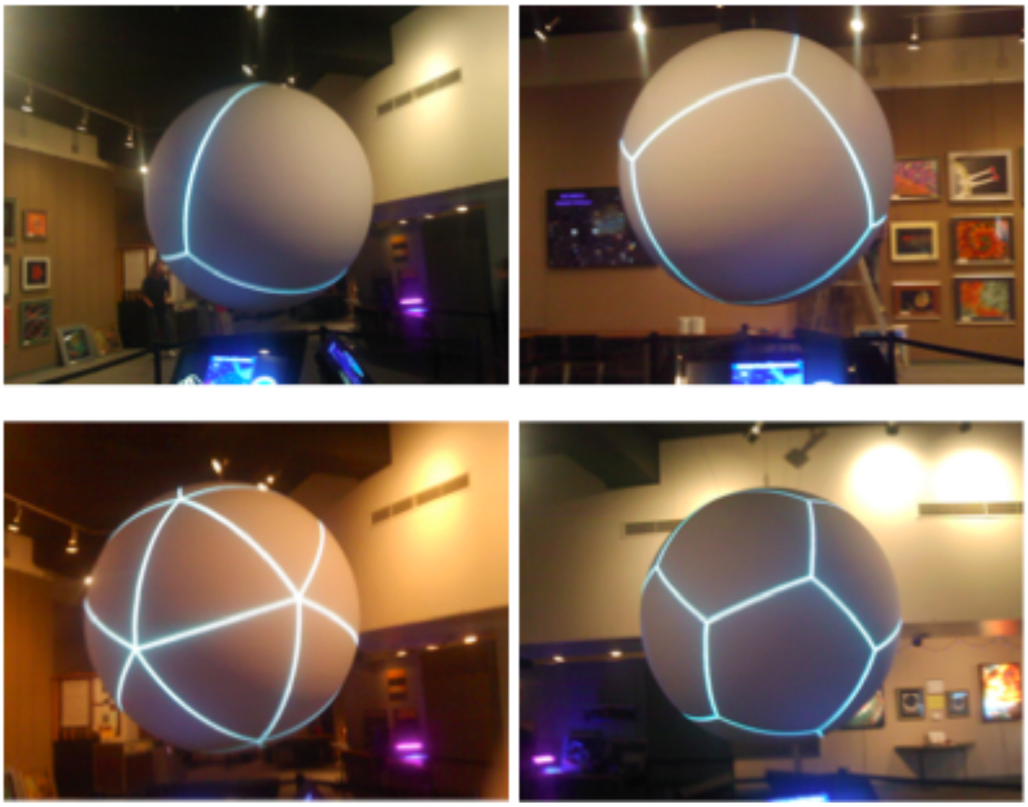


Figure 5. Projections, onto the sphere, of inscribed Platonic solids. Top row: the tetrahedron and cube. Bottom row: the icosahedron and dodecahedron. (The projection of the octahedron is shown in Figure 3 earlier.)

The examples of Figures 3 and 5 are intended to highlight the distinctions between planar and spherical geometry; but it is fun to try other examples of "standard" Logo programs on the sphere to see what happens. The classic turtle-drawn "flower" of Figure 1, consisting of twelve "petal-like" shapes, was drawn in outline by the following program:

```
to petal (size, moves, smallturn, bigturn)
  repeat 2
    repeat moves
      forward size
      right smallturn
    right bigturn

repeat 12
  petal (5, 10, 6, 116)
  right 30
```

A reader familiar with planar Logo patterns will note that the angles used for the repeated "petal" shapes would not result in closed turtle-walks on the plane; but they do on the sphere. The Figure 1 flower also, parenthetically, illustrates the ability of the Turtle on a Sphere program to include "fill" commands (for brevity, the code for this portion of the flower program is not shown).

Figure 6 shows two other famous turtle-drawn programs, realized in their spherical versions. At left, the "dragon" curve (as described in [1], p. 93) is shown; on the sphere, the standard turtle-turn for the dragon curve has been changed from its usual value of 90 degrees to a "softer" value of 86.5, to produce a more aesthetically appealing version of the curve. At right in Figure 6, the "C-curve" (see [1], p. 92) has been drawn. The program for drawing this shape is as follows:

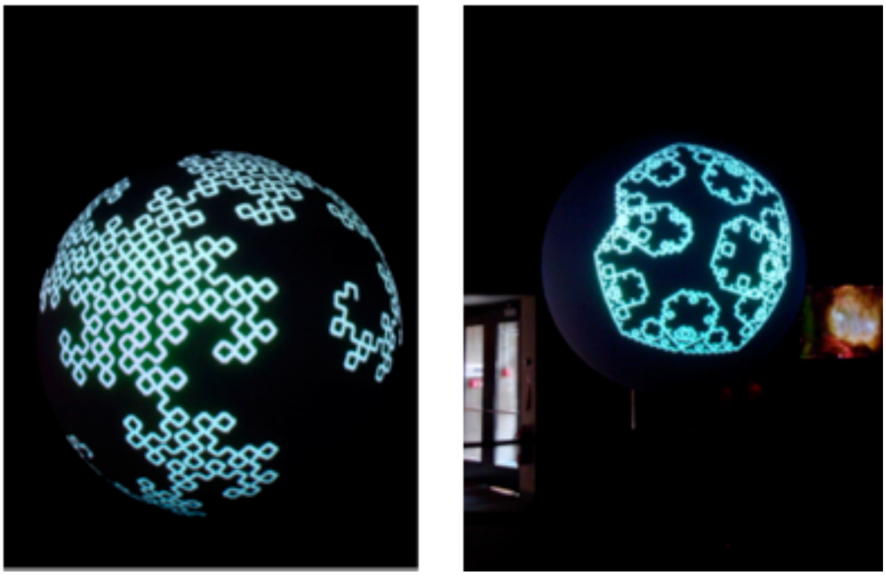


Figure 6. Two recursive patterns drawn on the sphere. At left, a dragon curve; at right, a C-curve.

```
to ccurve (side, level)
   if level == 0:
      forward size
   else:
      ccurve (side, level - 1)
      right 90
      ccurve (side, level-1)
      left 90

ccurve (2, 10) ; create a 10th-level c-curve with a side of 2
```

In the plane, the C-curve would not "close in" on itself, but instead looks like a rather elaborate "C" shape; on the sphere, as seen toward the right of Figure 6, this large C-curve makes a (highly attractive) closed shape.

Projection and Display Technology, Unorthodox Turtle Geometries, and Programming as Performance: New Directions for Constructionism

The "Turtle on a Sphere" system is still in a relatively early stage of development, and has only been tested very informally with local students. Our goal is to use this as a springboard for future work, culminating in a system that can be appropriated and extended by interested parties–

researchers, teachers, hobbyists, and students. Before its public release, however, any system along these lines would need additional work: extensions to the language (such as analogues to the Logo "setpos" and "getpos" commands, among others) and refinements of the current interface.

The larger subject of this paper, though, goes beyond one particular software system. Turtle on a Sphere is a single instance of what could well be a much more pervasive technological idea–namely, the use of customized surfaces and projectors to produce "screens" of many shapes, sizes, and environmental settings. While the implementation of NOAA's Science on a Sphere display is highly sophisticated, the basic idea–using multiple projectors to create a unified picture–is potentially applicable to a variety of other surfaces. One might imagine using similar techniques to create graphics on (e.g.) a cylindrical surface; a conical surface; a pseudosphere model (to explore ideas in hyperbolic geometry; cf. [10] for a general discussion of computational approaches to this subject); or even the interior (as opposed to exterior) of a hemisphere. Each new surface for display does, admittedly, require significant effort: an underlying software translator must be developed to control the output of multiple finely-situated projectors. Nonetheless, once such a driver has been implemented for a particular surface (say, a cylinder), it can be ported to a wide number of sites, just as the Science on a Sphere display has now been installed in numerous public settings.

We would argue, further, that there is a strong potential for accessible home or classroom implementation of such displays in the future. A "do-it-yourself" Science on the Sphere could plausibly be implemented using a smaller sphere (of perhaps 1 foot in diameter and composed, say, of white acrylic) and a collection of four affordable nanoprojectors of the type currently emerging in the commercial market. Going just a bit further, it should be possible to employ affordable 3D printers to create a variety of customized geometric surfaces (cones, cylinders, etc.) in white plastic; and these surfaces could be the basis of a wide range of classroom or home displays. In other words: the hardware components behind the Science on the Sphere display could be re-implemented in smaller form, with cheaper materials; while the sophisticated software driving the display could be downloaded to homes and classrooms, just as it is now in museum settings. The result, within a decade or so, could well be a pervasive presence of beautiful new display surfaces, displaying homemade or student-created graphical effects tailored for particular geometries.

Such systems are themselves part of a still larger, growing ecology of novel display technologies–including numerous competing technologies for three-dimensional and volumetric displays. [3, 5] In effect, by considering the roles of these emerging technologies for constructionist education, we can re-imagine children's programs as *performances* in the making–as things to run in numerous public settings and in specialized or customized environments. Programs could run on surfaces arranged in playgrounds, theme parks, classrooms, or private homes; graphical effects could combine elements of projected displays and three-dimensional screens; small personalized projectors could be used to superpose graphics on top of still other displays (for example, children might use their own personalized projectors to produce handheld animated effects combined with those produced by the SoS display).

Constructionism, as an educational philosophy, involves a close attention to the intellectual role of design and personalized creation; but at the same time, as we noted at the outset, children's artifacts attain particular educational potency when they are embedded within social practices–for instance, when they can be presented in public, shared with an audience, and linked to meaningful physical settings and environments. Novel display technologies–of which Science on a Sphere is merely an early example–offer an avenue through which to rethink these aspects of constructionist education. Such displays offer cognitive and intellectual challenges–"powerful ideas" (in Papert's phrase) of three-dimensional and non-Euclidean geometry and visualization. At the same time–and equally important–they offer beautiful and motivating opportunities for children to realize, share, and express their ideas.

Acknowledgments

This research was made possible through a grant from the University of Colorado Outreach Office, and is funded in part by NSF grant IIS0856003. Thanks are due to Francisco (Tito) Salas, Jim Love, and Mike Levy of the Fiske Planetarium in Boulder, Colorado, whose help and collaboration made it possible to develop the spherical-turtle language and interface. This project began as a result of wonderful conversations with Uri Wilensky; thanks also to Ann Eisenberg, Gerhard Fischer, Clayton Lewis, and Antranig Basman for their contributions and discussions.

References

[1] Abelson, H. and diSessa, A. [1980] *Turtle Geometry.* Cambridge, MA: MIT Press.

[2] Craft Technology Lab website: www.cs.colorado.edu/~ctg

[3] Dodgson, N. [2005] Autostereoscopic 3D displays. *IEEE Computer*, 38:8, pp. 31-36.

[4] Eisenberg, M. [2010] Computational Diversions: Turtle Really and Truly Escapes the Plane. *International Journal of Computers for Mathematical Learning*, 15:73-79.

[5] Favalora, G. [2005] Volumetric 3D Displays and Application Infrastructure. In *IEEE Computer*, 38:8, pp. 37-44.

[6] NASA website (list of "Science on a Sphere" locations): www.nasa.gov/centers/goddard/multimedia/frozen/soas_locations.html

[7] NOAA website for "Science on a Sphere": http://sos.noaa.gov/index.html

[8] Noss, R. and Hoyles, C. [2006] Exploring Mathematics through Construction and Collaboration. In Sawyer, K. (ed.) *Cambridge Handbook of the Learning Sciences*. New York: Cambridge University Press, pp. 389-405.

[9] Papert, S. [1991] Situating Constructionism. In Harel, I. and Papert, S. (eds.) *Constructionism*. Norwood, NJ: Ablex.

[10] Stevenson, I. [2000] Modelling Hyperbolic Space: Designing a Computational Context for Learning Non-Euclidean Geometry. *International Journal of Computers for Mathematical Learning*, 5:2, 143-167.

Co-operative learning of Logo in the classroom – development of project "Recycling"

Tatjana Balvočienė, *t.balvociene@gmail.com*
The teacher of IT, expert, Silute Vydunas gymnasium, Lithuania

Vaida Paukštė, *vaida.paukste@gmail.com*
The senior teacher of IT, Vilnius Abraomas Kulvietis secondary school, Lithuania

Abstract

Co-operative learning (also known as learning in small groups) – strategy that helps students to achieve better learning results, to develop social and communicative skills, put a positive effect on students' opinions and values associated with the school and other students, especially with training and learning.

Co-operative learning can be successfully applied in the classroom of Logo when getting acquainted with that system and preparing more complicated projects. The article presents and discusses example of how the Logo can be used in the classroom for co-operative learning. When doing a bigger project students from the beginning of the project distribute the work and responsibilities: gather information and share it, draw or collect the necessary project drawings, describe the events and procedures and, finally, everything combine into a single job. In each stage different student's abilities are revealed (since the work and students are very different).

In this paper the project-game "Recycling" (Figure 1) is described. It can be prepared by the 2-3 students group who are of the age 12-13. With the help of the teacher students plan their work, provide each member with a specific activity and at the end of the project evaluate the process and achieved results.

Figure 1. Project-game "Recycling". The model shows how the waste should be dragged by the mouse into a suitable container.

Key words

Co-operative learning; project; Logo; *Imagine Logo*.

The situation in Lithuanian schools

Co-operative learning

Co-operative learning is also called learning in small groups when students in the class are divided into small groups of two to four members. The main aim of learning in small groups – to achieve that each student when communicating with others, actively performing given tasks, would learn better. Co-operative learning is not one of the teaching methods, but rather an approach to the same training, based on social interaction and fostering cooperation among small groups of learners (Sahlberg, 2004).

A few years ago (2002-2006 years) Lithuania Ministry of Education was carrying out Lithuanian School Improvement Programme for the Millennium[1] of Lithuania, one part of which was a long-term teacher training (learning) based on co-operative learning.

IT learning usually is an individual student's work, when deepening your knowledge into the possibilities of computer programs. Nevertheless, it is perfectly adaptable in the classroom, where Logo system is used for co-operative learning.

Student's IT skills for working with the Logo system

One of the parts of IT course in class 5-6 is defined as the Curriculum framework for primary and basic (lower secondary) education approved by the Minister of Education (2008)[2] – "Constructing by a computer". This course of study is particularly suitable and recommended for the Logo system. In class 5-8 up to 108 lessons can be devoted to IT course.

Integrated IT program is added to the field of IT learning. This program is implemented through other subjects when using IT for learning purposes. In class 7-8 it is especially recommended to integrate IT learning into other subjects. In such lessons students can successfully apply existing IT and work skills with the Logo system, particularly in the development of more sophisticated projects.

The Logo system can be successfully integrated into other disciplines or being taught during after school activities in upper grades. There are several versions of Logo system translated into the Lithuanian language: *LogoWriter, Comenius Logo, Imagine Logo*. All the examples described in this article were made with *Imagine Logo*, but these ideas can be perfectly realized with different version of Logo system.

In this paper we describe how work is organized with the students in class 6 at the end of the school year. At that time, students have already acquired many skills, as defined in the Framework Programme – learn how to work with folders and files (to copy, move, rename, etc.) to prepare and form a simple text, work with graphics editor, use the web services (find a text and visual information, add it to the wanted folder) to work with *LogoMotion* graphics and animation editor, *Imagine Logo* system to create new objects (buttons, Turtles), change their options, describe the procedures and simple events items, to create movement with the help of the commands.

[1] More on this project can be found at http://www.mtp.smm.lt/

[2] The programs define the state-level curriculum. Schools and teachers, in accordance with the Framework programs, develop school and classroom-level (according to their abilities) to achieve the best possible results (Curriculum framework for primary and ..., 2008)

Project-game "Recycling"

The idea of the project

Students with better skills on Logo system, using co-operative knowledge may develop advance projects, such as the creation of their favourite animated illustrations of literary works or to develop an interesting game.

An example of the project-game on ecological theme "Recycling". On the "scattered" display it is proposed to sort the waste – take every item which is given with a mouse and drag it into a suitable container. Object is emitted only if the container is chosen properly, otherwise it remains visible on the screen in place, which has been towed. Both containers and waste are represented by the Turtle with masks.

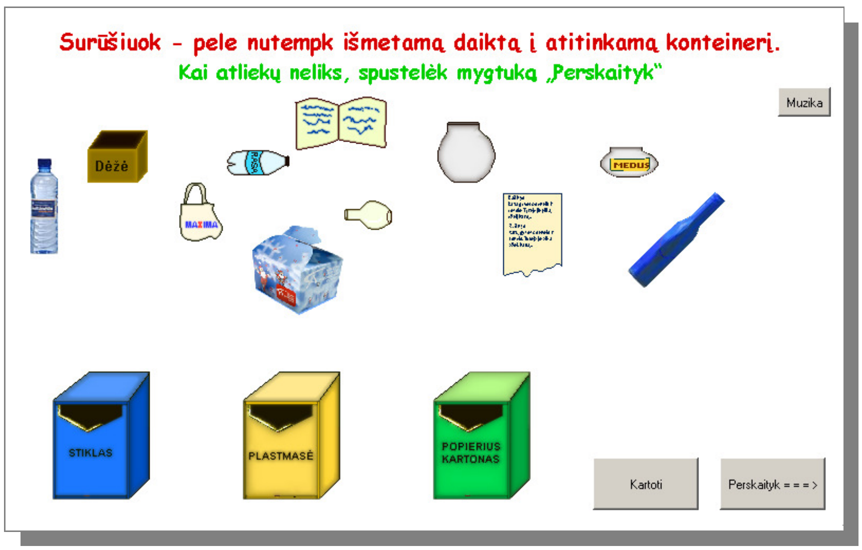


Figure 2. An example of the project-game "Recycling". The model shows how the waste should be dragged by the mouse into a suitable container

Such drafting may be integrated with the natural sciences and / or biology lessons. Before the start of the project preparation, IT teacher should consult the colleagues who teach children science or biology lessons that the importance of recycling would be discussed. Thus, together with teachers, students would learn how various waste is sorted, why is it necessary to do, what damage can be done to nature and the environment, without sorting waste. Students also can read more information about recycling on the specified websites. Lithuanian students used the information from the site "Do not be indifferent – recycle!" (http://atliekos.am.lt). The website provides explanations and tips. Analogical Logo project created by other students or teacher could motivate students to inquire about waste recycling and to prepare the project on that theme.

Expected Results

Expected student achievement:

- Working in teams, students will be able to plan, prepare and present an integrated science project, demonstrating how to sort the emission of paper, plastic, glass items.
- Students will be able to describe the object for Logo items – for a sheet, Turtle or a button.

This project helps to develop not only skills of working with information technology, but also the social skills of pupils are taught (to assume responsibility for the overall result, to help for a class friend, to agree on a collective working).

The project is not complicated, but for one student it would take a longer time to prepare it. If students work in small groups, the final results can be achieved much faster.

Work steps

This article describes how students' work is organized in pairs, when 3 lessons are devoted for the preparation of the project. The teacher should advice students to split into small groups, according to their wishes and abilities. If the number of students is odd in the classroom, one student with a teacher may consult their classmates on the subject (when giving the task for the student, the teacher should take into account the level of achievement and communication skills of the student) or in one group there will be only three students.

The first lesson

- Before starting the work on the Logo system, the teacher explains the general principles of the project development and progress:
- necessary masks are prepared (recycling objects, containers);
- Turtles with the masks chosen by students (containers and recycling items) are laid out on the sheet;
- to describe events Turtles are used. Turtles will help to drag the "waste" in to appropriate containers.

Start procedure is described and **repeat** button is created.

It is necessary to emphasize the relevance of the Turtle-container names (for example, "cardboard", "glass", "paper") – they are used to describe an event **onLeftUp** hidden under Turtle mask which symbolizes recycling item. Example, if the event describes the commands **if overlap? [cardboard] [hideMe] [ ]** the waste will be thrown out (the Turtle will hide) when it is dragged by a mouse to the suitable container. That is why it is necessary to give a name for the Turtle symbolizing that container.

To make the work faster in the beginning a teacher could initiate all class-joint activities: each student could paint one element of sorting the waste or one of the containers for recycling. For this they can use the painting program or the Internet, then arrange the items (for example remove the background), to save as the Logo masks – several of the masks are shown in Figure 1-2. Students who work faster can prepare and adapt more pictures. Prepared masks are copied into a file which is accessible from any class computer and each group of students from this shared folder chooses images which they will later use in their work.

At the end of the lesson further work steps and the activities of the next lesson are discussed:

- student group (pairs) receives a package of measures – cards with a reminder (explanation) of steps of project;
- a teacher explains that at the beginning of the second lesson students will use these cards (if it is necessary use the teacher's assistance), will perform individual tasks, and then work together in pairs.

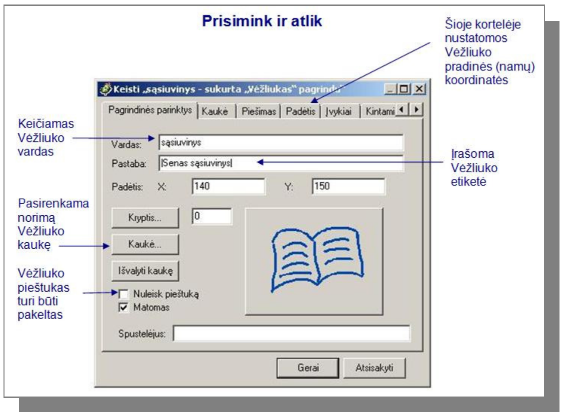


Figure 3. The card explaining student's work – to recall how and where Turtle's name must be changed, Turtle label is added, the initial (home) position is set, etc.

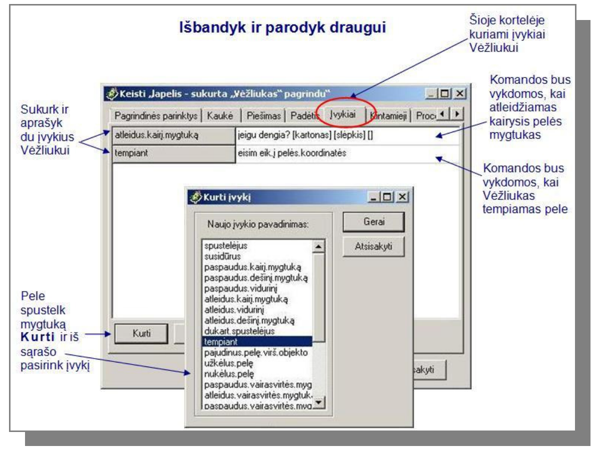


Figure 4. The card explaining students' work – how the Turtle's events are developed and described for the project.

Second lesson

Individual work, which takes about half of the second lesson, is organized as follows:

- one student plans the Turtles (symbolizing recycling items and containers) layout on the screen (calculates coordinates which show the position of the Turtle mask on the Logo sheet); remembers how to change Turtle options – name, home position, etc.;
- the other student examines the events described in Turtle, enabling to realize dragging of the "waste" (Turtle with a mask) into the right "container" (which also is represented by the Turtle mask); these students get Logo design template, which has at least two Turtle containers with appropriate names, such as glass and plastic;
- if there is a small group, which has three pupils, a third member chooses information about sorting from the Web sites, predicts what links will be created into information sources from *Imagine Logo* sheet, remembers how to do it;
- teacher advises students who need help.

Later on, students working together use the produced information: change the options of every Turtle – give it a name (according to the mask it represents), indicate to raise the pen up, describes the events. To make the work faster the text can be copied from one ID card to the other making necessary changes.

After finishing those works not complicated procedure **start** is added to the project. This is needed to return Turtles into their "home" ("home" coordinates should be indicated in every ID card of Turtle options):

```
to start
   tell all
   penUp
   home showMe
end
```

Repeat button is created. In the ID card of this button the procedure **start** should be performed. When the button is pressed all the images of thrown away goods are returned to their "home" and can be recycled again.

Third lesson

Third lesson is dedicated to presentation of the projects and discussions:

- every group of students is given questionnaires to discuss the work process (*Where we succeeded the most? What we failed this time? Why? What should we do differently in the future?*), achieved results (*If we succeeded in developing the project? What can we improve or add in it?*);
- the time is given to reflect and discuss issues (about 10 min.);
- each group should present their work and the results of their discussion;
- at the end of the lesson the teacher suggests to choose the most original, smartest, most realistic, etc. project. Even the smallest input of each student should be noticed because it raises learning motivation. That's why teacher should prepare various nominations in advance for all the groups.

The opportunities of project development

A similar project development and organization could be performed differently: it depends on the class achievement level, students' abilities to work together, the number of lessons and the experience of the teacher:

- if students work in larger (3-4) groups and / or more lessons can be given for project preparation, the project could be bigger–more sheets could be created, animation, sound effects could be used. Figure 5 shows the eco-design, which contains information about the importance of recycling. There can also be added animated elements such as hedgehogs, caring bottles of mineral water to a waste container;
- more participants, students teachers, parents could be invited in to project presentation.

An example given in the article is described in the textbook of IT for classes 5-6 (Balvočienė, Kriščiūnienė, 2008), but these or similar projects are appropriate in the work with the higher classes. Then more complex Logo items could be adapted in the project and projects may be larger.

Can we define learning in small groups as co-operative learning?

According to Bennett and others (2000), the key elements that separate the co-operative groups from the normal ones are: **a positive interdependence, individual responsibility, close interaction, communication skills assessment**.

The elements of co-operative learning in this project:

- **small positive interdependence of group members is formed** to achieve the same team goal (the aim is to finish the overall project), to use similar measures (every student gets a part of material for individual preparation), to be of the same importance for the group (when returning to a group work everybody presents their work done);
- every student in each group **is responsible** for the analysis of a piece of material and presentation of it to other group members (to other group members if it is not a pair work);
- discussion and presentation of the work prepared assures one more element of co-operative learning – **evaluation and assessment**. It is very important to highlight not

only a created product but also the process of work, so pupils are educated and students' co-operative skills are developed, learning motivation is being raised.

Of course, there is a possible range of situations how to prepare the project. However, in most cases remains the possibility of co-operative learning elements developing not only academic but also social skills of students.

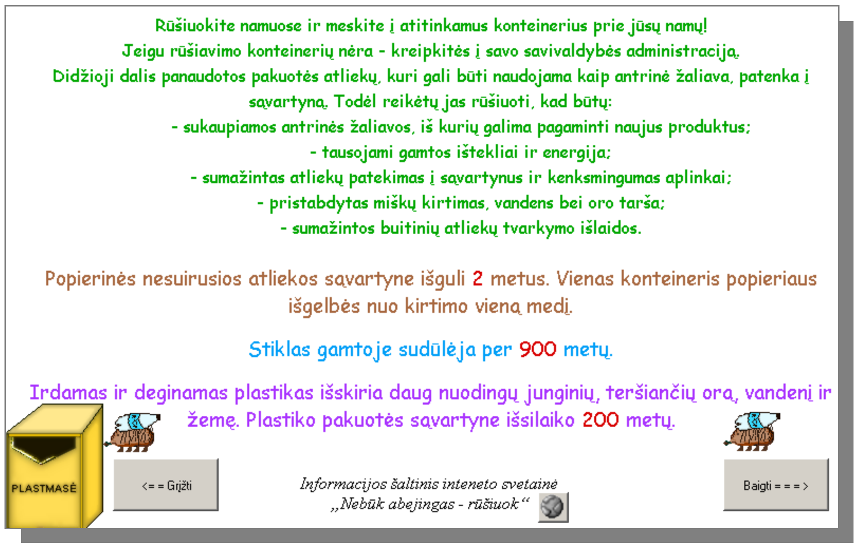


Figure 5. Project-game "Recycling", the second page – a detailed information about the importance of waste sorting, the influence of not sorted waste on the environment, ecology is presented.

References

Balvočienė T., Kriščiūnienė N. (2008) *Projektų mozaika*. Informacinių technologijų vadovėlis V-VI klasei. Kaunas: Šviesa.

Barrie Bennett, Carol Rolheiser-Bennett, Laurie Stevahn, (1991). *Cooperative learning: Where Heart Meets Mind. Educational Connections*, Toronto, Ontario.

Curriculum framework for primary and basic (lower secondary) education (approved by the minister of education and science of the republic of Lithuania, 26 august 2008, resolution no. ISAK–2433)

Sahlberg P. (2004) *Mokymosi bendradarbiaujant principai.* / Sėkmingo mokymosi link. Serija „Į pagalbą mokytojams". Vilnius: Leidykla „Sapnų sala", pp. 53-62.

Roamer Too: a new educational robot as an emotional pedagogical companion

Secundino Correia, *secundino@cnotinfor.pt*
Cnotinfor, Innovation workgroup, Coimbra, Portugal

Juliana Costa, *juliana.costa@cnotinfor.pt*
Cnotinfor, Innovation workgroup, Coimbra, Portugal

Marco Estanqueiro, *marco@cnotinfor.pt*
Cnotinfor, Innovation workgroup, Coimbra, Portugal

Inês Castro, *inês.castro@cnotinfor.pt*
Cnotinfor, Innovation workgroup, Coimbra, Portugal

Ana Baptista, *ana.baptista@cnotinfor.pt*
Cnotinfor, Innovation workgroup, Coimbra, Portugal

Short presentation

In times of constant integration of technologies in every day life, teachers have to be aware of new trends to motivate students, keep them interested and achieve significant learning. This implies to choose between different technologies the ones that better fits a particular domain of knowledge or age group.

One of those approaches is robotics. They have been increasingly introduced into the classroom as a new learning strategy. They provide an interesting and engaging way to address traditional subjects in an innovative manner. There have been a number of robots used as a learning tool and the results are quite impressive, students get better at problem solving insight, they are more motivated to work in groups and they start to better understand abstract concepts.

This poster presents the research under progress on LIREC project with a new educational robot, Roamer Too, and the migration of a pedagogical virtual companion into Roamer Too. LIREC (LIving with Robots and intEractive Companions - supported by the EU FP7-ICT-2007 project - LIREC: 105554) is focused on how humans can establish long-term relationships with robots and virtual companions and the implications this may have on its design, construction and usability. Within the project, a virtual companion was conceived, Little Mozart, whose goal is to establish a meaningful interaction with children on how to compose and improve their knowledge of melodic composition and basics of musical language.

One of the goals is to investigate if Roamer Too can become an extension of our virtual companion and establish a long-term relationship with children, and even how we could improve Roamer Too introducing behaviors with affective feedback. Migrating Little Mozart to Roamer Too sets a few challenges; the most obvious are the embodiment itself and the emotional behavior.

Joining the great learning experience that Roamer Too already provides with the possibility to engage the child in a more meaningful relationship, we expect to end up with a more complete educational robot.

Keywords

Roamer too; learning; interaction; emotion; companion; migration

Setting Powerful Ideas to Music

Sterling Beckwith, *beckwith@yorku.ca*
Music & Humanities, York University, Toronto

Abstract

Soon after first connecting with the Logo Project at MIT 40-odd years ago, the author joined the ranks of those who sought to expand its original pedagogical vision to embrace the arts. The ideals of Constructionism's founders have continued to inspire his efforts to stimulate musical awareness and support creative imagination by helping children to design their own simple musical compositions. In the course of this work, much was learned about music, about young minds, about computing, and about constructive education.

Initially, his student research team at York built an extensive library of special Logo-based software routines, designed to drive a series of hands-on exercises and projects in computer-assisted musical construction. Successful trials of York's first portable "musical computer" in Ontario schools revealed some basic requirements for an educational environment conducive to creativity. Eventually a commercial software package embodying key ideas from the York project was commissioned for the earliest personal computers.

Seymour Papert's insistence that computers can link abstract thinking with concrete know-how was a major influence on this work. However, the nature of children's mental processes while composing remains as much a mystery as it was before computerized music production became widespread. And a recent survey of available software oriented toward composing for beginners reveals disappointingly little attention to its suitability for young users. Some promising exceptions, as well as recent proposals for a radical reorientation of programming itself, could help to awaken new interest in the potential of digital media to stimulate musical thinking and facilitate its expression.

While the Constructionist vision of computer-mediated, self-directed learning has inspired successful efforts to energize and enliven the teaching of science and mathematics, a preoccupation with the power and glamour of new media resources has sometimes prevented students from developing the skills and acquiring the life experience they need to undertake serious creative work in the arts.

In reviewing some of the chief lessons gleaned from his earlier work, the author, following Papert's injunctions, hopes to contribute to a continuing dialogue about the role of the arts in education, the proper and improper uses of disembodied media, and the various means by which we appropriate and invent new knowledge.

NOTE: This paper will be accompanied by a series of visual projections, amplifying or illustratiing key points.

Keywords

musical composition; constructionism; music pedagogy; educational software;
creativity in children; Logo; artificial musical intelligence; programming languages.

Setting Powerful Ideas to Music

Sterling Beckwith, *beckwith@yorku.ca*
Music & Humanities, York University, Toronto

A. Joining the Movement

1) Border crossings

All of us have been touched, in one way or another, by the same particular current of ideas about computation, constructive learning, and children's minds that was notably espoused by Seymour Papert and others at MIT in the 1970s and 80s. Papert's fascination with how children learn to make sense of the world—how their encounters with physical objects, complex relationships, and external forces can stimulate their intellectual and personal growth—has by now become our own.

Educational dogmas and fashions have come and gone in the years since The Children's Machine and Mindstorms first appeared. Yet people of all backgrounds and specialties, in many countries, still find in these books both an inspiration and a challenge. The 'children's machine' par excellence is, of course, not the computer but each child's own embodied brain. Papert insisted that children need the serious fun of discovering how their world works, by playing with it and constructing complex ideas about it. Has not this same capacity for playful discovery given rise to some of humanity's proudest achievements, in such fields as mathematics, science, or the arts? Yet somehow we seem to have forgotten how to make it flourish for ordinary kids in ordinary classrooms. Perhaps that is why, some forty-odd years later, Papert's powerful ideas can still challenge teachers at all levels and in all fields, mine included, urging us to use new media only in ways that will respect, and join forces with, every child's native genius for learning.

Some years before Papert's work became widely known, I found my way to the AI Lab at MIT, and was encouraged, by Marvin Minsky and others, to join the handful of pioneering experimenters who frequented the room in which its Logo Project was housed. To those of us who lacked a background in electronics or engineering, Logo seemed the perfect entry-point into an exciting new world of futuristic educational machinery — simple enough for our kids (and even us) to master, yet with all the allure and limitless promise of a newly emergent technology. We saw how quickly children were captivated by the activity of writing a simple computer program, then watching as a toy robot carried out their instructions. New pathways to personal knowledge seemed to open, when children could "teach the machine" how to achieve their desired goal step by step, and then actually see, feel, or hear whether it succeeded—or at least figure out why it failed, and correct their program accordingly. And for at least a few of those kids, using Logo to draw geometric shapes, drive toy robots, or generate simple tunes did seem to stimulate and engage them more than school ever had before.

The invitation to speak at CONSTRUCTIONISM 2010 could not have reached me at a better time. I had already begun to look back on my years "in the movement" and the questions we grappled with along the way, some of which still perplex me a generation later. Hence the hopeful title of my presentation, which points to an ongoing process, but promises no pat answers or foolproof solutions. In it, I will try to explain why extending the impact of Logo to include certain kinds of music learning seemed at first not only plausible, but a promising good fit. Eventually, my Logo experience worked just as Papert predicted—that is, it called into question most of my assumptions about composing, computing, self-instruction, creativity, and even about the nature of musical knowledge itself. The rest of my paper focuses on a few of the issues thus raised, and hints at some possible directions for future development.

Music is one subject on which everyone is entitled to their opinions, so I hope you will feel free to complain, disagree, or argue with anything I say, and share questions of your own, whenever and wherever you can find me. After all, isn't that why conferences like this one are still worth attending in person?

2) A voice for the Turtle

As examples of "concrete thinking", the early exercises in turtle geometry still seemed pretty limited and abstract to me. Though one heard talk of encouraging children to "think like a real mathematician", Logo's emergent "Mathland" was evidently far more modest in scope. Geometry had after all been for centuries a preferred zone for the exercise for young minds; so not surprisingly, it was there, rather than in the loftier realms of contemporary higher mathematics, that the Logo turtle found its pedagogical niche. But compared with what usually goes on in a typical math classroom, the Logo style of solving geometry problems by trial-and-error programming must have seemed far more active, more concrete and more engaging.

If any comparable "safe zone" for children's musical problem-solving existed then, one would have been hard put to locate its limits. Under the aegis of the Logo Project, both Jeanne Bamberger and I set out to remedy that lack. (Many of you are no doubt familiar with the remarkable contributions she has since made to constructionist pedagogy as a result.) Though we began with only a meager handful of sound-defining primitives, we did have available a barebones device called the Logo Music Box, of which a couple of samples had been cobbled together by Papert's brother. Its limited repertoire of programmable beeps, pops, and gritches could only be brought into some semblance of audible order by executing programmed instructions written in Logo code.

Looking back, I can see how lucky we were that our robotic music-making capabilities were so limited. Our very low-fi version of "the children's machine" really did sound more like a turtle in heat than anything else, and was devoid of any obvious sonic appeal. Logo's very abstractness, on the other hand, was for some music educators its great attraction as a potential new medium for learning. In music classrooms, the emphasis is often so overwhelmingly on the live activity of producing and combining sounds that there is little or no time to reflect on the structure or design of what is being played. (Much of that will anyway have been pre-composed by various dead white guys—but more on that theme later.)

A further advantage, it turned out, could be gained from using Logo code as a means for expressing musical intentions. To get beyond the note-by-note processing that common music notation seems to demand, and learn to think in the musical equivalent of complete words, clauses, or sentences, can take years, and sometimes never happens, so strong is the unconscious cognitive prejudice imposed by the standard notational graphics in use today. What a relief it was to discover that the power and simplicity of recursive definition in Logo encouraged its users to work mainly with whole musical gestures and even larger spans, instead of those isolated atoms called "notes"!

3) Pioneers on the Logo frontier

I came back to my Canadian university post from MIT in the early Seventies, eager to begin developing computer-assisted methods of musical exploration for Ontario schoolchildren. As a teacher, performer, and former child improviser myself, I had long had a particular interest in helping children create music of their own. I was naturally eager to see whether Logo, though offering only limited access to the twin powers of programmed control and automatic sound generation, could nevertheless help reduce some of the technical obstacles that discourage children's creative play with melodies and rhythms. Would novices start to think a little more "like a composer", I wondered—or even, more like a programmer— if they were able to use the computer as a kind of musical Lego set, first imagining a desired musical gesture or phrase, then instructing the machine how to realize it step by step, and finally hearing the result played back, if only sketchily, by the digital Music Box?

With the help of a generous government grant, I set up the first Logo development lab in Ontario, staffed with York undergraduates, only one or two of whom were majoring in Computer Science. The sonic output of our robot MusicBox was still painfully crude, and Logo a very clumsy and blunt instrument for musical manipulation, which we could access only via long-distance phone connection from a research center in a distant city. Yet at the time, what we were attempting seemed like a real breakthrough.

Before offering it to kids and their teachers as a music-making vehicle, we greatly extended the Logo language by building a full range of musical pseudo-primitives, as well as new ways of handling and combining list-structures to represent extended sections or whole pieces of music. Then we grappled with how to make this new vocabulary readily accessible to the novice user, trying various shorthand schemes to organize the available choices into a coherent and easily remembered alphabet.

From the outset, we recognized that "real" composing in Logo was way beyond what even the most audacious partisan of artificial intelligence would have dared to attempt. But Logo did give us something almost as valuable: an arena for playing WITH music, enabling kids to build their tunes and other patterns out of nameable, repeatable, transformable, recursively definable entities—motives. phrases, whatever— musical patterns that are less "abstract" but more easily recognized and remembered than mere "notes".

Eventually we ran Canada's first in-school trials of Logo Music, with Ministry of Education support. To be sure, not much extended composing was attempted in those early trials. (Exercises in de-composition predominated.) Both we and the children were feeling our way forward, in what for the cooperating teachers was still uncharted territory.

4) Discovering connections

If Logo was ever to realize its promise as a new approach to music and other arts, or to non-mathematical learning of any kind, let alone as a vehicle for educational reform, an explicit rationale would be needed, but was still lacking in these early days. Those of us with musical interests thus had an extra incentive to work at explaining what it could and couldn't do for us.

To be sure, some kind of involvement in Music-making is known to be encouraged and prized by all human communities (even including MIT professors!). Perhaps this is because making and sharing music can so easily engage both mind and body, by actions the human organism seems expressly designed to perform. Even the littlest children spontaneously use musical sound, speech, rhythm, and gesture in every possible combination to communicate with and respond to others. (One wonders whether Papert ever considered how SINGING develops a child's ability to express ideas with and through the body—without the help of robots or programmable gadgets of any kind.) Adults may also turn to Music to help them achieve emotional expression, cultural rapport, meditative awareness, or social integration.

Besides, Music has a long history of reciprocal involvement with whatever society's latest high-tech advances happened to be. Indeed, few human activities, or even other fine or performing arts, are so strongly linked to multiple cooperating technologies of symbolic communication and physical production. As Jaron Lanier reminds us: "In most historical eras, and in most cultures, we have put as high a priority on creating objects that make new sounds as we have on finding ways to kill one another."

If all this is true, shouldn't the study of music provide ample opportunity for the same kinds of responsive, engaging hands-on transaction Papert sought to introduce into the Math classroom? I imagine some of you are eager to answer: "Of course! But Isn't knowing Music just another mode of knowing Mathematics? After all, we teachers can use musical sounds to illustrate audibly some simple mathematical relationships, especially for those children who are uncomfortable with numbers." Yes, the spirit of Pythagoras does live on, even here at CONSTRUCTIONISM 2010!

In my view, Music is really more like Language, since it comes in a bewildering variety of different local flavors, traditions, genres and levels—each intelligible mainly to a specific group of users and listeners, yet clearly related to one another and drawing from a common behavioral foundation. To be sure, just as experts in Linguistics have evolved their own special vocabulary and symbology for analyzing what languages have in common, and how individual languages work, so Music Theorists too use words and symbols to explore the workings of musical perception, and to analyze specific pieces or styles. If some would call that a more "mathematical" or "scientific" approach to knowing Music, it's easy to see why. But that is only part of the picture.

The unique constructive role musical training can play in mental functioning and development is now better understood, and has recently claimed the attention of a broader public, thanks to the neurobiological research reported by Sacks, Levitin, and others. A typical Music student begins by learning how to transform the body into an Instrument for the generation and control of sound, with or without the support of specially crafted external objects. Then, one learns how to convert musical data that has been symbolically coded as Notation into a mental image of the sounds and patterns so represented; and finally, how to translate that image into the corresponding physical actions needed to execute an appropriate Performance. More advanced musicians will eventually discover how to commit to Memory an entire composition, line by line and section by section, so that they can in turn teach others how to bring it to life.

Composing requires perhaps the highest level of mental preparation, plus the projective power to imagine whole complexes of structured sound, while awaiting the collaboration of other musicians (or, less satisfyingly, of a well-equipped synthesizer) to hear their effect realized. "Composition," according to Canadian composer Alan Belkin, "is first a matter of craftsmanship— refined use of the materials—and only subsequently enters the domain of art." Yet it is hard to overestimate the degree of bodily, mental, and emotional coordination that a child engaged in even the simplest acts of musical invention must bring to bear. Inventing original music used to be considered too difficult for any but advanced graduate students to attempt. No wonder it gets so little attention in conventional school Music programs, especially when compared to what happens in the visual and graphic arts, where even the youngest kids get to make their own original artworks with their own hands.

Acquiring particular skills and techniques is no doubt important, and there are valuable lessons to be learned in the strategy and tactics of artful construction that can prove applicable in other domains as well. But that could hardly be the whole story. In any case, as the American literacy researchers Pearson and Dole point out, "we have to consider the possibility that all the attention we are asking students to pay to their use of skills and strategies and to their monitoring of these strategies may turn relatively simple and intuitive tasks into introspective nightmares…What really determines the ability to comprehend anything is how much one already knows about the topic."

Not every child will take to Music as a preferred venue for creative work; but those who do will continue to need a wide range of experiential support, beyond what computer exercises alone can provide. Learning too is an art of sorts, as Papert eventually recognized, one for which most children are gifted by nature. Yet in a realm like Music—rather a messier one than Pythagoras once assumed!—they will not proceed very far except by engaging continually with other minds and bodies, other natural and imagined worlds. It's hardly suprising, then, that learning to understand and appreciate Music turns out to be no easier to manage than learning to reason logically or solve math problems; nor should it require any less time or life experience than learning to enjoy a good book. At least some of that enabling experience, moreover, might well derive from children's own attempts to write, and to compose.

5) Taking the next step

After several years of non-stop discussion, report-writing, conference-going, and action testing, the end result of our early experiments with Logo Music was a new commercial software product, the brainchild of a brilliant former project assistant, Michael Ross, that distilled and repackaged all we had learned. Although not written in Logo or Lisp, this remarkably compact yet versatile program took the original "blocks + procedures" model about as far as it can go. Called TINKERTUNE®, it was produced for the first generation of Atari personal computers in 1986.

As developers, our first job had been deciding how NOT to build a software system to support children's composing exercises. Anything like the prestigious laboratory computer music systems of the day, aimed at advanced and avant-garde composers (e.g. Music5, Cmusic, Max/MSP, etc.), was ruled out from the start, as too demanding of extra-musical attention. Such daunting complexity seemed quite beyond the capacity of most children or their harried teachers to assimilate.

We also ruled out standard notation, which required reading and writing skills too hard for many beginners to master, and was too tricky to program. Without it, our users would have to forfeit membership in the worldwide community of the musically literate, at least for the time being. But we hoped our program would gain in accessibility by using simpler graphic substitutes that were easier to implement on early-model PCs.

The Atari's design made possible a whole new user interface, complete with joystick-controlled cursor, replacing Logo's command-line input method with a simple but effective form of direct selection from a single on-screen menu. That reduced the user's memory load still further, by keeping in constant view the whole range of available operations, as well as all currently available motivic blocks. At the same time, we wanted to reduce or eliminate the need to name and identify everything. Our use of alphabetic keys to both represent and trigger specific musical fragments, keeping the letter P to represent whatever was just Played, for easy rehearing, was a step in this direction.

During the construction process, both bottom-up and top-down views of the work-in-progress needed to be available. An "assembly line" on the main screen offered a simple way to keep track of where each component fitted into the evolving composition. You could then keep referring back to and reusing previously added material as the piece grew—step by step, through trials and retrials, choices and rejections—into a larger and more satisfactory whole.

Though the choices it offered were still limited, TINKERTUNE was a big step forward. Our hope was that this new program could eventually become a springboard for a whole range of tune-building and composing aids, adapted to various styles. But improvements in personal computing hardware came so fast that the Atari platform was obsolete before we could get our package to market. We did however learn some valuable lessons from the attempt.

B. Back to the Future?

About two years ago, I decided it was time to get back in touch with my earlier interest in computers and musical creativity, reconnect with some like-minded teachers, and help them set up children's composing projects in a few of Toronto's public and separate schools. It should be possible by now, I thought, to find all sorts of suitable software packages to buy for this purpose—but if not, we could always sit down and create some of our own, using all the latest user-friendly development tools.

It turned out, of course, that none of this was as easy to accomplish as I imagined. Nor was it likely to happen soon. The software situation looked particularly grim. Though hundreds of different software products claiming to help people make their own music were now available, I found very few developers whose designs took seriously the needs of children, or addressed the job of learning to compose in a way that was neither simplistic nor trivial. In my frustration, I looked for someone to blame. Is it, I wondered, the fault of the Developers, whose ever-more-feature-laden (and ever more expensive) sequencer packages still dominate the music software market, and although clearly meant for use in professional recording studios, are being increasingly sold to schools and foisted on the very young?

Just then, I heard from Wally Feurzeig—a voice from the past, inviting me to contribute to a conference on the future of the movement I joined a generation ago. But will Music be a part of that future? Will the global convergence of today's enormously more powerful digital media ever allow room for our children to be treated as creators, not just as consumers? Let me share three of the lessons we learned from those early experiments, hoping they may help encourage a new generation of constructionist thinkers, teachers, software developers and musicians to join forces and continue where we left off.

Lesson 1: Computers won't automatically reinforce CREATIVITY

Easier isn't always better. Putting into a child's hands a slick and easy way of notating musical ideas and hearing the notes played back instantaneously and automatically, while impressive and fun, doesn't in itself make satisfying music happen. Facility of execution or ease of recording are only beneficial when linked to an active and fertile musical imagination, fed by wide and deep contact with the musical ideas of other composers, past and present. Today's media-savvy children still need what only prolonged, mindful exposure to good teachers, and to a stimulating variety of other people's music, can offer.

Technical advances are a mixed blessing. To work in any way with Music, even as a beginner, one can hardly escape becoming conversant with technologies of various sorts, particularly those connected with symbolic Representation and sound Production. However, Music's very dependence on facilitating technologies can also create more barriers for the novice.

"Composers" don't wear wigs any more. Some would argue that by adapting computers to assist in so many aspects of music-making, the entire discipline of composing is already being reshaped and redefined, at the expense of unique capabilities that humans have learned to exercise over the centuries. Certainly digital sound processing—the manufacturing, manipulating, massaging, and merchandising of "interesting" new timbres and new mixtures of recorded or electronically-generated sounds—fits well with what computers do best. Is that perhaps why so much attention is focused, in the computerized practice of many composers today, on tweaking and refining the quality of each individual timbre or sound-mix, rather than on larger-scale issues of form or expressive content?

No robots need apply! We wanted to build a playground where kids can exercise and develop some of the skills and habits that would make them better able to imagine and shape their own compositions. But our goal was not to make the computer smart enough to do the composing for us. If anything, it was by examining how and why a computer program fails to deliver a musically satisfying result that we hoped to learn more about the creative thinking of human composers.

One thing computers can do well is to support trial-and-error, what-if testing, and unlimited rehearing and revising of what we have already chosen to record. All these are essential parts of the creative process, but were much harder for novices to do before computerized text processing and instantly playable music notation came on the scene. One suspects, though, that the more we involve computers in automating the generation of sound patterns and resolving issues of abstract compositional design, the less we can count on what bodily involvement and contextual embeddedness have always done to ensure that music retains its powerful expressive appeal.

Lesson 2: Fancy GRAPHICS alone won't save us

Some far-sighted educators, including Papert himself, looked to digital media to free children's learning from the tyranny of Text, which used to be the privileged medium in which all worthwhile knowledge, and the most prestigious creative achievements, must perforce be expressed. This issue is particularly acute for those who work with Music. It's not enough just to get the computer to generate an audible sequence of pleasing sounds. Without the aid of a notated score, a budding composer's effort will leave no trace to work from or refer back to. So we were obliged to invent, and implement, several different kinds of visible support.

Computer graphics has come a long way since those early Atari days. Inventing visually engaging interface designs has become a major preoccupation of software developers. In the twenty-five years since TINKERTUNE was released, a fancy GUI has become obligatory for even the simplest music program. And the variety of designs is mind-boggling. Being able to manipulate playable tokens and experiment with their relative placement on a two-dimensional touch screen, where x = musical time and y = musical pitch, is no small advantage for a beginner. This couldn't be done without the graphic capabilities of a modern personal computer. The skill needed to edit or reorganize recorded sound samples or MIDI tracks, when they are represented visually on a screen, is no longer text-based, and already "concrete" enough to please any good Constructionist. But much more could still be done to integrate visual thinking with musical thinking in the look and feel of software composing utilities.

Visual analogies are not a panacea, however. Especially when it comes to software intended for kids, some of the same problems of function and readability we faced with TINKERTUNE are still around, and deserve special attention. Cross-media "equivalents" that are intended to clarify otherwise invisible relationships or facilitate human-computer interaction need to be handled with particular care and expertise.

Lesson 3: What CHILDREN want is not always what they need

When it comes to creating music, kids want to choose what musical ideas they work with, even if what they like is attuned to what they can share with their peers (especially true for teens). Children are not easily fooled. They know that "Music" equals "Songs," and that Songs are About Something. What initial appeal do abstract Composing exercises have for the average kid? Not much, except perhaps as a game teachers might want to let them play instead of doing regular schoolwork. However, children can be easily led. Many young people now carry their own music player with them everywhere, and have access to an unimaginably vast range of recorded music from which to choose their personal listening fare. If what all those iPods are actually pumping through all those earbuds is in fact nothing but the same commercially driven pop-star hits everyone else is listening to, who is to blame for that?

It's clearly not enough just to let the kids have "their" music, even if we can no longer insist that they be taught to revere "ours". This is where a constructionist pedagogy that includes exercises in Composing may have most to contribute. Without skilled, discriminating, empathetic listeners, the promise of universal musical enlightenment offered by today's convivial digital media turns into nothing more than a sick joke. Imagine a World Cup football match, telecast to every corner of a world in which no one has ever played the game!

At the same time, it is important not to make Composing too quick or too easy. If the process of musical construction becomes too facile, too automatic, or too random, that trivializes the exercise and cheapens the experience. And as Alan Kay has warned: "Media can also lure us into thinking we are creating by design when in fact we are just tinkering."

Our experience supports Kay's conviction that children need and will thrive on truly difficult tasks, as long as the difficulties are not overwhelming. We know that schoolchildren will gladly work hard and long at something as absorbing and fun as constructing their own music. But they must be free to focus full attention on musical materials and musical results, and not forced by the system to fiddle with extraneous details. Provided it is carefully designed and thoughtfully integrated with other approaches, special-purpose computer software can indeed help, by opening vast new possibilities for creative exploration of musical structure at every level. At least then, whatever smarts may ultimately accrue are more likely to be the child's, not the computer's.

* * * * * * *

References

Bamberger, Jeanne (1991) *The Mind Behind the Musical Ear: How children develop musical intelligence*. Harvard University Press, Cambridge.

Beckwith, Sterling (1975a) *The Interactive Music Project at York University*. ONTERIS, Toronto.

Beckwith, Sterling (1975b) *Talking Music with a Machine.* In College Music Symposium, Spring.

Belkin, Alan (2002) *Encouraging musical creativity*. Available online at: http://www.musique.umontreal.ca/personnel/Belkin/creativity.html

Kay, Alan (1991) *Computers, networks, and education*. In Scientific American, September.

Lanier, Jaron (1997) *Music, nature, and computers*. In Terra Nova, Summer.

Levitin, Daniel (2008) *The World in Six Songs: How the musical brain created human nature*. Dutton, New York.

Papert, Seymour (1993a) *Mindstorms: Children, computers, and powerful ideas*. 2nd edition, Basic Books, New York.

Papert, Seymour (1993b) *The Children's Machine: Rethinking school in the age of the computer*. Basic Books, New York.

Pearson, P. D. & Dole, Janice (1987) *Explicit comprehension instruction: a review of research and a new conceptualization of instruction*. In Elementary School Journal, 88/2.

Ross, Michael (1986) *TINKERTUNE® software manual*. Minerva Research, Victoria, BC.

Sacks, Oliver (2007) *Musicophilia: Tales of music and the brain*., Knopf, New York.

Programming Standing Up: Embodied Computing with Constructionist Robotics

Matthew Berland, *matthew.berland@utsa.edu*
Dept. of Interdisciplinary Learning & Teaching, University of Texas at San Antonio

Taylor Martin, *taylormartin@mail.utexas.edu*
Dept. of Curriculum & Instruction, University of Texas at Austin

Tom Benton, *soibois@hotmail.com*
Dept. of Curriculum and Instruction, University of Texas at Austin

Abstract

IPRO (shown below) is a mobile, constructionist, virtual robotics programming environment designed to teach computational literacy. Students use a simplified programming language to control the behavior of a robot agent as it undertakes various collaborative and competitive tasks. Each student programs his/her own robots using a handheld device, and the robots compete on a shared stage. The project uses handhelds to encourage collaboration and embodied cognition through physical movement and the sharing of content while programming. The design is based on other robotics environments for teaching introductory programming in which there have been measurable learning gains (Berland, 2008; Martin, 2007; Wilensky & Stroup, 1999a). IPRO is structured as a participatory simulation in that students will participate in a constructionist shared space (see Wilensky & Stroup, 1999b).

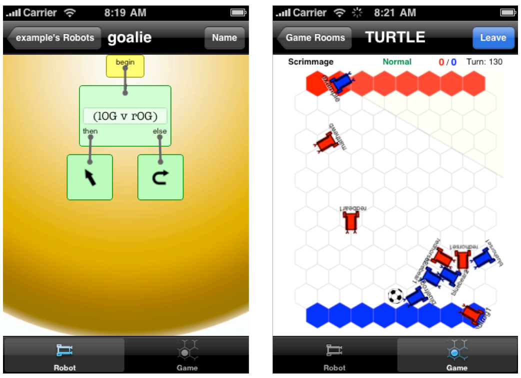


Figure 1 - IPRO Circuit and Play Modes

Constructionist programming tasks are especially well suited to this type of collaboration because they involve complex, concrete tasks with a strong connection of the conceptual and the technical. Students collaboratively refine their conceptual understanding by freely sharing their technical products. Our goal is that students in the project will approach programming as an active, physical task, and that they will be motivated to engage with the content as a result. The design of IPRO is focused on determining pathways to stronger understanding of programming content as well as to levels of increased motivation.

Keywords (style: Keywords)

Embodied cognition; robotics; collaborative; handheld; programming; computational literacy

Introduction

Given the computer's iconic status as the quintessential amodal, decontextualized thinking device, discussing embodied programming is a bit vague at best. However, if we are to understand computational thinking (CT) and programming from a constructionist perspective, considering the implications of embodied programming is essential.

There is a profound disconnect between the excitement surrounding the potential of computer-proficient youth and the actual act of programming a computer. Typically a solitary art, it often invokes the stereotypical image of a young man hunched over a glowing screen, and the path to programming proficiency is frequently presumed to demand monk-like devotion to the study of arcane syntax and keystrokes. Even for computationally literate young people who happily use software in imaginative and innovative ways, learning programming may hold little appeal when pursued in solitude. Our challenge is clear: how can programming in the classroom be recast as engaging, inclusive, and social?

While instruction in many disciplines has been made more active, mobile, and collaborative, programming instruction has resisted those advances (Ben-Ari, 2001). The goal of this project is to introduce an element of mobility into programming instruction and encourage students to collaborate as they move and congregate. We call this "Programming Standing Up". The importance of engaging and effective programming instruction cannot be understated: programming is "hard to learn" (Guzdial, 2004), a core thinking skill (diSessa, 2000), and a core job skill. Some benefits of mobility with constructive work are also well established – Klopfer, Squire, & Jenkins (2002) showed that students engage in more complex problem solving about real world content when they are able to work together in a physical space.

Constructionist programming tasks are especially well suited to this enhanced collaboration because they involve complex, concrete tasks with a strong connection between the conceptual and the technical. Students collaboratively refine their conceptual understanding by freely sharing their technical products. Pedagogy in programming classes has often valued reproduction of 'authentic' professional practice over innovative pedagogy (Ben-Ari, 2001). However, as we know from Smith, diSessa, & Roschelle (1993), authentic demonstration of expert practice is not necessarily the best path to understanding. Our goal is that students in the project will approach programming as an active, physical task, and that they will be motivated to engage with the content as a result.

To that end, we are developing, implementing, and deploying a constructionist mobile, collaborative programming platform focusing on the design, generation, and evaluation of algorithmic knowledge, strategies, and models; these are the basic elements of computer science education as described by Robins, Roundtree, and Roundtree (2003).

IPRO (for both "iPod Robotics" and "I (can) program!") is a virtual robotics environment that builds upon the previous designs of the authors, along with previous research on the development of novice programming environments (a more detailed description is below). Students use a robust programming language to control the behavior of a robot agent as it undertakes various collaborative and competitive tasks. The design is based on other robotics environments for teaching introductory programming in which there have been measurable learning gains (Martin, 2007). IPRO is structured as a participatory simulation in that students will participate in a constructionist shared space (see Wilensky & Stroup, 1999b, for more detail).

The research design around IPRO is focused on determining pathways to stronger understanding of programming content as well as to levels of increased motivation.

Our primary hypotheses are:

1. Learning to program on mobile devices will lead to a greater incidence of on-task interaction by students with their peers as well as with the world around them.
2. These new interactions will lead to an operationalized understanding of key computational concepts, and will flatten the learning curve as students move on to new programming environments.
3. These new interactions will also lead to increased student engagement with programming.
4. The process of learning to program standing up will be dramatically different from learning to program at a stationary computer.

Constructionist Pathways to Computational Literacy

The term *computational literacy* has often been used to describe proficient usage of a few standard desktop computer applications. For example, as a secondary school teacher, one author taught a class called "computer literacy," which focused on Microsoft Word, Excel, and PowerPoint. Papert (1980) argues that computational literacy should instead mirror ‘print literacy’ more closely. The modern conception of print literacy does not stop at reading - the print literature individual should be able to express herself in writing as well. As such, computational literacy should not stop at the ability to use computer software; rather, it should include the ability to create and/or manipulate computer software (or hardware) to communicate and disseminate ideas. The expressive and authoring aspects of computational literacy have been long ignored in the pre-collegiate curriculum, but a shift towards a more participatory picture of media and technology education is underway; this shift is supported by research while reflecting changing relationships between young people and technology (Jenkins, 2006).

Much of the modern constructionist work uses computational literacy as a focus to teach complex content (e.g., Blikstein & Wilensky, in press). However, different forms of content and different sets of students are amenable to different approaches, and teaching ‘deep’ computational literacy is not identical to using computational literacy to teach other content (such as mathematics). How do we help students develop this type of deep computational literacy? This is a fundamental question of constructionism. Embodied cognition provides one key to answering this question.

Embodied Cognition and IPRO

Embodied cognition recognizes and tries to understand how being in a body and interacting with a physical world shapes and impacts the development of thinking, problem solving, and learning. The typical picture of programming alluded to earlier, the solitary male hunched over a computer monitor, seems the complete antithesis of an embodied activity. However, programmers often describe the computer as an extension of themselves, as a highly responsive tool for interacting with the world (diSessa, 2000; A. Randall, writer of Perl 6 (O'Reilly), personal communication, 2010). While we will not turn our students into expert programmers overnight, we can help students develop programming experience in ways that might lead to this type and degree of proficiency.

In placing the programming environment on a mobile device, which a student uses while standing up in a group of other students, we change fundamental aspects of the system characterizing the relationship between the learner and what is to be learned. That system now includes new pathways, such as moving one's own body or other objects in the classroom, the ability to easily show other students what is happening with your simulation or with the code underneath it, and the ability to organize in groups that afford more direct sociability than if students (or pairs) are seated at individual computers. Considering some of the main themes of embodied cognition can help us identify implications of this changed system. If cognition evolves from perception and action (Anderson, 2007; Fischer, 1980; Seitz, 2000; Varela, Thompson, & Rosch, 1993; Wilson, 2002), programming a robot will be easier if one can attempt to embody a robot during the process. This relies in large part on possibilities for off-loading some cognitive work to the environment (Hutchins, 1995; Kirsh, 2008); in this case that might include acting out what one thinks certain commands or combinations of commands might do, potentially leading to better programming and clearer communication between students about programming. Embodied cognition claims that much less of what we do is guided by plans made ahead of time, and more is guided by on-the-fly tightly coupled act-plan cycles (Suchman, 1988). If so, the mobile programming environment affords exactly this type of work; fresh ideas can be explored in physical space, programmed, and evaluated in virtual space without interruption. This tight coupling of the physical and the virtual is important to have in mind while viewing this project through the lens of embodied cognition. The authors have done significant research on the effects of virtual versus physical environments on instruction in programming as well as mathematics problem solving and have found unique benefits in both cases. For example, virtual environments can help struggling children learn mathematics faster because children can see any or all steps of a problem solution as many times as they like (Martin & Schwartz, 2005). Meanwhile, working in the physical environment is often linked to deeper conceptual processing and understanding (Penner, Lehrer, & Schauble, 1998). We hypothesize that the IPRO environment could capitalize on the benefits of both these approaches.

Why Constructionist Virtual Robotics?

Constructionist research suggests that IPRO will be beneficial for learning because it approaches learning as an active participant (Harel & Papert, 1990); embodied cognition research suggests how Programming Standing Up could change the nature of the programming task to make it more accessible. But why do this programming in the context of robotics?

Strong Support for Constructionism

Robotics have been associated with constructionist approaches since its beginnings (Papert, 1980). It continues to be a core element of many constructionist projects (e.g., Hancock, 2003; Portsmore, 1999; Resnick & Ocko, 1991). As Wilensky (2000) notes, tools that utilize the individual components needed to complete the aggregate can both help the student understand the final concept but also allow the investigator to understand the process of learning. Wilensky differentiates “black box” projects in which subjects begin in the middle of the process of creation with “glass box” projects in which subjects can see the process of creation from start to finish. Programmable autonomous robotics curricula provide a consummate example of this kind of “glass box” work.

Significant Evidence to Suggest Learning Benefits for Math and Science

As part of a broader initiative to improve math and science education, several thousand schools have implemented robotics classes and clubs for K-12 students. Anecdotal evidence about how robotics classes have improved student interest level, creativity, and reasoning skills are well documented (Genalo & Gilchrist, 2006; Lau, McNamara, Rogers, & Portsmore, 2001). Recently, work on the specific benefits of robotics and LEGO-like toys has been materializing. Using a simulated standardized math test, Lindh and Holgersson (2007) tested 996 fifth and ninth grade students in Sweden. They found that, for students who were slightly below average in math, taking robotics in 5th grade improved math test scores relative to their counterparts who did not take robotics. Wolfgang, Stannard, and Jones (2003) followed a group of 27 students from pre-K through 12th grade and discovered several significant correlations. Pre-K students who were adept with (non-robotic) LEGO blocks went on to score high on standardized high school tests, took more honors and higher math courses, and had a higher weighted grade point average in math courses.

Access issues

On one hand, the immediacy and physicality of robotics would appear to make it a natural context for teaching programming. Coded commands can instantly be visualized as a simulation during testing – following a successful test, students can program their actual robot and get the satisfaction of seeing their programming "come to life." However, robotics is sometimes characterized as having narrow appeal, primarily to the types of students already sitting in the seats of our college engineering classrooms. As such, an important element of our research agenda is examining how IPRO can increase participation of underrepresented groups in engineering. Considering girls as one of these underrepresented groups, an examination of the literature suggests favorable reading of how girls benefit from robotics courses and competitions. Beisser (2006) reports that girls immersed in a LEGO/Logo environment demonstrate significant improvement in self-efficacy beliefs regarding computer use and their likelihood of being computer professionals in the future. Weinburg and colleagues (2007) show that participation in Botball competitions led to increases in both self-efficacy perceptions and ratings of interest in STEM careers for 7th grade girls. In a qualitative case study of girls in a Botball program, Stein & Nickerson (2004) found that girls were equally as interested in the competitive nature of the game robotics environment as boys and were not driven away by it.

We believe this evidence suggests that the issue of equity and robotics is a real problem and worthy of further study, particularly in light of the widespread use of robotics activities in classrooms and after school clubs.

IPRO: A Constructionist Mobile Programming Environment

IPRO is an iPhone & iPod Touch virtual robotics programming environment and game space that we are developing for this project. It is made up of two fundamental components: a development environment (‘the board’) for the IPRO language and a shared game space (‘the field’) where students’ virtual robots will coexist.

Elements of IPRO

The Field

The field (shown in Figure 2) is where students’ robots compete in teams, collaboratively and competitively accomplishing tasks and working towards goals. Students will each design their own robot on the board based on challenges to come in the field. The simplest field activity is a variant of soccer. Soccer is a game in which several studies have shown success to scaffold students in robotics and computational literacy (Sklar, 2002).

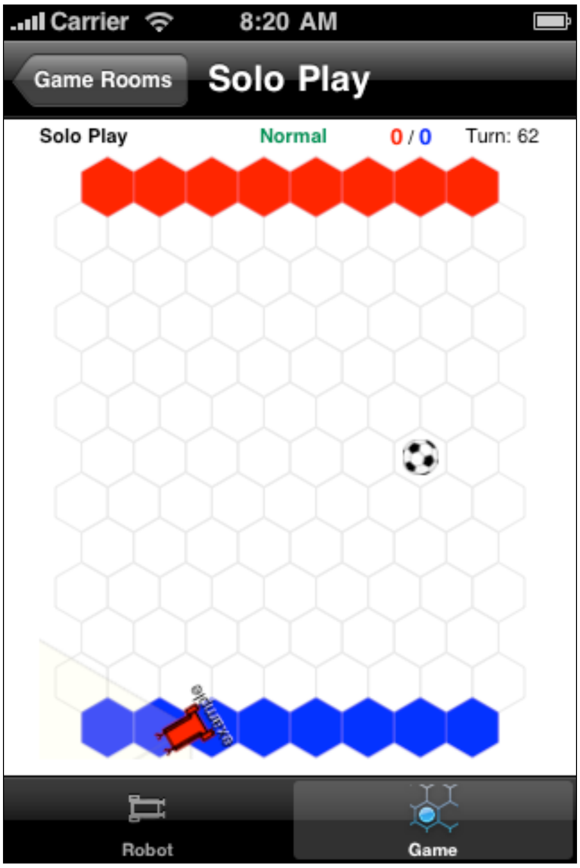


Figure 2 - The IPRO Field

The Board

The board is where students program their robots using single-layer encapsulated functions. The language and function are similar to programming in Scheme, a programming language commonly used for teaching and learning (such as in the textbook by Abelson, Sussman, & Sussman, 1985).

IPRO Activity: IPRO-Soccer

The IPRO framework can be used for a variety of activities, but the initial version is designed to support multi-agent online games of robot soccer. There may be up to 12 robots on the field at any given time, divided equally into red and blue teams as they enter the game. Each team attempts to move the ball into the goal (both shown in Figure 3). Each student designs one robot, but they must work in concert with teammates in order to be successful. The benefits of framing IPRO activities in terms of soccer are:

1. It is a familiar game to many teachers who have used robotics in their classroom support.
2. There is a significant amount of outside information about strategies for both human and robot soccer.
3. There is relatively rapid feedback about the success of a strategy.
4. It scales well with robotics and can be attempted with physical robotics as well (Sklar & Eguchi, 2004).

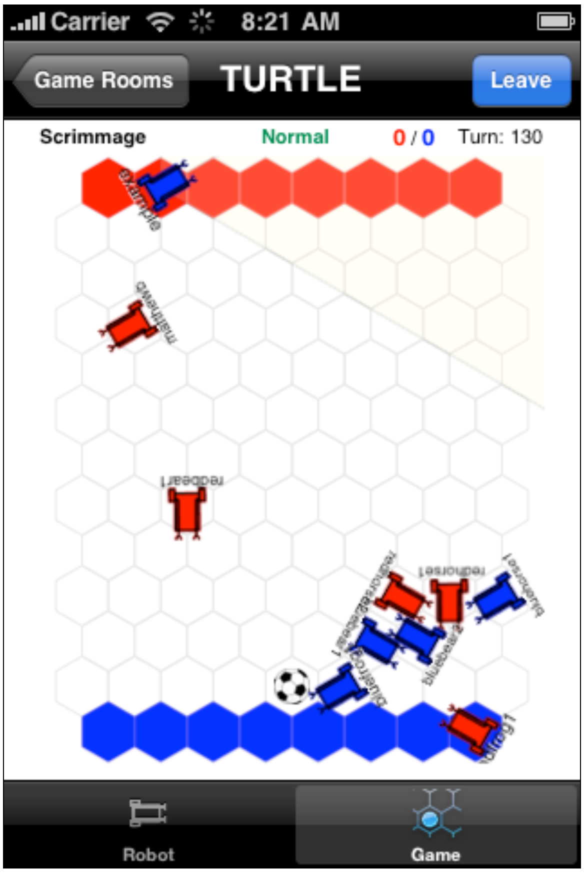


Figure 3 – IPRO Team Play

IPRO Language

The IPRO language is a visual programming language based deriving from the Scheme programming language implementing a simple functional reactive programming paradigm (e.g., Cooper & Krishnamurthi, 2006), in that it uses the concept of signals rather than constants. An example is shown below in Figure 4.

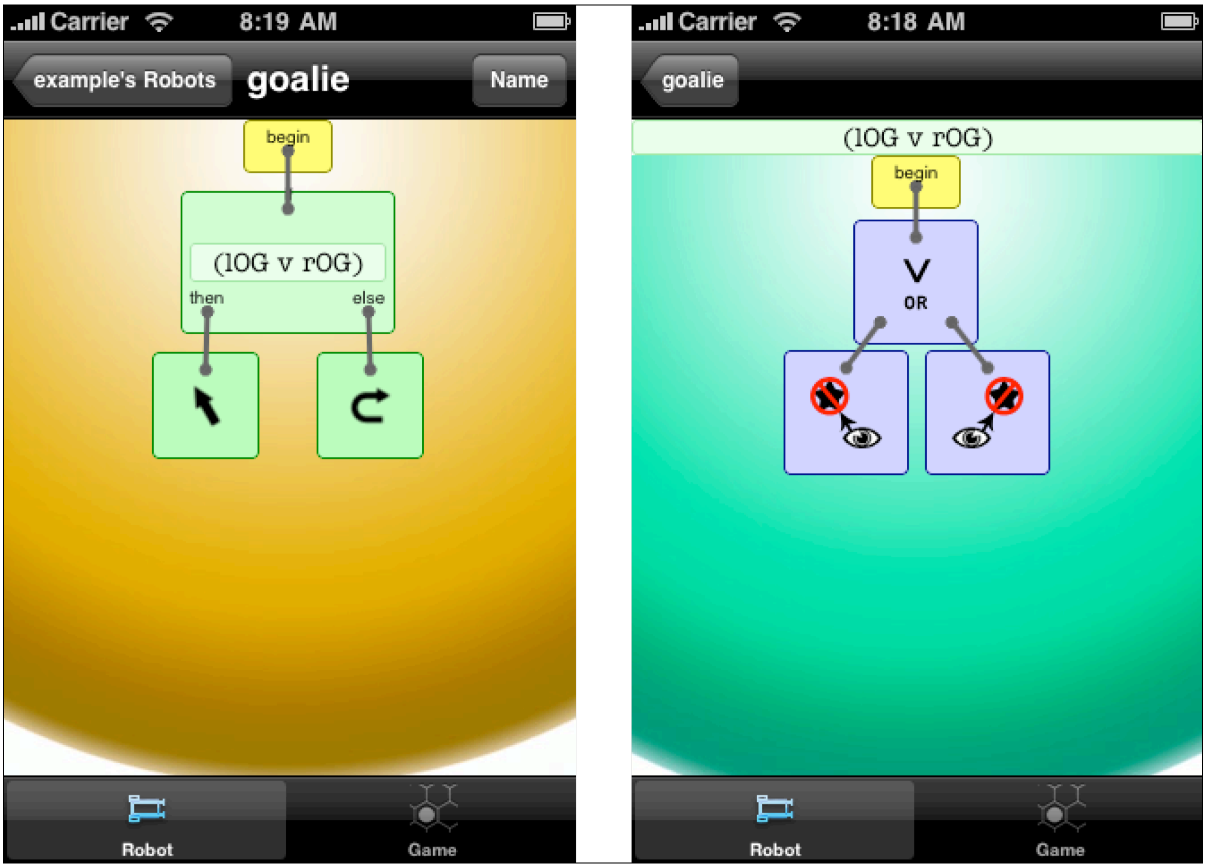


Figure 4 - IPRO Logic Flows

The fundamental components of a program are sensors and motors. Each IPRO virtual robot has a symmetrical left and right version of each sensor, and it can move in one of four directions on a hexagonal grid each time step. The semantics of the language remain fairly simple:

1. During each time step, a set of conditional logic branches is evaluated.
2. Each conditional branch is true or false based on the logic of the sensors. The default set of sensors:
   a. ROBOT-SENSOR = Returns a value that corresponds to an inverse of the distance from the agent-robot to the nearest robot.
   b. GOAL-SENSOR = Returns a value that corresponds to an inverse of the distance from the agent-robot to the agent-robot's targeted goal.
   c. BALL-SENSOR = Returns a value that corresponds to an inverse of the distance from the agent-robot to the nearest ball.
3. The output of the conditional logic must necessarily be some action in the virtual space: MOVE-FORWARD-LEFT, MOVE-FORWARD-RIGHT, MOVE-BACKWARD-LEFT, MOVE-BACKWARD-RIGHT, or TURN-RIGHT.

There are several important differences between IPRO and Scheme.

1. IPRO is significantly simplified, only using those primitives that are relevant to an activity.
2. IPRO is constantly evaluated, so that the student can see the effects of her program immediately and make changes accordingly.
3. No Syntax or symantic errors are possible – all possible programs compile. However, they may not all be relevant to an activity.

Understanding & Modeling the Process of Learning to Program

The IPRO environment presents a unique opportunity to model how students learn and collaborate effectively in the classroom. By collecting data on how students share programs and programming wisdom with each other using their handheld devices, we can, over time, characterize the collaboration that happens as well as create predictive models.

These novel forms and structures of data, along with the analysis, visualization, and modeling techniques we propose, have not heretofore been accessible to school-based studies. Berland (2008) uses multi-agent network theory to illuminate both qualitative/exploratory data and quantitative performance data in mixed-methods studies (using the model from Abrahamson, Blikstein, Lamberty, & Wilensky, 2005), but thus far the resources available thanks to the cyber infrastructure of mobile hardware (GPS, short-range wireless connections, and custom applications) have not been utilized to map and model collaboration.

IPRO collects data not only on students' individual programs but on how they share specific parts of those programs. A concrete example of how this might look in a robot soccer game follows:

1. Juliana builds program to search for the soccer ball. Her program includes three functions, one of which checks to see if the robot is pointed in the correct direction ("correct-direction?").
2. Maria is building a striker, but her striker should only target the correct goal, so she asks Juliana for help. She walks over to Juliana and asks to use her "correct-direction?" function.
3. Juliana shares the function with Maria by utilizing the "share" button on her interface.

All of this data is logged by the system, so we can follow the sharing as it happens see who is collaborating with whom. We can analyze this data statically as well as dynamically.

Conclusions

Like engineering, programming in the classroom does not lend itself to clean cycles of textbook-administered instruction and assessment. However, as computational and systems-thinking skills become increasingly vital, understanding and improving how they are learned has become a necessity. As such, our paper addresses areas of both educational theory and practice. This study is designed to innovate the experimental study of embodied cognition theory as well as use constructionist robotics to better understand the collaborative learning process. Our goal is to reconsider programming instruction and generate testable, predictive models to usefully guide future research. Positive findings will contribute to improving the practice of computer science instruction in the classroom by demonstrating improved learning outcomes in core content areas and presenting programming as an engaging activity for all learners.

References

Abelson, H., Sussman, G.J., Sussman, J. (1985). *Structure and Interpretation of Computer Programs*. New York: MIT Press and McGraw-Hill.

Abrahamson, D., Blikstein, P., Lamberty, K. K. & Wilensky, U. (2005). Mixed-media learning environments. *Paper presented at the annual meeting of Interaction Design and Children 2005*, Boulder, Colorado.

Anderson, M. L. (2007). How to study the mind: An introduction to embodied cognition. In F. Santoianni & C. Sabatana (Eds.), *Brain development in learning environments: Embodied and perceptual advancements*. Cambridge: Cambridge Scholars Press.

Beisser, S. R. (2006). An Examination of Gender Differences in Elementary Constructionist Classrooms Using Lego/Logo Instruction. *Computers in the Schools, 22*, 7-19.

Ben-Ari, M. (2001). Constructivism in computer science education. *Journal of Computers in Mathematics and Science Teaching 20*(1), 45–73.

Berland, M. (2008). *VBOT: Motivating Complex Systems and Computational Literacies in Virtual and Physical Robotics Learning Environments*. Retrieved from ProQuest Digital Dissertations. (AAT 3307005)

Berland, M., & Wilensky, U. (2004). Virtual robotics in a collaborative constructionist learning environment. *The annual meeting of the American Educational Research Association*, San Diego, CA, April 12 - 16, 2004.

Blikstein, P., & Wilensky, U. (in press). MaterialSim: A constructionist agent-based modeling approach to engineering education. In M. J. Jacobson & P. Reimann, (Eds.), *Designs for learning environments of the future: International perspectives from the learning sciences*. New York: Springer.

Cooper, G.H. & Krishnamurthi, S. (2006). Embedding Dynamic Dataflow in a Call-by-Value Language. *15th European Symposium on Programming*, Vienna.

diSessa, A. A. (2000). *Changing Minds: Computers, Learning, and Literacy.* Cambridge, MA: MIT Press.

Sklar, E. and Eguchi, A. (2004). RoboCupJunior -- Four Years Later. *Proceedings of the Eighth International RoboCup Symposium (RoboCup-2004)*

Genalo, L., & Gilchrist, J. (2006). Home Schoolers in an Engineering/Education K12 Outreach Program. *Proceedings of the American Society for Engineering Education Annual Conference and Exposition, 2006*, Retrieved March 1, 2009, from http://soa.asee.org/paper/conference/paper-view.cfm?id=1209.

Guzdial, M. (2004). Programming environments for novices. In S. Fincher and M. Petre (Eds.), *Computer Science Education Research* (pp. 127-154). Lisse, The Netherlands: Taylor & Francis.

Hancock, C. (2003). *Real-time programming and the big ideas of computational literacy.* Unpublished doctoral dissertation, MIT.

Harel, I., & Papert, S. (1990). Software design as a learning environment. *Interactive Learning Environments, 1*(1), 132.

Hutchins, E. (1995). *Cognition in the Wild*. New York: MIT Press.

Jenkins, H. (2006). *Confronting the Challenges of Participatory Culture: Media Education for the 21st Century*. White Paper for the MacArthur Foundation.

Kirsh, D. (2008). Distributed cognition: a methodological note. In Itiel Dror and Stevan Harnad (Eds.) *Cognition Distributed*. Philadelphia: John Benjamins Publishing Co.

Klopfer, E. & Squire, K. & Jenkins, H. (2002). Environmental Detectives: PDAs as a Window into a Virtual Simulated World. *IEEE International Workshop on Wireless and Mobile Technologies in Education*, 2002, Växjö, Sweden. 95-98.

Lau, P., McNamara, S., Rogers, C., & Portsmore, M. (2001). LEGO Robotics in Engineering. *Proceedings of the American Society for Engineering Education Annual Conference and Exposition, 2001*, Retrieved March 1, 2009, from http://soa.asee.org/paper/conference/paper-view.cfm?id=16121.

Lindh, J., & Holgersson, T. (2007). Does lego training stimulate pupils' ability to solve logical problems? *Computers & Education. 49*, 1097-1111.

Martin, F. (2007). Little Robots that Could: How Collaboration in Robotics Labs Leads to Student Learning and Tangible Results. *Intelligent Automation and Soft Computing, 13*(1).

Martin, T. & Schwartz, D.L. (2005). Physically distributed learning: Adapting and reinterpreting physical environments in the development of fraction concepts. *Cognitive Science, 29*(4), 587-625.

NRC (2009). *Report on a Workshop on the Scope and Nature of Computational Thinking.* Retrieved December 15, 2009 from nap.edu: http://www.nap.edu/catalog.php?record_id=12840.

Papert, S. (1980). *Mindstorms*. New York: Basic Books.

Penner, D.E., Lehrer, R., & Schauble, L. (1998). From physical models to biochemical systems: A design-based modeling approach. *Journal of the Learning Sciences, 7*(3&4): 429-449.

Portsmore, M. (1999). ROBOLAB: Intuitive Robotic Programming Software to Support Life Long Learning. *APPLE Learning Technology Review*, Spring/Summer, 1999.

Resnick, M., & Ocko, S. (1991). LEGO/Logo: Learning Through and About Design. In (ed. by I. Harel and S. Papert (Eds.), *Constructionism*. Norwood, NJ: Ablex.

Robins, A., Roundtree, J., & Roundtree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer Science Education, 13*(2), 137-172.

Seitz, J. (2000). The Bodily Basis of Thought. *New Ideas in Psychology, 18*, 23-40.

Sklar, E. & Parsons, S. (2002). RoboCupJunior: a vehicle for enhancing technical literacy. *Proceedings of the AAAI-02 Mobile Robot Workshop*.

Stein, C. & Nickerson, K. (2004). Botball Robotics and Gender Differences in Middle School Teams. *Proceedings of the 2004 American Society for Engineering Education Annual Conference*.

Suchman, L. (1988). Representing practice in cognitive science. *Human Studies, 11*(2-3), 305-325.

Varela, F. J., Thompson, E., & Rosch, E. (1993). *The embodied mind: cognitive science and human experience*. Cambridge, MA: MIT Press.

Wilensky, U. (2000). *Modeling Emergent Phenomena with StarLogoT*. Retrieved December, 2000, from CONCORD.org.

Wilensky U., & Stroup, W. (1999a). *HubNet*. Center for Connected Learning & Computer-based Modeling. Northwestern University. Evanston, IL.

Wilensky, U., & Stroup, W. (1999b). Learning through Participatory Simulations: Network-Based Design for Systems Learning in Classrooms. *Computer Supported Collaborative Learning (CSCL'99)*. Stanford University: December 12 - 15, 1999.

Wilson, M. (2002). Six views of embodied cognition. *Psychonomic Bulletin & Review, 9*(4), 625-636.

Wolfgang, C., Stannard, L., & Jones, I. (2003). Advanced Constructional Play with LEGOs Among Preschoolers as a Predictor of Later School Achievement in Mathematics. *Early Child Development and Care, 173*, 467-475.

Pendulum: A Programming Toolkit for the Development of Physically Interactive Art Applications

Tom Benton, *soibois@hotmail.com*
Dept. of Curriculum & Instruction, University of Texas at Austin

H. Taylor Martin, *taylormartin@mail.utexas.edu*
Dept. of Curriculum & Instruction, University of Texas at Austin

Matthew Berland, *matthew.berland@utsa.edu*
Dept. of Interdisciplinary Learning & Teaching, University of Texas at San Antonio

Workshop Description

This workshop will introduce participants to Pendulum, a programming toolkit designed to acquaint learners with computational thinking, programming, and interface design while also exploring the role of play and embodied cognition in these areas. Pendulum includes a continually evolving library of modular abstractions developed in Puredate (Pd), a graphical programming language used for creating interactive musical and visual applications. While Pd can accommodate nearly any kind of conceivable input, Pendulum focuses on the Nintendo Wii controller as an end-user input device; trading mouse and keyboard-based input for position, acceleration, and multi-axis rotation presents a number of exciting opportunities and challenges to the programmer in addition to providing an opportunity to study how dynamic physical activity and consequently rich input impact learning and practice in programming.

As a constructionist learning technology, Pendulum takes advantage of a number of features of the Pd language. Modular abstractions permit fruitful exploration and theory-making by allowing learners to incrementally increase the complexity and sophistication of their programs; for instance, a user might begin by simply plugging his or her Wii input into a virtual scope, visualizing rotation around a particular axis. Next, this same data could be expressed numerically and the user could scale it or apply any kind of mathematical function to it. At any point in this process the input could be directed towards one or more of many available functions for synthesis and manipulation of audio and video, but the learner has ample opportunity to test and refine his or her understanding along the way. As a dataflow programming language, there is no compiling and programs can even be editing while running, providing literally instantaneous feedback. Because Pd is open-source and existing programs can be modularized rather easily, Pendulum applications are highly shareable in whole and in part, in face-to-face learning communities or across the internet. Perhaps most powerfully, changes in programming understanding and computational thinking have dramatic payoffs in breadth of potential creative expression. For instance, learning a new computational concept such as recursion could completely reinvent a learner's notions of how the virtual musical instrument that they've constructed might manipulate or produce audio. Subsequently exploring potential applications could further refine their computational recursion understanding.

The workshop will begin with a short tour of the Pendulum system and the presentation of some projects that have been developed by learners of a variety of ages and backgrounds. Following this, participants will be able to discuss, tinker, and construct, as a larger group at a main workstation or at one of several satellite stations. All are invited to download the appropriate software (www.activelearninglab.org/pendulum) prior to the workshop and use their own machines if they like, with the presenters providing as many Wii controllers as they can make available. Near the end of the workshop participants can present what they have created if they choose to and the presenters would welcome questions, feedback, and a discussion of related topics of interest.

Keywords

keyword; programming, computational thinking, embodied cognition, music, art, dance, interfaces, open source

Wired, But Not Connected

Dr. Boris Berenfeld, *bberenfeld@ilaet.co.uk*
President, the International Laboratory for Advanced Education Technologies

Abstract

Much money has been spent linking schools to the Internet, but are students actually connected intellectually? Do they harness the Internet for anything more than retrieving additional content? This paper introduces a new instructional model that we call *telecollaborative inquiry* in which connectivity recasts how students learn. Telecollaborative inquiry builds communities of networked classrooms that engage students in distributed, collaborative knowledge-construction. Mirroring professional and scientific communities of practice, this paradigm leverages Internet connectivity, technologies, and social networking to teach content and foundational skills like critical thinking, communications, information literacy, and teamwork. It enables synchronized investigations that produce datasets and intellectual discourse that are richer than what individual classrooms can deliver. It has the power to transform education and justify the investments in wiring schools. Telecollaborative inquiry-based curriculum was first piloted in the 1990s as the Global Lab project in 30 countries and today, an updated version of Global Lab (v.3.0) deploying Web 2.0 advances is being piloted in 150 upper-elementary classes throughout Russia with plans to scale worldwide. Based on these trials, the developers are innovating a scaffolded curricular design based on granular instructional modules called Global Learning Units (GLUs™). Each GLU converts a specific instructional topic into a bite-sized telecollaborative investigation, providing all the resources and tools needed to deliver telecollaborative inquiry. When aligned with instructional objectives, the progression of GLUs covers the scope, sequence, and content of traditional curricula, building science content and process skills more effectively than single-classroom inquiries. Tightly integrating content, data collection and analysis, and student communications into a Web-based curricular infrastructure, GLUs provide the framework and scaffolding to make telecollaborative inquiries a reality in mainstream science classrooms. They offer educators a strategy for implementing telecollaborative inquiry-based curricula that will enable students across mainstream education to construct the knowledge and skills necessary for achievement in higher education and professional endeavors.

Keywords

telecollaborative, inquiry, collaborative, science education, constructivist, project-based, networking

A New Generation of Knowledge & Skills

Billions of dollars have been spent worldwide to wire classrooms to the Internet and the common rationale is connectivity will transform education. Yet in typical classrooms today, the Internet is a digital pipeline—in effect, a digital library. Open it and content pours out in a one-way flow—to students, not from them. Certainly, this has pedagogical value, but to date, the Internet has hardly renewed how teachers teach and students learn. Classrooms remain insular and teacher/textbook-centered, and hands-on science projects are often parochial.

This model of classroom education still reflects the needs of the Industrial Revolution and analog economies when workers were expected to follow instructions rather than solve problems. Economic and social modalities have evolved, however, as the digital revolution unfolded in the late 20th century and the Internet Age reached full stride in the beginning of the 21st century.

Worldwide, business, education, and political leaders have recognized that primary and secondary education must adapt to the realities of rapidly-changing economies.

National economies are increasingly intertwined in a global grid, demanding that workers have international perspectives. Science, technology, engineering, and mathematics (STEM) are producing the innovations and technological advances that fuel economic growth and prosperity, placing a premium on STEM literacies. Moreover, how people work is changing as workplaces capitalize on the communications, productivity, and transaction processing afforded by digital technologies. Freed from geographical constraints, people increasingly collaborate in virtual groups to leverage expertise and perspectives. A comprehensive survey of corporate human resource managers revealed that after professionalism and a work ethic, collaboration was the skill most valued for new workers (Casner-Lotto & Barrington, 2006).

As a result, there are widespread calls for schools to teach not only content, but also such skills as teamwork, critical thinking, problem solving, communications, and information literacy. Moreover, schools must teach STEM-related skills like the scientific method, data evaluation and analysis, and objectivity.

To meet these needs, schools have to focus less on individual learning and more on group learning by introducing project-based inquiries into classrooms. Students should actively participate in their educations, collaborating in hands-on activities to construct knowledge while building foundational skills. Although more research is needed, evidence indicates computer-based collaborative learning can enhance higher-order thinking, student satisfaction, and improved productivity (Resta & Laferrière, 2007).

Yet collaborative inquiries within classrooms, when they occur, are face-to-face. Students do not participate in virtual teams communicating with web-based multimedia technologies, as they will in the workplace. Learning remains local rather than distributed as students use the Internet simply to access content that augments their textbooks. Data sets that students generate locally are for single points in time and location, and thus cannot be compared and analyzed with larger, protocol-bound datasets, as is done routinely in business or science.

Additionally, collaborative approaches are not mainstream in classrooms. It is difficult for teachers trained in 20th century pedagogical strategies to merge standards-based content with building high-level process skills such as collaborative problem-solving. Individual teachers, working on their own, frequently lack the scaffolding and resources to deliver collaborative inquiries. Ironically, on their own time, children use the latest digital technologies to intellectually engage each other and are very fluent in virtual interactions, knowledge building, and networking. These are the modalities that must be brought into classrooms if schools are to offer next-generation learning environments.

Telecollaborative Inquiry

Beginning in 1991, a new learning paradigm has been in development in the form of the Global Lab project (Berenfeld, 1994, Berenfeld, 1999). Developed at TERC (www.terc.edu) with extensive support from the National Science Foundation, Global Lab was the first full-year, online middle- and high-school science course. Piloted in over 200 schools from 30 countries, the curriculum engaged students in a global community of practice in which they conducted hands-on environmental monitoring and data collection for air, soil, and water quality (Berenfeld & Bannasch, 1996).

Global Lab utilized remote hosting (a precursor of today's cloud computing) and rudimentary social networking (a precursor of Facebook and Twitter) to pioneer a new pedagogical strategy that we call *telecollaborative inquiry* (Berenfeld, 1994). The *tele-* of telecollaborative inquiry advances the collaborative work of single classrooms with entirely new learning capabilities and outcomes. It enables learning across distances in geographically distributed groups. It also permits inquiries that are not only collaborative, but also synchronized.

Global Lab was widely acclaimed for its innovations, but at the time, few teachers were prepared to deliver telecollaborative inquiries and the networking technologies were nascent (Berenfeld et al, 2010). In 2008, with the advent of Web 2.0 technologies and growing needs to teach STEM content and process skills, the project was successfully relaunched in 150 urban and rural schools across eight time zones in The Russian Federation as a test-bed and is now being scaled and adapted for worldwide participation (www.globallab.ru/). Its developers are now preparing Global Lab 3.0 to facilitate the adoption of telecollaborative inquiry into mainstream classroom instruction.

Wired and Connected

Global Lab demonstrates that the power of telecollaborative inquiry lies in "the teachable moments" resulting from its inherent, almost Hegelian dialectic of uniformity and diversity.

Uniformity: Same-aged students use the same curriculum, tools, and procedures to gather data on their local environments. They follow the same strict protocols to make measurements and observations on the very same day. Thus, each class's dataset is directly comparable to all others.

When Linda Maston's eight-grade Global Lab class in San Antonio, Texas, had the opportunity to investigate the air quality in its classroom, her students seized it (Berenfeld 1993, Yazijian, 1998). Unable to leave their inner-city school to conduct investigations outdoors, her students had to use their classroom for their study site. The classroom, however, lacked windows, prompting students to question its air quality.

Using the project's tools and instructional materials, Maston's students measured sulfur dioxide, ozone, and carbon monoxide levels in their classroom and found them to be low. But when they measured CO_2 levels, they found levels as high as 2100 parts per million. After team of students had obtained outdoor CO_2 readings of 350 ppm and the class compared its findings with those from other Global Lab classes worldwide, students became alarmed. Belonging to an international community, they asked their peers for feedback. "*What are some of the CO2 levels that people are getting inside their various classrooms? Ours are extremely high*." A class from Aiken, South Carolina, replied that they too had high readings, but not in spaces that opened to the outdoors.

Maston's students decided to assess CO_2 levels throughout the school and found consistently high readings everywhere but in shops with garage doors opening to the outside. They presented their findings to the school board, which dispatched four environmental control officers to investigate. Maston reported what happened: *"They [the officers] first went into the counseling office where the counselors and teachers told them about what was going on. They were not impressed, so they were brought to our classroom. As soon as we pulled out the data and the graphs showing the patterns that we had found, they suddenly started to take notes."*

The officers then took readings with their professional equipment. Maston continued: *"The moment of glory came when the officers got exactly the same reading as we got!"* As a result of her students' Global Lab investigations, the school's ventilation system was repaired.

"The CO2 study was (the students') pride and joy. They were just so pleased and proud of themselves that they had managed to do what nobody else had been able to accomplish in 17 years. To have their data taken seriously by adults in general, and the district in particular, was just awesome for them. They are so used to failure that it's hard to convince them sometimes that they are doing good work."

Maston's students performed real-world scientific research with their Global Lab peers and their findings made a difference in their lives.

Dissimilarities: When a Global Lab class compares its data to those of all other classes, a global snapshot emerges, stimulating curiosity and opportunities for teachable moments. The simple question of why different locations have different soil temperatures can drive a variety of inquiries. Students can visualize their data and incorporate additional community-wide metadata such as latitude, elevation, mean air temperature, and geographical coordinates to discover potential causes for the discrepancies. They might find a correlation between soil temperature and latitude, for example, and try to determine causality, revealing such factors as the angle of the sun. Or they may wonder about an outlier along a certain latitude range and discover that a nearby mountain range at that location affects climate, which impacts soil temperature.

Any individual class can make the same measurements, but what separates Global Lab from standard curricula is its students place their findings into regional and global contexts. Students across the Global Lab community have differing cultures, perspectives, and experiences. Similarly, their local environments all differ geographically, geologically, climatically, biologically, and historically. These differences are reflected in the project's datasets. These dissimilarities create a dynamic learning environment that produces the motivation and research questions for inquiries.

When engaged in telecollaborative inquiries, students learn that when partnering with peers around the world, they must work responsibly not just for good grades but also for each other. After finding errors in the data submitted by other Global Lab classes, students at a Moscow high school sent the following message to the community: "*It is natural for every scientist to make mistakes. But the low accuracy of the data may lead to wrong conclusions. In science, this problem is one of the most important. In our scientific community, we have to overcome it too…We invite everyone who has any idea on improving the accuracy of our work to communicate with us*." Few other teaching approaches so encourage students to demand accountability of themselves.

A single classroom collecting data will have difficulties in revealing trends and patterns. Measuring soil temperature at different depths, for example, will not offer much meaning for students. Students can measure soil temperatures over the school year, but when graphed, this data will indicate only that temperatures generally conform to local temperature changes. By itself, a single dataset is not necessarily thought provoking and offers limited scientific and pedagogical value.

With telecollaborative inquiries, students learn about causalities and correlations as they explore the patterns and trends in their data. The traditional "compare and contrast" mode of analysis offers new meaning and relevance: Why are my data different from theirs? Is this finding a discovery or an anomaly or mistake? How do we know that our data can be compared to everyone else's? Did we use the same procedures? Was my thermometer at the same height above the ground as theirs? At the same distance from our schools building? Was it in shade? Why does all of this matter? Students engage in real science in a ways that are nearly impossible with individual classroom inquiries. Again, this power lies in uniformity and dissimilarities (Berenfeld, 1994).

Designing Telecollaborative Curriculum

The keys to building effective telecollaborative inquiry curricula are:

- engaging students in a community of peers;
- providing community-shared goals and the scaffolding to meet them;
- and ensuring students are invested in the outcomes.

Global Lab seeks to meet these objectives by structuring the school year into three progressive stages.

The first, *Meeting Your Global Lab Community*, is dedicated to community building. At the start of the Global Lab year, each class introduces itself, its school, its community, and its region with multimedia presentations they create using tools on the project's web site. Presentations can include text, audio, image, and video data, and students are encouraged to personalize them with their interests and other information. They can deploy a tool developed for the project called Annotator to annotate their images with text. When other classes place the cursor over an individual student in a class photo, for example, that student's name and messaging automatically appears. Once submitted, all presentations are easily accessible by all other classes.

Each class also submits basic metadata about its location, such as its geographical coordinates, elevation, and mean air temperature, into a database designed for easy data extractions, comparisons, and visualizations. Using the database's sophisticated search engine, students can access data within ranges, enabling them to identify all Global Lab classes within ten degrees of a latitude or with certain levels of precipitation. They can compare their data to project-wide averages, to groups of schools, or to individual classes. In addition to this data mining, they can visualize numerical data using graphs, histograms, scatter plots, and pie charts.

When a class's information is uploaded, it automatically appears as a star at the appropriate location on a map of Russia (soon to be expanded to a world map when the project recruits schools internationally), showing the distribution and scope of Global Lab classes. Placing the cursor over any star automatically displays that class's metadata. Consequently, students soon understand that they have joined a community of peers.

The Global Lab curriculum also endows the community with purpose. The project breaks with traditional curriculum, which nearly always specifies what students study, by enabling students themselves to decide what they will spend the school year examining. This is their study site, an important Global Lab innovation that has been adopted by other projects (Berenfeld, 2010). The study site is a piece of land near the school whose environmental characteristics students will investigate over the curriculum. The project guides students in the selection, such as ensuring that they can access it within a class period, but the choice still belongs to them. The study site may simply be on school grounds but by enabling students to choose an object of study beyond their classrooms, the project invests them with a sense of relevance and ownership in their learning.

The second phase, *Building Investigative Skills*, provides the scaffolding to perform true science inquiries. Students first make qualitative observations and careful surveys of their study sites, and progress to quantitative measurements. Using the same protocols, tools, standards, and schedules, and with instructional guidance, they precisely gather data on the characteristics of their site's soil, air, and water in five content modules around primary Earth science topics—Understanding Weather & Climate, Forms in Nature, Atoms & Molecules at Work, The Sky Above, and How Does a Seed Know? Moreover, they research the site's history and uses as well as its scientific characteristics to make learning interdisciplinary.

As data is collected, Global Lab classes submit the information to the community database for comparison and analysis. Students gain the ability to place their local environments into regional and then global contexts, and are encouraged to raise questions and discuss their findings on project forums (this discourse is used for student assessments).

To prime students for telecollaborative inquiry, Global Lab promotes intra-classroom collaborations. Students work in small teams when they start investigating their study sites. Supported by the curriculum, teachers present the job descriptions for the various teams, permitting students to join groups of their choice. Students assume specific roles, which rotate over the school year. The teams, which include biologists, zoologists, cartographers, geologists, meteorologists, historians, and artists, take responsibility for certain tasks and data collections. The curriculum provides each team with its own scaffolding in students' Global Lab Journals. Teacher materials include suggestions for small group management, role rotations, and conflict resolution. Thus, students work together both face-to-face and virtually to build inter-personal skills, teamwork, and trust.

What makes Global Lab an authentic networked student science laboratory is not just shared curriculum, resources, procedures, and goals, but also synchronicity. Students make measurements concurrently, sometimes at the same time of day relative to time zones. This simultaneity makes data truly comparable as well as builds a sense of community. Synchronicity is exemplified by two highlights of the Global Lab year—the Fall and Spring Snapshots, which occur on the winter solstice and spring equinox. Patterned after the International Geophysical Year of 1957, the Snapshots are skill-building activities in which all schools make identical measurements of their study sites at the same hour on the same day. Students prepare for the Snapshots for a month with skill-building activities and once they have submitted their data, spend a month analyzing the various datasets.

With the community functional and students having acquired collaboration and basic investigative skills, classes enter the final stage of Global Lab—*Extended Investigations*. Drawing from their observations and measurements made during *Building Investigative Skills,* each class engages in open-ended telecollaborative investigations in a field of its choosing. The curriculum supports such topics as air and water quality, tracking pesticides, nitrate studies, butterfly migrations, lichens and other bioindicator plants, and UV and stratospheric ozone. Students submit ideas for investigations, frame their research questions, develop research plans, and search for collaborators. Anecdotal evidence indicates that learning to work in and with groups spurs students' willingness to telecollaborate (Means, 1998). Classes identify collaborators via forums or by searching the database for potential partners with appropriate environmental conditions. Throughout their inquiries, students are asked to peer review each other's work for accuracy and rigor.

This stage reduces the scaffolding as the project transforms from curriculum-directed to student-directed and curriculum supported. From selecting the study site onwards, students are given increasing latitude to make their own choices, building their stake in the project's outcomes. Like entrepreneurs in their own learning, they take initiatives and assume responsibility for their work. They perform basic science and learn of the need for cooperation, feeling valued as they grasp the importance of their data to the community. They discover that making measurements using standards and strict protocols is not arbitrary but essential for gathering meaningful data. They learn to separate facts from speculation, make sense of data, and understand the value of metadata. Thanks to the affordances of telecollaborative inquiry, they experience science as collaborative knowledge construction, a perspective seldom conveyed by textbooks and traditional instruction.

The Granularity of Daily Instruction

Telecollaborative inquiry is a new instructional paradigm for science classrooms. To facilitate its adoption by teachers, the current Global Lab, version 3.0, is pioneering an innovation that aligns the curriculum to the realities of daily instruction. Curriculum and content are delivered in granular units called Global Learning Units (GLUs™), each providing one to two class periods of investigations. All GLUs use a nine-stage structure that scaffolds and guides students through their work with a standardized web interface that branches off through the use of tabs and icons.

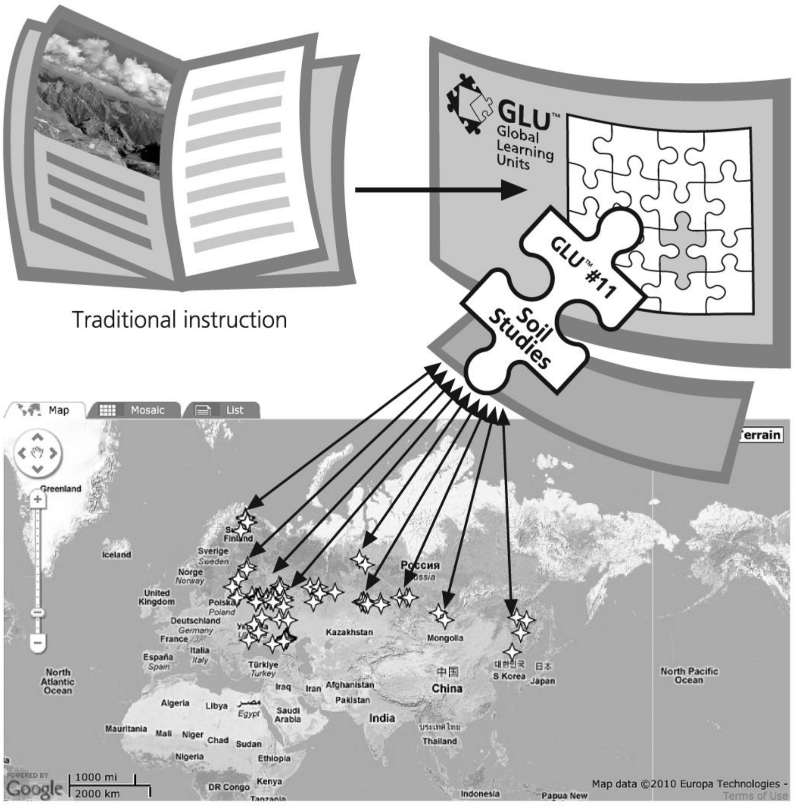


Figure 1. Global Lab converts conventional instruction into GLUs. When aligned with instructional objectives, GLUs cover the scope, sequence, and content of traditional curricula, building science content and process skills more effectively than single-classroom inquiries. The map shows schools in The Russian Federation that synchronously perform GLUs in telecollaborative investigations.

The common web interface furnishes the curriculum, collaborative tools, and resources for all GLU activities. In addition to providing a daily structure for teachers, GLUs offer an alternative strategy to either digital or hard-copy textbooks for delivering content. Each GLU includes the content, background, and vocabulary that students need to learn, thereby tightly aligning content with instruction. As a result, a GLU is a self-contained educational ecosystem that teachers can integrate into daily practice.

All GLUs feature common components. The first, “Introduction,” introduces students to the GLU’s topic and activities. The second, “Glossary,” provides the vocabulary and concepts that the GLU addresses. Students can add to it as needed. “Resources” allows students to access all relevant content with a click. Content, therefore, is a seamlessly merged with the curriculum and quickly available, not ensconced in a textbook or web site. The “Work with data” component guides students in data observations, collections, and analysis. “Our gallery” is where students post video, photos, artwork, metadata, and anything else about themselves and their work for other classes to access, enriching and personalizing investigations.

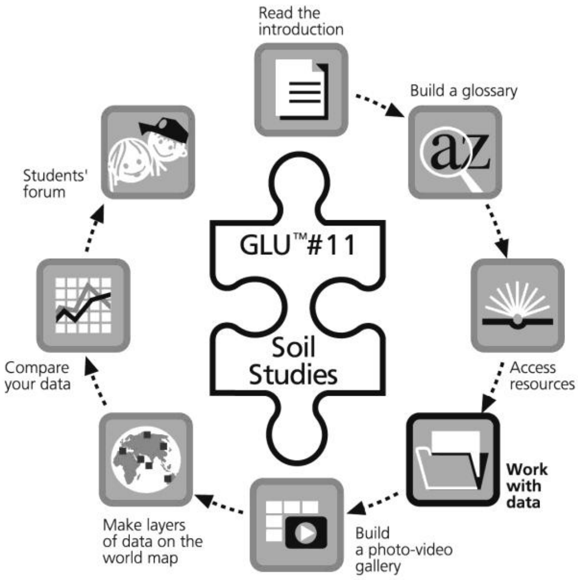


Figure 2. Every GLU is a standalone learning unit containing all necessary resources and capabilities to teach an instructional topic as telecollaborative inquiry. Each box in the figure represents one component of a GLU.

By clicking the “On the map” icon, students access a map of the project’s community to view the findings of other schools. In “Compare data,” they compare their findings with other classes using the Global Lab database and search engine. Students are encouraged to reflect on both the entire scope of data as well as data subsets, and to pursue further inquiries. They can examine metadata to identify a class or classes with which to collaborate. “Students’ forum” enables students to discuss their findings and explore why their data may be similar or dissimilar. Teachers obtain support from their peers by using the “Teachers’ forum” to exchange ideas and tips.

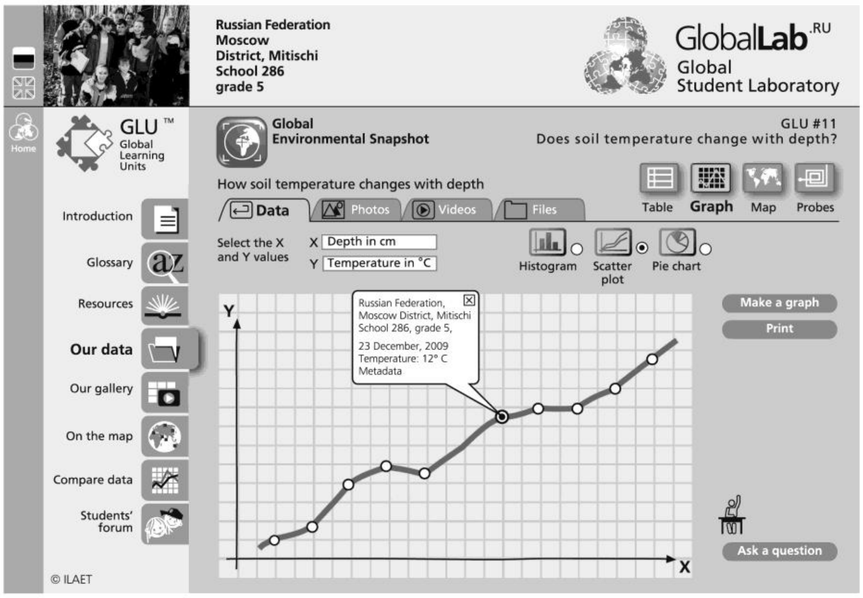


Figure 3. This is a GLU’s data-processing component. This example is an investigation into soil properties where students analyze how soil temperatures change with depth.

GLUs offer an effective strategy for converting traditional curriculum into telecollaborative curriculum. Global Lab weaves together content, curriculum, tools, and resources into a carefully sequenced structure to build knowledge and skills within a synchronized community. GLUs obviate the need for textbooks, regardless if they are print or digital. They are a user-friendly framework for providing everything that teachers need to guide students through true collaborative investigations and build their content mastery and skills.

Fulfilling the Promise of Connectivity

Telecollaborative inquiry can potentially justify the tremendous investments in wiring classrooms. Students may have digital access to each other via the Internet, but they are not connected intellectually—at least not in the classroom. Outside of school, children are digital natives who routinely use cutting-edge technologies to communicate and exchange information. When in classrooms, students are like nodes in a power grid that has been short-circuited; they generate ideas and interests, but have nowhere to go with them.

Global Lab realizes the power of connectivity by engaging students in intellectual pursuits that are bi-directional and fully interactive. With a telecollaborative inquiry project, the Internet does more than provide content and resources—it becomes the means for learning, fulfilling the promise of the wired classroom. Global Lab harnesses the latest communication technologies to not only render learning more meaningful, to not only make the Internet a richer source of information, but also to enhance teaching and learning.

Global Lab uses much more than standard broadband Internet access. It leverages today's networking advances that are ideal for telecollaborative inquiry. One example, of course, is social networking. Even in its first version and well before the term "social networking" reached popular culture, Global Lab encouraged spirited and thoughtful communications between students who were continents apart. The project never wanted students to just exchange numerical data with each other; it wanted them to share interests, ideas, questions—the rich intellectual discourse that drives collaborations and learning. Future versions of Global Lab may utilize more advanced communications such as live video, web conferencing, and networked telephony to make investigations even more vivid, dynamic, and interactive.

The strong social dimensions of telecollaborative projects, however, contrasts with common classroom practices. Teachers need a structured workplan each week and interactions among students must be controlled and often minimized. Most importantly, teachers focus on individual work and achievement (Resnick, 1987). Despite the project's pedagogical power, many teachers struggled to implement the first Global Lab as daily instruction and, instead, used it after school or to augment traditional curriculum (Means, 1998).

GLUs are a response for delivering many-to-many communications within a structure designed for classrooms. They present content and traditional curriculum in an integrated digital ecosystem that is built on the granularity of individual class periods. They offer a framework with which to adapt curriculum for telecollaborative inquiries, complete with teacher supports.

Another technological innovation that is prime for telecollaborative inquiries is cloud computing, a networking paradigm that companies around the globe are increasingly adopting. The original Global Lab was possible only because it could offer a remotely-hosted infrastructure available to all designated users, which, while not cloud computing, was its precursor. This infrastructure, however, is modest compared to the Web 2.0 technologies of today's Global Lab. Moving forward, the project will leverage cloud computing to deliver shared resources and communications—from content, curriculum, and applications to teacher training and student assessments—to thousands of schools.

Global Lab is a test-bed for telecollaborative inquiry. Its materials are being translated into English to again enable worldwide participation, and it will be refined, scaled, and evaluated. The project will continue to explore how social networking can help transform education, how cloud computing offers schools new capabilities and economies, and how conventional curriculum can become more effective pedagogy. Global Lab is a working laboratory for how educators can bring true science into classrooms to prepare children with the knowledge and skills needed for tomorrow's world.

References

Casner-Lotto, J. and Barrington, S. (2006). Are they really ready to work? Employers' perspectives on the basic knowledge and applied. Skills of new entrants to the 21st. Century U.S. Workforce. 2009 (May 17).

Berenfeld, B. (1994). Technology and the new model of science education: The Global Lab Experience. *Machine-Mediated Learning*, vol. 4. 203-227.

B. Berenfeld, B. & Bannasch, S. (1996). Global Lab: From Classroom Labs to Real-World Research Labs. *Microcomputer-Based Labs: Educational Research and Standards*. R. Tinker, Ed. Berlin: Springer Verlag. 247-257.

Berenfeld, B. (1999) The internet In our classrooms: teaching tomorrow's skills for tomorrow's world. *Human Development Network. Secondary Education Series. Science and Environment Education. Views from Developing Countries*. Washington, DC: World Bank. 215-235.

Resta, P. and Laferrière, T. (2007) Technology in support of collaborative learning. Educational Psychology Review, 19, 65-83.

Berenfeld, B., Lovyagin, S.N., Yazijian, H., Kovalevskaya, E., Mazhurina, K., Vekhter, B. (2010) *Global Lab: Harnessing Social Networking and Cloud Computing for K-12 Science Learning.* Proceedings of the Society for Information Technology & Teacher Education.

Berenfeld, B. (2010) *Leveraging Technology for a New Science Learning Paradigm*. Proceedings of the MIT Learning International Networks Consortium.

Berenfeld, B. (1993) "A moment of glory in San Antonio: A Global Lab story." *Hands-On,* 16(2), TERC, Cambridge, Massachusetts.

Yazijian, H. (1998) "The World's Biggest Science Class," *UNESCO Sources*, Volume 98, February 1998.

Means, B. (1998) Models and prospects for bridging technologically supported education reform to scale. *American Educational Research Association Annual Meeting*. San Diego, CA.

Resnick, L. B. (1987) *Learning in school and out.* Educational Researcher, 13(9), 13-54.

Constructionism through construal by computer

Meurig Beynon, *wmb@dcs.warwick.ac.uk*
Department of Computer Science, University of Warwick, UK

Antony Harfield, *ant@dcs.warwick.ac.uk*
Department of Computer Science, University of Warwick, UK

Abstract

Traditional computer *programming* is not well-aligned to the needs of constructionism. Orthodox programming principles are oriented towards prescribing processes that address clearly specified uses. Functional specification and optimised execution do not encourage interactive exploration and open-ended interpretation. We propose making *construals* by computer using *Empirical Modelling* principles as an alternative to conventional computer programming. The merits of this approach are discussed and illustrated using construals for Sudoku solving.

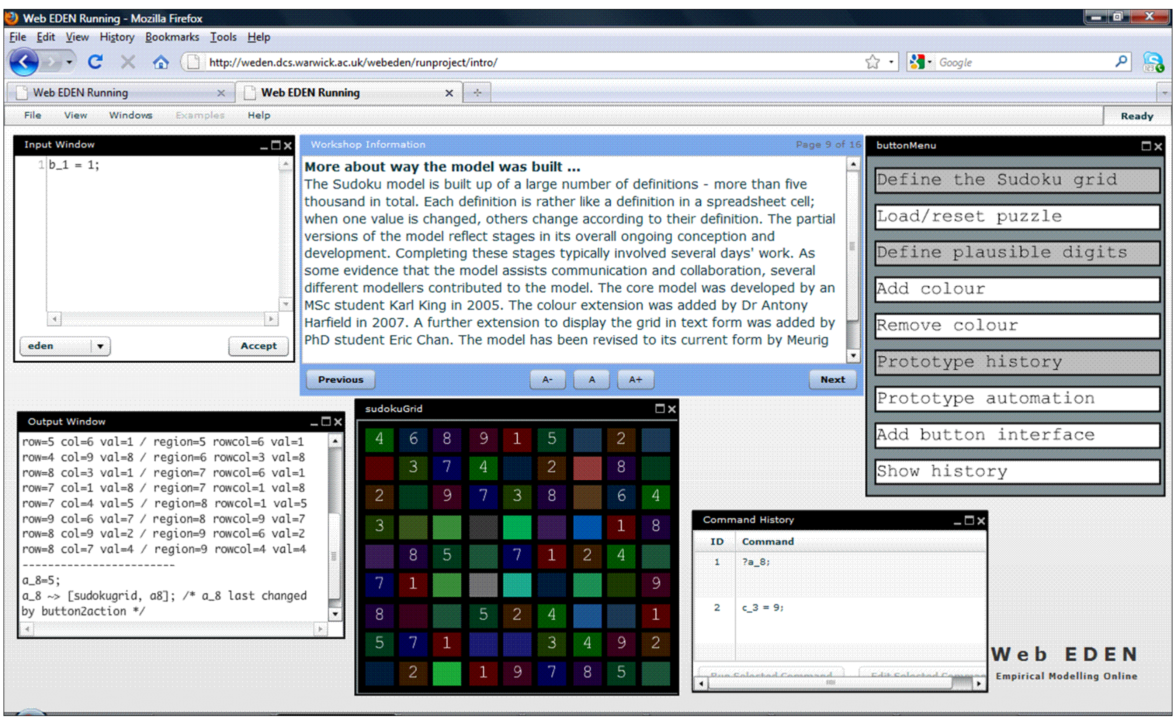


Figure 1. A screenshot depicting the online Sudoku solving construal

Our Sudoku solving construals are made up of definitions that express *dependencies* between *observables*. Many kinds of human agency can be expressed through modifying the current set of definitions. The construal serves as a shared artefact with which developers, teachers and pupils can all interact concurrently in essentially the same way, each according to their role and experience. Our preliminary experiments with schoolchildren highlight potential for rich and radically new kinds of learning experience and unprecedented scope for recording, monitoring and intervening in support of constructionist learning. Further empirical study is a vital next step.

Keywords

Constructionism, programming, learning technology, Empirical Modelling, construal, dependency

Constructionism and Computing

As Richard Noss (2008) observes in his *Open Letter to Logo and EuroLogo Communities*, Seymour Papert introduced **constructionism** as a name for "a pedagogy based on building and sharing physical, virtual and intellectual structures". This characterisation makes no reference to computers, but in practice it was the computer, and specifically the use of Logo programming, that launched the concept of constructionism. No educational movement has since contributed more to the cause of constructionism than the Logo community. Yet, as Noss also remarks, "the fundamental hope of Logo's creators and its later adherents remains largely unfulfilled".

The challenges for constructionism as a computer-based activity mirror those facing computer programming within computer science. The *Call for Papers* highlights the fact that "The developers of Logo and similar computational environments have ... encouraged learners to better understand the world and their place in it by building their own meaning-making models based on iterative, interactive exploration and testing of ideas and notions." It endorses a vision for constructionism in which, in the words of one conferee: "I don't see any hard edges between creating, sharing, consuming and learning. I want a system that allows people to shift effortlessly between doing these things."

As computing technology matures, and computing systems become ever more complex, so "constructionist" activities in this spirit become more relevant for software development. Agile development methodologies embrace the notion of "iterative, interactive exploration and testing of ideas and notions." They demand environments in which developers can play many different roles - building, learning, communicating - and "shift effortlessly between doing these things." But even if we look beyond programming paradigms and languages based on Logo, the vision for efficacious software engineering of this kind also "remains largely unfulfilled". Even within established computer science, the problem of giving computing support to constructionist principles is unresolved (Ben-Ari, 2001; Beynon and Harfield, 2007; Beynon, 2009).

We believe the underlying problems faced in educational technology and in complex systems development have a common root. They both relate to the difficulty of aligning computing programming with learning, as is essential both in the classroom and in the software house.

In this paper, we illustrate an alternative approach to developing software that we believe offers much better prospects for supporting constructionism. To reflect the radically different orientation of this approach, we conceive constructing software as 'making a construal' rather than 'developing a program'. A **construal** is a computer-based artefact similar in character to the structures conceived by Papert, but with essential qualities that are not apparent when we interpret it as a conventional "computer program". A construal admits myriad interactions that are not preconceived, for instance, and for which there may be no specified *a priori* use or interpretation. The interpretations it supports depend in general on the experience and the skill of the individual human agent who is interacting with it. On this account, they also depend critically on qualitative and experiential aspects of the construal – how its state is communicated and perceived, how it can be manipulated and how quickly it responds.

One reason why conventional programs - such as the Logo programs originally studied by Papert – fail to support constructionist learning as powerfully as we might like is that they are built with quite different objectives in mind, and according to quite inappropriate principles. For instance, in order to make practical progress, computer programmers must specify the interactions that their programs support and align these to specific purposes. To this end, the programmer abstracts automatable patterns of interaction and interpretation ('uses') from the environment. By contrast, a construal is a source of concrete open-ended experience that is not typically intended to be understood in isolation from its environment. In this respect it resembles a spreadsheet or a database in which the symbolic data stands in an intimate relation to meaningful – and current - entities in the external world. When separated from the external referent, or no longer current, such symbolic relationships lose their significance. Interactions with spreadsheets and databases thus directly address sense-making.

A construal for Sudoku solving

The construal we shall study relates to the process of human solving of Sudoku puzzles. Of its nature, a construal is an organic artefact that evolves over time as it is deployed by different interpreters. We build construals using Empirical Modelling (EM) principles that we have developed over many years. A full discussion of EM is beyond the scope of this short paper – interested readers can consult the EM website for more details. The principal construal used for illustrative purpose is available online, and other variants of this Sudoku construal can be downloaded from the EM archive.

The key elements in a construal are **observables**, which represent meaningful entities that have an identity and current status or value in the referent, and **dependencies**, which reflect perceived connections between observables similar to those that link cells in a spreadsheet. As the word ‘perceived’ suggests, the construal is to be understood in conjunction with a human agent acting within an environment that encloses or evokes an external referent. There is no clearly specified set of appropriate processes and states associated with a construal. The states of the construal are intended to evolve in intimate conjunction with the states of mind of its human interpreter – or more precisely with those of its human *interpreters*, since there is no absolute constraint on the interpretative role that human agents can adopt in interaction with its current state. Meaningfulness to the human interpreter is all that constrains the evolution of the construal, which lends an open-ended exploratory character to its construction. This does not rule out the possibility that patterns of interaction and interpretation emerge, some of which may be automated to realise program-like functionality.

The Sudoku construal – or to be more precise, *a state* of *a* Sudoku construal – is depicted in Figure 1. Because of the highly interactive nature of construals, it would be helpful at this point for a reader unfamiliar with EM to invoke this construal via the online ‘General Introduction to the Script’ at http://www.dcs.warwick.ac.uk/~wmb/sudokuExperience/workshops/. As explained in that introduction, the underlying structure of the construal is a moderately large set of definitions (some five thousand in all), each of which specifies the value of a different observable either explicitly, or by a formula in terms of other observables. The size of the set of observables is largely due to the fact that each observable associated with a particular cell of the Sudoku grid has a counterpart in each of the other 80 cells. Examples of observables include:

```
d_3 = 7;            ## the contents of the cell D3 in the 3rd row and 4th column of the grid
d3_fixed = 1;       ## the status of the value in this cell – given in the puzzle, so fixed
d3 is mkstr(d_3);   ##  the string that is displayed to indicate the value in the cell
## … the x and y coordinates of the cell D3 in the screen display:
D3_X1 is grid_startx + column(3.0) + (3.0 * spacer_x) + 2.0;
D3_Y1 is grid_starty + row(2.0) + (2.0 * spacer_y);
## … the foreground and background colours of the cell D3:
D3_fgcolour is d3_fixed ? SD_fixed_fgcolour : SD_fgcolour;
D3_bgcolour is colourclue(possibledigits2binary(possdigit34),colourvalues);
```

The construal is built using the EDEN interpreter, depicted in Figure 1 in its online variant. Definitions are first entered through the Input Window at the top left, either individually, or through file inclusion. The current values and definitions of observables can then be queried and displayed in the Output Window at the bottom left. Initially, before any definitions have been entered, the interpreter affords only these two windows; other components of the display in Figure 1, such as the Sudoku grid and the ButtonMenu panel at the top right are themselves specified by sets of definitions via the Input Window. In Figure 1, the grid and button menu supply the interface for a novice user. They allow digits to be entered into the grid directly, and allow pre-packaged sets of definitions to be introduced to the model at the click of a button.

The perspective on a construal that best matches Papert's constructionist vision is that afforded by open interaction via the Input Window. Through this window, any of the observables associated with cell D3 can be freely modified. We can determine whether or not the value in cell D3 is to be deemed fixed, change the digit in the cell whether or not it is deemed to be fixed, relocate the cell D3 on the screen, modify the string that displays the value of the digit 7 etc.

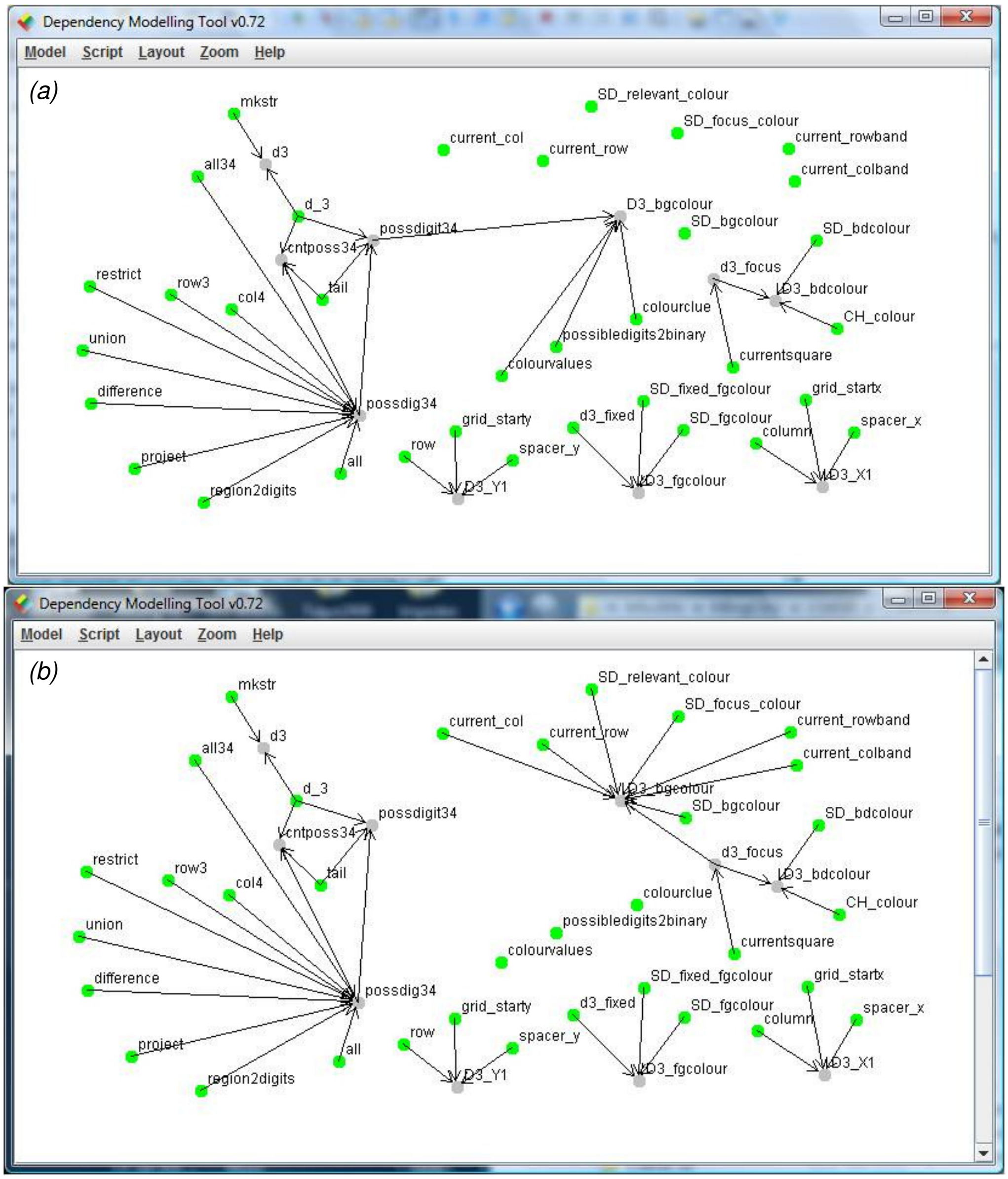


Figure 2. Observables and dependencies associated with cell 34 in a Sudoku grid

Figure 2(a) uses Allan Wong's *Dependency Modelling Tool* to give a comprehensive overview of the net of observables and dependencies associated with the cell D3 in Figure 1. The dependencies in this net are acyclic, and the green nodes correspond to observables with explicit definitions. From the diagram, we can infer that the attributes of cells are independent of their location on the screen. We can also see that the background colour of cell D3 depends on the observable possdig34, which records the set of *plausible* digits for a cell, given its status and that of cells in the same region, row or column. This dependency is the focus of a later section.

The dependency net may appear to be static and structural in character, but in fact it is dynamic and fluid. At first sight, it seems obvious that the dimensions of a cell should be independent of its location, but – in fact – it might assist a partially-sighted person if the grid could be moved around so that the cell at the centre of the screen was enlarged. Of course, what changes can be made to the dependencies are constrained if the construal is to represent a Sudoku grid. This is no more than what is expected of structure that is negotiated through experience and established by convention. In this respect, a construal provides a representation that not only favours a constructionist stance, but is in the broader sense *constructivist* in spirit (Latour, 2006).

Simple redefinitions play a vital – though sometimes hidden – role in the Sudoku construal. For example, when a solver focuses on a specific cell (say E2) and enters a digit (say 6) from the keyboard, they in effect instruct automated agents to redefine the current cell, thereby defining an observable (e2_focus) to indicate that E2 is currently selected, and to make an appropriate redefinition (e_2 = 6). More elaborate sets of redefinitions are associated with a shift in perspective on the solving task, such as is involved in switching in and out of the 'colour Sudoku' mode (cf. Figures 1 and 3). The complex reconfiguration of dependencies invoked when the 'Remove colour' button is pressed can be seen by contrasting 2(a) with 2(b). Setting up these different modes and patterns of agency for interaction involves the iteration and testing of ideas characteristic of constructionism. Once created, they can then be recorded and replayed.

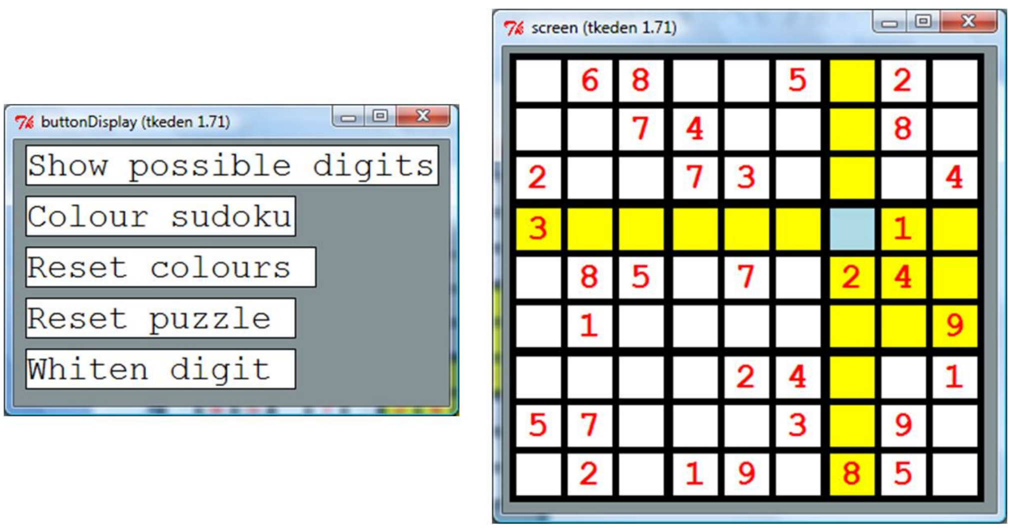


Figure 3. The initial state of the Sudoku solving environment for the ACE sessions

Blending learning, teaching and development

At the heart of constructionist education is the idea of shifting effortlessly between the work of the learner, the teacher and the developer. The distinction between these roles is very stark in conventional programming. In construals, by contrast, the net of observables and dependencies serves as a playground where many agents can act, potentially even concurrently. What distinguishes the roles of agents is their expertise and level of privilege where interpreting and modifying definitions is concerned.

The potential for blending roles that construals afford has been illustrated informally in educational activities we have carried out at the University of Warwick using our Sudoku-solving construals. The first of these was in connection with two short visits by local schoolchildren under the auspices of the "Aiming for a College Education (ACE)" programme in January and February 2008 (see Figures 3 and 4). The second, in July 2008, was a week-long workshop entitled “The Sudoku Experience” for pupils on the UK Young Gifted and Talented (YGT) programme (see Figure 1). Further feedback on this workshop has been given by Daria Antonova, a high-school student affiliated to the Nokia Toijala Center scheme in Finland.

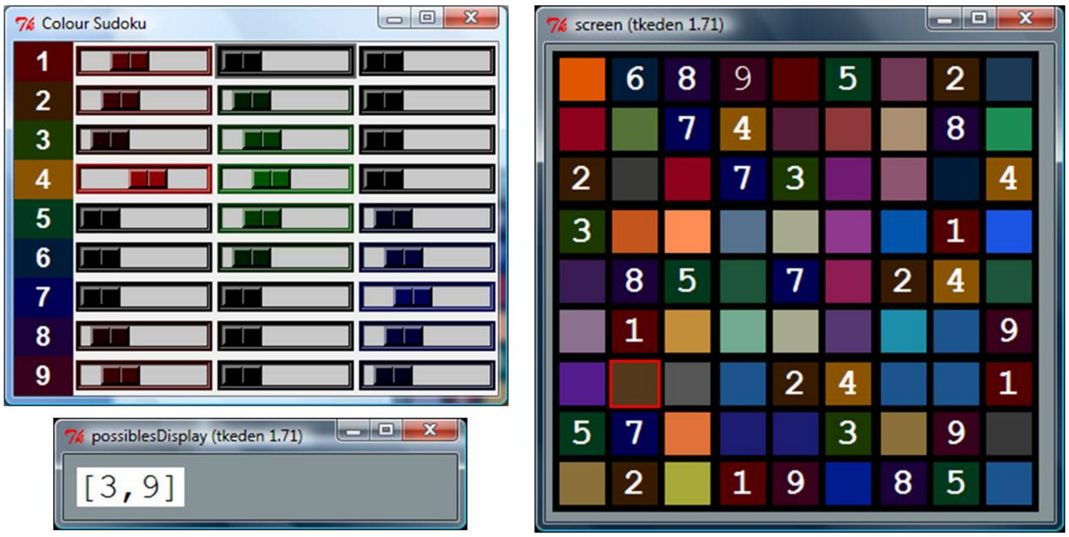


Figure 4. The state of the Sudoku solving environment after colour has been introduced

The variant of the Sudoku construal used in the ACE activity highlights some of the ways in which the perspective of developers, teachers and learners can be blended. The educational objective was to expose pupils to abstract reasoning and problem-solving in a way they might find entertaining. The basic construal on which the activity was based was initially developed by an MSc student Karl King in 2006 (cf. Figure 3), and subsequently elaborated to include a cell colouring by Harfield in 2007 (cf. Figure 4). These were combined into a single artefact by Beynon for the ACE workshop, when the simple button interface in Figure 3 was added. This process of re-use and adaptation is characteristic of our construals and illustrates the scope for sharing and communication in development that EM affords.

The adaptations of King’s and Harfield’s construals for the ACE activity are typical of those that a teacher might make. They introduce interfaces that disclose observables and allow them to be manipulated in appropriate ways. The basic affordances in the button menu in Figure 3 make it possible to inspect the plausible digits for the currently selected cell and to reset the puzzle. In a routine solving process, the solver looks for cells that admit *just one* plausible digit, or instances of rows, columns or regions in which there is only one cell in which a specific digit can be placed. We introduced these abstract rules to the pupils before they used the construal.

Several skills are engaged in applying these simple rules. The yellow highlighting of cells in Figure 3 helps to maintain focus on the digits that lie in the same row, column and region. A teacher who wished to assess a pupil’s skill in identifying such relevant cells might remove these highlights by redefining a single observable – SD_relevant_colour. Once pupils are skilful in such identification, colour can be exploited in another way, as illustrated in Figure 4.

The basic principle behind “colour Sudoku” is that a colour can be associated with each empty cell so as to reflect the ‘plausible digits’ for the cell. For the cell B7 in Figure 4, for example, the set of plausibles is [3,9] because 3 and 9 are the only digits not found in its row, column or region. Each empty cell can then be coloured according to its set of plausibles. One way of doing this is to assign a unique colour to each digit and then colour the empty cell as a mix of the colours of its plausibles. In the Sudoku construal the resulting colour for a cell is obtained by adding together the RGB values of the assigned colours for each of the possible digits.

The two basic rules for Sudoku solving identified above have interpretations in colour Sudoku. The colour of a cell having only one plausible digit (such as E1 in Figure 4) has the same colour as that digit. The fact that there is only one location in which a particular digit can be placed within a row, column or region can be disclosed by modifying the colour associated with that digit and observing which cells are then affected. The interface in the left panel of Figure 4 enables the colour associated with any specific digit to be modified. An application of this rule is then illustrated in Figure 4, where the R component of the colour associated with the digit 4 is being enhanced, and the impact on cells of the grid observed. In this case, the cell A1 is the only cell in the top row that changes colour.

Extensions to King's original construal of the above kind are of interest to the teacher in several ways. They can be used to enrich the learning experience for the pupil, or to probe the nature of a pupil's difficulties. They can also be used in gathering insight into Sudoku solving prior to framing activities for pupils. When solving puzzles without computer support, a solver may survey the plausible digits for cells and commit them to memory, or record the plausible digits as a list in each cell. To some degree, efficient solution of puzzles appears to rely on good fortune in identifying cells to which one of the basic rules applies. Studying colour Sudoku helps to expose issues that relate to perception, cognition and memory in the solving task.

Because of the brief and cursory nature of their visit, we had no opportunity at the time to study the pupil's reactions to the ACE sessions in a formal way. A feature of the Sudoku construals is that the entire history of interactions associated with each individual pupil is recorded as a sequence of redefinitions of observables, which can then be replayed. Figures 3 and 4 are screenshots from just such a replayed sequence of interactions on the part of one pupil. What is more, because each state constructed from the history is retrieved as a set of definitions, it serves as an interactive environment in which the analyst can also interact. This is also illustrated in Figure 4 – the display of plausible digits at the bottom left was not in fact invoked by the pupil who created the history, but was added when the history was replayed.

From our preliminary studies of these histories, and our informal observation at the time, it was apparent that the range of pupil reactions was broad. For some pupils familiar with Sudoku, the elementary puzzle in Figure 1 proved too simple. Others – who perhaps lacked experience and motivation for puzzle solving – did not understand the task, but seem to have completed the grid in a random manner so as not to lose face. One pupil chose not to use the colour Sudoku interface, maintaining that they preferred the presentation in Figure 3 to that in Figure 4. A virtue of the openness of the construal is that it enabled us to adapt to special circumstances. For instance, our button interface made no provision for changing the underlying Sudoku puzzle, but we could show the more advanced pupils how to load different puzzles via the Input Window.

The *Sudoku Experience* workshop

The *Sudoku Experience* workshop consists of a number of activities to be undertaken inside the online Web Eden environment. Each activity starts with the Sudoku construal loaded in a specific state together with a guidebook for the student to follow or work through (see Figure 1). For some of the activities the guidebook resembles a tutorial (e.g. in early activities the student is taught how to use the environment). For other activities the guidebook is a guided exploration of the construal (e.g. to give insight into essential elements of the construal). In other contexts the guidebook proposes open-ended creative tasks for the student to undertake (e.g. building an extension to the construal by modifying dependencies). In this way, the environment is utilised for both 'instructionist' and 'constructivist' learning. In order for a student to progress to a deeper level of learning in these workshops it seems to be necessary for exploratory activities to be preceded by activities of a more closed and tightly prescribed nature.

The workshop had three core elements: playing, exploring, building. Students began on day one by playing Sudoku. This could have been on paper or using the Sudoku construal. In its basic state, the Sudoku construal appears to be nothing more than an interface for playing Sudoku (much the same as the many Sudoku programs that are freely available). However, the vanilla construal provides an interface for adding layers from which a student can begin to explore ways to solve a puzzle. For example, a student might choose to show the plausible digits for a particular cell as shown in Figure 4. The set of plausible digits for a cell is an observable in the construal (cf. possdig34 in Figure 2) that is defined by a dependency. This is one of many dependencies that are already given to the students in the Sudoku construal for them to explore.

An important feature of a construal is that it is always live. There is no separation between a 'build' mode and a 'play' mode. A construal is open to exploration and redefinition at the same time as a student is using or playing with it. In the Sudoku construal, a student might start playing by entering some digits into the puzzle, but then explore what this means in terms of the underlying dependencies, or try to add new dependencies to derive new insight into the puzzle. One of the activities that involved adding colour to assist solving serves as a good example.

By midway through the workshops, the students were familiar with solving Sudoku and able to identify and write dependencies that describe simple rules for solving Sudoku. They had also been introduced to the colour version of the Sudoku construal. At this point we gave the students an opportunity to construct their own version of colour Sudoku.

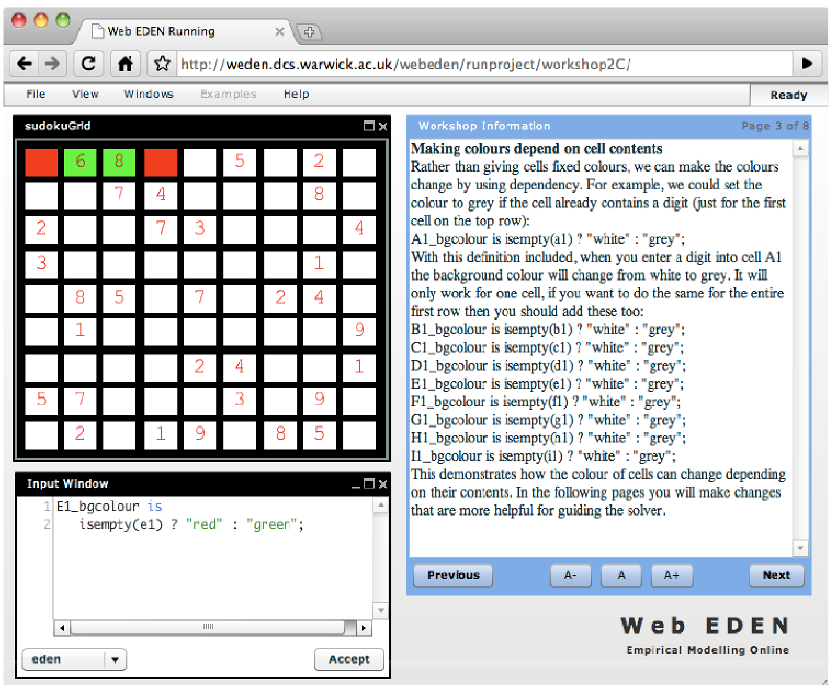


Figure 5. The Sudoku construal as used for the colour creation activity

The colour Sudoku activity acquaints students with the way in which the visual colour clues in Figure 4 are specified, and equips them to explore alternative ways in which these could have been specified. It begins by introducing the observables for defining the background colour of a cell. Students then start by experimenting with the dependencies that define the colour based on the value in the cell, as shown in Figure 5. In previous activities, students had already explored the observables for determining the possible digits of a cell. Using this knowledge we set them the task of creating dependencies that colour the cells in a similar manner to the colour Sudoku game they played at the beginning of the workshop.

The colour Sudoku activity illustrates how the Web Eden environment supports two key aspects of constructionism: helping students to achieve new knowledge and skills through disciplined guidance, and offering students opportunities to actively explore and experiment in order to develop knowledge based on their own experiences.

In the colour Sudoku activity, before attempting to 'actively explore and experiment' a student must be equipped with some knowledge of how to change the colour of cells and to access the plausible digits of a cell. The information window on the right hand side of Figure 5 provides disciplined guidance to colour specification in the colour Sudoku construal. It functions as a guidebook to the construal in much the way that a tourist guidebook assists a foreign visitor, pointing out places of interest, things to try, and essential facts that you cannot get by without.

With the guidebook as a reference for preconceived interactions that might be useful, the student can explore the construal and experiment by *changing* observables or dependencies. Such modifications might be achieved through a graphical interface (provided by the teacher or developer) or by entering definitions into the Input Window. In Figure 5, a new definition for the background colour of cell E1 is entered via the Input Window. In the colour Sudoku activity, the student is encouraged to experiment by setting cells to different colours in this way. An alternative colour scheme might associate the brightest colour with individual digits, and hence represent cells with many plausible digits by dark colours, for instance. Many students were able to progress to derive definitions for the colour of an individual cell based on plausible digits and thereby eventually build up a specification for the full colour grid, as shown in Figure 6.

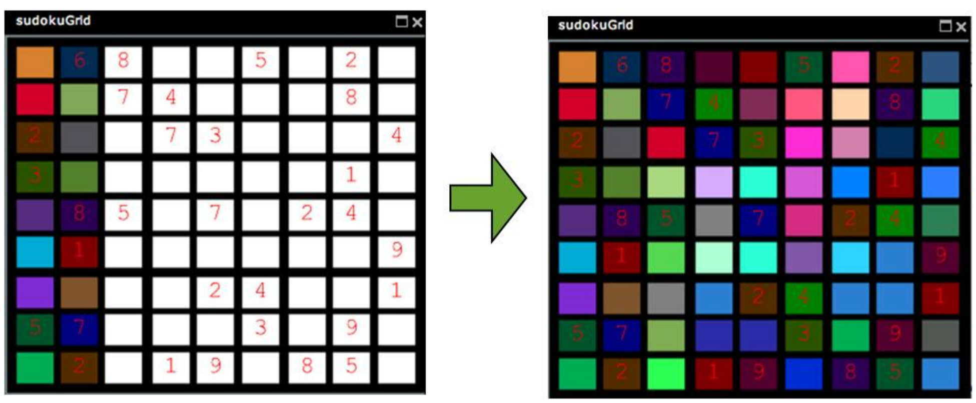


Figure 6. The Sudoku construal as used for the colour creation activity

Some feedback and reflection

Positive feedback from students during the workshop indicated that there is potential in the approach. Comments such as "I did it! It works!" suggested that students enjoyed the activities. Several appreciated the power of the metaphor of *making and following a guided walk* we had invoked. One student expressed what he liked about the activities: "It was amazing to see what we have actually done to the sudoku board and it was good that you said we could 'wonder (sic) off the path' a bit, e.g. changing colours and numbers, which was good fun."

Some of the most encouraging comments were directly related to the exploratory nature of the environment: "A very good idea to have a 'path' but it was flexible." Others saw the value in building up towards a larger goal: "A good building progress from each of the tasks, so at the end you can put all of them together." Students found experimenting valuable, even when it was the smallest change: "Having lots of stuff to do in every workshop, even if it looks easy, you still feel a small achievement when you actually change a square to blue or make the number 7 green." Students were also willing to try things out for themselves and take their learning into their own hands: "Lots and lots of colours recognized, i decided to have a little play and started putting random colours in like magenta, turquoise etc. and was really good fun making it multicoloured". The environment enabled some students to go beyond the set tasks: "I know that the sudoku grid is supposed to give blanks when 0 is entered into a square but this does not allow other programs such as the addition program to work properly. A solution to this would be to set a key such as 'c' to a blank and tell users to press 'c' or clear to empty a box."

Many of the critical comments related to literacy demands. One criticism was that there was too much material to read: "There was a lot of writing to read during the first two tasks 18 pages and 14 pages, maybe other people feel differently, but the majority of the task was reading." And whilst some students remarked that: "it was well written and easy to understand", others struggled: "I had difficulties to knowing how to do things, as I don't think it was explain very well. In the introduction I got confused straight away but then when I went onto workshop 2 I worked out what to do. I think it needs to be made clearer how to do things." Another said: "I really didn't understand a lot of it - some of the basic stuff made sense and I think I got some of the stuff about the colours, but a lot of it went over my head. A bit of getting to know the basics first would have helped, I think if I knew more about programming, I would have enjoyed it more."

The wide range of notations in the environment confused one student: "I have run into more problems. Which (thing) should I be using; eden or scout". Another was mystified by aspects of the environment: "I also found the output box confusing and don't quite know why we need it." These issues point to a need to refine the environment for educational use. Although restricting what tools the learner has available may be undesirable from a constructionist viewpoint, it may sometimes be better to hide parts of the environment that are unlikely to be used.

Overall, the breadth of the issues addressed in the *Sudoku Experience* workshop and the open-ended interaction it provokes go beyond blending developing, teaching and learning, blurring the very boundaries between what is being taught, learnt and constructed. The student feedback points to the difficulty some face in adapting to the fuzzily-defined objectives and rich and messy interactions that discovery in a constructionist spirit can entail. Looking to the future, it is vital to distinguish construal-by-computer from the sharply-defined goals and neat rationality of programming. In these respects, Antonova's independent verdict on the workshop activities is encouraging: "They turned out to be pretty interesting and dont really require programming skills or previous knowledge of programming language, just some logic. I had to think quite a while about some of exercises to find answers but after you find them, exercises don't seem hard."

References

Ben Ari, M. (2001) Constructivism in Computer Science Education, Journal of Computers in Mathematics and Science Teaching 20(1), 45-73

Beynon, M. (2009) *Constructivist Computer Science Education Reconstructed*. HEA-ICS ITALICS e-Journal, Volume 8 Issue 2, June 2009, 73-90

Beynon, M. and Harfield, A. (2007) *Lifelong Learning, Empirical Modelling and the Promises of Constructivism*. Journal of Computers, Volume 2, Issue 3, May 2007, 43-55.

Latour, B. (2006) The Promises of Constructivism, In Ihde, D. (ed.) *Chasing Technoscience: Matrix of Materiality*, 2006, 27-46

Noss, R. *Open Letter to Logo and EuroLogo Communities*, at the url: http://www.aup.edu/news/downloads/constructionism2010_Noss.pdf (accessed 28/03/10)

The EM website at http://www.dcs.warwick.ac.uk/modelling/ (accessed 28/03/10)

The EM archive at http://empublic.dcs.warwick.ac.uk/projects/ (accessed 28/03/10)

The Web Eden interpreter at http://go.warwick.ac.uk/webeden/ (accessed 28/03/10)

Constructionist learning by computing for construal

Meurig Beynon and Antony Harfield, *wmb@dcs.warwick.ac.uk*
Dept of Computer Science, University of Warwick, UK and Tessella plc, Abingdon, Oxford, UK

Introductory description and overall goals

A *construal* is a physical object that supports sense-making through exploratory interaction and interpretation. We introduce a prototype environment for creating construals by computer in which developers, teachers and learners all interact in essentially the same way, blending their activities in a way that is exceptionally well-aligned to the constructionist ideal. We shall illustrate the use of such an environment with reference to construing human solving of Sudoku puzzles.

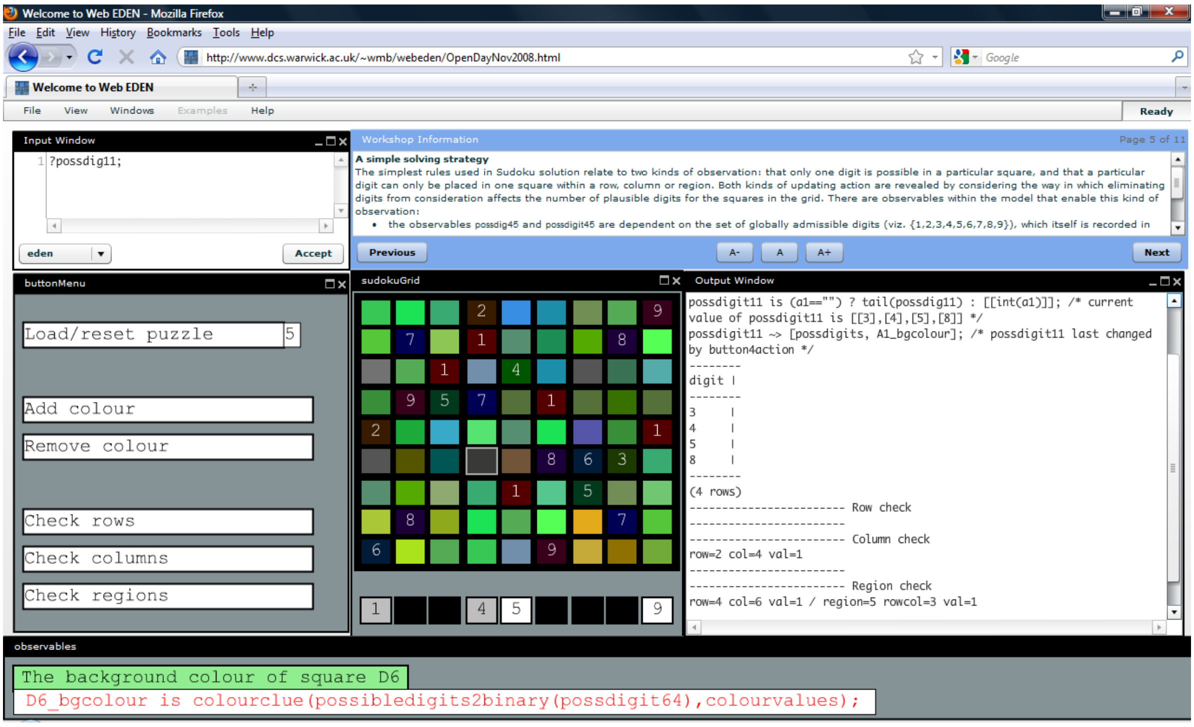


Figure 1. The Sudoku Experience – construing human solving of Sudoku puzzles

Method

The principles of computing for construal will be introduced through a series of demonstrations and exercises designed to give experience of acting and collaborating in the roles of developers, teachers and learners. A key feature of the tools we shall exploit is model-building with dependency such as is represented in spreadsheets and dynamic geometry environments.

Expected outcomes

Attendees will gain practical experience of computing for construal, and of the potential benefits for learning in a constructionist idiom. We shall highlight the unusual potential for new kinds of empirical study and evaluation that computing for construal affords, and hope to gain critical and constructive feedback from educational experts to guide future design and development.

Keywords

Constructionism, construal, spreadsheets, dynamic geometry, Empirical Modelling

"Matsiko": Rwandan children doing curiosities investigation with their laptops

Silvia Kist, *silvia@laptop.org*
One Laptop per Child Association

Juliano Bittencourt, *juliano@laptop.org*
One Laptop per Child Association

David Cavallo, *cavallo@laptop.org*
One Laptop per Child Association

Abstract

This paper's goal is to explore one of One Laptop per Child micro-level learning initiatives being developed in Rwanda that is part of a larger strategy for macro-level change in the educational system. This strategy is based on the framework developed by Cavallo(2004) which suggests models of growth from small progressive pedagogical initiatives to large scale computers in education programs through the development of exemplars, models and symbolic expressions (language). Such models of growth were designed in order to overcome the school system assimilation mechanisms through changes in the learning culture and to spread innovative educational practices in laptops saturated communities.

The model of growth is discussed via the implementation of the Matsiko, a project-based practice that explores the act of making questions to develop student's critical thinking and scientific inquiry through the investigation of their own curiosities about their world. The main goal of the activity is to provide children with personally powerful learning experiences by formulation of questions and "construction" of their answers. By instantiating such types of activities we hope to provide teachers with **exemplars** and **models** of learner-centered and constructionist approach they can relate to.

By providing exemplars of innovative practices, we show what is possible to do, what might be accomplished as we try to set new cultural expectations about what children can achieve and learn when given the right opportunities. It does not mean to create a model to be replicated. Instead, it means to introduce powerful ideas about learning, taking advantage of local culture, in a way to spread new educational practices, showing new possibilities for instantiating dynamic learning environments, illustrating with our practice in Rwandan schools. We also explain how this specific action aligns with a vision of the laptop's use in education.

In this paper we will talk about the principles behind Matsiko, look through its roots in Fagunde's (1999) learning project methodology and discuss the adaptations made for the local context. It also describes Matsiko implementation in schools of Kigali-Rwanda, justifying how it makes a common cause with the country's vision for economic development.

Keywords

Curiosity, One Laptop per Child, one to one, Learning Project, Constructionism, Technological Fluency

Introduction

Matsiko means **Curious** and **Curiosity** in Kinyarwanda, the mother tongue spoken in Rwanda. When we talk about Education, *Matsiko* is a meaningful word. Children in very early ages show curiosity and interest to learn about the world.

Curiosity for Piaget “plays a part in the search for coherence and organisation. It is a motive force in the need to order reality.” He also viewed curiosity as a product of cognitive disequilibrium caused child’s attempt to assimilate new information into existing cognitive structures (Loewenstein, 1994). When such structures aren’t able to assimilate new information that disturbs their system, a gap (*lacunae*) is created which impels the system to look for equilibrium, therefore creating an internal motivation for the children look for the answers that satisfy their curiosity/lacunae.

But when children enter school, they start suffering a “curiosity castration," an expression created by Paulo Freire. Freire (1985) believes that the question’s repression is a dimension of a larger human's repression, of his expressiveness in his relations in the world and with the world. The school, traditionally authoritarian, refuses the children’s question or creates bureaucracy to the act of asking. The school pours out answers (content) that were not asked for and does not allow questions out of its program. The only acceptable question is the one about the content that already has a closed answer. But, in fact, “what is authoritatively intended with the imposed silence, in the name of order, it is to drown in it the inquiry” (Freire, 1985, p.47). Passive people who do not inquire go in the opposite direction of the demands of our times.

Freire insists, “*Education in general is an education of answers, rather than being an education of questions. An education of question is the only creative education and able to stimulate the human capacity to marvel, to respond to his astonishment, and solve their real essential and existential problems. It is knowledge itself.*” (Freire, 1985, p.52)

By sharing Freire’s ideas about the importance of curiosity as part of education and Papert’s ideas about constructionism, the One Laptop per Child (OLPC) Learning Team introduced the Matsiko idea to schools in Rwanda, as one of its micro-level initiatives in the country.

One Laptop per Child is an NGO, whose mission “is to create educational opportunities for the world's poorest children by providing each child with a rugged, low-cost, low-power, connected laptop with content and software designed for collaborative, joyful, self-empowered learning.” (OLPC) In order to improve the quality of education, OLPC's mission is not only to provide laptops, but to create a culture of learning, engaging children in their own education, and developing the passion for learning.

The goal of this paper is to detail one of our initiatives to turn the OLPC mission into reality. We will highlight Matsiko and its principles, showing new possibilities for instantiating dynamic learning environments, illustrating with our practice in Rwandan schools and also trying to explain how this specific action aligns with our vision of computers use in education, the laptop initiative’s model of growth and Rwanda’s vision for economic development[1].

One Laptop per Child in Rwanda

Rwanda is a small country of eastern-central Africa squeezed between the Democratic Republic of Congo, Kenya and Tanzania. Despite its size, it has a very high population density with more than 10 millions inhabitants, with almost 1 million living in the capital city of Kigali.

As with many different African countries, Rwanda has plotted a national plan to guide the country’s growth, and in Rwanda's case, out of the devastated landscape left by the 1994 Genocide. The VISION 2020, as the plan was named, “*seeks to fundamentally transform Rwanda into a middle-income country by the year 2020*” (Rwanda Vision 2020, p. 9). Rwanda

[1] We would like acknowledge of all OLPC learning team based in Rwanda, specially, Jimmy Parfait, Désiré Rwagaju and Joy Riach for their help in the implementation of this project.

Vision 2020 elected the area of services, specifically tourism and ICT, as the ones with the best chances to become the country's economic engine.

It is not necessary to say that this choice brings its own myriad of challenges, but at least it sets the country's focus on developing its population as the solution for social and economic growth. Essentially education becomes the country's major priority and a world-class educational system becomes a necessary step. For education to accomplish its role, a transformation of the learning environment is also essential. Technology enables the transformation, not only by providing access to high-quality current materials for every child in a more cost-effective way, but also by providing a learning environment where children can truly develop the requisite knowledge and 21st century skills.

The vision that One Laptop per Child initiatives might be one of the solutions to leap ahead in education was what motivated the Rwandan government to commit its scarce resources to such a complex project. In November 2007, Rwanda started one small pilot with 100 laptops and two classrooms in the Rwamagana B Primary School. In 2008, the pilot was expanded to 10,000 laptops in 17 schools, which provided valuable experience. Moving to the next phase, the government ordered 100,000 laptops in 2009 and started to prepare for a large-scale deployment in 2010.

One Laptop per Child valuing the vision and commitment of Rwanda moved its Learning Team to the country in 2009 and started to work in partnership with the government to spur the implementation of the project in the country and in the region. The experiences described in this paper represent a segment of the activities of the OLPC Learning Team in Rwanda and are a small part of a larger implementation project.

A View of the Rwanda School System

There are no formal studies about the current school and teaching practices in Rwanda that we can use to draw a picture of the pedagogical aspect of education in the country. Such a portrait is necessary for the purposes of this paper in order to discuss the context where the Matsiko Project is being developed, how it connects with OLPC's work in the country and why we consider it relevant.

We can count on some official information to understand the big picture. Rwanda has 2.2 million children enrolled in primary education. Until 2009, the country had 6 years of primary education with grades conveniently named from Primary 1 – P1 to Primary 6 – P6. In 2008, the government started a program to expand primary education to 9 years as part of its commitment to the Millennium Development Goals of achieving universal primary education by 2015. Primary education is defined as fee-free, meaning that students do not pay fees to attend to school.

In 2008, the Ministry of Education also started to move the education in the country to a tri-lingual system. This means that Kinyarwanda (the country's mother tongue), French and English are all studied simultaneously at school, with English being the main language of teaching. Another major change in the recent past for the primary education was the adoption of double shifts as a strategy to get better teacher/student ratios. Even with this measure, classes with 70 students are not unusual, especially in the lower primary grades. Lack of textbooks and libraries are also a big challenge to the government.

After 18 months of working with schools in Rwanda, it is possible to share some observations of the pedagogical practices in the country. Such testimonials do not have the value of a formal study, and, most importantly, should not be generalized to all schools in the country. Although we think they bring some context to this paper. The pedagogical practices of primary education teachers in Rwanda are based on oral instruction, copy and memorization. Class activities usually consist of the teacher reciting a lesson, students taking notes, and then the teachers making questions about the subject with the expectation for a single "right answer". This reflects a noticeable lack of qualification of the teachers that, despite their wiliness and tremendous enthusiasm, were not prepared as professional teachers. Most of them qualified as a teacher by simply having finished Secondary school.

Models of Growth

To spread change in the educational system is a much more complex task than it might initially appear from a management perspective. The injection of new ideas into schools always faces the problems of rejection or accommodation to the pre-existing structures and practices. As with any attempt to bring new ideas to the school system, the OLPC initiative in Rwanda faces the same challenge: to devise strategies of implementing new educational practices on a large scale.

Most efforts to school reforms, independent of the quality of its ideas, have failed with the attempt to grow from pilots to large scale because of the use of predetermined, usually massive, fully formulated designs imposed from above. The adoption of a cascade model, training the trainers, does not work most of the time because quality decreases with the growth of the chain. The trainers tend to replicate the form without understanding the principles (Cavallo, 2004).

We need to approach educational change in a more systemic way, thinking about it not as the outcome of a single action or as an instantaneous process. Many initiatives will be necessary to create a new learning culture and in each place a different strategy is required, because the needs, concerns and possibilities of the circumstances are different.

Cavallo (2004) proposed a new framework for thinking about change and growth. Based on Kuhn's description of components of a paradigm, he suggests three fundamental elements to think the change: exemplars (real experiences), models (expectations of the outcomes) and symbolic expressions (language about learning).

Laptops may provide a means for new models of growth. Rather than needing to rely on a centralized, standardized reform, we can develop high-quality, localized models of improved practice, and utilize the network and rich media to create mechanisms for spreading. A foundation is thus created for three distinct, but overlapping, phases: enabling powerful learning in and out of school; the positive change to specific school practices; and the transformation of schools from funnels of received information to engines of knowledge construction and appropriation.

We also do believe that change is inherently a process of learning, where we cannot accelerate or impose our ideas onto people's minds. We do want to create a different mindset about learning. But, as when we work with children, we should create opportunities of learning and of development. For that we need to destabilize teachers' certainties about their practices and about learning. Many teachers do believe in what they do because it is the only way of working that they know up until now. They do not feel the need to change or to try new approaches. Therefore the production of necessity is one of the challenges for promoting educational change.

As outsiders, we also cannot foster change by ourselves since we lack the cultural grounding necessary to promote emergent designs. As a principle, we work through the development of people, so in the future they can assume a role of leading and supporting the change. "Rather than focusing only on change within an existing institution, we adopt a broader view of change with human agents as carriers hosted by a variety of institutions with the change developing through improving practice and developing ideas through the reflective trial and error of creating exemplars." (Cavallo, 2004, p. 109)

The Models of Growth framework in Rwanda

Papert used Piaget theoretical framework to compare the school as a complex system, that has mechanisms to protect itself from constant buffering of small chaotic changes. As an organism, school tries to resist changes to its previous structures, therefore transforming any reform attempt into something that can assimilated into the current system. Piaget also offers us a framework to think educational transformation by postulating "process of growth happens inside pockets of stability(...) that eventually become strong enough to overcome the resistance" (Papert, 2001)

Among OLPC's 5 principles (OLPC) two of them were designed to avoid that the usage of laptops happen strictly when the teacher or the school allows. **Child's ownership** of the computer (kids take laptops home) and **saturation** try to assure that children have access to the machine whenever they want.

However, the simple distribution of laptops doesn't guarantee by itself that school system is going to move from its current state. Teachers can simply assimilate the laptop to their current practices and children may not reach powerful uses of their machines without some support.

Matsiko is one instance of micro-level activities that aims to contribute to fill this gap using the context of child's ownership and saturation to spread and achieve macro-level impact in the school. The principal goal of the activity is to provide children with personally powerful learning experiences through the raising of their questions and "construction" of their answers. By instantiating such kind of activities we hope to provide to teachers with **exemplars** and **models** of learner-centered and constructionist approach they can relate to.

During the development of Matsiko activities we got involved Rwandan university students (through volunteer and internship programs) so they could understand the principles underlining this kind of activities and possible develop local and more culturally fitted versions of it. By the development of local people we aims to provide them with a new language and models of learning and teaching that can spread and grow in the "fertile ground" created by a community saturated with laptops.

The Matsiko Principles

The Matsiko initiative is one pedagogical practice that tries to develop student's critical thinking and scientific inquiry through the investigation of their own curiosities about their world. The Matsiko concept is deeply rooted in Fagundes' work (Fagundes, 1999), especially her Learning Projects methodology. It was developed during the work with 6-year olds in 1:1 contexts in Brazil (KIST, 2008) and it acquired its current format during the work with students in Rwanda. It is not definitively a one size fits all formula, but a pedagogical practice with some principles and many possible implementations. It has five core main principles:

- It needs to start with a kick-off activity that engage children in the practice of questioning;
- Students must be encouraged to make questions and each child needs to do at least one personal question;
- Students must decide which question they want to investigate and usually only one project is carried out at a time;
- The teacher acts not as the question's answer but as the student's guide in the investigation process;
- The investigation of the question is a cooperative construction process executed by the children. It isn't only an issue of providing the "right answer" nor to trivialise the process only doing a search on the Internet; and
- Students must create models and simulations as part of their research and be able to participate in design critics sessions. The models need to be tested and confronted with the real world and therefore debugged.

In Fagundes' Learning Projects (LP), the main focus is also the development of an investigation question by each student in the classroom that should work on their own investigation during a few weeks individually or in small groups. Based on Piaget's ideas, Fagundes assumes that when a child does a question, he or she already has a simple hypothesis about its answer. She also thinks that an authentic question is deeply connected to the student's personal history and interests; therefore, the child already brings with it a big motivation to investigate its answer, a pool of information/experiences that allowed him to formulate it. The project needs to serve the children's passions and curiosities and not a pre-defined curriculum.

The major difference in Matsiko from the LP's is that, at least in the beginning, only one question is investigated at a time by a class. With the LP's, there are many questions at the same time by different groups of children. The rationale behind **many questions** is that different children inside a classroom have distinct curiosities and nurture different passions; therefore, they should work in projects that they are interested in. By selecting only **one question** we know that the selected one will not reflect the curiosities of all students and lose part of its meaningful aspect and as a result, will cause some students to not get engaged. Nevertheless in a classroom of 6-year olds in Brazil, as well as, in our experience in Rwanda, students still lack enough autonomy to develop the whole investigation by themselves. In the case of early elementary school, children are still too young to work without close supervision of the teacher. In Rwanda, the students are, most of the time, very passive inside the classroom, always waiting for the teacher's instructions. In addition, Rwanda has limitations because of the lack of written documents and of materials to help them find information or to create scientific experiments.

We agree in principle that **many questions** is a better practice, yet the required autonomy does not develop fast. We see the Matsiko approach as an intermediary step to develop the child's capacity to work more autonomously. By investigating a single question at a time per class, we try to keep some important ideas of Fagundes' methodology but try to bring it closer to actual possibilities of implementation by teachers and interns.

In a pedagogical practice of the question, like Matsiko, the teacher is not the answerer of the child's question. Quite the contrary, he/she needs to control his/her own desire to provide the right answer to the child and instead to work as an advisor by helping the child to construct knowledge. In other words, it is important to understand the child's thoughts behind the question and based on that, to try to do an intervention that allows the child to see the contradictions of his/her initial ideas. Through this game of equilibrium and disequilibrium of the child's hypotheses, the teacher will guide the child to a more stable explanation created by the child on his/her own.

Fagundes' work is deeply influenced in the early works of Papert (1980) and both share a common root in Piaget's ideas. Although, in the last decades with the advent of Internet, the idea of building things and debugging got de-emphasized compared to the informational aspect of the computer's use.

How it Works

In this paper we are only focusing on the Matsiko initiative developed in the scope of the OLPC initiative in Rwanda. In 2009 we did the Matsiko Project with 5 different groups of students in the format of summer camps and after school programs. It involved 200 students from two public schools in Kigali. The students were selected by the schools and studied in the 4th and 5th grades; the average age of students was 14 years old (the youngest was 10 years old and the oldest 23 years old). The first Matsiko group worked 4 hours per day during the one-week summer camp. The other four groups, since it was during school period, worked 2 hours per day in the opposite class shift, once a week, during 10 weeks.

As mentioned before, the first step for all Matsiko begins with a **Start Up** activity. The chosen activity was to read to the children from a book entitled, Curiosity is the Award Itself (*Curiosidade Premiada in Portuguese)* (Almeida, 2000). This book is originally written in Portuguese, but it was translated into English and most importantly into Kinyarwanda. The book tells the story of Gloria, a little girl whose family thinks she is sick because of her habit of making lots of questions. Taken to a doctor, Gloria is diagnosed with "Accumulated Curiosity" and the prescribed treatment is to try to satisfy her curiosity and the book illustrated how her family transformed this problem into an opportunity to discover interesting things about the world.

In the context of Rwanda education, this story was fundamental to engage the students in the activity, especially when you consider that making questions about what you are interested in is not a common practice at school. The fact of listening to a story was meaningful to their oral culture. In the same way, it was important for the kids to have the book loaded onto their laptops. It opened the possibility to come back, to see the images, to read it again and again and also to spread the story into their school and into their homes.

After reading aloud the story to the whole class, we asked the children to find the book on their laptops and read it. In practice, children got inspired by the story and started to express their own curiosities, like a ritual to become part of the "Matsiko Club." Since they had many questions, we asked them to choose the 1 or 2 they consider the most interesting and to write them in the text editor. By doing that, they started to learn how to find and to open the book's file in the Sugar's journal (to open, to change the page, to zoom in and out) and also how to use the Write activity for notes (to open, to save, to find the document in journal). Next, all the students shared their questions with the class.

Since the beginning, we had many hypotheses of which kind of questions Rwandan students would formulate and how hard it would be for them to express them (even in Kinyarwanda). Fortunately the children were very fertile in expressing their curiosities and theories about the world. Some of the questions the children made are listed below:

- Why some people are white and others are black? (Kagugu school,G1)
- Why HIV is transmitted only by sexual relations? (Kagugu school,G2)
- Under this world, there is another world? (Kagugu school,G2)
- Why do elephants have big noses? (Kagugu school,G2)
- Where is the path to the moon? (Kagugu school,G3)
- If we dig very deep can we find the hell? (Kagugu school,G3)
- How does the airplane fly without wind? (Kagugu school,G3)
- Where does the human come from? (Nonko School,G1)
- Why snake does not have legs? (Nonko School,G1)
- People use to say that the earth turns around the sun, why don't we feel it? (Nonko School,G2)
- Why do people say that a stone doesn't have life? (Nonko School,G2)
- If Egypt is up and the heaven is up too, is the heaven in Egypt? (Kagugu School,G2)

In this small sample of a much larger set of questions, it is impossible not to be mesmerized by the creativity and diversity of the children's curiosities. Through the analysis of those questions, we can unveil a lot of information about the children's logic, theories, misconceptions, beliefs, values, concerns, fears and also about their educational system.

Let us stop for a moment and take a closer look at the question: "**If Egypt is up and the heaven is up too, is heaven in Egypt?**" When we first listened to this question, we got quite confused about its meaning. It was necessary to ask further explanations to the student who made it. He explained to us that when he goes to church, the priest says that when they die, they will go to heaven (in many languages, including Kinyarwanda, the words for "heaven" and "sky" are the same). When he studies geography at school and the teacher draws the map of Africa on the chalkboard, Egypt is always above Rwanda (up, if you consider the orientation of the map drawn on the board). In his line of inference, Heaven was probably located on Egypt because this country is "up."

We believe this question is a privileged example to show how children try to make sense of their personal experiences and all the information they receive, and that they do this by an active and constructive process, in this case elaborating a theory about the physical location of heaven. It also shows to us how superficial and abstract the study of geography was for this student who, only by seeing an illustration on the chalkboard, could not create a spatial representation of the map.

Continuing on the activity, after the class had the discussion about the questions, as a democratic practice, the students voted for the question that would guide their investigation. During the Matsiko implementation in Rwanda, the following questions where selected by different groups of children:

"Where does the sun go when it is night?" (Kagugu school, G1), **"Where is the path to the moon?"** (Kagugu school, G2), **"If Egypt is up and the heaven is up too, is heaven in Egypt?"** (Kagugu school, G3), **"Where does the human come from?"** (Nonko school, G1) **"People use to say that the earth turns around the sun, why don't we feel it?"** (Nonko school, G2)

Each and every question has a story, as simple as it might be. By enunciating the question, we know that children already know something about the question's background and sometimes have hypotheses about possible answers. The next step was to ask children to formalize such information by writing it in their laptops. To record their previous ideas is important for the students to notice what they knew about the subject, what they thought they knew and later on to compare with what they are learning.

After planning collectively how to investigate their question, they started to put it into practice. In some cases, we started testing the previous ideas, where kids could make it through drawing, text, video, programming or experiments. In other cases, as the one about the question **"Where does the sun go when it is night?"** (Group 1), we started by observing the sun and its position in the sky.

During Group 1's investigation, after observing the sun, we had a discussion about its positions in the sky at different times of the day. Where is the sun in the middle of the day? Where is the sunrise? Where is the sunset? Based on the observation, two ideas came from the students: the sun is small and moves. In order to put this idea in contradiction, we proposed for the students to look for information on the Internet, on Wikipedia, and on the offline contents on the laptop to check if those ideas were right. During this time, they learned how to access the Internet.

Since Internet was not working well, we decided to look for information on the laptop's library and we found a book about the solar system with images. In the graphical representation of the solar system on the laptop, the students could observe that the sun is bigger than the earth and that the earth and other planets turn around the sun.

At this point, one question came from the students: "If the sun is bigger, why when we look at it in the sky, it looks small?" One student expressed the hypothesis that it happens because the sun is far away. In order to test this hypothesis, we went out of the classroom and did an experiment to understand why things far away look like they are smaller. All the students used the camera to make a video of the globe, an object available at school. In the beginning, the globe was close up and it looked big. Then one student started to walk with the globe in her hands and the farther the globe moved, the smaller it looked. (see Figure 1).

Figure 1. Children doing an experiment about perspective where the laptop records the globes movement.

Through this experiment, watching the video made by each child and reflecting about it, it was possible to understand the relativity of our perception, in this specific case, why things far away look smaller.

The students also found in the solar system book that there are other planets. In addition, another student found information that the earth turns to make day and night and it takes 24 hours. We used the globe and the children's body to simulate the earth's movement to make day and night.

For symbolic representation of this knowledge, they needed a tool where they could play, try, experiment, and represent their comprehension. For this we started to use Squeak/Etoys. Etoys is a media-rich authoring and programming environment loaded on the laptop, where children can draw or import objects and attribute commands for these objects, making animations through their own scripts. Etoys is a tool to think with. Its principle is to "make abstractions more palpable, allowing children to visualize and explore new ideas." (Etoys site).

In our work, using Etoys, the students constructed their own simulations about the solar system. Some students drew all the objects, others imported images from the Internet and from the laptop. After that they used the programming model of the Etoys, choosing commands to create their own scripts in order to simulate the earth's movements.

When they shared with classmates and we compared their simulation with what they had learned, they faced some problems like: size of the sun and the earth and the trajectories of the earth and the sun. We discussed again which is bigger, how much bigger is the sun, and what are the earth movements, if the sun moves. We also used the body to simulate these movements. They arrived at the conclusion that their initial hypotheses were false. The sun is bigger and does not move or turn. The sun remains in the same position. It is the EARTH that turns, around itself and around the sun. At night the EARTH shows another part of the sun.

After that discussion, the students came back to Squeak/Etoys to change their projects by making the sun bigger than the earth and the moon smaller than the earth and by programming the moon's movement to turn around the earth and the earth to turn around itself. To implement the model and reflect about it was a precious moment of learning. Before the simulation on the XO laptop, the students appeared to understand, but when they expressed by simulation, they realized they did not understand initially.

This means that it was necessary for the students to express their own ideas by doing, even committing mistakes. By debugging the models, the students face contradiction with their previous theories and this is a way to build new knowledge. That is what we call, "constructive mistake," which is necessary in order to advance in their knowledge construction.

Figure 2. Solar system simulation done in Etoys(left) and a girl explaining her work (right).

What are Children in Rwanda Developing, Using the Laptop to Investigate Curiosities?

When kids are doing Matsiko, they are developing many skills that we cannot measure through traditional examinations. The main point in this practice is not the learning of the formal contents, out of context and meaningless for the students. The proposal is to learn meaningful content through procedures that develop their own competence and skills to keep learning. They are experiencing new ways to learn.

Through this, they are developing the Century XXI skills, highlighted in the Vision 2020 document, which are Creativity, Innovation, Critical Thinking, Problem Solving, Communication and Collaboration. Making investigations, they are developing science concepts too.

Through 1:1 model combined with project-based practices, students are developing Technological Fluency (MIT). Instead of using the laptop to learn office skills, they are finding ways to use the laptop with their real problems, inside a context, in facing a need and in a more sophisticated way. It changes the mindset about the way the laptop can help in learning process.

Today we need to understand the technology deeply, but this does not happen out of our reality. We know that we learn when something is meaningful to us, when we make sense in our significations system. When kids were making the investigation, they were learning a lot about the laptop use, but it was not necessary to have a class to teach how to use the laptop. The students were learning about the machine during the investigation process, since they had a need to learn it. We taught some techniques to them, but it was not a technique by itself. The technique was a way to solve a problem, to improve an experiment or a model.

In addition, Matsiko also promoted the students to express themselves creatively, building models about the world. Moreover they are developing literacy. This is a big issue everywhere. The scholarly system is failing in this area.

Matsiko mindset brings many possibilities to develop literacy. Literacy does not mean to be able to transform code in sounds and to write (encode) without mistakes. We understand literacy as an ability to use the written language in specific situations, as well as, to understand meaning and to express meaning, being able to create things with the language. The investigation's question creates the context for children to read something meaningful and also to write with authorship. They have the opportunity to express their ideas by writing without copying models. For that we proposed that students had to document their project in a diary on the laptop. This had the goal to make them reflect and to be aware about what they were learning. Beyond this, by keeping the diary, they had the opportunity to write with authorship without copying from the blackboard. When we do this in a blog on the Internet, as we did with 6-year old children in Brazil (Kist, 2008), it is more powerful, but we need to adapt our ideas to our real constraints.

Reading skills also were developed because the students had the book loaded onto their laptops and we observed them reading it many times, in Kinyarwanda and in English and also comparing the two languages. Moreover, the information about the projects subjects was another topic of interest for them to read. Since the laptop's internal library was only available in English, it created a barrier for most students. The Matsiko activities were made in Kinyarwanda to make students comfortable with expressing their ideas, but by developing the project, the students started learning some English words.

At School, What is Changing?

Matsiko is an OLPC initiative that happens inside the school, with students, but until now it is not a scholarly activity. Teachers are welcomed in the Matsiko class but they are not responsible for the activity. It happens as an extra-curricular activity for a limited group of student. Even so, we could observe some movements at school.

All students wanted to be part of it. Considering the complexity of the work and the limited number of people to execute it, we could only work with a limited number of students, but we had problems because every day the number of students that showed up grew bigger.

The scholarly community has seen the students' engagement with the activity; the students came everyday, worked hard and had fun. The investigation question got spread across the schools. Students that were not in the Matsiko group knew about the question, others asked for the explanation about the research and how to have the Matsiko book loaded onto their laptops.

The vision about the laptop is changing also. Until now, the teachers and the parents did not have a clear idea about what was possible to accomplish with the laptop. To play or to obtain ICT skills were their previous ideas. After Matsiko, they could see that they can use the laptop to learn and to create things. The headmasters of two of the schools that we work with asked us to offer the same experience to more students.

The families started to come to the school and show more interest about what their children were learning. In one situation we praised a child to his father saying how smart his child was, but the father did not believe us and said in simple English: "no, not my son." Other parents were proud of their children when they saw their projects on the laptop. The community, in general, that seemed suspicious about the students' abilities started to change their opinion when they saw and listened to the children's explanations about their projects.

Final Considerations

We acknowledge that a large-scale implementation of Matsiko inside Rwandan schools would be precipitated. Making it would imply a radical change in all methodology of teaching. As we said before, change is a process of learning that cannot be imposed top-down to teachers. To engage in a proposal like Matsiko, teachers need to feel that necessity, they need see that it works, and they need to feel comfortable with that. That takes time. Nevertheless, Matsiko brings concrete basis to question the existing practices and further create such need. Matsiko shows to the community some possibilities of how the laptop can help in the learning process and also to break the mindset that children can only learn by being taught.

Proposals like Matsiko may not be adopted as an institutional practice, but it can spread through the work of individuals. During Matsiko activities, we have met local people that have shown potential to become the agents of change. Those people have been working with us, reflecting about our actions, getting appropriated to the principles and even creating their own ways to implement it. They will be the ones that will adapt Matsiko ideas to other situations and probably to create their own learning projects. “Individuals more than institutions are the generators of growth, enhancement, and sustainability” (Cavallo, 2004, p. 105).

Finally, it is important to highlight that we do believe that macro level changes are going to be the outcome from many micro level actions and individuals will be the seeds of it. Matsiko is one of these micro-level actions that try to rescue the child’s ability to marvel, by applying the biggest powerful idea of all: to make a question.

References

Almeida, F. L (2000) *Curiosidade Premiada*. Ática. São Paulo/Brazil.

Cavallo, D. (2004) *Models of growth – towards fundamental change in learning environments*. In BT Technology Journal. Vol 22 No 4 . October.

Etoys website. http://www.squeakland.org/index.jsp (Accessed January 2010).

Fagundes, L. et al. (1999) *Aprendizes do futuro: as inovações começaram!* MEC, Brasília/ Brazil:.http://mathematikos.psico.ufrgs.br/textos.html. (Accessed January 2010)

Freire, P. et al. (1985*) Por uma pedagogia da pergunta*. Paz e Terra. Rio de Janeiro/Brazil.

Kist, S. (2008) *Um laptop por criança: implicações para as práticas de leitura e escrita*. Master. thesis, Universidade Federal do Rio Grande do Sul, Porto Alegre/Brazil.

Loewenstein, G (1994), *The Psychology of Curiosity: A Review and Reinterpretation*, Psychological Bulletin 1994, Vol 166, No. 1, pp75-98.

MIT Media Lab.*Technological Fluency.* http://llk.media.mit.edu/projects/clubhouse/research/ handouts/fluency-v3.pdf (Accessed January 2010).

OLPC website. http://laptop.org (Accessed January 2010).

OLPC 5 Principles. http://wiki.laptop.org/go/OLPC:Five_principles (Accessed January 2010).

Papert, S. (1980) *Mindstorms: Children, Computers and Powerful Ideas.* Harvester Press. Brighton.

Papert, S. (1991) *Change and Resitance to Change in Education. Taking Deeper Look at Why School Hasn’t Changed.* In Novo Conhecimento. Nova Aprendizagem. Papers presented on the International Conference New Knowledge, New Learning - Lisbon, October 2000. Lisbon: Calouste Gulbenkian Foundation, 61-70

Rwanda Vision 2020. (2000) Ministry of Finance and Economic Planning, Republic of Rwanda.

The Principles of Educational Robotic Applications (ERA)

Dave Catlin, *dave@valiant-technology.com*
Valiant Technology Ltd, Grange Mills, Weir Road, Balham, London SW12 0NE, England

Mike Blamires, *michael.blamires@canterbury.ac.uk*
Dept of Education, Canterbury Christ Church University, Canterbury, Kent, CT1 1QU

A framework for understanding and developing educational robots and their activities

The original educational robots were the Logo Turtles. They derived their rationale from constructionism. How has this changed? This paper postulates ten principles that underpin the effective utilisation of robotic devices within education settings. We argue that they form a framework still sympathetic to constructionism that can guide the development, application and evaluation of educational robots. They articulate a summary of the existing knowledge as well as suggesting further avenues of research that may be shared by educationists and designers. The principles also provide an evaluative framework for **E**ducational **R**obotic **A**pplications (ERA). This paper is an overview of the ideas, which we will develop in future papers.

Keywords (style: Keywords)

Educational Robotics, Constructionism, HRI, HCI, Robotic Applications, Machine Mediated Learning, Cross Disciplinary Research, Collaboration, Logo, Roamer, Turtle, ERA, STEM.

Introduction

Logo combines philosophy, educational theory, artificial intelligence, cognitive science, developmental theory, neuroscience, robotic engineering and computer science. It emerged in the 1960s when most of these disciplines were still in their infancy. Post modernism, logical positivism, phenomenology and deconstructionism were disrupting age old philosophical positions. Turtles, the first breed of educational robot, emerged as part of Logo and shared its intellectual grounding particularly its constructionist approach to education. While the intervening years have seen significant developments in the underpinning sciences, little has been done to review their overall and collective impact on the way we use educational robots.

While never becoming extinct, real Turtle robots faded into the background as researchers almost exclusively worked with virtual robots. This is changing. Writing in the Scientific American, Bill Gates predicted “robots will be the next hot field” (Gates 2006). Certainly, this rise in popularity has started to appear in education. Consequently a review of the intellectual and practical basis relating to our use of educational robots becomes urgent. This paper is the result of that review. We propose that ten Educational Robotics Applications (ERA) Principles summarise the value of robots and robotic activities in any educational context.

We start by making a set of simple claims why we think these Principles are of value. We follow this with a description that references some of the supporting evidence and conceptual grounding. In order to provide some degree of ‘future proofing’ and to make the postulates independent of the type of robot, we have kept the descriptions as abstract as possible. Where contextual instances help to clarify our meaning we have used examples.

Although we call these Principles we are aware of their hypothetical nature. Over the coming years we expect research activity will gradually confirm, change, delete or find evidence that will steadily transform the postulates into verified principles. We finish the paper with a brief introduction to the e-Robot project which aims to accomplish this validation process.

Introducing the ERA Principles

The Principles are not stringently independent ideas. They form a holistic set of values that integrate in different combinations. For example Personalisation Engagement and Equity share an affinity. Personalisation also resonates with the Practical, Curriculum and Assessment, and the Pedagogical Principles.

The use of robots involves the interaction of students, teachers and technology. We have grouped the Principles under these headings more to assist their recall than an exacting effort of categorisation.

Technology
1. Intelligence
2. Interaction
3. Embodiment

Student

4. Engagement
5. Sustainable Learning
6. Personalisation

Teacher

7. Pedagogy
8. Curriculum and Assessment
9. Equity
10. Practical

Table 1 The ERA Principles

Why the ERA Principles?

The Principles present a framework that:

1. Explains:
   a. How robots help students learn
   b. The benefits of educational robots to teachers
2. Offers a check list for those who want to:
   a. Design educational robots
   b. Develop activities that use educational robots
3. Helps justify the investment by schools in robotic technology
4. Suggests underlying cognitive and developmental processes
5. Provides researchers with a set of claims to evaluate

Intelligence

Educational Robots can have a range of intelligent behaviours that enables them to effectively participate in educational activities.

An exploration of this principle needs to explain what we mean by:

1. Intelligent behaviour
2. Effective participation

For our purpose we recognise intelligence as belonging to a spectrum of behaviours focused on intentional goals (Sternberg 1985, Stonier 1997, Freeman 2000, Sternberg et al 2008). This means the robot need only possess task specific intelligence, which targets explicit learning objectives, rather than a general ability to act in unstructured situations. In this sense educational robots need to help students acquire specific knowledge, provoke them into thinking, help to develop skills or provide them with experience of situations and knowledge structures that mirror useful thinking patterns. They provide students with opportunities to use their knowledge in problem solving and engage in knowledge transfer, generalise concepts and develop their social skills.

Currently deep-down in their microchips, educational robots are based on what Winograd and Flores termed Western rationalistic tradition (Winograd and Flores 1986). These represent powerful thinking patterns capable of supporting many useful educational applications. Logo is an example. When a version of it is internalised into a robot's core behaviour it dictates what the robot can and cannot do. As technology and our understanding of educational robotics develop we expect to find new "core" behaviours capable of supporting different learning experiences.

Effectiveness contains the notion of efficiency, which we take to mean improvement. That is, students grasp ideas faster; get a better understanding of concepts, etc. This is relative. We grasp the idea faster than if we used some other method. It depends on which student and which method and what works well for one student may not work so well with another.

Effectiveness also depends on the skill and experience of the teacher. Teachers teach: the technology is a tool to help – not replace them. Not every teacher will exhibit the same aptitude for using educational robots, irrespective of their general teaching skill. Whereas an adept, well trained teacher will achieve brilliant results, a robot will not make up for teaching deficiencies.

Generally, the measure of effectiveness is statistical. In most applications, with most students and most teachers, we expect intelligent robots will enhance educational achievement. If a robot does this for just one student it is valuable. The need for the statistical verification is economic: it is hard to justify the cost of a robot system for singular teaching successes.

Interaction

Students are active learners whose multimodal interactions with educational robots take place via a variety of appropriate semiotic systems.

Working with robots is an active learning process, which is generally more effective because it is multi-modal. Interaction always involves the use of a semiotic system. Semiotics is usually defined as the science of signs (Halliday 1978). Crystal (1999) offers a more appropriate definition, which captures the heart of any educational enterprise:

Semiotics: The study of signs and their use, focussing on the mechanisms and patterns of human communication and on the nature and acquisition of knowledge.

Signs evoke meaning through culture and context. For example in the West the colour red implies danger whereas in China it means good luck. However, the "value" (meaning) of the sign changes according to its use. So for example a red cross suggests medical help. Education is about learning the signs and signifying practices of our culture.

Logo is a semiotic system. We communicate our ideas to a robot by manipulating Logo symbols (commands) according to rules (programming syntax). The robot provides feedback through its movement – a sort of mechanical “body language”. We can use this “body language” schema to understand other semiotic systems. For example if we place a robot on a number line and make it move by manipulating symbols (numbers and operation signs) using the rules (addition, subtraction, multiplication and division) students can explore the semiotic systems of numbers and arithmetic. Consider the equation (+4) – (-3) = (+7). Students are normally taught to solve this problem by remembering a meaningless rule like two minuses are a plus. Using the robots students use their visual, kinaesthetic and spatial modalities to develop mental models of negative number arithmetic. Importantly, they learn through understanding (NCTM, 2000 and Bransford, et al 2000). They see that on the number line to get the robot from (-3) to (+4), the robot has to travel (+7). This emphasises the meaning of the number system, particularly the relationships between positive and negative integers and the idea of subtraction as “difference”.

Up until now robots have been dumbstruck[1]. Yet, natural language is humanity’s major semiotic communication system. Valiant’s new Roamer is changing that. The basic robot has a very powerful speech capability. This opens up many tantalising possibilities. For example by incorporating Logo’s list processing ability, we can explore embedding in the robot the language ideas explored by Golenberg and Feurzeig (1987). In Incy Wincy Spider, an Early Years comprehension activity (Valiant 2009), Roamer sings out the verses of a nursery rhyme. The students realise the robot has “got it wrong” and their task is to teach it to get the verses in the right order. They do this by pressing the keys representing the “action” of the rhyme.

The Incy Wincy activity involves sequencing, a precursor to programming, which has been the primary way we interact with educational robots. If we transform the phrase “human communication” used in Crystal’s definition of semiotics to the more apposite “Human Computer Interface” (HCI) and Human Robot Interface (HRI) we open exciting new possibilities. Forerunners of this technology are already finding their way into toys (Bartneck and Okada, 2001). And the work of some researchers on sociable robots (Brazeal 2004, Dautenhahn 2007) shows the possibility of very natural interactions between student and machine. For example AnthroTronix used Roamer as a basis for their Cosmobot robot. They have developed an interactive glove through which children can operate the robot through American Sign Language. The Principle also embraces the idea of tangible computing, which involves students purposeful construction of environments that control the behaviour of the robot.

How can this assist education? Vygotsky’s concept of “tools” is a fertile starting point. The influential Russian psychologist proposed that just as we used tools to impact our external environment we need tools to modify our behaviour. Semiotics was the foundation of these ‘mental tools’ by which Vygotsky meant language (Wertsch 1985). Clearly robots represent physical tools which Papert, borrowing ideas from Winnicot (1971), called “transitional objects” or “objects to think with” (Papert 1980). Activity Theory (Leontiev 1978, Davydov and Radzikhovskii 1985, Engeström 1987, 1999) grew out of Vygotsky’s work. This theory orientates us to a world of objects and our mental interactions with them. Some work on this has been done in relationship to Activity Theory and HCI (Nardi 1996). It is our contention that extending this work into educational robotics will provide a deeper understanding and offer new perspectives on the Interactive Principle.

Logo Turtle robots formed the prototype educational robot system. Logo offered new ways for students to develop mathematical, computational, geometric and scientific skills (Cuoco 1990, Kyngos 1992). From the initial conception of Logo (Feurzeig, et. al. 1967) to the existence of effective educational applications took many years and a great deal of research (Papert et. al. 1971 to 1981). As new robotic and HCI/HRI technologies emerge they will need to undergo the

[1] The Tasman Turtle and some toys like Furby had limited speech capabilities.

same process, but gradually we will see an increase in the capability of robots to support teachers and help provide valuable learning experiences.

Embodiment

Students learn by intentional and meaningful interactions with educational robots situated in the same space and time.

We propose that by interacting with physical robots students can have positive educational experiences. And in a special caveat the claim extends to positive experiences that at a minimum are qualitatively different to those with virtual robots. While 30 years of practical work in schools has shown that thousands of teachers share this intuitive view, there is little hard data to verify the claim. Such evidence is contradictory, flimsy or does not target embodiment (Mills et al 1989, Gay 1989, Syn 1990, Weaver 1991, Mitchell 1992, Betts 1997, Adolphson 2005).

Our proposition does not critique the value of educational software. Instead, we aim to affirm the potential of physical robots. Our claim is built on a theoretical framework that has two strands:

1. Work by various authors in the areas of embodied cognition, AI and robotics
2. The original body syntonic claims of Seymour Papert (1980a)

Embodiment in cognitive science claims three things:

1. Mind has evolved, not as a machine, but as an integrated element of an organism embedded in a society and in a physical temporal world.
2. Mind and body are intimately intertwined. They form an ‘adaptive system’ – that works together to survive and thrive as their environment changes.
3. Most embodied cognitive processes are subconscious.

The concept of embodiment is rooted in biology (Muratana and Verela 1987). Despite this some writers have applied the term to software (Franklin 1997). Others argue that bodies are essential to cognition (Pfeifer and Scheier 1999). A survey (Ziemke 2001) looks at what kind of body is required. We restrict our meaning to living entities (students/teachers) and physical robots.

Embodiment is about how we engage with the world, extract and share meaning through our interaction with it and the objects it contains (Dourish, 2003). It is self evident that this applies to robots. But does it apply to virtual robots? It could; however, engagement is not with the “real world” and interaction is not with “real artefacts”. What appears on the screen is, at the very least, someone’s conceptual interpretation of the real world. Here we use the term real, in the way a thirsty man would view a real glass of water compared to a virtual glass of water.

Berthelot and Salin (1994) found that lack of experience with meso and macro space restricted elementary school students’ ability to cope with micro space[2]. We have seen students confused by the forward command moving a virtual turtle upwards on the computer screen. Going forward across the floor is the same for student and robot. This is the core of Papert’s body syntonic idea: students can ‘play turtle’. They can project themselves out of their ego centric mind, ‘stand in the shoes’ of the robot and directly perceive the world from its perspective.

Exploring the idea of embodiment could lead to new understandings about educational robots. Consider the proposal that maths is not an objective science, but that it arose out of the various ‘image schema’ derived from repetitive embodied experiences (Lakoff and Nunez 2001). These pre-linguistic entities provide a source for linguistic metaphors like ‘source – path – goal’, which sympathises with the attributes of mobile educational robots. Although this theory is controversial (Gold 2001, Madden 2001) many maths educators believe the work has merit (Schiralli and

[2] Micro Space is the space accessible without moving: things on your desk – the computer screen. Meso Space is on a room level and Macro is wide open spaces - something you journey through.

Sinclair 2003, Tall 2003). We believe that further research into embodiment will aide our understanding of educational robotics.

Engagement

> *Through engagement Educational Robots can foster affirmative emotional states and social relationships that promote the creation of positive learning attitudes and environments, which improves the quality and depth of a student's learning experience.*

In 1992 Classic Roamer debuted in America when a Chicago teacher tried it with a second grade student who normally never engaged in school work. He decided to make Roamer turn "all the way around". So he programmed it to turn 8, which made it turn 8 degrees. He was shocked at this small movement. He was also captivated and went on to experiment with 1, 2 and 3 digit numbers. He subconsciously gained experience of equivalency and after 45 minutes discovered 360 was the "magic number". Thirty years of ad hoc observations of students using robots has shown this is not an uncommon example of the Engagement Principle. Educational robots and their activities have a propensity for capturing students' attention.

Engagement is a far richer and apposite concept than the ubiquitous, "makes learning fun". For example work done at CNEFI[3] in Paris used Roamer to change the attitude of an adolescent who had been 'brain damaged' in an auto accident (Sarralié 2002). The student had lost the ability to do simple arithmetic. He was very aggressive towards the teachers trying to restore his competency. Eventually, they gave him a Roamer activity, which necessitated him performing basic calculations. The robot task captured his attention, helped him realise his incapacitation and made him amenable to working with the teachers. It is fair to say that fun was not a part of this experience, but engagement was very much in evidence.

While many children seem to possess a natural fascination for robots, this is simply an advantageous starting point. What Bruner (1966) called the "will to learn" is a factor in sustaining engagement. Teachers can motivate students, help develop interests and trigger their curiosity (Hidi and Renninger 2006, Keller 2000 and Arnone and Small 2010). We claim that educational robotics provide skilful teachers with many ways of achieving these conditions.

Engagement involves the relationship a student forms with the robot. The classic ideas on transitional objects (Winnicot 1971, Leslie 1987) all relate to the cognitive processes of young children. Recent work has shown that:

1. Our relationship with physical objects also involves emotional and social experiences
2. The experience is not restricted to young children
3. Robots fall into a new category between inanimate object and living thing

Sherry Turkle cites evidence of children talking about their experience with Sony's robot dog Aibo as if it was one of their toys, yet they interact with it as though it were a real puppy (Turkle et al 2006). She classifies robots as "relational artefacts" and splits them into Rorschach and evocative types. Like the Rorschach test, aka ink blot tests, Turkle shows that student responses to the robots mirror underlying issues in their life and reveal their strategies for dealing with their concerns. She describes the evocative aspect as philosophical: something that makes people think (Turkle 2007). Papert's famous anecdote about his childhood experience with gears is an example of an evocative object at work (Papert 1980b). Not in the cognitive sense that the young Papert acquired a mental model that years later would help him understand equations; it was the wider philosophical effect that inspired his extraordinary career.

Engagement is about capturing a student's attention. In our Chicago anecdote the student became absorbed in the turning problem. We mentioned his subconscious experience of

[3] CNEFI - Centre National d'Etude et de Formation pour l'Enfance inadaptée" (CNEFI) - National Centre of Study and Training for children with special needs)

equivalence, something the curriculum did not require him to learn for another two years. This is an example of the "natural" learning of mathematics Papert so earnestly advocates. It is also an example of an intuition, which is an intrinsic element of the engagement principle. No one taught the Chicago student equivalence. Yet he happily "unthinkingly" used these concepts. This is the crux of a definition of intuition: immediate apprehension by the mind without reasoning (Allen 1990). This definition gives intuition a disreputable reputation. Some psychological studies make no distinction between intuition and guessing (Myers 2002). Comparative philosopher Hope Fitz combines Eastern and Western traditions to offer an alternative view. She sees intuition as an integral process of the mind, which is grounded in sub conscious memories and experiences. While it is linked to reason, the act of insight does not involve reason (Fitz 2001).

Insights are not accidents. Our subconscious accounts for most of our mental activity (Bragg et al, 2008). It is through attention that we build and access our intuitive knowledge. Poincare (1905) described the process in terms of creative mathematics. He deliberately immersed himself in anything relating to a problem. He relied on his intuitive skills to channel insights into his conscious mind. Discussing this idea Papert (1978 and 1980) suggests this process is not restricted to a mathematical elite. We go further and speculate that is not restricted to mathematics. It empathises with the ideas of expert knowledge discussed by Bransford et al (2000), the psychological studies on implicit learning (Goschke 1997) and perhaps the more sensational and speculative assertions made by advocates of accelerated learning (Jensen 1995). Our claim is that through engagement in robot activities students develop their intuitive understandings.

Sustainable Learning

> *Educational Robots can enhance learning in the longer term through the development of meta-cognition, life skills and learner self-knowledge.*

School is not just a place for the acquisition of knowledge and skills. It plays an important part in the personal development of students. The English National Curriculum (2010) specifically states the need to help students acquire communication skills, the ability to work with other people, to present ideas and to be confident.

The way we use educational robots automatically engages students in situations where the opportunity exists to develop these skills. For example, the Robotic Performing Arts Project (Catlin 2010) illustrates an opportunity for students to develop their cognitive, social, personal and emotional skills in an authentic learning situation.

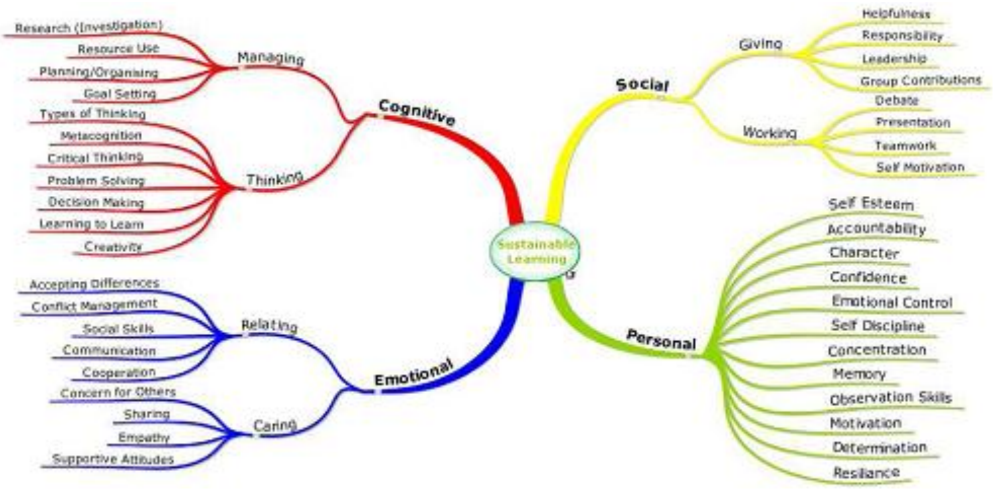


Figure 1: Mind map of typical sustainable learning criteria relevant to educational robots. - adapted from the Iowa 4H Program (2010).

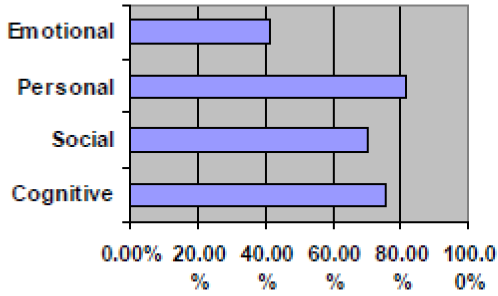


Figure 2 Involvement of sustainable criteria in a sample of 30 Classic Roamer activities.

Pedagogy

> *The science of learning underpins a wide range of methods available for using with appropriately designed educational robots to create effective learning scenarios.*

A central question in our project is what pedagogy justifies our belief that robots have a role in education? In the development of Logo, Papert synthesised ideas of Artificial Intelligence and the constructivist approach to education. That is, we understand the world by constructing mental models from our experiences. We assimilate or accommodate new experiences into our existing concepts or we accommodate them by modifying our existing ideas. Logo and Turtle robots provided experiences in a way that brought students into direct contact with some powerful and important ideas, particularly in mathematics.

Is this the only way we can or should view the educational process? We have already cited the potential insight we might gain from a review of Vygotsky and Activity Theory. While there are differences in these and other ideas, there are also many similarities. What clearly emerges is not some definitive truth about the way we learn but more of an orientation. This is starting to become known as the science of learning. Papert talks about the spirit of Logo and that life is not about “knowing the right answer”, but getting things to work. We need to adopt this pragmatic approach and let the science of learning inform and sometimes inspire our development of educational robots and their activities. Ultimately our judge of success is not whether we have a consistent developmental framework, but whether we can connect learning science and the technology with successful classroom practice.

Another aspect of pedagogy is a set of strategies that help us to create and analyse educational robotic activity. An analysis of work with Valiant’s Turtle and Classic Roamer has identified 28 different methods for using educational robots (Catlin 2010a).

Catalyst	Demonstration	Games	Presentations
Challenges	Design	Group Tasks	Problem Solving
Conceptualisation	Engagement	Inductive Thinking	Projects
Cooperation	Experimentation	Links	Provocateur
Creative	Experience	Modelling	Puzzles
Curriculum	Exploration	Memorisation	Relational Artefact
Deduction	Focussed	Pacifier	Transfer

Table 2: Pedagogical tools for educational robots

Most activities employ several strategies. For example a Roamer Activity called Robot Rally Race (Valiant 2009) starts with a challenge to find the fastest route, involves experimentation while the students try to find out how fast the robot travels over different terrains, and uses this statistical data in a focussed task to calculate the fastest way from start to finish. Table 2 is not a closed list. We expect to find other tools as the power of robots grows – for example Valiant’s work on robotics and storytelling is likely to yield some new approaches.

Curriculum and Assessment

Educational Robots can facilitate teaching, learning and assessment in traditional curriculum areas by supporting good teaching practice.

Most formal education takes place in schools. The “local” community decides what the students should learn and typically demand “proof” of achievement. While the curriculum and assessment methods vary between different communities there are many similarities. If educational robots are to make a significant impact they must be able to address the two items that concern teachers the most:

1. Teaching the curriculum
2. Assessment and testing

The Curriculum and Assessment Principle includes the phrase “good teaching practice”. How does this affect how a teacher teaches? Does it alter their traditional role as a dispenser of knowledge and what do educational robots have to contribute to this situation? These questions lead us to consider and develop another of Vygotsky’s innovative ideas: the Zone of Proximal Development (ZPD) defined as what the learner can do alone and what they can do with assistance (Vygotsky 1978). We predict the ZPD concept will develop to embrace technology in general and inteligent robots in particular. The characteristic of this model is that the teaching and learning experience will be more flexible than the Logo model of student teaching the robot or the teacher dispensing knowledge. It will be a dynamic model allowing any of the participants to be a teacher or a student.

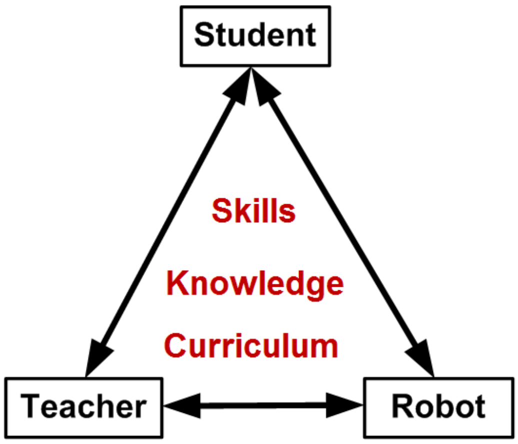


Figure 3: The dynamic relationship between teacher, student and robot shows that the learning and teaching interactions are bi-directional.

This proposition assumes that educational robots can be applied broadly across the curriculum. Turtle robots were tightly linked with mathematics and Roamer, Lego and other robots have made clear links with STEM (Science, Technology, Engineering and Maths) subjects in general. However, it is clear that robots are not restricted to these domains. In 1992 Harrow schools in the UK ran a district wide robotic art project. Students had to make Roamer into animated sculptures of fantastic insects. Perhaps more surprisingly is the use of robots in the study of moral and social values (Bers and Urrea 2000). Currently Valiant is developing a library of between 200 and 300 free and commercially available Roamer K-12 activities in all subjects Some of these, like the fantastic insects, are major projects; others like the Incy Wincy activity are completed in a lesson. The potential for activities far exceeds what a school could use in a balanced approach to teaching.

Formative assessment is a crucial part of effective learning environments particularly when it forms an unobtrusive element of an activity (Bransford et al 2000b, Black and Wiliam 2006). Feedback is embedded in robotic goal orientated action. Robots inherited this trait from Logo. Students propose an interim solution and then decide if it is satisfactory or whether they need to and/or how to make improvements. This makes formative assessment a natural part of this dynamic interactive process.

Personalisation

Educational robots personalise the learning experience to suit the individual needs of students across a range of subjects.

Ellwood Cubberley, a contemporary of John Dewey and Dean of Education at Stanford urged we view schools as factories in which the children were raw products to be shaped and fashioned to meet the demands of twentieth-century civilisation (Cubberley 1916). His rhetoric got worse: "*the business of schools was to build its pupils according to specifications laid down" and this required "continuous measurement of production to see that it is according to specification, the elimination of waste…*" Contrast this with the educational aims stated in the UN Charter for the child. It charges nations with developing the child's personality, talents and mental and physical abilities to their fullest potential (United Nations 2001). Robots support the UN child centred vision.

Table 3 Ways educational robots support the Personalisation Principle

1	Self Expression	Educational robots are tools that allow students to explore ideas and express their understanding in personal creative ways.
2	Flexible Use	Robots are adaptable to the needs of the teaching situation (see Practical Principle) and the needs of the individual student.
3	Differentiation	Robot activities find a natural level of difficulty. They support the constructionist principles and recognise that students build their own understandings in their own ways. They support struggling learners and challenge gifted students.
4	Learning Styles	Robots engage in multiple modal experiences: • Kinaesthetic • Visual • Spatial • Auditory • Tactile

These ideas are familiar to constructionists and have drawn their fair share of criticism. Let's deal with some the most common. Students setting goals does not lead to lower standards or the study of irrelevant topics. While students make the choices, good constructionist teachers "rig the deck". They motivate and encourage students. In fact once ignited students' imagination usually outstrips the activity objectives and pushes beyond expectations. This is not about achieving par; it is about the excellence beyond that. In a Classic Roamer task the students had to make a robot dog. Suddenly it needed "a wagging tail". How to do this was far beyond the teacher's skill and knowledge level, but not beyond her teaching skills. The students found a solution - a rubber tube that wagged furiously as Roamer wiggled its bum!

Equity

> *Educational robots support principles of equity of age, gender, ability, race, ethnicity, culture, social class, life style and political status.*

Before we can understand how robots help with equity we need to understand some of the issues involved. Equity means giving students an equal chance for a good education. Or does it mean giving them a fair chance? It turns out that equity is very hard to define, and how you define it affects how you deal with it (Ainscow et al 2006). Equal chance for example could mean making sure that each school has the same level of funding, resources, quality of teaching, etc. A fair chance would perhaps look at compensating for disadvantages.

Society can only determine a curriculum culturally entailed in favour of the mainstream of the community. For anyone who belongs to a cultural group that is not part of the mainstream, and whose sub group would produce a different curriculum, they have to make more effort to achieve academic success. There are those who argue such a curriculum represents a lingua franca for a society (Hirsch 1988). If minority students want to fully participate in main stream culture, they need to overcome cultural barriers. Though in practice mainstream-culture eventually changes because of input from minority participants (Lave and Wenger 1991).

Inequity arises from things like unequal funding (Kozol 2005), lack of qualified teachers, high quality materials, equipment and laboratories (Darling-Hammond 2005), overcrowded classrooms (Ferguson 1991) and poor quality teachers (Dreeben 1987).

Research and classroom practice show that minority pupils perform better when teaching is filtered through their own cultural experiences and frames of reference (Gay 2000). We claim:

1. Robots are tools that allow students to express themselves from their cultural perspective
2. The creative nature of robot activities makes them amenable to cultural modification

Because most societies have a tradition of artificial life (Simons 1986), robots have the potential to be culturally acceptable. Most cultures have developed the art of puppets and many technically advanced cultures created automaton of various types. Robots are another manifestation of this tendency. The mechanisms behind robots as transitional and relational objects make robots potentially tools through which children can express themselves. In a study of Huli children in Papua New Guinea, anthropologist Laurence Goldman (1998) concluded:

> In their “as-if vignettes”, pretenders are constructing, experiencing and implementing their models of the world, models that are always culturally encumbered and inflected.

his is the same mechanism Valiant has observed with students of indigenous cultures like the Maori, Australian Aborigines and some Native American peoples using Roamer. Students project their imagination into artefacts. With robots these imaginations come to life and enable students to express themselves in a way that reflects their heritage and situatedness in the modern world. They can connect their heritage with technology in their terms.

A robot teacher recently appeared in a Japanese school (Demetriou 2009). Saya, a humanoid invention of Professor Hiroshi Kobayashi, took the class register. Work at Carnegie Mellon with the robot Asimo is exploring and perfecting a robot that can read to students (Mutlu et. al. 2006). At a cost of $1M Asimo is a long way from classrooms, but it does imply that technology can “make up” for the poor quality of teachers. This argument is already well advanced with cognitive tutors (Woolf et al 2001, Koedinger, 2001). We do not subscribe to this view. Some very early research showed that technology together with teachers working with students got better results than students learning with teachers or technology alone (Dalton and Hannafin, 1988). This is very old research, but we suspect it still has validity. We believe that as robots become more adaptive and capable of providing sustained, uninterrupted interactions with the students, the teachers will be able to concentrate on working in ways that have greater impact on a student’s learning. This demands higher teaching skills not lower. It helps make teachers more effective.

Practical

> *Educational robots must meet the practical issues involved in organising and delivering education in both formal and informal learning situations.*

We often see approaches to education produce spectacular results in research or other controlled circumstances, followed by limited success or even outright roll-out failure. While we believe robots and ERA compliant activities will make a positive educational contribution, careful implementation and management is necessary if a school is to take full advantage of what robots offer. The Practical Principle considers this on two levels:

1. Systemic Implementation
2. Classroom Practicality

The Classic Roamer had a 95% penetration of UK Primary schools. This does not mean schools are getting the most out of them or using them regularly. Taking care of systemic changes issues will help people get the most out of robots. The following comments apply at the level of classroom, school, school district or even whole country.

Table 4 Elements of change adapted from Thousand and Villa (1995)											
Vision	+	Buy In	+	Skills	+	Resources	+	Plan	=	Change
	+	Buy In	+	Skills	+	Resources	+	Plan	=	Confusion
Vision	+		+	Skills	+	Resources	+	Plan	=	Resistance
Vision	+	Buy In	+		+	Resources	+	Plan	=	Fear
Vision	+	Buy In	+	Skills	+		+	Plan	=	Frustration
Vision	+	Buy In	+	Skills	+	Resources	+		=	Vacillation

Table 4: Summarises the elements required to make systemic change and what happens when an element is missing. Schools or districts wishing to integrate robots into delivering the curriculum need to address each of these issues. We propose the ERA Principles will help people develop an understanding and vision of how robots can be used.

At the moment most people think school robotics means students building robots. This type of activity is in fact a subset of the more general use of robots. Most teachers would not deem it practical to have to build the robot to engage in the Chicago Activity. For teachers to buy-in to using robots they must perceive their value outweighs the effort in dealing with the logistics and the preparation process. We are not trying to imply that there should be no applications that involve engaging in technical activity, but there needs to be activities that can be "ready to go in minutes" and do not require technical expertise. This does not mean the robots need to be crude. You do not need to be technically savvy to use sophisticated technology like a TV.

1	Individual work
2	Group work
3	Whole class learning
4	Home schooling
5	Learning support
6	Gifted programmes
7	SEN Interventions
8	Project work
9	Play
10	Games
11	Competitions
12	Collaborations

Table 5: One aspect of a robot's practicality is its ability to be used in many different teaching scenarios.

We do not feel that robotics will receive the kind of investment in skill training that has been expended on ICT (technology). Therefore it is essential that training is in-built into the activities: a sort of just-in-time and on-the-job approach. This was not feasible a few years ago but with the advances in online training and quality of open source platforms like Moodle it is now possible. Where teachers do go on training courses, online systems will act as support when they return to the hubbub of the classroom.

Budgets are always tight in schools – particularly if the school does not have a "vision". However, it can help if robots integrate with equipment schools already have.

So many times we have seen robot projects, particularly events like out of school competitions, generate huge amounts of enthusiasm. When the students go back to school that energy dissipates into the mundane. With proper planning teachers can use these events to boost the student's interest in regular lessons. Pupils cannot learn from using a robot alone. It is one element in a complex process. Well planned use of robots will ensure that the student has an opportunity to link their robotic experiences with formal aspects of the curriculum.

Conclusions

The ERA Principles represent the issues surrounding educational robotics. While this paper presents a quick survey of some of the pertinent arguments and hints at some of the evidence, it is clear that a lot of research is necessary to advance the subject. For many the research strictures dictated by NCLB's[4] positivistic approach to research is nonsense. However, there was a point to it. Many whims have been perpetuated onto schools. Our disagreement with NCLB lies with the rejection of the normative and interpretative research methodologies (Cohen and Mannion 1994). Perhaps this is not surprising because many of these techniques are ideal for studying the use of robots in schools. We also believe that what passes for longitudinal research is too short term. A three year research program would have missed the effects of Papert's gear experience. It is our intention to set up the e-Robot project which will aim to gather research information from an online community. The aim of this is to start to gather and collate the research necessary to develop the ERA Principles.

References

Adolphson K. (2005) *Robotics as a Context for Meaningful Mathematics,* Proceedings of the 27th annual meeting of the North American Chapter of the International Group for the Psychology of Mathematics Education. Online 09.04.10:
http://www.allacademic.com/meta/p_mla_apa_research_citation/0/2/4/7/4/p24749_index.html

Ainscow, M. Dyson, A. Kerr, K. (2006), *Equity in Education: Mapping the Territory.* The Centrefor Equity in Education, The University of Manchester. Online 22.03.10.
http://www.education.manchester.ac.uk/research/centres/cee/publications/Fileuploadmax10Mb,84135,en.pdf

Allen, R.E. Editor (1990) The Concise Oxford Dictionary. Oxford University Press

Arnone, M.P. and Small, R.V. (2009). *Curiosity, Interest and Engagement: A New Research Agenda*. White Paper, Center for Digital Literacy, Syracuse University, Syracuse, New York.
http://en.wikipedia.org/wiki/Human_robot_interaction - cite_ref-1

Bartneck, C. and Okada, M (2001). *Robotic User Interfaces.* Proceedings of the Human and Computer Conference: 130-140. Online 08.04.10
http://bartneck.de/publications/2001/roboticUserInterfaces/bartneckHC2001.pdf

Berthelot, R. and Salin, M. (1994). *Common spatial representations and their effects upon teaching and learning of space and geometry.* In J.P. da Ponte & J.F. Matos (Editors.), Proceedings of the Eighteenth Annual Conference of the International Group for the Psychology of Mathematics Education Vol. 2 (pp. 72-79)

Bers, U.M. and Urrea, C. (2000) *Technological Prayers: Parents, Children Exploring Robotics and Values, in Robot for Kids,* Druin, A and Hendler, J. (Editors) Morgan and Kaufmann Publishers

Betts, J.S. (1997) *I Thought I saw a Roaming Turtle, in Logo: A Retrospective.* Edited Maddox, C.D and LaMont Johnson, D. The Hayworth Press. Also published as Computers in Schools Vol 14 Numbers 1/2 (1997).

Black and Wiliam (2006) Inside the Black Box: Raising standards through classroom assessment. NFER Nelson. Online 10.04.10.
http://blog.discoveryeducation.com/assessment/files/2009/02/blackbox_article.pdf

Bragg, M, Calvert, G. Conway, M. and Papineau, D. (2008) *Neuroscience*, In Our Time, BBC Radio 4, Broadcast 10:00 am Thursday 13th November 2008. Online 20.04.10
http://www.bbc.co.uk/programmes/b00fbd26

Bransford, J. D. Brown, A. L. and Cocking, R. R. Editors (2000), *How People Learn: Brain, Mind, Experience and School*. (p57) National Research Council.

Bransford, J. D. Brown, A. L. and Cocking, R. R. Editors (2000a), *How People Learn: Brain, Mind, Experience and School*. (p31 -50) National Research Council.

[4] No Child Left Behind – President Bush's view on education which insisted that schools only used researched supported teaching methods.

Bransford, J. D. Brown, A. L. and Cocking, R. R. Editors (2000b), *How People Learn: Brain, Mind, Experience and School.* (p131-154) National Research Council.

Breazeal, C. L. (2004) http://www.amazon.co.uk/Designing-Sociable-Intelligent-Robotics-Autonomous/dp/0262524317/ref=sr_1_5?ie=UTF8&s=books&qid=1225484833&sr=8-5 *Designing Sociable Robots (Intelligent Robotics & Autonomous Agents)* MIT Press

Bruner. J.S. (1966) *Towards a Theory of Instruction.* Cambridge: Harvard.

Catlin, D. (2010) Robotics Performing Arts Project: An approach to STEM through cooperation not competition. Paper presented at the Constructionism 2010 Conference, Paris, France

Catlin, D. (2010a) 28 Strategies for Using Roamer Valiant Technology. Accessed 10.06.10 http://www.valiant-technology.com/uk/pages/archives.php

Crystal, David (1999) *The Penguin Dictionary of Language (2nd Edition),* Penguin Books.

Cubberly. E. P. (1917) Public School Administration pp337 -338, Houghton Mifflin / the Riverside Press

Cohen, L. and Manion, L. (1994) Research Methods in Education. Routledge.

Cuoco, A. (1990) Investigations in Algebra. MIT Press.

Dalton, D. W., and Hannafin, M. J. (1988) The Effects of Computer-Assisted and Traditional Mastery Methods on Computation Accuracy and Attitudes. Journal of Educational Research 82/1 (1988): 27-33.

Darling-Hammond, L. (2005) New Standards and Old Inequalities: School Reform and the Education of African American Students, p204, in Black Education Editor King, J. E. Published for the American Educational Research Association by Lawrence Erlbaum Associates.

Dautenhahn K. (2007) Methodology and Themes of Human-Robot Interaction: A Growing *Research Field*. International Journal of Advanced Robotic Systems 4(1) pp. 103-108 Online 23.04.10 https://uhra.herts.ac.uk/dspace/bitstream/2299/3814/1/902281.pdf

Davydov V. and Radzikhovskii L. (1985) *Vygotsky's Theory and the Activity Orientated Approach in Psychology* (in James Wertsch Editor Culture, Communication and Cognition: A Vygotskian Perspective) Cambridge University Press.

Demetriou, D. (2009) *Robot Teacher Conducts First Class in Tokyo School.* Daily Telegraph. Online 24.03.10 http://www.telegraph.co.uk/technology/5311151/Robot-teacher-conducts-firstclass-in-Tokyo-school.html Accessed 24.03.10.

Dourish P. (2003) *Where the Action is: The Foundations of Embodied Interactions*. MIT Press

Dreeben, R. (1987) *Closing the Divide: What teachers and administrators can do to help Black students reach their reading potential.* American Educator , 11(4) 28-35.

Engeström, Y. (1987). *Learning by expanding: An activity-theoretical approach to developmental research.* Orienta-Konsultit Oy Online 09.04.10 http://communication.ucsd.edu/MCA/Paper/Engestrom/expanding/toc.htm.

Engeström, Y. (Editor). (1999). *Perspectives on Activity Theory*. Cambridge University Press.

English National Curriculum (2010) Online 10.04.10 http://curriculum.qcda.gov.uk/new-primarycurriculum/ essentials-for-learning-and-life/index.aspx

Ferguson, R.F. (1991) *Paying for Public Education: New Evidence on how and why Money matters*. Harvard Journal on Legislation. 28(2), 465-498

Feurzeig, W. Papert, S. et. al. (1969), *Final Report Computer Systems for Teaching Complex Concepts*: Bolt Beranek and Newman Inc. Report No 1742 submitted to the Department of the Navy Office of Naval Research Washington, D.C. under Contract Nonr-4340(00)

Fitz, H.K. (2001) *Intuition: Its nature and uses in Human Experience.* Motilal Banaarsidass, Dehli.

Franklin S.A. (1997) *Autonomous Agents as Embodied AI*, Cybernetics and Systems 25(8) p499-520

Freeman, W.J. (2000) *How Brains Makes Up their Minds*, Phoenix

Gates, B (2007) *A Robot in Every Home.* Cover Story January 2007 Issue of Scientific American Magazine
Online 17.04.10 http://www.cs.virginia.edu/~robins/A_Robot_in_Every_Home.pdf

Gay, G. (2000) *Culturally Responsive Teaching: Theory, Research and Practice.* Teachers' College Press.

Gay, P. (1989) Tactile Turtle: explorations in space with visually impaired children and a floor Turtle. The British Journal of Visual Impairment V11, 1, 23-25.

Gold B. (2001) Read This! The MAA Online book review column Where Mathematics Comes by George Lakoff and Rafael E. Núñez
Online 10.04.10: http://www.maa.org/reviews/wheremath.html

Goldenberg, P.E. and Feurzeig, W. (1987) *Exploring Language with Logo.* MIT Press.

Goldman, L.R. (1998) Child's Play: Myth, Mimesis and Make Believe, Berg.

Goschke, T. (1997) *Implicit Learning and Unconscious Knowledge: Mental Representation, Computational Mechanisms and Brain Structures* (p247 -329)in Knowledge, Concept and Categories Lamberts, K. and Shanks, D. (Editors) Psychological Press.

Halliday, M. (1978) *Language as social semiotic.* London: Edward Arnold.

Kynigos, C. (1992) *The Turtle Metaphor as a Tool for Children's Geometry* (p97-126) in Learning Mathematics and Logo, Editors Noss, R. and Hoyles, C. MIT Press.

Hidi, S. & Renninger, A. (2006) *A four-phase model of interest development.* Educational Psychologist, 41, 111-127. Online 20.04.10
http://www.unco.edu/cebs/psychology/kevinpugh/motivation_project/resources/hidi_renninger06.pdf

Hirsch, E.D. (1990) *Cultural Literacy – What Every American Needs to Know*. Houghton and Mifflen

Iowa 4H Program (2010) Iowa State University.
Online 10.04.10 http://www.extension.iastate.edu/4h/explore/lifeskills/

Jensen, E. (1995) *The Learning Brain*, Turning Point Publishing.

Keller, J. (2000) *How to integrate learner motivation planning into lesson planning: The ARCS model approach.* Paper presented at VII Semanario, Santiago, Cuba, February, 2000.
Online 20.04.10
http://mailer.fsu.edu/~jkeller/Articles/Keller%202000%20ARCS%20Lesson%20Planning.pdf

Koedinger, K.R. (2001) *Cognitive Tutors as Modelling Tools and Instructional Models,* Chapter 5 in Smart Machines in Education, Forbus, K.D. and Feltovich, P.J. (Editors) AAAI Press/MIT Press

Kozol, J. (2005). *The Shame of the Nation: The Restoration of Apartheid Schooling in America.* New York: Crown Publishers.

Leslie, A.N. (1987) *Pretense and Representation: The Origins of "Theory of Mind".* Psychological Review, 1987, Vol 94, No 4,412-416 Online 20.04.10
http://faculty.weber.edu/eamsel/Research%20Groups/Belief%20Contravening%20Reasoning/Pretense/Leslie%20(1987).pdf

Lakoff, G. and Nunez, R. (2001) *Where Mathematics Comes from: How the Embodied Mind Brings Mathematics into Being*. Basic Books.

Lave, J. and Wenger, E. (1991) *Situated Learning: Legitimate peripheral participation.* Cambridge University Press.

Leontiev A.N. (1978) *Activity, Consciousness and Personality*. Prentice Hall.

Online: 09.04.09 http://www.marxists.org/archive/leontev/works/1978/index.htm

Madden J. (2001) *'Where mathematics comes from: How the embodied mind brings mathematics into being',* Notices of the AMS 48(10), 1182-1182.
Online 10.04.10http://64.17.226.144/notices/200110/rev-madden.pdf

Mills, R. Staines, J. and Tabberer, R. (1989) *Turtling without Tears:* A report of the DTI/MESU curriculum development project: the benefits associated with the use of floor turtles in the classroom. National Council for Educational Technology.

Mitchell, D. (1992) *An Investigation in to the use of Roamer Turtle in Developing the Spatial Awaerness of Visually Impaired Children*. MA Dissertation, University of Reading.

Muratana, H.R. and Varela F.J (1987) *The Tree of Knowledge: The Biological Roots of Human Understanding.* Shambhala.

Mutlu, B. Forlizzi, J. Hodgins, J. (2006) *A Storytelling Robot: Modeling and Evaluation of Human-like Gaze Behavior* p 518 – 523, 6th IEEE-RAS International Conference on Humanoid Robots. Online 10.04.10
http://bilgemutlu.com/wp-content/pubs/Mutlu_Humanoids06.pdf

Myers, D.G. (2002) *Intuition: Its powers and perils.* Yale University Press.

Nardi B.A. Editor (1996) *Context and Consciousness: Activity Theory and Human Computer Interaction*. MIT Press

NCTM (2000), Principles and Standards for School Mathematics, National Council for Teachers of Mathematics.

Papert, S. (1978 and 1980) *Poincare and the Mathematical Unconscious* in Asthetics in Science (Judith Wescheler Editor) MIT Press. Reprinted as Epilogues as Mathematical Unconscious in Mindstrorms, (1980) Basic Books

Papert, S. (1980) *Mindstorms: Children, Computers and Powerful Ideas*, (p11 and 160) Basic Books.

Papert, S. (1980a) *Mindstorms: Children, Computers and Powerful Ideas,* (p63, 68 and 205) Basic Books.

Papert, S. (1980b) *Mindstorms: Children, Computers and Powerful Ideas,* (vi - viii) Basic Books.

Papert, S. et al (1971 to 1981), *The Logo Memos*, MIT Artificial Intelligence Lab

Pfeifer, R. & Scheier, C. (1999). *Understanding Intelligence*. Cambridge, MA: MIT Press.

Sternberg, R. J. (1985). *Beyond IQ: A triarchic theory of human intelligence*. Cambridge University Press.

Sarralié, C. (2002) *The Roamer: an object for readapting in the case of adolescents with a cranial trauma,* Valiant Technology Ltd.
Online 10.04.10 http://www.valiant-technology.com/uk/pages/article_cranialtrauma.php

Schiralli, M. and Sinclair, N. (2003) *A Constructive Response to Where Mathematics Comes from,* Education Studies in Mathematics Vol 52, No 1 p79-91
Online 10.04.10. http://www.math.msu.edu/~nathsinc/papers/Response_to_WMCF.pdf

Simons, G. (1986) *Is man a robot?* John Wiley and Sons.

Sternberg, R.J. Kaufman J.C. and Grigorenko E.L. (2008) *Applied Intelligence.* Cambridge University Press.

Stonier, T.T. (1997) *Information and Meaning; An Evolutionary Perspective*. Springer

Syn, H.N. (1990) *Roamer and its Applications for Visually Impaired Children.* Microscope, MAPE

Tall, David (2003). *Using Technology to Support an Embodied Approach to Learning Concepts in Mathematics*. In L.M. Carvalho and L.C. Guimarães *História e Tecnologia no Ensino da Matemática*, vol. 1, pp. 1-28, Rio de Janeiro, Brasil.
Online 10.04.10 http://www.warwick.ac.uk/staff/David.Tall/pdfs/dot2003a-rio-plenary.pdf

Thousand, J.S. and Villa, R. A. (1995) *Managing Complex Change Towards Inclusive Schooling*. In Thousand, J.S. and Villa, R. A (1995) Creating an Inclusive School. Association for Supervision and Curriculum Development (ASCD).

Turkle, S. Taggart, W. Kidd, C.D. and Daste, O. (2006) *Relational Artifacts with children and elders: the complexities of cybercompaionship.* Connection Science Vol. 18 No. 4 p347-361
Online 27.03.10.
http://web.mit.edu/sturkle/www/pdfsforstwebpage/ST_Relational%20Artifacts.pdf

Turkle, S. Editor (2007) *Evocative Objects: Things we think with*. MIT Press.

United Nations (2001) United Nations Convention on the Rights of a Child Article 29 (1): The Aims of Education. General Comments CRC/GC/2001/1 Online 20.04.10
http://www.unhchr.ch/tbs/doc.nsf/(symbol)/CRC.GC.2001.1.En?OpenDocument

Valiant Technology (2009) Robot Rally Race. Roamer Activity Library. Accessed 10.12.10.
http://www.valiant-technology.com/uk/pages/activity_search.php

Vygotsky L.S. (1978) *Mind in Society: The Development of Higher Psychological Process* Edited by Cole, M. John-Steiner, V. Scribner, S. and Souberman, E. Harvard University Press.

Weaver, Constance L. (1991) *Young Children Learn Geometric Concepts Using Logo with a Screen Turtle and a Floor Turtle.* ERIC #:ED329430

Wertsch J.V. Editor (1985) *Culture, Communication and Cognition: A Vygotskian Perspective* Cambridge University Press

Wingorad, T. and Flores, F. (1986) *Understanding Computers and Cognition: A new foundation for Design.* Addison Wesley.

Winnicott, D.W. (1971) *Playiing and Reality*. Routledge.

Woolf, B.P., Beck, J., Eliot, C. and Stern, M. (2001) *Growth and Maturity of Intelligent Tutoring Systems: A Status Report*. Chapter 4 in Smart Machines in Education, Forbus, K.D. and Feltovich, P.J. (Editors) AAAI Press/MIT Press

Ziemke T. (2001) *Are Robots Embodied?* Proceedings of the 1st International Workshop on Epigenetic Robots p75-78 Online 10.04.10
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.64.9256&rep=rep1&type=pdf

Turtle, Art, TurtleArt

Paula Bontá, *paula@playfulinvention.com*
Playful Invention Company. Montreal, Quebec, Canada

Artemis Papert, *artemis@turtleart.org*
Independent artist. Aberdeen, Scotland

Brian Silverman, *brians@playfulinvention.com*
Playful Invention Company. Montreal, Quebec, Canada

Abstract
TurtleArt is a microworld for exploring art through turtle geometry. It is similar to the Logo programming language in that the main actor is a turtle that can draw. It is different from Logo and other constructionist systems in that it is focused on art. The vocabulary is small but provides rich control of colours, shades, pen widths, etc.

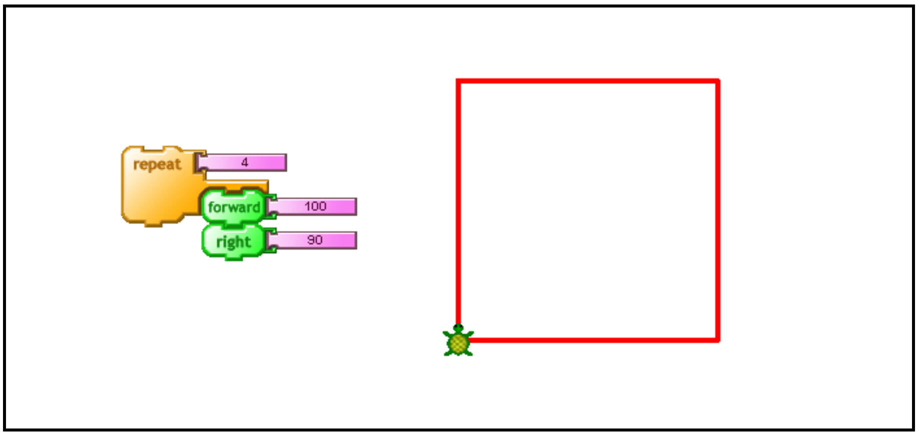
Introduction

TurtleArt does only one thing and tries to do it very simply and very well: bring geometry and art together through programming. Like the Logo programming language turtle geometry has a central role. However the main focus of TurtleArt is static artistic images.

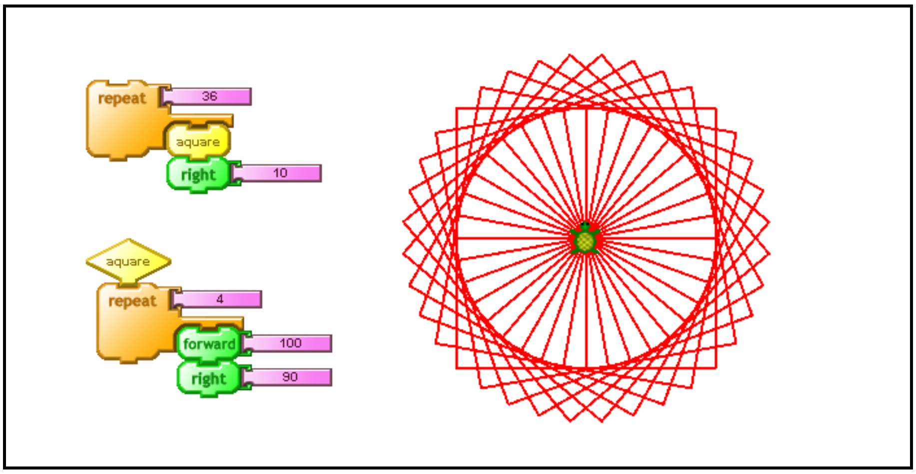
Programming in TurtleArt

TurtleArt follows the recent trend of programming by snapping together blocks. It borrows from the earliest versions of Logo by having a vocabulary centred around Turtle Geometry.

Here is the TurtleArt version of a square:

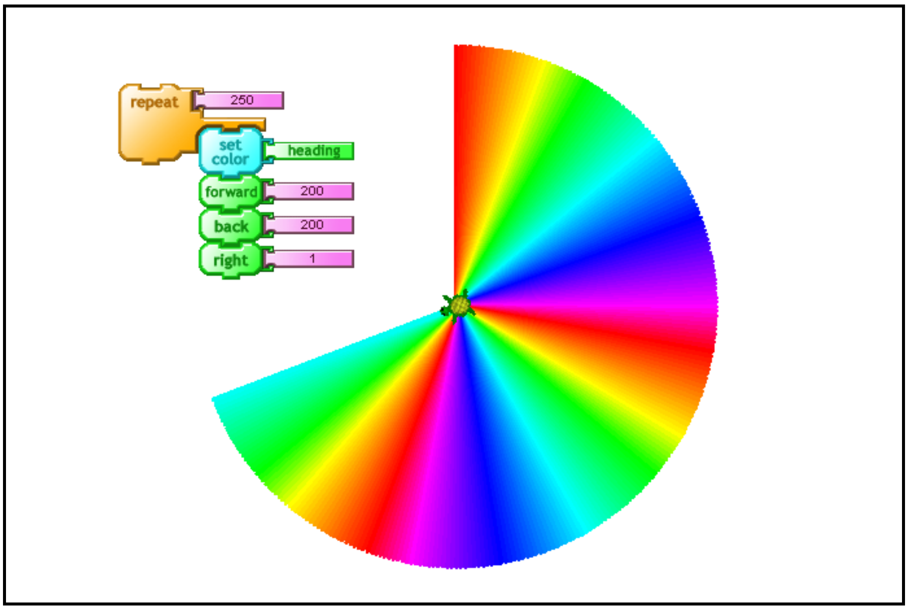


Procedures are defined by adding a "hat" to a stack of blocks. Once defined they can be used in other stacks:

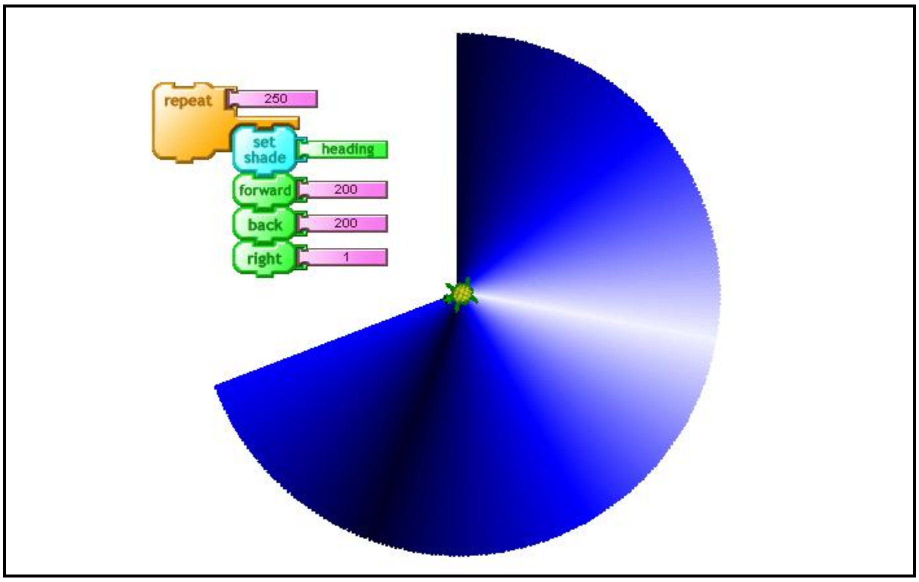


The turtle's drawing colour is controlled by setting a colour (hue) and a shade (lightness).

The colours:

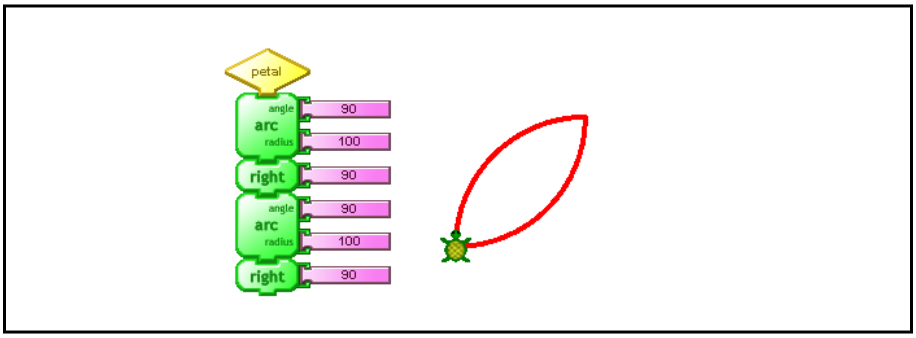


and the shades:

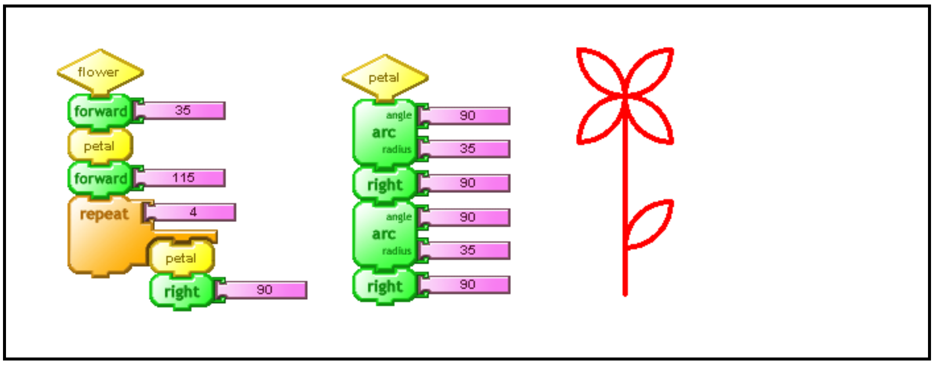


TurtleArt has an arc block. It allows drawing circles and arcs of a desired radius without knowing pi or the formula connecting a radius to a circumference.

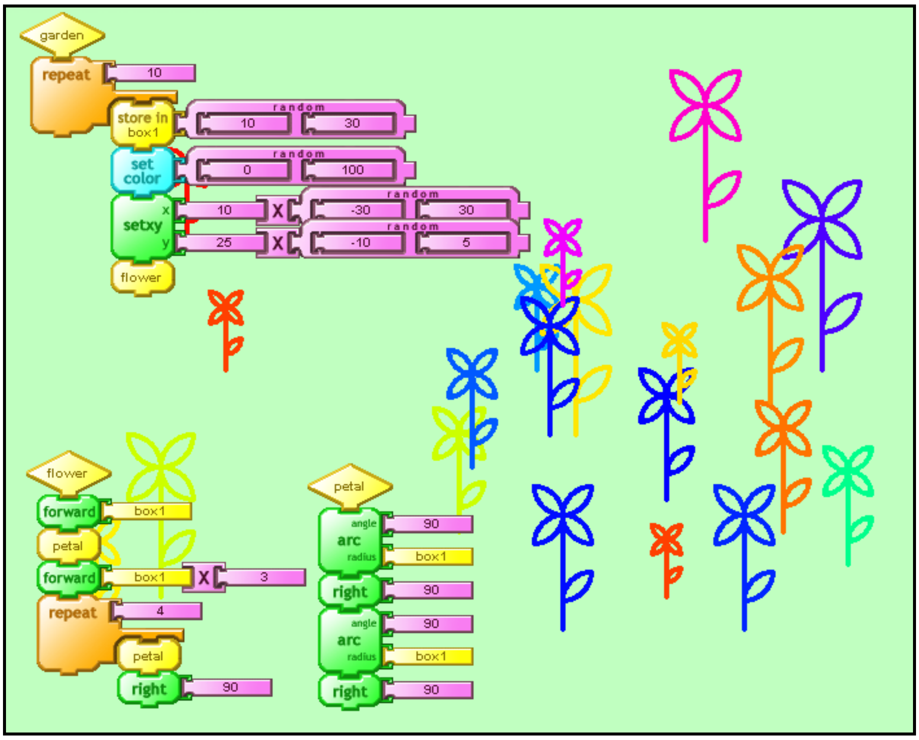
Two arcs put together make a petal:



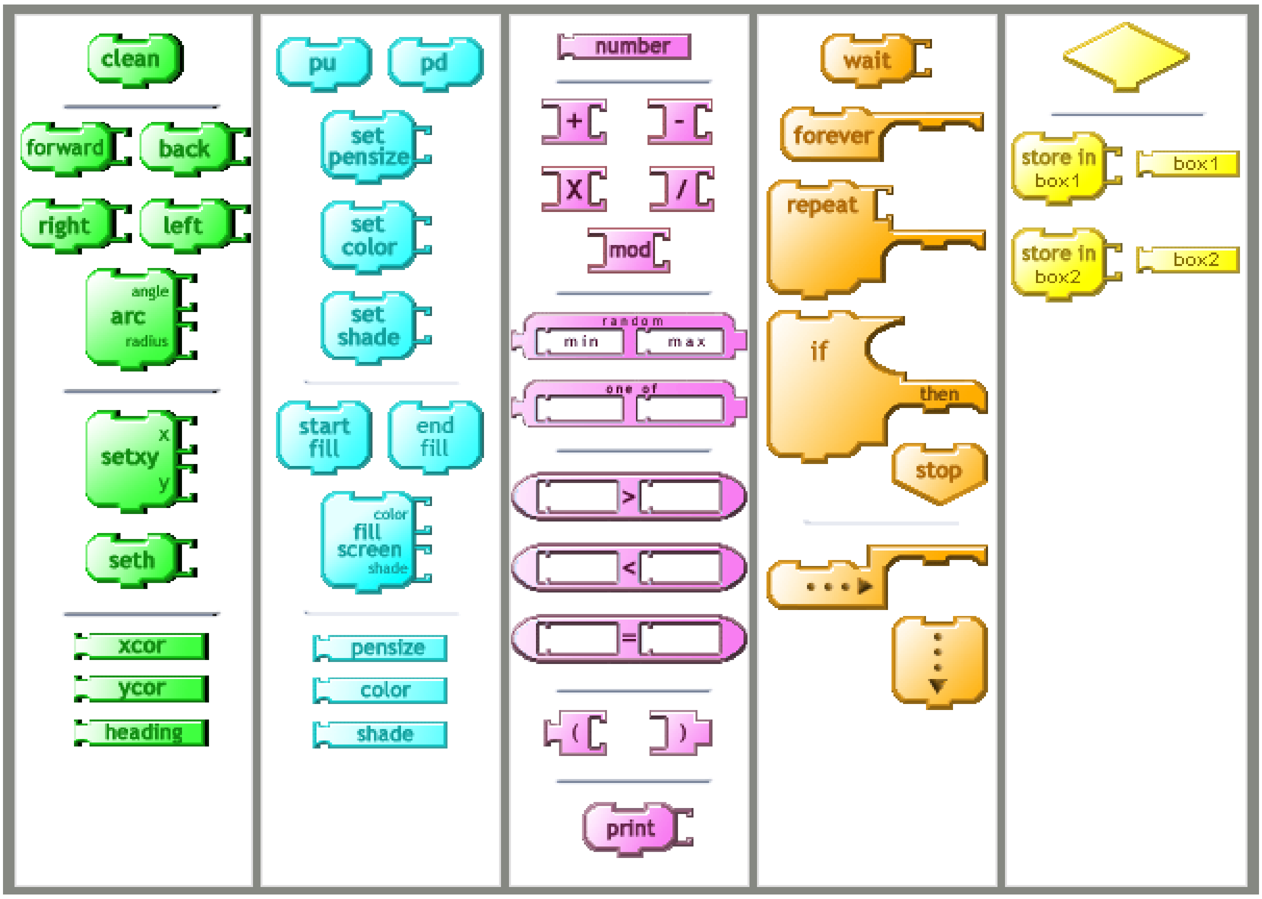
With several petals you can make a flower:



And with several flowers you can make a garden (Papert, 1971):



Here is the entire TurtleArt vocabulary:



Design of TurtleArt

Design is the art of making choices. Some are based on objective criteria. Others are more subjective and based on the preferences of the designers. We'd like to say something about some of our more salient choices.

In TurtleArt we tried to keep everything simple. Period. It is extremely minimalistic. We felt that if something isn't used often then it should not be there. A more common design approach is to feel that it's ok to have things that are useful only in rare cases. The thought is that if you don't need it just don't use it. We prefer otherwise. We agree with the Zen of Palm: "focus on what users do 80 percent of the time and try to ignore the other 20 percent" (PalmSource, Inc., 2003).

TurtleArt has only one object, the turtle. TurtleArt is also single threaded. Most recent versions of Logo have multiple objects of multiple kinds and support multiple threads. We consciously moved "backwards" in the name of simplicity. If the goal is static images, one turtle and one thread is good enough, and in the name of simplicity good enough turns out to be good enough.

The central exploration in TurtleArt is an artistic one. We feel that we can best empower that by a sharp focus on that exploration. There will be people who bump against the walls or the ceiling. Those people have other software choices available to them, some excellent. It is a non-goal for us to have TurtleArt also be good as a video game maker, a simulation language, or a system for multimedia presentations.

Sometimes the hardest part of a design is deciding what not to include. It's easy to invent new features. It's often even relatively easy to implement them. However, a minimalist design aesthetic suggests that things should only be included if they really fit. An example is that we included only a minimal set of mathematical operations. Trigonometric functions and square root are absent. Yes, this a limitation, but a conscious one.

Another important facet of the design is that we explicitly encourage programming. For example, there is no direct manipulation way of setting the turtle’s pen colour. Scratch and PicoBlocks have a SETCOLOR block. However they have a “point and click” way of setting a colour in addition to the programmatic way. In TurtleArt we skipped the “point and click” option in order to encourage a description that can be readily manipulated through programs.

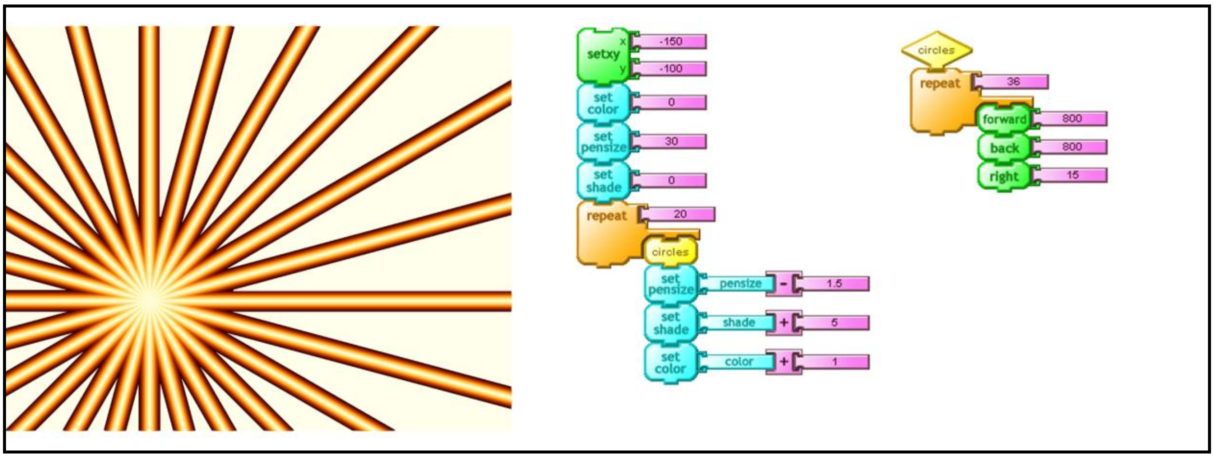
In the design of the user interface and of the blocks language we chose to emphasize visual stability. Nothing flashes, resizes, moves or animates automatically. We feel that this leads to a simpler, friendlier more tangible user interface. We recognise that this is opposite to the modern trend of having animations as a part of the interaction design and having elements that constantly update to provide real time feedback.

Overall, TurtleArt conforms to the design principles described by Resnick and Silverman (2005). Notable exceptions are that the "walls" aren't that wide and there aren't "many paths, many styles".

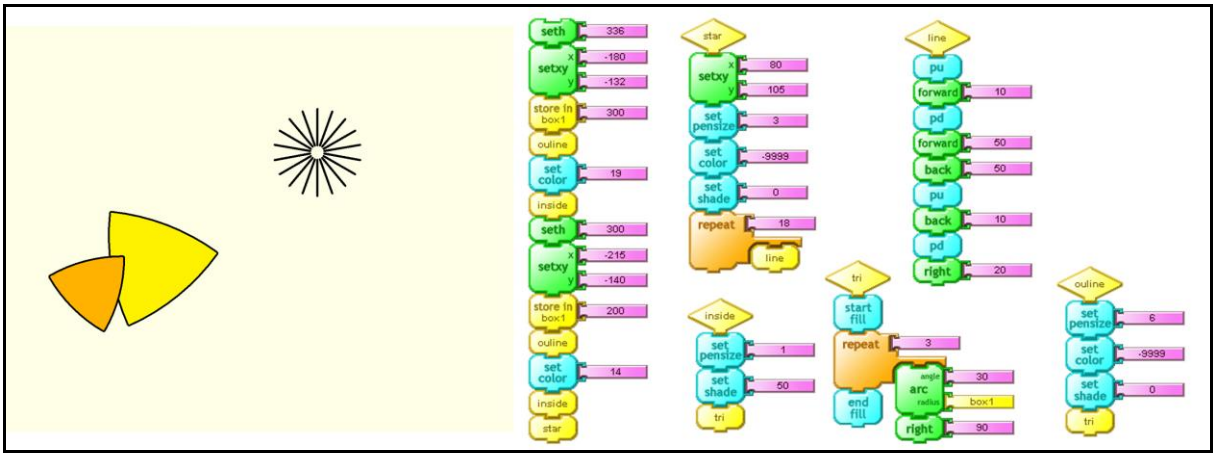
The Art of TurtleArt

We consider our work with TurtleArt as an exploration. It is an exploration of the interaction of formula, choice, and chance. Formula is very common in mathematical and computational art, e.g. fractals, Julia and Mandelbrodt sets, tilings, tessellations, etc. Choice is the result of an artist making a decision, picking a shape, colour, texture, composition of elements, etc. Chance is added to the mix using randomness, the result may be constrained but is never fully determined.

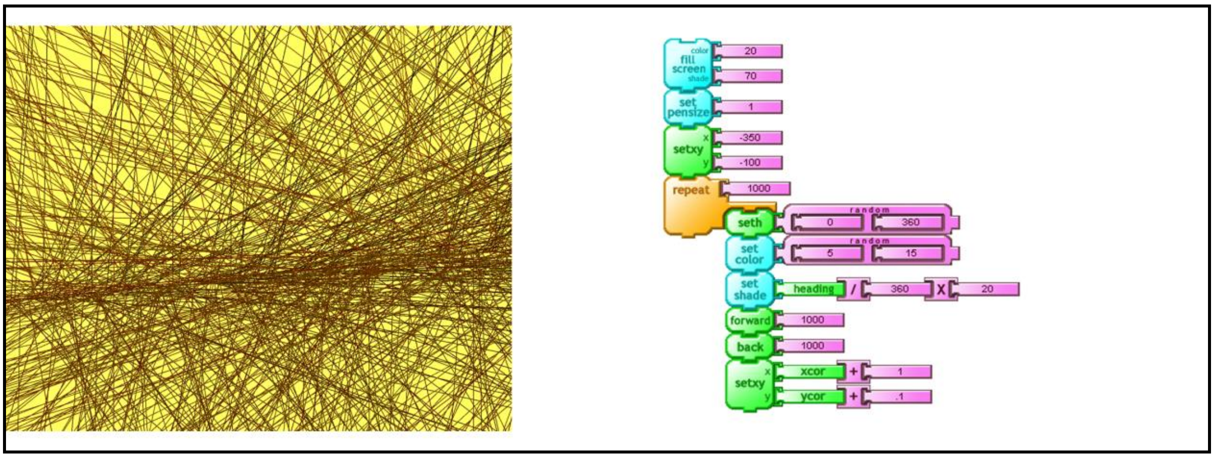
We’d like to present some images where formula, chance, and choice are mixed in varying proportions.



"Lion's Teeth" is based on a simple formula. Draw a line, turn a little, repeat. The basic pattern is repeated with different pen widths and colouring.



"Simple" has very little formula. Every element is placed by choice. Doing this often requires a fair bit of code.



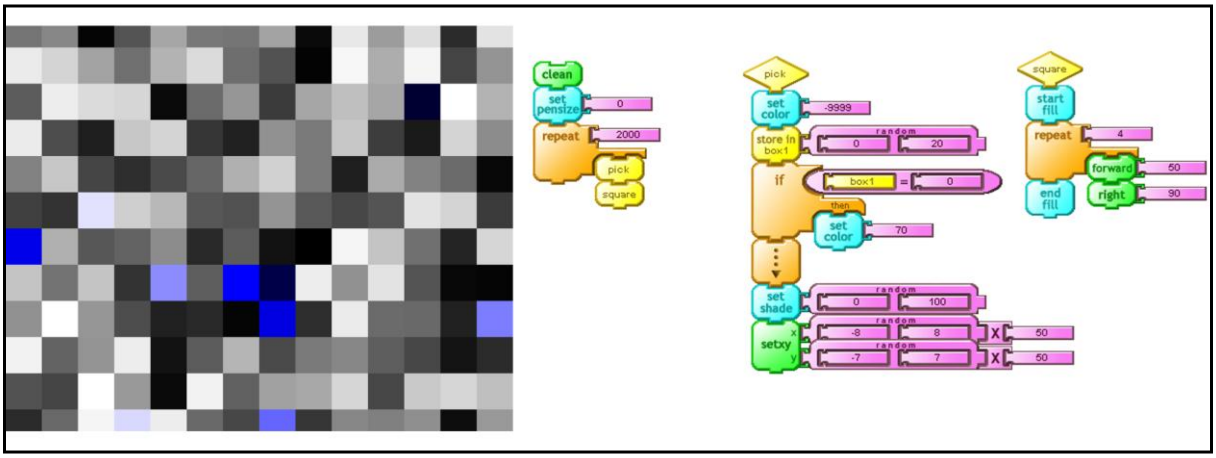
"Bridge to Nowhere" gets its texture from a controlled use of randomness. Run it again and it will be different in detail, though still very similar in texture.

Art through Programming

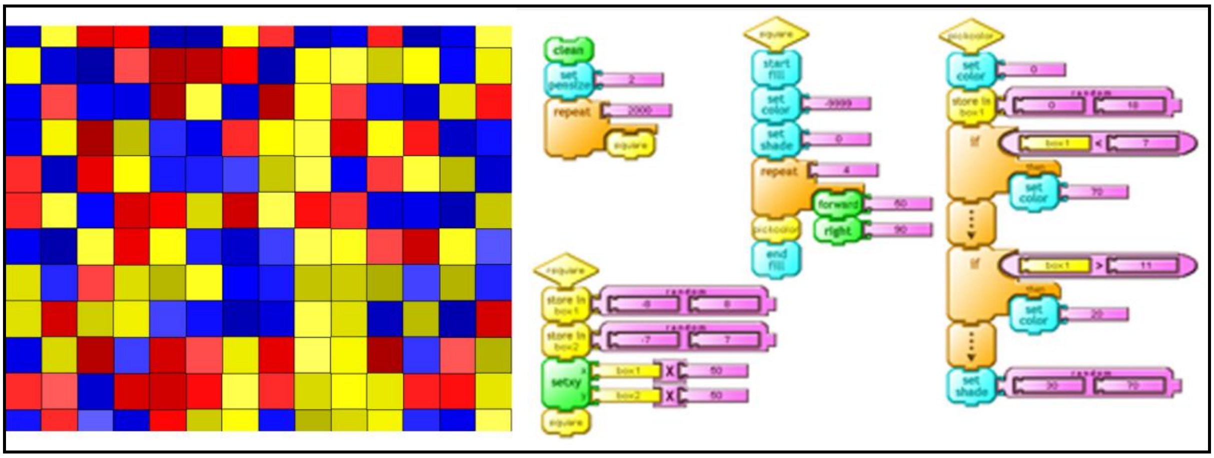
With TurtleArt we create images by creating programs. This has a collection of affordances. One obvious one is that a computer can draw thousands or even millions of strokes in a short amount of time. Less obvious is that programming allows for an evolution of images. We often create images that leave us with the feeling of being a good idea but needing further exploration. It isn't unusual for us to explore variants of a particular theme. We can modify parameters and code sequences in ways that change images to include what we like and leave aside what we don't. This process works exceptionally well with a programmatic description.

Here is an example of an evolution of images.

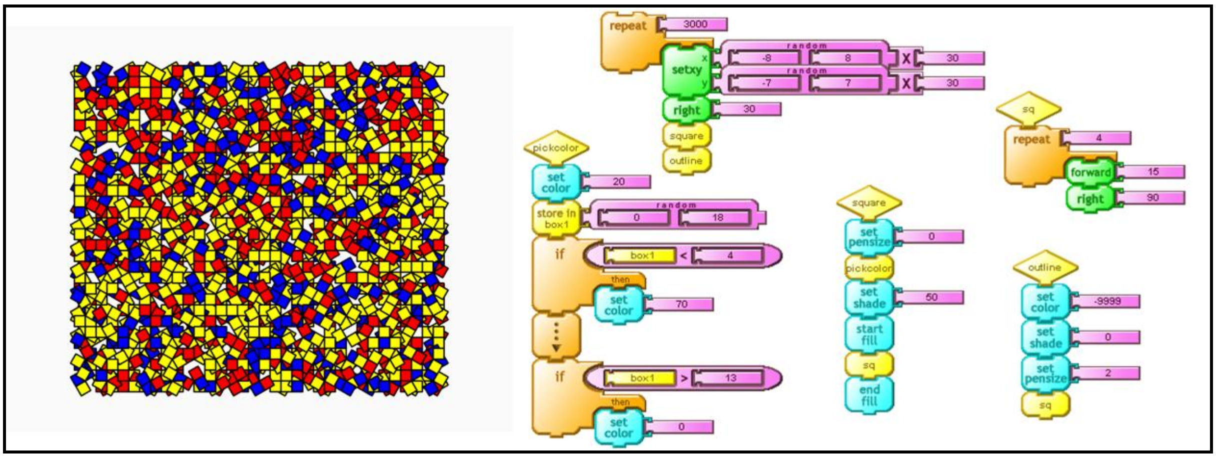
A first image looks like a low resolution picture...



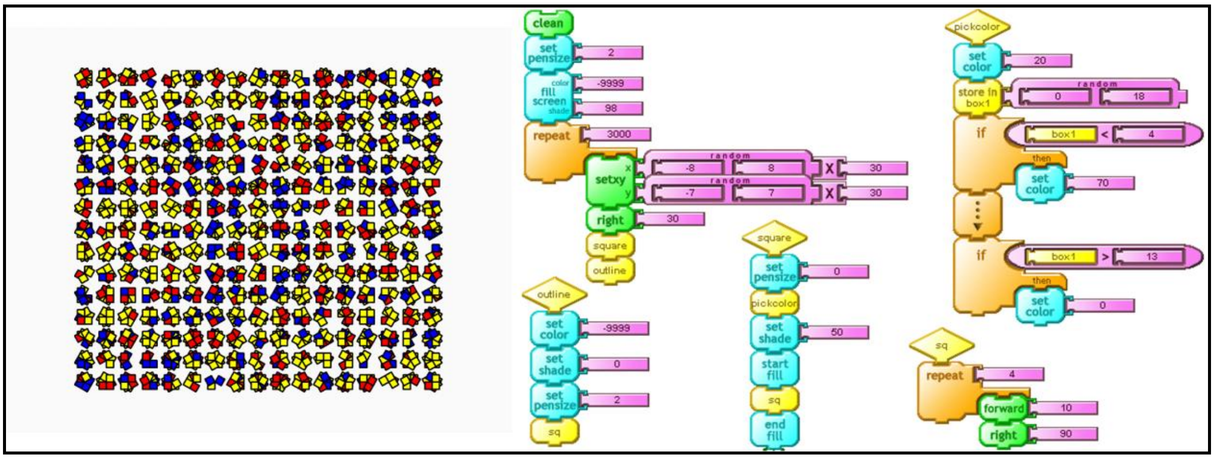
add some colour...



smaller squares, less lined up...



form little groups...



Conclusion

In 1971 Papert and Solomon described twenty things to do with a computer. These included making turtles, drawing pictures, playing music, building balancing robots and other activities. Back then only a few of those activities were possible, Turtle Geometry being one of them. It took a couple of decades before technology made it possible to do most of the other twenty things in a way that was accessible to children. A way of understanding the evolution of Logo and its successors is that Logo grew to be able to do more and more of the things described in that early paper. Along the way Turtle Geometry and drawing with the Turtle slipped to the back burner. We feel that this is unfortunate. TurtleArt tries to bring Turtle Geometry back and to do it in a way that empowers artistic expression.

For more information about TurtleArt, please visit http://www.turtleart.org

Acknowledgments

Mitchel Resnick, Natalie Rusk, Robbie Berg were centrally involved the design of PicoBlocks, essentially the same blocks language as the one in TurtleArt.

References

PalmSource, Inc. (2003) *Zen of Palm*, Document Number 3100-002-HW, June 13, 2003. http://www.accessdevnet.com/docs/zenofpalm.pdf (accessed March 2010).

Papert, S. (1971) *Teaching children to be mathematicians vs. teaching about mathematics.* Artificial Intelligence Memo No. 249 and Logo Memo No. 4.

Papert, S. & Solomon, C. (1971). *Twenty things to do with a computer*. Artificial Intelligence Memo No. 248 and Logo Memo No. 3.

Resnick, M. & Silverman, B. (2005) *Some Reflections on Designing Construction Kits for Kids*. Proceedings of Interaction Design and Children conference, Boulder, CO., USA.

Turtle, Art, TurtleArt

Paula Bontá, *paula@playfulinvention.com*
Playful Invention Company. Montreal, Quebec, Canada

Artemis Papert, *artemis@turtleart.org*
Independent artist. Aberdeen, Scotland

Brian Silverman, *brians@playfulinvention.com*
Playful Invention Company. Montreal, Quebec, Canada

Introductory description and overall goals

TurtleArt is a microworld for exploring art through turtle geometry. It brings programming and art together. The main focus of TurtleArt is static artistic images.

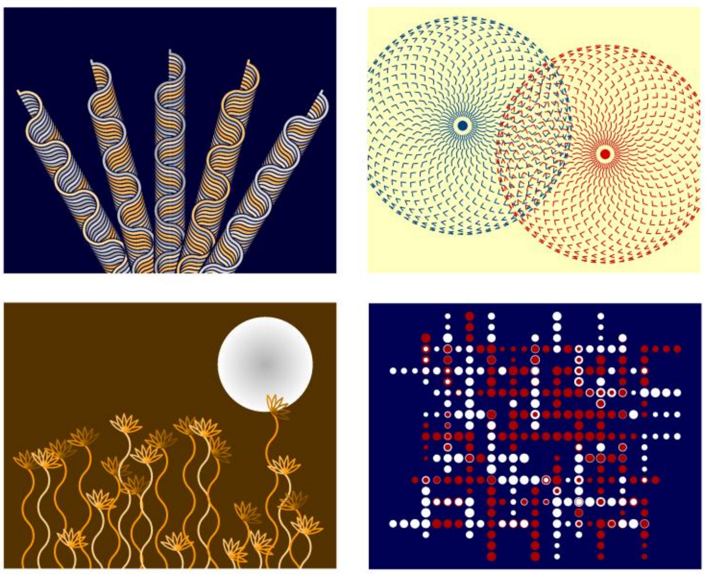
TurtleArt follows the recent trend of programming by snapping together blocks. It borrows from the earliest versions of Logo by having a vocabulary centred around Turtle Geometry.

Method

In this workshop you will get an introduction to programming in TurtleArt but mainly you will be given lots of hands on time to do your own exploration and create your own images.

Expected outcomes

Create artistic images via programming.



Keywords

Turtle, TurtleArt, art, Logo, programming, turtle geometry, constructionism.

A Visual Programming Language for Educational Robotics Based on Constructionist Ideas

Leonardo Cunha de Miranda, *professor@leonardocunha.com.br*
Institute of Computing, University of Campinas (UNICAMP), Campinas, Brazil

Fábio Ferrentini Sampaio, *ffs@nce.ufrj.br*
Electronic Computing Center and Informatics Postgraduate Program, Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil

José Antonio dos Santos Borges, *antonio2@nce.ufrj.br*
Electronic Computing Center, Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil

Abstract

Educational robotics is a playful and challenging activity that puts an emphasis on education during the creation of hardware and software based solutions. This work presents a visual programming environment built based on constructionist ideas. This computational solution allows students to program and to control electronic components such as LEDs, displays, motors, and light/temperature sensors, connected to different hardware of educational robotics using graphics elements of a visual programming language. One of the main distinguishing factors of this environment is the possibility to visually simulate the implemented logic on the screen before transferring it to the hardware.

Keywords

Visual Programming Environment; Educational Robotics; Hardware and Software; Computers in Education

Introduction

For many years researchers have been debating the different possibilities of using Information Technologies (IT) in educational settings. They seek to establish with these new technologies teaching and learning environments which are rich and motivating for learners.

Among the broad spectrum of ideas and proposals regarding computing artefacts, it is remarkable to note that most of the solutions presented explore predominantly the software. However, the demand for new hardware devices in education is growing, evidenced mainly by the efforts of the academic community to propose the inclusion of robotics for teaching purposes, supported by positive results presented in, e.g., Silva (2009), Alimisis *et al.* (2007), Demo and Marcianó (2007), Norte *et al.* (2005), Alimisis *et al.* (2005), Alves *et al.* (2005), Santos and Menezes (2005), Zilli (2004), Steffen (2002), Chella (2002), d'Abreu *et al.* (2002), and Kouznetsova *et al.* (2001).

Educational robotics is a challenging and fun activity that allows students to create solutions, whether they are composed of hardware or software, aimed at solving a particular problem. Most educational projects that use robotics in the classroom make use of constructionist approaches to support the teaching process, giving to the students a real possibility of knowledge construction while they develop their projects. In other words it is said that in so far as the students are deeply engaged in such activities they also have the opportunity to develop a more accurate understanding of scientific phenomena. Thus, robotics is a new educational tool that is available to the teacher, through which many theoretical concepts, sometimes difficult to understand, can be shown in practice, motivating both the teacher and primarily the student.

According to Zilli (2004), educational robotics can develop the following competencies: logical thinking, manipulative and aesthetic skills, integration of concepts learned in various areas of knowledge for development projects, representation and communication, work with research, problem solving through trial and error, the application of the theories in concrete activities, use of creativity in different situations, and critical thinking related to the topics covered by the project.

One may mention some advantages with the adoption of educational robotics kits on the market in general: 1) hardware and software products targeted to meet specific educational purposes; 2) flexibility to use them in different applications; 3) existence of technical user documentation, including in some cases, teaching materials; and 4) easier to own and operate by users unfamiliar with the technologies involved (electronics and computers).

The focus of this paper is to present a software solution, i.e., the ProgrameFácil[1] environment, implemented based on constructionist ideas during the Master's research of the first author. The choice for the development of ProgrameFácil was driven by the need for a visual programming language – some possible forms of a visual representation are presented and discussed in Chang (1987) – that has a user-friendly interface allowing users to program the hardware of an educational robotics kit, also developed by the same researchers team, called RoboFácil (Miranda, 2006). The process of programming with ProgrameFácil had to be intuitive, devoid of command-line interface, and without the need to know the electronic architecture of the hardware.

This article is organized as follows: the next section gives a general introduction to ProgrameFácil. Following this the reader will find four subsections giving more details about the interface of this environment, its main objects, how it operates, and some examples. Moreover,

[1] The name of this environment in Portuguese – ProgrameFácil – means easy programming.

in a later section, we present some constructionist ideas that provided the basis for this project. In the end, the authors draw some conclusions and present avenues for future work.

The ProgrameFácil Environment

ProgrameFácil is a Visual Programming Language (VPL) based on the manipulation of graphical icons that enables programming electronic and/or electromechanical devices, such as LEDs, displays, light and temperature sensors, and step motors making use of iconic symbols to encapsulate some traditional programming structures such as conditionality and repetition. Initially, the language was designed to control the RoboFácil's hardware, since its original version could only be programmed using assembly and/or C languages.

The conception, design and implementation of ProgrameFácil always took into account the need to create an intuitive environment in order to make it pleasant to use and an efficient resource to control the electronic and the RoboFácil's hardware. In this sense it was designed to present to the user an interactive environment consisting of two hypothetical worlds: the first one, called *MyWorld*, specifies the desired configuration of the hardware – e.g., LEDs, motors and sensors – and presents its behaviour while simulating a program developed by the user. The second, called *MyProgram*, is the place where the user constructs the program which will control the hardware detailed in *MyWorld*. Such worlds are presented in the ProgrameFácil environment through two windows. Figure 1a shows the *MyWorld*'s window and Figure 1b presents *MyProgram*'s window.

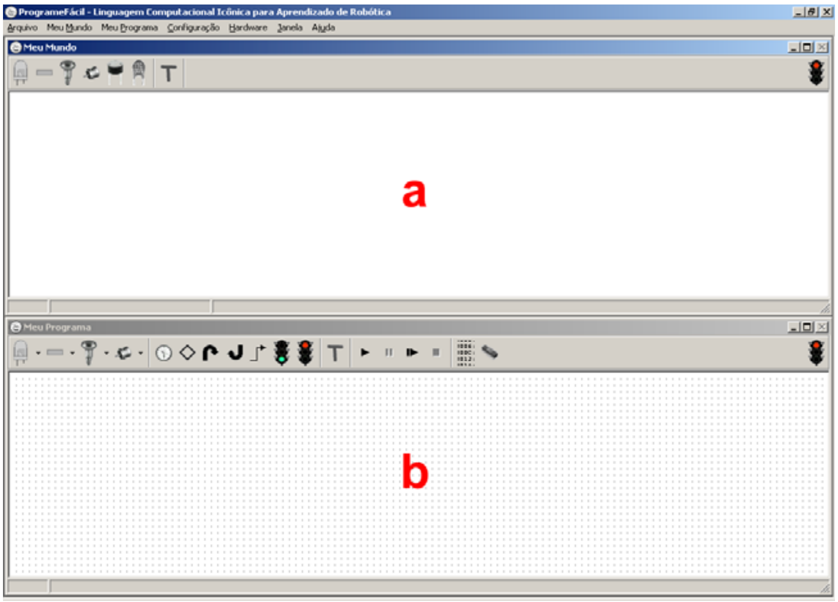


Figure 1. The ProgrameFácil environment presenting MyWorld (a) and MyProgram (b) windows

The adoption of explicit and different stages to draw and run/simulate models – mapped on the environment in different windows – aims to facilitate the investigation of the logic used in each program created by the user. When the user is satisfied with the behaviour of the program in his/her computer he/she can download it to RoboFácil's hardware. In this case a compiler is invoked to convert the icons[2] that make up the program into assembly macro-codes[3]. The

[2] Icons in the context of the ProgrameFácil environment can be defined as graphic symbols representing electronic devices or structures in programming languages.
[3] In this work the term assembly macro-code refers to the hexadecimal code set – bytecode – which represents the virtual assembly of RoboFácil's hardware.

interpreter in RoboFácil's firmware – discussed in detail in Miranda (2006) – in turn then converts programs written in the assembly macro-codes generated automatically by ProgrameFácil into statements that can be implemented in the hardware, such as activating/deactivating an LED, writing messages on the display or moving the step motor, among other possibilities.

MyWorld and *MyProgram* Windows

MyWorld is the window where the user can specify the hypothetical world that represents a hardware configuration of an educational robotic kit. The concrete objects that represent the electronic elements available for selection by the user are displayed in a toolbar of this window, except for the comment-object which applies only to allow the insertion of text in the template.

Figure 2 shows the toolbar of the *MyWorld* window with its hardware-objects: 1) LED, 2) display, 3) lamp, 4) motor, 5) light sensor, 6) temperature sensor, and 7) comments. The hardware-objects presented in *MyWorld* were abstracted from real life. Therefore, to associate them with a physical hardware, it is necessary to know their physical characteristics and actions allowed in reality.

Figure 2. MyWorld's window toolbar

The *MyProgram* window can be defined as the place where the user builds the program that will control the operation of existing objects in *MyWorld*. This process takes place by defining the actions and links between control structures such as conditional and repetition, using iconic symbols to represent them. These symbols are presented in the toolbar of the *MyProgram* window.

Figure 3 shows the toolbar of the *MyProgram* window with the following hardware-objects presented: 1) LED, 2) display, 3) lamp, and 4) motor. Also available here are the programming-objects: 5) timer, 6) IF conditional control structure, 7) looping-start, 8) looping-end, 9) line of programming, 10) program-start, 11) program-end, and 12) comments.

Figure 3. MyProgram's window toolbar

The goal is to make it possible to construct a programming logic between these elements, thus forming what is defined in the context of the ProgrameFácil as the Program of Model. To achieve this purpose the language was built with five rules:

- **1st:** Each object has one or no successor in the logical structure of programming;
- **2nd:** The program-end object (Object 11 of Figure 3) – which represents the end of the program – cannot have successors;
- **3rd:** The object IF (Object 6 of Figure 3) – which represents the IF conditional control structure – will have up to two successors;
- **4th:** Each object can have one or more predecessors in the logical structure of a program;
- **5th:** The program-start object (Object 10 of Figure 3) – which represents the beginning of the program – cannot have predecessors.

The inclusion of new hardware-objects in the *MyWorld* window and, as a consequence, in the *MyProgram* window depends their existence in the hardware.

When a program is simulated the executor starts with the programming-object program-start – represented by a green traffic light – and ends with the programming-object program-end (represented by a red traffic light).

Objects

An object in ProgrameFácil is a graphical representation, similar to an icon, that can be manipulated in both the *MyWorld* and *MyProgram* windows. Objects in ProgrameFácil were divided into three categories to better identify their purposes: hardware-objects, programming-objects, and supportive-objects.

The hardware-objects represent electronic components and were divided into two sub-categories: input-hardware-objects and output-hardware-objects. Programming-objects refer to common structures used in programming languages, e.g., loops, conditionals and delays. In its turn, the supportive-objects are intended exclusively to provide facilities and operational resources to the user, such as the possibility to include comments in the models.

The output-hardware-objects are presented in both windows, but have very different characteristics, e.g., the output-hardware-object LED in *MyWorld* has as property named *Color* to distinguish the color of the LED the user wants to work with. In *MyProgram* this same object has a property called *Set* used to set the LED to be on or off during the execution/simulation of the model.

In practice, to create a program using the ProgrameFácil VPL, you must perform three distinct steps: 1) select the hardware-objects to be used in *MyWorld* window, 2) include in *MyProgram* the representation of hardware-objects selected, and 3) create the flow of programming in the *MyProgram* window by selecting the appropriate icons in the *MyProgram*'s window toolbar.

Simulator

The environment provides a compiler which converts the programs constructed with the ProgrameFácil VPL into assembly macro-codes that can be executed by RoboFácil's hardware. The translation is performed by matching assembly macro-codes for each hardware-object and/or programming-object presented in the program created by the user, which then will be understood by the parser component of the RoboFácil's firmware.

The process of compiling a model is done with a single mouse click on the button corresponding to this functionality. Upon completion of this process the compiler can provide a window stating the result: build successful or performed with compilation errors.

The system also provides two traffic light icons attached in both windows (positioned to the upper right corner). Their function is to indicate the status of a model: under development (red), paused (yellow) or simulating (green).

In *MyWorld*, during a simulation, the objects will change their properties according to the program being executed. As the simulation proceeds, it is also possible to see in *MyProgram* an execution pointer – red rectangle – surrounding the command that is being interpreted. This feature is especially useful for debugging purposes.

Examples

Some examples are given in order to show the implemented features and the possible use of this environment.

Figure 4 shows a simple model constructed in ProgrameFácil. This first model has only one red LED (E1). When running this model in ProgrameFácil or in RoboFácil's hardware, E1 will blink every each second.

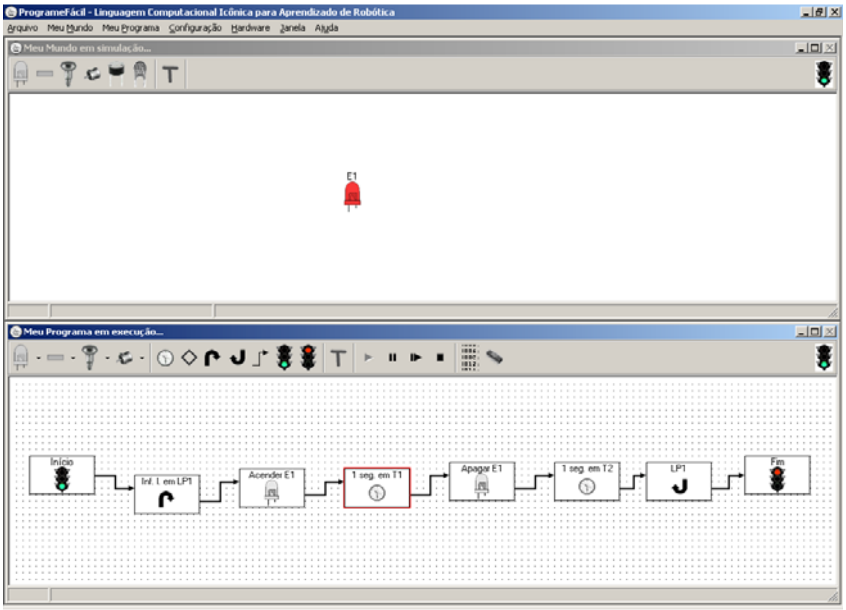


Figure 4. Example of a LED blinking

Figure 5 presents another example with three lamps – L1 (yellow), L2 (green), and L3 (red) – and also a light sensor (SL1). When running this model in ProgrameFácil or in RoboFácil's hardware, L1 will turn on and then SL1 will be tested. In the case it is sensing light around it, L2 will be turned on, otherwise L3 will be on. Note that as L1 is near SL1 (so it is sensing light), the L2 was turned on when this model was simulated in ProgrameFácil.

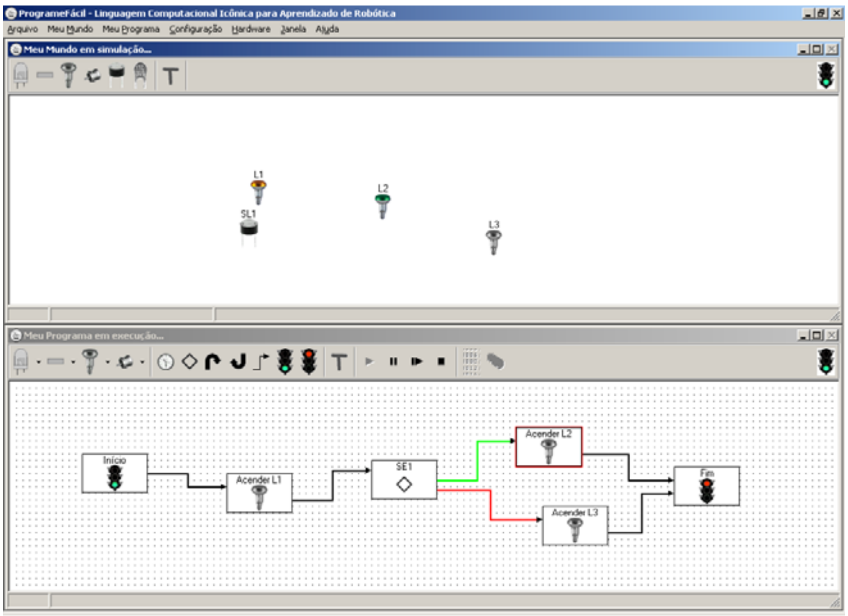


Figure 5. Example with lamps and light sensor

Figure 6 shows a third example of a model constructed with ProgrameFácil. This model has two green LEDs (E1 and E2), and also a light sensor (SL1) and an alphanumeric display with green backlight (D1). The model exemplified here aims to turn on E2, if SL1 is under the "natural" light. When running this model in ProgrameFácil, the message "No light" will appear on D1 for five seconds whenever SL1 is not under "natural" light (this was the condition when this model was simulated).

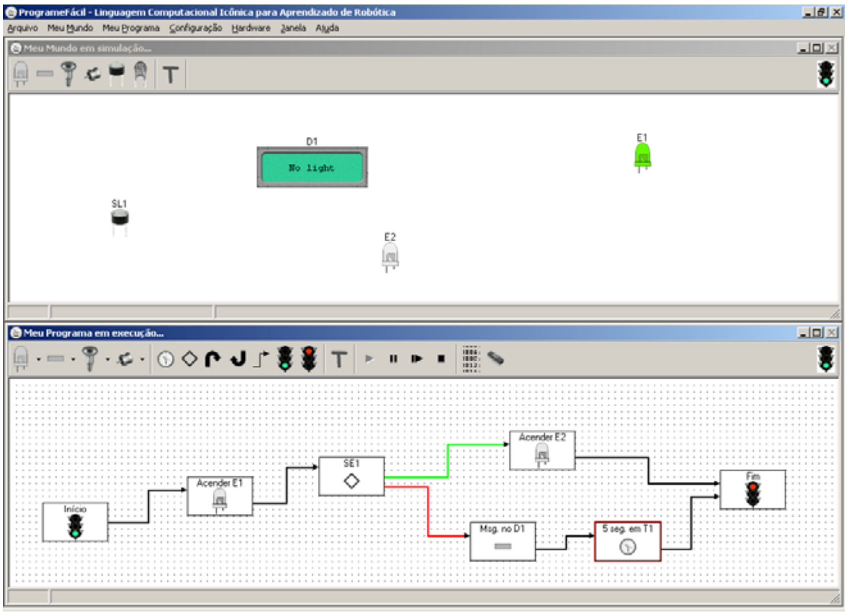


Figure 6. Example with LEDs, light sensor, and display

The last example presents a more complex situation (Figure 7). This model has one yellow lamp (L1), one temperature sensor (ST1), two motors (M1 and M2), and two displays with blue backlight (D1 and D2). In this example, when the ST1 sensor is below its trigger level, i.e., when in ProgrameFácil's model L1 is far from ST1, M1 and M2 will be switched on and this fact will be reported for user through D1 and D2.

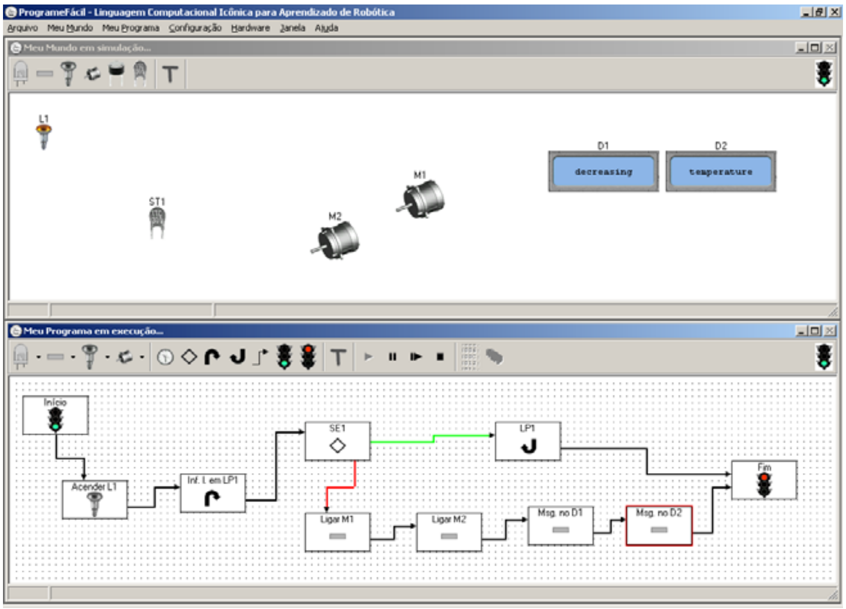


Figure 7. Example with lamp, temperature sensor, motors, and displays

Constructionist Ideas and ProgrameFácil

The ProgrameFácil environment was conceived in line with the ideas of Papert and LOGO (Papert, 1980, 1993). We believe that its graphical interface together with its iconic language allow students to learn to program robotic devices in a more enjoyable way, since before the existence of this environment the process of programming RoboFácil's hardware was done via assembly and C languages. Moreover, the features that were incorporated into the system, as discussed earlier, enable students to focus on the process of *exploring possible solutions* instead of investing his/her time to program a certain solution.

The process of constructing a solution is made by the students through the manipulation of concrete-abstract objects in the environment. We believe that they are concrete in the sense that they resemble very much the electronics presented in the hardware and they also show on the screen most of the real properties of such hardware.

The possibility of performing a simulation on the screen – in *MyWord* window – allows the students to easily – and quickly – confront his/her ideas initially thought to solve a certain problem with the output of the – hypothetical – hardware. This feature gives them the possibility to visualize and to reflect on how each part of its solution works before downloading it to the – real – hardware.

Conclusions

Educational robotics, although not new, is not yet widespread in Brazilian schools (and possibly the same situation happens in many developing countries). A possible reason for this is the relatively high cost of the necessary hardware for many educational institutions.

The VPL presented here combines theoretical knowledge from different areas such as education, computer science and engineering to provide a feasible alternative for the high cost of robotics toolkits available in the market. Although the ProgrameFácil environment was originally designed to be used with RoboFácil's hardware it can be easily integrated with different hardware designs such as GoGo Board (2006) and Lego Mindstorms (2006).

An important feature of the ProgrameFácil environment is its built-in simulation tool. Among other advantages, this feature minimizes the need to have a specific hardware for each group of students, thereby helping to reduce costs in setting up workshops and laboratories to work with robotics. However, it is important to note that you must have a sufficient number of robotics toolkits for all students participate in the process of constructing and testing their projects.

Experiments initially conducted with undergraduate students in computer science, demonstrated the potential application of the solutions described here. These tests have allowed our research team to obtain a more practical view of the use of digital artefacts in real-world educational scenarios, giving us feedback on some improvements to be implemented.

As future work we propose to carry out pilot studies to develop thoughtful pedagogical proposals that could explore the environment presented in this work. We believe these proposals, as mentioned before, have to be anchored in constructivist ideas in other to explore better the student's potential to work with thought-provoking problems.

References

Alimisis, D., Karatrantou, A. and Tachos, N. (2005) *Technical School Students Design and Develop Robotic Gear-Based Constructions for the Transmission of Motion*. In Proceedings of the 10th European Logo Conference (EuroLogo 2005). pp. 76-86.

Alimisis, D., Moro, M., Arlegui, J., Pina, A., Frangou, S. and Papanikolaou, K. (2007) *Robotics & Constructivism in Education: The TERECoP Project*. In Proceedings of the 11th European Logo Conference (EuroLogo 2007). pp. 1-11.

Alves, A.C., Blikstein, P. and Lopes, R.D. (2005) *Robótica na Periferia? Uso de Tecnologias Digitais na Rede Pública de São Paulo como Ferramentas de Expressão e Inclusão*. In Anais do XXV Congresso da Sociedade Brasileira de Computação (CSBC 2005). pp. 2594-2602.

Chang, S.K. (1987) *Visual Languages: A Tutorial and Survey*. IEEE Software, 4(1), 29-39.

Chella, M.T. (2002) *Ambiente de Robótica para Aplicações Educacionais com SuperLogo*. Master’s thesis, University of Campinas.

d’Abreu, J.V.V., Gonçalves, L.M.G., Garcia, M.F. and Garcia, L.T.S. (2002) *Uma Abordagem Prático-Pedagógica para o Ensino de Robótica em Ciência e Engenharia de Computação*. In Anais do XIII Simpósio Brasileiro de Informática na Educação (SBIE 2002). pp. 428-439.

Demo, G.B. and Marcianó, G. (2007) *Contributing to the Development of Linguistic and Logical Abilities through Robotics*. In Proceedings of the 11th European Logo Conference (EuroLogo 2007). pp. 1-10.

GoGo Board (2006) Available at: http://www.gogoboard.org.

Kouznetsova, I., Gorlitskaya, S. and Sidorov, V. (2001) *LOGO&LEGO for Informatical Training of Children with Different Educational Levels*. In Proceedings of the 8th European Logo Conference (EuroLogo 2001). pp. 137-142.

Lego Mindstorms (2006) Available at: http://www.lego.com/eng/education/mindstorms.

Miranda, L.C. (2006) *RoboFácil: Especificação e Implementação de Artefatos de Hardware e Software de Baixo Custo para um Kit de Robótica Educacional*. Master’s thesis, Federal University of Rio de Janeiro.

Norte, S., Castilho, N., Condado, P.A. and Lobo, F.G. (2005) *GoGoBoard and Logo Programming for Helping People with Disabilities*. In Proceedings of the 10th European Logo Conference (EuroLogo 2005). pp. 171-178.

Papert, S. (1980) *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, New York.

Papert, S. (1993) *The Children’s Machine: Rethinking School in the Age of the Computer*. Basic Books, New York.

Santos, C.F. and Menezes, C.S. (2005) *A Aprendizagem da Física no Ensino Fundamental em um Ambiente de Robótica Educacional*. In Anais do XXV Congresso da Sociedade Brasileira de Computação (CSBC 2005). pp. 2746-2753.

Silva, A.F. (2009) *RoboEduc: Uma Metodologia de Aprendizado com Robótica Educacional*. Ph.D. thesis, Federal University of Rio Grande do Norte.

Steffen, H.H. (2002) *Robótica Pedagógica na Educação: Um Recurso de Comunicação, Regulagem e Cognição*. Master’s thesis, University of São Paulo.

Zilli, S.R. (2004) *A Robótica Educacional no Ensino Fundamental: Perspectivas e Prática*. Master’s thesis, Federal University of Santa Catarina.

Take up the challenge – reflection on POLLOGIA competition

Agnieszka Borowiecka, *agn_b@oeiizk.waw.pl*

Maciej Borowiecki, *maciek_b@oeiizk.waw.pl*

Wanda Jochemczyk, *wanda@oeiizk.waw.pl*

Katarzyna Olędzka, *katarzyna@oeiizk.waw.pl*

Agnieszka Samulska, *samag@oeiizk.waw.pl*
All from Computer Assisted Education and Information Technology Centre (OEIiZK), Warsaw

Abstract

In the school year 2008/2009 we started organising POLLOGIA - national competition on programming in Logo for lower secondary school students (gymnasium). It was done in cooperation with Microsoft's educational programme "Partners in Learning". The POLLOGIA competition is based on the experience gained during organisation of local competitions for Mazovia District – Logia (for lower secondary school students) and miniLogia (for primary school students).The tasks on this competition are from turtle graphics, operations on words and lists. Competition has three stages. In the article we describe our experience and difficulties connected with organising POLLOGIA, students' problems and organisational issues.

Figure 1. POLLOGIA competition logo

Keywords

Logo, Imagine Logo, programming in Logo, competition, lower secondary school, gymnasium

Difficult beginnings

In the school year 2008/2009 we started organising **POLLOGIA** - national competition on programming in Logo for lower secondary school students (gymnasium). It was done in cooperation with Microsoft's educational programme "Partners in Learning". This long-term educational programme empowers students and teachers to realize their full potential, by partnering with education and government leaders through providing education resources – tools, programmes, and practices.

Figure 2. LOGO Web portal in Microsoft's educational platform "Partners in Learning"[1]

Figure 1. LOGO Web portal in Microsoft's educational platform "Partners in Learning"

The POLLOGIA competition is based on the experience gained during organising of local competitions for Mazovia District – Logia (for lower secondary school students) and miniLogia (for primary school students)[2]. Prepare a competition for the whole country requires a different organisational and pedagogical approach. Following the rules of Eight Big Ideas Behind the Constructionist Learning Lab[3], especially the one about hard fun, we started POLLOGIA competition.

Firstly, we realised that the competition required some promotional actions to be taken to popularise the idea. We prepared leaflets and posters to send them to every lower secondary school in Poland (about 6500). The magazine "Teachers' voice" took the patronage under the competition.

Secondly, there were some obstacles of organisational nature. Whereas first stage of the competition did not need access to school laboratories, in the second one, the organisers had to secure school laboratories on a specific day and strict hours. The third stage was organised in Warsaw, so we had to think about accommodation for 30 students and their guardians. This was connected with some financial problems, since students were from various regions of Poland.

We also faced problems in the field of the level of students' preparation. Logo language is not taught in every secondary school, its popularity differs in various regions of Poland.

[1] LOGO Web portal in Microsoft's educational platform "Partners in Learning" http://www.pdp.edu.pl/logo

[2] Borowiecka, A. and other (2008) – Konkursy Informatyczne LOGIA i miniLOGIA, OEIiZK

[3] "The third big idea is hard fun. We learn best and we work best if we enjoy what we are doing. But fun and enjoying doesn't mean "easy". The best fun is hard fun. Our sports heroes work very hard at getting better at their sports. The most successful Carpenter enjoys doing carpentry. The successful businessman enjoys working hard at ma king deals." Gary Stager, Papertian Constructionism and the Design of Productive Contexts for Learning

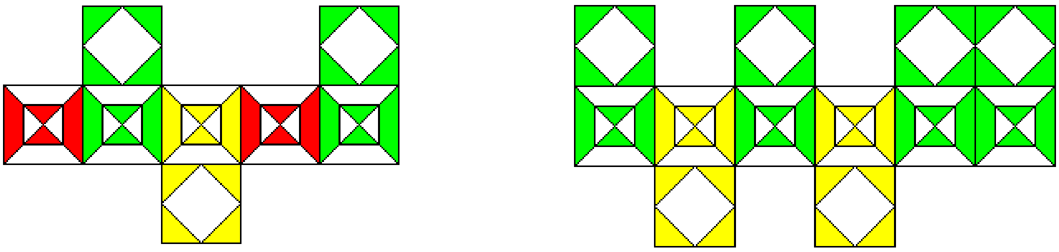
For example in Mazovia district there is a tradition of logo programming competitions. As a result, students and teachers are familiar with logo environment and programming competitions issues. We also considered the problem of teachers' preparation from methodical and organisational point. We thought about popularising constructivisticts ideas. Along with competition tasks we prepared some materials which were helpful in preparation for the competition.

Challenges for students

Generally speaking the tasks on POLLOGIA competition differ in terms of difficulty not only between stages but also among one stage. We plan various tasks to meet expectation of different level preparation and experience from students' perspective. As an exemplification, tasks from the second stage there will be presented. There were three tasks – a graphical one, a recursive function and a task where operations on words were needed.

Chain Problem

The first tasks was to write a procedure **LZ :x :y :**z with three parameters which describe accordingly a number of red, green and yellow elements in the chain. The width or height of a picture should be not less than **400**.



LZ 2 2 1 *LZ 0 4 2*

Figure 3. Examples of calling function LZ

The difficulties connected with this task were to apply appropriately repeat instructions to avoid problems with missing elements (x, y, z were not necessary equal) and with the constraint of width/height of the picture. Students had to notice that the algorithm for short chains and long one was different. In the latter, they had to divide the width of the picture by numbers of elements, whereas in the former, they had to consider not only a number of elements but also its location.

Multiple Problem

The second task had to differentiate students. Pupils with less knowledge could present a solution only partly corrected and those with more experience – the full correct. The task was to write a function described in recursive way. When we have a and b, we have to count the last digit of Ln value from correlation as described below:

$$L_1=a,\ L_2=b,\ L_3=L_1*L_2,\ L_4=L_2*L_3,\ \dots\ L_n=L_{n-2}*L_{n-1}$$

Students had to write a function **LN :a :b :n** where **a,b**$\in$**<1,100>** and **n**$\in$**<3;1000000000>**. The additional constraint was to avoid long waiting for the result when the computer counts.

We will present the solution in three steps according to the level of difficulty.

Solution Number 1 – correct only for very small n

The function is written directly from the definition of the given task. We take two numbers **a** and **b**, and next we multiple them. We repeat this activity **n-2** times. The result of this function is the last digit of the counted number.

```
to LN :a :b :n
  repeat :n-2
   [
     let "c :a*:b
     let "a :b
     let "b :c
   ]
  output last :c
end
```

Solution Number 2 – for small n

It is not difficult to notice that when we are interested in the last digit only, we do not have to consider the whole numbers but only concentrate on the last digit. Therefore we can cut number in each iteration.

```
to LN :a :b :n
  repeat :n-2
  [
    let "c last :a*:b
    let "a :b
    let "b :c
  ]
  output :c
end
```

Solution Number 3 – for small and big n

Analysing the sequences described by formula given in the task, one can notice some regularity.

For example:

For initial numbers 0 i 0 we get 000000000000…

For initial numbers 1 i 1 we get 111111111111…

For 2 i 1 we get 2122482122482122 48…

For 3 i 1 we get 313397313397313397…

….

For 9 i 9 we get 991991991991991991…

Every digit is repeated at least every six places. Therefore, when for given initially numbers we generate the six first digits, we can find even a distant element very fast.

```
to LN :a :b :n
  let "temp word last :a last :b
  repeat 4
   [
    let "c last :a*:b
    let "a :b
    let "b :c
    let "temp lput :c :temp
   ]
  output (item 1+mod :n-1 6 :temp)
end
```

In such kind of task newbie students can solve it in the way similar to presented in solution number one, the better ones in number two and more experienced ones in number three. This was typical differencing tasks to help us to engage all students, on the one hand, and to choose the best ones, on the other.

Ciphered words

The third task was to cipher a word in a specific way. When you have a word **w** to be ciphered and a key **k**, you write down letters in a table as described below. In the first row you put first **k** letters from the left to right, in the second one, you put the next **k** letters, but from right to left, in the third one – like in first one from left to write, etc. You read the ciphered word when you look at subsequent columns.

For example, when you want to cipher a word **“eurologo2010paris** with key **6** you have to prepare a table like this

↓	↓	↓	↓	↓	↓
e	u	r	o	l	o
0	1	0	2	o	g
p	a	r	i	s	

The ciphered word will be **e0pu1ar0ro2ilosog**. In this task students have to be fluent in using operation on words, they have to know how to extract specific element and build new words. Also some mathematical knowledge is needed.

```
to cipher :w :k
  let "temp "
  repeat :k
   [
     let "x repCount-1
     repeat  1+(count :w)/:k
     [
      let "y repCount-1
      ifElse mod repCount 2=1
        [let "nr :y*:k+(1+mod :x :k)]
        [let "nr :y*:k+(:k-mod :x :k)]
      if :nr<=count :w [let "temp lput item :nr :w :temp]
     ]
   ]
  output :temp
end
```

By solving such tasks, students learn how to

- efficiently use the most important procedures and functions in Logo including operation on words and lists;
- apply iteration and recurrence;
- divide a problem into sub-problems, to form procedures with – and without - a parameter;
- be able to scale a drawing and to find proportions;
- test procedures with parameters – for different values, with special consideration of boundary conditions.

Organisational issues

The POLLOGIA consists of three stages. At the first level, which lasts about six weeks – students independently solve three graphical tasks. Beforehand, students have to create an account on the platform. At that stage, tasks can be solved at home or at school. The standard task is to write a procedure which would draw an expected picture on the screen. Students have to upload their solutions via a special form on the platform. Tasks are assessed and the results are published on the platform. To our surprise in first stage students from 15 districts (out of 16) took part. For the second one were qualified about 160 students.

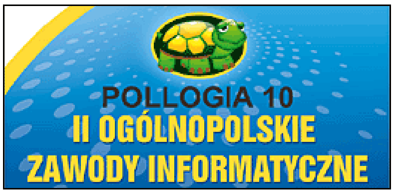


Figure 4. POLLOGIA competition promotional poster

At the second stage – regional one – participants solve three tasks. One is usually graphical, one with function with arithmetic operation and one with operation on words. The competitions are conducted at schools. The tasks are sent to teachers in ciphered *pdf* format and a password is published at a certain time on the platform. The time for solving the tasks is limited to 120 minutes. After the competition, students upload their solutions on the platform. The team of experts assess all answers according to the established criteria and presents outcomes in points along with a list of participants of the third stage (finalists).

The third stage is organised in one place. Thirty participants from different districts of Poland solve three algorithmic tasks. The time is also limited to 120 minutes. The tasks at this stage are similar to those of the second, but they are more difficult. They also include problems connected with processing lists. The finalists were from Wroclaw, Katowice and Warsaw area.

During the competitions we met a lot of gifted students. In terms of preparation there are questions as flowing:

- How much support do students received from their teachers?
- Did they work independently or some else helped them?
- What is the influence of school computer science lessons?

We face the problem connected with dialect Logo. There is a lack of free implementation of Logo in Polish language. This could be an obstacle for some students. We turn to creators of Imagine Logo to prepare a simplified version of Imagine Logo. In this version, called ImagineLite, there is only an interpreter and possibility to save procedures as text files. In this place, we would like to acknowledge the creators of Imagine for permission, especially Peter Tomcsanyi, for his contribution.

Summary

Summing up, the organisation of POLLOGIA competition was a big challenge for students who prepared and participated, for their teacher and for us – organisers. It took time and effort but we can say after *Eight Big Ideas Behind the Constructionist Learning Lab*, that it was fun, but hard fun. We decided to continue … and this year is the second edition of the competition.

References

LOGO Web portal in Microsoft's educational platform "Partners in Learning" http://www.pdp.edu.pl/logo

Stager, G., (2005) *Papertian Constructionism and the Design of Productive Contexts for Learning*, Proceedings of EuroLogo 2005. Warsaw. 43−53.

Borowiecka, A. and other (2008) – *Konkursy Informatyczne LOGIA i miniLOGIA*, OEIiZK

ICT in teaching children aged 6-9 years. The Polish approach. (Ist educational stage)

Agnieszka Borowiecka, *agn_b@oeiizk.waw.pl*

Maciej Borowiecki, *maciek_b@oeiizk.waw.pl*

Jan A. Wierzbicki, *jwierzbi@oeiizk.waw.pl*

All from: Computer Assisted Education and Information Technology Centre, Warsaw, Poland

Abstract

We discuss the problems of introducing ICT to the teaching and learning process in first stage of a school education. We describe the main points of a new polish core curriculum for this first educational stage, which assume the application of ICT to this stage.

In main part we describe educational packages, which were created by OEIIZK for the order of Polish Ministry of Education under the project "The use of modern technologies at the stage of early childhood education". The packages are dealing with the following subjects:

- We start work at the computer with the young children
- Preparation of teaching aids for the youngest children
- Logo in teaching the youngest children
- Working with graphics, film, sound
- The safety of the youngest children while using a computer

The aim of this packages is to inspire the teachers to rational and useful work with the youngest pupils and to the use of modern technology during the lessons. Finally we discuss the feedback of the created packages.

Figure 1. Educational packages DVD

Keywords

Early childhood education, didactic microworlds, educational movies, safety work with computer, teaching aids, Logo.

Introduction

The reform of the Polish educational system, launched in the 2009/2010 school year reduces school age from 7 to 6 years (In the most EU countries, children start school at age 6) [1].

Now we are under transitional period in which it is not mandatory targeting 6-year olds in schools. The parents are the one who decide whether their children will learn in school at age 6 or 7 years. However the first step teachers must reckon with the presence of the 6 year old pupils, and adapt to them the way of conducting lessons.

At the same time since the beginning of the 2009/2010 school year, a new course called "Computer Classes" has been introduced into primary school curriculum. This new subject is introduced in the first two cycles of education (from 6 to 12 years) [2].

For the first time the youngest pupils from grades I-III are obliged to use computers in their learning process.

That is why there is the need of proper preparation of the first step teachers, to use ICT during teaching and learning process.

It seems to be very important to propose the teachers not only a number of the well prepared materials, which they can use during the computer classes with the youngest children. It is important to encourage teachers to generate their own materials.

These objectives are included in the educational packages implemented by OEIiZK for the Polish Ministry of Education under the project "The use of modern technologies at the stage of early childhood education."

Prepared materials are related to the such important issues as: start working on the computer, preparing teaching aids, learning through fun, safety during the work with computer [3].

ICT in the new core curriculum – Ist educational stage

The new core curriculum includes the following skills for the student terminating the first grade [1, 2]:

1. He/She uses the computer on the basic level: runs the programs, use the mouse and the keyboard,
2. He/She knows how to use the computer, in the way to not endanger his/her own health,
3. He/She follows to the restrictions of the computer use.

The skills of the student at the end of the third grade, are:

1. He/She knows how to operate with the computer:
   a. how to use the mouse and the keyboard,
   b. how to correctly name the main elements of the computer system unit;
2. He/She knows how to operate the selected programs or educational games, developing his/her interests, uses the options in the programs;
3. He/She can search for the information and knows how to use it:
   a. browsing web pages which were selected by the teacher (for example the web of his/her school),
   b. recognizes the active elements on the website, navigates the pages in a specified range.
   c. runs animations and multimedia presentations;
4. He/She creates text and drawings:
   a. enters with the keyboard letters, numbers and other characters, words and sentences,
   b. creates the drawings using the selected graphics editor, e.g. using ready done geometrical figures;

5. He/She knows the risks arising from the use of computer, Internet and multimedia:
   a. knows that the work on the computer is tiring for the eyes, can cause spine strain, limit social contacts,
   b. is aware of the dangers arising from the anonymity of contacts and giving his/her address to the others,
   c. follows to the restrictions of the computer, Internet and multimedia use.

Working with computers is a particularly difficult task for teachers teaching young pupils. They should introduce themselves to the school use of modern technology in such a way as not to disrupt the harmonious development of their students.

ICT should be used to support teaching and learning, it is not an end - a separate course that focuses only on employing computers in isolation from other activities.

It becomes necessary to gain by the teacher who works with children in grades I-III, a broad knowledge of the methods of using the computers, Internet and multimedia teaching aids.

Educational packages developed within the project "The use of modern technologies at the stage of early childhood education"

In the autumn 2009 Methodical Center for Psychological and Pedagogical Support (CMPPP) carried out on the behalf of the Polish Ministry of Education project "The use of modern technologies at the stage of early childhood education".

The project was created to help the teachers to prepare themselves to teach with the computer.

It provides, inter alia, the preparation of five packages for direct use in school while working with children. The aim of the project was also to develop cascade training for preparing to work with the packages - 16-hour training for master trainers and 12 hours for ordinary primary school teacher (classes I-III). The teacher training included also 2-3 weeks of teacher self work using online method.

This training part of the project was carried out by OEIiZK.

Team of consultants and experts from Computer Assisted Education and Information Technology Centre (OEIiZK) has developed for the project the following packages:

1. We start work at the computer with the young children
2. Preparation of teaching aids for the youngest children
3. Logo in teaching the youngest children
4. Working with graphics, film, sound
5. The safety of the youngest children while using a computer

Each of the packages can be used separately. However the contents of the different packages are often linked. For example, the same pictures can be used as illustrations in the application "Words puzzle" (package: Logo in the teaching the youngest children) or for creating a movie from photos (package: Working with graphics, film, sound).

A graphics files needed to complete the exercises can be downloaded from the Internet, but in such cases appears the issue of copyright and Internet security (package: The safety of the youngest children while using a computer).

That is why the allocation of a specific issues should be targeted to the specific situations. The elements of the packages should be used by teachers in accordance with the needs for a particular lessons.

In addition to the educational packages, a syllabus for training trainers and teachers, trainer guide and teacher handbook have been prepared [3]. On the OEIiZK platform of a distance learning (http://szkolenia.oeiizk.waw.pl) additional supporting teacher training course has been placed as well.

A pilot training for the trainers from all over the Poland took place in October 2009 in OEIiZK. This training involved 80 trainers. Next the trainers trained 1200 I[st] stage teachers in their provinces.

Each of the trained teachers received a manual and a DVD with educational packages.

Package: We start work at the computer with the young children

This package covers the specificity and the rules of teaching the youngest pupils with the ICT support.

It contains descriptions and examples of implementation of the valuable educational software (Sebran, Kea Coloring Book, Tux Paint) and suggested interesting websites.

The integral part of the package are detailed scenarios of lessons, along with the supplementary material (student work cards, presentations for use by the teachers, drawings for the exercises, etc.).

The package contains four scenarios: "Getting Started with PC", "Playground", "Proper writing" and "We tell a story". The DVD also contains PDF files, and instructional videos to help teacher to become familiar with the presented educational software.

Figure 2. Kea Coloring Book Software

Package: Preparation of teaching aids for the youngest children

The aim of the materials placed in the second package is to supply teachers with the knowledge and practical skills for the needs of self design the teaching aids using the computer.

They include descriptions how to prepare a simple teaching aids by using office programs such as: PowerPoint, Word and a program to generate a crossword - EclipseCrossword.

The main task of the package is to show the teacher the ability to create valuable teaching tools using publicly available tools (word processor, presentation preparation program), which does not require the expertise or programming skills.

The included DVD contains the examples of teaching aids, which are designed to be an inspiration for the teacher for creating its own ones, relevant to the content they teach young pupils. The package contains three scenarios: "The first spring flowers", "Animals Protected in Poland" and “Proper writing". In addition to the scenarios, the teacher may use supplementary materials such as demonstrations, exercises for the student in the form of presentations, multimedia, crosswords, etc.

Most of prepared aids does not require to work with the pupil in the computer lab. They can or even should be used while teaching pupils in the regular classroom equipped with one computer and video projector or interactive board.

Students should carry out various exercises under the teacher supervision, and try to solve problems in the group.

Figure 3. The first spring flowers

Package: Logo in the teaching the youngest children

This Logo package does not require any programming skills from students or teachers [4].

It includes ready-made applications – didactic microworlds – using which pupils can develop the ability to identify directions, rotations to the right or to the left, mapping shapes, finding the repeated elements. Students are also trained to add integers, increase their perception and focus attention.

The basic application of the package is "Control the Turtle". Three lesson scenarios use this application: "We start playing with the turtle", "Small and great turtle steps” and "We teach a turtle how to draw smarter".

The set of exercises, and a version of the program which allow the teacher to modify the tasks in order to prepare his own lesson with this application is also included.

The package also contains three other didactic microworlds to work with children: "Alphabet puzzle", "Words puzzle" and "Playing with the memory".

Teachers can prepare their own sets of exercises for these applications, using photos or graphics or pictures done by the pupils.

Figure 4. Words puzzle - didactic microworld

Package: Working with graphics, film, sound

In the package “Working with graphics, film, sound” we discuss the ways of creating educational films consisting of a sequence of images – Digital Story Telling. There are described the rules for searching the imagines, the methods for editing them and in detail the process of creating a movie in a Photo Story Application [5].

The teacher may carry out under this package the four following scenarios: "We tell a story", "Digital ZOO", "Where does the honey come from?" and "Digital legends”.

Supplementary materials in the form of work sheets, examples of films, film scripts and the images examples which can be use for creating the movie are included with the scenarios.

In addition, teachers receive ideas for the new tasks for students and instructional materials.

Figure 5. Digital ZOO – educational movie

Package: The safety of the youngest children while using a computer

The last of these packages is devoted to the introduction of the issues associated with various hazards which can appear during work with computers, especially when we are using the Internet resources [6].

The classification of the most common hazards for the computer users is presented in a concise manner. The special attention is focused on the youngest students.

We discuss also the ways to prevent hazards and show the examples of the didactic materials covering these matters for the use while working with students.

In the package, there are three scenarios: "Safe work with the computer", "Safe use of the computer – computer lab statute" and "Computer lab statute - video story”.

The student may also watch the instructional films showing correct behavior in the lab and the exercises which can be performed during the intervals of using the computer.

Teachers receive the information concerning the preparation of the computer lab in order to work in it with the youngest pupils, safe behaviors during Internet surfing, methods of blocking unwanted content, copyright, etc.

Figure 6. Safe use of the computer – computer lab statute

Summary

The main aim of the materials prepared within the project is to inspire the teachers to rational and useful work with the youngest pupils and to the use of modern technology during the lessons. The materials are not a full multimedia programs, in which students choose several different options, with no reflection to the relevance of actions they take.

Using a computer is not an end aim in itself - the student should assimilate the new knowledge in an constructivist approach: learning by creation.

The prepared packages have met with the great interest. Trainers and teachers participating in the trainings have been expressing a positive opinions about their usefulness.

A quote from the Forum of Trainers:

"I have noticed in other posts, that the participants impression after your trainings are the same as after mine ones. All participants praised the prepared training manuals and materials. I heard from several participants that finally they got a manual, materials and aids that are tailored to children's learning in the classes I to III."

A quote from the Teachers Forum for the package: We start work at the computer with the young children:

"I have examined the scenarios and I can say that I could done them all while working with children in my school. Indeed, each of them in interesting ways (depending on the ongoing lesson theme) teaches pupils specific skills and broaden their knowledge. Children learn the rules for carrying out work on the computer, done the content of the curriculum of early child education (math, polish language and literature, art ....). This contents are correlated and pupils can have a great fun while learning, this is very important in work with this age group."

A quote from the Teachers Forum for the package: Logo in the teaching the youngest children:

“It's a great program that develops logical thinking, but also the three-dimensional imagination and abstract thinking.

Children quite unconsciously improve visually - motor coordination. Learn about the world of mathematics, without being aware of it...”

References

[1] Web page of Polish Ministry of Education – *http://www.men.gov.pl*

[2] OEIiZK web portal about the polish educational reform – *http://www.oeiizk.waw.pl/reforma*

[3] Project teacher’s handbook – available at *http://www.oeiizk.waw.pl/reforma*

[4] Papert, S. (1980) *Mind Storms: Children, Computers, And Powerful Ideas.* Basic Books.

[5] About Digital Story Telling – *http://www.techteachers.com/digstory/ideas.htm, http://www.storycenter.org*

[6] Computer Hazards – *http://www.cln.org/spam.html*

The Mandelbrot Set Fractal as a Benchmark for Software Performance and … Human Creativity

Pavel Boytchev, *boytchev@fmi.uni-sofia.bg*
Dept. of Information Technologies, Faculty of Mathematics and Informatics, Sofia University

Abstract

Science and art are considered as distinct areas of the global gamut of human activities. Scientists draw inspiration from art and artists embed science in their work. This paper presents the results of a one-year experiment, which started from testing a dialect of Logo, resolved some mathematical and programming challenges of dynamical systems with complex numbers and ended up as an artistic computer-graphics exhibition in Sofia University.

The paper begins with a short historical and mathematical description of the Mandelbrot set fractal, two small fragments of which is shown in Figure 1. A program generating the fractal image is provided as pseudo code as well as a short discussion about the colour schemes.

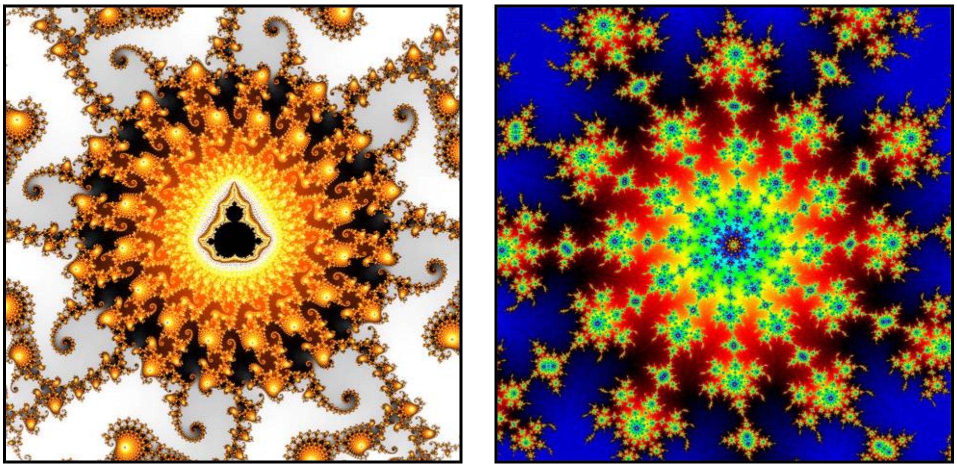


Figure 1. Two fragments from the Mandelbrot set fractal

The focus of the paper is on making of the gallery, on the ideas that have been considered and the decisions that have been made. Several interesting observations are also discussed. They include a multidisciplinary research and a collaborative work of a group of unknown people. Except for an exhibition, the result of this work is the acknowledgement that:

- People want to create and want to share their creations.
- It is challenging to try new ideas and approaches. To be creative, someone needs to have the anxiety to try something that he/she has never done before

This paper is an illustration how science and art can fit together in a mutually enriching way following the core ideas of learning by creating tangible artefacts and gaining new knowledge and skills.

Keywords

Fractal, Mandelbrot set, science and art

Introduction

Being the first international conference dedicated predominantly to Constructionism, it is likely that many papers will focus on Constructionism from the educator point of view. To become conscious of this learning theory and its practical applications, the author decided to submit himself to an experiment where he will play the role of a student learning through the Constructionism approach. This role would require **learning an entirely new set of multidisciplinary ideas**, **collaborating with unknown so far peers** from all over the world and **producing tangible artefacts** with application in scientific, artistic and educational contexts.

A fractal beginning

The Mandelbrot set fractal

The mathematician Benoît Mandelbrot was the first person who used a computer to visualize the behaviour of a dynamic system. In 1975 he introduced the word *fractal* (from the Latin *fractus*, *broken*) to denote objects with fractional dimension (Mandelbrot, 1983). His research built the foundation of fractal geometry – the link between classical Math and the “chaos” of atmospheric turbulence, biological populations and the stock market. Decades before fractals were named, other mathematicians had studied them – Weierstrass, Koch, Lévy, Cantor, Poincaré and Julia.

The fractal found by Mandelbrot is called *the Mandelbrot set* – Figure 2. This is a set of all points in the complex plane, for which the iterative application of a polynomial with an initial value of 0 generates a bounded sequence. In fact, a fractal is the boundary of the set. The colouring is determined by the speed at which the sequence crosses a preselected limit, beyond which there are no members of the Mandelbrot set (Mandelbrot, 2004). Mathematically, the Mandelbrot set M is defined as the set of all points with a finite supremum (i.e. least upper bound) of $f^n(0)$ where $n \to \infty$ and $f(z) = z^2 + c$.

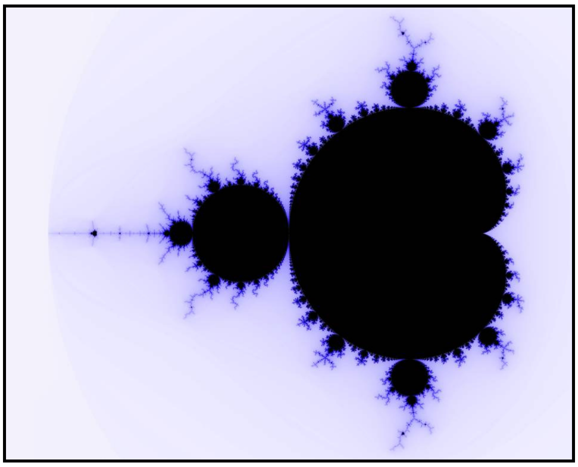


Figure 2. The Mandelbrot set fractal

Software implementation

The generation formula for the Mandelbrot set is not directly convertible into a Logo program code, because most Logo dialects do not support complex numbers. The solution is to use the geometrical representation of complex numbers – each of them corresponds to the coordinates of a point in the plane.

Another problem occurs with the implementation of the supremum. It involves infinity and it is practically impossible to do infinite number of calculations for the generation of a single point of the set. Fortunately, there is a mathematical proof that in many cases it is possible to predict whether the supremum is infinite. For example, if the distance between point z and point $(0,0)$ is 2 or greater, then z will go into infinity as we continue to do more iterations. For such point there is no sense to continue with more iterations – this point does not belong to the Mandelbrot set.

Sometimes, for a given c we need a few thousands of iterations, for another c we may need billions, and yet, there are values for which we can never cross the boundary set by threshold 2. This is the reason for the introduction of a limit of iterations. If we reach the limit and z is still within the boundary, then we assume that z will always stay inside (and thus, z will be a member of the Mandelbrot set).

The algorithm generating the Mandelbrot set can be expressed in pseudocode as:

```
1: For each pixel (x0,y0) do:
2:     n,x,y ← 0
3:     while n<max do:
4:         if x²+y²>2² then: plot(x0,y0,n) stop
5:         x´ ← x²−y²+x0
6:         y  ← 2xy+y0
7:         x  ← x´
8:         n  ← n+1
9:     plot(x0,y0,0)
```

Line 3 ensures that the program will do a finite (at most max) number of iterations, line 4 checks the threshold, and lines 5 to 7 implement the calculation $z \leftarrow z^2+c$. The procedure *plot* is used to draw a single pixel at coordinates (x_0,y_0). The third input of *plot* is the colour, which is usually set to the number of iterations n. If we select a colour gradient that extends from 0 to $max-1$, then some details might be invisible – they will appear as indistinguishable colours. For example, colours corresponding to n=350 and n=400 in Figure 3 will both look blue if the colour gradient spans over the whole interval from 0 to 1000. However, if the colour gradient is shorter (e.g. 200) and tiles until it reaches 1000, then the colours of n=350 and n=400 will be quite different. It is a matter of aesthetical judgment to decide the size of the colour span and the order of colours in it.

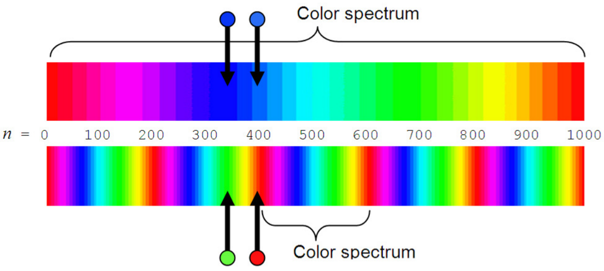


Figure 3. Colour gradient span interval of 1000 (up) and 200 (down)

This colour scheme is often referred to as *Escape Time Colouring*, because the colour depends on the speed of escaping from a circle of radius 2. The scheme produces visible colour bands for faster escapes. There are options to smooth the colours (Stevens, 2005) and one of them is the *Normalized Iteration Colouring*, where the colour is determined by $n+\log_2 \log_{f^n(0)} 4$.

First posters

In the autumn of 2008 it was decided to make several performance tests of the Lhogho[1]. Designed as a compiler, it was expected to outperform any other Logo in term of numeric processing. A benchmark should do millions of math operations and the calculation of the

[1] Lhogho web site: http://lhogho.sourceforge.net

Mandelbrot set loomed as a perfect candidate – it required an unavoidably huge amount of mathematical calculations for determining the colour of just a single pixel.

Attempts with other Logos needed more than 10 minutes to generate the simplest picture of the Mandelbrot set. Using Lhogho took just a few seconds without any code optimizations. When the benchmark program[2] was ready, it was decided to visualize the generated image in order to verify that all calculations were correct.

The first dozens of zoom-ins showed images, which strikingly resembled things in the material world, like a forest, a solar protuberance, and a coffee's cream. Some of the images were digitally manipulated and blended with real photographs. The results were sent to colleagues and their feedback was extremely eloquent – "*Make an exhibition!*"

The next six months were spent in finding new interesting fragments in the fractal and converting them into artistic pictures. Figure 4 is a map of all places selected for the exhibition.

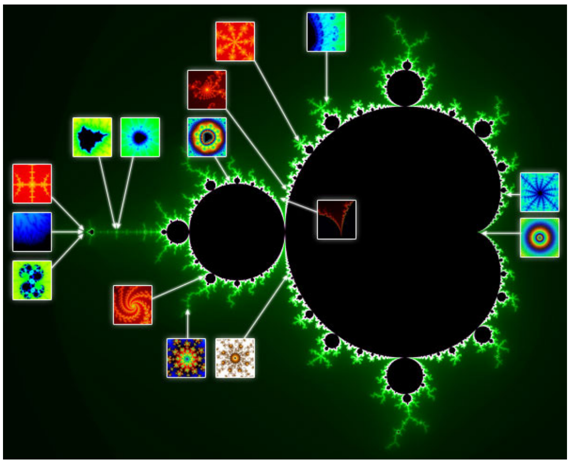


Figure 4. Points of interest

Observations

The actual making of the posters required answering many questions, like:

- What information must be included in each poster? How to present it?
- What real life object or phenomena can be illustrated?
- How to get a suitable photograph entirely legally and at a sufficient resolution?

Several interesting and somewhat unexpected observations were reached during the making of the exhibition. The next subsections describe some of them.

The depth of the fractal

All close-ups of the fractal showed that it has practically infinite levels of details. There are areas that can be zoomed in thousands and millions of times; and yet they continue to provide new and different shapes.

Figure 5 shows the initial fractal in the top left-most image. There is a small square in the middle. The content of this square is shown zoomed 10 times in the next image, then its centre is zoomed 10 times, and so on. The last image uses a magnification of 10,000,000 times. In some of the studies of the fractal, we had zoom factors of 1,000,000,000,000, which reached the precision limit of the mathematical libraries used for the calculations.

[2] Its source code is included in the Lhogho package and is licensed under GPL.

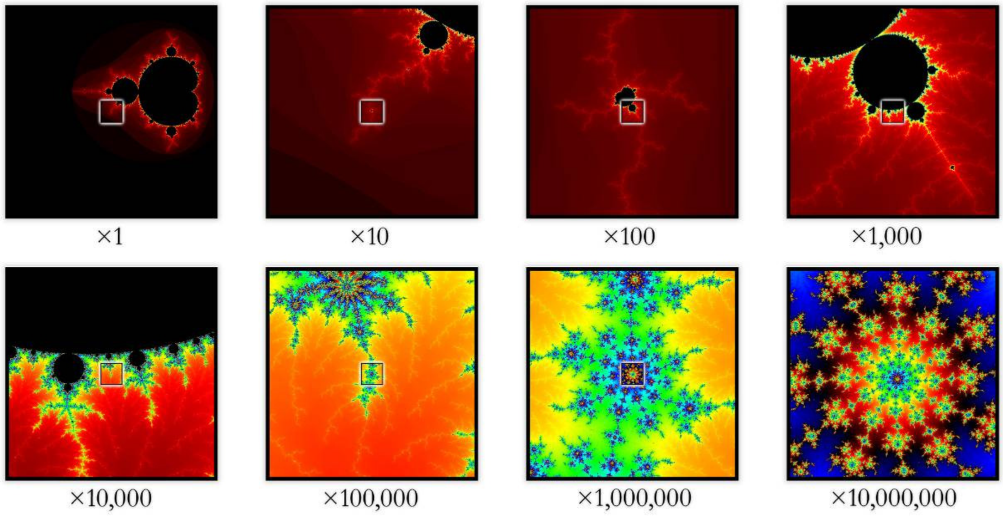


Figure 5. Zooming in up to 10 million times

One of the posters is dedicated to the unlimited diversity of the fractal – the middle poster in the 4th row in Figure 10. It says that if the zoomed fragment is real size, then the other end of the fractal will reach the Sun. A slightly higher calculation precision would provide zoom factors where the fractal will become larger than the known Universe.

Loading the posters with information

It was problem to decide what information to include in each poster. Two of the most important elements were the fractal image and its artistic interpretation. However, these two elements were not enough, because each poster should provide a unique viewpoint on something new and should present navigational data so that people can recreate the fractal if they want to. Figure 6 (left) shows what is "hidden" in each poster – there is the computer-generated fractal (1), the coordinates of its centre (2 and 3), its scale (4, 5 and 6). Thumbnails (7, 8 and 9) visualize the process of reaching the fractal image at steps by a scale factor of 10.

Image (11) is the artistic representation of the fractal. It shows a relation between mathematically defined (and computer generated) image with objects, events and ideas from our lives. Although sufficient by itself, each artistic representation is accompanied with a short text describing something interesting about the topic (10). Finally, a credit line (12) reveals the collaboration with other people.

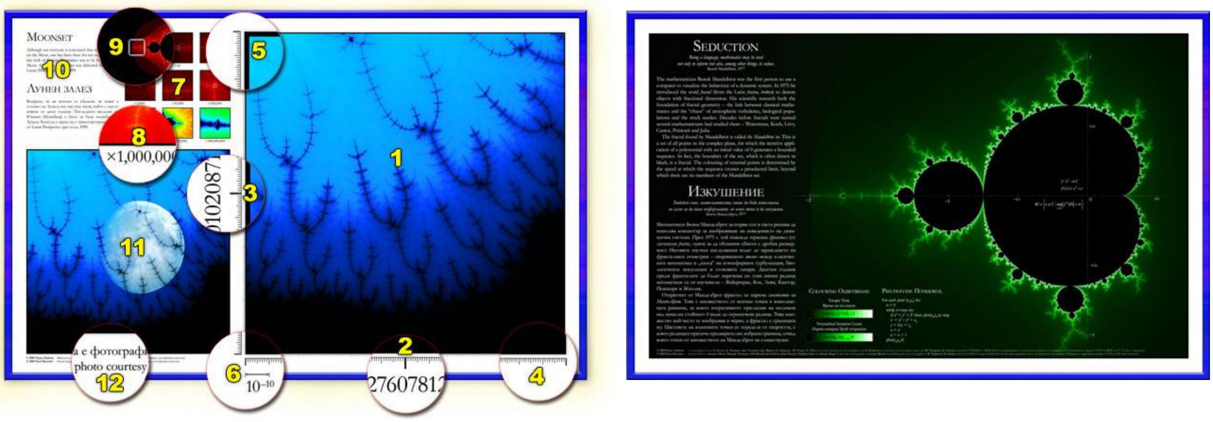


Figure 6. Information presented in each poster (left) and the introductory poster (right)

All posters use the same layout to make it easy for viewers to find what they are interested in – technical details, artistic interpretations or the additional interesting facts.

There is one additional introductory poster, Figure 6 (right), which shows a high-resolution image of the whole Mandelbrot set along with a brief history of the fractal and its generation methodology. This poster also contains the mathematical backbone of the fractal and a pseudo code of a program that draws it.

International, national and familial collaboration

The making of the posters required collaborative efforts of many people … at every step from the initial design to the final hanging of the posters in the exhibition hall. For example, people contributed with digital photographs, with ideas, with support for the physical making of the exhibition, with the translations and language tuning, and so on.

Some of the artistic fractal representations are based on digital photographs by other people. We identified the authors and they were from all over the globe – from Japan to Canada. They were all asked for permission to use their work. We did not know any of these people and the initial hopes were that approximately 10% of them would agree. The process of getting permissions was expected to be rather long. However, these hopes were groundless. Surprisingly, authors of the photographs gave permissions, so we say again *Thank you* to Annette Olson, Daisuke Tomiyasu, Elfi Berndl, Jon Sullivan, John French, Nicholas Gere and Simon Tong.

All posters are bilingual – the texts are both in English and Bulgarian. Louise Blyton from Australia and Svetla Boytcheva from Bulgaria provided valuable fine-tuning of the texts' contents and style.

A group of colleagues from the Sofia University and the Bulgarian Academy of Science (namely: M. Todorova, B. Sendov, D. Dobrev, E. Sendova, E. Stefanova, E. Kovatcheva, N. Nikolova) provided technical and moral assistance. Elitza Boytcheva, author's daughter, gave comments and ideas, as well as found factual bugs in the artistic images in some of the posters.

Figure 7. Prof. Dobrev and prof. Sendov assisting the exhibition installation

Learning in a multidisciplinary way

The preparation of each poster included the hunt for some interesting and little-known facts. It was quite challenging to browse hundreds of documents until a proper story was found. As a result, the posters now refer to a broad range of topics covering time, space and life.

- *Multidisciplinary topics*. The posters describe objects and events from various knowledge areas like Astronomy, Biology, Palaeontology, Physics, History, Geology, Mathematics, Meteorology, Geography, Engineering, Manufacturing, Jewellery, Crafts and Game design.
- *Time dimension*. The topics span from 100 million years in the past up to 5 billion years in the future of the Earth.

Figure 8. Hand-made elements

- *Life dimension*. There are posters about prehistoric life of spiders, wasps, bees, flies and mites, and yet there are posters about microorganisms and … extraterrestrial life.
- *History dimension*. Several of the posters present historical events and people – Eugene Shoemaker, who was buried in the Moon; the Chinese monk Li Tian, who made the first firecracker; Menaechmus, a tutor of Alexander the Great, who discovered the hyperbola and Appollonius, who named it.
- *Artefacts dimension.* This dimension shows some interesting facts about the objects around us. For example, a cup of coffee simulates turbulent atmospheric phenomena at a scale that is still not reachable by modern supercomputers; the zipper was initially used only for boots and children's clothing; and the left and right shoes did not exist two centuries ago, because of manufacturing problems.
- *Geographical dimension.* The facts in the posters refer to various locations on Earth, like Spain, where the oldest amber with three orders of flying insects was found; or the Tokara Islands in Japan, where the longest solar eclipse for this century was observed in 2009; or the night skyline of Hong-Kong.
- *Computer graphics dimension.* Most of the artistic interpretations of fractals are done by blending digital photographs with computer-generated images. This blending is a complex process, which requires various techniques. Some of the objects, however, are entirely artificial – see Figure 8. They have been model from scratch with an image manipulation program. For example, the jewellery shaped like the Mandelbrot set is entirely artificial; each piece of the 361-pieces puzzle is made from scratch; the pagoda is built by borrowing shapes from the fractal.

The exhibition

The initial ideas for an exhibition appeared in late 2008. Half a year later, the posters were almost complete (as image files). The last step was to print the posters, put frames and hang them somewhere. A suitable event was the 120th anniversary of the Faculty of Mathematics and Informatics at Sofia University.

The name of the exhibition was chosen to be "*Seduction*". Although it sounds too provocative, it is based on Benoît Mandelbrot's words:

> *"Being a language, mathematics may be used not only to inform but also, among other things, to seduce."*

With the help of colleagues and the financial support by the Faculty, the exhibition was opened in October 24, 2009. The first visitors were the people from the cleaning stuff who provided an initial feedback. An interesting but still unexplained observation was that many of the later visitors saw the exhibition in groups of four - Figure 9. A set of thumbnails representing all 15 posters (except the introductory one) is shown in Figure 10. The exhibition has an on-line version[3] for those, who cannot visit the real one. The online posters, however, do not demonstrate the full beauty of the fractals and the extreme level of details.

Figure 9. The exhibition and the visitors

Except for the permanent exhibition at the Faculty of Mathematics and Informatics, the posters are also included in a private collection in Australia. Individual posters are sent to science and educational institutions in North America. Mini-exhibitions have been set up for two nation-wide events – the *Conference of the Union of Bulgarian Mathematicians* and the *National IT Olympiad* for 5-12 grade students.

Lessons learned and future plans

One of the most important things that was learned for the last year is that the majority of the nowadays projects are results of collaborative efforts and utilize various resources from different places. People who create and construct scientific and artistic artefacts are happy to see that their work is being used by others, or at least that it inspires other people to be creative.

The other important lesson is not to be afraid of experimenting with ideas and techniques, which are entirely new to you. This exhibition is made by a person who has never made any other exhibition. Almost everything in the process was new – from requesting permissions for using photographs and working with a company for large-scale digital print, to putting frames on the posters and hanging them on the walls.

It would be perfect if the exhibition triggers the creation of various educational, scientific and artistic activities, which could be:

- Investigating the Mandelbrot set fractal and finding other interesting places
- Looking for relations between other fractal images and the real life
- Seeking additional information for interesting, but little known facts
- Practicing skills for digital image manipulation
- Developing new, faster algorithms for generating fractals
- And finally – exploring new directions for creativity

[3] On-line exhibition: http://mandelbrot-set.elica.net

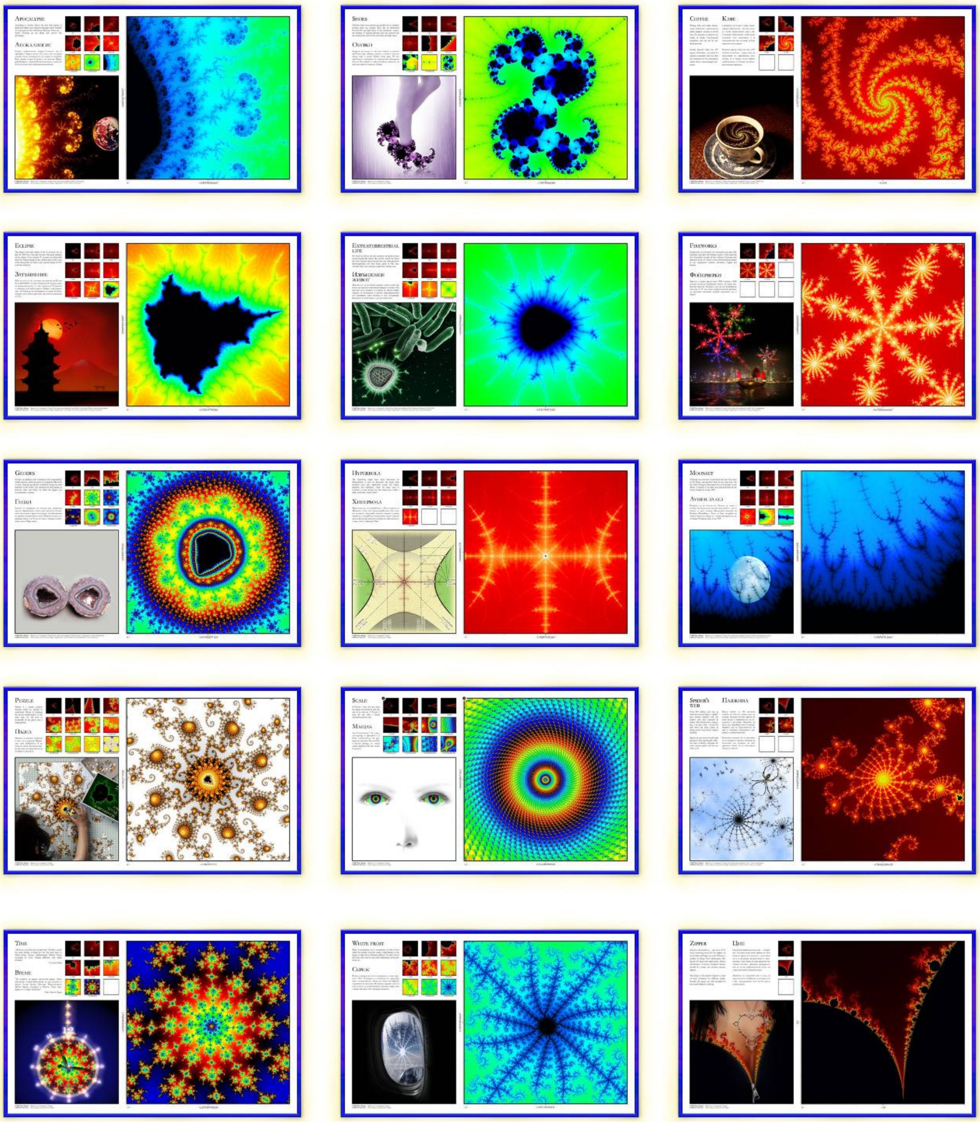


Figure 10. The full set of posters

Fortunately[4], it is hard to make any plans for the future. The exhibition was triggered by a boring work and it is impossible to imagine what element of the exhibition will trigger new ideas. That is perhaps the best lesson learned – even the most boring reality can turn on people's imagination and creativity. They only need to realize the precise moment and to capture the initial momentum. What happens next is something that cannot be planned and foreseen.

Postscript

It is author's belief that everything that is created as a result of some official or unofficial project is actually a (re)source of a new project. Similarly, the making of the exhibition is not a stand-alone effort and it initiateed a lot more activities. Science and art can mutually catalyze themselves and the bird view of the exhibition illustrates this – Figure 11.

[4] Yes, fortunately.

The development of Lhogho initiated the fractal exhibition, which is the main topic of this paper. The exhibition evolved into an award-winning animated film called "*A Journey in the Mandelbrot set*". It shows a virtual tour to the locations in the exhibition and it won the first place in the *Fractal Movies* section of the *Spring 2010 Fractal Art Contest* organized by FractalForums.com.

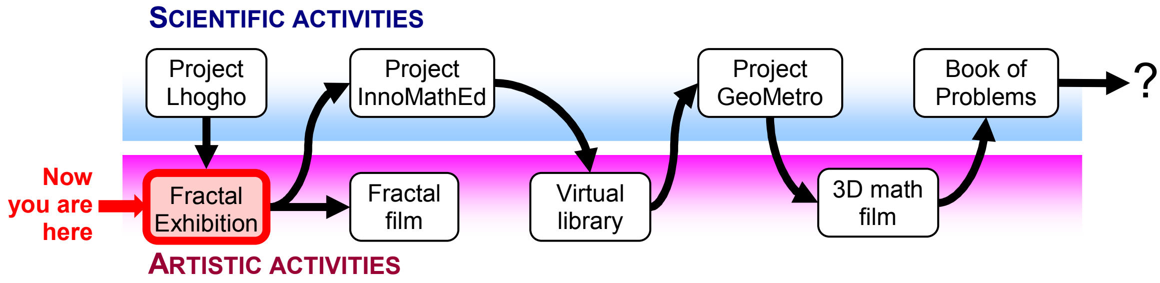


Figure 11. Mutual catalyzation of scientific and artistic activities in an educational context

One of the posters from the exhibition contains a mathematical problem about conical sections. This problem initiated the participation in the international project InnoMathEd[5] (Innovation in Mathematics Education on European Level) and a set of interactive 3D applications have been developed. This set gave birth to a virtual library with several dozens of 3D virtual models for drawing mathematical curves, modelling geometrical transformations, generating ruled surfaces and so on. Each model is accompanied with an animation, which is available to everyone[6].

The collection of animations inspired the beginning of a new project, called GeoMetro, which made the author to create another 3D film[7]. This math film shows seven different mechanisms for ellipse construction. Some of them are pictured in Figure 12. The plans for this film go beyond mere presentations. Its purpose would be to become the math problem conditions in a book of problems about ellipses. The conditions of the problems would be the film itself. This is a new and unique project which goal is to provide unprecedented motivation for students through 3D multimedia. The book of problems will be accompanied with a software library where mathematical virtual experiments could be done.

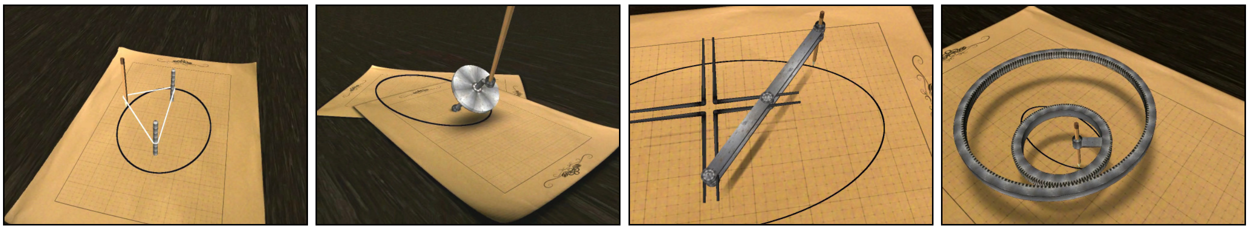


Figure 12. Snapshots from the 3D math film "Ellipses…"

References

Mandelbrot, B. (1983) *The Fractal Geometry of Nature*. W. H. Freeman & Co., USA

Mandelbrot, B. (2004) *Fractals and Chaos : the Mandelbrot set and beyond*. Springer-Verlag, New York

Stevens, R. (2005) *Creating Fractals*, Charles River Media, Inc., Rockland, Massachusetts

[5] Project InnoMathEd: http://www.math.uni-augsburg.de/prof/dida/innomath

[6] Mathematical devices: http://www.youtube.com/user/ElicaTeam#grid/user/6534E936D46257BF

[7] The 3D math film "Ellipses…": http://www.youtube.com/watch?v=1v5Aqo6PaFw

Constructionism and Creative Movement: A Manifesto

Andee Rubin, *andee_rubin@terc.edu*
TERC 2067 Mass Ave Cambridge MA 02140 USA

David Capps, *dcapps@hunter.cuny.edu*
Dance Program, Hunter College 695 Park Avenue NY NY 10021 USA

Elizabeth Bradley, *lizb@colorado.edu*
Dept of Computer Science University of Colorado 430 UCB Boulder CO 80303 USA

Abstract

The lens of constructionism has traditionally been focused on the learning of topics such as mathematics, science, and computation. Of the arts, only music has played a significant role in the development of constructionist ideas. In this paper, we present a manifesto based on our belief that dance pedagogy can both exemplify and expand the current views of constructionism, and we contend that choreography and creative movement improvisation can be viewed as constructionist pursuits. The diagram below, for instance—which lays out the cues given and followed by dancers during a piece—is just one example of a constructionist-style representation that has been used to great advantage by dancers and choreographers.

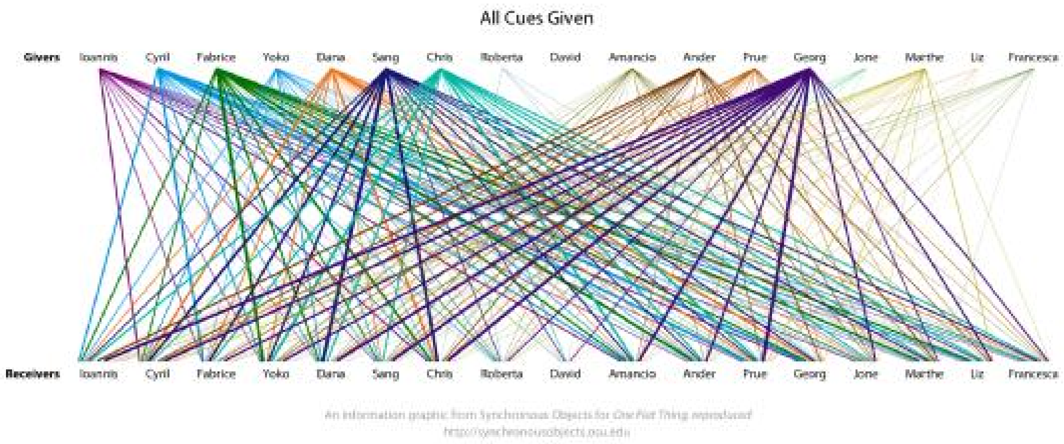


Figure 1. A chart of the interactions of individual dancers in "One Flat Thing, Reproduced" by William Forsythe (reproduced, with permission, from `synchronousobjects.osu.edu`*)*

In our work together—as dancer and choreographer, computer scientist, and math educator—we have all stretched our worldviews and forged unanticipated intellectual bonds. Our goal in this paper is to share our excitement and discoveries, by not only examining dance from a constructionist view, but also inviting the traditional constructionist community into the world of creative movement. This expansion of the boundaries of constructionist inquiry, we believe, will provide the constructionist community with new metaphors, new dilemmas, and new collaborative potential.

Keywords:

Representation, Motion, Dance, Choreography, Improvisation

Introduction

A basic tenet of constructionism is that learning is most effective when it arises as part of an activity that the learner experiences as constructing a meaningful product. Often this product is computational or mathematical: a geometric model, a piece of software that accomplishes a personal goal, a video game, a small robot that reacts to external stimuli. A “meaningful product,” however, can just as easily be a piece of music, a novel, a movement improvisation, a dance. And while we don’t always think of these as “learning activities”—perhaps because they don’t take place in a classroom setting with explicitly stated learning goals—it is hard to imagine creating an artistic object without learning something. In fact, many artists understand their creative process specifically as research: asking questions and “constructing” answers that function in their medium.

The process of artistic creation easily exemplifies another common aspect of constructionism: that creating something tangible provides the learner/creator with the opportunity to revisit her product and “debug” it. While we don’t often think of writing music as involving debugging, any composer will tell you that a piece of music goes through many iterations during its creation. Moreover, the creator’s goal is often in flux, via an ongoing “feedback loop” between the product/answers and the goals/questions. Thus, as the product is brought into being, it is continuously creating new parameters for debugging. Choreography, in virtually all of its many forms and approaches, fully embodies this process via its creation of “knowledge” in the form of movement structures.

While the field of constructionism is comfortable focusing on science, mathematics, and computation, examining dance — an art form that uses human movement as its medium—from the constructionist perspective introduces challenges for constructionist thinking. How can constructionist vocabulary be extended to authentically describe both the learning of movement skills and concepts, and the creative design of movement constructions? What new problems and contradictions arise from using constructionist principles to examine dance? What insights might choreographers and dance teachers gain from this interdisciplinary endeavor? And what insights can teachers of other disciplines gain from behaviors and processes that exist in the dance field?

Learning and designing dance

While there are separate words for “dancing” and “choreographing,” modern dance—understood as a generative approach to the study and creation of movement rather than as a received movement or choreographic style—regards learning movement skills and learning to create original movement constructions as deeply connected activities. That is, learning to dance, in the modern dance paradigm, involves both acquiring the ability to execute received movement patterns and being involved in the ongoing articulation of new movement ideas. In this approach, therefore, “dancing” fundamentally connects performance skills with choreographic investigation: any execution of movement is fully integrated with the conceiving of movement ideas, ideals, and approaches. Dance, then—from pedagogy to performance to artistic creation—can be understood as a fundamentally constructionist learning activity in which the student acquires performative and creative skill simultaneously, intertwining the conceiving, doing, evaluating and revising of the materials of the form. From this perspective, we can look at a finished, performable dance as the result of a spiralling creation, execution, and reflection process: one that has a vocabulary, an epistemology, and a changing spectrum of goals, all informed by a rich historical set of subtexts.

The deep structure of modern-dance training clearly embodies these ideas. It is organized around the pedagogical subtext that every dancer should be both a creator and an executor of movement—as much during preparatory training as in fully professional behavior[1]. Improvisation is used early on in order to open students' minds and bodies to physical possibilities, and to create an immediacy of movement that reflects a creative mindset. Often, the movement creation process involves an iterative dialog between student, teacher, and perhaps other performers. The teacher has five primary pedagogical goals for his or her students:

- to look at movement from an aesthetic/artistic standpoint
- to learn basic kinesiology
- to become versed in the wide variety of movement concepts and skills
- to establish a ongoing creative relationship among those elements
- to combine all of these knowledge bases in order to mobilize the body in an articulate, creative, expressive, and intelligent fashion

The conceptual vocabulary of dance concerns time (rhythm, form), space (facing, location, direction), dynamics (effort, weight, force), anatomical function (mechanical efficiency and articulateness of the instrument), basic movement patterns (walk, run, jump, fall, and other "pedestrianisms," among many other elements) and more-advanced movement patterns that are considered an evolving "lingua franca" in the field. While conveying this multi-valent vocabulary to his or her students, a modern-dance teacher strives to plant "seeds" of knowledge for movement, which the students then are asked to use both in practicing given action patterns or set phrases and in constructing their own movement—through improvisation, spontaneous choreography or other student-created artifacts.

The teaching/learning strategies involved in this are process oriented: imaging a position or its derivative, for instance, exploring a movement direction using different parts of the body, or sensitizing the kinesiological function of joints. Some conceptual structures exist to scaffold this process: Rudolf von Laban's grid of time/space/energy elements, for instance—which is described at more length in the following section—provides the opportunity to experience a movement concept in many self-directed ways. For example, the spatial concept of *forward* can be explored through a variety of strategies—gesture of various body parts, locomotion, facing and focus—within a fully variable grid of dynamics and rhythms. Similarly, a student can focus on abrupt and sustained rhythms while creatively utilizing a wide variety of spatial and dynamic qualities. In so doing, the knowledge of the movement concept is mobilized in such a way as to give the student agency over her or his investigation.

The overarching goal of this arena of dance instruction is not only to teach students how to do steps, such as *pliés*, but to understand *why* one does *pliés*—what they are "about" (*viz.,* verticality and weight)—and therefore how they might be applied in new environments or experiences. This intentionally generalizes the student's kinaesthetic and aesthetic knowledge about the step, making it into something that she or he can adapt to a new situation. An imperfect analogy would be to the concept of recursion; students learning to program need to learn the meaning and function of recursion, not just how to write a program in a particular language that accomplishes it.

After each class exercise—and sometimes during or before it—a modern-dance teacher, like someone who is teaching programming, will often unpack the pedagogical subtext that he or she had in mind. Likewise, he or she often provides a phrase of movement that intentionally mobilizes the concept or element at hand, so that the student also learns to apply the experiential knowledge to received movement that might be part of a performed dance. From the students' perspective, this activity is a constant interweaving of concept and experience. This

[1] Other genres, and even some modern "techniques," are somewhat different: in these, there is no connection to experiential creativity, dancers simply execute pre-defined movement, and there is little expectation that they will create it—or modify it very much.

structure of alternating constraint and exploration is typical of constructionist approaches to teaching and learning.

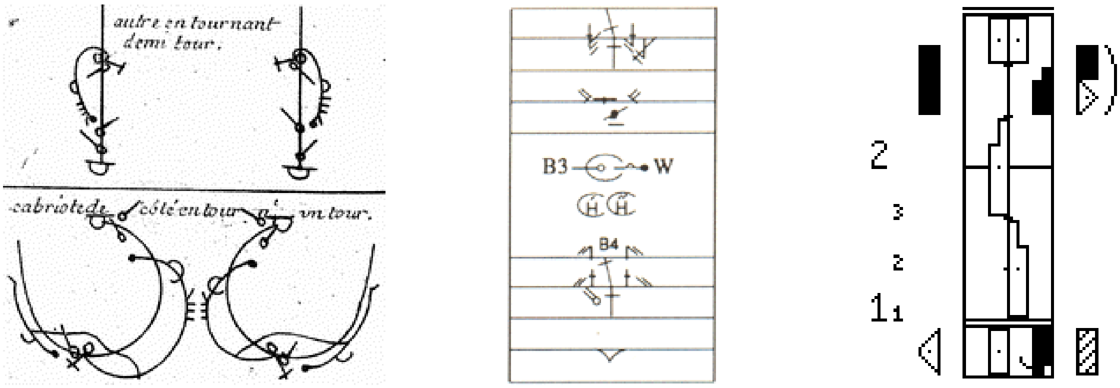
Movement improvisation, a critical element in modern-dance training, establishes an experimental structure that aims to mobilize skills and understanding without imposing a specific outcome. This exercise aims to nurture the flexibility of body and mind that is central to the ideals of humanistic education. It also seeks to entrain a level of expressivity and artistic acumen that is central to the role of creative collaborator as well as the role of performer of choreographed dances. Improvisation is equally central to the training of choreographers who aim to create a personalized movement language and constructions. In all four roles—student, collaborator, performer, and choreographer—the constructionist perspective on knowledge is central to the modern-dance paradigm. The "product" is, in the case of training for a dancer, increased ability to use the body imaginatively. In the case of a choreographer interested in making an ultimately repeatable dance, improvisation can generate new movement behaviors that will become material for elaboration.

Representation in Dance

While dance is inherently an embodied representation of movement ideas, choreographers also use varied graphic representations of movement. These representations differ along several axes: choice of "vocabulary," conception of time, and, most importantly, whether they intend to specify a dance precisely or to generate a new dance every time they are performed. A brief history of dance notation exemplifies these contrasts.

Most dance representations specify <u>motion</u> and therefore represent movement over time, not position at a single time. In the Baroque era, for instance, Feuillet notation was used to record the floor plan and movement sequences of European court dances. An example is shown in Figure 2(a). In this notation, symbols referred to fairly complex predetermined movement sequences—"steps" like a *demi tour* (half turn). They were placed along maps of the movement of the body as a whole through space, and often keyed directly to notations of musical accompaniments. Importantly, this system relied on foreknowledge of the steps themselves, which for the most part had verbal names that were central parts of training methodologies for dancers. Steps that were not in this vocabulary could not be represented with this notation.

In the 20th century, Rudolf Benesh and Rudolf von Laban created movement representations that were much more general and expressive. Both were designed to be style-neutral, relying solely on knowledge of the spatial, temporal and energetic dynamics of the human body, rather than on experiential familiarity with the subject movement itself and its specialized semantics (e.g., names of specific steps, such as *pliés*.) These notational systems are harder for humans to learn than Feuillet notation, because they break the movement down into small, abstracted components, but they are much more widely useful. Benesh notation, currently utilized particularly by ballet dancers, graphs positions on a staff similar to musical notation, using symbols to indicate posture, body part by body part. A single position is shown in Figure 2(b). Time is represented moving from left to right on a staff, with the equivalent of musical "rests" and time signatures indicating the duration of transition from position to position.



(a) (b) (c)

Figure 2. Dance notation: (a) Feuillet (b) Benesh (c) Labanotation. These images are reproduced, with permission where copyright exists, from [Feuillet 1700], www.benesh.org, and www.dancenotation.org, respectively.

Labanotation, widely used in a variety of movement disciplines and shown in Figure 2(c), also indicates time via a linear staff. Unlike Benesh notation, the staff is vertical, moving from bottom to top. It focuses not on sequences of positions, but on directional actions through space: a particular symbol describes a direction of movement in relation to the core and front facing of the body, and the placement of the symbol within the staff attaches that action to a particular body part. The vertical scale of the symbol indicates the duration of the action; a lexicon of graphic modifiers specifies details such as degree of rotation, degree of contraction or extension, and contact with objects or other bodies. Significantly, Labanotation places information about the action of the weight bearing structures (typically feet, but possibly knees, shoulders, pelvis) at the core of the staff, so that displacement through space (locomotion), indicated by the same symbols applied to body parts, can be immediately visualized. von Laban also devised a separate system of analysis and description of performative qualities of movement, mentioned earlier in this paper, which generalizes the actual actions but specifies in great detail the experiential qualities of the movement. In this “Laban Movement Analysis” scheme, observations of time (fast/slow), space (indirect/direct) and force (light/strong) are combined to generate a grid of possibilities such as punch (fast, direct, strong) or flick (fast, indirect, light), which are indicated in the score via specialized symbols.

The tension between these different notational approaches—particularly the trade-off between expressiveness and unwieldiness—brings up a variety of issues that are interesting from a more general perspective on representations. Many dancers and choreographers find Labanotation difficult to use, for example. The decomposition of the body into parts is just not a natural way for people to think about motion. Style-specific representations like Feuillet notation, on the other hand, are much more natural for their user communities, but they obviously fail if the movement to be described does not pre-exist in their vocabulary. Labanotation can capture arbitrary, unfamiliar movements, but its reductionist process makes it awkward for dancers who for the most part approach movement tasks via an integrative and kinaesthetic knowledge process. This same tradeoff arises in many disciplines, of course: languages that describe objects at a small grain size can represent a wider range of objects, but require more complex “interpreters;” languages with a smaller vocabulary, in which the objects that are described are more “intuitive” are limited in what they can describe, but easier to understand. Machine language, which few of us ever see or use anymore, is at one end of the spectrum. The formal language of quantum physics might be at the other end of the spectrum.

The representations in Figure 2 are intended to specify movement sequences precisely. Other representations are more like jazz charts, in that they generate a new dance every time they are performed. All of the realizations of such a representation are related in fundamental ways but can be very different in actual execution. A famous example of this kind of semi-specified motion is choreographer William Forsythe's recent piece "*One Flat Thing, Reproduced*." The piece is "defined" deterministically via specific rules about how each dancer's actions serve as cues for other dancers. The cue sheet in Figure 3 is a detailed representation of this: what each action is are and how each one is triggered by specific actions of other dancers[2].

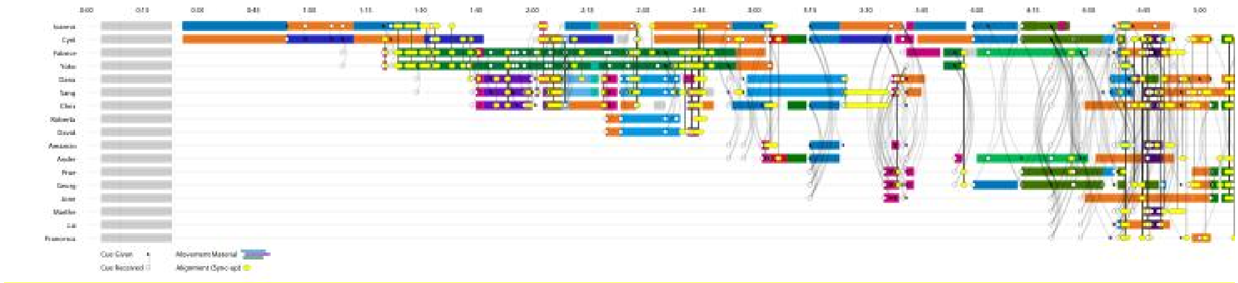


Figure 3. A portion of the cue sheet from "One Flat Thing, Reproduced" by William Forsythe (reproduced, with permission, from `synchronousobjects.osu.edu`*)*

Each time these rules are executed by a group of dancers, a new version of the dance emerges. The variations among executions of the dance arise from the inevitable small differences in the ways dancers move: in movement timing and conscious decisions made in the moment of performance. Because each dancer's motion takes place as a response to other dancers' motions, small and large shifts in behavior can "snowball" in unpredictable ways. Some choreographers—Forsythe in *One Flat Thing*, for instance—use this in explicit ways, specifying rules that govern the interaction of each member of a group of dancers. As is the case in agent-based models such as those of birds and vehicles, the behavior that emerges from these individually simple rules can be rich—and often unpredictable. In dance, this often catalyzes an iterative creative process, wherein choreographers and dancers work out the effects of different rules via collaborative discovery. Note that this agent-based dynamics is more complex than most of the classic NetLogo examples, in which a group of identical agents carry out identical procedures. In these systems, patterns arise (*viz.,* bird flocks or traffic jams) as the agents simultaneously follow their rules. There are examples of parallel agents that behave probabilistically and generate predictable statistical structures, such as normal curves. Neither of these is isomorphic to the creative technique exemplified by *One Flat Thing*, where individual agents follow different rules, and those rules vary in time and space.

There are fundamental differences underlying the different dance representations and choreographic behaviors that are described above. Benesh, Feuillet, and Labanotation are generally used in cases that involve a central omniscience—e.g., a teacher or choreographer—or a traditional, fixed sequence of steps. In this context, movement sequences are pre-determined and the dancer's responsibility is simply to replicate them as accurately as possible. In other situations—e.g., Forsythe's composition—the dance is not generated by a central omniscience, but rather *in the moment* by the disseminated action of agents following individual rules. Importantly, traditional forms of ritual or performative dance forms, such as ballet, may not be distinguishable from, or differently pleasing than, dances that made with agent autonomy. But there are *huge* differences in the experience of dancers as they learn the different kinds of

2 The chart in the abstract is an overall summary of who interacts with whom in the same dance.

dances—and tremendous cultural dissonance if one has been trained in the genre at the other end of the spectrum.

All of these ideas and observations have direct parallels in traditional constructionist concepts regarding primitives and the ways in which they can be combined. Various mathematical systems are built on different sets of axioms and rules. Programming with Scratch is different from programming with Logo, for instance, because of differences in:

- the primitives,
- the representations of primitives (i.e. visually, with shapes and colors vs. with words), and
- the rules for combining them.

In current modern dance parlance, the word “technique” describes access to movement potential and the ability to mobilize elements like forward, rotation, flow, verticality, etc.. The dancer/performer must be able to embody as diversely as possible movement ideas like turning, gesturing, and locomotion, just as we hope students who use programming environments can use primitives as diversely as possible in their pursuit of a meaningful goal.

A Future Synergy

In this paper, we have argued for moving beyond the traditional “movement simply in service of mathematics” view and breaking out of the purely visual representation. In the long run, the benefits of opening the boundaries of constructionist inquiry should accrue to both the constructionist field *and* to the field of dance. Co-author Capps’ involvement in this work, for instance, has led him to new procedural ideas as well as new vocabulary for his work as a dancer, teacher, and choreographer, including a computer-human duet that explored the notions of theme, variation, and chaos (see Figure 4).

Figure 4. A scene from “Con/cantation: chaotic variations,” by David Capps and Elizabeth Bradley.

Co-authors Rubin and Bradley have found modern dance to be a compelling arena for studying constructionism "in the wild." We have all learned that modern dance pedagogy is constructionist in its view of the power of construction, the relationship of process to product, the value it places on self-examination and revision, the investment of the learner in the process, and the hope it embodies that learning is a source of joy.

This work has also brought to the fore many issues that connect computer science, mathematics, and education—and, indeed transcend those fields. As a start, here are four hopefully provocative questions for us to all consider together:

- What new forms of representation for traditional constructionist topics do dance representations suggest?
- What is the relationship between design and performance, e.g. between choreography and realized dance?
- How does improvisation fit into a constructionist perspective?
- What is the parallel in other fields of "technique" in dance?

References

Feuillet, R. (1700) *Chorégraphie, ou L'art de d'écrire la dance, par caracteres, figures et signes desmonstratifs, avec lesquels on apprend facilement de soy même toutes sortes de dances,* Chez l'auteur...et chez Michel Brunet, 1700.

Teaching living-art: drawing choice and rendering behaviour.

Carol-Ann Braun, contact@inner-media.org

Artist and Director, Association Concert-Urbain, Paris, France.

Abstract

Making living-art involves programming representations capable of adapting to an environment and engaging in an aesthetic exchange. A living-artwork does not necessarily copy nature or mimic human conversation. It does, however, imply the creation of a fully constituted rhetorical system that provides conditions for a poetical "give and take" between art-work and public, involving mutual recognition if not "communication" in the full sense of the word.

Figure 1. "Corps Complices", Catherine Langlade (2009), co-produced with the CUBE, France.

How does one teach students to make "living art"? More often than not, the techniques associated with artificial intelligence are simply transferred to an artistic environment. The assumption: that aesthetic intelligence is a subset of an objective, general intelligence that "is", a priori. When it comes to teaching young programmers outside of Art Departments, the aesthetic dimension of intelligence is not even taken into consideration.

Our paper will broach some aesthetic issues raised by living-art. First, we shall consider the nature of the digital sign and evoke a few aspects of intermedial aesthetics. We shall then set the stage for understanding what is specific to digital representation, by exploring the notion of choice. Examples of student projects, built around the tree structure of *Un Conte à Votre Façon,* by Raymond Queneau, will be described. Next, we shall discuss immersion, i.e. what happens when an entire environment "behaves". We shall use examples taken from storyboards written by young game designers, inspired by Italo Calvino's *Invisible Cities,* in order to challenge standard notions of seamlessness and transparency. Living-artworks designed by professional artists working at the "Living Art Atelier", at the CUBE, France, will also be discussed.

The questions raised by this paper are not answered. They concern differences between terms such as "analogy" and "metaphor" when applied to information technologies. We allude to aesthetic theories and notions about "process" and "transparency" without having enough time or space to explain them in detail. Our approach is not philosophical, but the examples discussed provide interesting clues and venues for further research.

Keywords

Aesthetics, intermedia, metaphor, analogy, artificial intelligence, living-art, behavior, choice, immersion, representation, communication, process.

Living-art

Making living-art involves programming representations capable of adapting to new information and engaging in aesthetic exchange. A living-artwork does not necessarily entail copying nature or human conversation. It does, however, imply the creation of a fully constituted rhetorical system that provides conditions for an experimental and poetical "give and take" between art-work and public, involving mutual recognition if not "communication" in the full sense of the word.

An example of a living-artwork

At A Distance (figure 2) is such a work, created by the photographer Damaris Risch at the Atelier du CUBE, on the outskirts of Paris, France. The Cube is France's largest digital art center, whose activities include sponsoring artists interested in making living-artworks. It has also launched a seminar on the subject, oriented towards professionals in theater, dance, music and the visual arts.

At A Distance was first shown during the outdoor "Festival Premier Contact", launched by the CUBE in 2003. It consists of over a hundred photographed self-portraits, organized around a semantic map. Programming was done with a software package called Virtools (used for game design), hooked up to a neural network.

The photographs blend into each other and create the illusion of a seamless "living portrait", generating a subtle range of responses to the gestures and attitudes of those who pause in front of it: the character has a tendency to smile when someone comes near; at night she sleeps, though she'll wake up if a noisy scooter passes by. This "living portrait" is a full-fledged interlocutor and the effect is compelling: we seem to communicate with the image, and it seems to communicate with us, as if moved by a will of its own. And the public relates to it—Risch had to wipe lipstick off the piece several mornings in a row…

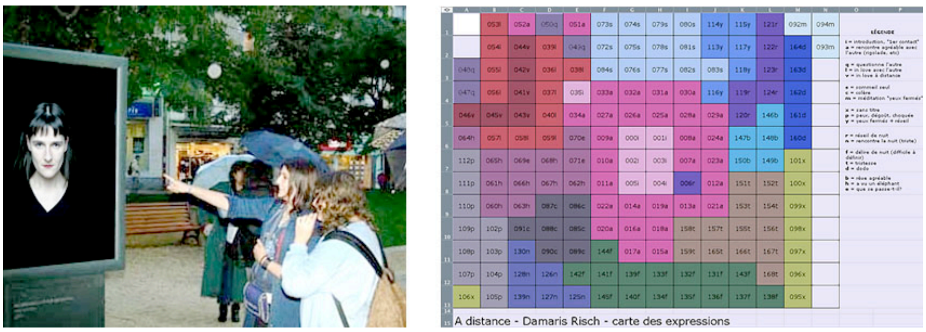


Figure 2. "At A Distance" by Damaris Risch, 2005, and a map of moods used to create paths and variations. C-oproduced with the CUBE, France.

At A Distance is a good illustration of the artistic horizon opened up by artificial intelligence. However much Risch's enigmatic smile might resemble Mona Lisa's, it has no equivalent in the history of painting and photography. It is derived from the application of information technologies to a new type of "representation", difficult to link up to established aesthetic criteria.

Where does one start when attempting to judge the artistic quality of such a piece? In the context of 20th century art, *At A Distance* resembles dozens of other large photographic portraits. In purely pictorial terms, it is not as inventive as, say, a Picasso. From a commercial point of view, say a perfume advertisement, it is a bit drab. The dialogue set up by the work isn't especially refined, either, however intriguing the experience of communicating with an autonomous image might be. Who cares if yet another pretty face smiles or not?

In spite of the quality of the photographs animated by Risch, the “wow” factor here is primarily technological and, secondarily, aesthetic. Something unprecedented, however, is at play. What current aesthetic criteria apply to this new type of “representation”?

Teaching living-art

From an aesthetic perspective, technology is neither a transparent nor a neutral conduit for a representation. The tools one uses and how one thinks are not two separate issues.

Much has been written on the relation between appearance, structure and process in painting and sculpture. One aspect of the subject is of particular interest here: “A work or art is not an analogy. It proposes a meaning in the form of constellations,” (Francastel,1965). Pictorial form doesn’t mimic the outside world. It is a unified way of being, of doing, of seeing, geometrically organizing spatial relationships within a whole and between its parts, “caught between material abstraction and the concretization of a concept.” Such principles are at the core of teaching in Art History and Visual Arts Departments.

When it comes to teaching outside of Art Departments, the aesthetic underpinnings of intelligence are not even taken into consideration. More often than not, the techniques associated with artificial intelligence are applied to an artistic environment without thinking that artificial intelligence might operate differently if exercised in an aesthetic manner. Aesthetic intelligence is a subset of an objective intelligence that “is”, a priori.

Our paper will attempt to illustrate the aesthetic tensions linked to living-art. The tensions concern transparency and process. First, we shall consider the nature of the digital sign and evoke a few aspects of intermedial aesthetics. We shall then set the stage for understanding what is specific to living-artworks, by considering the notion of choice. Examples of student projects, built around the tree structure of *Un Conte à Votre Façon,* by Raymond Queneau, will be described.

Next, we shall discuss the nature of immersion when an entire environment “behaves”, i.e. embodies a rhetorical system and builds a specific relationship with its public. We shall use examples taken from storyboards written by young game designers, inspired by Italo Calvino’s *Invisible Cities,* in order to challenge standard notions of seamlessness in the experience of immersion.

Finally, we shall return to a few examples of living-artworks created at the Atelier du CUBE. We shall quote from a recent publication on the subject, entitled “Living-Art”, written by Florent Aziosmanoff, Art Director at the CUBE. He sets forth a particularly astute way of articulating the technical parameters of an aesthetic relationship. Once again, we shall be very critical of the work presented, not because it isn’t interesting, but because there is still so much to do.

Drawing choice

Metaphors

Digital representations are not bound by the same constraints as tangible media. Granted, a digital collage can look just like a paper collage. This much said, the artistic lineage between the two is not procedural but stylistic; it is it is superficial, not essential. With paper, edge is definite, surface real, and depth an illusion. On the screen, paper is simulacra, coded in text; its edge is arbitrary (or, strictly speaking, only visual), its surface is hypothetical and its depth is an optional dimension to be explored if so desired. Articulating the space between the center and the edge of a digital “re-presentation” can’t mean the same thing that it does on paper.

Digital media use metaphors to re-instate limits, artificially. These metaphors prolong known ways of finding form and meaning in a new technological context. “Metaphors, problems and technologies are interrelated. Metaphors set problems that technologies are commonly put forward to address. These technologies in turn promote metaphors that set the problems. Technologies also provide metaphors of each other...” (Coyne, 1995). Metaphors facilitate the acceptance of innovation too, as exemplified by the easy switch from hand-held cutter to virtual scissor to make a digital collage.

It is easy to mix metaphors with digital imagery. Sandwiched between code, image and gesture, a digital sign determines function, choice, and movement. If a hyper-linked sign (e.g. →, an arrow indicating “next”) has symbolic value subject to interpretation, it also has use value: a click on the sign leads to another level of information. What was once separate, i.e. "you don’t have to know how to read in order to turn a page," is now conjoined. The augmented digital sign is both a messenger and a hinge (Jeanneret, Souchier, 1999), at the juncture of medium and genre.

Both conceptually and procedurally, the horizon is wide open to intermedial practice and thought. This makes it practically impossible to define how a given digital technique might influence the construction of a given representation. In this context, the relation between “material abstraction and the concretization of a concept”, to refer to Francastel’s expression once again, no longer operates in the same way. Here, material is abstract to begin with; concepts are rendered “concrete” by joining and articulating metaphors generated by technologies at several removes from perceptible form.

Does this necessarily condemn digital art to being mimetic? and if so, does the mimesis of media count as “analogical thinking”? Is there any reason for artificial intelligence to inhibit thinking in “constellations”, as understood by Francastel, when describing aesthetic intelligence? Without venturing into the art-historical and philosophical issues implied by these questions, beyond the scope of this paper, a few concrete examples of class-work done with them in mind helps understand what is at stake in asking them.

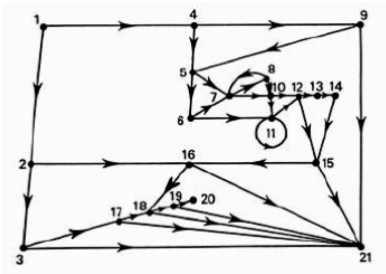
The Opacity of Choice

Choice is one of the “motors” of process in art (even when an artist chooses to abandon choice in favor of chance). As for interactive art, it is primarily about staging choice. This common ground is a good starting point for a discussion about analogy, process and transparency.

The examples discussed here are based on *Un Conte à Votre Façon* (1973), by Raymond Queneau, founding member of OULIPO (Ouvroir de Literature Potentielle), a movement exploring formal constraints in what was then called “potential literature”. *Un Conte* begins with a question asking if the reader is interested in a story about three little peas; if so, he’s to move on to paragraph 4; if not, then he’s to go to paragraph 2. From paragraph to paragraph, the reader skips his way through a whimsical narrative until he runs out of choices and the story comes to an end *(figure 3).*

“Little pea” in French is “petit pois”, which sounds just like “petit poids”, which means “light weight”, i.e. like the weight of bits of information needed to pass from one node to another. *Un Conte* is laced with more or less cryptic references to information technologies: loops, the purr of machines, information break-downs, suggestions to consult a “dictionary”, etc. Queneau parodies the structure of “command-response” and the “yes-no” of a binary world.

Un Conte floats between the reading and writing process battled out between several voices, often at odds: a “classical” narrator, with no story to tell; banter among peas who interrupt and provoke each other; whimsical circular instructions that catapult one to the end of the story without warning; the machine’s voice, “suspended” in time, waiting for instructions; not to mention our own inner-voice, anticipated in the text. By hopping from one register to the next, by having his characters bicker instead of building narrative, by having us drop his text to go look up a word in a dictionary, by undermining our sense of purpose and efficiency, Queneau resists the transparency of choice.



4. Il y avait une fois trois petits pois vêtus de vert qui dormaient gentiment dans leur cosse. Leur visage bien rond respirait par les trous de leurs narines et l'on entendait leur ronflement doux et harmonieux.
    a. si vous préférez une autre description, passez à 9
    b. si celle-ci vous convient, passez à 5
5. Ils ne rêvaient pas. Ces petits êtres en effet ne rêvent jamais.
    a. si vous préférez qu'ils rêvent, passez à 6
    b. sinon, passez à 7
6. Ils rêvaient. Ces petits êtres en effet rêvent toujours et leurs nuits sécrètent des songes charmants.
    a. si vous désirez connaître ces songes passez à 11
    b. si vous n'y tenez pas, vous passez à 7
7. Leurs pieds mignons trempaient dans de chaudes chaussettes et ils portaient au lit des gants de velours noir.
    a. si vous préférez des gants d'une autre couleur passez à 8
    b. si cette couleur vous convient, passez à 10.
8. Ils portaient au lit des gants de velours bleu.
    a. si vous préférez des gants d'une autre couleur, passez à 7
    b. si cette couleur vous convient, passez à 10

Figure 3. Queneau, Raymond, 1967. "Un conte à Votre Façon", in Oulipo, La literature potentielle (Céations, Re-créations, Récréations). Paris: Gallimard, coll. "Folio/Essais", 1973.

It may seem at first glance that this approach to choice is too far removed from the subtle equilibrium of living-artworks to be of any relevance here. With living-art, choices are not necessarily explicit or even perceptible. Yet behavior is a consequence of choice, however disguised or subtle.

Un Conte lays bare the mechanisms of choice. By playing with all the things choice can mean or do in an interactive work, Queneau manages to give weight to transition, to the space between one state and another.

It is interesting to note that in one the first digital interpretations of *Un Conte* (no longer to be found on the Internet), transitions were the first aesthetic features of the work to disappear. A simple click on a "yes" or a "no" button, situated right next to the question, made text magically appear or disappear. The technological "plus": empowering result over choice, erasing doubt and irony, and short-circuiting the option to simply disobey and choose neither "yes", or "no" so as to skip around the text at one's own leisure.

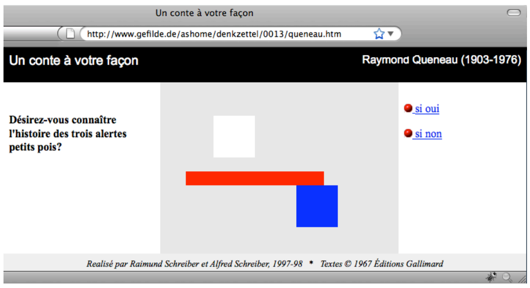


Figure 4. http://www.gefilde.de/ashome/denkzettel/0013/queneau.htm

Fifteen years later, much the same can still be found on-line, with a twist: a little geometric graphic animation evokes the idea of variation, though the link with the sequence of the text is not clear (*figure 4*).

Drawing Choice

Resistance to transparency is implicit in the tree-structure drawn by Queneau (*figure 3, above*). A tree-structure is not a simple, neutral element in a technological equation. Choice has a shape, and that shape is neither "de facto" nor without consequences on the representations it structures. Queneau's drawing looks like a painting. It is framed in a box. It contains a horizon line, large open spaces, and even a hint of perspective. It is "pictorial" and cleverly self referential; the path of choice "winds up" in a corner opposite to the corner where it starts. In a sense, the end mirrors the beginning. True to form, the story is not about different ways of getting somewhere, but different ways of going nowhere.

Below are some examples of the paths of *Un Conte* drawn by young apprentices in a class on interactive writing. Needless to say, each drawing reveals an entirely different approach to choice.

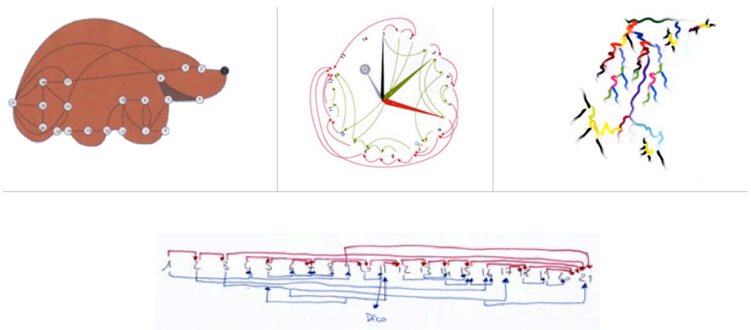


Figure 5. Representing the invisible: diagrams for Queneau's "Un Conte à Votre Façon". From left to right : hedge-hog, clock, strokes, and, below, 'yes' / 'no', by apprentices at the CFA'Com, Bagnolet.

While figuring out the links between each paragraph one student wound up with a drawing of a hedge-hog *(figure 5).* This got her thinking of burrowing through tunnels, blindly, oblivious to the options lying ahead. At best, a "yes" could mean "turn right" and a "no" stand for "turn left". A loop could be a dead end; consulting a dictionary could mean breaking out of the tunnel into sunlight.

More importantly, the drawing raised the issue of what role to give the mounds of dirt left behind. Each choice implied residue dirt. What to do with it? Were the mounds of dirt left from previous readings and "passages through the text" to remain visible?

Her neighbor in class drew a tree-structure in the form of a clock. Each choice implied "skipping" to another time of the day. In this example, "turning back the hands of time" was not possible; each choice was a kind of "life choice", day leading into night leading into dawn, etc. We discussed at length the relation between the "real time" it took to make a choice and the "fictional time" of the story. Bridging the two tempos became the central idea of this student's multimedia rendering of Queneau's text.

Another student decided to imitate the process of painting, building up an image stroke by stroke, scraping away options and re-painting over old choices. Each "reading/painting" of the Queneau's fable contained similar motifs, but distributed in different combinations on the surface of the "screen-canvas".

Students singled out verbs, adverbs, adjectives, etc to describe the gestures, tempos and textures to be associated with choice: to match, to glue, to turn, to skip, to unfold, to roll, to tear, to zap; quickly, slowly, hesitantly, indifferently, collectively, alone; rough, smooth, jagged, arbitrary, evanescent, chemical…etc. They were then free to combine any of the words in order to imagine how to stage choice digitally.

One student, for example, decided to equate "yes" with black and "no" with white, both co-present on the screen. The resulting interpretation of Queneau's text was an Escher-like animation, switching in and out of positive and negative modes, and, during the choosing process, vacillating between the two. Choice became a kind of morph, focused on "the seam" between two representations.

Figure 6. "Un Conte a Votre Façon", by Raymond Queneau. Interpreted by Antoine Denize and Carol-Ann Braun, in Machines à Ecrire, by Antoine Denize and Bernard Magné, Gallimard, 1996.

The screen captures seen above *(figure 6)* avoided all mention of the word "yes" or "no". Queneau's text was split in two parts. In the first part, choices were made on the pages of a school book; moving objects (spilt ink, a rolling marble, pen nibs in a box) marked each step in the narrative. Spilt ink meant being interested in dreams; the appearance of a shadow meant being interested in "why" something had happened….Throughout the "choosing process", traces of past choices accumulated on the page. A second part was built around a game of hop-scotch, allowing an uninterrupted account of the choices made in the first part, all while maintaining a sense of what was being skipped over.

Beyond the mouse

One could argue that these forms of choice are hemmed in by pre-determined formalisms intrinsic to "informatics": screens, pointers, mouse pad, etc.

Why bother with all the trappings? One student imagined *Un Conte* as a "live" obstacle course in the Alps. Form here was woven in a mix of auditory and visual cues embedded in an "augmented" landscape, orchestrating the voices of the narrator, the characters, the instructions, and musings about choice in the fast lane.

Choice can take any shape, any scale, any speed; yes, it can be orchestrated on an immersive, polyphonic and multi-modal scale. The important thing to remember here is that choice is not "free" and "transparent" but an opaque gesture laden with aesthetic meaning; it is an essential material factor in the "concretization of concepts".

Rendering behavior

With living-art, choice is a sub-set of a larger concept, that of "behavior". At issue here is not only the viewer's behavior but also the art-work's behavior; in a viewing situation, both are interdependent.

An episode from Italo Calvino's *Invisible Cities* is useful in understanding what is involved. At one point, Marco Polo, recounting his many travels to a curious Kublai Kahn, describes a bridge, stone by stone. Kublai Kahn asks: "Which stone props the bridge up?" Marco Polo answers that no single stone is more important that the other. It is the shape of an arc, formed by all of them, that keeps the bridge up. Kublai Kahn then answers, "Why talk to me of stones, then?" To which Marco Polo replies: "Without the stones, no arc exists…" (Calvino, 1972). The form behavior takes is both material and immaterial, spanning and structuring the passage between two points.

This exchange between Marco Polo and Kublai Kahn brings to mind Francastel's constellations, "caught between material abstraction and the concretization of a concept." More than a topography, each of the cities described by Calvino embodies a deep symbolic structure, imagined around a dozen key abstract figures: exchange, desire, signs, gaze, memory, the double…

The inhabitants of *Ersilia*, for example, “stretch strings from the corners of the houses, … according to whether they mark a relationship of blood, of trade, authority, agency. When the strings become so numerous that you can no longer pass among them, the inhabitants leave: the houses are dismantled; only the strings and their supports remain.” Here, Calvino has given physical form to invisible relationships. These relationships are a kind of social “fabric” that both reflect and impose behavior.

The metaphors of interest here are procedural. They relate to models that organize behavior. To quote Coyne again: “…categories of objects and actions do not exist merely in isolation but are formed into experiential gestalts—basic metaphor structures- […] There is a metaphor structure pertaining to containment…distinguishing an interior from an exterior…Other metaphor structures pertain to paths, links, forces, balance, the up-down orientation, the part-whole relationship and the center-periphery relationship.” (Coyne, 1995) Applied to living-art, these metaphors provide a vocabulary for articulating relationships, with any technology or assemblage of technologies and representations, on any scale.

With living-art, the challenge is in re-presenting relationships that remain abstract, that are drawn in an invisible space. The space is not a physical architecture but an intelligent environment. Immersion here is more than spatial, it is about how behaviors — those of an environment, those of visitor — are structured by a specific kind of abstract form.

Animating immersion

Using Calvino’s *Invisible Cities* as a spring board, students in game design at the Ecole Supérieure de Création Interactive Numérique, Laval, were given the task of making environments behave. Although gifted in 3D rendering, they were not asked to do so with the sophisticated tools at their disposal. It was important to get them to think “out of the box”, specifically, out of the “black box” of tools with built-in answers to the questions being asked.

Based on the city called *Ersilia*, described above, one student scenario read like this:

“*20 or so visitors, each dressed in full-body haptic feed-back suits and stereoscopic glasses, share a virtual space. The floor is a large checker-board of square tiles, that light up at each step taken, projecting images of all sorts. As soon as someone looks steadily at one of these images, it is frozen in space and a thread links up the image and the person. Ersilia is also capable of understanding thought and emotion; if two people look at each other and their hearts skip a beat, a colored thread is drawn between them as well. These virtual links create paths, but also obstacles, eventually trapping people in an elastic but resistant web. When it becomes impossible to move, Ersilia creates a holographic “double” of each visitor, freeing them to contemplate the city its maze of links from afar.”*

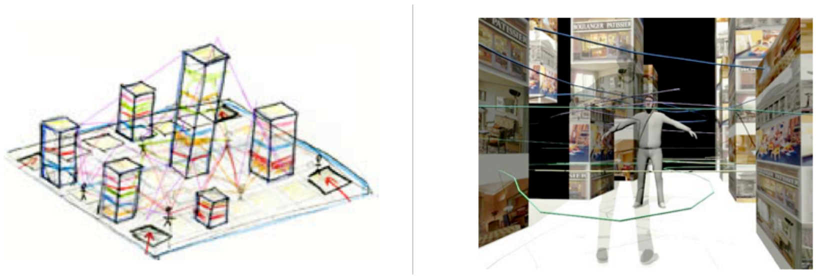


Figure7. Storyboard for Ersilia, of Calvino’s “Invisible Cities.” By students in game design, ESCIN, Laval.

The students’ interactive adaptation of *Ersilia* doesn’t simply propose a “décor” for a text. Their *Ersilia* is an environment that “behaves”: it shows initiative, independently of any “command” performed by a visitor; relationships between it and its visitors determine the way it evolves.

These links also have a physical impact on how people continue to move and relate to each other. Moreover, the project is open to a three-way exchange: people and the art-work, people with each other, and internally, among fragments of the work itself.

Calvino's *Ersilia* includes specific types of relationships (blood, trade, authority, agency…). Calvino does not describe these in detail. Game designers, however, need to find ways of explicitly factoring them in. In spite of instruments measuring heartbeats and capable of capturing thought, the students never managed to translate what "blood" or "authority" might mean in the context of an intelligent, autonomous, environment. Their problem was two-fold.

Reasons for behaving

First, the 3D rendering they "threw in"—even though specifically asked not to—numbed their imagination. *Ersilia* in 3D is more weighted down by gravity than by the links between people and objects. It reflects decisions made by tools that assume that horizons are a starting point, that people walk, that buildings are built sky-wards, etc., typical of gaming environments that hide the aesthetic nature of choice within a seamless geography. Everything here "works" fine.

The problem: *Ersilia* is made of seams, i.e. woven by the lines between people and things. In the student's renderings, these lines look and feel like traces left by bullets or strings of chewing gum. In Calvino's text, they hold more "weight" than the buildings themselves.

3D rendering techniques offer no ready-made solutions to "filling in" interstices, to giving more presence to the space between things than to things themselves. The analogical nature of the tools used by the students asphyxiated their aesthetic intelligence.

Second, concepts such as "authority" or "blood" had to be translated into behaviors and, specifically, behaviors that made sense in a virtual environment. This meant thinking about what it means to relate to someone "with authority", or how to move towards an image "as if it we wanted to buy it", or be looked at by a building as if we "belonged to it."

Getting perceived information to have a symbolic impact is not easy. A face looking straight at an image can mean several different things: attentiveness, fear, boredom, puzzlement… The author has to be clear about the assumptions that determine how signals are interpreted. This also means getting the viewer to understand (before her attention wavers) what in her presence is being taken into formal consideration by the work. Patterns need to be established that help the viewer "read" a larger, formal, scenario that defines the terms of co-presence within an augmented representation.

What are the terms of the "aesthetic and social contract" between the work and its public?

Staging reciprocity

An indirect answer to this question lies in a description of how to get students to imagine living-art scenarios on their own, without standing on the shoulders of great thinkers such as Raymond Queneau and Italo Calvino. We shall describe, briefly, the tenor of one of the classes taught at the "Living Art Seminar", initiated by Florent Aziosmanoff, Art Director at the CUBE, Paris.

One of the classes begins with an analysis of the formal conventions that pertain to a selection of works by photographers, sculptors, musicians, dancers, etc. Students are asked to retain key concepts about how each work structures space and maintains the public's attention. They are then asked to write a sentence synthesizing one or a combination of the works analyzed. The sentence is supposed to be "personal," in the hope of bringing the student's imagination to mitigate the very formal approach to the class.

Each sentence is a springboard for a "living-art scenario". Some examples of sentences: "Narcissistic trio opens sand"; "Spiral bird-flight lights against presence"; "Blue silhouette jiggles against yellow signature"; "A ruptured shadow transcends insults"…. Not all the quotes are clearly traceable to a given art work. “Spiral bird-flight lights against presence” was inspired, in part, by Robert Smithson’s *Spiral Jetty*. It became the following living-art scenario:

“A spiral of light is projected on the wall. Left alone, it has a tendency to dissolve. When a visitor appears, the spiral tightens up. When more visitors appear, the spiral starts turning. The tone of the small-talk among visitors affects the spiral’s behavior: when it is upbeat, the spiral tends to turn rapidly and projects holograms of sparks in the visitor’s space; when it is downbeat, the spiral retracts and quiets down. Once spectators understand the dynamic, a balance of power emerges: the visitors become musical instruments of sorts, and the spiral their conductor.”

Once a scenario has been written, the class tries to make the scenario “come alive”. A group of students act out the part of the living-art work, another group takes on the role of the public. A relational “dynamic” emerges from this dance, each side taking its cue from the other. This allows a relationship between the work and its public to be articulated through gesture and in conversational terms. As people talk to each other, they also talk to the work, who (?) joins in. The model is that of an exchange, not a “command”: choices exist, but they are subsumed to the rhythms of conversation. With this in mind, students examine each of the stages of the communication process. They begin by defining how the work evolves of its own accord, with no one around; if visitors show up and start talking, they enact the factors that cause the spiral’s behaviour to evolve. The terms of recognition and reciprocity are articulated through these movements.

An aesthetics of living-art?

Through continued “conversational” exchanges of this sort, an aesthetic relationship is defined and deepened. *Symbiose*, by the Collectif Experientiae-Electricae, will serve to conclude this chapter on “staged reciprocity”. The work is currently featured at the Armenian Pavillion at the World’s Fair in Shanghai, 2010. *Symbiose* simulates artificial life on earth. The terrain includes bacteria that evolve on land, in volcanoes and in the ocean.

Figure 8. “Symbiose”, Collectif Experientiae-Electricae (2007), co-produced with the CUBE, France.

If the “system” runs out of water, the bacteria die; too much water and the grass disappears. The work doesn’t respond to touch (the central image is misleading in that respect). Visitors are signalled on screen by little graphic symbols that correspond to sms they can send with their mobile phones. These symbols participate in *Symbiose’s* eco-system: the messages sent by viewers live off the bacteria; too many messages imperil the whole system.

Here, representations “feed” off of algorithms. The cross-over between “image” and “artificial life” is explicit. But is the approach “aesthetic”, or are we still mired in a “wow” effect? Experientially, we are faced with a coherent, complex world. Yet a troubling lack of “material abstraction and conceptual concretization” persists. Intelligence here is showy. Its mechanisms are “transparent”. This living-art is mired in mimesis, with little of the synthetic complexity characteristic of Queneau or Calvino. At best, these works assemble a “constellation of analogies”. The artistic intelligence is not quite there…yet.

References

Aziosmanoff, F. (2010). “Living Art”. Paris: CNRS Editions.

Calvino, I. (1978). *Les Villes Invisibles*. Paris: Editions du Seuil.

Coyne, R. (2005). *Designing Information Technology in the Postmodern Age, From Method to Metaphor*. Cambridge: The MIT Press. pp. 97, 265, 286.

Jeanneret,Y., Souchier, E. (1999) “Pour Une Poétique de l’Ecrit d’Ecran.” *Xoanna*, No6, Paris.

Queneau, R. (1973). Un conte à votre façon, in *Oulipo, La literature potentielle (Céations, Re-créations, Récréations)*. Paris: Gallimard, coll. “Folio/Essais”.

Getting to Know Scratch

Mitchel Resnick, *mres@media.mit.edu*
Karen Brennan, *kbrennan@media.mit.edu*
MIT Media Lab

Introduction

Scratch builds upon decades of research on Logo and other constructionist programming environments. Available free of charge, Scratch aims to distinguish itself from other environments by making programming *more tinkerable* (enabling learners to create programs by snapping together graphical blocks), *more meaningful* (adding programmability to the media-rich activities that are popular in today's youth culture), and *more social* (providing young people with the opportunity to remix and build upon one another's ideas, images, and programs).

With Scratch, young people can create their own interactive stories, games, animations, and simulations – and share their creations with one another online (Figure 1). The Scratch website (http://scratch.mit.edu), launched in 2007, has become a vibrant online community, with members (mostly ages 8 to 16) sharing, discussing, and remixing one another's Scratch projects. With more than 1,000,000 projects, the collection of projects is incredibly diverse, including science simulations, virtual tours, newsletters, adventure games, animated dance contests, and interactive tutorials. As young people program and share Scratch projects, they develop as computational thinkers: they learn important computational and mathematical concepts, as well as strategies for designing, problem solving, and collaborating.

Figure 1. The Scratch programming environment, sample projects, and sample blocks stack.

Workshop activities and expected outcomes

The Scratch workshop at *Constructionism 2010* will involve a mixture of presentation, hands-on activity, and discussion. The workshop will begin with an overview of the ideas and motivations underlying the design of Scratch, and analysis of how Scratch has been used in different contexts and settings. Participants will then have the opportunity to create their own interactive projects with Scratch, reflect on their learning experiences, discuss how Scratch compares with other constructionist environments, and brainstorm about future directions for the design and use of Scratch. We welcome participation by people of all backgrounds (educators, researchers, developers) and all levels of Scratch experience (novice to expert).

Keywords

Scratch; programming; learning; collaboration; community; media

FirstBridge Under Construction: Me and My Avatar

Georgi Stojanov, *gstojanov@aup.fr*
Dept of Computer Science, Maths, and Science, The American University in Paris

Charles Talcott, *ctalcott@aup.fr*
Dept of Global Communications, The American University in Paris

Sandra Bruno, *sandra.bruno@iufm.u-cergy.fr*
Université de Cergy Pontoise IUFM de Versailles

Abstract

The main goal of this article is to give an overview of our experience of designing and teaching the first-semester learning community "FirstBridge" set of courses at The American University in Paris (AUP) - *Social Robotics* and *Self and the Automated Other* - from a constructionist point of view. We would like to go beyond the typical course assessment and offer our reflections on what and how the students (and the professors) have been learning, and, on a meta-level, what they think they have been learning.

One of the key elements of Piaget's genetic epistemology is that cognitive development occurs in stages gradually moving from concrete "here and now" sensory-motor experiences via concrete operations towards internalized abstract, formal operations (Piaget, 2000). In some way, this abstract high-level knowledge is seen as "superior" to the knowledge acquired in physical interactions. However, as Ackerman (1996) observes: "[A]n increasing number of psychologists and cognitive scientists have adopted the view that knowledge is essentially situated and thus should not be divorced from the contexts in which it is constructed and actualized. […] <u>It challenges the prevalent view among developmentalists (such as Piaget and Kohlberg) that removed, analytical modes of thought are necessarily more advanced forms of cognitive functioning</u>. It questions the notion that cognitive growth consists in an uni-directional progression from concrete to abstract, from fusion to separation […]." (*emphasis added*)

What Piaget neglected, in some sense, is that the cognitive development of adults continues further to levels of deeper understanding (c.f. Campbell and Bickhard, 1993) and it is virtually always based on physical operations that transform internal/abstract ideas into tangible and sharable objects. These created objects allow for a new way of looking at one's internal constructs and thus enable one to take different perspectives as well as undertake different types of manipulations. Most significantly, the cognitive loop is closed via the social space: the created objects are visible and manipulable by others. With this in mind, we have tried to create a FirstBridge that enables assimilation and accommodation within the social context of a "learning community". Students can collectively see and discuss what they constructed, check what they learned or what skills they acquired, and then extend and transfer these skills to different contexts. The structure of both courses and their interaction is outlined.

This FirstBridge has had a positive impact on getting students interested in ICT, mathematics, and natural sciences in general, as revealed by increased number of ICT majors as well as students' testimonials during one-on-one interviews at the end of the semester. We intend to track all students that took this FirstBridge until their graduation and see the long term effects with respect to students who have taken different FirstBridge courses.

Keywords

Mind, modelling, analogy, scale models, Lego NXT Mindstorms, Alice programming environment

FirstBridge Under Construction: Me and My Avatar

Georgi Stojanov, *gstojanov@aup.fr*
Dept of Computer Science, Maths, and Science, The American University in Paris

Charles Talcott, *ctalcott@aup.fr*
Dept of Global Communications, The American University in Paris

Sandra Bruno, *sandra.bruno@iufm.u-cergy.fr*
Université de Cergy Pontoise IUFM de Versailles

Abstract

The main goal of this article is to give an overview of our experience of designing and teaching the first-semester, learning community "FirstBridge" set of courses at The American University in Paris (AUP) - *Social Robotics* and *Self and the Automated Other* - from a constructionist point of view. We would like to go beyond the typical course assessment (that normally assumes a fixed set of well-defined course goals) and offer our reflections on what and how the students (and the professors) have been learning, and, on a meta-level, what they think they have been learning.

The paper opens with a brief introduction about the concept of FirstBridge at AUP. This section is followed by a short summary of the place and role of constructionism within our teaching/learning philosophies. In the next section we highlight a few of the constructionist dimensions within the two courses. We then present some examples of the student projects, as well as their feedback collected via anonymous course evaluation (at the end of the semester) and via informal one-on-one interviews.

In the conclusion, we summarize and evaluate our application of constructionist methodology in the classroom, highlight the successes and attempt to address the observed shortcomings.

Keywords

Mind, modelling, analogy, scale models, Lego NXT Mindstorms, Alice programming environment

Introduction

The first author of this paper has an extensive research experience in the domain of artificial intelligence and particularly in the field of developmental robotics (e.g. Stojanov et al 1997; Stojanov, 2001; Stojanov et al 2006). The author's research focused on the application of Jean Piaget's insights from developmental psychology to the construction of artificial intelligent agents, in order to gain insight into the fundamental processes of learning and cognitive development in artificial and natural systems. The opportunity to teach a FirstBridge course immediately resonated with this past experience as the "learning community model" provided a pedagogical challenge where we could apply these insights to a classroom context. In the following subsections we briefly introduce the concept of FirstBridge at AUP and then the philosophy of the authors' particular FirstBridge: Me and My Avatar.

What is FirstBridge at AUP?

Our experiments in constructivist pedagogy occur within a specially designed General Education curricular structure named "FirstBridge". FirstBridge is a collaborative "Learning Community" for first-semester freshman at The American University of Paris and is comprised of two courses from different disciplines that are thematically linked and that take different disciplinary approaches to diverse issues such as global warming, artificial intelligence, human rights, civil conflict, visual thinking, food, to name but a few of them. The reader can find an excellent description of the general goals of the FirstBridge concept in (Clayson, 2008).

In our case, we have created and combined two distinct courses for our “Me and My Avatar” Learning Community: one from Computer Science, CS 221 Social Robotics, and one course from the Humanities, CL 100, that draws on Philosophy of Mind, Comparative Literature, and Global Communications and Media Studies.

One of the curricular specifications of the FirstBridge program was to introduce first semester students to complex, high-level problems and to devise task-based lesson plans. One of the primary learning objectives is for students to explore creative problem-solving within an interdisciplinary framework while sharpening the fundamental academic skills of critical reading, writing, and thinking. The students meet twice a week for each course plus one period per week for and interdisciplinary Reflective Seminar where the connections between the two courses are explored. The Reflective Seminar is also the place where the students, with both professors, try to “step-out” and reflect about what they have been learning, thus “constructing a public entity” out of the knowledge and the skills.

The philosophy behind the Me and My Avatar FirstBridge

The pedagogical approach of this FirstBridge, partially based on the insights gained during our research in the cognitive development of artificial and natural system, is essentially constructionist. Although we adopt the general big picture about the development of knowledge via the Piagetian processes of assimilation and accommodation we strongly believe that physical and social interactions in adult cognitive development play a far bigger role than suggested by Piaget.

One of the key elements of Piaget’s genetic epistemology is that cognitive development occurs in stages gradually moving from concrete “here and now” sensory-motor experiences via concrete operations towards internalized abstract, formal operations (Piaget, 2000). In some way, this abstract high-level knowledge is seen as “superior” to the knowledge acquired in physical interactions. However, as Ackerman (1996) observes:

“[A]n increasing number of psychologists and cognitive scientists have adopted the view that knowledge is essentially situated and thus should not be divorced from the contexts in which it is constructed and actualized. […] <u>It challenges the prevalent view among developmentalists (such as Piaget and Kohlberg) that removed, analytical modes of thought are necessarily more advanced forms of cognitive functioning</u>. It questions the notion that cognitive growth consists in an uni-directional progression from concrete to abstract, from fusion to separation […].” (*emphasis added*)

What Piaget neglected, in some sense, is that the cognitive development of adults continues further to levels of deeper understanding (c.f. Campbell and Bickhard, 1993) and it is virtually always based on physical operations that transform internal/abstract ideas into tangible and sharable objects. These created objects allow for a new way of looking at one’s internal constructs and thus enable one to take different perspectives as well as undertake different types of manipulations. Most significantly, the cognitive loop is closed via the social space: the created objects are visible and manipulable by others; a fact that gives the person who created them the possibility to gain distance and to relate to these objects from a different perspective. In addition, our minds opportunistically extend themselves via these artefacts in the sense suggested by Andy Clark (e.g. Clark, 2004).

With this in mind, we have tried to create a FirstBridge that enables assimilation and accommodation within the social context of a “learning community”. We start by identifying a concept/a situation/an idea and then devote a significant portion of class time to building **models**

of those concepts (using different media: from physical scale models to virtual worlds). The objective behind this is to create object-referenced learning opportunities for further group discussion and reflection. Students can collectively see and discuss what they constructed, check what they learned or what skills they acquired, and then extend and transfer these skills to different contexts.

To quote Ackerman (1996) again:

"People cannot learn from their experience as long as they are entirely immersed in it. There comes a time when they need to step back, and reconsider what has happened to them from a distance. They take on the role of an external observer, or critic, and they revisit their experience "as if" it was not theirs. They describe it to themselves and others, and in so doing, they make it tangible and shareable." (*emphasis added*)

Ackerman (1996) calls this movement of immersion and distancing a "dive-in and step-out" process. When developing our FirstBridge, one of our main considerations towards the application of a constructionist learning approach was to create a series of controlled learning experiences where our students could "dive-in and step-out".

The Practice of Me and My Avatar

As mentioned above, the notion of a "model" is crucial for this FirstBridge. Here, it is used to cover the full gamut from physical scale models, to stories as a particular way of describing one's experience, to metaphors as models of one situation in terms of another, to scientific theories as models of particular part of the world. We have designed a series of assignments where the students build models of increasing complexity.

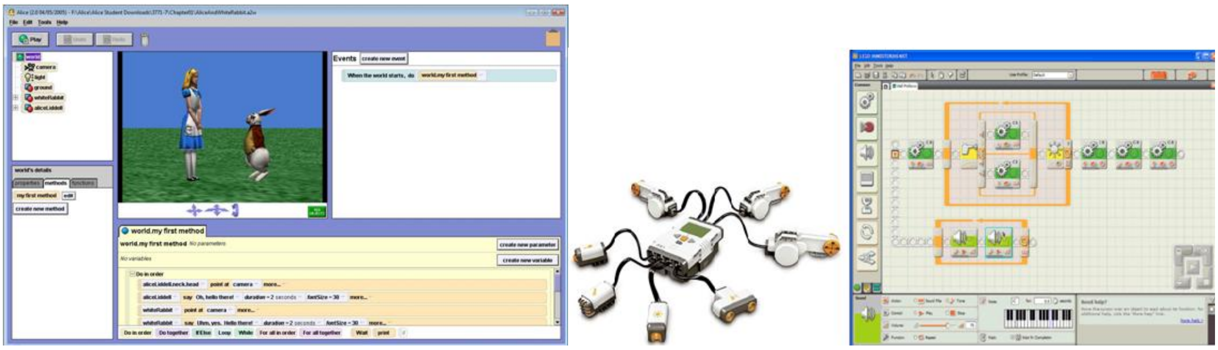
In Social Robotics, the **first assignment** in the series is to **build a scale model of the ground floor** of the University building. The only criterion, students are told, is that the scale model should be recognizable as the ground floor of the University building. Below (Figure 1) are four examples of these scale models.

Figure 1. Samples of students' projects for the assignment "Build a scale model of a University ground floor level"

When these models are discussed in class, we point out to our students how, for the very same building, each of us constructs his or hers own model of the building: stressing some parts while completely omitting others (e.g. the chess board like tile floors, the walls and the windows).

These processes, alongside with embellishment, simplification, and addition (as elaborated by Goodman, 1978) are universal to all modelling activities. Goodman's 1978 book *Ways of Worldmaking* is first in the series of assigned readings, and the students often recognize the above mentioned processes during their work on this first assignment. This is discussed during one of the first reflective seminars.

The second **assignment is to build a virtual model of the same ground floor** as in the previous assignment using the Alice programming environment. Alice is a friendly 3D programming environment with an intuitive drag-and-drop interface (Figure 2a). In essence, it is a fully-fledged Java based programming language that allows students to learn the basics of object oriented programming. In the FirstBridge context, our main goal is not to teach the basics of OOP but to give students a tool for creating models, in this case, appealing 3D movies, or virtual worlds.



a) b)

Figure 2. a) A screenshot of the Alice programming environment (more info on Alice available at alice.org; b) Lego NXT kit (programmable brick, sensors and motors) and the visual programming environment (more info available at mindstorms.lego.com)

By drawing analogies with the previous assignment, students easily adopt the language of *object* oriented design: virtual objects are similar to the physical ones and virtual environment makes it easier to create and manipulate them. While talking about Alice programs, we adopt the "programs as movies" metaphor suggested by Alice creators (Dann, Cooper, Pausch, 2005; see also Adams, 2006). After this first, fairly simple virtual model, we move further on building quite complex programs: from short movies to highly interactive games. This is done gradually by introducing students to methods and functions as ways to change objects' behaviours and to interact with them.

Finally, **the third assignment is to build and program a given behaviour for a Lego NXT Mindstorms kit robot** (Figure 2b)**.** Robots, in this context, are more complex models that consist of physical parts, and the program that controls the behaviour of the robot (or its virtual part).

After the introductory tutorial, we move on and construct "intelligent" mobile robots inspired by Braitenberg's vehicles (Braitenberg, 1982). Again, students find the transfer of skills from the Alice environment very natural and helpful in understanding the new medium.

As it can be seen, differences among these models vary according to different dimensions: physical or virtual, static or interactive. From the simplest one: physical and static to the most complex ones that are partly physical partly virtual *and* interactive.

Within the Humanities-based companion course, *Self and the Automated Other*, the challenge is to overcome the 'instructionist' (Fischer et al., 2007) pedagogical tradition and its formal 'idealism' by creating constructionist assignment sequences that become increasingly 'materialist'. The course begins with an exploration of the concept of "concepts", which students are encouraged to sketch, draw, figure and 'model' in order to apprehend "concepts" not as abstract intangibles, but as concrete, mental "building blocks" with various affordances.

Course readings are presented less as ideas and theories that are to be learned by rote and placed into the students' "exam-ready retrieval systems" than as guides to and examples of various constructionist approaches to modelling the self and consciousness, which have contributed to the development of a disciplinary history. For example, while reading René Descartes' *Meditations* students re-perform Descartes' famous experiment with wax (but this time with French chocolates) in order to re-construct their own experience-based model of Cartesian rationality. Students use Venn diagramming to model the "mind-body problem" within the history of philosophy of mind; construct maps and timelines to depict the narrative sequences of films such as *Memento* (Christopher Nolan 2008); and eventually produce "full-scale" multimedia models of once-abstract philosophical "concepts" for their final projects.

During the reflective seminars we organize discussions where students are urged to look for connections between the two courses and apply concepts and skills that they acquired in one course to the content of the other course.

Students' final projects have included video performances, paintings, an opera complete with a musical score and recorded songs, blue-tooth controlled Lego robots emulating Braitenberg 'vehicles', etc. In the course, the students move from abstraction toward a collectively-constructed concretization of philosophical and literary 'ideas' and 'concepts'.

The core concept, theme or task that links the two courses is that of 'Modelling', but, as evidenced above, at many different levels: from perceptual phenomena to aesthetic and literary depictions; from commonsense thinking and perception to philosophical reflection and scientific reasoning.

Conclusions and discussions

In the above case study, we have tried to summarize our approach in designing and teaching the FirstBridge Me and My Avatar as a part of the general education requirement of the students at the American University of Paris.

In conclusion, we would like to say something on the student feedback. Overall, as measured by the anonymous evaluations at the end of the semester, student feedback has been positive. At times, we have encountered some resistance at the beginning of the semester but this resistance was mitigated by the end of the first month. This FirstBridge has had a positive impact on getting students interested in ICT, mathematics, and natural sciences in general, as revealed by increased number of ICT majors as well as students' testimonials during one-on-one interviews at the end of the semester. We intend to track all students that took this FirstBridge until their graduation and see the long term effects with respect to students who have taken different FirstBridge courses.

As a general observation, to our knowledge, no one has made the connection so far between Clark's thesis of the extended mind (e.g. Clark, 2004) and Papert's constructionism. We believe that further exploration of this link will enhance our understanding of the crucial mechanisms for knowledge development.

At a meta-level, the exercise of the FirstBridge was an excellent opportunity to "dive-in and step-out" from the immediate experience of teaching a course (in computer science, for example). It has definitely been an enriching experience and we are looking forward to the disequilibrational adaptation. The present paper is a result of this process.

For more samples of student projects please visit ac.aup.fr/gstojanov/CONSTRUCT10

References

Ackermann, E. (1996). *Perspective-Taking and object Construction.* In Kafai, Y.B., and Resnick, M. (Eds.) (1996) *Constructionism in practice: designing, thinking, and learning in a digital world.* Lawrence Erlbaum Associates, Mahwah, New Jersey. Part 1, Chap. 2. pp. 25-37

Ackermann, E. (2001) *Piaget's Constructivism, Papert's Constructionism: What's the difference?* In the proceedings of the CONFERENCE ON CONSTRUCTIVISM: USES AND PERSPECTIVES IN EDUCATION, Vol. 1 & 2, Geneva, Research Center in Education, Cahier 8, September 2001, pp. 85-94.

Adams, J. (2006) *Alice in Action: Computing Through Animation*, Course Technology.

Braitenberg, V. (1984) *Vehicles: Experiments in Synthetic Psychology* Bradford Book / The MIT Press, 1984.

Campbell, R. L., Bickhard, M. H. (1993*). Knowing Levels and the Child's Understanding of Mind.* Behavioral and Brain Sciences, 16(1), 33-34.

Clark, A. (2004) Natural-Born Cyborgs; Minds, Technologies, and the Future of Human Intelligence. Oxford University Press, USA

Clayson, J. (2008) *Radical bricolage: building coherence in the liberal arts using art modeling and language*. In International Journal of Education Through Art, Vol. 8 (2) pp. 141-161.

Dann, W. P., Cooper, S., Pausch, R. (2005). *Learning to Program with Alice.* Prentice Hall.

Fischer, G., Rohde, M., Wulf, V. (2007) *Community-based learning: The core competency of residential, research based universities*. International Journal on Computer- Supported Collaborative Learning (iJCSCL), Vol. 2, No. 1. pp. 9-40.

Goodman, N. (1978) *Ways of Worldmaking* Indianapolis: Hackett.

Kafai, Y.B., and Resnick, M. (Eds.) (1996) *Constructionism in practice: designing, thinking, and learning in a digital world.* Lawrence Erlbaum Associates, Mahwah, New Jersey.

Piaget, J. (2000) *Studies in reflecting abstraction* (edited and translated by Robert L. Campbell) Hove: Psychology Press.

Repenning, A., Basawapatna, A., Han Koh, K. (2009) *Making University Education more like Middle School Computer Club: Facilitating the Flow of Inspiration.* WCCCE'09, May 1–2, 2009, Burnaby, BC, Canada.

Stojanov, G., Bozinovski, S., and Trajkovski, G. (1997*). Interactionist-expectative view on agency and learning*. IMACS Journal for Mathematics and Computers in Simulation 44(3):295–310.

Stojanov, G., (2001) *Petitagé: A case study in developmental robotics.* Proceedings of the 1st International Workshop on Epigenetic Robotics, Lund University Cognitive Studies, 85, Lund, Sweden

Stojanov G., Trajkovski G., Kulakov A., (2006). *Interactivism in artificial intelligence (AI) and intelligent robotics.* New Ideas in Psychology, Elsevier, Vol. 24, Issue 2, 163–185.

Early introduction to algebraic thinking in technological environments

Cristianne Butto-Zarzar, *cristianne_butto@hotmail.com*
Universidad Pedagógica Nacional - Ajusco, Mexico

Ana Isabel Sacristán, *asacrist@cinvestav.mx*
Dept of Mathematics Education, Cinvestav, Mexico

Abstract

A number of authors have investigated the feasibility of introducing young children to algebraic ideas: e.g. generalized arithmetic (Mason *et al.*, 1985); meaning of operations (Slavit, 1999); generalisation and progressive formalisation (Blanton and Kaput, 2002); algebra as a representation tool and resolution of problems (Da Rocha Falcặo, 1993); design of tasks to support algebraic thinking in elementary school (Blanton and Kaput, 2002); operating with the unknown (Carraher *et al.,* 2001); algebra from a symbolisation point of view (Kaput *et al.,* 2008) and the transition from arithmetic to algebra and the use of symbolic generalisations in a computer intensive environment (Tabach *et al.,* 2008).

This paper reports on an on-going study on early introduction to algebraic thinking in students of elementary school in Mexico, based on a teaching model that incorporates two routes of access: proportional reasoning and generalisation processes. The choice for the first route (proportional reasoning) is based on the familiarity that children have with this mathematical content at the fifth grade of elementary school. The second route examines the fact that the mathematical content is linked conceptually and historically (Radford, 1996) to functional variation. It's worth noting that at this level most of the students are in transit from additive to multiplicative thinking. The experimental work – which is currently in its second phase – involves paper-and-pencil, Logo and Expresser activities. For this, a teaching sequence was developed; pre- and post-questionnaires, as well as clinical individual interviews, were used to complement the data collection. For the construction of the teaching model, a theoretical framework was used that relies on the Local Theoretical Models (LTM) of Filloy (1990) and Filloy, Rojano and Puig (2008), and also takes into account Vygotsky's (1978) idea of Zone of Proximal Development (ZDP).

The study is situated at the end of the elementary school curriculum, in the fringe of pre-algebraic thinking, where algebraic syntaxes have not been introduced yet. In this study the algebraic ideas are introduced through a teaching sequence in two versions: pre-symbolic (perception of the idea of proportional variation) and symbolic in Logo and/or eXpresser environments (find and express a general rule, as well as incorporate it). The use of those technological environments, which have graphical, numerical and programming properties, can allow activities involving pattern recognition, where students can go from particular cases to expressing general rules and to testing those rules.

The first phase results have revealed that students are capable of understanding ideas of proportional variation, that they can discover a pattern and formulate a general rule, while they transit from additive to multiplicative thinking.

Keywords

Early algebra in primary school, generalisation processes, Logo

Introduction

Theoretical and empirical studies have shown that the transition from arithmetic to algebra is an important step needed to access more complex ideas in mathematics, and that a series of obstacles have to be overcome in order to master the notion of symbolic algebra. Some results suggest that it may be possible to overcome or avoid these obstacles depending on the way algebraic thinking is conceived and the way that early algebra is introduced in early stages. It is also believed that if the routes of access are familiar to students – such as proportional reasoning in primary school – and are specifically situated within the curriculum – for example within the 5^{th} and 6^{th} years of primary school – students are able to access early algebraic thinking even though their mathematical reasoning is in transit from additive to multiplicative reasoning. On the other hand, it is well known that didactical times for the learning of algebra are long and it seems appropriate to initiate students to early algebraic thinking at early ages (7-11 years old), taking advantage of the sources of meaning that the curriculum contents in primary school offer. As a reaction to these ideas, many authors have focused on research on early algebraic thinking through different perspectives, as cited in the abstract in the previous page.

A study for early introduction to algebraic thinking in technological environments

The study reported here is on early algebraic thinking but stands apart from the works previously cited, in that it proposes a conceptual route that does not break with numeric or algebraic thinking. It is an introduction to algebraic ideas where an algebraic symbolisation is not necessary reached, although sometimes it can be attained.

The research is situated at the end of the curriculum of primary school, in the layer of pre-algebraic thinking where the students have not yet been introduced to algebraic syntax. In the present study, algebraic thinking is approached from the point of view of proportional reasoning and generalisation processes. The main purpose is to develop an alternative route towards building a teaching model that allows students to transit from additive to algebraic thinking, incorporating sources of meaning such as proportional reasoning from the curriculum. Thus, algebraic ideas are introduced along two main lines:

1.- Pre-symbolic – using the idea of proportional variation and symbolic – where the general rule has to be found and expressed by means of a series of problems in a didactical sequence.

2.- Starting from proportional reasoning, which is considered a part of the multiplicative field, we develop further this mathematical idea towards proportional variation, variable as a functional relationship and general number by means of generalisation processes. Generalisation processes mean that students are involved in the detection of patterns, and it is helpful for them to represent the pattern by means of a rule, an entry point to the symbolic.

Going from particular cases to the general rules and testing them, can be achieved in practice by exercising pattern recognition in a graphical, numerical and programming environment where shapes-detection, similarity, repetition and recurrence can be easily manipulated, by the students themselves. We consider environments such as Logo and eXpresser as ideal for those purposes, as is discussed further down.

Aims of the study

Thus, the aims of our study can be summarised as follows:

- To study early algebraic thinking with students of the last years of primary school (grades 5 and 6) in technological environments of learning.

- To design and implement sequences of activities with digital technologies, exploring the two aforementioned routes of access to algebraic thinking: proportional reasoning and generalisation processes.
- To test several modalities of the use of digital technologies in the classroom.

The technological environments of learning

As stated above, for our project, two technological environments have been chosen: Logo and eXpresser. Many studies have investigated the potential of Logo for mathematical learning, including algebra learning (e.g. Hoyles and Sutherland, 1987, 1989; Ursini, 1993), including one of our own studies (Butto, 2005). In that latter study, the potential of Logo to facilitate the understanding of, specifically, proportional reasoning in 11-14 year-old children working collaboratively in pairs, was investigated.

We also like the fact that Logo gives children more autonomy on their own learning (Hoyles and Sutherland, 1989). Schoenfeld (1985; cited in Hoyles and Sutherland, 1989), stresses the role of meta-cognition when students are led to think on their own actions and thoughts; they assume self-control on their activities, are capable of taking their own decisions, can change their strategies and the way they organise and solve the problems. Logo is an ambient where heuristics and mathematical ideas are recreated (Noss, 1986). Thus, Logo creates a bridge between students' actions and their understanding of the mathematical relations that they require to write a program. In this way, children are capable of capturing their understanding in symbolic form and clarifying it with the aid of the computer.

In the present study, Logo was used in parts of the preliminary phase, which we present here.

For the second phase, we are in the process of developing activities for studying generalisations processes with eXpresser. The eXpresser microworld is a free java-based software of the MiGen project (*Intelligent Support for Mathematical Generalisation*; http://migenproject.wordpress.com/), which is led by Richard Noss and Alex Poulovassilis in the UK. This software "seeks to provide students with a model for generalisation that could be used as a precursor to introducing algebra" (Geraniou *et al.*, 2009). Following Papert's constructionist paradigm, eXpresser provides several approaches that allow students to construct their own mathematical models: in eXpresser, students can build figural patterns of square coloured tiles and express the rules underlying them. Thus, in this microworld, children can work in a numeric, geometric and programming environment, and make use of patterns or regularities correspondingly to their inputs when they create a program and make sense of what they are doing when they validate their predictions.

The design of the research and teaching sequence

The work presented here uses as theoretical framework the Local Theoretical Models (LTM) perspective proposed by Filloy (1990) and Filloy, Rojano and Puig (2008), which include four components: 1) the Teaching Model; 2) the Cognitive Processes; 3) the Formal Competence Model and 4) the Communication Model. In this project the focus is mainly on the first two: the teaching model and the cognitive process.

It is the aim to design a teaching model where, in the learning sessions, children work in several environments: paper-and-pencil, Logo (and later also with eXpresser), in order to cover the two alternative routes of access to algebraic thinking that we have proposed – as outlined before, and shown in Figure 1 – thru a teaching sequence that is applied as a mean to promote access to initial algebraic notions.

That is, we also use the Vygotskian idea of Zone of Proximal Development (ZPD) – which Vygotsky (1978) defines as the distance between the level of current child's development and the higher level of potential development – in that the didactic sequence is intended to help students in their development through their ZPD. It is thus important to determine the level of potential development and the level of current development. For that we explore and analyse the children's zone of current development and the evolution toward the first algebraic ideas through the application of pre- and post- questionnaire and ad-hoc interviews. These questionnaires and interviews give us insights into children's initial and later notions, and ZPDs, about proportional reasoning and generalisation processes.

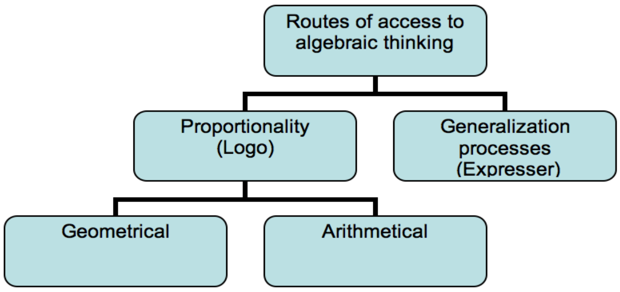


Figure 1: The two routes of access to algebraic thinking

In terms of the routes of access to algebraic thinking, we base the first route (proportional reasoning) on the familiarity that children have with this mathematical content at the fifth grade of elementary school. Another reason is the fact that the mathematical content is linked conceptually and historically to the idea of functional variation (Radford 1996). It is worth reminding the readers that at this level most of the students are in transit from additive to multiplicative thinking.

For the task design, we used Mason's (1985) idea about generalizing in algebra through four stages: 1) perceive a pattern, 2) express the pattern, 3) record the pattern and 4) test the validity of the formulas. Several researchers, such as Ursini (1993), assert that children find it difficult, when working with numerical patterns, to describe and express a pattern algebraically. Pegg (1990; cited in Durán Ponce, 1999) mentions that the discovery of patterns requires three processes: to experiment with numerical patterns; to express the rules by means of explanations and to encourage students to express their rules in a abridged way. Hoyles and Sutherland (1989) argued that the numerical and geometric environment of Logo allows children to observe numerical and geometric patterns and build general rules in algebraic or pre-algebraic terms.

Methodology

The study is being carried out with 20 students (10-11 years old) of the 5th and 6th grades of elementary school. Students in this age-group tend to privilege mathematics contents belonging to the field of additive structures.

As stated above, (pre- and post-) questionnaires were designed to explore children's numerical thinking; specifically to explore proportional thinking and generalisation processes. The didactic sequences are meant to develop those algebraic thinking processes. The activities of the teaching sequence are carried out with paper-and-pencil, as well as some with Logo. (Note: In the phase reported here, we only use Logo; in a later phase, we will also incorporate eXpresser activities).

The contents of the questionnaires and activity sequences consist of problems related to proportional reasoning and/or generalisation, with variables as general numbers or in functional relationships. In many problems or activities, children are asked to complete tables of values. One questionnaire is on proportional reasoning, and involves problems such as identifying proportional figures; completing visual sequences that follow proportional patterns; constructing proportional figures; and problems with liquid proportions. Another questionnaire is on processes of generalisation and functional variation: the problems involve things like completing terms of arithmetic and geometric sequences; ordering “cards” of values of weights and heights of children; analysing the increase of production of a plastics machine (see Questionnaire Problem 4, below); monetary distribution amongst several people in different proportions; and completing sequences of figures.

We have two didactic activity sequences. The first is on proportional reasoning and includes activities that involve drawing, with Logo, different sizes of squares and of other figures (such as chairs and tables) keeping the proportions (and observing the similarities in the procedures); identifying figures that are in the same proportions (e.g. figures of persons, of tables); finishing a drawing of a car that is proportional to a given one.

The second didactic sequence is on generalisation processes and involves: * Drawing with Logo different sized letters (such as ‘E’s), first in a sequence, then writing a general procedure for any given size. * Completing sequences of figures and of polygonal numbers and finding the general rules. * A problem involving a horse race, where each horse starts at different time and runs at different speeds. * Drawing Logo squares in different sizes, observing the invariants and the variable values and writing a general procedure. * Experimenting with a recursive Logo procedure for drawing a tree with as many branches as given by a variable. * Experimenting with a Logo procedure that draws a spiral star, and uses 3 parameters.

In order to study social interaction during the working sessions of the didactic sequences, a model of mathematical discussion, consisting of the following four components, is used: 1) Individual and collective presentation of different solutions. 2) Individual reconstruction of the process of solution of a problem. Students mention their own strategies and abandon those that are not efficient. 3) Collective exposition of the new knowledge. Students are asked to share what they have learned, comparing situations and beliefs. 4) Institutionalisation of knowledge.

According to Tudge (1992), social interaction among pairs promotes information within the ZPD, promotes cognitive development and leads thinking in children to progress towards adult models within a cultural practice. In this process of collaboration, students learn meanings, behaviours and adult technologies.

After the working sessions, students are given a post-questionnaire, then the children are interviewed in order to verify the evolution of algebraic ideas and confirm the results obtained in the questionnaires and from their worksheets.

Answers to the initial questionnaire were categorised initially in levels of mathematical conception (high, medium and low):

- *High level*: is characterised by the comprehension of proportional reasoning, functional variation and generalisation processes. Thinking is in algebraic or pre-algebraic terms.
- *Medium level*: is characterised by a transitional thinking, which goes from the use of additive or multiplicative resolutions, either in proportional reasoning or generalisation processes. Transitional: from arithmetical to pre-algebraic thinking.
- *Low level*: is characterised by the use of purely additive strategies and students present difficulties in understanding problems of proportional reasoning as well as generalisation processes. Purely additive thinking.

The first level of analysis includes the strategies used in the solution of the problems (Logo and paper-and-pencil), obtained by the analysis of students’ worksheets, and observations during the didactic sequence; these strategies were categorised as: arithmetic-multiplicative, incomplete-multiplicative and complete- multiplicative. The second level of analysis includes the social intervention in pairs during the working sessions and was classified according to Cobb (1994): univocal explanation, multivocal explanation, direct collaboration and indirect collaboration. The third level of analysis included the cognitive processes followed by the pairs in the solution of the problems by means of clinical interviews and cognitive maps with teaching. In general, the analysis of the longitudinal study is being done along two components of the LTM: the didactic and the conceptual models.

Results

Sample data from paper-and-pencil activities from the initial questionnaire

Questionnaire Problem 2

In the activity shown in Figure 2, children had to solve a word-problem dealing with mileage and gasoline consumption: in the first table of the column are the kilometres and in the second one the gasoline litres used. Below is part of the transcript (translated from the original Spanish) from the initial interview:

Interviewer: O.K. in question number 2 you had […]. How did you do it?

Child: With this (pointing to the first column), I thought that if 12 is half of 24, it had to be half the gasoline, and since 48 is the double of 24, it had to be the same below, the double of gasoline.

Interviewer: Why did you put in the relationship between the kilometres and the litres that it had to be “the 6th part”?

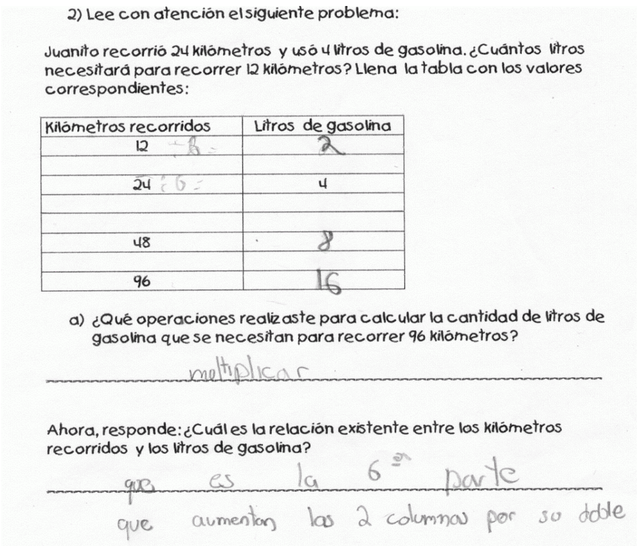
Child: Yes. Because I said: if 12 is divided by 2, its 6, then the 6th gives 2. If 24 is divided 6, it gives 4, and if 48 is divided by 6, it gives 8.

Interviewer: Why did you answer that it is the 6th part?

Child: Oh yes! I made a mistake…

Interviewer: How do they increase?

Child: Until here [points to a number] they increase by the double.


2) Lee con atención el siguiente problema:

Juanito recorrió 24 kilómetros y usó 4 litros de gasolina. ¿Cuántos litros necesitará para recorrer 12 kilómetros? Llena la tabla con los valores correspondientes:

Kilómetros recorridos	Litros de gasolina
12 ÷ 6	2
24 ÷ 6	4
48	8
96	16

a) ¿Qué operaciones realizaste para calcular la cantidad de litros de gasolina que se necesitan para recorrer 96 kilómetros?

multiplicar

Ahora, responde: ¿Cuál es la relación existente entre los kilómetros recorridos y los litros de gasolina?

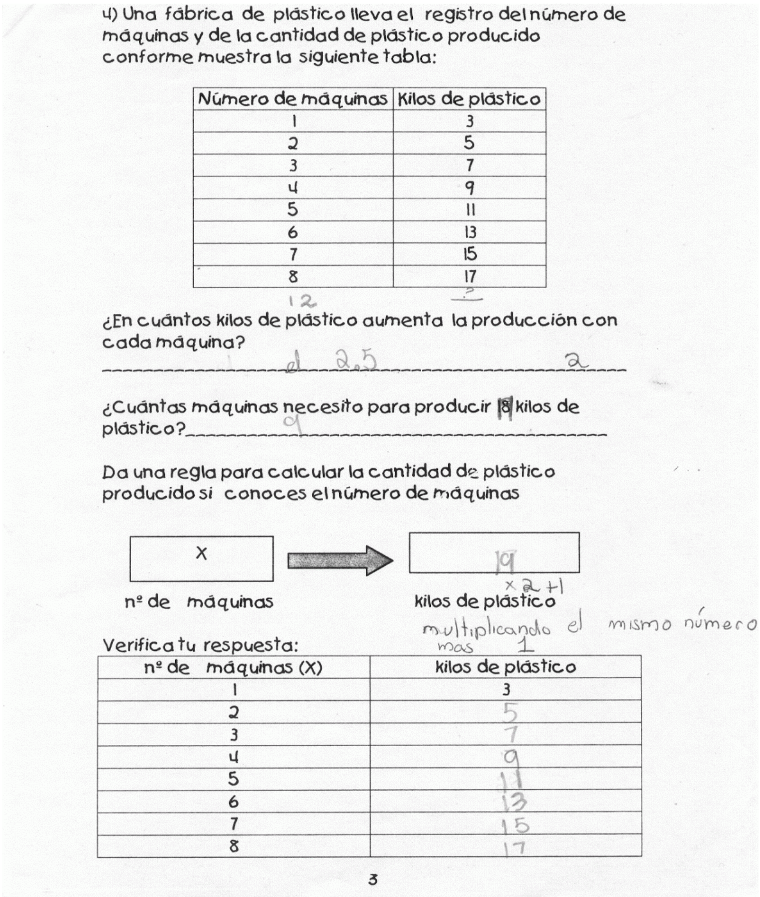
que es la 6ª parte

que aumentan las 2 columnas por su doble

Figure 2. A paper-and-pencil activity from the initial questionnaire: a mileage problem.

Questionnaire Problem 4

In the activity shown in Figure 3, children had to solve a word-problem dealing with a plastics factory that keeps records of machines and plastic quantities in kilograms (respectively, first and second columns of the tables). In this activity the linear relationship is explored. The children find the relationship among the number of kilograms of produced plastic and the number of machines involved, and they establish a pre-algebraic rule.



4) Una fábrica de plástico lleva el registro del número de máquinas y de la cantidad de plástico producido conforme muestra la siguiente tabla:

Número de máquinas	Kilos de plástico
1	3
2	5
3	7
4	9
5	11
6	13
7	15
8	17

12 2

¿En cuántos kilos de plástico aumenta la producción con cada máquina?

el 2,5 2

¿Cuántas máquinas necesito para producir 18 kilos de plástico? 9

Da una regla para calcular la cantidad de plástico producido si conoces el número de máquinas

x → 19

x 2 +1

nº de máquinas kilos de plástico

multiplicando el mismo número más 1

Verifica tu respuesta:

nº de máquinas (X)	kilos de plástico
1	3
2	5
3	7
4	9
5	11
6	13
7	15
8	17

3

Figure 3. Another paper-and-pencil activity from the initial questionnaire: the plastics factory.

Below is part of the translated transcript from the initial interview, after the interviewer reads out loud the problem and asks how it was solved:

Interviewer: [Reading the table] 2 machines, 5; 3 machines, 7; 4 machines, 9; 5 machines, 11; 6 machines 13; 7 machines, 15; 8 machines, 17.
By how many kilograms of plastic, does the production increases with each machine?

Child: 2 for each machine and then add 1.

Interviewer: How many machines do you need for producing 19 kilograms of plastic?

Child: 39.

Interviewer: If I want 19 kilograms of plastic, how many machines do I need?

Child: Te…, nine!

Interviewer: Why did you put here 39?

Child: I made a mistake. Nine!

Interviewer: What did you understand before?

Child: that it was the double and then you added 1 kilogram of plastic.

Interviewer: How did you find the rule?

Child: Because each machine produced 2 kilograms of plastic, plus 1, it's 3; that is, each machine added 1. That's why for 1 it's 3; for 2 it's 5; for 3 it's 7; for 4 it's 9, for 5 it's 12; for 6 it's 13; for 7 it's 15; for 8 it's 17.

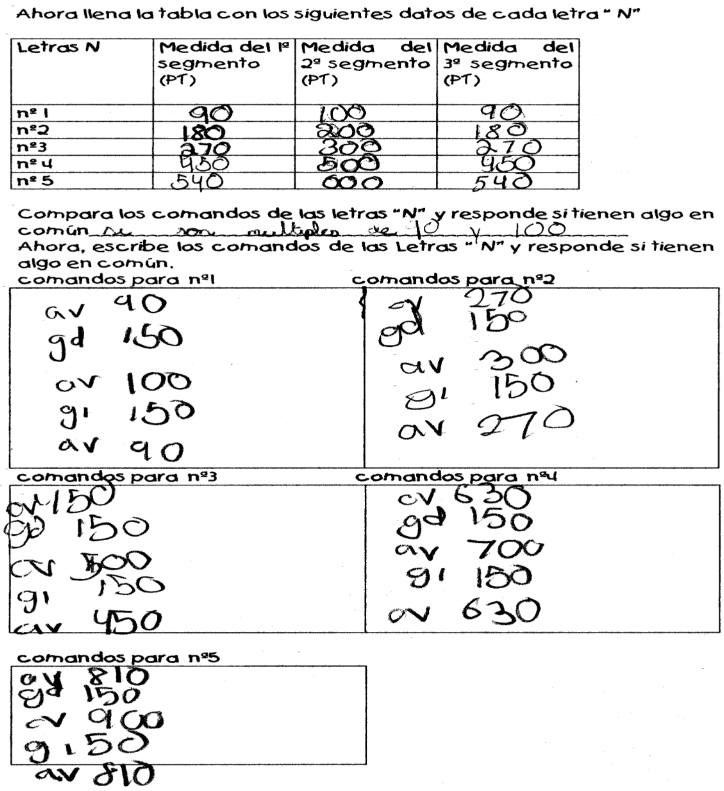
Interviewer: Tell me, what does the rule say?

Child: For each machine multiply by 2 and add 1.

Sample data from the Logo activities

In the following figures, some sample work is shown from the Logo activities carried out with the students during the preliminary study. This work was done using WinLogo.

Figure 4a shows a Logo activity worksheet for drawing a letter 'N' (shown in Figure 4b) in different sizes. In filling out the table and comparing the commands[1] for drawing each letter 'N' of different size, the students need to observe if there is a something in common. The fact that students themselves construct each letter 'N' is very important in helping them see the relationships.



Ahora llena la tabla con los siguientes datos de cada letra " N"

Letras N	Medida del 1º segmento (PT)	Medida del 2º segmento (PT)	Medida del 3º segmento (PT)
nº 1	90	100	90
nº2	180	200	180
nº3	270	300	270
nº 4	450	500	450
nº 5	540	600	540

Compara los comandos de las letras "N" y responde si tienen algo en común se son multiplos de 10 y 100

Ahora, escribe los comandos de las Letras "N" y responde si tienen algo en común.

comandos para nº1
av 90
gd 150
av 100
gi 150
av 90

comandos para nº2
av 270
gd 150
av 300
gi 150
av 270

comandos para nº3
av 150
gd 150
av 500
150
gi
av 450

comandos para nº4
av 630
gd 150
av 700
gi 150
av 630

comandos para nº5
av 810
gd 150
av 900
gi 150
av 810

Figure 4a. Logo activity for drawing a letter 'N'

```
para n :lado
av :lado
gd 150
av :lado
gi 150
av :lado
fin
```

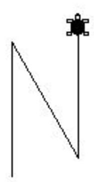


Figure 4b. Logo procedure for the letter 'N' with its result

[1] The Logo primitives are given in Spanish: AV is FD; GD is RT; GI is LT.

Another interesting activity is the one shown in Figure 5. In this activity generalisation processes are explored. The child is asked to observe a sequence of figures and then complete the figures for the 4th and 5th elements of the sequence. It is observed that the child completes the figures by counting the number of squares on the horizontal and vertical directions. Afterwards, the child fills a table of values and discovers a general rule for the figures. In the case of the child that filled the worksheet shown in Figure 5, he got the general rule for the case 11+10-1 and wrote how the number of H (for horizontal) and V (for vertical) squares are obtained from the figure number: H equals the 'figure number' plus 2; while V equals the 'figure number' plus 1.

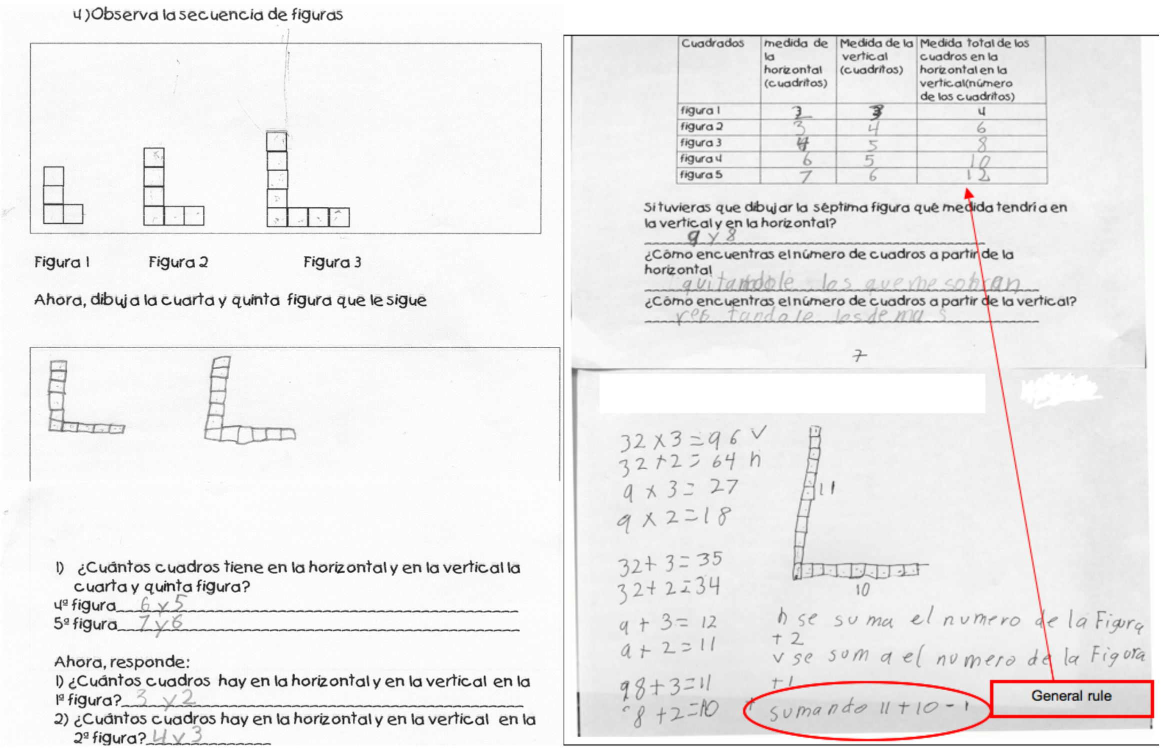


Figure 5. Pre-Logo activity in which a sequence of figures is observed and students have to reflect on what the general rule is that they would need to draw them in Logo

General results highlighting the changes between the beginning and end of the first phase

The results from the pre-questionnaire show that the participants were pre-algebraic students, which means that they didn't find difficult to understand some previous ideas, in spite of the difficulties that they found in the initial questionnaire. After the individual interview, it was confirmed that they reorganised their answers and were capable of recognizing their own mistakes as well as reorganise their thinking. In the pre-questionnaire, in problems that explore the idea of variable in a functional relationship, the students perceived the existent relationship among the problem quantities. They also perceived how the values of one of the variables increased and decreased, but they were not able to express this fact. They could express relationships between quantities in a table, but were unable to express them through a general rule. Instead they had to do this with a step-by-step description that did not allow them to generalise the relationship.

After the working sessions of the first phase, the children moved forward to more elaborate strategies in the resolution of the problems, showing conceptions of variable as a functional relationship and as general number. In the interview that was carried out after the working sessions, most of children changed the answers that they had given in the initial questionnaire. When the interviewer asked them to justify their answers, the children showed that they had advanced conceptually. For example, they were able to perceive a multiplicative relationship in problems involving a geometric sequence; in fact, two students perceived the functional relationship in data and were able to express a general rule. For the problems that approach the variable idea as general number, they understood, for example, how the quantities were varying, and seemed to attain an understanding of the relationships involved, even though in the paper-and-pencil environment some of them could not determine a general method and thus had problems in generating a general rule. However, most of the students were capable of verifying if a rule could function in all cases.

Final remarks

Introduction to early algebraic thinking along two routes of access (proportional reasoning and generalisation processes) – by means of a didactical sequence that takes into account the mathematical and cognitive background of the individuals – seems to be a viable approach and has correspondence with historical and curriculum perspectives. The results in the first stage of the study reveal some of the abilities and difficulties that are typical of the age-group we worked with.

With respect to the zone of actual development (ZDA), we verified that the interaction with the interviewer played a relevant role, because by this means, the students were able to make explicit the way they solved the problem as well as reconceptualise their knowledge. In some cases, help of the interviewer permitted that children could restructure their thinking by a simple solicitude of justification. In other cases, several levels of help were needed, depending on the real or actual evolutive level the children had. This real evolutive level (ZDA) could certainly be potentiated within appropriate technological environments for algebraic thinking, but also with a well structured design of activities from the didactical and psychological point of view.

References

Blanton, M., & Kaput, J. (2002). Design principals for tasks that support algebraic reasoning in elementary classrooms. In A. Cockburn & E. Nardi (Eds.), *Proceedings of the 26th Int. Conf. for the Psychology of Mathematics Education* (Vol. 2, pp. 105-112), Norwich, England.

Butto, C. (2005). Introducción temprana al pensamiento algebraico: una experiencia en la escuela primaria. Ph.D. thesis. Cinvestav, Mexico.

Carraher, D., Schliemann, A., & Brizuela, B. (2001). Can Young Students Operate on Unknowns? *Proceedings of the 25th Conf. for the Psychology of Mathematics Education,* Utrecht, The Netherlands, Vol. 1, pp. 130-140.

Cobb, P. (1994). A summary of four case studies of mathematical learning and small-group interaction, *Proceedings of the 18th Conf. of the Int. Group for the Psychology of Mathematics Education*, Lisboa, Portugal, Vol 2 ,pp 201-208.

Da Rocha Falcão, J. T. (1993). Algebra como ferramenta de resolução de problemas. In A. Schliemann et al. (Eds.) *Estudos em Psicologia da Educação Matemática*. Editora Universitaria UFPE, Recife, Brazil.

Durán Ponce, R (1999) Reconocimiento de patrones en secuencias numéricas y de figuras, por alumnos de sexto grado de primaria. M.Sc. thesis. Cinvestav, Mexico.

Filloy, E. (1990). PME Algebra Research. A working perspective. In G. Booker et al. (Eds.) *Proceedings of the 14th Conf. for the Psychology of Mathematics Education*, Oaxtepec, Mexico, Vol. 1, pp. PII1-PII33.

Filloy, E.; Rojano T. and Puig, L. (2008). *Educational Algebra. A Theoretical and Empirical Approach.* Springer, Berlin/Heidelberg/New York.

Geraniou, E., Mavrikis, M., Hoyles, C. & Noss, R. (2009). Design Decisions: A Microworld for Mathematical Generalisation. In M. Joubert (ed.) *Proceedings of the British Society for Research into Learning Mathematics* 29(3). Retrieved on 28 feb. 2010 from http://www.bsrlm.org.uk/IPs/ip29-3/BSRLM-IP-29-3-18.pdf

Hoyles, C & Sutherland, R. (1987). Ways of learning in a computer-based environment: some findings of the LOGO Maths Project. *Journal of Computer Assisted Learning* 3 (2), 67 – 80.

Hoyles, C & Sutherland, R. (1989). *Logo mathematics in the classroom*. Routledge.

Kaput, J, Blanton, M & Moreno, L (2008). Algebra From a Symbolization Point of View. In J. J. Kaput, D. W. Carraher, & M. L. Blanton (Eds.) *Algebra in the Early Grades,* pp.19-55. Lawrence Erlbaum Associates, Mahwah, NJ.

Mason, J., Pimm, D., Graham, A. and Gower, N. (1985). *Routes to roots of Algebra.* The Open University, Milton Keynes, UK.

Noss, R. (1986). Constructing a Conceptual Framework for Elementary Algebra through Logo Programming. *Educational Studies in Mathematics*, 17, 4, 335-357.

Radford, L. (1996). The roles of Geometry and Arithmetic in the development of Elementary Algebra: Historical Remarks from a Didactic perspective. In N. Bednarz, C. Kieran and L. Lee (Eds.) *Approaches to Algebra: perspectives for research and teaching*, pp. 39-53. Kluwer, Dordrecht /Boston/ London.

Slavit, D. (1999). The role of operation sense in transitions from arithmetic to algebraic thinking. Educational Studies in mathematics 37, 251-274.

Tabach, M., Arcavi, A. and Hershkowitz, R. (2008). Transitions among different symbolic generalizations by algebra beginners in a computer intensive environment. *Educational Studies in Mathematics* 69, 53-71.

Tudge, J. (1992) Vygotsky, the Zone of Proximal Development, and Peer Collaboration: Implications for Classroom Practice. In Moll, L.C. (Ed.) *Vygotsky and Education: Instructional Implications and Applications of Sociohistorical Psychology*. Cambridge University Press, Cambridge.

Ursini, S. (1993). Pupils Approaches to Different Characterizations of Variable in Logo. PhD thesis. University of London, Institute of Education, UK.

Vygotsky, L. S. (1978). *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press.

Robotic Performing Arts™ Project

Dave Catlin ***dave@valiant-technology.com***
Valiant Technology Ltd, Grange Mills, Weir Road, Balham, London SW12 0NE, England

An approach to STEM through cooperation not competition

The popularity of YouTube and the low cost of video equipment make it a practical possibility for students to create movie-shorts. In the Robotic Performing Arts™ Project students make a video of something similar to a puppet performance, replacing the puppets with Roamer® robots.

This paper discusses the potential of developing the project as an alternative to robotic competitions. While robot competitions are very popular, this project offers an alternative approach to STEM[1] education where the main focus is on student collaboration and the connection of STEM and other subjects through the cultural heritage of the student community. The paper presents an initial raison d'être as a project starting point.

Pilot projects are planned with DATA[2] in the UK and MESA[3] in the Washington State. These aim to develop and evaluate these basic ideas and create a practical framework to support this approach. The results of these projects will be presented in future papers.

Keywords (style: Keywords)

STEM, Educational Robotics, Roamer®, Performing Arts.

[1] STEM: Science, Technology, Engineering and Mathematics.
[2] DATA: The Design and Technology Association.
[3] MESA: Math, Engineering, Science Achievement

Introduction

This paper outlines the rationale behind a STEM based educational program entitled Robotic Performing Arts™ (RPA). As part of RPA, students use the Roamer® robot system to create a movie. I will show that this involves a production process rich with creative opportunities to explore **S**cience, **T**echnology **E**ngineering and **M**ath (STEM) as well as other areas of the school syllabus. I will review the program from four different perspectives:

1. STEM Opportunity
2. Educational Issues
3. Robot Projects and RPA
4. RPA Program Outline

Valiant Technology is developing the RPA Program with various partners and schools in the DATA[4] in the UK and MESA[5] in Washington State, USA. As the project progresses, I expect to bolster this raison d'être through increased practical experience and research data.

Early Work

Southmead Primary School, Wimbledon, South London produced "Coodies Circus" in 1989 (Valiant, 1989). The students wrote the script and built the robots using Classic Roamer®. They created the scenery, programmed the robots and performed the voice-overs. In those days, making a video was literally a major production effort. The filming and film editing was done by a professional film crew. Now most schools own video cameras and editing software, allowing students to take over the video production tasks.

The use of stories with robots predates the Southmead work. Valiant Technology used contextual to support the use of their Turtle robot (Ginn 1985). Since then students and teachers have informally written and used stories as part of their regular use of Roamer®. New research advancing our storytelling and robotics methodology forms part of a parallel R&D Project (Catlin and Royce 2010).

STEM Opportunity

Performing Arts and STEM might seem an oil and water combination. This is far from the truth. Producing movies, stage plays, and even musical recitals and dance can require the solving of highly technical problems. The use of robots as the actors creates an environment rich in STEM opportunities.

One obvious STEM strand is the design and construction of robot characters (Jones 1991,

Figure 1 Roamer® provides a sculptural plaform for character creation.

[4] DATA – the Design and Technology Association is the recognised professional organisation representing all those involved in D&T education in the UK.

[5] MESA is a programs in several US states aimed at supporting disadvantaged and underrepresented students to achieve academically in math, science and engineering.

Valiant 1992, Catlin 2007). With Roamer®, this can involve anything from simple art and craft approaches, to the use of kits like Lego, Fischertechnik, or K'nex. In schools or colleges with well-resourced shop facilities students can design and make components to create unique robots. This adaptability makes Roamer® suitable for K-12 schools with wildly different technological capabilities.

In building a robot students manipulate the parts and so engage in exploring the principles of mechanics and structures, and their underpinning mathematical and scientific principles. They do this in a tactile, primordially practical and physical level. I think this is so important. Through the process of building a robot, you understand forces by seeing and experiencing their effects. You gradually develop an intuitive understanding of the mechanical sciences.

Once students have created the characters, they program them to perform their role within the performance. This involves synchronizing movement and action, which in turn involves a wide range of simple and complex mathematical problems. With Roamer®, students can program the robot using one of the Standard Keypads Modules (KPM) or the technically more complex PIC Programming methods[6].

The creation of a storyline for the RPA production has great potential to connect students to almost any topic and subject. For example, these Hollywood productions can engage students in scientific topics:

1. Star Wars — Space and space travel
2. The Impossible Journey — Human physiology and biology
3. Jurassic Park — Palaeontology

These fictional, even fantastical, movie plots contain a vestige of real science. How much science and how real is controversial. Some commentators hate the scientific inaccuracies (Rogers, 2007). I agree with the more sanguine, pragmatic stance taken by Professor Sidney Perkowitz (Perkowitz 2007) who discusses the trade offs between dramatic license and scientific principles, but points out that even though the real science may end up on the "cutting-room-floor", movies tend to appoint science advisors whose job is to get the best representation from the plot line. This creative, but realistic interpretation of science becomes the task of the students.

However, using artistic media to inspire students does not need real science as its basis. The Jules Vern story "Journey to the Centre of the Earth" may be impossible for humans, but could a robot be able to do better? Just answering the question engages students with real science. Leroy Dubeck at Temple College has pioneered some efforts in this area (Dubeck, et.al. 1993; Dubeck and Tatlow 1998).

I anticipate being able to adapt some of the approaches developed by these pioneers to make them applicable to the broad aims of RPA. But in general, when students become scriptwriters they control the realism of the science presentation.

Over the last few years, the deep connections between language and mathematics have become apparent (Lakoff and Nunez 2001: Paulos 1998). I claim that narrative is equally potent in all STEM disciplines. Through the careful selection of assignments, teachers prompt students into thinking about topics in and "out of the box". That is instead of learning to regurgitate

[6] Roamer® is a modular system. The KPM is one module, which can be configured to provide different programming environments (behaviors). The four Standard KPMs provide a scaffolded Logo-like environment. Through PIC programming the students create unique autonomous Roamer® behaviours. See www.valiant-technology.com

normative views, creating stories encourages students to make the ideas personally meaningful. And for the record, embedding information into stories is recognized memorization technique (Byron 2006).

In some preliminary work in this area, students wrote stories that embrace keywords from forthcoming study topics – for example speed, time and acceleration (Catlin 2010). The students' stories showed their naïve, intuitive understanding of these concepts and made clear what they misunderstood. This gave the teacher a grasp of the student's prior knowledge – a critical principle in the science of learning (Bransford et.al 2000). More interestingly, when the students tried programming Roamer® to animate their stories, they found their perceptions challenged or affirmed.

Educational Issues

Here, I want to talk about a few educational issues that RPA will address:

1. Interdisciplinary Teaching
2. Student Engagement
3. Equity

Interdisciplinary Teaching

For practical reasons the way we teach artificially splits knowledge into different subjects. Today many teachers recognize the importance of reconnecting knowledge into a holistic experience (Wood, 1997: Bolak, et. al. 2005). This has led to a rise in interdisciplinary projects. Roamer® is inherently a cross curricular device. The power of Robotic Arts is its ability to interconnect Roamer® to a wide range of subjects, particularly bridging the chasm between Science and the Arts.

Student Engagement

Engagement is one of the ten ERA Principles (Catlin and Blamires 2010). Engagement is the capture of the students' attention, transforming them from bystanders to an active participant. It entails ideas like motivation and recognizes that learning itself is generally an enjoyable experience that does not need "jazzing up" with gimmicks.

In 1997, students from the Fleet Primary School in Lincolnshire decided to create a performing arts project about a circus. They designed and made various automata for each act and performed a Christmas Circus presentation for their parents (Valiant 1997). Head teacher Trevor Thomson was instrumental in the whole process. This example displays some manifestations of engagement:

1. Students immerse themselves in learning
2. Topics of study follow student interests and enthusiasms
3. Teaching to test is abandoned
4. Learning becomes authentic
5. Students give up their time to the project: in school breaks and at home students took every chance to work on the project
6. Knowledge ceases to be isolated: it becomes interconnected to other topics in the same subject and across subject boundaries
7. Knowledge acquired through engagement becomes memorable
8. Knowledge is not simply a remembrance of facts; it also contains emotional content

These qualities apply to the engagement principle, whether Roamer® or the RPA project is involved or not. You cannot guarantee what will engage students. My claim is that Roamer® has a propensity to support engagement and its nature is to sustain and encourage the positive learning aspects that derive from this approach. I predict we can say the same about RPA program. Together Roamer® and RPA offer a strong possibility of engaging students in positive learning experiences.

I contend that most students who enter robotic competitions are already interested in STEM subjects. Since many of these projects form part of after school programs, they tend not to attract students with little interest in these topics. I expect the focus on performing arts will attract students not normally interested in STEM topics. This is not an attempt to "fool students into a math class". Instead, it is about providing the student with a new perspective on the subject. Ralph Llewellyn, University of Central Florida created a physics course based on movies (Llewellyn 2002). His course invitation captures the zeitgeist of RPA:

> *Physical Science is a course for liberal arts students who are seeking to understand the world they live in… …to revitalize the course and, hopefully, ignite in students the flame of passion for science through the study of films.*

Equity

Research and classroom practice show that minority pupils perform better when teaching is filtered through their own cultural experiences and frames of reference (Gay 2000). Combining Roamer® with RPA offers a way of achieving this through several mechanisms:

1. Robot ancestry links them to almost every culture in the world
2. Application of dramatic traditions to educational robots
3. Robots act as transitional objects
4. Culture and RPA share the same narrative foundation

Robot is a Czech word for work that has found its way into all the major languages of the world. The word, which means hard labor and serfdom, came to prominence when the Czech writer Karel Capek used it in his play R.U.R[7] (Capek 2006). A robot is an artificial person: a cheap source of labour.

The idea of artificial life has many antecedents in many different cultures: for instance the creation myths where life springs from blood, mud, sticks, or stone. In other stories, statues or machines come to life. Chinese, European and Arabic cultures developed traditions of automata.

Examples of Artificial Life in Various Cultures			
Creation Myths		**Statues that come to life**	
Creature	**Culture**	**Creature**	**Culture**
Mbongwe	Bantu	Pygmalion	Greece
Obatala	Yoruba	Tilomatta	India
Tawa	Hopi	Gesar of Ling	Tibet
Tepeu	Mayan	Kwha-Shu	China
Glooscap	Mik Maq	Talus of Crete	Greece

Puppets show another form of the human anthropomorphic tendencies that have evolved in most cultures. We do not restrict this process to artificial humans. We endow animals, trees, rivers, machines – we give all of them the gift of human-like-life. I believe that Roamer® is an

[7] Rossum's Universal Robots

extension, indeed a fusion, of this inclination with robotics. As such, I think it offers a potential bridge between heritage and modern education.

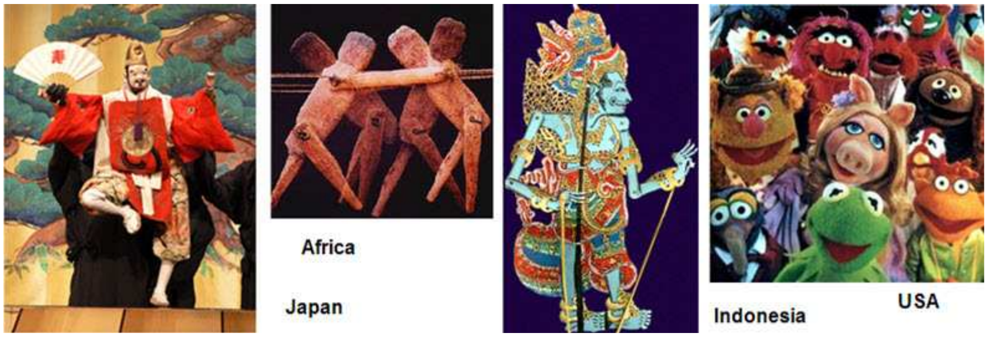


Figure 2 Puppets have a long tradition in most cultures.

Several lines of study offer themselves as ways we might use the insights of other disciplines to to develop and use robots within the RPA context. Western Dramatic theory starts with Aristotle Poetics (Aristotle 1997). He gets straight to the point:

> *Imitation is natural to men from childhood onward, one of the advantages of men over the other animals consisting precisely in this, that men are the most imitative of things and learn by imitation.*

Aristotle is talking about mimesis, the art of imitation. Despite its Eurocentric academic tradition, mimesis is the common denominator that runs through the cultural output of all societies. We add to this the related idea of verisimilitude, the techniques an artist uses to persuade the audience to suspend disbelief and embrace the story. Each culture has traditional methods for achieving verisimilitude (Bell, 2001) and it is a task of this project to study, explore and experiment with culturally responsive methods of doing this with educational robots.

While the foregoing remarks relate to traditional cultures, the analysis applies equally as well to contemporary cultures – for example Hip Hop. It applies to highly localized, dynamic cultural variations.

Studies of young children in Western cultures show the importance of transitional objects (Winnicott 1971) in the development of students' abilities to think about their world (Piaget 1962). These psychological mechanisms are not restricted to occidentals. Research in Papua New Guinea reveals the same imaginative and richly poetic process amongst Huli children (Goldman 1998). Papert referred to the original Turtle robot as a transitional object – an object to think with (Papert 1980). We normally associate transitional objects with small children. Originally, Papert thought of the Turtle in the same way. Later, he came to realize "what was good for thinking was good for thinking", no matter your age and no matter your level of sophistication (Papert 1983). We need to add substance to Papert's bold statement. We need to understand about transitional objects with older children and adults. This entire area needs more work, particularly regarding the practical exploration and integration of Activity Theory and Object Relationship Theory (Leiman 1999).

Robots created as neutral technological devices support the ERA Principle of Equity (Catlin and Blamires 2010). Roamer® is a tool and like a pencil, for example, it does not belong to a specific culture. It is a tool used by an artist in the production or consumption of cultural meaning (Hall, 1997).

Maori students used Classic Roamer as part of a RPA Project. They created Maori robots programmed to perform their traditional dances.

An American square dance version of the Maori project.

Figure 3: The design of Roamer robots can reflect the students cultural interests.

ERA compliant robots, like Roamer®, support these ideas. Roamer®'s amorphous shape provides a backdrop onto which students can sculpture a design that is an expression of themselves and their culture. It does not need to be high-tech or low-tech.

MIT's Edith Ackermann acknowledges the role of design as a means bringing the imagination into existence (Ackermann 2007). She makes the point made earlier by Schön.

> *…learning is designing, and designing is a conversation with – and through – artifacts. (Schön, 1983)*

While Roamer® provides an artifact with latent design capability; the RPA element brings an endless source of relevant problem solving scenarios. Together I believe they offer an environment rich in creative learning possibilities.

Roamer's link to ancestral origins provides one RPA cultural connection. Narrative is another. It is a potent factor in human life and child development and has particular relevance to RPA. Narrative is at the heart of a child's play. It is central to indigenous teaching practice (Peat 1994) and it provides a bridge that can link a student's cultural experience with the knowledge acquisition goals of the curriculum. In its broadest sense, narrative includes art, dance, movies, stories, plays, etc. Non-verbal forms of mimesis narrative precede language (Donald 1991). Yet narrative is the essential outcome of language. Narrative is the primary way we give meaning to our experiences (Polkinghorne 1988, Bruner 1987 and 1991).

Stories lie deep in the heart of every culture. The great Hindu epic the Mahabharata poetically tells the reader what to expect as the narrator summarizes the potency of the story:

> *Whatever is here is found elsewhere. But whatever is not here is nowhere else…*
> *…It's about you… If you listen carefully, at the end you'll be someone else.*

(Translation van Buitenen 1973)

Storytelling has an established role in modern education (Salans 2004; Gersie and King 1990). More recently, some researchers have experimented with storytelling and robotics (Druin and Hendler 2000, Stanton et al 2004). Others are currently engaged in combining all these threads into a new genre of educational robotics activities (Catlin and Royce 2010). RPA will draw on much of this work.

RPA enables students to play with STEM ideas, using robotics as an expressive medium, but from within the student's cultural and social milieu. If offers them the opportunity to understand concepts on their terms.

Robot Projects

Robot projects have become relatively common. For example: First Robotics - Lego League, BotBall, ROBOlympics, etc. Elements of these are:

1. Overtly STEM orientation (which does not appeal to everyone)
2. Focused on students building robots
3. Competitive in their nature

RPA focuses on cooperation and not competition. An RPA task list involves far more than a simple focus on STEM skills. It involves language arts, music, art, media studies, and arts and crafts. Nevertheless, embedded in these tasks is the potential for some *hard-core* STEM work.

- Creative Writing
- Storyboarding (Planning)
- Designing and making Robot Characters
- Set Design and Construction
- Creating Special Effects
- Create and Produce Musical Scores and Sound Effects
- Creating and Recording Roamer® Dialogs
- Filming
- Post Production

List of RPA Tasks

All the students involved in this project will engage directly or indirectly with STEM tasks. Hoever, even those who remain on the periphery of dealing with explicit technology will get the opportunity to see the value of STEM knowledge. This can be enough to reverse the normal *"Teach-Practice-Interest"* strategy. That is:

1. Teach methods
2. Let the students practice the taught methods
3. Try to persuade students what they've learnt is practical, useful and interesting

This is the wrong way round. It is far better to create the interest and engagement first.

RPA Project Outline

Schools are exceptionally conservative and very reluctant to move away from traditional approaches. There are many calls for innovative programs, but few educators are willing to embrace the systemic change needed to ensure significant improvement. I propose we investigate several ways of organising RPA. This gives schools the option of choosing an adaption that best suits their circumstances. The most obvious basic approaches are:

1.	Classroom Project	This is run as an ongoing project within the regular timetable. A theme is chosen from the curriculum. I would expect such projects to have their ambition curtailed by practical teaching issues. Nevertheless, they can still be valuable.
2.	After School Program	This provides an option for "going off curriculum", though it can be organized as a supplement to normal school work. Generally, we anticipate the scheme to be less formal.

For those willing to try major change I think the RPA offers a fascinating option: it can be run as an Inspiration Program at the start of the year. It should be a part of the regular timetable.

Sometime ago Valiant was working with a teacher who suggested some dates for a trial of project. The teacher commented, "… *these dates were good because the exams are over and* ***we're free to do the interesting stuff***!" We need to be doing the interesting stuff all year.

Brophy suggests that our first task should be to focus on motivating the students to learn (Brophy, 2004).

The RPA project has the potential to provide an inspirational environment. It does this by changing the *"Teach-Practice-Interest"* model and starts with need and motivation. Generally, within a project, students discover something they want to do but do not know how to. This authenticates the learning scenario. The need for knowledge comes first. The solution becomes "cool". Once learnt, students take ownership of the knowledge and actively seek new scenarios where they can use their newly found wisdom.

Supporting and Managing a Project

The RPA approach tries to use best teaching practice as outlined by the science of learning (Bransford et. al. 2000 and Sawyer 2006). Trying to classify it using such labels as constructivism, constructionism, guided participation, instructionalism, etc. is problematic. These terms seem to be embroiled in an academic "storm in a teacup". Mayer for example claims that:

> *…there is sufficient research evidence to make any reasonable person sceptical about the benefits of discovery learning —practiced under the guise of cognitive constructivism or social constructivism….*
>
> *(Mayer 2004)*

When viewed through the eyes of practitioners like Trevor Thomson, Mayer's arguments appear to be of the strawman variety. He discusses the notion of "pure discovery" learning – a thing never found in Trevor's classrooms even though he would claim to be an ardent supporter of discovery learning.

The RPA approach requires teachers to be at the "top of their game". One aim of the program is to serve communities normally classified as disadvantaged. A challenge arises here because the quality and qualifications of teachers in these communities is generally lower than average (Darling-Hammond, 2005). A standard instructional design strategy to overcome these issues is to provide teachers with scripted responses to anticipated situations. Dylan William criticized this approach (William 2007). He pointed out that even the best-regulated classroom is still a chaotic place, only capable of description by Chaos Theory. Effectively you cannot prejudge the complexity of learning situations. Imperceptible differences in scenarios can cause the same intervention to be a miserable damp squib in one case an instigator of multiple epiphanies in the other.

This complexity highlights the difficulty in trying to endorse one teaching strategy over another. The various theories serve a purpose in suggesting options, explaining processes as they unfold, but I think it is impossible to train and support teachers to run RPA projects based on theoretical analysis. What we really need is to "clone Trevor Thomson". We need to study the way our best teachers work and find ways to pass this expertise to less experienced colleagues. I believe we can do this through the creation of online communities. Although this aspect of RPA requires a major study, at this stage we can highlight a number of criteria the system will need to address:

1. Project themes — Project themes need careful selection to ensure they have the latent potential to meet study objectives. Anti-constructionists perpetuate the myth that "students" choose to study what they are interested in. They do, ***but***, in practice the teacher "rigs the deck". The students at Fleet School chose to do a project on the circus. Trevor had planned that they would make that choice. The subterfuge makes a huge difference to the student approach to the project. Another way of looking at this is that the teacher's role is to motivate the students' choices.

2. Guide — Students will get ideas and enthusiasms. Again the teacher's role is to "manage" the process. In the circus project the Fleet students wanted to visit a circus[8]. Trevor took the opportunity to turn this into a writing assignment. Instead of visiting an evening performance, he suggested the students write to the circus, explain their project and ask if they could visit them during the day. A teacher, who had run away to the circus and become a clown, acted as their tour guide. Trevor set the next writing assignment as an essay: A day in the life of a circus performer. You cannot plan or script this kind opportunity, but you can learn to leverage it.

3. Support — Trevor Thomson is a confident and brave teacher. His students' imagination often dragged him out of his depth. They would ask questions that he could not answer, or they would want to do things he thought impossible. His solution to this was to enlist the help of his local community: if he did not know how to do something, he would find someone who did know. With our ability to build online communities through systems like My eCoach[9], this becomes a practical possibility.

Conclusion

This document presents the case for approaching STEM by utilising the ethnic traditions of communities. It has also lays out the work necessary to create such a programme. The challenge is to follow this path, not in a dogmatic fashion, but with openness and a clear focus on the needs of the students and teachers.

References

Ackermann, E.K. (2007) *Experiences of Artifacts: People's Appropriations / Objects' 'Affordances'.* Available at http://web.media.mit.edu/~edith/publications/2007.explo.ofarti.pdf

Aristotle (1997) Poetics. Translated by S. H. Butcher. Dover Press.

Bell, J. Editor (2001) Puppets, Masks and Performing Objects. New York University and MIT Press

Bolak, K., Bialach, D., & Dunphy, M. (May 2005). *Standards-based, thematic units integrate the arts and energize students and teachers.* Middle School Journal, 31(2), 57 - 60.

Bransford, J. D. Brown, A. L. and Cocking, R. R. editors (2000) How People Learn: Brain, Mind, Experience, and School. Expanded Edition, Committee on Developments in the Science of Learning and Commission on Behavioral and Social Sciences and Education National Research Council National Academy Press

Brophy, J (2004) Motivating Students to Learn. Routledge.

Bruner, J.S (I987) *Life as Narrative.* Social Research Vol. 54, No. 1

Bruner, J.S (I991) *The Narrative Construction of Reality.* Critical Inquiry, 18:1, 1-21.

Byron, T. (2006). The Memory Experience. BBC Radio 4 first aired Wednesday 9th August 9:00 am. Available at: http://www.bbc.co.uk/radio4/memory/programmes/sharpen_memory3.shtml

[8] A circus was in the Fleet School area at the time – which was the hook that Trevor used to manage the students' choice of a circus as a theme.

[9] http://my-ecoach.com/

Capek, K. (2006) R.U.R. (Rossom's Universal Robots). A play translated by David Wyllie. eBooks@Adelaide http://ebooks.adelaide.edu.au/c/capek/karel/rur/

Catlin, D. (2007). *Robotics in Design Technology: Robotics is ideal for connecting design technology with other subjects.* Technology in Education Issue 154 Oct 2007. Accessed 10.06.10: http://www.valiant-technology.com/uk/docs/Design_Technology.pdf

Catlin, D. and Blamires, M. (2010). The Principles of Educational Robotic Applications (ERA): A framework for understanding and developing educational robots and their activities, In Press.

Catlin, D. (2010) Introducing Key Mathematical Concepts using Robots and Storytelling. Accessed 10.06.10. http://www.valiant-technology.com/uk/pages/r2_activity_library.php

Catlin, D. and Royce, A (2010) Roamer® and Storytelling: New Ways to Use Educational Robots. Accessed 10.06.10 http://www.valiant-technology.com/uk/pages/r2_activity_library.php

Darling-Hammond, L. (2005) New Standards and Old Inequalities: School Reform and the Education of African American Students, p204, in Black Education Editor King, J. E. Published for the American Educational Research Association by Lawrence Erlbaum Associates.

Donald, M. (1991) Origins of Modern Mind: Three stages in the evolution of culture and cognition. Harvard University Press.

Druin, A and Hendler, J (2000) PETS: Programmable Electronic Teller of Stories. In Robots for Kids (Eds Druin, A and Hendler, J.) Academic Press

Gay, G. (2000) Culturally Responsive Teaching: Theory, Research and Practice. Teachers' College Press.

Dubeck, L. W. Moshier, S. E. Bruce, M. H. and Boss J. E. (1993) *Finding the facts in science fiction films,* The Science Teacher 60, April 1993.

Dubeck L. W. and Tatlow, R. (1998) Using STAR TREK: The Next Generation television Episodes to Teach Science, Journal of College Science Teaching

Gersie, A. and King, N. (1990) Storymaking: in Education and Therapy. Jessica Kingsley Publishers

Ginn, A. (1985) The Adventures of Myrtle the Turtle. Valiant Technology Ltd.

Goldman, L.R. (1998) Child's Play: Myth, Mimesis and Make Believe. Berg.

Hall, S. Editor (1997) Representation: Cultural representations and signifying practices. Sage.

Jones, R. (1991) *Roamer® a Vehicle for Design Education in Schools.* GO Issue 6. Valiant Technology Ltd. Accessed 10.06.10: http://www.valiant-technology.com/uk/pages/go_archives.php

Lakoff, G. and Nunez, R. (2001) Where Mathematics Comes From: How The Embodied Mind Brings Mathematics Into Being. Basic Books.

Leiman, M. (1999) The Concept of sign in the work of Vygotsky, Winnicot and Bakhtin: Further integration of objects relationship theory and activity theory, In Lev Vygotsky critical assessments Volume 4 edited by Peter Lloyd and Charles Fernyhough. Routledge.

Llewellyn, R.A. (2002) psc1121f - Physical Science - Physics in Films – Superheros. University of Central Florida. http://reach.ucf.edu/~psc1121f/main.html

Mayer, R.E. (2004) *Should There Be a Three-Strikes Rule Against Pure Discovery Learning? The Case for Guided Method of Instruction.* American Psychologist, *59*(1), 14–19. Also available at: http://projects.ict.usc.edu/itw/vtt/MayerThreeStrikesAP04.pdf

Papert, S. (1980) Mindstorms: Children, Computers and Powerful Ideas, Basic Books.

Papert, S. (1983) Talking Turtle: BBC Horizon Programme introducing the core concepts of Logo and Turtle Available from http://www.valiant-technology.com/uk/pages/archives.php#video

Paulos, J.A. (1998) Once Upon a Number: The Hidden Mathematical Logic of Stories. Basic Books

Peat, D. F. (1994) Blackfoot Physics: A Journey into the Native American Universe. Fourth Estate.

Perkowitz, S. (2007) Hollywood Science: Movies, Science and the End of the World. Colombia University Press.

Piaget, J. (1962). Play, dreams and imitation in childhood. London. Routledge & Kegan Paul.

Polkinghorne, D. E. (1988) Narrative Knowing and the Human Sciences. State University of New York Press.

Rogers, T. Accessed 10.06.10 http://www.amazon.co.uk/Insultingly-Stupid-Movie-Physics-Rogers/dp/1402210337/ref=sr_1_1?ie=UTF8&s=books&qid=1262326878&sr=8-1(2007) Insultingly Stupid Movie Physics. Sourcebooks, Inc

Salans, M. (2004) Storytelling with Children in Crisis, Jessica Kingsley Publishers

Sawyer, R. K. Editor (2006) Cambridge Handbook of the Learning Sciences, Cambridge University Press

Schön, D. (1983). *The Reflective practitioner: How professionals think in action.* New York: Basic Books.

Stanton, D. O'Malley, C. Bayon, C. Joiurcade, J.P. Sundblad, Y. Fast, C. Cobb, S. Taxen, G. Benford, S. (2004) The Kid Story Project: Developing Collaborative Storytelling tools for children with children. In Developing New Technologies for Young Children Edited by John Siraj-Blatchford. Trentham Books.

Valiant (1989) Coodies Circus, written and performed by students of Southmead Primary School, Wimbledon, London.
Accessed 10.06.10 http://www.valiant-technology.com/uk/pages/r2_activity_library.php

Valiant (1992): Roamer® Design Competition Key Stage 1 GO Issue 14 and Roamer® Design Competition Key Stage 2 GO Issue 15. Published by Valiant Technology Ltd. Accessed 10.06.10: http://www.valiant-technology.com/uk/pages/go_archives.php

Valiant (1997) Circus Project by Students of Fleet School, Lincolnshire. Head Teacher Trevor Thomson. Accessed 10.06.10 http://www.valiant-technology.com/uk/pages/freestuff.php

Valiant (1997): Grannies Great Leap designed and made by students from Primary School in Lincolnshire, UK. Accessed 10.06.10: http://www.valiant-technology.com/uk/pages/freestuff.php

van Buitenen, J.A. translator (1973) The Mahabharata, Volume 1: Book 1: The Book of the Beginning (Bk. 1) (Vol 1), University of Chicago Press.

William, D. (2007) Assessment, learning and technology: prospects at the periphery of Control; Keynote speech at the Association for Learning Technology Conference in Nottingham, England

Transcript: http://www.alt.ac.uk/docs/altc2007_dylan_wiliam_keynote_transcript.pdf

MP3 recording http://www.alt.ac.uk/docs/altc2007_dylan_wiliam_keynote_audio.mp3

Slides and video http://www.alt.ac.uk/altc2007

Winnicott, D. (1971) Playing and Reality. Basic Books.

Wood, K. (1997). Interdisciplinary instruction: A practical guide for elementary and middle school teachers. Merrill.

Storytelling with Roamer®

Alison Royce, *alison.royce@btinternet.com*
Freelance storyteller.

Dave Catlin, *dave@valiant-technology.com*
Valiant Technology Ltd, Grange Mills, Weir Road, Balham, London SW12 0NE, England

Exploring Storytelling Methods Useful with Robots

This workshop presents our work on combining the power of storytelling with educational robotics. It will be led by Alison Royce - a drama teacher, actress and now a freelance storyteller. She is supported by robot designer Dave Catlin CEO of Valiant Technology. It will use Valiant's Roamer robot. It is suitable for those with a general interest in constructionism and educational robots. People are particularly invited who are interested in a use of robots as relational objects.

Classic Roamer has been in schools since 1989. It is a combination of Turtle and Big Trak. The latest version, Roamer-Too, is a talking robot which interacts with students via any semiotic system, whether traditional Logo or speech or the wave of a hand. Students can transform Roamer into different characters using engineering techniques or arts and crafts. However, Roamer is also ready for action "straight out of the box". Primarily the student's focus is not on building robots, but using them to engage difficult ideas.

Storytelling is one of the oldest educational tools used by every culture in the world. Embedded in stories is the wisdom of generations. When we engage stories we explore ancestral patterns of thinking and reconstruct their knowledge, taking from them meanings relevant to our time and space. A storyteller with a glove puppet can draw students out of themselves and put them into the story so they can explore the world from a new perspective. Storytelling props can be anything; a piece of wood or a tin of beans become things that prod our imagination. We construct stories from the "things to hand": Sherry Turkle calls these "relational objects". When the robot is a part of this bricolage there is a shift from cognitive to affective thinking. The opportunity exists for students to apply their innate cultural knowledge to problem solving.

Method

Our efforts to fuse together the various elements of traditional storytelling with Roamer technology is at an early stage. We wish to enrich our general understanding by engaging with the considerable experience available at the conference. The workshop will offer people the opportunity to engage with some of the techniques we have so far developed and use that experience to suggest ways of developing this whole approach.

Expected outcomes

We hope participants will leave with a general understanding of the approach and inspired to develop new to stories embracing different themes.

Keywords

Educational Robots, Roamer, STEM, Storytelling, Constructionism.

Stimulating different intelligences in a congruence context

Toni Chehlarova, tchehlarova@mail.bg
Institute of Mathematics and Informatics at the Bulgarian Academy of Sciences

Evgenia Sendova, jenny@math.bas.bg
Institute of Mathematics and Informatics at the Bulgarian Academy of Sciences

Abstract

The paper deals with didactical scenarios on the notion of *geometric congruence* based on previous successful experience with using a specific model T which reflects the relations among structures, activities and intelligences (in the sense of Howard Gardner). The underlying idea is that different representations of the same object of study (Figure 1) enable the learner to use his/her strong intelligences, and to enhance and develop the rest in the intelligence spectrum.

Figure 1. Various representations of geometric congruence

We present a series of problems and the response of students of different age. The learning process is in harmony with the fundamental ideas of the constructionism – it involves computer modeling of artefacts created in the best tradition of the Bulgarian folk art – wood-carved ceilings (Figure 2), painted ceramics, embroidery, crocheting, etc., to be presented and shared within a larger audience.

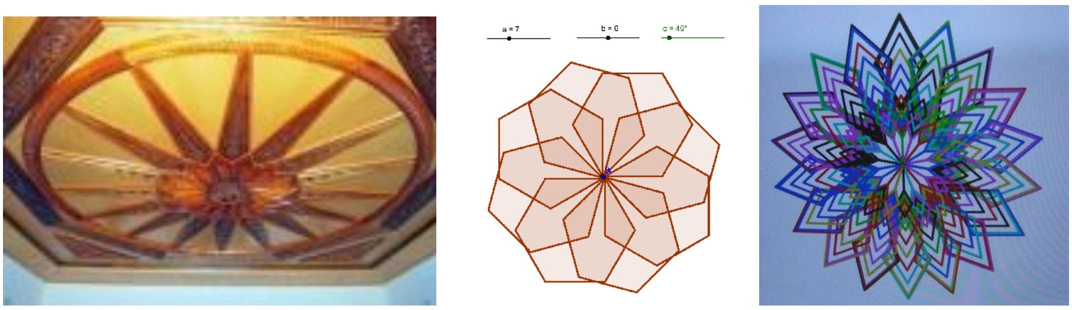


Figure 2. A wood-carved ceiling and models in different computer environments

Our experience with students and teachers alike suggests that harnessing the strong intelligences of the learners enhances the understanding and appreciation component of the learning process.

Keywords

Constructionism, geometric congruence, art, intelligence

Introduction

The symmetry and the asymmetry co-exist in unity both in nature and art. A creative process involves specific decisions about the symmetry-asymmetry ratio of the object being created, and this very ratio is a crucial aspect when evaluating its esthetics. The design and the creation of objects sharing the features of fantasy and nature is often based on geometric transformations such as *congruences* (translations, rotations, reflections and compositions of those). Thus it seems natural to encourage learning this geometry topic by integrating it with the design and reconstruction of artistic artefacts. With this idea in mind we have developed didactic scenarios on geometric congruences in the frames of the *InnoMathEd* European project [1]. The design and the implementation of these scenarios is just an element of a more ambitious goal - we expect our students to look for manifestations of geometric congruences, discover them and use them in various activities, and thus – to be able to find patterns and relationships deepening their knowledge and understanding of the surrounding world.

The congruence scenarios are based on previous successful experience with using a specific model T [2] which reflects the relations among structures, activities and intelligences (in the sense of Howard Gardner [3]) - Figure 3.

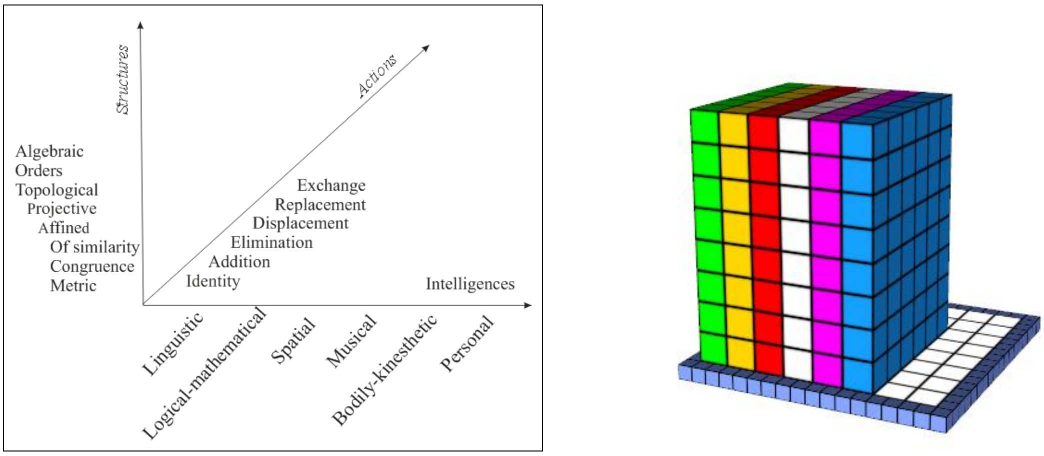


Figure 3. The structure–activity model T of the intelligences

The congruences

The notion of *congruence* can be introduced by various methods and activities involving different intelligences (a subset of the T model).

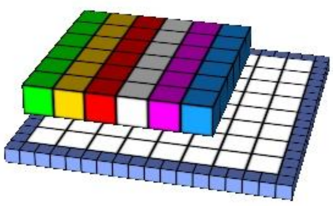


Figure 4. The congruence layer

The students:

- observe objects from nature, architecture, popular customs, science and discover (recognize) congruencies
- create congruent objects by means of various ideas and tools (e.g. by folding a sheet of paper, turning a slide or a piece of glass with a drawing on it, by means of an appropriate software, etc.)
- transform figures in order to get a specific one
- reach the level of free use of geometric transformation as a means for achieving the goal of a more complex project

The first tasks for the students are to discover and describe the harmony based on congruences in different sports and folk dances (Figure 5). Then they are expected to perform symmetrical movements with arms and legs. And finally – to present (just with their hands) geometric figures such as triangle, trapezoid, etc.

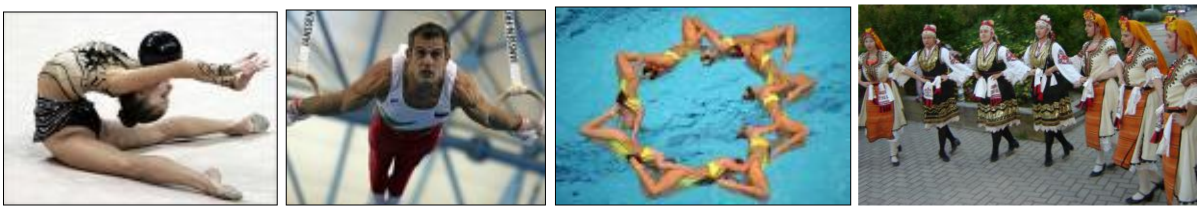


Figure 5. Recognizing congruences in sports and dances

Next the students can use various computer environments for exploring congruences [4-7]. They can draw by “free hand”, work with specific tools in graphic editors, grids in 2D and 3D, dynamic constructions, and of course - programming in Logo (Figure 6).

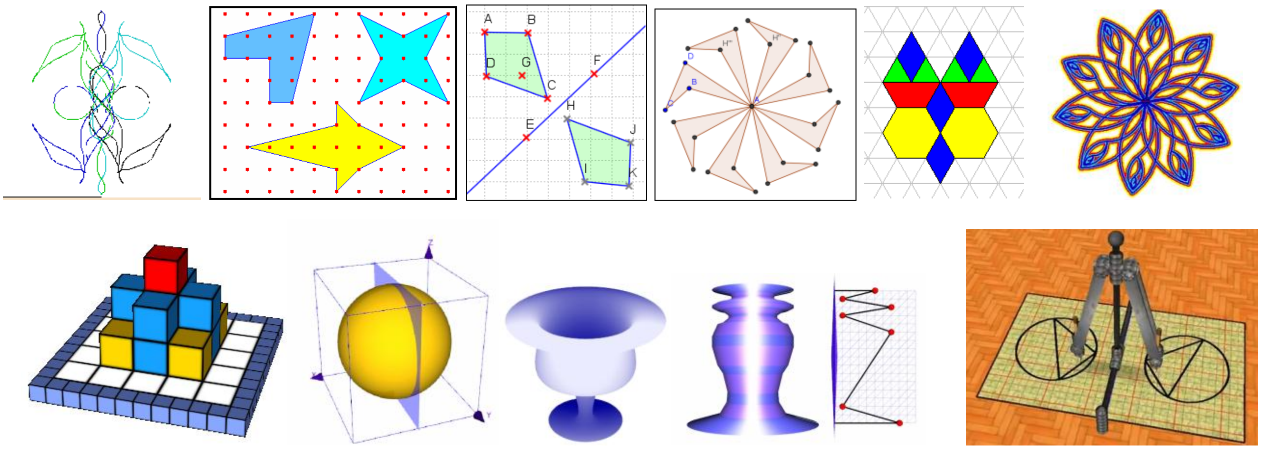


Figure 6. Different computer environments for exploring congruences

In order to make the right choice for a specific project it is important for the students to realize that each environment has its advantages and disadvantages.

For those who are fond of music, the symmetry could be demonstrated in the context of famous music fragments: (Figure 7). Here is how a 16-year old student with a high motivation for science reacted to this musical illustration of symmetry during a lecture on music and mathematics:

> When Prof. Noam Elkies played the traditional Paganini theme in reversed it turned into variation 18 of Rahmaninoff's Rapsody on the Theme of Paganini. For anyone with at least an ounce of knowledge in classical music, this had to have been an epiphany moment...

Thus the symmetry could not only be seen but could also *heard*.

Figure 7. Part of 24th capriccio theme by Paganini and the 18th variation theme by Rachmaninoff

A good project for the musically inclined is to find the musical interpretation of the geometric congruences [8].– Figure 8.

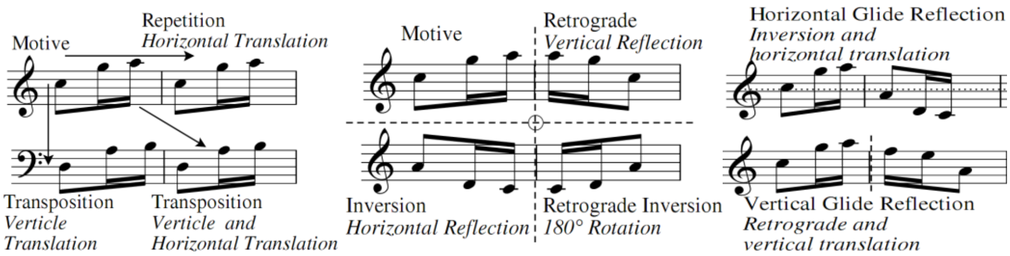


Figure 8: All congruence transformations in musical space

Congruences could be found in various aspects of a music piece when considered as a sequence of clear-cut units. The repetition (translation) and the symmetry in a melodic structure are easily recognized in its formal description (by letters or digits) (e. g. AABBCCDD, ABA, 11112233, 11122111) where the same letter (digit) would mean that the corresponding units have the same melodic structure possibly modified by transposition (keeping the intervals). The same idea is used for describing the harmonic structure, e.g. the string TTSDDSTT reflects the harmony of 8 bars being based on the tonic (T), subdominant (S) and dominant (D) chords.

Analyzing the structure of a set of music pieces in terms of congruences can be very helpful within a project for writing algorithmic music [9]. Similarly, it is a good challenge for the students to recognize congruences in the natural language and the literature works (e.g. to search and check for palindromes, to analyze the structure of a drama, of poetic forms such as sonnets, haiku, etc.). Then they could describe formally these structures, and finally – to get acquainted with the standard description, e.g. in the case of drama – with *Freytags pyramid* (Figure 9) [10].

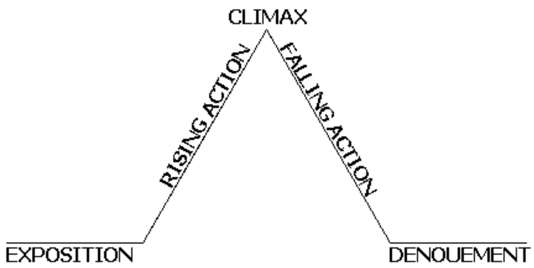


Figure 9. The structure of a drama as presented by Freytag [10]

Especially interesting in mathematics context are problems with the following operations: adding, removing, replacing, and exchanging [11 - 14].

What follows are fragments of our experience with design and reconstruction scenarios involving congruences. We present a series of problems and the response of learners of different age.

Congruences and design

Problem 1. Study the tessellation below and create your own tessellation design.

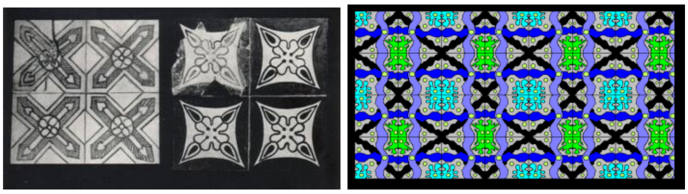


Figure 10. Painted ceramics from Veliki Preslav and a modern tessellation

After observing the above samples of tiles the students are expected to realize that since the tile has the form of a square and decoration with 4 axes of symmetry the options for different tessellations based on these tiles are very limited (if the tiles have to have common vertices).

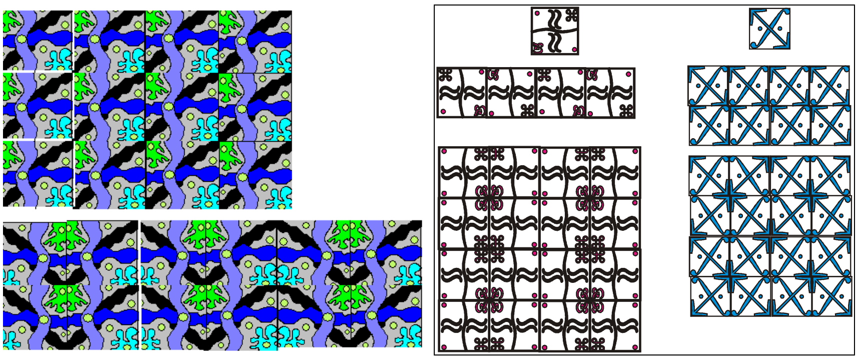


Figure 11. Tessellations with a ready-made and with learners' own tile-model

This project was used in educating pre-service teachers who were given the opportunity to use software of their choice. The computer models in Figure 10 and Figure 11 were made with Paint and CorelDraw. The strength of using a programming language (Logo in our case) was obvious – precision of the tiling, using procedures with parameters for the size of the tile, the number of the rows and columns and turning it in a dialog with the user made the programming efforts worthwhile.

Problem 2. Make a model of a ceramic plate in the style of "Troyan drop" [15]. (the first two in Figure 12).

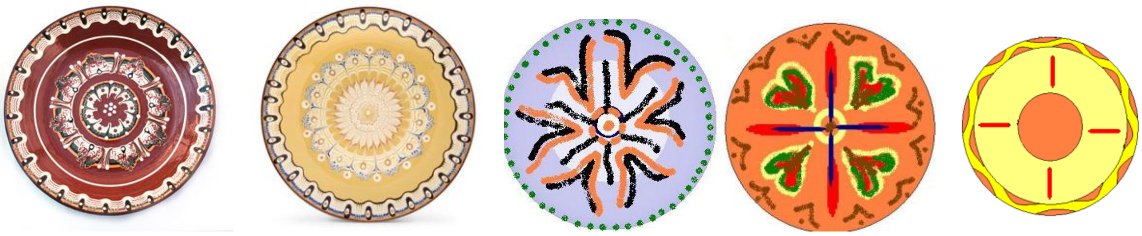


Figure 12. Samples of hand made ceramic plates and computer models in the same style

When painting in a *free hand* style the students would start with the axes. All of them used an even number of axes and then continued the painting by coupling the motives. The zoom mode was used to improve the precision of the work.

Problem 3. Help the archeologists to reconstruct the artefacts (Figure 13) by creating computer models.

Figure 13. Models of ancient artefacts to be reconstructed

One of the shortcomings of using Paint is that only the rotation at angle multiple of 90° is possible. If the students choose dynamic software, e.g. GeoGebra, they can insert twice the picture and then rotate the first copy around a center at an appropriate angle. The first figure is made transparent and then the two figures are rotated until they fit (Figure 14).

Figure 14. Reconstructions by means of Paint and GeoGebra

The idea could be extended in the case of rotational artefacts.

Problem 4. Create computer models of the artefatcs in Figure 15.

Figure 15. Modern and ancient artefacts based on rotation

Some of the students decided to use a pattern to be copied and translated appropriately (Figure 16). The difficulties when passing from 2D to 3D were discussed with the teacher.

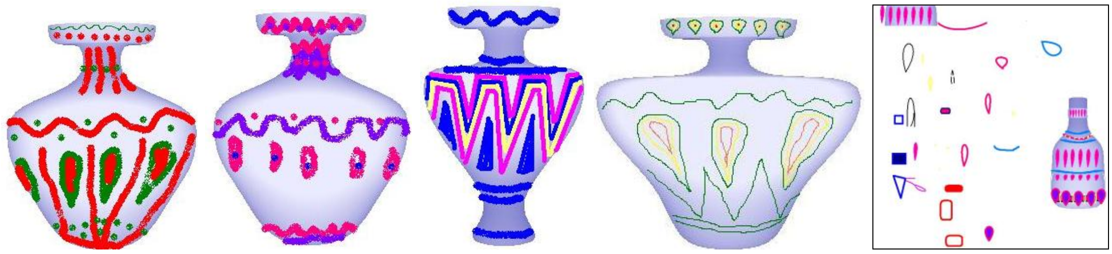


Figure 16. The students' design

Problem 5. Make a model of a wood-carved ceiling as the ones in Figure 17.

Figure 17. Woodcared seiling from Triavna and Plovdiv and some computer models

In the case of the last model in Figure 17 the students started with a procedure drawing a Y-shaped element, then added parameters for the size and the angle, made a branch with a varying number of Y-shaped elements and rotated it around the starting point giving rise to a whole class of ceiling models.

Another rich idea for design projects is the Bulgarian embroidery and crocheting.

Problem 6. Create a computer motive for Bulgarian embroidery (Figure 18-1, 2).

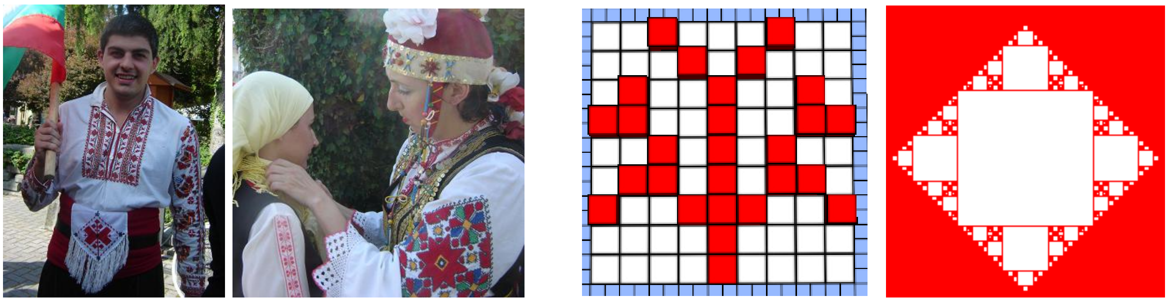


Figure 18. Bulgarian folk dancers and computer models inspired by the embroidery of the costumes

Here the students are expected to assess the specifics of the software and to choose the most convenient one for creating the basic motive and then to decide what geometric transformation to apply so as to get the desired composition.

The project for creating crochet models involves precision and option to play with various parameters, thus the choice of Logo is natural:

Problem 7. Create a design of a lace napkin for serving tea or coffee (Figure 19-1).

Figure 19. Masterpieces of older and younger Bulgarian women

It is interesting to note that this project is usually offered at the beginning of a Logo course for graduates in different fields who want to qualify for teachers in IT. As a rule they get excited of the “no threshold, no ceiling” motto of Logo and enjoy playing with parameters to produce various rosette models, and further – to create an interactive program. Of course, the ceiling could be only your own imagination. What if the crochet model contains a unique element which happens to be symmetric (Figure 20-1)…This gives rise to the next project.

Problem 8. Create a stylized model of a butterfly:

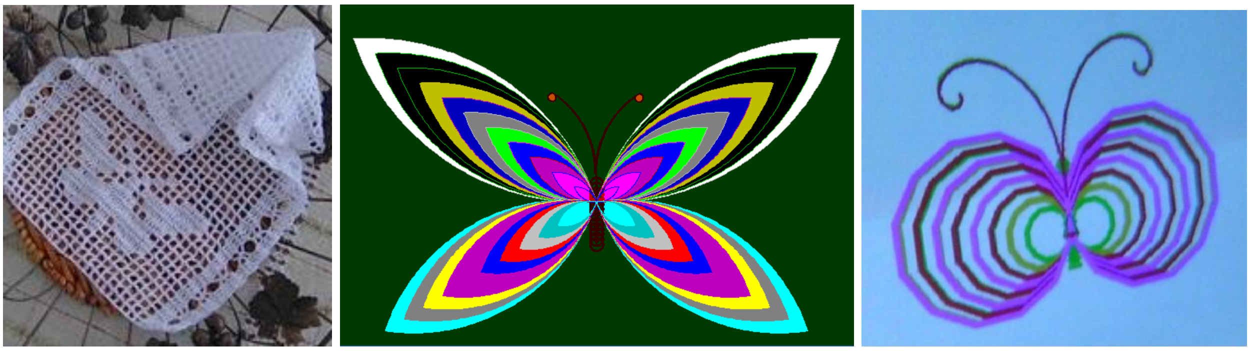


Figure 20. Crochet and Logo models of a butterfly

Coming back to our main object of study, the symmetry, let us recall what the great mathematician Hermann Weyl thinks of it: *Symmetry is one of the ideas by which man through the ages has tried to comprehend and create order, beauty, and perfection.*

Is there an innate link between symmetry and beauty? Could symmetrical composition be an intricate part of defining beauty? Plastic surgery advertisements often read: *The symmetry is beauty*. At the same time a recent study by Harrison [16] shows that although we commonly perceive faces as symmetric, in fact there are asymmetries in them.

Maybe this interplay between symmetry and asymmetry in search of beauty was what inspired an in-service teacher (working in a school for fashion design) to create a pattern which was originally symmetric and then became asymmetric after she added a decoration (Figure 21)**:**

Figure 21. Creating balance of symmetry and asymmetry

Discussion

Many authors have been looking for various means to support the understanding of knowledge, including the notion of *congruence*. The T model provides options for seeking completeness with respect to the intelligences and the actions used. There are examples in the literature for the use of linguistic, spatial, and musical intelligence but we haven't come across a goal-oriented reference to the bodily-kinesthetic intelligence in this aspect. Often the teachers create situations for recognizing and building structures involving congruences and addition. Such situations could be visualized by a subset of the T-model (Figure 22-1).

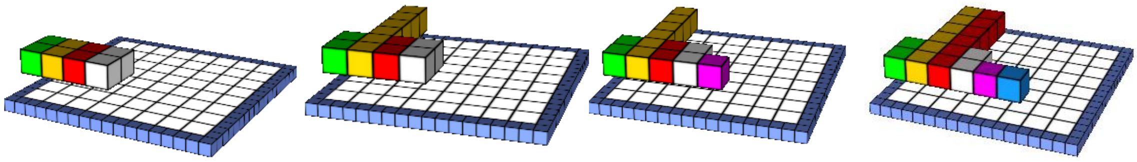


Figure 22. Visualisng systems of problems on congruence involving various intelligences and actions

Ideas for including the actions *adding, elimination, displacement, replacement* and *exchange* in systems of mathematical problems on congruences are presented in [11-14] and could be visualized as in Figure 22-2. If the rest of the itelligences are involved the completeness of the congruence layer (Figure 4) would be achieved.

The problems involving the bodily-kinesthetic intelligence could be presented as in Figure 22-3. When organizing the learning process in harmony with the constructionists ideas it is important to emphasize on the sharing. It is related to thinking on the thinking, to the estimating one's own actions, to understanding and estimating the actions of the other members of the team – all being components of the personal intelligence (Figure 22-4).

The T model could be used also at a problem level. When preparing a problem to be offered to the students the teacher has expectations about the possible solutions and mistakes the students would make. Based on the various possible solutions the teacher can fix the actions being used by his/her students. Thus some *gaps* will appear which could be filled by new problems.

Conclusions

Our experience with learners of different age (pupils, pre-service and in-service teachers alike) suggests that harnessing the strong intelligences of the learners enhances the understanding component of the learning process.

As emphasised by Clayson [17] the visual exploration with models helped the learners increase their visual vocabulary by forcing it to describe (sometimes) unexpected shapes.

Furthermore, assisting the learners in their choice of a computer environment appropriate for a specific project goal is a crucial factor for the students to experience satisfaction, joy, self-confidence and motivation for further activities, and even to reach thoughts impossible for them before that.

In conclusion, we felt very encouraged to hear what one of the teachers we have recently been working with wrote after a Logo course: *It is not possible to "feel" informatics, music, art, poetry, even dancing, not to forget the dreams, if they are in isolation.*

Acknowledgemnts

We wish to thank the European Union for funding the *InnoMathEd* project

References

1. Kenderov, P. *Innovations in mathematics education: the European projects InnoMathEd and Fibonacci,* in *Mathematics and Education in Mathematics* - Proceedings of the 39th Spring Conference of the Union of Bulgarian Mathematicians, Albena, April 6–10, 2010

2. Chehlarova, T. (2009*) Structure – activity model T of the intelligences*. Scientific papers, Plovdiv University "P. Hilendarski". vol 46, book. 2, Methods of education, pp.37-48

3. Gardner, H. (1993) *Multiple intelligences. The theory in practice*. Basic Books, New York, 304 p.
4. The National Library of Virtual Manipulatives, http://nlvm.usu.edu/
5. GEONExT, http://geonext.uni-bayreuth.de/
6. GeoGebra, http://www.geogebra.org/cms/
7. Elica, http://www.elica.net/site/index.html
8. Hart, V. (2009) *Symmetry and Transformations in the Musical Plane*, Proceedings of the 12th Annual BRIDGES Conference: Mathematics, Music, Art, Architecture, Culture, Banff, Canada (to appear)
9. Sendova, E. (2001) *Modelling Creative processes in Abstract Art and Music*, Eurologo 2001, Proceedings of the 8 th European Logo Conference 21-25 August, Linz , Austria
10. Freytag, G. (1863) *Die Technik des Dramas*, (in German).
11. Grozdev, S. and Chehlarova, T. (2007) *Cubes and constructions*. Association for the Development of Education, Sofia.
12. Grozdev, S. and Chehlarova, T. (2009) *Congruences in triangular nets*. 6th Mediterranean Conference on Mathematics Education 22 – 26 April 2009, Plovdiv, pp. 239–248.
13. Chehlarova, T. and Sendova, E. (2009) *Explorations with rotational solids*. 6th Mediterranean Conference on Mathematics Education 22 – 26 April 2009, Plovdiv, pp. 259–268.
14. Grozdev, S. and Chehlarova, T. (2008) *Formation of an idea of a symmetric solid*. VI congress of the mathematicians of republic of Macedonia, Struga, 19 – 22.10.2008.
15. http://troyamica.com/en/about_us.html
16. Harrison, D. http://www.upscale.utoronto.ca/GeneralInterest/Harrison/Parity/FaceStudy/FaceStudy.html
17. Clayson, J. (1988) *Visual Modeling with Logo*, The MIT Press, Cambridge, Massachusetts, London, England, pp. 115-116

Representation systems of 3D building blocks in Logo-based microworld

Han Hyuk Cho, *hancho@snu.ac.kr*
Dept of Mathematics Education, Seoul National University, Seoul 151-748, South Korea

Hwa Kyung Kim, *indices@smu.ac.kr*
Dept of Mathematics Education, Sangmyung University, Seoul 110-530, South Korea

Min Ho Song, *mino@snu.ac.kr*
Dept of Mathematics Education, Seoul National University, Seoul 151-748, South Korea

Ji Yoon Lee, *lily1982@snu.ac.kr*
Dept of Mathematics Education, Seoul National University, Seoul 151-748, South Korea

Abstract

Logo has influenced many researchers and learners for the past decades as a 2D turtle geometry environment in the perspective of constructionism. Logo uses the metaphor of 'playing turtle' that is intrinsic, local and procedural. We, then, design an environment in which the metaphor of 'playing turtle' is applied to construct 3D objects, and we figure out ways to represent 3D objects in terms action symbols and 3D building blocks. For this purpose, design three kinds of representation systems, and asked students make various 3D artifacts using various representation systems. We briefly introduce the results of our investigation into students' cognitive burden when they use those representation systems, and discuss the future application measures and the design principles of Logo-based 3D microworld.

Keywords

Microworld; Logo; Representation system; Building blocks; Mathematics education;

1. Introduction

Papert introduced a virtual environment to create 2D figures using the two basic commands (rotate and forward), and learners create figures using the powerful metaphor of 'playing turtle.' Abelson and diSessa (1980) dubbed Logo as the turtle geometry, and compared it with the coordinate geometry, describing the features intrinsic, local and procedural.

Development of the spatial sense is one of the major goals in mathematics education. The Principles and Standards of the National Council of Teachers of Mathematics (2000) recommends the visualization and reasoning of 2D and 3D as a major capability to be enhanced by students. In particular, computer utilization is essential in visualization of the spatial sense in 3D.

The 2D Logo figures and 3D building blocks are introduced in Korean elementary mathematics textbooks. As for Logo, the Java Applet environment developed to enable making commands in Korean is introduced in mathematics textbooks for third and fourth graders. Students, then, learn about the metaphor of 'playing turtle', and are involved in activities to produce rectangles and equilateral triangles. Meanwhile, 3D building blocks are covered in textbooks for the second and sixth graders in Korea with a focus on daily activities. Here, 3D building blocks refer to three dimensional objects produced by connecting cubes of the same size. 3D building blocks are excellent manipulative to develop space sense with regard to mental rotations and visualization. Some mathematical activities applying 3D building blocks are finding the number of cubes piled up or looking for the arithmetic and geometric patterns.

If one thinks of the process of connecting cubes to construct 3D building blocks in a procedural manner, the metaphor of 'playing turtle' is useful for this process. In other words, if connecting one block to the next block is thought of as the 'forward' in the turtle's gesture, choosing what planes of the existing cubes to newly connect to is the 'rotate' in the turtle's gesture. In designing a Logo-based 3D microworld environment, the turtle creates a cube that surrounds its periphery whenever it moves, and the process of connecting cubes is represented by string action symbols. In this paper, we introduce the artifacts made by students as well as students' construction activities using various 3D representation systems. We examine the cognitive burden students bear when they use 3D building blocks representation system, and examine situations where each representation system is effectively applied.

2. Theoretical perspective

Ackermann (2004) argued that in comparison of Papert's constructionism and Piaget's constructivism, although both theories are of the same view, the roles of the media are more emphasized in constructionism. Here, the media being emphasized is a physical construction environment for mental construction. Likewise, Kafai and Resnick (1996) said that the core of constructionism is the mental construction through physical construction.

- Constructionism is both a theory of learning and a strategy for education. It builds on the "constructivist" theories of Jean Piaget, asserting that knowledge is not simply transmitted from teacher to student, but actively constructed by mind of the learner. Children don't get ideas; they make ideas. Moreover, constructionism suggests that learners are particularly likely to make new ideas when they are actively engaged in making some type of external artifact-be it a robot, a poem, a sand castle, or a computer program-which they can reflect upon and share with others. Thus, constructionism involves two inter-wined types of construction: the construction of knowledge in the context of building personally meaningful artifacts (Kafai and Resnick, 1996).

The computer environment to implement constructionism must be the one where learners can construct the artifacts they want and construct knowledge through the very construction activity. In other words, through this learning by making, learners naturally come to know of powerful ideas through the activity of physical construction.

Bruner (1966) remarked that children's intelligence development is that of enactive, iconic and symbolic representations and that of adjustment capabilities amongst them. In Logo, learners produce the turtle's movement on the screen through the representation of symbolic commands that coincided with the turtle's gesture and their own gesture and identify the outcomes in an iconic manner. In other words, visualization in Logo is the turtle's enactive representation and the iconic representation of the outcomes of the vestige. Such a metaphor of 'playing turtle' is more dynamic than that of a static figure, which connects the external media and the individual's mind in a friendlier manner (Christou et al., 2007b).

Arabic numerals and Roman numerals are different representations for the same numerals, but are starkly different in the cognitive and calculation processes. The difference can be seen in Roman numerals versus Arabic numerals, where the same number is represented by different numerals. Cognitive operations are not independent of the symbols that instigate them (Gonzalez and Kolers, 1982). Similarly, Norman (1993) said that the cognitive burden differs depending on the representation system. Resnick and Silverman (2005) still emphasize the roles of programming in the computer fluency aspect. But the programming environment must be improved for the convenience of users. That is why we suggest heterogeneous representation systems to produce procedural building blocks using string symbols in the Logo-based programming microworld.

The plane geometry can be approached using different representations. Abelson and diSessa (1980) introduced vector geometry, a different system from turtle geometry. It is a method of producing a curve which is similar to a function graph in the vector perspective into the Leibniz style. In this case, the turtle moves to the extent of the given vector to the direction of X-axis and Y-axis. This method, thought local and procedural like the metaphor of 'playing turtle' is not intrinsic as the direction of axes are already set. These two different representation systems produce the gesture of 'forward' and 'rotate' using the metaphor of 'playing turtle', and construct local and procedural plane figures by connecting these gestures. Students might have different cognitive burden in these two different representations, which have their own strengths.

3. Design principles of 3D building blocks.

3.1 JavaMAL microworld

The JavaMAL (http://www.javamath.com/class) microworld we are to use is a web-based environment, where Logo and DGS (Dynamic Geometry System) are integrated, and the text command can be stored, modified, and communicated on the attached web-board and executed there to reconstruct mathematical objects. Programming in JavaMAL microworld is mostly text-based, but commands can also be made using a mouse for convenience. The Input in both English and Korean is enabled in programming, and the Logo commands (forward and rotate, etc.) enable not only the turtle but also other objects (point, turtle tile and turtle net) to move.

In JavaMAL microworld, forward and rotate commands are the basic movements in the turtle geometric perspective and the move commands in the horizontal and vertical directions are the basic movements in the vector and coordinate geometric perspective. Those movements can be mathematically represented using the followings.

$$\Delta s,\ \Delta\theta\ /\ \Delta x,\ \Delta y$$

Abelson and diSessa (1980) said that turtle geometry and vector geometry are two different representations of the same thing, and the two representations of the same thing can often lead to insights that are not inherent in either of the representations alone. Expecting for this insight, we are to design building blocks where the two different representations (ver. A, ver. B) are both enabled.

3.2 3D Building blocks in JavaMAL

Let us call the command to make the square face with the metaphor of ‘playing turtle’, string ‘m,’ and the command to fold the dihedral angle between faces, another string symbol, ‘<, >.’ One can make a cube by folding the square faces (shown in Figure 1), and let us represent folded cube as another string symbol, ‘s’. Then, ‘ss’ becomes a command to continuously connect two cubes. Next, newly assume the commands to changes the direction of the cube to be connected as the turtle commands of L(left), R(right), U(upward) and D(downward). Then, the representation of ‘ssRssLss’ will have the cubes connected in the order as in Figure 1 to represent building blocks. True, the same strings can enable squares to be connected instead of cubes.

Various building blocks can be produced using this representation system. For instance, Figure 2 is an artifact called ‘swimming baby’. We produced an arm, a leg and a head by connecting several square faces and cubes. Cubes were made by connecting square face (m), and square face (m) and cube (s) can be simultaneously connected as in Figure 3. A railway and a train can be made using square face (m) and cubes(s), respectively. And pieces of the SOMA cube can be made into cubes (s), each piece can be connected to invisible square face (m), and the connected invisible faces can be folded. Note that this would generate the animation effects to connect the SOMA cubes (shown in Figure 3).

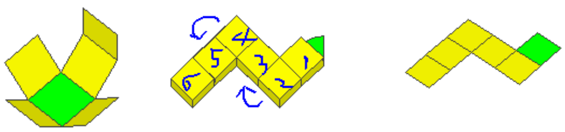


Figure 1.

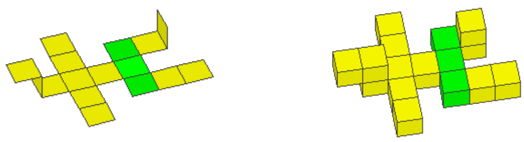


Figure 2. Swimming baby

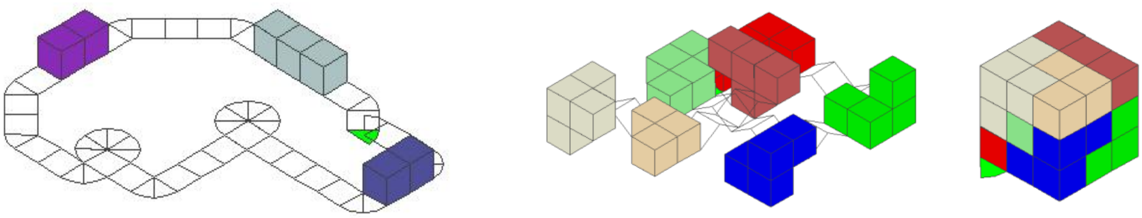


Figure 3. Train and SOMA cube

3.3. Representation system related to the metaphor of ‘turtle playing’

Some researches try to represent 3D object using its frames, where each frame of the 3D object is drawn by placing the turtle in 3D space. For instance, MaLT (Kynigos and Latsi, 2007) and VRMath (Yeh and Nason, 2004) produce the turtle’s gesture in 3D and visually show the vestige of the turtle (shown in Figure 4, Figure5, respectively). These environments were proposed by

Abelson and diSessa (1980) as turtle geometry view point in which cubes are piled up following the commands of the turtle. Let this kind of representation system be ver. A.

In ver. A, only ‘s’ makes cubes with the turtle moving forward, and commands ‘R, L, U, and D’ change the head direction of the turtle without making cubes. Ver. A uses the metaphor of ‘playing turtle’. However, although intuitively understandable and successful outcomes are generated by the metaphor of ‘playing turtle’ in 2D, difficulties might arise in connecting them to our daily activities in 3D. A turtle’s act of turning right and turning left on a 2 dimensional horizontal flat plane befits our daily context, and turning upward and turning downward are influenced by gravity as they are equivalent to walking on the vertical wall, which has a different context from our daily one. Likewise, Morgan and Alshwaikh (2008) said that students find it difficult to connect the daily gestures to the metaphor of ‘playing turtle’.

- While students were making use of similar gestures, we found that the meanings they tried to make with the gestures were different from those anticipated, and that the “playing turtle” metaphor did not easily transfer into the 3 dimensional context (Morgan and Alshwaikh, 2008).

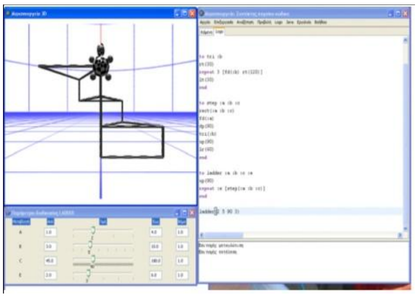


Figure 4. MaLT

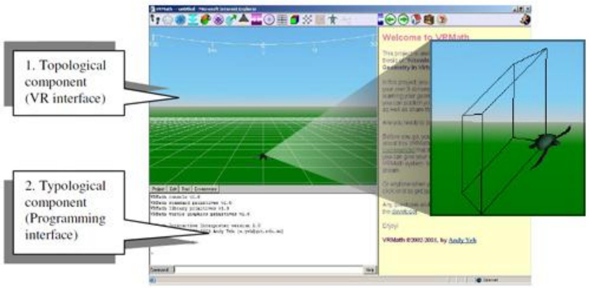


Figure 5. VRMath

3.4. Representation system related to the metaphor of ‘mouse attachment’

There are several computer environments that provide the proper environments for making 3D building blocks. For instance, DALEST project (Christou, et al., 2007a) or Freudenthal Institute’s WisWeb (http://www.fi.uu.nl/wisweb/en) is a computer environment to produce and visualize 3D building blocks (shown in Figure 6, Figure 7, respectively). When you click on any face of the cubes in Figure 6, you can attach another cube to that face. Here, we think of a command system using the metaphor of adding blocks by mouse clicking. Let’s choose one of the six faces of *(n-1)*-th cube and connect the *n*-th cube to be added to that chosen face. This is similar to the aforementioned vector geometry: three axes of a space are preset and strings that connect blocks along the axis are preset. For instance, blocks are added in such directions where string

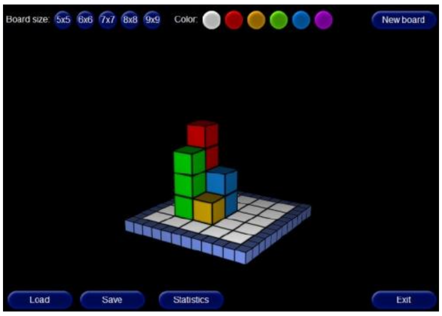


Figure 6. DALEST Project, Cubix Editor

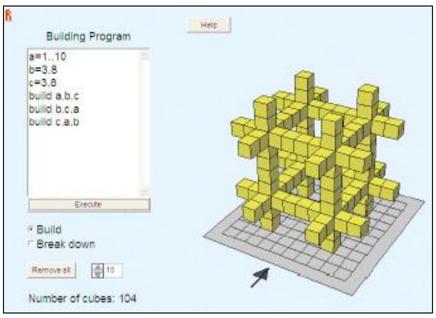


Figure 7. WisWeb

‘s, b’ is in the X-axis direction, ‘r, l’ is in the Y-axis direction, and ‘u, d’ is in the z-axis direction. We call this kind of representation system ver. B.

3.5 Three kinds of representation systems of 3D building blocks.

Let us be reminded that the 3D procedure is hard to be connected to our daily ‘gestures’ in ver. A. We would better think of representation mechanism more similar to our daily gestures. Of course, the representation system, this time, must be procedural. Kynigos and Latsi (2007) mentioned that when we move in real 3D space the up and down directions are usually stable because of gravity. Moreover, we walk in a 2D horizontal plane while the 3D turtle moves in different planes in 3D space. In a similar context, the reason why it is difficult to connect daily gestures to building blocks is because of ‘turn upward(U), turn downward(D)’. Thus it seems nice to introduce the metaphor of ‘elevator’, or ‘layers’ (combined version of ver. A and ver. B), by considering several horizontal and vertical layers. Within the horizontal layer, turtle moves and rotates related to the action symbols ‘s, L, R’. But within the vertical layer, turtle move upward or downward related to the action symbols ‘u, d’, which represents the construction of a cube one level upward and one level downward. In other words, we use ver. A for a turtle movement in horizontal direction, and ver. B for a movement in vertical direction. We might compare it to a scene where a turtle is on an elevator to mover around within a building to create building blocks. We call this representation system ver. C.

Ver. C is a form that combines ver. A’s commands ‘s, L, R,’ and ver. B’s commands ‘u, d.’ And ‘u, d’ in ver. C are equal to u = ‘UsD’, d = ‘DsU’ in ver. A. The three kinds of representation system can be summarized as in Table 1. We guess that the representation system ver. C might relive the difficulties among students having difficulty in ver. A.

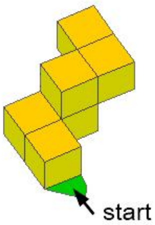
Movement	ver. A	ver. B	ver. C
Horizontal	Move Forward (s) Turn Right (R) Turn Left (L)	Move Forward (s) Move Backward (b) Move to the Right(r) Move to the Left (l)	Move Forward (s) Turn Right (R) Turn Left(L)
Vertical	Turn Upward (U) Turn Downward (D)	Move Upward (u) Move Downward (d)	Move Upward (u) Move Downward (d)
	ssRsUsDsLs	ssrurs	ssRsusLs

Table 1. Three kinds of representation system of 3D building blocks

4. Design activities with 3D building blocks.

Papert (1980) said that conversation with a computer must be as natural as learning French by living in France instead of learning a foreign language in a classroom. Resnick (2008) said that among Papert’s ideas, he was significantly influenced by ‘hard fun' and ’lower floor and higher ceiling.' We derive from the same idea. We expect that through the designing activity of making their own artifacts in the representation system of building blocks, the students could feel ‘hard fun,’ reflect on their own building block procedure, and build up their knowledge.

4.1 Free design activities of 3D building blocks

We introduced the representation system of 3D building blocks and carried out a design activity to enable them to make their own artifacts in the training program for gifted middle school students and in the teacher training program at Seoul National University, both of which were conducted in the summer of 2009. We introduced three representation systems to the participants, who then created the artifacts whatever they wanted using 3D building blocks, and gave each artifact a name. Figure 8 is an artifact named "P-51B Mustang Flying Fighter" made by 7th grader, and Figure 9 named "A Sailing Ship" and "Yu-na Kim, the Figure Skater" respectively made by mathematics teachers.

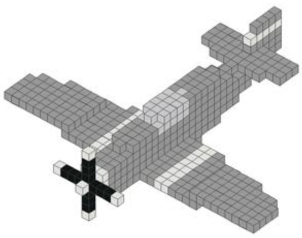


Figure 8.

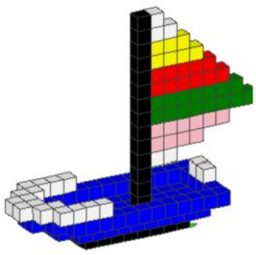


Figure 9.

4.2. Study

Students tend to prefer ver. B and ver. C to ver. A when they performed free design activities, but they prefer to use ver. A under the circumstances where they had to use recursive algorithms or production rules. As such, we conducted a simple test to see which one between ver. A and ver. C was regarded as the more difficult representation system by participants and how difficult it was in which circumstances. Ver. B was excluded in the experiment as it was believed to be meaningless in terms of measurement as the turtle's head direction is consistent.

We conducted a simple test for a total of 26 students in the first and the second grade in middle schools who were taking the training program for gifted students at Seoul National University in the winter of 2009. We only analyzed the data of 24 students leaving out two students because their correct answer rate is below 20% and the average time spent for each question was less than five seconds, which showed they had not taken the test seriously enough.

The test was conducted on a computer basis in a classroom where PCs were already set up. Questions were posted on the Web, and students' answers and the time spent for each question were recorded using the LRS (Learner Response System) installed in a JavaMATH server. Participants could start the test by pressing the start button, spend up to 60 seconds per question, and move onto the next question once they chose an answer.

There are fifteen ver. A problems and fifteen ver. C problems consisting of heterogeneous string representations. Thus students answered a total of 30 questions, and the questions were presented to students randomly without order. They scored one point if they got it right and zero point if they got it wrong, and each version had the full score of 15 points.

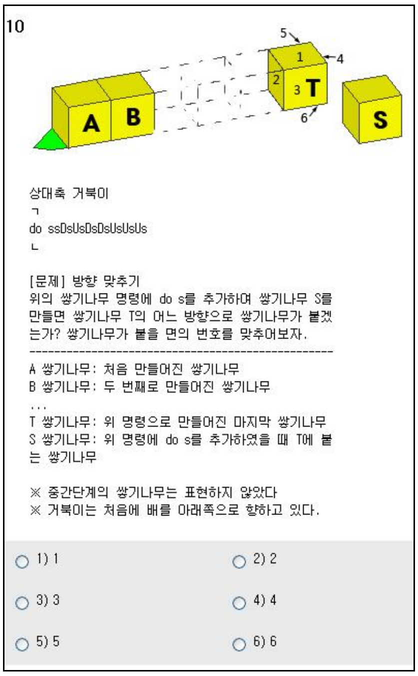


Figure 10. Example of the problem

As shown in Figure 10, the string representation and the resultant 3D building blocks were omitted in the middle part and only the first and the last cubes were suggested in each test problem. We then asked the participants where it would be attached on the last cube when a new cube was added. Participants were asked to input their answer by choosing one of the six faces of a cube. For instance, we suggested 'ssDs' to students and then asked them where the next 's' would be connected.

We analyzed the total scores and the total time spent by the 24 students in each ver. A and ver. C test using the paired sample t-test. As a result, they scored lower points in ver. A than in ver. C, but it took longer responding time in ver. A. The outcomes are summarized in Table 2. The total scores and the total time spent showed significant differences between the two 3D representation systems when the statistical significant level is 0.05. (t_1=-0.15, p_1=.000*; t_2=4.026, p_2=.001*)[1]. Out of the 15 questions, it was also reaffirmed that a question related to the combination of vertical and horizontal gestures showed significant differences. Moreover, the mixture of the vertical (D, U) and horizontal (R, L) action symbols raised the degree of difficulty in ver. A.

	Ver. A (turtle metaphor)			ver. C (elevator metaphor)		
	Mean	SD	%	Mean	SD	%
Assignment done (max.15)	10.46	3.16	69.73	13.54	2.32	90.27
Total time (s)	397.17	128.75		333.21	127.32	
Time per assignment (s)	37.97			24.61		

Table 2. Result

4.3 Discussion

As guessed earlier, the turtle's gesture in 3D was confusing to participants when the commands of 'turn upward and turn downward' were mixed. As emphasized by Papert (1980), the representation of building blocks using the turtle metaphor has educational significances in that various mathematical circumstances can be thought of by serving as an 'object to think with'. However, as it is not naturally connected to daily gestures, it might generate cognitive burden. As such, a representation system easily relatable to daily gestures could enable students to maintain a more comfortable status. In spite of this, we need to acknowledge that things that are easily accepted cognitively may not be always the best as 'hard fun' and 'higher ceiling' were emphasized by Papert. We need to use each version properly as considering which version is the most useful in which circumstance.

Students usually used ver. B and ver. C when they were asked to create whatever they want, but we observe some situations where they found ver. A more useful. Using the 3D representation system of building blocks, we were able to think about the "metaphor of growing" through the string substitution rule s='ss'. We observed the students who conducted s='ss' string manipulation after they made their artifacts freely. For example, we observed a student who made a soccer player with ver. A trying to increase its size twice by extending in the vertical and horizontal direction twice. Note that ver. A command has only string 's' which generates 3D building blocks (shown in Figure 11). However, those who made a soccer player with ver. B or ver. C commands could not use the string substitution rule s='ss'. Through such string manipulation, the students were actually able to feel the power of turtle metaphor system ver. A.

[1] t_1 and p_1 represent t-value and p-value respectively in total score comparison of ver. A and ver. C, and t_2 and p_2 represent t-value and p-value respectively in total responding time comparison of ver. A and ver. C. Asterisk means p-value is less than 0.05, which is a statistical significant level.

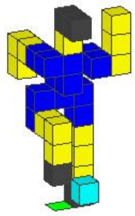
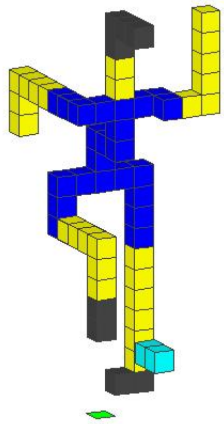
do_0	do_1
	

Figure 11. Soccer player when string 's' is manipulated to 'ss' in ver. A

5. Closing remarks

5.1 Mathematics Education

The number of blocks related to the substitution rule s='ss' has the pattern of geometric sequence. Then, what can one do to make the arithmetic sequence pattern and the Fibonacci pattern? Moreover, we can create a self-repetitive image or a recursively growing form by the string substitution, and we can think of the string manipulation with regards to probability. For example, 'z' in ver. A is a dummy variable with no special meaning. Once we make a flower by building blocks with a dummy variable, we can change the length of its leaves using the probability rule. After including z command to the both-side leaves, command z='ztscu, z[2]', and differentiate it once. Then, the probability that the length of leaves gets longer (ztscu) and the probability that the length of leaves stays the same (z) become 1/2, respectively. That is, upon each execution, the form of building blocks is to change (shown in Figure 12).

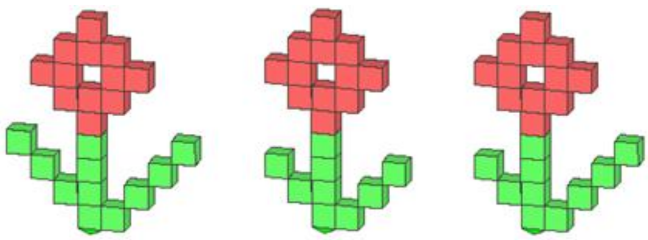


Figure 12. String manipulation of flower leaves with regards to probability

Furthermore, we can think of a case where the probabilities of z='sz' and z='LszR' are p and 1-p, respectively, and think of a case where the differentiation is made in each level by *n*th times. Here, the probability for the last cube to be positioned in a certain spot is related to the binomial distribution. This would enable mathematical knowledge to be connected to a physical

[2] The string symbol 't' means to make transparent building blocks and the string symbol 'c' means to make colorful building blocks.

substance. Trinomial distribution can be made using the same method. We expect that such an activity would facilitate learners to more easily take mathematical knowledge.

Next, learners would be able to intuitively understand mathematical features of 3D figures like Euler's polyhedral formula through building block activities. In fact, gifted middle school students intuitively explained why the value of 'V-E+F' is maintained through the activity of procedural connection of building blocks. As seen step1 from Figure 13, as one cube is added, two faces disappear as they overlap, and 4 edges and 4 vertexes disappear as they overlap. Therefore, if Euler formula works in each cubes (V_1-E_1+F_1=2, V_2-E_2+F_2=2), Euler formula also works in Euler formula (V-E+F=2) in the connected circumstance because overlapped Euler characteristic (V(4)-E(4)+F(2)=2) disappears although the sum of each building blocks' Euler characteristics is 4, which is the sum of V_1-E_1+F_1=2 and V_2-E_2+F_2=2. However, when there is an overlap of the existing parts as in step 2 and step 3, one can intuitively understand that the formula does not work. In fact, middle school students discovered the counterexample of the torus form themselves. In the torus form, 4 faces are to disappear, so they intuitively explained that V-E+F=0.

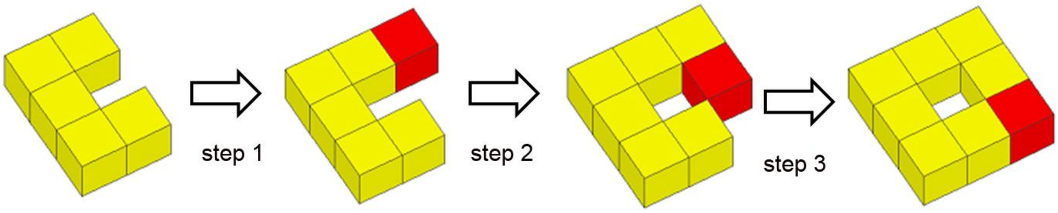


Figure 13. Euler's formula and an counterexample (exception)

5.2 Concluding

As in Logo, we consider the representation system of 3D building blocks as an intrinsic, local and procedural system. In addition, we propose 3 different types of representation systems (ver. A, B, C), and introduced several artifacts made by learners using these representation systems. We also figure out the difference of difficulties felt by students in different versions, and identified the difference through the structure of the representation systems. More researches are needed to design a desirable representation system of 3D objects.

Also more researches are needed on the three kinds of proposed representation systems. First of all, there is a need to systematically research cognitive burden of learners on different versions. It is also necessary to devise more systematic investigation techniques to measure the level of difficulties felt by learners, and to investigate how difficult learners would feel about certain occasions. Moreover, based on the researches about the level of difficulties felt by students, one needs to find out when each version is useful. Researches that utilize 3D building blocks as mathematical education tools in a classroom and investigate into mathematical effects are required as well. We believe that intrinsic, local and procedural representations of 3D building blocks and physical configuration activities under this representation system are needed for an educational goal of developing the sense of space in mathematical education.

6. Acknowldegement

This work was supported by the Seoul National University Science-Gifted Education Center.

7. References

Abelson, H. and diSessa, A. (1980) *Turtle geometry*. Cambridge, MA: MIT Press.

Ackermann, E. K. (2004) *Constructing knowledge and transforming the world.* In M. Tokoro and L. Steels(Eds.), A learning zone of one's own: Sharing representations and flow in collaborative learning environments, Amsterdam: IOS Press.

Bruner, J. (1966) *Towards a Theory of Instruction*, NY: Norton.

Christou, C., Jones, K., Pitta, D., Pittalis, M., Mousoulides, N. and Boytchev, P. (2007a) *Developing Student Spatial Ability with 3-Dimensional Applications.* Congress of European Society for Research in Mathematics Education, 5th CERME Conference 2007.

Christou, C., Pittalis, M., Mousoulides, N., Pitta, D., Jones, K., Sendova, E. and Boytchev, P. (2007b) *Developing an Active Learning Environment for the Learning of Stereometry*. EuroLogo 2007.

Gonzalez, E. G. and Kolers, P. A. (1982) *Mental manipulations of arithmetic symbols*. Journal of Experimental Psychology: Learning, Memory, and Cognition, 8(4), 308-319.

Kafai, Y. and Resnick, M. (1996) *Constructionism in practice.* NJ: Lawrence Erlbaum Associates, Publishers.

Kynigos, C. and Latsi, M. (2007) *Turtle's navigation and manipulation of geometrical figures constructed by variable processes in a 3d simulated space.* EuroLogo 2007.

Morgan, C. and Alshwaikh, J. (2008) *Imag(in)ing three-dimensional movement with gesture: "playing turtle" or pointing?* Proceedings of the British Society for Research into Learning Mathematics 28(3).

National Council of Teachers of Mathematics. (2000) *Principles and standards for school mathematics*, VA: NCTM.

Norman, D. A. (1993) *Things that make us smart: Defending human attributes in the age of the machine.* Perseus Books.

Papert, S. (1980) *Mindstorms: Children, computers, and powerful ideas,* Cambridge, Massachusetts: Perseus Publishing.

Resnick, M. (2008) *Falling in love with Seymour's ideas.* American Educational Research Associaion (AERA) annual conference, NY.

Resnick, M. and Silverman, B. (2005) *Some reflections on designing construction kits for kids.* Proceeding of interaction design and children conference, Boulder, CO.

On the design of Logo-based educational microworld environment

Han Hyuk Cho, *hancho@snu.ac.kr*
Dept of Mathematics Education, Seoul National University, Seoul 151-748, South Korea

Hwa Kyung Kim, *indices@smu.ac.kr*
Dept of Mathematics Education, Sangmyung University, Seoul 110-530, South Korea

Min Ho Song, *mino@snu.ac.kr*
Dept of Mathematics Education, Seoul National University, Seoul 151-748, South Korea

Ji Yoon Lee, *lily1982@snu.ac.kr*
Dept of Mathematics Education, Seoul National University, Seoul 151-748, South Korea

Abstract

We study to design educational Logo-based microworld environment equipped with 3D construction capability, 3D manipulation, and web-based communication. Extending the turtle metaphor of 2D Logo, we design simple and intuitive symbolic representation system that can create several turtle objects and operations. We also present various mathematization activities applying the turtle objects and suggest the way to make good use of them in mathematics education. In our microworld environment, the symbolic representations constructing the turtle objects can be used for web-based collaborative learning, communication, and assessments.

Keywords

Microworld; Logo-based environment; Symbolic representation system; Mathematics education;

1. Introduction

Papert (1980), who had proposed 2D Logo, observed that students enjoy drawing pictures. He viewed the core elements of drawing as lines and angles. Accordingly, he proposed a Logo microworld where a virtual turtle can draw pictures with lines and angles using the given embodied commands. That is to say, the Logo is designed as a virtual environment that can become a mental playing ground similar to a learner's activity of direct bodily movement. Calling this Logo-based activities as turtle geometry, Abelson and diSessa (1980) compare it to coordinate geometry taught in existing math courses at school. Logo is also a representation system that students use to express, as if speaking, various mathematical objects, physical objects, or situations by using commands that are familiar to them. Logo goes further to become a powerful tool for expressing, investigating, and analyzing mathematical objects or situations. Centered on action symbols of forward and rotate, Logo is maintains its consistency while changing representation systems according to various demands, the Logo proposed by Papert is an excellent representation system for drawings on 2D-planes. Also the 2D L-system introduced by Lindenmayer transforms forward and rotate command into f, + symbols respectively while grafting into Logo environment, becoming an experimental tool for natural phenomena, fractal investigation, and others through the resulting differentiation of representation (Wagon, 1991). Studies have also been made showing several turtles at once in order to reenact physical phenomena or traffic situations (Klopfer, Colella and Resnick, 2002). Moreover, as spatial sense of 3D-figures came to be regarded as important, a turtle moving in 3D has also appeared (Morgan and Alshwaiks, 2008).

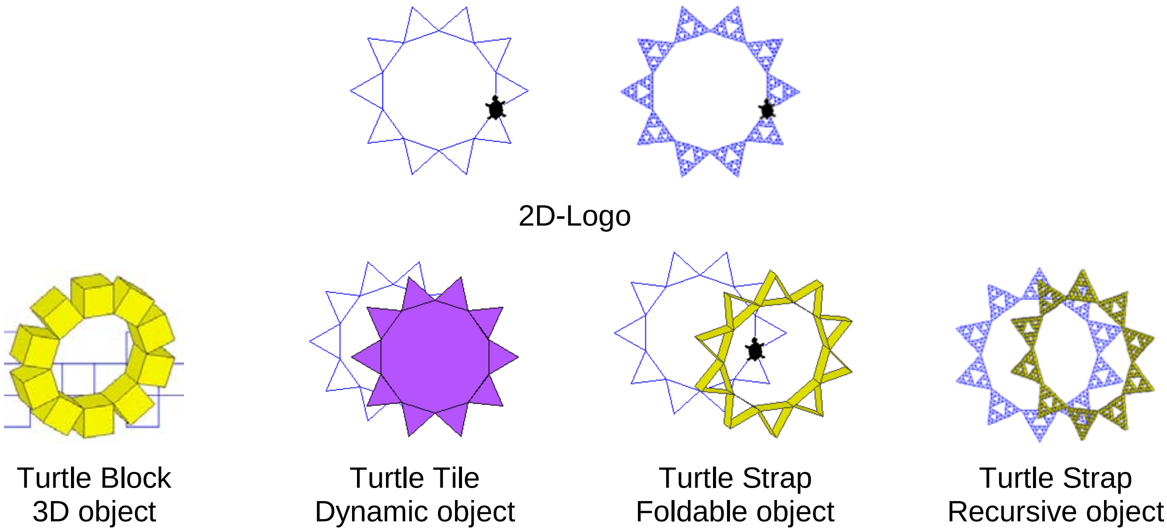


Figure 1. Turtle objects and turtle operations

The basic philosophy of Logo is to express the core of an object in the most basic language. Therefore, the application of Logo to various situations requires that one finds the core that represents the situation and that the language representing the core must be natural as well as meaningful to the learner. Extending the turtle metaphor of 2D Logo, we design simple and intuitive symbolic representation system that can create several turtle objects and operations as shown in Figure 1. The turtle objects and operations are powerful tools to construct mathematical situations for mathematization, and we suggest the way to make good use of them in mathematics education in this paper. Taking note that representation systems are based on symbols, and we also examine how the symbolic representation system constructing the turtle objects and operations can be used for web-based collaborative learning, communication, and assessments.

2. Design principle of turtle objects

Mathematical objects or situations expressed using symbolic representation system become important tools for mathematization. The process of expressing, analyzing, and interpreting objects using symbols is a must needed process in the abstraction and generalization of mathematics. Mathematical communication through symbols also enables the internalization of mathematical knowledge, allowing learners to go beyond a mere acquisition of knowledge. However, it is also true that the difficulty of such symbolic representation also becomes an obstacle for learners. Stating that knowledge with stories appropriate to situations are more valuable to learners, de Jong and Monica (1996) emphasize the importance of situational knowledge. Thus, they minimize the obstacles to learners by introducing commands that are meaningful to them as symbols while also facilitating various activities by examining symbols from a powerful-idea perspective. The translation of Logo's basic commands into meaningful symbols enables an approach to Logo through the perspective of operating symbols, and such an approach may become the link to mathematization of elementary and secondary mathematics.

Van der Meij and de Jong (2006) study representational activities through various methods of media. Mentioning how a single representation aids the understanding of another representation, they state the need for understanding previously abstract concepts more deeply through translation of representations. They also mention that the addition of meaningless information to users may only cause greater confusion. In other words, what is demanded is the activity of representing mathematical objects through symbols that are meaningful to the learners. Wilensky (1991) mentioned whether something is abstract or concrete is not an inherent property of the thing, but rather a property of a person's relationship to an object. And good concrete manipulatives are those that aid students in building, strengthening, and connecting various representations of mathematical ideas (Clements, 1999). This requires an environment in which one can find the core of one's representational object and method while representing it in a language that is meaningful to learners. As the main example of such an environment, Logo is a microworld that considers segments and angles between segments as the core elements of drawing on a 2D-plane and makes a turtle draw them. We now propose a representation system that can become a learner's language like Logo commands on an extension of the Logo turtle's action symbols.

Operating symbols can be applied to various situations that may be represented with a limited number of commands. Figure 2 is a game in which an avatar is moved in four directions to move the balls to a designated place. This game is made simply of keyboard operations, but this game can also be made of symbol manipulation as well. The symbols represented in the center represent the alien's movement in four symbols of u(p), d(own), r(ight), l(eft).

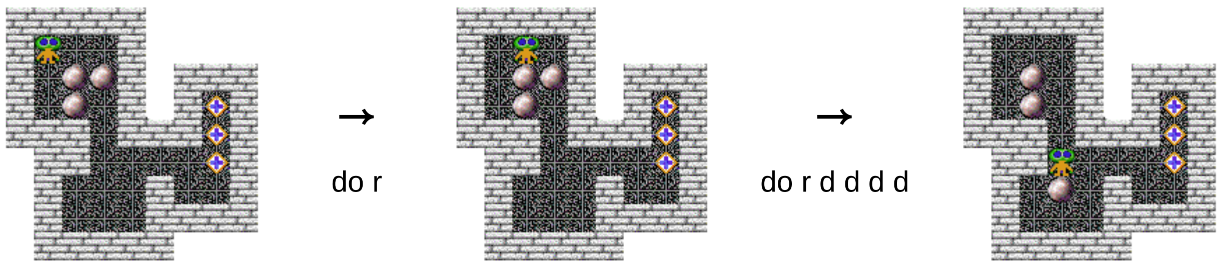
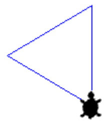


Figure 2. Symbolic representation in a game

The motion of Rubik's cube can be represented by 6 symbols: R, L, F, B, U, D. When we discuss the way of manipulation of Rubik's cube, we are apt to use these symbols. Eidswick (1986) approached to solution of Rubik's cube by using those symbols algebraically. Thus even when they are games, it is possible to represent complicated situations using simple symbols to which users invest meaning; and the case also holds true in reverse. That is to say, symbol-based microworld always has to have the use of operating symbols in mind; it must achieve an environment in which users can become familiar to operating symbols and engage in activities of representing complicated objects in simple basic pieces. It is possible to execute and represent symbols using ‘do’ command in JavaMAL microworld. For example, we can replace the command ' fd ' to ' f ' and ' rt ' to ' < ', respectively, as you can see in Figure 3(forward default value is 10, and rotate default value is 120).



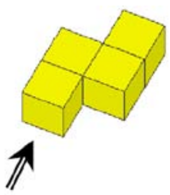
fd 10; rt 120; fd 10; rt 120;fd 10; rt 120;
do f < f < f < ;

Figure 3. Triangle (2D-figure) and two different representations

2.1. Turtle block

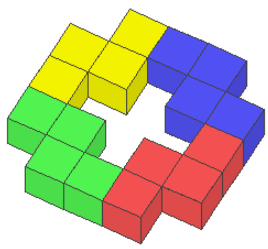
The Logo proposed by Papert (1980) is an excellent representation system of drawings on a 2D-plane. Due to the development of media, there is increased activity of operating 3D-figures as well as greater emphasis on its importance. It seems nice Logo’s representation system evolve into a system that can represent 3D-figures naturally in turtle language. The use of complicated commands will clearly enable the creation of more realistic 3D-figures, but it would be difficult to expect their use as methods of communication. When learning Logo for the first time, learners use certain angles such as ‘rotate 120’ or ‘rotate 90’ to draw equilateral triangles or rectangles; they then compound these basic figures to represent complicate figure somewhat abstractly. Similarly using turtle blocks as basic form, we represent objects as 3D-figures similarly involves representation centered on an object’s characteristic form; the representation of simplified forms of objects with turtle blocks makes turtle block commands as easy to use as natural language.

This study uses turtle objects and turtle metaphor to propose a representation system in which the turtle creates various 3D-objects. In turtle block metaphor, the turtle creates a block that surrounds its periphery whenever it moves. That is, within the same level, we can order the turtle to construct 3D-object by using s (step forward), L, R (90-degree turn to the left, right). Symbolic commands are needed for the turtle to go upstairs or downstairs in 3D-space, and we have used symbols of u (up one level) and d (down one level) for the turtle to make turtle blocks while moving up or down as if riding an elevator. The object as seen in Figure 4 is a tetracube which is belongs to the mathematics curriculum in Korean elementary mathematics education, and the right-hand side object is created connecting the basic tetracubes.



```
do s s R s L s
```

→



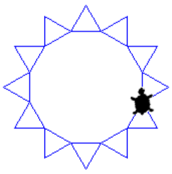
```
repeat 4 {
    do s s R s L s
    do R
}
```

Figure 4. A tetracubes and turtle block commands

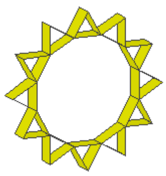
2.2. Turtle strap

Students who come in contact with various activities in Logo environment gain full experience of horizontal movement on a plane. Consequently, students need 3D representation that will provide experience of vertical, up and down movement. The turtle strap is perceived as a long paper strap that is folded here and there to create a figure. For this paper folding metaphor, the turtle uses the symbols m (move forward while making a band) and <, > (fold band towards left, right). For example, the right-hand side turtle strap in figure 5 is a 3D version of 2D-logo object by folding turtle strap at 30-degree angles, shown with the commands.

do m : make a unit turtle strap do > : folding up to right do < : folding up to left ddv : default degree value in folding turtle strap
※ In JavaMAL, '**rt'** means **'rotate'** (turn counterclockwise)**,** not **'**right turn'.



```
repeat 12 {
   repeat 3 { fd 10; rt -120;}
   fd 10; rt 30;
}
```



```
ddv = 30;
repeat 12 {
   repeat 3 { do m >>>> ;  }
   do  m < ;
}
```

Figure 5. 2D Logo commands and turtle strap commands

Note that the turtle strap is just a translation of a 2D-logo figure into a 3D-figure, and the turtle strap is a flexible concrete manipulative that can makes possible the activity of actually folding and unfolding a paper strap. As an application, Figure 6 gives an example how to represent physical rolling bridge in terms of turtle strap and turtle folding operation.

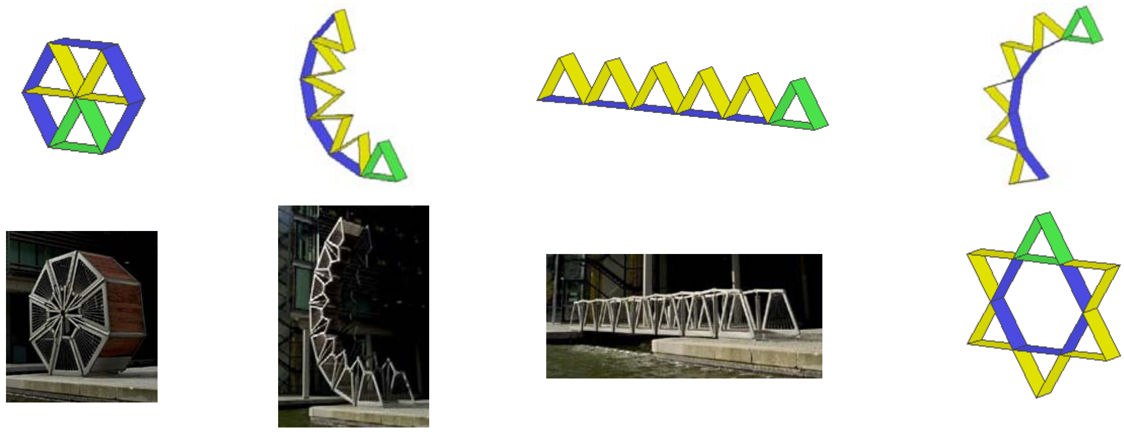
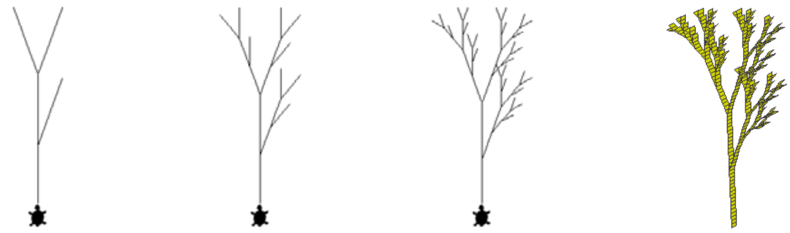


Figure 6. Rolling bridge and its turtle strap representation

2.3. L-system

Recursive functions play an important role in deductive and inductive reasoning and are widely used to explain natural phenomena such as fractals in 2D Logo. But Recursive functions are usually taught in secondary or university computer science curriculum because they take a difficult form of citing themselves. Proposed by biologist Lindenmayer, L-system is a formal representation system of diagrammatizing plant growth given by initial value and production rules. Wagon (1991) introduces turtle commands into this L-system to propose an L-system composed of turtle action without using difficult recursive functions. Following the same idea, we have defined f as the symbol for the turtle's forward movement of a certain distance, < and > as symbols for left, right turn of certain angles, and [and] as the symbols for saving and citing the turtle position and direction. Figure 7 shows production rules and tree growing image in each step; in reality, the trees in Figure 7 double in size as they move from left to right, but the image has been adjusted to identical height in order to compare the shape of the trees.

Production rule : X=' f [> X] f [< X] > X ' and f = ' ff '



Do_2 X; do_3 X; do_4 X; do_5 X; with turtle strap

Figure 7.Trees growing and turtle step representation

3. Mathematical objects and mathematization

3.1. Mathematical objects

Various objects, such as those in Figure 8, can be made using the turtle objects introduced above. The important point here is that an environment has been designed for the natural use of commands, like natural language, regarding the basic elements of representing solids. Such command is used in the same context of existing Logo, and several representation systems can be shared or used in combination. For example, turtle block can be used to make turtle shape in Figure 8.a. As in Figure 8.b, they can be used to understand the equation 1+3+5+7+9=25, visualizing the sum of odd numbers becomes a perfect square. Figure 8.c uses both L-system and Turtle strap at once. Here, each tree in Figure 7 is presented in the form of mathematical bar graph; the horizontal strap presents the domain of function while each tree presents a function value.

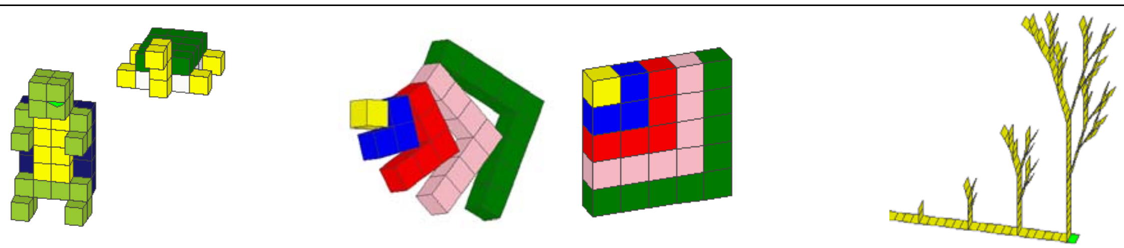
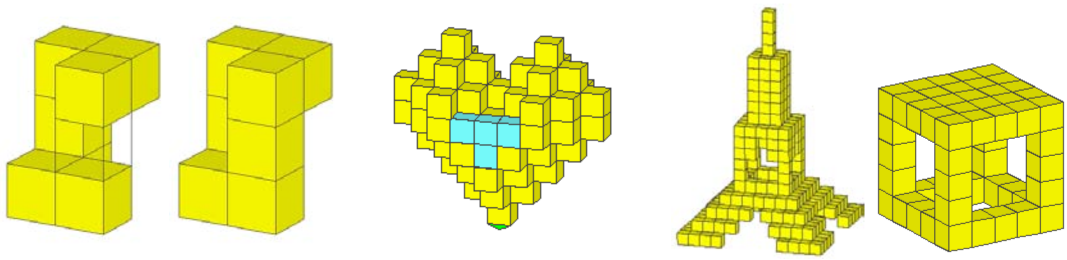


Figure 8.a Figure 8.b Figure 8.c

Figure 8. Mathematical objects

Turtle blocks can assist in the formation of mathematical objects for the learning of discrete mathematics-related content. For example, in the case of Euler characteristic number (v-e+f), all 3D-figures in South Korean curriculum are limited to cases in which the Euler characteristic number is 2. The Euler characteristic number can become a powerful investigation tool for various 3D turtle blocks as shown in the Figure 9. As the turtle blocks can provide various Euler characteristic numbers not equal to 2, the turtle blocks become appropriate thinkable objects for mathematization activities regarding Euler characteristic numbers. The figures shown in Figure 9.a is made by attaching 8 turtle blocks. The Euler characteristic number for this figure is 1 (Figure 9.a, left, a portion of the figure's center is made invisible). And the Euler characteristic number for the 3D heart (Figure 9.b) is 2. How about the Eiffel tower (Figure 9.c, left)? The answer is -6. The Euler characteristic numbers of two objects in Figure 9.c are the same. In fact, two objects in Figure 9.c are homeomorphic.



Euler #	1	2	?
	Figure 9.a	Figure 9.b	Figure 9.c

Figure 9. Turtle blocks and Euler characteristic numbers

3.2. Mathematization

Good manipulatives are those that aid students in building, strengthening, and connecting various representations of mathematical ideas (Clements, 1999). Because Turtle strap is a 3D-object, it is possible to do rotating or folding operation, and this enables the viewing of the object's movement from various side (for example, examination of only the x-axis or y-axis changes) or the examination of figure-to-figure transformation process through folding animation. L-system provides recursive behavioral patterns according to generating principles. Instead of focusing on turtle behavior, these patterns highlight the relationship between a figure's parts to its whole. Although the meaning found from turtle behavior had begun recursive thinking, the focus eventually lies in the translation of behavior into symbols and the finding of regularity between symbols. Focusing neither on drawing order nor turtle behavior but on the generating principles and change of symbols corresponds to a raise in standard.

Such activities are needed for mathematization as construct mathematics from the realistic phenomena. Figure 10 is an example of drawing mathematical concepts out of various phenomena, combining them and facilitating deep understanding of the mathematical concept through the learning by designing. Figure 10.a shows the physical parabola that would appear when making a hole in the sand box, and Figure 10.e shows a mathematical parabola that is constructed using the function equation $y=x^2$ since Figure 10.b represents a bar graph for the function $y=x^2$. Counting the blocks in Figure 10.c can be interpreted as an integration of the parabola and Figure 10.d represent a turtle strap constructed by attaching tangent segment of the parabola.

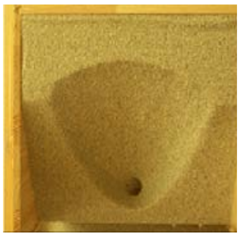
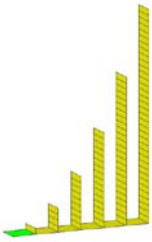
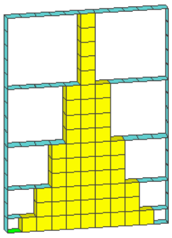
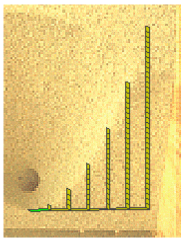
				
Physical parabola	Function equation	Integration	Differentiation	Mathematical parabola
Figure 10.a	Figure 10.b	Figure 10.c	Figure 10.d	Figure 10.e

Figure 10. Physical parabola and various mathematization of the parabola

4. Design of Communication Environment

Since our proposed representation system is based on action symbols, it is possible to combine them to represent various mathematical situations, and the representation can be saved, modified, and transmitted to others in web-board format in JavaMAL microworld. The activity of writing, revising, and transmitting ideas through web-based microworld is fundamental to create a web-based communication environment for mathematics education. Connections with the Learner Response System also enable immediate observation of a learner's present learning condition in an experimental environment. Going beyond previous levels of simple questioning over the exactness of knowledge, one can observe learning conditions in following microworld activity. The metaphor of Logo's turtle moving along with the mouse can also be used to create an electronic blackboard where learners exchange opinions on the microworld screen as a whiteboard.

4.1 Communication: Web-board system

Representing various media in a single space while designing their whiteboard, Hwang, Chen and Hsu (2006) introduce a function of representing and saving letters on a screen. The two methods of representing letters on the screen are input through a keyboard and drawing with a mouse. We have linked each of the two methods with turtle behavior. Input through a keyboard was linked to a turtle writing where he is, and letters drawn by a mouse were linked to a turtle following a mouse. When keeping in mind that the turtle draws extremely complicated drawings in Logo, the act of drawing letters while following a mouse becomes a very simple operation. When letters are drawn through a metaphor of 'turtle following mouse,' it becomes possible to represent this procedure with turtle behavior. That is to say, when this can be textualized and saved, and when this can be used to save and execute JavaMAL commands on a board, it becomes possible to communicate simple content as if writing on JavaMAL.

Text-based JavaMAL microworld representations can be easily linked with web-boards. Clements (1999) mentioned that benefits of computer environment are recording, replaying and linking between a concrete and symbol. The web-board in (http://www.javamath.com) is designed for linkage to JavaMAL microworld. As shown in Figure 11, the board is divided largely into two sections. On the left is JavaMAL microworld, and on the right is the board contained JavaMAL microworld commands and others replies. Commands are delivered between each section through JavaScript. When JavaMAL microworld commands and descriptions are written in the board, the board displays separate JavaMAL command buttons and a command screen. Because JavaMAL commands are displayed on that board in text format, it is unnecessary to save one's work, install programs, or saves files in order show others.

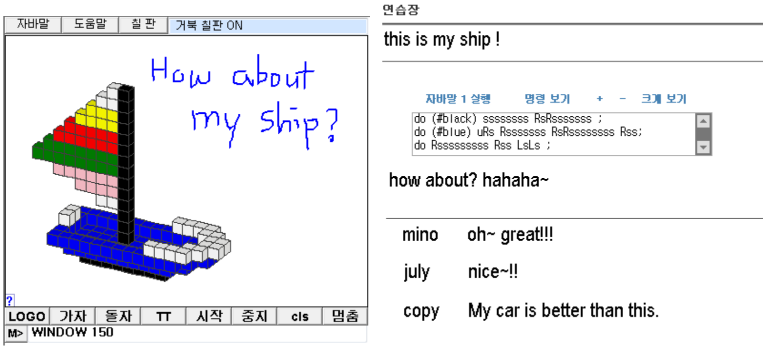


Figure 11. JavaMAL microworld and Web-board communication

4.2 Assessment: LRS (Learner Response System)

Comparing constructionism to constructivism, Ackermann (2004) mentions that the role of media is stronger in constructionism. Kafai and Resnick (1996) also mention constructionism as a mental construction made by physical construction; the physical construction in this case is constructed of a combination of several media, and Papert (1980) proposed Logo as an example of such media. There are many other studies aimed at introducing media in education. Hwang, Chen and Hsu (2006) present an environment that integrates various media through a whiteboard that they designed themselves, using it to study the influence that such an integrated environment has on the learning of a learner. The point of Hwang's white-board is that the learner represents opinions regarding a single topic by voice and writing. This presents a possibility that many students will express their opinions as well as receive other people's opinions and revise them. Studies have also been made on interactive response systems such as Clicker in order to assess the current learning conditions of students in the classroom scene using media.

Bruff (2009) mentions that the use of IRSs increase students' class concentration, class participation, and class enjoyment, having the effect of providing valuable feedback regarding the class to both instructor and learner. In particular, Roschelle (2003) discusses the role of IRSs in a network environment. This study integrates microworld to network-based media, such as computers, to construct a richer learning environment. LRS (Learner Response System) is a type of assessment tool used to measure current comprehension conditions of learners in class. The system operates in a similar manner to network-based IRSs such as WILDs mentioned by Roschelle (2003). The LRS introduced in this study was developed for use in conjunction to JavaMAL microworld. Here, learners connect to web-boards, connecting to LRS as if reading postings. They then experiment in the given JavaMAL environment and respond based on the experimentation. This is possible because JavaMAL microworld is a text-based environment. Through alteration in settings, LRS can also be changed into a testing environment for prepared questions. In Figure 12, on the left is the LRS screen for statistics, in the right is the LRS screen for question and in the middle is the JavaMAL microworld for experiment.

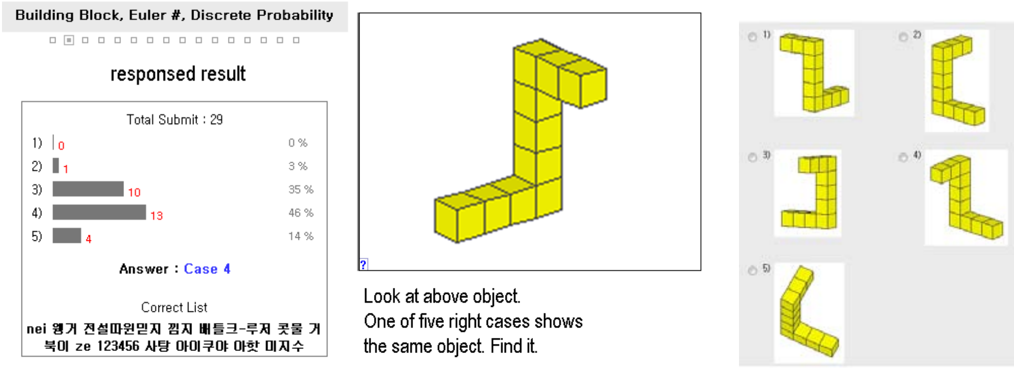
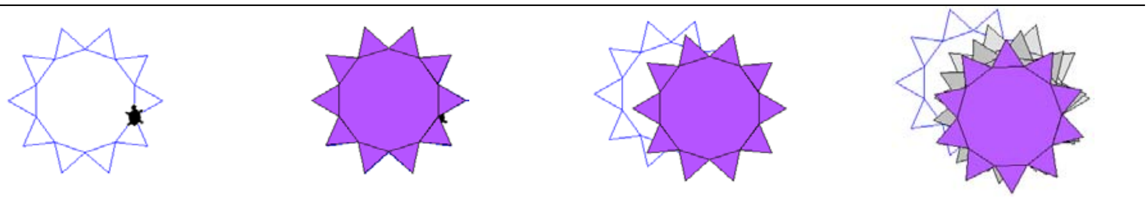


Figure 12. Framework of the LRS (Learner Response System)

5. Closing remarks

5.1. Turtle tiles and turtle nets

Cho, Han, Jin, Kim and Song (2004) showed that drawings made with Logo's turtle can be made into turtle tiles and given movement animation. Turtle tiles are dynamic objects that can be given rotating, moving, and flipping operations with a mouse. Whereas Logo draws fixed drawings, turtle tile changes fixed drawing into movable objects. Here, a metaphor of 'a turtle laying eggs' has been used. The Logo's turtle lays eggs while moving to draw pictures. And at the command to make turtle tiles, turtle tiles are made into the polygon created when the eggs are connected. Figure 13, left, is a base drawing that the turtle makes while hiding invisible eggs on the way. Figure 13, right, shows turtle tiles that are made by the hidden eggs and moved by the mouse or commands.



2D Logo	Make a turtle tile	Move the turtle tile	Rotate the turtle tile

Figure 13. Turtle figure and turtle tile

3D-figures can be made by attaching several turtle tiles and folding them in controlled angles. The turtle moves while laying eggs even when making a development figure, and newly created turtle tiles attach to existing turtle tiles as turtle tiles are created by laying two eggs on top of two previously-laid eggs. In other words, in order to be added, a new turtle tile must share one side with an existing turtle tile. Figure 14 shows the order in which the turtle created and attached turtle tiles, along with the folding of the development figure.

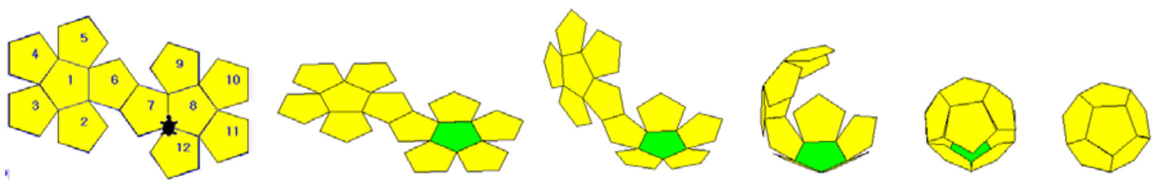


Figure 14. Turtle net and polygon folding

5.2. Turtle vector geometry

Abelson and diSessa (1980) call Logo as turtle geometry, comparing it to coordinate geometry while stating its characteristics as intrinsic, local, and procedural. However, depending on the situation, there are cases that require the turtle to have an absolute axis. Introducing the metaphor of ‘turtle on flowing river,’ Cho, Kim and Song (2006) show that concepts of basic calculus can be introduced to lower-grade students as pre-calculus. In the Logo proposed by Papert (1980), the turtle moves across a white field covered with snow. Now suppose that the turtle is on a river that flows from left to right at a regular speed, and the turtle is swimming upstream towards the opposite side of the river. Consequently, when the turtle stays still, it will flow along with the river; and when the turtle swims at a regular speed, it will move in a diagonal towards the upper right direction. This is the metaphor of ‘turtle on flowing river.’ As the most important core concept in Logo, action symbols are not merely symbols that represent turtle movement but are concepts that can develop and expand into various directions. Move command is a vector representation that combines move in horizontal and vertical directions. Whereas Logo’s move commands of ‘forward’ and ‘rotate’ are relative, movement using ‘move’ becomes movement on an absolute axis.

Abelson and diSessa (1980) state that turtle geometry and vector geometry are two different representations of the same thing, and that two or more representations of the same thing can often lead to insights that are not inherent in either of the representations alone. Cho, Kim and Song (2007) propose a circle model for drawing cycloid-family through the composition of move command, along with two representation methods for illustrating this in DGS. Figure 15 shows figures drawn by move commands and trigonometric commands.

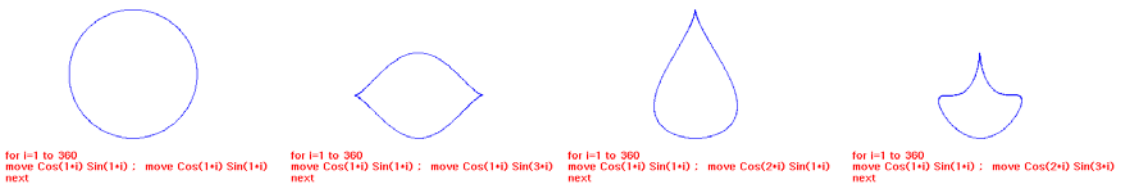


Figure 15. Figures with ‘move’ commands

5.3. Concluding

In this study, we design an educational Logo-based microworld environment equipped with 3D construction capability, 3D manipulation, and web-based communication. Extending the characteristics of 2D Logo’s behavioral symbols, we design symbolic representation system that can create turtle strap, turtle blocks, do_n recursion. The proposed symbolic representations can be used for various mathematization activities, web-based collaborative learning, communication, and assessments, and we examine actual examples of design that connects LRS and web-board to JavaMAL microworld.

6. Acknowledgement

This work was supported by the Seoul National University Science-Gifted Education Center.

7. References

Abelson, H. and diSessa, A. (1980) *Turtle geometry.* Cambridge, MA: MIT Press.

Ackerman, E. K. (2004) Constructing knowledge and transforming the world. In M. Tokoro and L. Steels(Eds.), *A learning zone of one's own: Sharing representations and flow in collaborative learning environments.* Amsterdam: IOS Press.

Bruff, D. (2009) *Teaching with classroom response systems: Creating active learning environments.* Jossey-Bass Publishing.

Cho, H., Han, H., Jin, M., Kim, H. and Song, M. (2004) Designing a microworld: activities and programs for gifted students and enhancing mathematical creativity, *Proceedings of ICME TSG 4*, 110-120.

Cho, H., Kim, H. and Song, M. (2006) The qualitative approach to the graphs of functions in a microworld, *The SNU Journal of Education Research* **15**, 129-140.

Cho, H., Kim, H. and Song, M. (2007) Mediating model between Logo and DGS for planar curves, *PME Conference 2007*.

Clements, D. (1999) 'Concrete' manipulatives, concrete Ideas, *Contemporary Issues in Early Childhood* **1(1)**, 45-60.

de Jong, T. and Monica, G. F. (1996) Types and qualities of knowledge, *Educational psychologist* **31(2)**, 105-113.

Eidswick, J. (1986) Cubelike puzzles – what are they and how do you solve them, *The American mathematical monthly* **93(3)**, 157-176.

Hwang, W. Y., Chen, N. S. and Hsu, R. L. (2006) Development and evaluation of multimedia whiteboard system for improving mathematical problem solving, *Computers and Education* **46(2)**, 105-121.

Kafai, Y. and Resnick, M. (1996) *Constructionism in practice.* NJ: Lawrence Erlbaum Associates, Publishers.

Klopfer, E., Colella, V. and Resnick, M. (2002) New paths on a StarLogo adventure, *Computers and Graphics* **26(4)**, 615-622.

Morgan, C. and Alshwaikh, J. (2008) Imag(in)ing three-dimensional movement with gesture: ''playing turtle'' or pointing?. *Proceedings of the British Society for Research into Learning Mathematics* **28(3)**.

Papert, S. (1980) *Mindstorms: children, computers, and powerful ideas.* Cambridge, Massachusetts: Perseus Publishing.

Roschelle, J. (2003) Unlocking the learning value of wireless mobile devices, *Journal of Computer Assisted Learning* **19(3)**, 260-272.

Wagon, S. (1991) *Mathematica in action*, NY: W. H. Freeman.

Wilensky, U. (1991) Abstract mediations on the concrete and concrete implications for mathematics education, In I. Harel and S. Papert(Eds.), *Constructionism*, 193-203. Norwood, NJ: Ablex.

van der Meij, J. and de Jong, T. (2006) Supporting students' learning with multiple representations in a dynamic simulation-based learning environment, *Learning and Instruction* **16**, 199-212.

Transitions

Why transitions? Because, as in musical composition, we are modulating from one key to another. Our core theme remains constant and identifiable, but its colours and textures are evolving.

The change of name from EuroLogo to Constructionism already signaled the transition to come. Early in 2008, three members of the EuroLogo 2009 Program Committee gathered in London to begin planning the next conference. Richard Noss, Ivan Kalaš, and I met in Richard's office in the London Knowledge Lab. We were later joined by Celia Hoyles at what has turned out to be our traditional London dinner spot, the Chelsea Arts Club.

While we all appreciated the quality of the interactions in Bratislava and previous EuroLogo conferences, we knew the community that had supported these events needed to be expanded, and that we must involve more young people. We also wanted to broaden the range of disciplines represented to include kindred constructionist spirits in the arts, music, dance, the humanities and social sciences. We suddenly realized that what really held us together was not simply Logo -- or any the other Logo-like or Logo-inspired languages -- but rather the underlying ideas and philosophy.

We agreed to focus on these common roots, and to explore their ramifications for learning and thinking. We would hope to take stock, look for lessons learned, the better to move forward. And so, Constructionism 2010 was born.

We have made the transition towards a conference with not just other colours and textures, but a greater variety of them as well!

We have 15 wonderful plenary speakers, some 85 papers, 5 plenary session panels, a constructionist concert, 2 constructionist dance performances, many posters and 16 incredible workshops! The whole thing will be filmed by energetic, creative AUP young people.

If our community is to prosper and grow in influence, we must seek out new participants from our traditional constituencies as well as from new ones. We need more young faces and fresh ideas. One of the panels, chaired by Paolo Blickstein, will be devoted to the work and thought of many young constructionists: graduate students, recent graduates, young researchers and teachers.

Constructionism 2010 is also about computational tools and the communities that are using them. We have panels on Scratch, organized by Mitch Resnick and his colleagues, and NetLogo, put together by Uri Wilensky and his students. They will discuss the philosophy, impact and applications of these important constructionist environments. And on Friday we have 16 workshops that will offer hands-on experience with these languages and others. Plus workshops on robotics, dance, theatre and art.

But we also have participants whose constructionist work is less dependent on digital technologies, yet who use other tools to make sense of their worlds. I hope you will find their work in music, film, dance, painting and theatre to be stimulating and informative.

And so many participants! We will be tightly packed at the FIAP, the Musée d'Orsay where we will have our conference dinner, and at the American University of Paris where the workshops will take place. But I know the week will be exciting and will re-energize us all.

There is one person, however, who, because of his accident in Hanoi several years ago, is unable to join us. Yet Seymour Papert is here in spirit and we send him our very best wishes. To honour Seymour's vision, we will have a panel discussion, chaired by Uri Wilensky, titled *Mindstorms Over Time: His Student's Reflections on Seymour Papert's Constribution to*

Learning and Education Research, to discuss Seymour's influence on constructionism, on us, our work, our students and our own acts of construction.

Seymour Papert opens the London Knowledge Lab, October 2005

I would like to thank all of our speakers, the panel and workshop organizers, the poster writers and everyone who helped to make this event a success. I thank the paper reviewers, the Program Committee and all of my hard-working and supportive AUP friends, colleagues, student helpers, communication and film and web people. I am also grateful to the President of AUP, Celeste Schenk, who unhesitatingly agreed to host our gathering and has encouraged us throughout!

A note of special appreciation goes to my Co-Chair, Ivan Kalaš, for his friendship, insight, hard work and willingness to endure endless late-night consultations via Skype. Thanks, also, to Ivan's wonderful colleagues at Comenius University, who gave generously of their time and skills.

Constructionism 2010 has been three long years in the making. But judging by the number of participants and their enthusiasm for our expanded program, it appears that our transition is well underway. The constructionist approach is dynamic; it requires us to change and adapt as we go along. Transitions are inherent in the process.

We have come this far, let us keep up the momentum.

James Clayson

A Constructionist Toolbox in the Upper Elementary Classroom - 10 Years of Integrated Robotics Projects

Mary Lurgio, *mlurgio@Smithfield-Ps.Org*
Anna M. McCabe Elementary School, Smithfield, RI, USA

Denis Coffey, *dcoffey@psinc3.com; dcoffey@lesley.edu*
Paramount Solutions, Inc. Woodstock, GA, and Lesley University, Cambridge, MA, USA

Abstract

This paper presents the evolution of successful robotics projects and processes used in a regular education classroom over the last ten years. This classroom is in Smithfield, Rhode Island, USA. The process and activities used, with students having no prior robotic knowledge, provides them a “toolbox” of knowledge and skills needed to construct meaningful learning and to demonstrate that learning to others. Their learning is applied and authentic, as is the assessment, and culminates in presentations of their projects meshed within an annual theme at the end of the academic year. The process is hands-on, differentiated and integrated throughout the curriculum. Robotics projects that have been made possible through these processes include Marionettes, Insects, Electronic Jewelry, Aboriginal Art and Landscape, Fractals, Monarch Butterflies, Cole Porter’s ‘You’re the Top,’ Feeding Frenzy Critters, Green versus Mean, Geology Bots, and Leonardo DaVinci inspired inventions

Keywords

Educational robotics, robotics in the classroom, integrating robotics, robotics

Background

Robotics in school setting has been pursued for the better part of two decades. In recent years, here in Rhode Island, the trend has been for robotics classes to move from being integrated in the public school classroom to after-school, home schooled, private schools or club activities. Reasons for this transition are varied but have been justified based on the increased time required for mandated State testing and the expenses involved with purchasing the latest technologies. However valid these reasons may be or appear to be, the authors have found that the youngsters, particularly at the elementary level, truly enjoy the learning environment that they experience while ‘doing’ robotics and that they do learn. Robotics is a verb not a noun. It is not taught as a subject area but as a way to actively engage in learning. How it is integrated across the content areas in the classroom is key to its success. The constructionist environment described here has been successful and has evolved over many years. Most of the students have not had any experience in building, modeling or sharing/explaining their learning.

So, we whole-heartedly agree with Papert (1980) who said, “But children, what can they make with mathematics? Not much. They sit in class and they write numbers on pieces of paper. That’s not making anything very exciting. So we’ve tried to find ways that children can use mathematics to make something—something interesting, so that the children’s relationship to mathematics is more like the engineer’s, or the scientist’s, or the banker’s, or all the important people who use mathematics constructively to construct something.”

What we do

The processes and actions that we instill in the students allow each student to build and to demonstrate their learning. Students are given “play” time to naturally experience materials and concepts. Students have time to explore and experiment with their ideas with the self-focus of ‘can I get my robot to do such and such?’, ‘how did you do that?’, or ‘let’s try this.’ This playtime allows for reflective writing that is just that - students reflecting on what they observed and internalized, the problem solving concepts tried out, results obtained, and the evidence of burgeoning appropriate vocabulary and teamwork.

When students write about what is personally meaningful to them, as opposed to an assigned prompt, the writing is much better in quality. It has been described as ‘technical’ writing about robotics or robotics-related activities but we say that the writing is ‘purposeful.’

Their weekly writing homework for robotics is kept in a yearlong binder. It is part of the Project’s presentation, which culminates at the year’s end state wide event-Robotics Park. Students are told on the first day of school about this expectation of participating in Robotics Park (2010), http://www.risf.net/RoboPark.htm, again, giving their work purpose. For eighteen years hundreds of students from Rhode Island have displayed their creations at this annual spring celebration of the Rhode Island School of the Future. The students present and demonstrate their projects in a given category whether Robotic Animal Design, Robotic Interactive Device, Chain Reaction Machine, the Robotics Park Parade, and/or Creature Feature Feeding Frenzy.

At the start of the school year the overall Project or big idea is announced and discussed with the students. A theme or frame for the Project is selected for their constructions. Discussions are open and their initial ideas are recorded and saved for a later day. Once the overall theme has been shared, their journey starts with learning and utilizing the tools in the toolbox. It should be mentioned that while we know the tools and the process to the end, we don’t know what obstacles or rather problem-solving opportunities that we’ll meet along the way. Hence we, the authors, learn and demonstrate to the students how we learn. Then they realize that they have partners to help them and partners willing to accept their ideas. The authors are more like coaches and facilitators that allow for learning that is more lateral as opposed to top down teaching methods.

The students are presented with an open-ended Project idea and a schedule when the Project must be completed. This is a real deadline and it is usually students’ first time with a long term project as well. In all their endeavors throughout the year they are asked how they advanced the Project and to record their progress and contributions.

The Toolbox; what and why

To construct, one needs tools. Table 1 lists the “tools” that are used and what they offer. LEGO® is the medium of choice because of the high degree of design engineering that goes into the product and the mathematical relationships that are built into the pieces. They are colorful, tactile and playful and are highly familiar to students. As the students’ experiences grow, materials other than LEGO® can and are often integrated to the mechanized creations.

Table 1. Tools and techniques used in creating designers of robots in the fifth grade

Tool	Why
#1 Three piece LEGO® Activity	Effective communication, common language, basic building; Bottom up/orientation, mathematical arrays
#2 Klutz Crazy Contraptions: A LEGO® Inventions	Extending vocabulary, following directions; Gaining understanding of which pieces fit together; Choosing a challenge, documenting and letter writing

Tool	Why
Book (Klutz)	
#3 Gearing	Simple and compound concepts, model building, gearing up/down, fractions, gear ratios, torque, friction
# 4 Fan Project	Introducing motors and where they go. Controlling motors with programming via Dacta Control Lab Logo. Introduction to sensors as control mechanisms. Words have meaning and connections to real devices via the interface box. Logo natural language is very helpful.
#4 ZNAP	Compound parts allow for bigger, quicker models. Pieces can interface with regular LEGO®.
#5 Adding Motors	Re-visiting Klutz book to take mechanical to mechanized devices sometimes under control of sensors.
#6 B-I-Y site	Exploring options for finding web resources to assist team. Choose and build a device to share with class See how parts can work together. Economical use of parts to create movements and behaviors. http://www.build-it-yourself.com/biy-blocks/localhost/index.html
#7 P.I.E. site	Same as above. PIE modules can be found at: http://www.pienetwork.org/a2z/m/modules/
# 8 Robotic Kits	Mindstorms RCX programming in Robolab and MicroWorlds (LCSI, Terrapin) Mindstorms NXT programming in NXT-G Handy Crickets programming in Cricket Logo and MicroWorlds PicoCrickets programming in Logo Blocks
#9 Programming in conjunction with a design engineering process	Create subassemblies and devices for prototypes Demonstrate basic forward, backward, left and right Which language will suit the Project the best? NXT, Pico Cricket, any Logo, Robolab, MircoWorlds (LCSI, Terrapin) Sensors- controlling variables. Choosing Project events.
#10 Ancillary materials	Finishing the robot-“dressing the skeleton” Backdrops, banners, schedules, brochures

TOOL #1 Three-piece LEGO® Activity

We start with the 3-Piece LEGO® Activity. Though deceptively simple, the activity is quite powerful. It quickly illustrates the need of a common language by which we can all communicate and understand each other. Students choose a partner and small bins of just LEGO® bricks in different sizes and colors are given to each team. A divider/carrel is placed so that Student A can choose any 3 pieces of LEGO®, stick them together in a desired configuration and then write the directions for how to build it and hand it to Student B. Student B also chooses, builds and writes directions. Then the carrel is removed. Each student now tries to build using only the directions written by the other student. Even though students have been told not to say anything or point at a piece, they find it hard not to do so when they see their partner unable to build from directions that are usually very vague and not precise. They usually write as follows: ‘get the big grey piece and stick it under the small green one and put the yellow on top and you are done.’

When students can speak, the discussion of what went wrong becomes quite lively. She didn’t know what I meant! It was right there! He picked up the wrong piece. A mini lesson is given on how LEGO® is described mathematically such as 1x6, 2x8 [Hayward (1996)]. This lesson connects their previous abstract knowledge of multiplication and mathematical arrays to the concrete LEGO® pieces along with the importance of orientation terms, such as horizontal/vertical, for a starting point. Then the process is repeated. Students are now able to follow one another’s written directions.

TOOL #2 Klutz Crazy Action Contraptions

Klutz Crazy Contraptions: A LEGO® Inventions Book by Rathjen (1998) kit contains about 70 LEGO® technic pieces and gives visual directions for making several "contraptions" and three challenges. Students again choose a partner and choose one of the listed challenges, such as: Using only the pieces in the kit to create letters of the alphabet past the letter 'r'; can you create a bridge greater than 42 inches that can be held by the ends without breaking?; can you construct a tower taller than 40 inches?

Students have to check their kit against the inventory checklist, actual sized, so the 1:1 mathematical correspondence is there. All the pieces are labeled, so the students' vocabulary is extended which is evidenced by their homework. And then the fun (problem solving) begins. No one's first attempt is perfect and they repeatedly try to better their results even at recess time. They feel successful when they have "beaten" the challenge as stated in the kit. Then they are told that previous students are the World Record Holders. This is enough to motivate some students further. Each team shares their solutions with the class and what they have learned. Each challenge has a focus; the alphabet challenge is creative and parts do not have to be attached to one another, but you need an economy of parts to create more letters and a complete alphabet. The tower challenge focuses on stability and having a sturdy base thereby sacrificing parts to create a wider base but this trade off impacts how tall it can be. The bridge challenge focus is on flexibility and weight so the design will not fall apart when lifted by its ends. Students will encounter these same issues (creative problem solving, economy of parts, stability/robust design, and weight) throughout the Project. Students then write to the Klutz Company and tell of their results.

TOOL #3 Gearing

Students in teams of 2 or 3 build a wall of technic beams and place various axle-gear combinations in the wall. They explore the connected movements and try to have every gear move when only one gear is driven by hand. Questions arise such as how many different size gears do you see? How do you measure gears? What sizes so they come in? What are the differences? Do any behaviors that catch your eye? After such exploration the idea of meshing gears is introduced. Then each team is asked to complete the following grid with a 'yes' or 'no' if the gears mesh along a beam.

Gears	**8 teeth**	**16 teeth**	**24 teeth**	**40 teeth**	
8 teeth					
16 teeth					
24 teeth					
40 teeth					

Why do some gears mesh well and others not? Are there any two gears that never mesh in any way? The terms for a driver gear, driven (follower) gear, and idler gear are explained. The motions of adjacent gears are gear ratios are discussed. Extensions to pulleys are made. Once simple gearing and its relationships are understood the students move on to the compound gear train. They build a 27:1 compound gear train using 3 combinations of 8- and 24-tooth gears. Torque and speed trade off are seen and felt and the students seem to understand the powerful and 'slow elephant' versus the reduced load that a 'fast cheetah' can carry.

TOOL #4 ZNAP

ZNAP is a discontinued LEGO® product. It looks very different from traditional LEGO® but still has the capability of interfacing with other LEGO® pieces. Many parts are compound parts (braced rectangles and curves) as opposed to simple one-element pieces (1 x 8 beams). Students discover that they can build something quite large rather quickly like a lawnmower. ZNAP parts are grouped together in bins and the design booklets are again in a visual format.

Students now have to designate a “parts person” to get all the ZNAP pieces necessary to build what the team has chosen. Some of the items that can be built have a motor powered by a battery pack and this is our focus for this activity. Cars on tracks, helicopters, and airplanes are popular choices. A wire to a battery pack tethers these first “robots” and students observe that there is only one speed. One student observed that his car kept jumping the track and that if he could reduce the speed, the car would stay on the track and thus a segue to programming.

The next few tools are worked concurrently in the classroom.

TOOLS #5 Motorizing / #6 BIY/ #7 PIE / #8 Robotic kits/#9 Programming

After working with ZNAP, which has motors, students have discovered that the motors on a battery pack were limited to one speed. So motors themselves are not the answer to creating a robot, programming movement and using sensors are the keys. Now is the time to build some subassemblies and simple devices. Dividing the class and working in small groups, students are given tasks to do within each group. Some teams are given directions to build different subassemblies. Some teams look through the online web resources at B.I.Y and P.I.E. and choose a device to build. The teams are helped with building and programming as necessary. Sharing at the end of each of these sessions is critical because everyone is doing something different but all want to know what their peers are doing. It is also expected that all teams help each another as needed. Models are kept as an “idea bank” for the Project.

These sessions evolve to further work exploring the sensor options available. Lateral learning spreads like wildfire when one team knows how to make the robot find a line, “speak” or play music. They instantly become peer teachers to the other groups. The “ooh factor” is a powerful motivator. The great part is that every group has a new bit of information to share with the others as well. Critical mass has occurred. Project focus can begin. They are ready to apply and synthesize their knowledge and use their tools.

Tool #9 Programming / #10 Ancillary

The design engineering process is continual and ongoing, as students now have chosen the event they wish to do for Robotics Park whether Interactive, Feeding Frenzy, or Chain Reaction Machine among others. Also much class time is needed to produce all the written research, publications, schedules, invitations, brochures and background artwork needed to showcase their work, as well as integrating math and language arts along the way. Robots may need to be made to look like a fairy, a Monarch butterfly or the Mona Lisa so students are not limited to just LEGO®. Art materials need to be customized for each of the robots in the project. Working with a real deadline is a new experience for students and they do feel the pressure, as the countdown to the big event looms ever closer. Students are always asked to write what they think Robotics Park will be like. And these “before” pieces of writing show nervousness and excitement.

Table 2. Showcase of projects

Grade/Year	Project	Technology / Critical Learning Ideas
5th / 2000-01	Robotic Marionettes	Control Lab; LEGO® and ZNAP
	Stage was 4’x8.’ See figure 1. Marionette Stage, Fairy and Knight	XYZ movement programmed in XY-plane with mechanical movement in Z
4th /2001-02	Stage was 10’x14.’ See figure 2. Rainforest Backdrop and RI Blue Bug	Control Lab/RCX Six-legged creatures and winged movements
4th/2002-03	Electronic Jewelry [Martin et al (2006)]	Handy Crickets using Cricket Logo

Grade/Year	Project	Technology / Critical Learning Ideas
	See figure 3. Interactive necklace	Designed and constructed interactive necklaces with programmed percent of compatibility displayed on chevron, Boolean algebra
5th/2003-04	Aboriginal Art	RCX using MicroWorlds Logo
	See figure 4A. Aboriginal landscape and Toas	Roving robot finds various toas and send notice to computer via IR signal. Computer via logo draws the particular toa on the screen.
5th/2004-05	Migrating Monarchs	Handy Crickets using Cricket Logo
	See figure 5. Rabble of Monarchs and Migrating Monarchs	Monarchs (line) follow migration paths with wings moving using only 2 motors.
5th/2005-06	Fractal Koch ArtBot See figure 4B. ArtBot picture	RCX programmed to draw a Koch snowflake.
	Chain Reaction Machine (CRM): You're the Top	RCX with Robolab and NXT software, Crickets with Cricket Logo
	See figure 6. Pictures of Mona Lisa and Tower of Pisa	Vertical roulette wheel precision control with NXT motors. Synchronization of CRM to Cole Porter's 1934 hit 'You're the Top.'
5th/2006-07	CRM: James Bond	RCX/NXT
	See figure 7. 007 in Thunderball	Homemade sensors used
5th/2007-08	Feeding Frenzy Pilot Event	RCX and Robolab
	See figure 8. CRM: Green vs. Mean See figure 9. Hungry Critters Jeffy and Curious George Jr.	Novel Feeding Frenzy Challenge designed. Time constrained search of and environment for food (CD) with happy and sad behaviors using only 3 motors and light- and touch-sensors.
5th/2008-09	Rockin' Robots	NXT integration with geology
	Feeding Frenzy	RCX
	See figure 10 of Rockin' Mine fields and Robot Sedimenter in the Grand Canyon	First robots programmed in NXT-G. Students and teachers learning new technology together.
5th/2009-10	Leonardo Da Vinci inspired creations for Feeding Frenzy and CRM.	Parachute, Dragon, Mona Lisa, Catapult, Horse, Aerial Screw, Armored Tank, Battleship Oars, Revolving Bridge. All NXT technology.used.
	CRM: Smart Transportation	Ready for Robotics Park April 10, 2010

Figure 1. Marionette Stage, Fairy and Knight

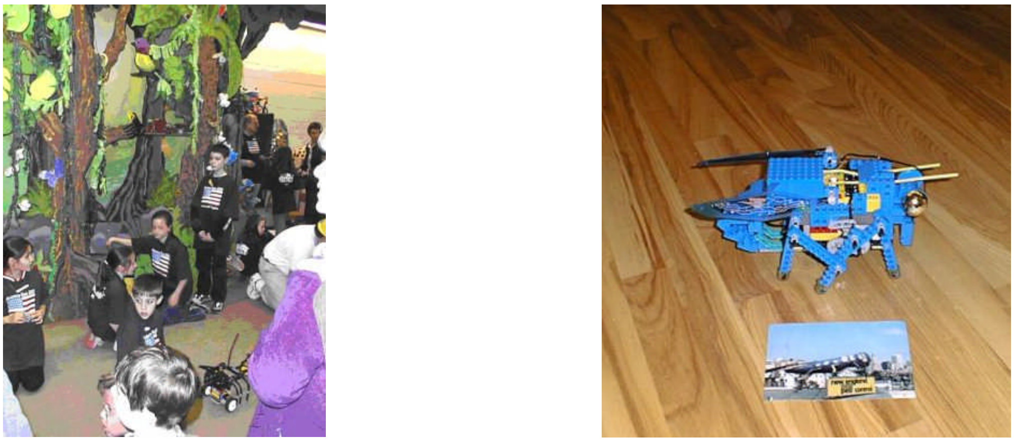


Figure 2. Rainforest Backdrop and RI Blue Bug

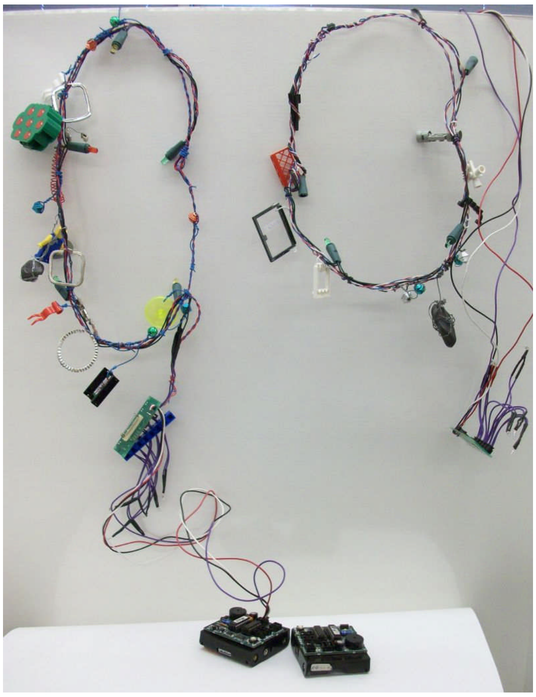


Figure 3. Interactive Necklace

Figure 4. Aboriginal Landscape and Toas and ArtBot

Figure 5. Rabble of Monarchs and Migrating Monarchs

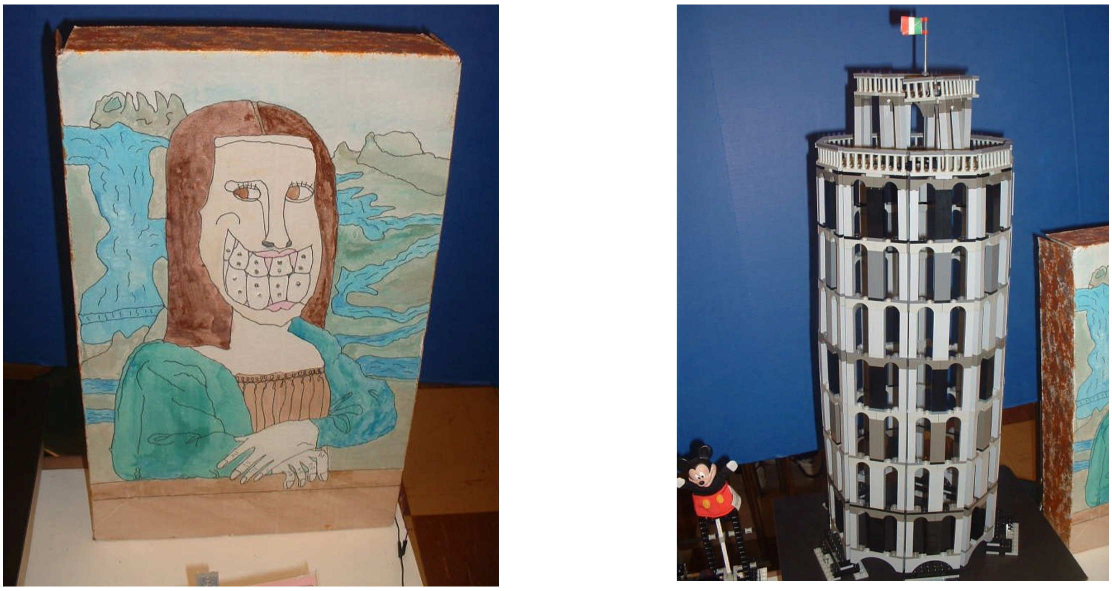


Figure 6. Picture of Mona Lisa and Tower of Pisa

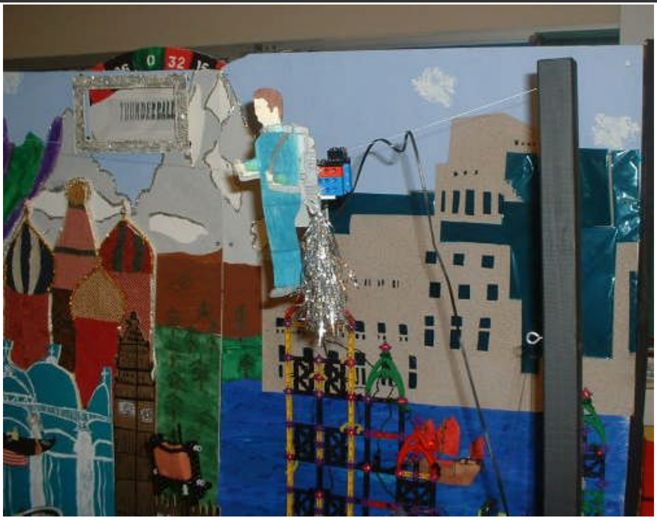


Figure 7. 007 in Thunderball

Figure 8. CRM: Green vs. Mean

Figure 9. Hungry Critters Jeffy and Curious George Jr.

Figure 10. Rockin' Mine Fields and Robot Sedimenter in Grand Canyon

Summary

SCHOOLS often want to showcase problem-solving activities but very often miss offering the students problem solving opportunities that are personally connected to the students. Robotics does offer such opportunities if the creations are open-ended and meaningful to the students. The bottom line is captured by Papert [6] who said, "One of the worst things we do in our schools is compartmentalize. We cut things in bits. One of the worst cuts we make is dividing the aesthetic from the knowledge, from the science. This is a disaster, because the source of the children's energy is largely in two areas that we see here: their social relations and their aesthetic drive. This is what produces the energy, and we cut this off."

Robotics Park showcases their products - the end result of months of hard work. The process that each student has gone through, documented in their robotic binders is their personal journey of success including all the trials and tribulations in between. The binders contain weekly homework differentiated to what each student did that day. The students **must** include a labeled drawing and their robot does not go home with them. The students are so invested in their work that they can draw from memory. They are graded for demonstrating science/math concepts. They self assess their own work by their robot doing what they wanted it to do.Does the robot work?There have been many tears shed and sometimes students want to give up. We let them know what we have asked them to do is not the impossible but the difficult and when that success comes it is theirs alone and no one can take it away from them. The payoff for us as teachers is, when this success happens, that you witness that moment of absolute joy. That is what makes having a constructionist classroom worthwhile; it is a unique educational experience. Robotics Park is their day. Parents and other onlookers are literally amazed at what children can do.

Integrating robotics into our classrooms produces energy, takes time to nurture, requires discipline and teamwork, and gives the students a real-world experience of designing, building and presenting project with a deadline. It is loud and messy, which is disconcerting to adults who think "Oh, they are just playing with LEGO®." Our projects have shown that it is so much more.

References

Papert, Seymour (1980), "Constructionism vs. Instructionism" - a videotaped speech delivered to Japanese educators in the 1980s http://www.papert.org/articles/const_inst/const_inst1.html.

Robotics Park (2010), http://www.risf.net/RoboPark.htm, Rhode Island School of the Future.

Hayward, M. (1996) Multiplication with LEGO Dacta, Tips and Techniques, Volume 2.

Rathjen, Don Klutz. (1998) Crazy Action Contraptions: A LEGO® Inventions Book (1998), Klutz, Scholastic Inc. New York

Martin et al (2006) *Robotic Jewelry in a Fourth Grade Classroom: Inventing Locally Contextualized Mathematics as Part of a Collaborative Engineering Project.* In Proceedings of 2nd International Conference Informatics in Secondary Schools: Evolution and Perspectives

Papert, Seymour, (1990) "*A Critique of Technocentrism in Thinking About the School of the Future,*" M.I.T. Media Lab Epistemology and Learning Memo No. 2.

Development of an Undergraduate Multidisciplinary Engineering Project

Dr David Ryan Smith, *david.smith@brunel.ac.uk*
School of Engineering and Design, Brunel University, Uxbridge, Middlesex, UB8 3PH, UK

Dr Jo Cole, *joanne.cole@brunel.ac.uk*
School of Engineering and Design, Brunel University, Uxbridge, Middlesex, UB8 3PH, UK

Abstract

During their time at university it is necessary for undergraduate engineering students to develop not just technical skills related to their chosen engineering subject, but to also develop team working, time management, self organisation and decision making skills that will enable them to work effectively as engineers in the real world after graduation. These important transferable skills are highly sought after by industry and any chance to identify where such skills have been successfully used during an undergraduate degree course is a valuable addition to a student's CV when subsequently entering the job market.

To address the need of developing transferable skills, the School of Engineering and Design Multidisciplinary Project (MDP) was introduced in 2007 to provide first year undergraduate students with an opportunity to work together in multidisciplinary teams on a design and construction project. Each team is comprised of students from across the range of subject areas within the School and tasked with designing and building a robotic vehicle to tackle an obstacle course. The basis for the kits provided to each team are Lego Mindstorms robots for a majority of groups while the remaining groups are provided with a Parallax Basic STAMP 2 chip and a micro-controller chip to design their vehicle around. Figure 1 shows a selection of the 50 completed project builds from the 2009 MDP, showing the wide array of designs produced by the students.

This paper describes the main aims of the MDP and gives an overview of how it has developed over the last three years to become a key part of the engineering undergraduate programme at Brunel University.

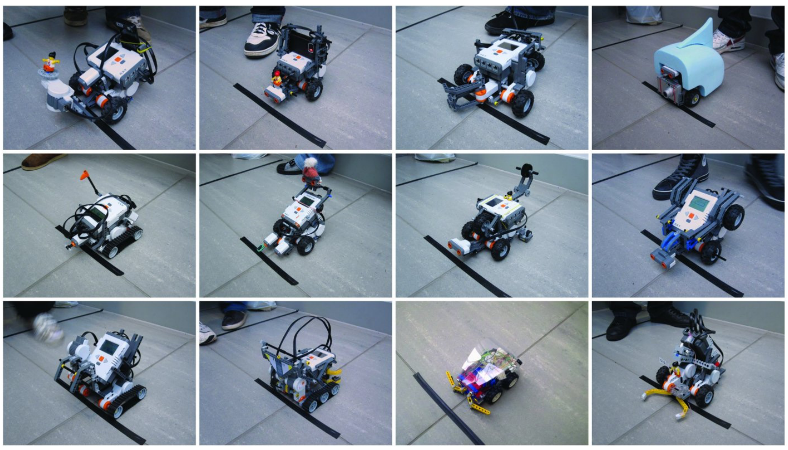


Figure 1. A selection of the wide variety of completed robots from the 2009 MDP

Keywords

Multidisciplinary; undergraduate project; large group teaching; transferable skills development

Introduction

The School of Engineering and Design Multidisciplinary Project (MDP) is a week long project based activity involving first year undergraduate students from across the School subject areas of electronic and electrical engineering, computer systems engineering, mechanical engineering, civil engineering and design. The MDP was developed as a teaching activity that would remove the barrier of academic ability to taking part by involving a non-discipline dependent technical element, the primary emphasis being on the utilisation of problem solving skills that students have begun to develop in their first term at university. Working together in mixed discipline teams would also allow the students to gain an appreciation for the many other branches of engineering there are outside their own field, many of which they will need to work closely with in industry.

A number of similar projects have been developed in the US where multidisciplinary undergraduate projects are a necessary requirement for degree courses to be accredited by ABET (Blandford et al., 2001, DePiero and Silvovsky, 2007). Although multidisciplinarity in undergraduate engineering courses in the UK and Ireland is not currently a requirement for degree course accreditation, there are a number of examples of universities incorporating multidisciplinary aspects into undergraduate project activities to address the needs of industry. Cambridge University provide a multidisciplinary 'Integrated Design Project' for level 2 students (Long et al., 2009) and a Lego Mindstorms based project activity has been successfully incorporated into the undergraduate curriculum at the National University of Ireland (Ringwood et al., 2005).

Themes and goals

Each year there are around 450 students that take part in the MDP. The students are organised into 50 mixed discipline groups of 8 or 9 students and tasked with designing, building and demonstrating Lego Mindstorms and BASIC Stamp micro-controlled vehicles to tackle an obstacle course. The 50 groups are split into five 'themes', ten groups per theme, each of which has a selection of different obstacles and hazards to negotiate on a course along with specific challenges to complete, for example: autonomous or wireless control for navigating a specific route through the obstacle course, detection and intelligent avoidance of hazards on the obstacle course, identification, collection or transportation of target objects placed on the obstacle course.

Figure 2. The MDP obstacle course layout

An overview of the technical goals for each of the project themes is given in Table 1, while Figure 2 shows two photographs of the obstacle course. The three target objects were positioned within the small black taped square along the front side of the course, students electing which of the target objects they were going to try and identify, pickup or transport at the start of their assessed demonstration run. The target objects were: a rubber egg, a small role of black electrical tape and a wooden cylinder of approximately 70 mm high and 15 mm diameter.

The technical challenge of each project theme is set at a difficulty level that all students taking part should feel is achievable, all students being able to make valuable contributions to the project work independent of their engineering discipline.

	Goals
Rover A	1. Wireless or autonomous control of a rover 2. Negotiation of a course containing a number of hazards 3. Data retrieval/analysis of a target object *Obstacle course route:* See-saw, arch, object (analysis), bridge, circular hazard, tunnel
Rover B	1. Wireless or autonomous control of a rover 2. Negotiation of a course containing a number of hazards 3. Collection and transportation of a target object *Obstacle course route:* See-saw, arch, object (collection), rubble, circular hazard, tunnel
Robot A	1. Wireless control of a robot for traversing an obstacle course 2. Capable of picking up a target object 3. Transportation and delivery of a target object *Obstacle course route:* See-saw, arch, object (collection), circular hazard, tunnel, object (deposit)
Robot B	1. Wireless control of a robot for traversing an obstacle course 2. Identification and avoidance of environmental hazards 3. Identification of a target object *Obstacle course route:* See-saw, arch, object (identification), rubble, circular hazard, tunnel
Vehicle	1. Autonomous control of a vehicle for traversing an obstacle course 2. Intelligent movement based on the environment 3. Data retrieval and awareness of the local environment *Obstacle course route:* See-saw, arch, rubble, bridge, circular hazard, tunnel

Table 1. The five MDP themes and their associated goals

Information and resources

Students are made aware of the MDP from their very first week of term via a short presentation and are then given further information in the build up to the project week. All the information is available to students on u-Link, the Brunel University web-based learning software, with printed copies of information also appearing on a dedicated MDP notice board throughout the academic term. The MDP takes place in the last academic week of December each year, a week in which all other teaching activities the students are involved with are suspended to allow sole concentration on the project.

Group and theme information and project kit inventories are made known to the students a few weeks in advance of the project week, to allow students time to make contact with their fellow group members and to do some background research on what equipment and software they will be using. The goal is to encourage all the students to be sufficiently prepared and organised that they can begin work straight away when they arrive to collect their project kit and begin the team work exercise.

Rover and Robot themes have as their basis Lego Mindstorms robotics kits which include an array of sensors (touch, sound, ultrasound and light sensors), motors, wheels, tracks and additional parts that can result in a multitude of possible designs. The central control brick is programmed using Lego software on a laptop or PC, programmes being downloaded to the unit via a USB cable. The Vehicle projects use a Parallax Basic STAMP 2 chip which can be programmed to drive Lego motors via a Pololu micro-controller. A general Lego and electronic components resource is made available to all project teams throughout the MDP week, via access to an electronics laboratory from 09:00 – 17:00 each day. Figure 3 shows some of the kit information provided to the students before the project week.

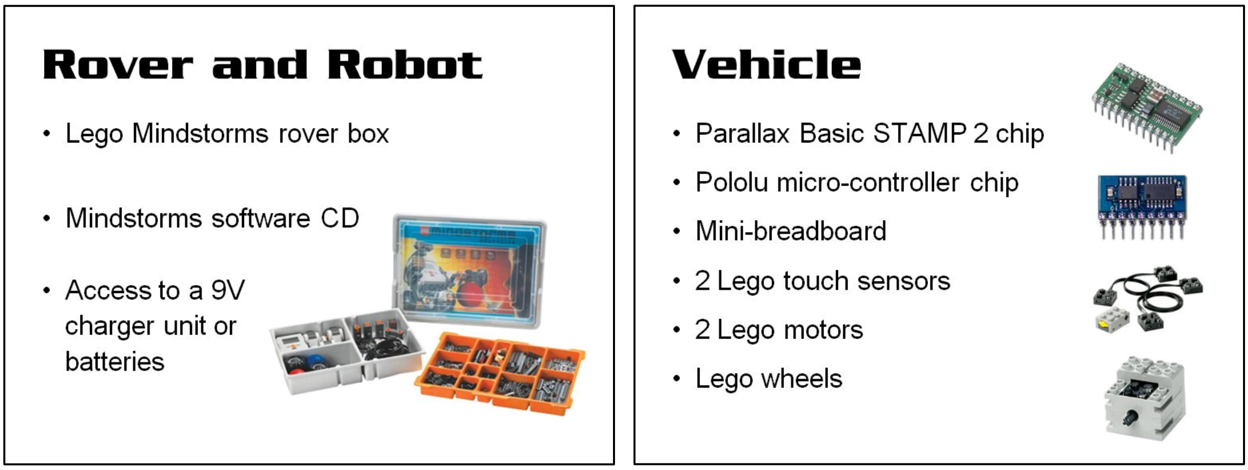


Figure 3. Slides from the 'MDP Information Pack' giving project kit inventory information

Information given prior to the project week also includes the names of members of staff available for support across the different subject areas, the location of computing laboratories with the required software installed (some of which are available 24 hours a day), the location of various laboratory and study spaces in the School that will be available for groups to work in during the project week and information about the MDP assessment components.

During the actual MDP week students are required to be present for kit collection on the Monday morning and then again on the Friday for the project demonstrations. How, when and where the students choose to work during the week is entirely down to them, the effectiveness of how well the group worked together and organised themselves being aspects of the MDP assessment.

Assessment and learning outcomes

The MDP is housed within a different teaching module in each of the different engineering subject areas, taking up the same module weighting in each case. For example, in electronic and electrical engineering the MDP counts for 25% of the first year workshop module. The stated module learning outcomes directly related to student participation in the MDP are:

- Design, build, test, evaluate, document and present small prototype systems to a given specification
- Undertake personal evaluation and reflection
- Work effectively as part of a team
- Communicate effectively in a professional manner

These learning outcomes are assessed in two parts, firstly by a group demonstration of the finished project build and secondly by submission of an individual report from each student. Each assessment component is worth 50% of the total MDP mark and is described in more detail below.

Group demonstration

Group demonstrations take place on the Friday morning of the project week, each team nominating a ‘driver’ for their completed robot and being assessed by two academic staff members as they tackle the obstacle course and their theme specific challenges. The group demonstration mark is assessed based on three categories:

- Analysis of the design problem (10 marks)
- Design choices made (10 marks)
- Success of the final design (30 marks)

The demonstration mark is awarded to the group as a whole, all group members receiving the same mark and grade dependent on their attendance at the demonstration and observed presence and contributions made throughout the project week by the rest of the group.

Individual report

Each student is required to submit an individual report by the end of the Wednesday during MDP week. A Microsoft Word template for the report is provided to all students, consisting of three sections that need to be completed. The three section headings are as follows:

- *Project Description* – a brief description of what your project was about and what the aims of the project were. You should include information about any background research carried out, design choices made and the reasons behind them.
- *Team Work* – a description of how your project group was organised. How was your group managed? Did you have group meetings? How were the group’s activities scheduled? How successfully did your group work together as a team? Were there any problems encountered and how did you overcome them?
- *Personal Contribution* – a description of your individual contribution to the project. Comment on the success, or otherwise, of your contributions and of the project so far. Do you think all of the original goals of the project will be achieved? What would you change or do differently if repeating the project?

The submitted report is then assessed on the following categories, the first three corresponding to the three report sections given above, the final category being self explanatory:

- Critical evaluation of the technical design (10 marks)
- Reflective review of how the team worked together (10 marks)
- Reflective review of personal contribution to the project (20 marks)
- Quality of the written work (10 marks)

Feedback on both the demonstration and individual report aspects of the MDP is provided, detailing a mark in each of the assessment categories along with supporting written comments and a final overall grade in each case.

Project demonstration day

The MDP obstacle course is set up in a large indoor area with several viewing levels that can accommodate the multitude of students and staff that come to watch the demonstrations. Taking place on the last day of term, the demonstrations allow staff and students to get together one last time before the winter academic break, providing most students with a good story to tell when they go home to visit friends and relatives. The atmosphere is always very good throughout the presentation day, with students cheering each Rover, Robot and Vehicle as it makes its way out of the tunnel and heads towards the course finish line. The excitement is highlighted by the addition of a competitive element, a prize being awarded to the best demonstration in each of the five themes. A selection of Robot theme projects from the 2007 MDP is shown in Figure 4.

The obstacle course is set up from the start of the MDP week to allow students to test their designs and is a hub of activity for the whole five days. Over the last two years the School has held open days for prospective engineering students during MDP week and a visit to the obstacle course to see the students working on their projects has become part of the tour, receiving positive feedback from parents who get to see what is going on.

Figure 4. A selection of Robot theme projects from the 2007 MDP

To give an idea of how the project demonstration marks are calculated it is worth contrasting two projects, one that obtained a very high group demonstration mark (grade A, 85%) and another that obtained only a threshold pass mark (grade D, 44%). The grade A project from 2009 was particularly impressive in that its design incorporated many elements that were not to be found in the general build manual present in the Lego Mindstorms kit. The group had opted to use a free rolling plastic ball as the rear 'wheel' of their Rover which greatly enhanced the manoeuvrability and enabled the Rover to easily traverse the obstacles on the demonstration course without hitting walls or falling off edges. The Rover B task of collecting and transporting one of the target objects was also achieved flawlessly by the use of a well designed motorised claw. The claw included a finely controllable gearing mechanism to open and close the pincers along with metallic attachments on the end of each claw that detected the successful collection of the wooden cylinder target object. This project demonstration clearly showed that the group had successfully analysed the original design problem and made sensible and novel design choices which resulted in a smooth run around the obstacle course.

In contrast, the grade D project from the same year was a cumbersome design very closely resembling the base unit design provided in the Lego Mindstorms manual. The Robot did not perform well when tackling the obstacle course and got caught up on the see-saw (not enough ground clearance), crossed the boundary of the circular hazard several times (poor steering control) and struggled to get through the tunnel (no use of sensors to detect the tunnel walls). The grabbing mechanism incorporated on the front of the Robot was also not designed to be strong enough or wide enough to physically hold the chosen Robot A target object (the rubber egg). These factors were strong evidence that little analysis of the original design problem had been carried out by the students and poor design choices had been made during the project week, the group allocating little or no time to trying out their Robot on the obstacle course and interacting with the different possible target objects.

It should be noted that no clear correlation has been found between a low group demonstration mark and a low individual report mark, students being able to describe their own design choices and ideas (that may or may not have been incorporated into the final project build), problems they encountered during the project week and provide suggestions for how they could have worked more efficiently and effectively in their team, independently of the success or otherwise of their group project demonstration.

Variations on a theme

The MDP has seen a number of changes over the last three years but the key aim of developing the transferable skills of Brunel University engineering students has endured. In the first year there was an additional project theme, 'Rocket', that involved teams making small model rockets and putting some type of sensor into the egg shaped nose cone. Wireless cameras were provided to each team as a default payload however some teams opted to include other types of sensor, such as an accelerometer or thermometer. Although the assessment of the project was purely based on the build of the rocket and demonstration of a working payload on the ground, all ten rockets built that year were successfully launched on an extremely cold December day on a university playing field. This project theme, although being arguably the most exciting and challenging of the different projects, was dropped after the first year for primarily weather reasons but also because of the need to clear the rocket launches with local airports. In the subsequent two years of the MDP all projects have been indoor obstacle course based to ensure all demonstrations can actually take place.

It is worth pointing out that trying to organise a multidisciplinary project that fulfils assessment criteria from across a range of subject areas is no easy task. The initial development of the MDP was marked by a high level of staff resistance to trying something so radically different in the teaching programme and a lot of effort was required by those involved with the MDP to actually make it happen. This resistance seems to have been common across the sector for some time (Denton 1997). The only way to satisfy all subject areas involved was to place the assessment emphasis on the development of transferable skills, such that the individual report component could be marked using the same assessment sheet across all subject areas. Each year there has been debate amongst the different subject area academics about changing the emphasis of the assessment to lie more 'in their area' or to change the weighting of the MDP in their respective teaching modules. Such changes have so far been resisted, the whole nature of the project being 'multidisciplinary', with students from each subject area in each team, essentially being lost if the students are each assessed differently depending on their specific engineering subjects.

Although the core content of the required MDP individual report has not changed over the three years, the length and format of the report along with the submission date have changed each time. The current MDP year saw the report reduced to two pages in length from the original five, the students being provided with a template document to ensure certain word limits and specific content requirements are adhered to in each of the three sections. This year also saw the individual reports submitted during the MDP week rather than at the start of the January term. This ensured that the students had to think more carefully about evaluating their personal contribution to the project and how their group design was going to perform on the day of the demonstration without being able to refer back to the demonstration run itself. These modifications resulted in the best selection of submitted individual reports to date, with many more students showing successful reflection on and evaluation of their own contributions to the project than they did when submitting reports several weeks later when thoughts about the project have long since passed.

The bullet point list below is a sampling of quotes taken from individual reports submitted as part of the 2009 MDP that clearly demonstrate the main aims of the project were not only understood, but achieved by a number of the students taking part:

- “Modern day engineering projects make it mandatory for engineers to be able to function as a team and I think it is useful that we learn this at an early stage of our degree programmes”.
- “I have really enjoyed this project. I have met new people and made a few friends, learnt new things and generally had really good fun which I will admit going into this week, I didn’t think I would”.
- “Some of the group did not believe they would be able to make significant contributions with their knowledge and skills, but at the end of the week it was obvious that each individual could contribute to the project in their own way”.
- “Our group did the work successfully as a team as everyone had an opportunity to express their views, opinions and plans regarding the project”.
- “The multidisciplinary project original goal I believe was to get us ready for the real world working environment where we would have to work with other engineers to solve problems and create new things. This project showed me that I can get a task done with others even if we haven’t known each other for very long and we can discuss and agree on a task and complete it very easily if we work together as a team. Overall, I believe the multidisciplinary project to be a success because it showed me that other engineers can work together to create greatness”.

Conclusion

The MDP project continues to develop over time, with increasingly impressive project designs and demonstrations being given by the students and more students writing good reflective and evaluative individual reports each year, showing that they have learned some of the transferable skills the MDP was designed to teach them. In addition to practicing verbal and written communication skills, project students have set up groups on social networking web sites to communicate their group work activities during the project week and a number of videos taken by students and by the sensors mounted on the rovers, robots, vehicles and rockets themselves have appeared on the internet. Students have also spoken to staff about positive comments from potential employers made when discussing their involvement with MDP in job interviews.

The MDP also fulfils a number of other objectives beyond the stated learning outcomes it was initially designed to address, the project week being a lot of fun for students and staff, improving staff student relations and social cohesion across the different subject areas within the School. Many students have commented that they made valuable new friendships during the project week with students from the other engineering subject areas that they would not have otherwise had the chance to meet. With a number of students each year asking when they will get to take part in a similar multidisciplinary project again, the type of teaching activity the MDP entails certainly seems to aid in breaking down any prejudice students may have about working with colleagues outside their own narrow disciplines.

Feedback from students after taking part in the MDP is generally very positive, the key exception being students from civil engineering who feel the most 'left out' when working with the other students in their group. Discussions about trying to tackle this problem are planned before the next MDP week. One possible solution is the inclusion of a civil engineering challenge on the obstacle course, for example a bridge must be designed by each team of students to specifically allow their vehicle to traverse a gap.

Lessons learned from the development of the MDP are applicable to group work activities in later undergraduate years in engineering, an area currently under investigation within the School of Engineering and Design with regard to multidisciplinary activities in MEng level programmes. This point was noted following a recent accreditation visit to the School by the Institution of Engineering and Technology (IET), the review document from the panel stating that "The panel commended the multidisciplinary project in the first year and was pleased to learn that this concept will be developed for use elsewhere in the programmes". The MDP focus on transferable skills means that it also has the potential to bridge between Schools. Knowledge gained from the MDP is currently being applied to an internally funded research project at Brunel looking into the development of project based undergraduate teaching activities with teams comprised of students from the School of Engineering and Design and the School of Arts.

For academics considering the development of multidisciplinary undergraduate activities in engineering programmes, studies carried out by Loughborough University describing necessary planning measures for undertaking such activities (Denton, 1997) and documentation describing a large array of project based teaching activities by the Project Based Learning in Engineering consortium based at the University of Nottingham (PEBL Consortium, 2003) are recommended reading.

References

Blandford, D. K., Hwang, D. J., Richardson, A. (2001) *LEGO 101: A Multidisciplinary Freshman Team Experience*. Proceedings of the 2001 American Society for Engineering Education Annual Conference and Exposition. Albuquerque, NM, June 2001.

Denton, H. G. (1997) *Multi-disciplinary team based project work: planning factors*. Design Studies, 18(2), 155-170.

DePiero, F. W., Slivovsky, L. A. (2007) *Multidisciplinary experiences for undergraduate engineering students*. Proceedings of the ASEE Annual Conference. Honolulu, HI, June 2007.

Long, P. J. G., Matheson, J. M. R., Palmer, P. R., Parks, G. T. (2009) *1B Integrated Design Project: Mobile Robot*. University of Cambridge course manual.

PEBL Consortium (2003) *A Guide to Learning Engineering Through Projects*. University of Nottingham report 43/99 PEBL: Project Based Learning in Engineering.

Ringwood, J. V., Monaghan, K. and Maloco, J. (2005) *Teaching engineering design through Lego Mindstorms*. European Journal of Engineering Education, 30(1), 91-104.

Constructing models and teaching modeling: difficulties encountered by pre-service teachers

Christiana Th. Nicolaou, *chr.nic@ucy.ac.cy*
Learning in Science Group, Department of Educational Sciences, University of Cyprus

Constantinos P. Constantinou, *c.p.constantinou@ucy.ac.cy*
Learning in Science Group, Department of Educational Sciences, University of Cyprus

Abstract

Modeling ability is a basic scientific process that connects theories and scientific data. Models are scientific constructs used for assessing the applicability of hypotheses, forming hypotheses and developing the mechanisms supporting the functionality of the physical phenomena. In the Didactics of the Natural Sciences the importance of the modeling ability lies in the fact that it could act as a medium for supporting the learning process and the development of students' learning. This research study aims to identify the difficulties encountered by pre-service teachers (PsTs) during modeling-based learning and teaching. The participants of the study were 21 PsTs of a science specialization course at the University of Cyprus during spring semester 2007. The purpose of the course was twofold: help PsTs develop the modeling ability through model construction and refinement and guide them to develop teaching strategies for promoting modeling-based learning. Curriculum was developed for both purposes. A series of diagnostic tests administered prior and after the implementation of the curriculum, the transcripts of the synchronous discussions pertaining to critique of the current educational research about the modeling ability, and PsTs' reports regarding action research studies aiming to the development of the modeling ability of elementary students constituted the means of data collection. Techniques of Phenomenography and Content Analysis were used for the analysis of the qualitative data collected. Analysis of the results indicated that PsTs efforts to construct models and teach the modeling ability are distracted by specific epistemological and pedagogical difficulties. Further quantitative analysis indicated that these two types of difficulties are correlated (Figure 1). Based on the qualitative interpretation of this relationship, three theoretical didactical approaches regarding teaching the modeling ability were sketched out: linear (blue rectangle), object-oriented (red rectangle) and aesthetic (green rectangle) modelers. Educational implications of these results are discussed.

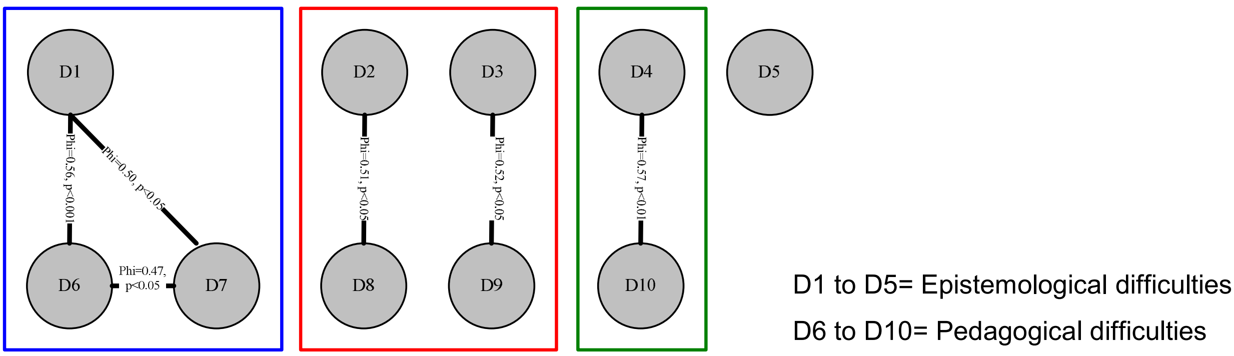


Figure 1. The Relationship between Epistemological and Pedagogical Difficulties

Keywords (style: Keywords)

Modeling ability, teaching the modeling ability, learning difficulties, teachers´ education

Introduction

Science teachers should learn about science, about current research on how students learn science, and about how to teach science. Typical undergraduate science courses don't promote these parameters and shove teachers to focus primarily on content (Duschl, Schweingruber, & Shouse, 2007) ignoring that students should be aware of the nature of science and participate in scientific practices. They ignore that teachers are learners that learn better and as a consequence teach better, when constructing knowledge or when teaching through knowledge construction (Papert & Harel, 1991). Modeling could constitute a way to overcome this drawback and promote scientific proficiency, as it is closely connected with understanding of the nature of science (Gilbert, 1991), and contributes to the active construction and revision of knowledge (models) by students themselves (Hestenes, 1987). Model construction is in line with the principles of constructionism, as teachers construct understanding of certain phenomena through the development of specific artifacts, models (Kafai & Resnick, 1996).

It is however possible that the construction process is obstructed by learning difficulties, which relate either to the content of the instruction or the pedagogical implementation of it. The study presented here is part of an on-going research program, through which we aim to investigate the development of the modeling ability of and the development of strategies to teach the modeling ability for PsTs working collaboratively. Specific modeling and pedagogical difficulties encountered by the PsTs are presented and their interdependence, as well as their role in knowledge construction process is discussed. The research questions investigated are: (1) Which are the difficulties related to modeling encountered by PsTs when constructing a model? and (2) Is there any interdependence of these difficulties with PsTs' effort to develop and implement interventions promoting modeling-based learning?

Theoretical framework

Models and the modeling ability

A *model* is a unit of structured knowledge used to represent observable patterns in physical phenomena. It acts as an external representation of a phenomenon that provides a mechanism accounting for the functions of the phenomenon and can be used for predicting the future behaviour of the it (Halloun, 2007). Moreover, models are scientific constructs used for assessing the applicability of hypotheses, forming hypotheses and developing the mechanisms supporting the functionality of the physical phenomena.

The *modeling ability* refers to the ability to construct and improve a model of a physical object, a process or a phenomenon (Hestenes, 1987). It is a basic scientific process that connects theories and scientific data. In the Didactics of Natural Sciences the importance of the modeling ability lies in the fact that it could act as a medium for supporting the learning processes and the development of students' learning. As a process of learning and teaching, it is compatible with constructionism (Kafai & Resnick, 1996), introduced by Seymour Papert and which is based on the constructivist ideas of Piaget, which support that knowledge is constructed by the experiences of the learner and suggests that humans learn when constructing new knowledge that involves tasks aiming to the development of artifacts, in this case models.

Moreover, modeling is an ability that includes reasoning skills necessary for the development of the learners' epistemological awareness. In order to facilitate the learning and teaching process, modelling, which is a complex ability, should be analysed in three constituent components:

modeling skills, meta-cognitive knowledge, and meta-modeling knowledge (Papaevripidou, Constantinou, & Zacharia, submitted). *Modeling skills* are: (a) Model formulation, (b) Identification of model components, (c) Comparison of models of the same phenomenon, (d) Model evaluation and formulating ideas for improvement, and (e) Model validation through comparison with phenomena in the same class. *Metacognitive knowledge about the modeling process* refers to the learner's capability to describe and reflect on the major steps of the modeling-based cycle. M*eta-modeling knowledge* refers to the learner's development of epistemic awareness regarding the nature and the purpose or utility of models.

Central to the process of teaching the modeling ability is the modeling-based learning cycle. The modeling-based learning cycle is a refinement of the learning cycle (Karplus, 1980) which consists of five nor discrete or linear parts (engage, explore, explain, elaborate and evaluate). As such, the modeling-based learning cycle is considered iterative in that it involves continuous comparison of the model with the physical system in reference. The purpose is gaining feedback for improving the model so that it accurately represents as many aspects of the system as possible. It is also cyclical in that it involves the generation of models of various forms until one can be found that successfully emulates the observable behavior of the system.

We consider that scientific modeling procedures can be simplified, so that they are coded in unison in the frame of one scientific framework. In other words, this research study does not refer to the possible differentiation of the scientific modeling procedures in different cognitive areas. This assumption is justified in the frame of a simplification for the purposes of teaching transformations for elementary education. This study focuses on inductive models and does not refer to construction of hypothetico-deductive models. This assumption is also justified in the frame of teaching transformations aiming to influence elementary education.

The role of teacher as central for learners to both use and understand the nature and role of models is emphasized by many researchers (Coll, France, & Taylor, 2005; Justi & Gilbert, 2002a; Justi & van Driel, 2005; Stylianidou, Boohan, & Ogborn, 2005). However, teachers do not hold scientifically correct ideas about models and modeling (Crawford & Cullin, 2004; Gilbert, 1991; Harrison, 2001; Justi & Gilbert, 2002b; Van Driel & Verloop, 1999, 2002). The next sections elaborate on the didactical obstacles and the consequent difficulties or ideas of teachers and students when constructing knowledge.

Learning difficulties

Learning difficulties are organized in several categories, which are of the same nature with the constituent components of learning in the Natural Sciences. Learning pertaining to understanding of ideas, concepts and principles of the Natural Sciences and provides the means through which students can think about unknown and new physical systems, refers to *conceptual understanding*. The acquisition of experiences with natural phenomena (*experiences*) provides the basis for the subsequent development of concepts and skills (Wellington, 1994). *Positive attitudes* towards inquiry feed student motivation and safeguard sustainable engagement with the learning process (Gibson & Chase, 2002; NRC, 1996). When students' thinking and understanding is away from the scientific, with regards to one of these three areas (conceptual change, experiences, attitudes), students face <u>conceptual difficulties</u>. Students' understanding regarding the essential principles of the nature of science, the structure and the development of science and scientific learning relates to *epistemological awareness*. Obstacles that emerge during student's effort to capture the essence and the structure of the epistemological procedure and the nature of science in general fall under the category of <u>epistemological difficulties</u> (Halloun, 1998). *Reasoning skills* provide the strategies and procedures for making operational use of one's conceptual understanding, in order to analyze and understand everyday phenomena, but also to undertake critical evaluation of evidence in decision making situations. Reasoning difficulties dissuade students' effort to develop these skills and constitute students incapable of describing explaining or understanding the underlying mechanism of phenomena or physical systems (Hammer, 1996). *Practical and scientific skills* relate, among others, to students' ability to (i) predict, (ii) design and carry out fair experiments, (iii) conduct detailed observations, (iii) use instruments for collecting data, (iii) collect, code, organize and interpret data, (iv) communicate results, conclusions and other information, and (v) raise investigative questions (Gott & Duggan, 1995; Gott & Duggan, 1996). Students' weakness to capture these skills leads to (a) practical difficulties, which relate to students' handling of instruments or tools in a way which leads to distortion of the results of an experiment, and (b) reasoning difficulties, when for example students are incapable of raising investigative questions.

Research in students' understanding and therefore the identification of learning difficulties could constitute a means for serving multiple didactical goals. Firstly, the development of specific scaffolding steps aiming to the appropriate direct manipulation and confrontation of difficulties and the development of student learning is supported. More specifically, teachers' awareness of the existence of specific difficulties leads their moves and strategies in the learning environment and allows for the development of questions that guide students' thinking and therefore the construction of knowledge and understanding of the phenomenon under study. Secondly, identifying the learning difficulties serves some indirect goals. It helps towards the development of didactical strategies and activity sequences that deal with learning difficulties. Curriculum design should include the development of specific strategies that encourage students to express their views so that difficulties are revealed and become the subject of dialogue and discourse. Awareness of students' learning difficulties also leads the development of appropriate assessment tasks, which evaluate whether the curriculum is successful in confronting the difficulties identified and therefore whether it fulfilled its goals.

Methods

Sample and Intervention

The participants of this study were 21 fourth year elementary teachers of a science specialization course, which adopted a blended e-learning approach, at the University of Cyprus (spring semester 2007). The intervention was based on an iterative procedure, which involved the learners in an active process of constructing and deploying successive models of the moon phases collaboratively, within their group, among groups in class or through the internet (Blackboard Learning System). The first part of the curriculum included tasks related to (a) studying moon data, extracting patterns out of them and developing hypothesis explaining these patterns, (b) constructing models using Stagecast Creator© describing some of the identified patterns, and (c) deploying models. This procedure of constructing and deploying of models was supported by the Virtual Learning Environment. For example, Group A constructed a model and uploaded it on the *Tool for Exchange of Contributions for Peer Reviewing* and then deployed the original model, based on the feedback included in the Assessment form of Group B, also posted on the Tool. Finally, Group B had to deploy its model based on the feedback included in the Assessment form of Group A. This procedure was repeated in three cycles and finally each group constructed six successive models, which explained three patterns identified from the moon data. The second part of the curriculum pertained to the development of strategies for teaching of the modeling ability. This last part of the course called PsTs to develop and implement, in collaboration with upper elementary students, curriculum units concerning the development of successive models of a specific physical phenomenon aiming to promote modeling-based learning.

Data collection

Three different means of data collection were used: (1) A series of *diagnostic tests*. These were administered to PsTs before and after instruction and assessed the constituent components of the modeling ability provided by Papaevripidou *et al.* (submitted) and discussed earlier in this paper. (2) The transcripts of the *synchronous discussions.* The discussions took place among PsTs once every two weeks beyond the actual class time and revolved around the critique of current educational research on the development of the modeling ability. (3) PsTs' reports regarding action research studies aiming to the development of the modeling ability of elementary students

Data analysis

Phenomenographic analysis was used to categorize the PsTs' responses derived from *diagnostic tests* (Marton, 1986). The result of a phenomenographic analysis is a set of related categories or conceptions pertaining to the phenomenon under study. Usually the categories are formed in relation to the content and the level of scientific correctness and are distinct from the rest according to qualitative criteria. PsTs' responses to the pre- and post-tests were studied and included into a hierarchical list of ideas.

For the analysis of *the synchronous discussions* and the *PsTs' projects*, content analysis was used. *Content analysis* included the use of codes which were either predetermined (a priori coding analysis) or emerged throughout the coding procedure (emergent coding analysis) (Coffey & Atkinson, 1996).

Results

Modeling Difficulties: PsTs' difficulties during model construction

The data analysed in our effort to describe specific modeling difficulties were collected prior (pre-tests), during (synchronous discussions), or after instruction (post-tests). Five specific modeling difficulties were identified. We considered that a PsT did encounter a difficulty when it was traced both prior and after instruction. This assumption was made in order to be able to compare the modeling difficulties with the pedagogical difficulties encountered by PsTs (see next section). If the difficulty was not identified when analyzing the data of the test after the instruction, we considered that the instruction is responsible for that change, and therefore the PsT does not face that difficulty after the course. Due to space limitations we will only present two difficulties.

1. Difficulty 1: PsTs believe that when constructing a model someone should know exactly how the phenomenon under study works

During a synchronous discussion, PsTs compared teaching with simulations and teaching through model construction. Some of them expressed the view that a learner builds a model when she wants to represent what she knows, while, in contrast, a learner can use a simulation in order to gain knowledge about a phenomenon:

PsT 10: *When building a model someone should be an expert of the phenomenon. By using a simulation she learns a lot about the phenomenon. This is why we should study the phenomenon of the moon phases (with the instructors' help) and then model it using the software.*

Four out of 21 PsTs encountered the idea that when constructing a model someone should know exactly how the phenomenon under study works. They also feel that external help, preferably by experts, reinforces a person's readiness to construct a model. In other words, they expect that the instructor or a book should provide the correct information to them and they capture modeling as a procedure of expressing knowledge and not a process of scientific improvement, which results to improved learning.

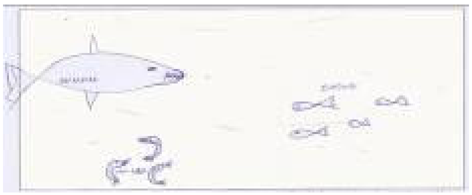
2. Difficulty 2: During model construction PsTs tend to place more emphasis on objects rather than any other element of the model

Diagnostic test 1 asked PsTs to observe a few minutes video about life in a sea ecosystem, then draw a sketch of their understanding of the phenomenon on paper and provide further explanation about it. In their effort to construct that (drawing) model they identified and included objects (e.g. shark, salmon, plants), variables (e.g. velocity of the fish, fish population), processes (reproduction, feeding) and interactions (e.g. the shark eats salmons, one fish attacks the other) in it. The frequencies of the elements of PsTs' models are presented in table 1. Both in pre- and post-test, PsTs tended to include more objects and interactions among models elements rather than variables and processes.

Table 1:
Summary of the elements included in PsTs' models (diagnostic test 1)

Model Elements	Objects	Variables	Processes	Interactions
Pre-test	72	23	47	64
Post-test	74	46	53	80

Here is a typical reaction of a PsT who constructed a model including more objects than any other form of elements (Figure 2):



Model components:
Objects: *Living organisms (i.e. shark, salmons, shrimps)*
Process: *Food chain (e.g. the shark eats the salmons and the salmons eat the shrimps)*

Figure 2. Model of the sea ecosystem by PsT 17 (pre-test response)

PsT 17 included three objects (shark, salmons and shrimps) and two object-object interactions (The shark eats the salmons and the salmons eat the shrimps) in her model. Moreover, she represented one process, nutrition. No variables were included in the model. We call this model object dominated. Other than the one process included in the model, this PsT developed a model which focuses on objects and their interactions. It warrants mentioning that the ideas of nine out of 21 PsTs fall into this category of difficulties.

3. Difficulty 3: When constructing a model, PsTs tend to include more interactions between objects rather than any other kind of interaction
4. Difficulty 4: When comparing models, PsTs tend to use superficial criteria like phenomenological characteristics or aesthetic criteria
5. Difficulty 5: PsTs fail to appreciate the comparison of a model and the phenomenon as an essential mechanism for improving the model

All five modeling difficulties are epistemological in nature. They fall under this category of difficulties as they relate to PsTs' epistemological awareness and more specifically, their inability to understand either the nature of models (e.g. difficulty 4) or the nature of the modeling process (e.g. difficulty 1).

Pedagogical difficulties: PsTs' difficulties for promoting modeling-based learning

Analysis of PsTs' final projects revealed five difficulties that are pedagogical in nature, as they pertain to PsTs' inefficiency to promote the development of the modeling ability to elementary students. Due to space limitations we will only present two difficulties.

6. Difficulty 6: PsTs seem to misunderstand or take for granted the process of deployment during the modeling procedure

PsTs developed activity sequences and instructed the development of successive models of a certain phenomenon to an elementary student. Some of the PsTs did not implement or misunderstood the model deployment phase of the modeling procedure. Despite the fact that their students created multiple successive models, they didn't prompt fruitful circumstances for the students to feel the need for improving their model and therefore move from one model to a successive one. PsT 10, for example modelled the phenomenon of photosynthesis in collaboration with her student. Among her final comments about her teaching were the following: *"…with minor changes that she (the student) performed alone, she finally constructed six successive models, which really don't look very different compared to each other. The final model is the one that the student considered as the best one to represent the phenomenon of photosynthesis."* This PsT didn't really try to involve her student in the process of deploying her model, and even if she did, she didn't really manage to succeed in it. Careful analysis of the steps undertaken by all PsTs during their modeling-based instruction indicated that nine out of 21 PsTs encountered this pedagogical difficulty.

7. Difficulty 7: PsTs tend to assume the role of expert authority, and perceive that their professional responsibility is the transmission of the "correct" scientific information and models to the students

Analysis of the final projects of the PsTs indicated that three out of 21 PsTs seemed to perceive themselves as expert authorities. They assumed that their professional responsibility pertains to the transmission of the "correct" scientific information to the students, who are asked to use them in order to build (or deploy) their models. PsT 10 first asked her student to draw a model about photosynthesis on a piece of paper and afterwards a model on the program Stagecast Creator™. They conducted an experiment pertaining to the effect of sunlight to plant development and the student was asked to assess the model according to the new information. Until that point, the process seemed to be close to the scientific one. However, instead of structuring her instruction around the experiment and its results, after conducting the experiment, she provided the student with extra material (powerpoint presentation) regarding the: *"In order to help her, I provided the powerpoint presentation. I wanted her model to be more accurate. This is the reason I gave her the correct scientific information".* It seems that the PsT did not trust the results of the experiment and considered them as not clear to the student. Instead of repeating the experiment or treating them in a way that the student would understand, she provided the "accurate and correct" information through the presentation.

8. Difficulty 8: PsTs overemphasize the role of objects in their instruction
9. Difficulty 9: PsTs overemphasize the role of the interactions between objects than any other kind of interaction in their instruction about models
10. Difficulty 10: PsTs tend to place emphasis on aesthetical improvements of their students' models

Interdependence of epistemological and pedagogical difficulties

A further analysis of the results revealed an interesting pattern described in Figure 1. We conducted a Chi Square Analysis for revealing the relationship among difficulties. Phi Coefficient indicated the correlations among them. The lines connecting different difficulties do not imply any causal relationship nor do they imply that one difficulty is causally related to the other. They rather indicate that PsTs tending to encounter the difficulty at the one edge of the line tend also to encounter the difficulty at the other edge of it. For example, it is assumed that PsTs encountering difficulty 3 (D3), construction of models with more interactions among objects rather than any other kind of interactions, tend to also encounter difficulty 9 (D9), they tend to guide their students to do the same by overemphasizing the role of the interactions between objects than any other kind of interaction.

We used the results of the statistical analysis to qualitatively describe PsTs' attempts to promote the development of the modeling ability of their students. These results suggest the existence of the following three theoretical didactical approaches regarding teaching the modeling ability:

a) *Linear modeler*s: teachers encountering a combination of D1, D6 and D7
b) Aesthetic modelers: PsTs encountering D4 and D10,
c) *Object-oriented modelers*: PsTs encountering the combinations D2-D8 and/or D3-D9

A modeler who uses the first theoretical approach for modeling (*linear modeler*) cannot conceive modeling as a cyclical procedure, which includes model construction and successive refinements of it after comparing it to the phenomenon. Instead, she considers modeling as a process with a starting and an end point, where she represents the phenomenon in the correct way. Even if she improves her model, the improvement is not a result of the comparison of the model and the data or the phenomenon. This theoretical approach to teaching also includes teachers' attempts to guide their students, who construct successive models of a specific phenomenon, to the same process. They design their instruction so as to follow a linear path where the student studies the phenomenon and construct one or more models which are not a result of comparing each model draft to the phenomenon, but rather a result of adding new information provided by the teacher or another source of information.

An *aesthetic modeler* is the one who construct models which are guided by aesthetic orientations. An *object oriented modeler* is the teacher who constructs models which include more objects or interactions among objects rather than any other model element. We consider teachers who use these two theoretical approaches to teaching as *superficial modelers*. They pay attention to surface characteristics of the phenomenon when modeling it and consider modeling as a representation process. Moreover, they ascribe, to modeler the responsibility of reproducing the obvious parts of the phenomenon and not the physical quantities or the underlying mechanism pertaining to it. Teachers who use this theoretical approach to teaching the modeling ability are more sensitive to guide their students to a process of model refinement, which is based on phenomenological features of the phenomenon rather than features that relate to physical quantities of it.

Discussion

With regard to the first research question, which relates to the modeling difficulties encountered by PsTs when constructing models, five modeling and five pedagogical difficulties were identified. Abd-El-Khalick and Lederman (2000) argue that epistemological development must be an explicit instructional goal. The modeling difficulties presented here, which are epistemological in nature, tend to dissuade and hamper the process of learning. This is the reason why such difficulties should be made explicit and where appropriate confronted in a learning environment so that conceptual understanding and acquisition of skills are achieved. Very often in teacher preparation reasoning, epistemological difficulties are not identified and therefore remain in the ecology of the learners and affect or even determine both the learning process and the subsequent teaching practice.

The second question refers to the possible interference of the epistemological and pedagogical difficulties. The results showed statistically significant relationships between these difficulties encountered by PsTs when developing models and promoting modeling-based learning and indicated the existence of three theoretical didactical approaches regarding teaching the modeling ability (linear, aesthetic and object-oriented modeler).

Linear modelers are those who present students the right answers (Van Driel & Verloop, 2002) or even demonstrate the (scientific) models as static facts (Van Driel & Verloop, 1999) instead of stimulating students to construct their own schemes or explanations and elaborate on their own ideas. A possible explanation for this tendency could lie on teachers' belief that their students encounter many difficulties and express misconceptions regarding models; and therefore are not able to participate in the modeling procedure effectively. Teaching the modeling ability in a cyclical way is not an easy task. It is easier to have the students build one model using all the "correct" or "scientific" ideas the teachers or books provide. The complexity of this procedure is also expressed by Justi and Gilbert (2002a), who state that teaching to construct models *de nevo* should be the last step of the three phases of "learning to model" framework. During this last phase students should actually work like scientists, not knowing the outcome beforehand. This phrase emphasizes the nature of the linear modeler´s teaching approach; she lacks understanding not only about how to guide students though the modeling procedure in a cyclical way, but she also presents students the "truth" when attempting to model a phenomenon. Philosophically, this teaching approach is in accord with logical positivism (Van Aalsvoort, 2004), an approach about 'ready-made science', not about 'science in the making'. A teacher whose thinking is in that line tries to bring out the rationality of scientific results. She places emphasis on the scientific results and truth and not the scientific work. This philosophical stand considers the rational necessity of elaborating a logical model that allows for the assignment of meaning to scientific concepts obtained by scientific methodology within the structure of a theoretical system (Flores, Lopez, Gallegos, & Barojas, 2000). In contrast, the constructivist orientation of teachers lies on the other edge of the spectrum. Teachers displaying a constructivist orientation indicate, for instance, that different models can co-exist for the same target, dependent on the researchers' interest or theoretical point of view or prompt their students to ground their models on the data collected, on the phenomenon itself and not on the correct answers (Van Driel & Verloop, 1999). In modeling based teaching and learning, this entails that the remaining major challenge for the teacher is to guide the students through the modeling procedure (observation of the phenomenon-data collection, identification of new relationships, and continuous improvements of the model) and not follow the logical positivism philosophy.

In an effort to provide an explanation for the presence of learning difficulties, Tiberghien (1994) elaborates on the theoretical basis of physics knowledge and states that when physicists interpret and predict experimental facts, they do not directly apply a theory to the situation but, by using the theory, they construct a model of the experimental situation. From the learner's perspective, interpretation of a phenomenon or a material situation, takes place by the construction of a "model" of the situation on behalf of her. Like physicists, the learner selects objects and elements, which are relevant according to her own point of view. Related work of the same researcher with fifth graders about heat and temperature (Tiberghien, 1980, 1985) indicated that their models are very close to the objects and events which are directly observable and perceived. The difference of students' models and physicists' models lies on inclusion of objects and events. The latter do not include objects and events as such, but use physical quantities with mathematical formalisms. The results of the present study are in accord to the findings of Tiberghien. Superficial modelers (*object-oriented or aesthetic modelers)* perceive modeling as a process of representation of the phenomenon rather than an epistemological analysis of it. They don't realize the important role of physical quantities in the modeling process and therefore don't bother including them in the models. Instead, they emphasize in the inclusion of the directly perceived objects or the aesthetical aspects of the phenomenon.

Clark, Richard, Ravit Golan, Luke, & William (2008) investigated students´ ability to provide scientific explanations during modeling and resulted in that, among others, they tent to use communicative and aesthetical criteria when evaluating scientific models. The results of diSessa (2002) reinforce these findings as he presented a coding scheme regarding students ability to judge the quality of representations, in which aesthetic criteria were included as non-scientific. Likewise, Van Driel και Verloop (1999) recorded teachers views about models and identified that some of them emphasize the physical appearance of the models. Teachers of that scale appreciate for example that "a model has the shape of a drawing" or "the most important difference between a model and the target concerns the scale". *Aesthetic modelers,* as reported in the present research study, act like the teachers of Van Driel and Verloop (1999). These teachers assign their students the responsibility to include aesthetic characteristics of the phenomenon and not the physical quantities in their models. The discussion about using aesthetic criteria for evaluating models has its roots in the philosophy of science. On the one hand, many scientists note the importance of aesthetic factors for developing theories (Fleck, 1935; Goodman, 1981; Kuhn, 1962; McAllister, 1989, 1990; Wechsler, 1978; Welsch, 1997; Zee, 1986). On the other hand, other scientists oppose to the use of aesthetic criteria in theory choice (Engler, 1990; Lakatos & Musgrave, 1978; Maxwell, 1998).Whether we find something beautiful or ugly must depend, to some extent at least, on our personal, subjective, emotional responses to that thing.

Interpretation of the results that derived from the present research effort can be supported by both perspectives of the philosophy of science. We consider that aesthetic modelers do not necessarily think unscientifically. They might represent, for example, successfully the underlying mechanism of the phenomenon under study in the constructed models and at the same time focus on including an aesthetic feel. This does not detract from the scientific process of model construction. If we guide learners not to think aesthetically, it does not follow that they will think (more or less) scientifically nor that they are more likely to promote appropriate criteria. On the other hand, if teachers rely exclusively on aesthetic criteria for theory choice, they do deviate from the scientific process for construction and evaluation of models.

All teachers studied in the frame of the present research constructed knowledge through model building and deployment and tried to transform their knowledge into teaching by guiding their students to also construct knowledge through developing successive models of a phenomenon. We showed that this construction process is obstructed by certain learning and teaching difficulties, which should be confronted in the learning environment.

References

Abd-El-Khalick, F., & Lederman, G. N. (2000). Improving Science Teachers' Conceptions of Nature of Science: A Critical Review of the Literature. *International Journal of Science Education, 22*(7), 665-701.

Clark, C. A., Richard, D. A., Ravit Golan, D., Luke, B., & William, P. J. (2008). A Microgenetic Classroom Study of Learning to Reason Scientifically through Modeling and

Argumentation, *Proceedings of the 8th International Conference for the Learning Sciences - Volume 3*. Utrecht, The Netherlands: International Society of the Learning Sciences.
Coffey, A., & Atkinson, P. (1996). *Making Sense of Qualitative Data. Complementary Research Strategies*. London: Sage Publications.
Coll, R. K., France, B., & Taylor, I. (2005). The role of models/and analogies in science education: implications from research. *International Journal of Science Education, 27*(2), 183–198.
Crawford, B., & Cullin, M. (2004). Supporting Prospective Teachers' Conceptions of Modelling in Science. *International Journal of Science Education, 26*(11), 1379-1401.
diSessa, A. A. (2002). Students' Criteria for Representational Adequacy. In K. Gravemeijer, R. Lehrer, B. van Oers & L. Verschaffel (Eds.), *Symbolizing, Modeling and Tool Use in Mathematics Education*. Dortrecht: Kluwer.
Duschl, R. A., Schweingruber, H. A., & Shouse, A. W. (2007). *Taking Science to School: Learning and Teaching Science in Grades K-8*. Washington D.C.: National Academy Press.
Engler, G. (1990). Aesthetics in Science and in Art. *The British Journal of Aesthetics, 30*, 24-31.
Fleck, L. (1935). *Genesis and Development of a Scientific Fact* (F. Bradley & T. J. Trenn, Trans.). Chicago: University of Chicago Press.
Flores, F., Lopez, A., Gallegos, L., & Barojas, J. (2000). Transforming Science and Learning Concepts of Physics Teachers. *International Journal of Science Education, 22*(2), 197-208.
Gibson, H. L., & Chase, C. (2002). Longitudinal Impact of an Inquiry-based Science Program on Middle School Students' Attitudes toward Science. *Science Education, 86*(5), 693-705.
Gilbert, S. (1991). Model Building and a Definition of Science. *Journal of Research in Science Teaching, 28*(1), 73-79.
Goodman, N. (1981). Ways of Worldmaking. *Leonardo, 14*(4), 351-351.
Gott, R., & Duggan, S. (1995). *Investigative Work in the Science Curriculum*. Buckingham: Open University Press.
Gott, R., & Duggan, S. (1996). Practical Work: its Role in the Understanding of Evidence in Science. *International Journal of Science Education, 18* (7), 791-806.
Halloun, I. (1998). Schematic Concepts for Schematic Models of the Real World: The Newtonian Concept of Force. *Science Education, 82*(2), 239-263.
Halloun, I. (2007). Mediated Modeling in Science Education. *Science and Education, 16*, 653–697.
Hammer, D. (1996). More than Misconceptions: Multiple Perspectives on Student Knowledge and Reasoning, and an Appropriate Role for Education Research. *American Journal of Physics, 64*(10), 1316-1325.
Harrison, A. G. (2001, 25–28 March). *Models and PCK: Their Relevance for Practicing and Pre-service Teachers.* Paper presented at the National Association of Research in Science Teaching, St. Louis, MO.
Hestenes, D. (1987). Towards a Modelling Theory of Physics Instruction. *American Journal of Physics, 55*(5), 440-454.
Justi, R., & Gilbert, J. (2002a). Modelling, Teachers' Views on the Nature of Modelling, and Implications for the Education of Modellers. *International Journal of Science Education, 24*(4), 369-387.
Justi, R., & Gilbert, J. (2002b). Science Teachers' Knowledge about and Attitudes towards the Use of Models and Modelling in Learning Science. *International Journal of Science Education, 24*(12), 1273–1292.
Justi, R., & van Driel, J. H. (2005). The Development of Science Teachers' Knowledge on Models and Modelling: Promoting, Characterizing, and Understanding the Process. *International Journal of Science Education, 27*(5), 549–573.

Kafai, Y. B., & Resnick, M. (1996). *Constructionism in Practice: Designing, Thinking and Learning in a Digital World (1-8)*. New Jersey: Lawrence Erlbaum Associates.

Karplus, R. (1980). Teaching for the Development of Reasoning. *Research in Science Education, 10*, 1-10.

Kuhn, T. S. (1962). *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press.

Lakatos, I., & Musgrave, A. (1978). *Criticism and the Growth of Knowledge: Proceedings of the International Colloquium in the Philosophy of Science, London, 1965*: Cambridge University Press.

Marton, F. (1986). Phenomenography-A Research Approach to Investigating Different Understandings of Reality. *Journal of Thought, 21*, 29-49.

Maxwell, N. (1998). *The Comprehensibility of the Universe: a New Conception of Science*: Oxford University Press, USA.

McAllister, J. W. (1989). Truth and Beauty in Scientific Reason. *Synthese, 78*(1), 25.

McAllister, J. W. (1990). Dirac and the Aesthetic Evaluation of Theories. *Methodology and Science*(23), 87-102.

NRC. (1996). *National Science Education Standards*. Washington, DC: National Academy.

Papaevripidou, M., Constantinou, C. P., & Zacharia, Z. (submitted). Enacting a Framework for Developing and Assessing the Modeling Ability in the Elementary Science Classroom. *Journal of the Learning Sciences*.

Papert, S., & Harel, I. (1991). Situating Constructionism. In S. Papert & I. Harel (Eds.), *Constructionism*: Ablex Publishing Corporation.

Stylianidou, F., Boohan, R., & Ogborn, J. (2005). Science Teachers' Transformations of the Use of Computer Modeling in the Classroom: Using Research to Inform Training. *Science education, 89*(1), 56-70.

Tiberghien, A. (1980). Modes and Conditions of Learning. An Example: The Learning of some Aspects of the Concept of Heat. In W. F. Archenhold, R. Driver, A. Orton & C. Wood-Robinson (Eds.), *Cognitive Development Research in Science and Mathematics, Proceedings of an International Seminar* (pp. 288-309). Leeds: University of Leeds.

Tiberghien, A. (1985). Heat and Temperature, the Development of Ideas with Teaching. In R. Driver, E. Guesne & A. Tiberghien (Eds.), *Children's Ideas in Science*. Milton Keynes, England: Open University Press.

Tiberghien, A. (1994). Modeling as a Basis for Analyzing Teaching-Learning Situations. *Learning and Instruction, 4*(1), 71-87.

Van Aalsvoort, J. (2004). Logical Positivism as a Tool to Analyse the Problem of Chemistry's Lack of Relevance in Secondary School Chemical Education. *International Journal of Science Education, 26*(9), 1151-1168.

Van Driel, J. H., & Verloop, N. (1999). Teachers' Knowledge of Models and Modelling in Science. *International Journal of Science Education, 21*(11), 1141-1153.

Van Driel, J. H., & Verloop, N. (2002). Experienced Teachers' Knowledge of Teaching and Learning of Models and Modelling in Science Education. *International Journal of Science Education, 24*(12), 1255–1272.

Wechsler, J. (1978). *On Aesthetics in Science*. Cambridge, Massachusetts: MIT Press.

Wellington, J. (1994). *Secondary Science*. London: Routledge.

Welsch, W. (1997). *Undoing Aesthetics*: Sage Publications Ltd.

Zee, A. (1986). *Fearful Symmetry: The Search for Beauty in Modem Physics*. New York: Macmillan.

Linkages, Languages: connecting traditional art and digital technologies

Maria Cosatto, *dalincosatto@wanadoo.fr* *http://www.maria-cosatto.com*
Independent visual artist

Abstract

This paper proposes a case study elucidating a creative process in the arts that well illustrates a constructionist approach. It describes the design and the implementation of IMAGO TRANSITUS, a video-light installation created during a residency early in 2008, in the Chapel St Eloi of the Domaine de Soucy, Fontenay-les-Briis, France.

As an artist, I'm making sense of space, time, light and materials with my artworks. I develop my own models exploring and using a variety of tools: materials, lights, virtual images, visual language, computer languages and software, silence, sounds, body movements…

In this particular case, the artwork was made of a wadding-light arrangement associated to a video projection. Digital contributes to the emergency of dynamic qualities of the material work while the later offers to the digital image a new medium of expression and existence.

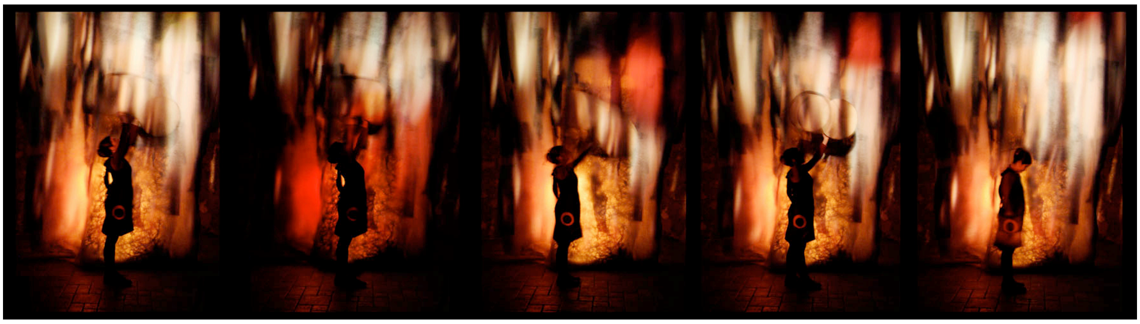


Figure 1. IMAGO TRANSITUS, five moments of the masterpiece. Performer: Irina Zhekova

The finalisation of the artwork can not be considered as the last step of the process. The exposition of the installation in a public space indeed induces a new dialogue with other artists and visitors that could lead me in return to modify the models, the techniques or the tools I use.

“ Linkages, Languages ” is the motto in which I build my artistic creativity as well as my own reality.

Keywords

Traditional art, digital technologies, linkage, language, creative process, interaction.

Describing the creative process of Imago Transitus

My artistic purpose is the creation of evolving illuminated art works. The marriage wadding-light becomes the material drive and the space to comprehend is conceived to be a place of actual experience.

Mindful of the interaction of the materials under the influence of lights, I built my work and its field of vision, using composite techniques including drawing, sculpture, photography, video, scenography. New technologies and the concept of reuse have increased in importance in my latest artworks.

The wadding, comprising fullness and emptiness, is the privileged area for a fluid, permeable and breathing gateway between exterior and interior. The digital technologies offer the movement to light who magnifies the materials as well as magnifying itself. Thereby the couple wadding-light enhances the visual and tactile qualities of the evolving masterpiece and questions the notion of borders and identity.

This paper describes the design and the implementation of IMAGO TRANSITUS, a video-light installation created during a residency early in 2008, at the Domaine de Soucy, France.

The creative process follows a constructionist strategy. I present a serial of mental and manual operations made during the experience to illustrate how intelligence (intuitive, logic, practice…) and environment (a shadow of a plant in sunset, a library of “trial and trash”, a software…) interact following the initial purpose set for this project: express visually the notion of flowing.

The artwork is conceived as a complex, open and dynamic system, the digital techniques as a tool and also as an exploration territory [4].

The language of visual art combines with the digital language, making sense. Due to its fluid nature, digital contributes to the emergency of dynamic qualities of the material work while the later offers to the digital image a new medium of expression and existence.

“ Linkages, Languages ” is the motto in which I build my artistic creativity as well as my own reality [6,7,9].

The context and the purpose

The place to invest is a former twelfth century chapel located in the park of the domain.

The visit of the chapel took place in December 2006. According to the proportions of the space and due to its size, I decided to work on verticality. The apse hosts the artwork. The medium artwork was made of a wadding-light arrangement.

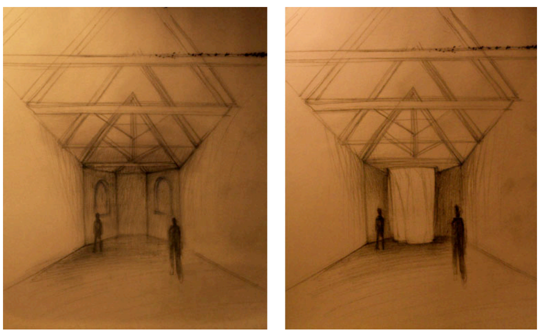


Figure 2. Sketch: interior space of the Chapel St Eloi (XIII century)

For this project, I was able to identify quickly my aesthetic choice clearly stating my intention: I wanted to express the notion of flowing.

To this aim, I used a video-projection. The first reflex was asking me which was the suitable image to build my video, but the answer did not reach immediately. The design and the implementation of the video followed a particular path.

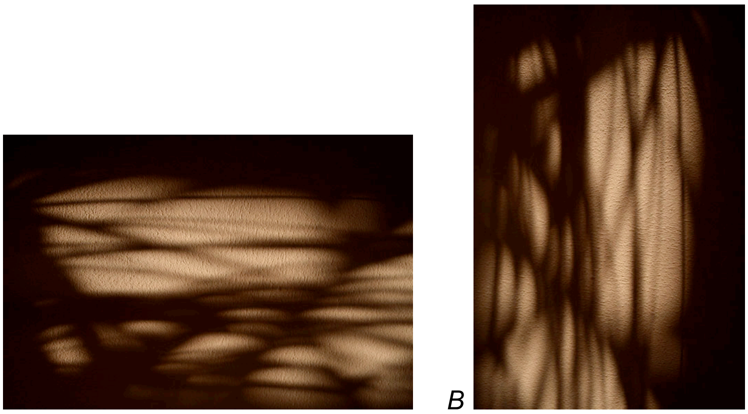
Figure 3. Wadding-light arrangement in the space of the apse. Stage of implementation. January 2008

Seeking the singular image

While I was writing the blurb of IMAGO TRANSITUS project at sunset, I have spotted the beautiful images of the houseplant leaves shadows placed near from my desk. I took a serial of photos I put in my picture library. I recognized them later as the images I was looking for, those which resonated with my purpose.

Freeze Frame and Insight

A particular photo (Figure 4A) drew my attention. The eye just stopped on it, whereas time seemed to do the same, it stopped. My perception was increasing. The presence of this image was of a rare density, almost hypnotic, so strong that it made me forget its context, as if there was no bottom [5]. What did it want to tell me? How this image was meaningful for me? Subconsciously, I knew that this representation appeared containing information and an expressive potential. How to decrypt it?



A *B*

Figure 4. The shadow of leaves plants at sunset. Digital image

The play, its stakes and the Interpretation

Playing with the graphics editing program I made a simply but decisive operation: the rotation of the image (Figure 4B). I suddenly took conscious of the meaningful of this image and recognized its capacity to express the notion of flowing, it was vertical and streaming.

Our intention, our intimate project determines our interpretation.

The image processing

In order to elaborate the video, the image of Figure 4B was transformed using a graphics editing program.

I wanted to build the video that once projected was going beyond the physical limits of the work. Thereby, it met the requirements concerning the proportions of the physical medium of the work as well as the physical constraints of the exhibition place without forgetting my artistic purpose.

The resulting image is presented in Figure 5.

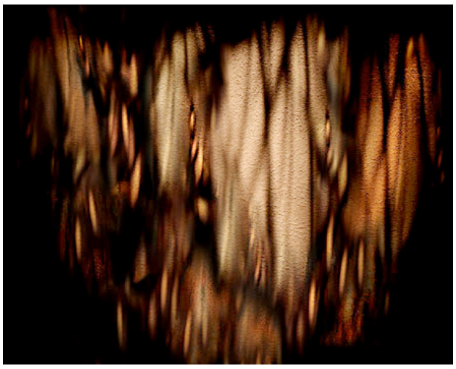


Figure 5. Changing the image of the figure 4B using a graphics editing program

The digital image has an evolving capacity through the fluidity of numbers and language being unstable, mobile, changing. But it is also inalterable, endlessly duplicable, transmitted without loss and stable through the pixels, the last components of the decomposition digital image [2].

Virtual images have a so real exploration potential!

Seeking movement

I used the morphing tool to create the motion between two identical images searching to erase the mechanical aspect of the movement. Morphing transforms gradually one image into another by computer processing.

Both departure and arrival points are the same. The movement is generated from the position of the key points. Obtaining a fluid motion necessitated a "trial and error" process.

From this moment I entered the domain of video.

The resulting motion picture represented properly the notion of flowing I was seeking for.

Figure 6. Morphing: From X to X

An evolving intention, a richest meaning

Watching the video again and again, a crucial need to add an event of short duration appeared: it would offset the endless fall. For instance, something that goes up, like a bubble coming out of a thick liquid.

My library of “trial and trash” provided me the image more than adequate. It was a detail in the photo-souvenir of a fleeting creation: a red bubble (Figure 7).

Figure 7. Red bubble in a photo of my Library of “trial and trash”. Fleeting creation.

This bubble intertwined with Figure 5 to obtain three new images (Figure 8). Then, from these new images, I built other motions and finally, a video richer in meaning.

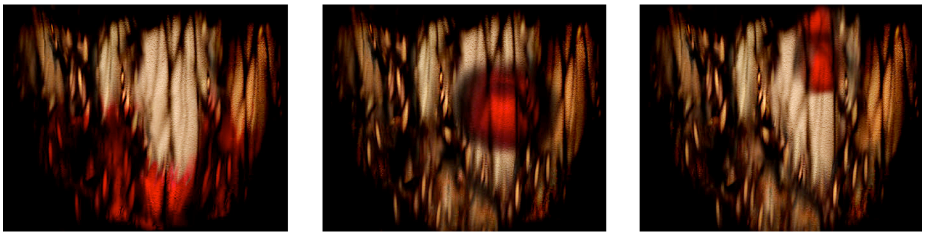


Figure 8. Three images enriching the video

The Video is ready

The intervention associated with this paper would present the motion picture in its entirety. The duration of this animation is about one minute.

The development of morphing and videos has been based on visual and tactile results obtained on the medium artwork. This one has been amended several times to meet the video needs. They are intimately linked.

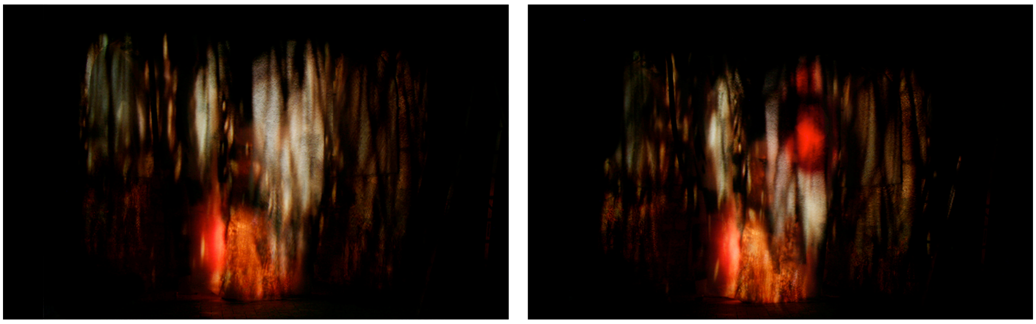


Figure 9. Imago Transitus, two installation views

Opening Imago Transitus [3,8]

This artwork was not standing alone. It was associated with other artistic projects.

The masterpiece enhanced its potentialities by the sensibility of the sound design proposed by Charlie Dalin. The motion pictures and the sound had a different looping period. This contributed to the renewal of perceptions and meaning.

Irina Zhekova injected the dimension of performing art. Her “bendir” was the magic magnifying glass of the Opening day.

Figure 10. Imago Transitus. Performer: Irina Zhekova

Concluding remarks

This contribution proposes a case study elucidating a creative process that well illustrates a constructionist approach.

As an artist, I’m making sense of space, time, light and materials with my artworks. I develop my own models exploring and using a variety of tools: materials, virtual images, visual language, computer languages and software, silence, sounds, body movements…

In the specific artwork presented here, IMAGO TRANSITUS emerges from the dialogue between a tool-matrix, a concrete arrangement wadding-light, and a tool-machine that processes a virtual image to transform it into an animated projected light.

When the installation is exposed in a public space, it’s time for me to hear visitors’ remarks and reactions. That allows me to see the impact of the artistic entity I created. The changes I could make are also highlighted. I can then envisage modifying the models, the techniques and the tools I use. This is a nourishing dialogue, an exchange that falls within the “Public Entity” mentioned by Seymour Papert.

This approach is inevitably reductive, as it is very difficult to reach and moreover to explain “l’infracassable noyau de nuit” [1] mentioned by André Breton.

Acknowledgments

I would like to thank James Clayson for encouraging me to present my work, for his great availability and his councils. I would also thank Sylvie Gillot for her listening and her precious help. I’m glad of our fruitful exchanges.

References

1. Breton, André *Préfaces "Les Contes bizarres" d'Achim Von Arnim.*

   "THE UNBREAKABLE CORE OF NIGHT". I chose this metaphor to say that creation is intelligible but irreducible: it always remains a part of intuition that can not be reabsorbed by logic.

   The word " infracassable " doesn't exist in French. It is composed by juxtaposing and overlapping prefix and words presenting common syllables: IN -un- / INFRA -infra- / FRACASS(ER) -to shatter- / CASSABLE -breakable-.

2. Couchot, Edmond http://www.ciren.org/artifice/artifices_1/couchot.html

3. IMAGO TRANSITUS http://www.maria-cosatto.com

   Residency: January - February 2008

   Exhibition: 15th mars -13th April 2008 Chapel St Eloi, Domaine de Soucy, Essonne, France.

   Sound Designer: Charlie Dalin www.myspace.com/charliedalin

   Performance: Irina Zhekova, 14 mars 2008, Opening day. www.myspace.com/iremusicspace

4. Lestorcat, Louis-José (2008) *Entendre l'esthétique dans ses complexités*. L'Harmattan.

5. Merleau Ponty, Maurice (1945) *Phénoménologie de la perception.* Éditions Gallimard.

6. Morin, Edgar http://www.emse.fr/aslc2009/pdf/Logique et contradiction E MORIN.pdf

7. Morin, Edgar *Pour une reforme de la pensée* http://college-heraclite.ifrance.com/documents/r *actuels/em reforme.htm*

8. Schmitt, Jean-Claude (1996) *La Culture de l'imago-Images médiévales.* CNRS Centre National de la Recherche Scientifique. Institut de l'information scientifique et technique.

9. Von Glasersfeld, Ernst (1988) *Introduction à un constructivisme radical. L'invention de la réalité- Contributions au constructivisme- sous la direction de Paul Watzlawick*. Éditions du Seuil. pp. 19 − 43.

Playful Turtle Geometry in the Paradise

Károly Farkas, *farkas.karoly@nik.uni-obuda.hu*
Dept of Software Technology, Óbuda University

László Csink *csink.laszlo@nik.uni-obuda.hu*
Dept of Software Technology, Óbuda University

Ildikó Tasnádi, *tasiildi@gmail.com*
Nemes Nagy Ágnes Secondary School of Humanities

Abstract

We present a turtle microworld that we have been developing and using for several years in a grammar school. We think that the example is important from a didactic viewpoint. The presentation will be done parallel on two different monitors in two Logo dialects; thereby comparisons are possible between the Imagine and MicroWorlds Ex dialects.

The innovative part of our paper is the generation of various polygons, spiral curves and rosettes with the same Logo procedure, just by changing the parameters.

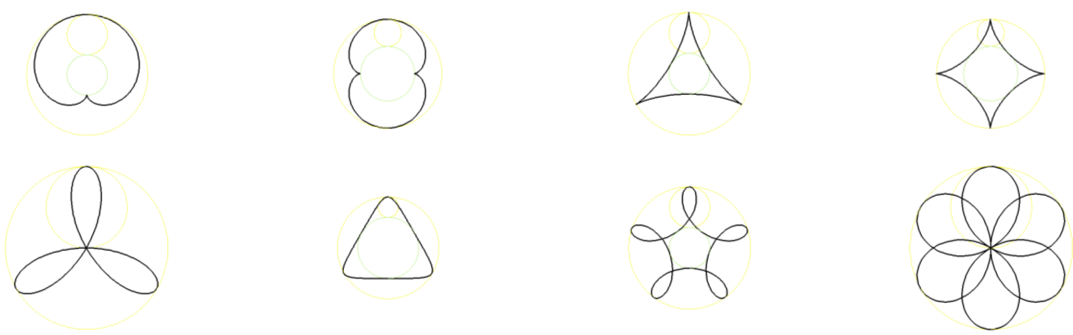


Figure 1. Cardioid, nephroid, deltoid, asteroid, rosettes, produced with the same procedure

The generation of these curves may invoke aesthetical pleasure and is an example how to develop mathematical and computational thinking. Understanding and remembering is more effective when it is linked to primeval stimuli. Turtle geometry is effective due to it's visualise and the experience of motion. Body syntonic significantly helps in understanding the algorithms which can be executed and lived through by the students as well. We think that some special curves are friendlier and easier to grasp if they are superposed by simple turtle motions showing curve ontology.

It is an important didactic advantage that the parameters of the curve generation function and the command parameters can be directly linked to various properties of the moving turtles; thereby the curve transformations can be easily realised and better understood using syntonicity [*]. Function analysis is thus more playful and enjoyable.

In our Paradise we move three people. Adam is performing some motion Eve is doing the same or some other motion independently from Adam, while Cain performs both motions. For superposition we use two ways, either methods attached to the turtles, or an infinite 'super' method.

Keywords

Superposition, playfulness, system, microworlds, genesis

ΠΑΙΖΟΝ ΜΗ ΒΙΑ ΔΙΔΑΣΚΕ ΤΟΥΣ ΠΑΙΔΑΣ! (Πλάτων)
Teach the children with play! (Plato)

'Take my hand. I'm a stranger in Paradise.' (Stranger in Paradise… Borodin - Tony Bennett)
http://www.tsrocks.com/t/tony_bennett_texts/stranger_in_paradise.html

1. Pedagogical Issues

The generation of curves may not only invoke aesthetical pleasure and but it is an example how to develop mathematical and computational thinking, how to use the computer effectively in teaching mathematics and physics, as well as computing. When studying mathematics, many students lose the thread when trigonometric functions and curves are discussed. Students are unable to link these curves to some reality. It helps if many curves are constructed at mathematics classes, but this is time consuming and not enjoyable for most students. Using a computer makes it easier to present many examples, also to keep the attention of students, and the visualisation is more dynamic as well.

When the various curves are produced by moving and animating turtles; the inherent knowledge can be discovered. Understanding and remembering is more effective when it is linked to primeval stimuli.

Turtle geometry is effective due to its visualisation and the experience of motion. Body syntonic – the possibility of linking the steps of drawing to the motion of our own body – significantly helps in understanding the algorithms.

Colleagues taking part at this conference well know the polygons and the Papert-type algorithms of drawing a circle. These algorithms can be easily executed and lived through by students too. It is generally agreed that the other mathematical curves can also be more easily understood and remembered using turtle geometry. We claim that more special curves that are not so easy to generate by traditional methods are also more understandable and friendly if the curve ontology is presented.

We generate the curves in an intrinsic way, from the point of view of the turtle in a relative coordinate system, thereby by attracting attention to the essential features of the curve.

It is an important didactic advantage that the parameters of the curve generation function and the command parameters can be directly linked to various properties of the moving turtles; thereby the curve transformations can be easily realised and better understood using syntonicity. Function analysis is thus more playful and enjoyable.

2. The Stage of Micro World

We move three main and later several minor actors in our Paradise. Let us create Adam, Eve and Cain:

```
newturtle "Adam setc 116 st pd            new "Turtle [name Adam pencolor 9]
newturtle "Eva setc 16 st pd              new "Turtle [name Eva pencolor 12]
newturtle "Cain st pd setpensize 3        new "Turtle [name Cain penwidth 3]
```

Adam performs some motion, Eve does the same independently from Adam, and Cain mimics both parents and performs both motions. Typically for turtle geometry, motions are put together from basic steps and turns, and Cain mixes these two kinds of elements. If these elements are infinitesimal, the resulting curve will be continuous. As we shall see, the sufficiently small element is often one turtle step or turn, or even a fraction of these.

We use two techniques for superposition.

- Superposition using the methods attached to the turtles

```
Set "Adam "onclick [forever [run :a]]        Adam'setEvent "onclick [run :a]
Set "Eva "onclick [forever [run :b]]         Eva'setEvent "onclick [run :b]
Set "Cain "onclick[forever[run :a run :b]]   Cain'setEvent "onclick [run :a run :b]
```

If, for example, the value of :a is fd 1 and :b is rt 1, then Cain steps ahead a little and turns, thereby drawing a circle.

```
make "a [fd 1] make "b [rt 1] everyone [clickon]
(Imagine version, see in the end of attachment.)
```

This is a novel realisation of the Papert-type circle drawing algorithm. The superposition has been successful.

In our next example we put Cain and Eve beside Adam, they all stand on the x-axis looking towards north. Then, if :a and :b are both equal to fd 1 , let us have a look at Cain's action:

```
make "a [fd 1] make "b [fd 1] everyone [clickon]
```

The result of two uniform rectilinear motions is also a uniform rectilinear motion whose speed is the vectorial sum of the speeds of the original motions. Cain, however**, does not do this in any Logo dialects**! He moves slower than expected. We think that the reason of this fault is that we use a Neumann-type computer having one processor only, so the three turtles do not work in parallel, and so Cain misses some steps. If we are not content with the type and genesis of the resulting curve but we are also interested in its relationship with its components, we need to find another way to represent superposition.

- Infinitesimal 'super' procedure

```
to sup :a :b
ask [Cain Adam][run :a]
ask [Cain Eva] [run :b]
sup :a :b
end
```

Running this procedure the turtles will execute our commands in a scheduled and exact way. For example,

```
sup [fd 1] [fd 1]
```

results in Cain's path being twice longer than that of his parents. When running **sup** we have to bear in mind that Cain acts twice in each sup loop, so some usual turtle methods have to be adapted. To draw the **inspi** curves, e.g., the following command can be used

```
make "fi 0 sup [fd 5] [rt :fi make "fi :fi + 1 / 2]
```

In the following we use both the above methods. To demonstrate the composite motions we normally attach methods to the turtles (animate Adam and Eve). When we wish to construct the exact resulting curve we use the **sup** procedure.

3. One of the Components is a Carrier Motion in a Straight Line

Adam's task is to increase the x coordinate in a uniform way: make "a [setx xcor + 1], while Eve moves in various ways.

3.1. Eve rotates

`make "a [setx xcor + 1] make "b [rt 1] everyone [clickon]` or

`sup [setx xcor + 1] [rt 1]` The result a sommersaulting turtle.

3.2. Eva performs an alternating motion

Let us create the alternating method:

```
to alter
fd 1 if (ask "Cain [abs ycor]) > 90 [rt 180]
end
sup [setx xcor + 1] [alter]
```

The result is the sawtooth wave.

3.3. Eve rotates with increasing speed

```
make "alfa 0 sup [fd 8] [rt :alfa make "alfa :alfa + d / 2]
```

We got an inward spiral (curve of increasing curvature) where the angle is incremented each time. Changing the starting angle :alfa and the **d** difference value we get the well-known series of curves.

alfa = 0 , **d** is 1, 2, 3, … 7

alfa=1 , **d** is 1, 2, 3, … 7

*Figure 2. The elements of the first and second row of the **inspi** curve matrix*

3.4. Eve circulates

```
sup [setx xcor + 1] [fd 1 rt 1]
```

The result is a cycloid. Changing Adam's (or Eve's) step length we get various cycloids.

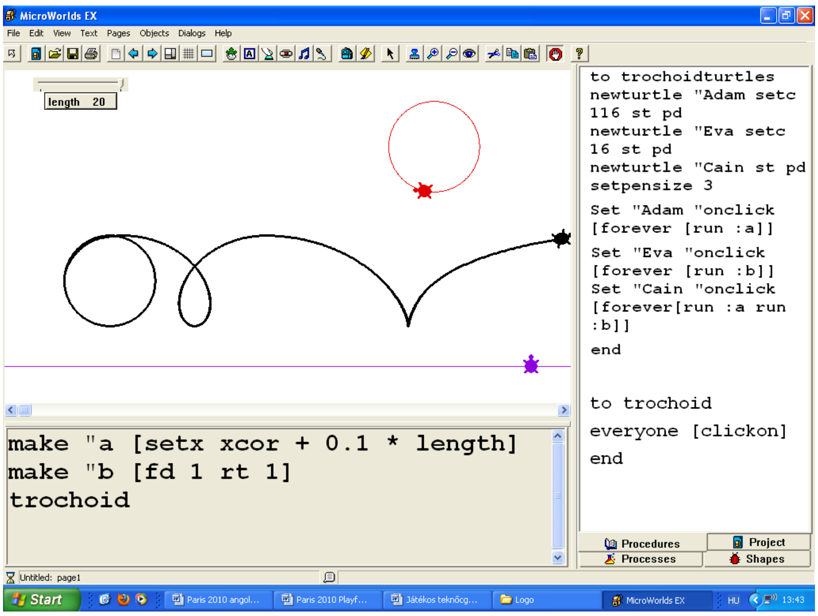


Figure 3. The cycloid is being transformed by extending Adam's step length

3.5. Eve performs harmonic oscillation

The projection of circular motion to any line parallel with a diameter yields harmonic oscillation. This is the syntonic interpretation as we teach it to our students.

Let us introduce a new turtle, Lucifer:

```
newturtle "Lucifer setc 56 st pd
sety ask "Lucifer [ycor]
```

Our procedure is the following:

```
Sup [setx xcor + 1 Lucifer, fd 1.57 rt .5] [harmonic "Lucifer]
```

```
to harmonic :Mrx
;Mrx circles
sety ask :Mrx [ycor]
end
```

(Lucifer's step size is π/2 as he moves twice in each sup loop.)

The resulting sine curve is

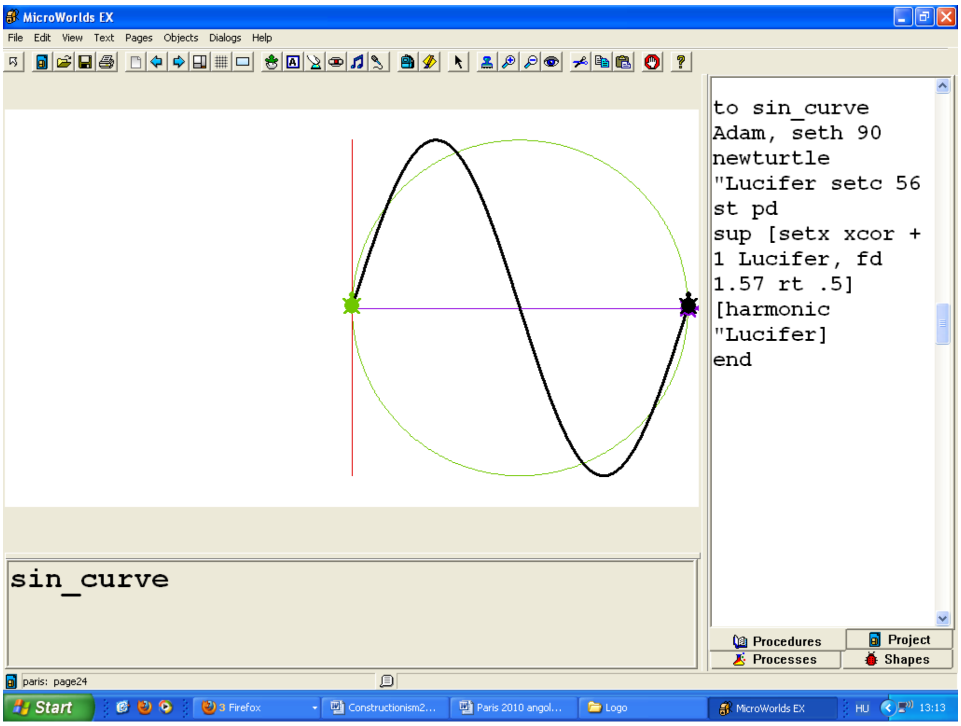


Figure 4. Generation of sine using ***sup***

The pair Eve-Cain was necessary only to explain the generation of harmonic motion. The model can be simplified if only Eve is rotating and Cain is taking the harmonic projection of this circular movement.

```
to sine
sup [setx xcor + 1 Eva, fd 1.57 rt .5] [harmonic "Eva]
end
```

By changing a parameter (e.g. with a slide) we can extend or shrink the sine curve.

3.6. The cosine curve

One way to generate the cosine curve is to use the procedure harmonicyx whose values display the projection of the circular movement to the x-axis, as the cosine function is the projection of the moving line to the x-axis in the unit circle. If we wish to draw the main cycle, let us move Eve ahead with a quarter-circle from the starting point.

```
to cos
Eva, repeat 90 [fd 3.14 rt 1]
sup [setx xcor + 2 Eva, fd 3.14 rt 1] [harmonicyx "Eva]
end
to harmonicyx :Mrx
sety ask :Mrx [xcor]
end
```

3.7. The tangent

It was hard to find a realistic pattern in nature for the demonstration of the tangent. However, we found an example in stage lighting. Let us make the light of a profile spot pass up repeatedly to a canvas and drag the canvas across. The highlight will show pieces of the tangent curve.

To the rectilinear carrier motion we have to superpose a vertical motion created by a light scanning up on the screen. The light source will be played by Lucifer (his name means literally 'light-bearer'), the moving point on the canvas will be the Snake. Lucifer keeps performing flips. The Snake is running back and forth on the canvas, linked to Lucifer (Lucifer is rotating him). The Snake faces Lucifer while Lucifer's light is rising; while they look to the same direction when Lucifer's light moves downwards.

The methods used can be found in the attachment.

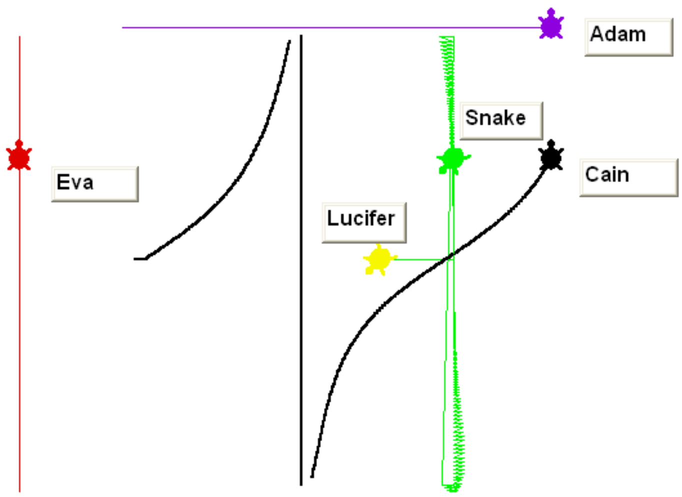


Figure .5 Generation of tangent in the Paradise

4. Adam Performs Rectilinear Accelerated Motion

```
make "a initialA make "b initialB sup [fd :a make "a :a + deltaA] [rt :b make
"b :b + deltaB]
```

When Eve rotates at constant speed (constant4 is zero), we get the well known **polispi** curves.

When Eve gains speed while rotating, we get various spectacular spirals, such as

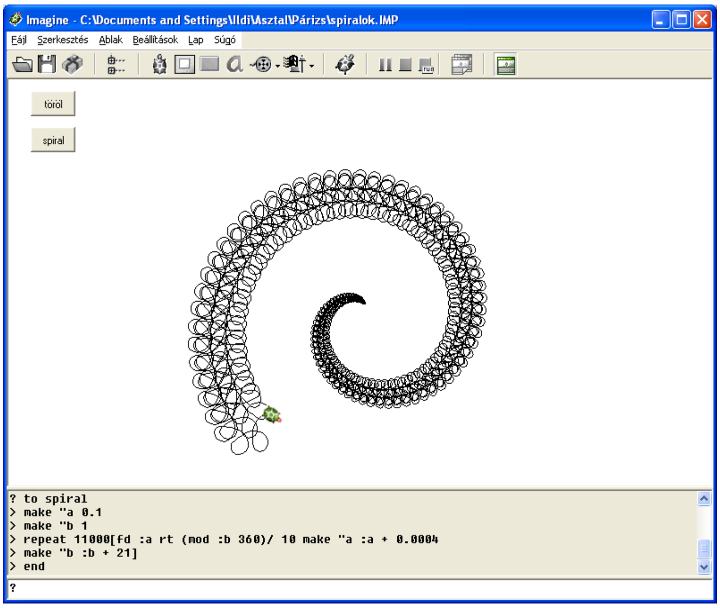


Figure 6. A 'real' spiral

5. Adam is Rotating

If Eve is also rotating, Cain is rotating at the speed of Eve and Adam added together. If Eve performs rectilinear motion, we get the curves as before, with Eve and Adam changing their roles.

6. Adam Rotates Eve, Eve Rotates Cain

This way Cain draws various curves resembling planet motions. When Adam's and Eve's direction of rotation is the same, the curves are epicycloids (see the first two curves in Figure 7). When Adam and Eve rotate in the contrary direction, we get hypocycloids. Using various parameters of the same procedure we generate the cardioids, the nephroids, the deltoids, the astroids and the rosettes, as well as their looped and extended versions. Adam, Eve and Cain may represent the Sun, the Earth and the Moon, respectively. The curves are the Moon's orbits in absolute coordinates.

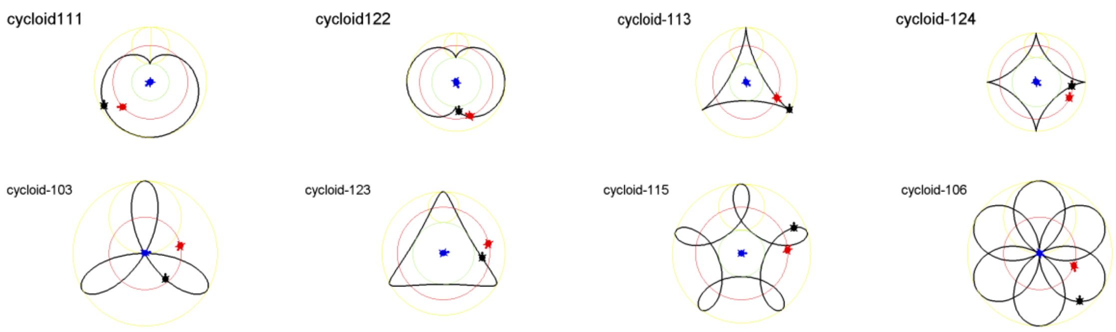


Figure 7. Some basic 'motives' of the listed curves

```
to cycloid :e :i :z
everyone [pu seth 0 setx 0] Eva,
sety 80
ifelse :e = 1 [Cain, pu sety 80 * :i
/ (1 + :i) pd Eva, seth 180] [Cain,
pu sety 80 + 80 / (:i + 1) pd]
make "d2 80 / (:i + 1)
roll
end

to roll
Adam, rt :e
make "alfa1 heading
Eva, setpos list 80 * sin :alfa1 80
* cos :alfa1 seth heading + :e
rt :z
make "alfa2 heading make "x2 xcor
make "y2 ycor
Cain, setpos list :x2 + :d2 * sin
:alfa2 :y2 + :d2 * cos :alfa2
roll
end
```

```
to cycloid :e :i :z
 ask all [pu seth 0 setxcor 0]
 ask "Eva [setycor 80]
 ifelse :e = 1 [ask "Cain [pu
   setycor 80 * :i / (1 + :i) pd]
   Eva'seth 180]
   [ask "Cain [pu setycor 80 + 80 /
   (1 + :i) pd]]
   make "d2 80 / (:i + 1)
   roll
  end

to roll
  Adam'rt :e
  make "alfa1 Adam'heading
   ask "Eva [setpos list 80 * sin
   :alfa1 80 * cos :alfa1
   seth heading + :e rt :z]
  make "alfa2 Eva'heading
  make "x2 Eva'xcor
  make "y2 Eva'ycor
  ask "Cain [setpos list :x2 + :d2 *
  sin :alfa2 :y2 + :d2 * cos :alfa2]
  roll
end
```

7. Both Components are Harmonic Oscillations

```
sup [harmonic "Lylith][harmonicxx "Lucifer]
```

The result is the Lissajous curve. To generate the various Lissajous curves in a spectacular way we use our method technique. To draw the resulting curve we construct motion elements as composites.

```
to Lissajous
Lucifer, fd 1 rt 1
Lylith, fd 1.5 * parameter / 10  rt 1 * parameter / 10
Adam, harmonic "Lylith
Eva, harmonicxx "Lucifer
Cain, setpos list ask "Lylith [xcor] ask "Lucifer [ycor]
Lissajous
end
```

Lucifer and Lylith keep rotating. The parameter changes the ratio of the speeds. harmonicxx is the projection of circular motion on the x-axis.

Figure 8. Some Lissajous curves

8. Summary of Curves

:a	:b	Resulting curve	Note
fd i	fd i	line	Vectorial sum of paths
fd i	rt i	circle	
increase xcor	rt i	somersault	
increase xcor	fd i rt i	cycloid	
increase xcor	alternating	sawtooth	
increase xcor	harmonic	sine	
Increase xcor	tangent value	tg	Tg is the ycor value of highlight
fd k	rt k	polygon	
fd k	rt inc k	inspi	
fd increase k	rt k	polispi	
fd increase k	rt inc k	spiral	
rt 1	rt 1	rotating	
rotate ”Eva	rotate ”Cain	rosetta	Adam rt 1 : epicykloid Adam lt 1 : hipocykloid
harmonic	harmonic	Lissajous	Lucifer and Lylith circle

i = motion element of infinitesimal size
k = constant

Summary

To generate curves with superposition, we need only relatively simple mathematical operations. Thus we can teach new chapters of geometry using turtle geometry in earlier age and with more effectiveness. An increasing set of curves can be systematised in a playful microworld, thus aiding the development of system approach.

Our work can be regarded as a continuation of research of, among others, Uzi Armon, Sergei Supronov and Izabella Foltynowicz. Our goal is to draw various curves playfully, to keep up motivation in informatics classes, and to demonstrate research to our students.

References

Abelson, H. and diSessa, A. (1986) *Turtle Geometry. The Computer as a Medium for Exploring Mathematics.* The MIT Press.

Alexandrov, K. and Soprunov, S. (1997) *One Approach to Mechanic Simulations in Logo*. In: M. Turcsányi-Szabó (ed.) Proceedings Sixth European Logo Conference, Budapest 114-121

Armon, U. (1997) *An Algorithm that Translates Intrinsic Equations of Curves into Intrinsic Procedures of These Curves*. In: M. Turcsányi-Szabó (ed.) Proceedings Sixth European Logo Conference, Budapest, August. pp. 378-389

Csink, L. and Farkas, K. (2008) *Turtle's Curves*. In: R. T. Mittermeyer M. M. Sysło (ed): Proceedings Informatics Education Contributing Across the Curriculum, Torun, Poland, 76-86

Farkas, K. (2003) Logo and native language. *Intrinsic procedures of some curves*. In: Secundino Correia (ed.): Proceedings of the 9th European Logo Conference, Porto pp. 69-79

Foltynowicz, I. (2007) *Cicloids and limacons in the turtle graphics*. In: Kalaš, I. (ed.) Proceddings of the 11th European Logo Conference, EuroLogo, Bratislava, Slovakia, August. p. 50.

Pólya, Gy. (1954) How to Solve It; Induction and Analogy in Mathematics. Princeton University Press, Princeton; and (1969) Patterns of Plausible Inference Princeton, N. J. Princeton.

http://www.freeweb.hu/mwlogo/Paris2010.html

Attachment

```
to tangent
;for Microworld Ex
Lucifer, st pd setx 0 seth 90 set "Lucifer "onclick [forever [lt 0,025]]
Snake, st setpos [0 0]
Adam, pu setpos [-200 180]
Eva, pu setx -280 seth 0
Cain, st pu setx -300
make "a [setx xcor + 0,05]make "b [sety ask "Snake [ycor]]
ask [Adam Eva Cain Lucifer][clickon]
end
to tg
make "k ask "Cain [ycor]
make "l1 (ask "Lucifer [heading]) < 165 make "l2 (ask "Lucifer [heading]) > 15
make "ji and :l1 :l2
make "l3 (ask "Lucifer [heading]) < 345 make "l4 (ask "Lucifer [heading]) >
195
make "bi and :l3 :l4
if :ji [ifelse :k < -170 [Cain, pu][Cain, pd] Snake, stick_to "Lucifer setx 57
]
ifelse not :k < 169  [Cain, pu make "a [setx xcor + 0,01]] [make "a [setx xcor
+ 0,05]]
if :bi [ifelse :k < -169 [Cain, pu][Cain, pd] Snake, stick_to2 "Lucifer setx
57]
tg
end

to stick_to :a
  towards :a  make "d distance :a
  make "alfa ask :a [heading] make "x ask :a [xcor]  make "y ask :a [ycor]
  setpos list :x + :d * sin :alfa :y + :d * cos :alfa
end

to stick_to2 :a
  towards :a make "d distance :a
  make "alfa ask :a [heading] make "x ask :a [xcor]  make "y ask :a [ycor]
  setpos list :x - :d * sin :alfa :y - :d * cos :alfa
end
```

The Microworlds ' Everyone [clickon] ' equal in Imagine:

```
Adam'forever [runEvent "onClick]
Eva'forever [runEvent "onClick]
Cain'forever [runEvent "onClick]
```

[*] We use 'syntonicity' expression after Papert. Mindstorms, page 63.

Algebra and Computer Algebra: Implications for High School Mathematics Examples from *The CME Project*

Al Cuoco, *acuoco@edc.org*
Center for Mathematics Education, EDC, Newton MA USA

Abstract

This paper builds on more than a decade of work at Education Development Center on the use of computer algebra with high school teachers and students. Widespread CAS use is still in its infancy in US precollege education. Its acceptance into the high school curriculum has been hampered, both by its prohibition on several high-stakes exams (the American College Testing (ACT) exam, for example) and by worries among many high school and university faculty that its use will diminish students' technical fluency with algebraic calculations.[1] The situation in the EU seems to be quite different, and I hope that I can learn at this conference how this technology is being put to use and how it has gained acceptance in European high schools.

But in a real sense, the CAS and the technology, while essential tools in what follows, are not the foci of this paper or of the conference. One of the goals of *Constructionism 2010* is to encourage "learners to better understand the world and their place in it by building their own meaning-making models based on iterative, interactive exploration and testing of ideas and notions." I want to focus on using this process as it applies to one corner of the world of mathematics.

In this paper, I'll look at several examples of how CAS environments can be used to model algebraic systems and objects. The advantages of computer algebra over other programming languages come from the fact that, in CAS environments, algebraic expressions are first-class objects. Because formal expressions are (in a sense that can be made precise) universal objects for building algebraic structures, models and experiments built in these media realize two of the goals of the constructionist approach that go back to the early days of Logo: such models and experiments are both are *general-purpose* and *extensible*.

Equipped with working computational models of algebraic systems, many high school students and teachers can gain first-hand experience with the ideas that led naturally to modern abstract algebra, providing one more example of what Richard Noss states in his open letter: "...that the Logo vision could catalyse a transformation—not just of the *ways* that people learn, or of the *methods* by which they are taught—but of *what* it becomes possible to teach and learn."

Most of my examples are taken from *CME Project*, a high school curriculum, funded by the National Science Foundation, and published by Pearson in 2009 [1]. Details about the program are at `www.edc.org/cmeproject`. The features of *CME Project* that are relevant to this paper are

1. The program is organized around mathematical habits of mind [3].
2. It makes essential use of a CAS in the last two years.

Keywords

Computer algebra system, algebraic thinking

[1] Indeed, in the US, Texas Instruments has had to produce two versions of its new handheld—one with a CAS and a lobotomized version without computer algebra.

Introduction

One[2] can build meaning-making models of phenomena in a whole host of media; I watch my grandson create models of all kinds of things with his Lego blocks, and I see him create his imaginary worlds, full of interesting characters, with his Wii. But what about mathematics? One can certainly use mathematics to model physical phenomena, but how can one model the phenomena of mathematics itself? Many years ago, my colleague Paul Goldenberg and I put it this way:

> *But mathematical objects are objects of the imagination, and many . . . don't have physical models. How, then, can people tinker with these systems or the mathematical objects of which they're built? They may not have physical models, but they* **do** *have computational models. Algebraic structures, functions, continuously varying systems, and combinatorial enumerations can all be modeled in computational environments. When students build computational models of mathematical structures—whether these are programming models in languages like Logo, ISETL, or Mathematica, templates for a spreadsheet, or constructions in tools like* Geometer's Sketchpad *or* Cabri*—they are reviewing, expressing, and getting a chance to examine the own ideas about these mathematical structures. At one level, they are getting the benefit that generally comes from writing out one's ideas carefully and in detail: that process, by itself, helps one organize one's thinking, and externalize it enough to review and examine it. Without computational technology, students had to be satisfied with their written notes. The students who could bring these notes to life entirely in their heads would have more success than those for whom the notes just sat motionless on the paper. But when the "notes" are executable on a computer, students can run the models they've made, verify their correctness or debug them, and even use them as parts of more complex models. Students who are not yet skilled enough to hold many parts of a model in their heads can build the parts one by one, show how they go together and, for the present, leave the orchestration to the computer. In short, computers can help students tinker with the physics of mathematics.* [8]

The habits of mind approach

About 40 years ago, early in my high school teaching career, I came to understand that the real utility of mathematics for many students comes from the kind of thinking that is indigenous to the discipline. In [2], I put it this way:

> *I didn't always feel this way about mathematics. When I started teaching high school, I thought that mathematics was an ever-growing body of knowledge. Algebra was about equations, geometry was about space, arithmetic was about numbers; every branch of mathematics was about some particular mathematical objects. Gradually, I began to realize that what my students (some of them, anyway) were really taking away from my classes was a* style of work *that manifested itself between the lines in our discussions about triangles and polynomials and sample spaces. I began to see my discipline not only as a collection of results and conjectures, but also as a collection of* habits of mind.
>
> *This focus on mathematical ways of thinking has been the emphasis in my classes and curriculum writing ever since, and I'm now convinced that, more than any specific*

[2] I'd like to thank Wally Feurzeig, both for his help with this paper and for all that he's done for mathematics education over the years.

> *result or skill, more than the Pythagorean theorem or the fundamental theorem of algebra, these mathematical habits of mind are the most important things students can take away from their mathematics education. For* all *students, whether they eventually build houses, run businesses, use spreadsheets, or prove theorems, the real utility of mathematics is not that you can use it to figure the slope of a wheelchair ramp, but that it provides you with the intellectual schemata necessary to make sense of a world in which the products of mathematical thinking are increasingly pervasive in almost every walk of life.*

When I first came to EDC in the early 1990s, my colleagues and I made a careful analysis of these mathematical habits of mind (see [3], for example), and we began developing high school courses and curricula organized around this analysis. *CME Project* is a direct descendent of that early work and the decades of classroom experience that preceded it; the evolution is described in more detail in [5].

By "mathematical habits of mind," I mean the mental habits that mathematicians use, often unconsciously, in their mathematical work. There are general mathematical habits—performing thought experiments, for example—and habits that are central to specific branches of mathematics. In analysis, for example, one often employs reasoning by continuity or passing to the limit. There are also important *algebraic* habits of mind that are the focus of the algebra courses in *CME Project*. These include:

- Seeking regularity in repeated calculations.
- "Chunking" (changing variables in order to hide complexity).
- Reasoning about and picturing calculations and operations.
- Purposefully transforming and interpreting expressions to reveal hidden meaning.
- Seeking and modeling structural similarities in algebraic systems.

Developing these and related algebraic habits is a pervasive goal throughout the program. So, for example, *CME Project* develops an approach to solving classical algebra word problems, not because of any intrinsic value in these problems and their stylized contexts, but because this class of problems, and the approach students use to solve them, provides an arena for developing the extremely useful habit of finding regularity in repeated calculations and forming processes from isolated computations.

Our choices of technologies and how we use them is also dictated by this goal of developing specific mathematical habits. For example, dynamic geometry environments can be used to help students learn to reason by continuity and to look for invariants under continuous transformations.

Computer algebra systems are ideal media for helping students develop algebraic habits like the ones described above. And access to a CAS gives students much more than computational power and the ability to perform complicated calculations.

Using a CAS to build algebraic habits of mind

Modern CAS environments contain a great deal more than the ability to treat algebraic expressions as first-class objects (that is, objects that can be named and that can be inputs to and outputs from functions). The TI-Nspire technology, for example, has graphics-handling capabilities (including equation graphing and dynamic geometry), a spreadsheet, a functional programming language, and a CAS, and all of these environments talk to each other. We make use of all of these capabilities in *CME Project*, but I want to focus here on the value-added that comes from computer algebra: the ability to use these packages with formal algebraic expressions.

Our group at EDC sees three overlapping uses for computer algebra that help students develop algebraic habits: CAS media can be used as

an algebra laboratory. CAS technology can be used to experiment with algebraic expressions in the same way that calculators can be used to experiment with numbers: generating data, making patterns apparent, and giving students the raw data from which they can generate conjectures. They provide teachers and students with general purpose tools for finding regularity in data, or for imposing regularity when no simple patterns can be found. CAS technology also has the potential to bring a renewed and modern emphasis on formal algebra—that is, the algebra of forms—to school mathematics (see [6] and [7] for more on this theme).

an algebraic calculator. CAS technology can be used to make tractable and to enhance many beautiful classical topics, historically considered too technical for high school students. This is the use of technology that reduces computational overhead and that allows students to easily perform calculations that would be impossible (or overly distracting) without the technology. It is also the use that surrounds one of the biggest worries of many teachers in the US: If the computer can perform the calculations, what is the value of teaching paper-and-pencil algebraic skills?

a modeling tool for algebraic structures. This is the use that's of most importance to constructionism. CAS technology allows students to build models of algebraic objects and systems that have no faithful physical counterparts. This use of technology adheres to our view that building a computational model for a mathematical structure helps one build the mental constructions needed to interiorize that structure [8, 10]. Furthermore, such computational models are *executable*, so that students can build working models of mathematical systems, turning the mathematician's thought experiments into actual experiments. As we said on page 2, what CAS environments add to other modeling environments is the facility to perform generic calculations with algebraic *expressions*—polynomials, rational functions, and formal power series. Hence these environments provide a medium for expressing abstract algebraic structure.

Of all the computational available environments, the TI-Nspire system is best suited for our purposes for several reasons:

1. It is first and foremost an educational tool, so that great care has gone into the design of its interface and its conventions. For example, it uses notation that is faithful to common mathematical notation—what you write on the blackboard is essentially what you type into the system.
2. It is available on a handheld device, so that students can use the system in or out of class.
3. The various other environments (dynamic geometry, functional programming, and spreadsheet, for example) are also designed for education, and the various environments interact. So, for example, a function defined in the programming environment can be tabulated in the spreadsheet.

Examples: A case study of $x^n - 1$

In this section, I'll look at each of the CAS uses described above—experimenting, calculating, and modeling—pointing out how they encourage the development of algebraic habits.

The context for these examples is the set of polynomials of the form $x^n - 1$, where n is a positive integer. These polynomials are ubiquitous in almost every branch of mathematics. From a high school curriculum perspective, they can be used to tie together many core results from algebra, geometry, and trigonometry. My goal in these examples is to show how CAS models of the mathematical objects help reify the objects in the minds of people who build the models. Twenty minutes after bringing the next example into a classroom or a workshop, there's no question about the fact that everyone feels that they are dealing with real objects.

Experimenting: Finding factors of $x^n - 1$

Most first-year algebra books contain the factorization

$$x^2 - 1 = (x-1)(x+1)$$

Sometime in high school, students may also see

$$\begin{aligned} x^3 - 1 &= (x-1)(x^2+x+1) \\ x^4 - 1 &= (x-1)(x+1)(x^2+1) \\ x^6 - 1 &= (x-1)(x+1)(x^2+x+1)(x^2-x+1) \end{aligned}$$

So, over the integers $\mathbb{Z}$, $x^2 - 1$ and $x^3 - 1$ each have two factors, $x^4 - 1$ has three, and $x^6 - 1$ has four. Is there any pattern to the number of factors as a function of n? That is, can we find any regularity in this table?

n	number of factors of $x^n - 1$
1	1
2	2
3	2
4	3
5	
6	4
7	
8	
9	

A CAS allows one to experiment with this question, generating data from which one can draw conclusions. For example, you can define a function that factors the polynomials:

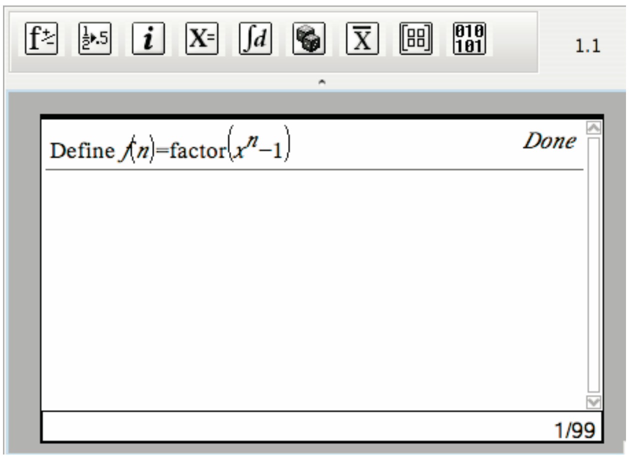
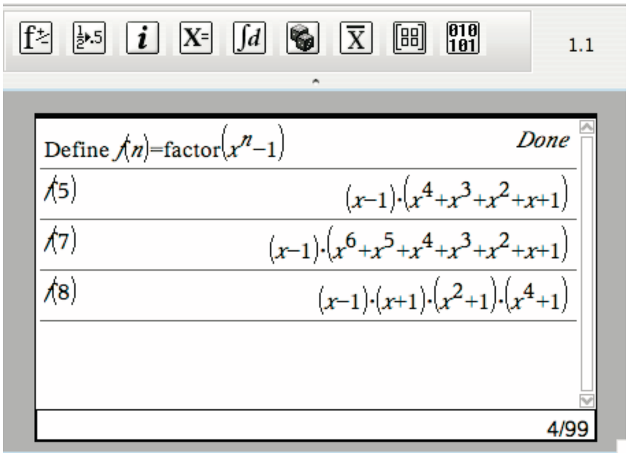


Figure 1: $f(n)$ factors $x^n - 1$ over $\mathbb{Z}$

The experiment might proceed as follows



At this point, two conjectures often emerge:

1. There are always at least two factors:

$$x^n - 1 = (x-1)(x^{n-1} + x^{n-2} + \cdots + x^2 + x + 1)$$

2. If n is odd, there are exactly two factors.

The first conjecture is true; the factor theorem from algebra 2 shows that $x-1$ must be a factor of $x^n - 1$ for any n, because 1 is a root of the equation $x^n - 1 = 0$. In *CME Project*, we ask students to explain why the right-hand side multiplies out to $x^n - 1$ *without carrying out any explicit calculations*, picturing how the the calculation would go if they did multiply everything out.

Conjecture 2 is false, as a little more experimenting shows:

n	number of factors of $x^n - 1$
1	1
2	2
3	2
4	2
5	2
6	4
7	2
8	4
9	3

When we've used this activity with students and teachers, several conjectures emerge:

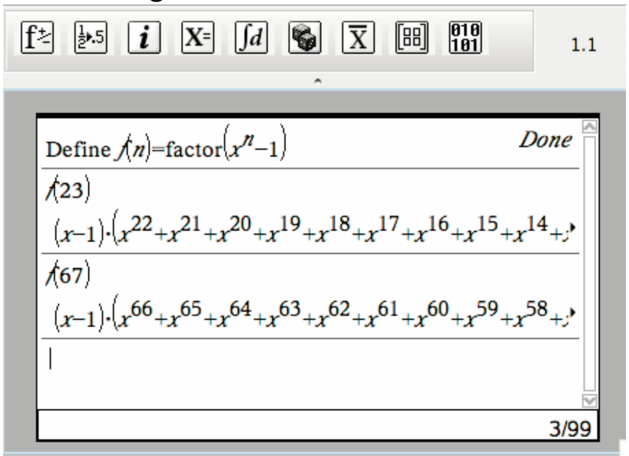
- If n is *prime*, there are exactly two factors.
- If n is the square of a prime, there are three factors ($x^9 - 1$, for example).
- If n is the product of two distinct primes, there are four factors ($x^{15} - 1$, for example).

In classroom discussions or in student work, these statements usually coalesce into a single conjecture:

> **Conjecture**: *The number of irreducible factors of $x^n - 1$ over $\mathbb{Z}$ is the number of positive integer factors of n.*

Here we have a conjecture for a non-obvious (and non-trivial) pattern in a sequence of polynomials. When I've used this activity with students and teachers, the question takes on a life of its own, and the laboratory environment afforded by the CAS helps establish the claim I made on page 4: the objects of the investigation (the polynomials) become *real objects*. Some other points about this investigation:

- The CAS can be used to check conjectures for large values of n, adding to the sense that one is working with real "things:"



- By looking at the actual factorizations produced by the CAS, rather than simply the number of factors, one can develop and prove more refined results. Indeed, the CAS can be used to inspire results about the factorizations of certain subsets of our sequence:

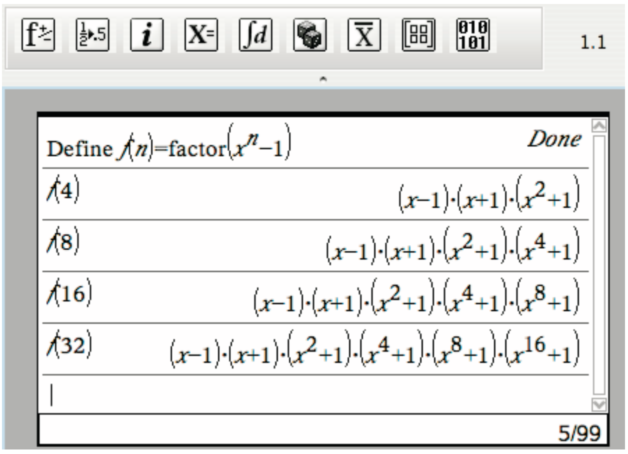


Figure 2: $f(2^k)$ as a special case.

- CAS use makes progress on a conjecture tractable for almost all second-year algebra students, and may of them will leave it at that. Others may take things a bit further and show why $\frac{x^n-1}{x-1}$ is irreducible if n is prime.

This is a good example of a low-threshold, high-ceiling activity. And the mathematics behind all this is central to many parts of algebra and analysis—it gets deep enough to challenge even the most advanced students. For example, if $\psi_k(x)$ is the polynomial whose roots are precisely the primitive kth roots of unity, then

$$x^n - 1 = \prod_{d|n} \psi_d(x) \qquad (*)$$

Here, the product is over all divisors of n. It can be shown (although the standard proof is quite hard in places) that each $\psi_k(x)$ is defined and irreducible over $\mathbb{Z}$, explaining why the conjecture on page 6 is, in fact, true. And equation $(*)$ can be used in a CAS to compute each $\psi_k(x)$ recursively.

More refined conjectures emerge from further experimentation. Whenever I use this activity with students or teachers, someone always asks if the coefficients of the $\psi_k(x)$ are always in the set

$$\{0, \pm 1\}$$

One can use a CAS to investigate this question. The first instance of a coefficient different from $0, \pm 1$ is in ψ_{105}. In fact, the coefficients of ψ_n can be made as large as one pleases [9]. There's much more to say about this example, but the point here is that we are now dealing with genuine models of real phenomena, with all the textured features of intricate physical systems.

Reducing overhead: The Polynomial Factor Game

The *Connected Mathematics Project* [11] introduces middle school students (ages 11–13) to primes and the prime factorization of integers via the *Factor Game*. This is a game for two players, played on a board like this:

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30

The rules of the game are up for negotiation in a class, but one version goes like this:

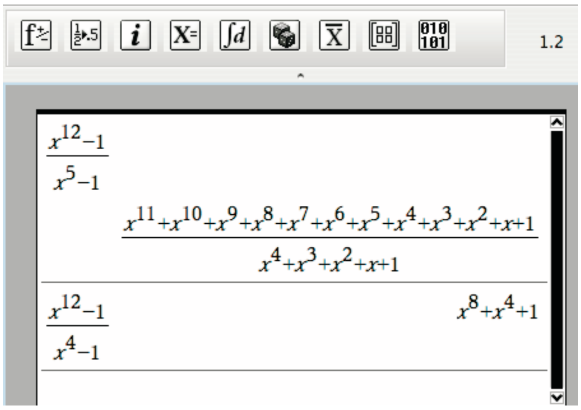
1. Player A picks a number n from the board, getting that many points, and the number is crossed off.
2. Player B gets the sum of all the numbers not crossed off on the board that are factors of n, and crosses them off.
3. B goes next, picking an available number and gets that value.
4. A gets the sum of the non-crossed off numbers that are factors of m.
5. If either player picks a number with no factors left on the board, he or she loses a turn and gets no points.
6. The game continues until there are no possible moves.

CME Project contains a game with the same rules, except the board looks like this:

$x-1$	x^2-1	x^3-1	x^4-1	x^5-1
x^6-1	x^7-1	x^8-1	x^9-1	$x^{10}-1$
$x^{11}-1$	$x^{12}-1$	$x^{13}-1$	$x^{14}-1$	$x^{15}-1$
$x^{16}-1$	$x^{17}-1$	$x^{18}-1$	$x^{19}-1$	$x^{20}-1$
$x^{21}-1$	$x^{22}-1$	$x^{23}-1$	$x^{24}-1$	$x^{25}-1$
$x^{26}-1$	$x^{27}-1$	$x^{28}-1$	$x^{29}-1$	$x^{30}-1$

The points that a player wins on a round correspond to the degrees of the polynomials that are picked.

The CAS is used here simply as an algebraic calculator. If a player wants to see if one of these polynomials divides another, he or she can simply check to see if the quotient is a polynomial.



It doesn't take long before students begin to see that this game "is the same as the middle school factor game." That is, a conjecture emerges

> **Conjecture**: $x^m - 1$ is a factor of $x^n - 1 \Leftrightarrow m$ is a factor of n

One direction of this implication is a nice application of the "chunking" habit: To see, for example, that $x^3 - 1$ is a factor of $x^4 - 1$, you can argue like this:

$$\begin{aligned}
x^{12} - 1 &= \left(x^3\right)^4 - 1 \\
&= (\clubsuit)^4 - 1 \\
&= (\clubsuit - 1)\left(\clubsuit^3 + \clubsuit^2 + \clubsuit + 1\right) \quad \text{(see the identity on page 6)} \\
&= \left(x^3 - 1\right)\left((x^3)^3 + (x^3)^2 + (x^3) + 1\right) \\
&= \left(x^3 - 1\right)\left(x^9 + x^6 + x^3 + 1\right)
\end{aligned}$$

The other direction of the implication (if $x^m - 1$ is a factor of $x^n - 1$, m is a factor of n) is much harder. One way to think about it it requires some facility with De Moivre's theorem and with roots of unity. Another approach (shown to me byVince Matsko) is to use the arithmetic structure of the ring of polynomials in one variable over the real numbers, a structure with many of the same features as the ring of ordinary integers. Briefly, it goes like this:

> Suppose that $x^m - 1$ is a factor of $x^n - 1$. Write $n = mq + r$ with $0 \le r < m$. Then
>
> $$x^{n-r} - 1 = x^{qm} - 1$$
>
> But $x^m - 1$ is a factor of the right-hand side of this equation (chunking, again), so it divides both $x^n - 1$ and $x^{n-r} - 1$, and hence divides their difference:
>
> $$x^{n-r}(x^r - 1)$$
>
> But $x^m - 1$ is relatively prime to x^{n-r}, so it must be a factor of $x^r - 1$. Since $r < m$, this implies that $r = 0$.

Modeling: Roots of unity

If you watch high school students calculate with complex numbers, many will act as if they are calculating with polynomials in i, with the additional simplification rule "$i^2 = -1$." There is a germ of an important idea here: students are noticing the structural similarities between $\mathbb{C}$ and

$\mathbb{R}[x]$—the two systems seem to "calculate the same." This is a good example of the universal nature of formal algebraic expressions mentioned on page 1: The complex numbers can be realized as a "quotient" of $\mathbb{R}[x]$ by the relation $x^2 + 1 = 0$ (see [4] for more on this theme). And in fact this construction, first articulated in this way by Kronecker, is perfectly general: every algebraic extension of a field K can be modeled as $K[x]$ with some extra relations.

This seeking structural similarities in algebraic systems is an important algebraic habit of mind, and it gets exercised when calculations in one system start to feel like calculations in another. But before the advent of CAS, I would have never thought of introducing it to any but the most advanced precollege students. Now it becomes, without all the trappings of abstract algebra, tractable to a wider set of students and teachers.

For example, many precalculus courses (including *CME Project*) contain a treatment of De Moivre's theorem, often stated like this:

$$(\cos\theta + i\sin\theta)^n = \cos n\theta + i\sin n\theta$$

De Moivre's Theorem implies several facts relevant to our family $x^n - 1$:

- The roots of $x^n - 1 = 0$ are

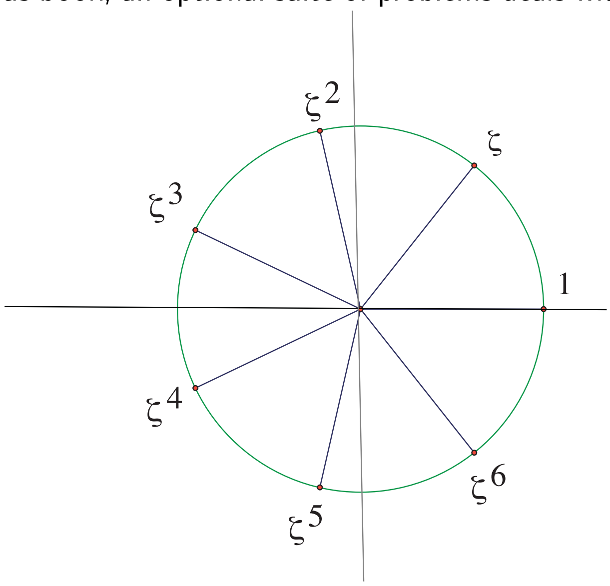
$$\left\{\cos\frac{2k\pi}{n} + i\sin\frac{2k\pi}{n} \quad | \quad 0 \le k < n\right\}$$

- If $\zeta = \cos\frac{2\pi}{n} + i\sin\frac{2\pi}{n}$, these roots are

$$1, \zeta, \zeta^2, \zeta^3, \ldots, \zeta^{n-1}$$

- These roots lie on the vertices of a regular n-gon of radius 1 in the complex plane.

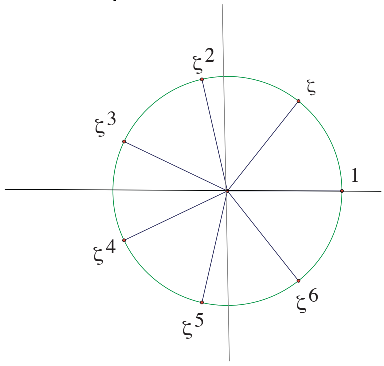
In *CME Project* precalculus book, an optional suite of problems deals with the 7th roots of unity:



Notice that

- The six non-real roots come in conjugate pairs.
- So, $(\zeta + \zeta^6)$, $(\zeta^2 + \zeta^5)$, and $(\zeta^3 + \zeta^4)$ are real numbers.
- Hence these three numbers *satisfy a cubic equation over* $\mathbb{R}$.

The object of the activity is to find this equation.



Let

$$\alpha = \zeta + \zeta^6$$
$$\beta = \zeta^2 + \zeta^5$$
$$\gamma = \zeta^3 + \zeta^4$$

To find an equation satisfied by α, β, and γ, we need to find

- $\alpha + \beta + \gamma$
- $\alpha\beta + \alpha\gamma + \beta\gamma$
- $\alpha\beta\gamma$

We find these one at a time...
The Sum:

Since $\alpha = \zeta + \zeta^6$, $\beta = \zeta^2 + \zeta^5$, and $\gamma = \zeta^3 + \zeta^4$, we have

$$\alpha + \beta + \gamma = \zeta^6 + \zeta^5 + \zeta^4 + \zeta^3 + \zeta^2 + \zeta$$

But

$$x^7 - 1 = (x - 1)(x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)$$

So,

$$\zeta^6 + \zeta^5 + \zeta^4 + \zeta^3 + \zeta^2 + \zeta = -1$$

The Product:

$$\alpha\beta\gamma = \left(\zeta + \zeta^6\right)\left(\zeta^2 + \zeta^5\right)\left(\zeta^3 + \zeta^4\right)$$

Notice that the right-hand side "feels like" a call to do a formal calculation. Indeed, we can get the *form* of the expansion by expanding

$$\left(x + x^6\right)\left(x^2 + x^5\right)\left(x^3 + x^4\right)$$

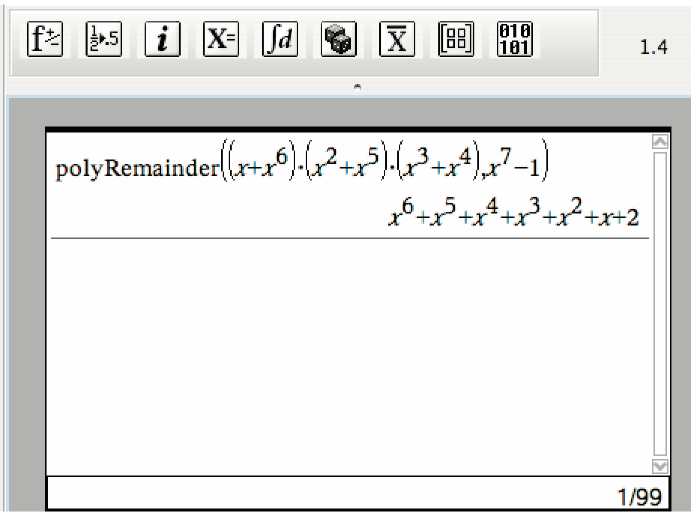
A CAS tells us that

$$\left(x + x^6\right)\left(x^2 + x^5\right)\left(x^3 + x^4\right) =$$
$$x^{15} + x^{14} + x^{12} + x^{11} + x^{10} + x^9 + x^7 + x^6$$

But if we replace x by ζ, we can replace x^7 by 1. So, if the above expression is divided by $x^7 - 1$ and written as

$$(x^7 - 1)q(x) + r(x),$$

then replacing x by ζ will produce $r(\zeta)$. A CAS can be used to do the calculation:



Since

$$\zeta^6 + \zeta^5 + \zeta^4 + \zeta^3 + \zeta^2 + \zeta + 1 = 0$$

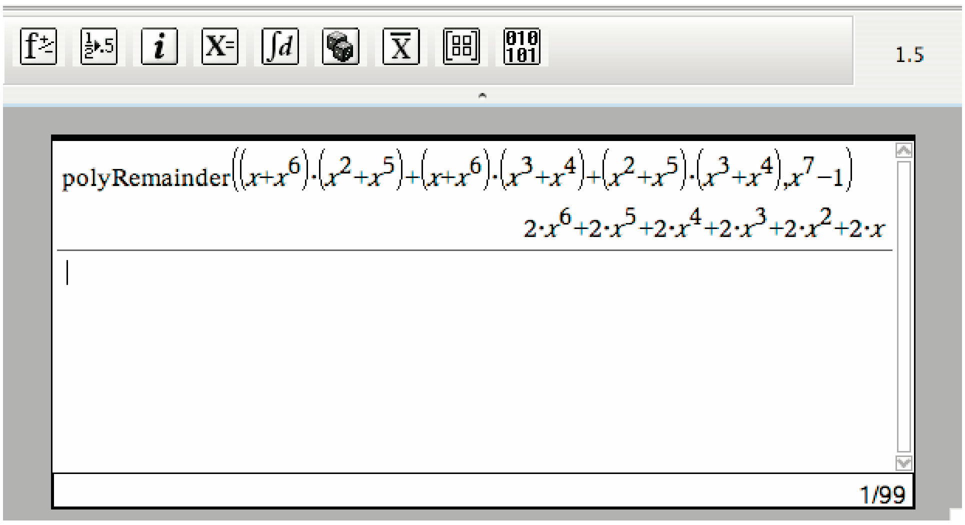
We get

$$\alpha\beta\gamma = 1$$

The sum, two at a time: Well, $\alpha\beta + \alpha\gamma + \beta\gamma =$

$$\begin{gathered}\left(\zeta + \zeta^6\right)\left(\zeta^2 + \zeta^5\right) + \\ \left(\zeta + \zeta^6\right)\left(\zeta^3 + \zeta^4\right) + \\ \left(\zeta^2 + \zeta^5\right)\left(\zeta^3 + \zeta^4\right)\end{gathered}$$

We can use a CAS, thinking of this as a formal calculation, reducing by $x^7 - 1$:



It follows that $\alpha\beta + \alpha\gamma + \beta\gamma = -2$, and our cubic is

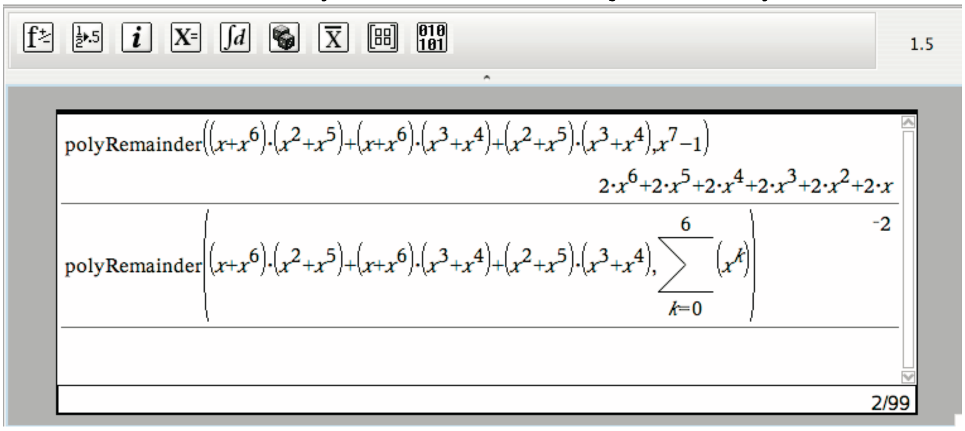
$$x^3 + x^2 - 2x - 1 = 0$$

There are several purposes for this exercise in addition to giving a concrete (computational) preview Kronecker's construction of splitting fields for algebraic equations:

- In an informal way, students preview the idea that one can model $\mathbb{Q}(\zeta)$ by "remainder arithmetic" in $\mathbb{Q}[x]$, using $x^7 - 1$ as a divisor.
- In fact, one can use any polynomial that has ζ as a zero—the smallest degree one is

  $$x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$$

  Doing so would have reduced significantly the simplifications needed at the end of each step, and the CAS would carry out the calculations just as easily.



- The CAS model allows students to experiment with arithmetic in $\mathbb{Q}(\zeta)$ by performing arithmetic with polynomials.

Veteran Logo users will recognize that this idea of of modeling algebraic structures goes back to, for example, the Logo activities in which students modeled $\mathbb{C}$ via arithmetic with paris of real numbers. The version presented here is a kind of refinement of those ideas, this time using formal algebraic expressions as the modeling tool rather than data structures like lists.

On CAS Use

CAS environments have been used for over a decade in undergraduate mathematics, and now, with the availability of these media on handheld devices, they are gradually making their way into precollege (upper secondary) programs. Especially in the United States, where jumping on bandwagons has a longstanding and quasi-respectable tradition in education, two opposing camps are developing:

- Many people are worried that the influx of CAS environments into precollege mathematics will produce a generation of high school students who reach for a calculator to factor x^2+x, much like the alleged current generation of college students who reach for a calculator to multiply 57 by 10.
- And there are those who adopt the motto "if the machine can do it, why bother teaching it?"—many educators are proclaiming that facility with algebraic calculation is unnecessary and that we can do away with those tortuous pages of factoring, simplifying, and solving.[3]

Experience tells us that both of these extreme stances will evolve eventually into something much less grandiose and that CAS environments will take their place alongside other useful computational media as enhancements to, rather than replacements for, the essential role that technical fluency plays in mathematical understanding.

[3] Paul Goldenberg was at a meeting of US mathematics curriculum developers some years ago when someone made the comment that algebra is dead, causing a roaring round of applause from the audience.

In this paper, I've provided one example of how CAS environments can be used to enhance the high school algebra curriculum. *CME Project* uses CAS technology to

1. *Experiment with algebra*
2. *Reduce computational overhead*
3. *Use polynomials as modeling tools*

Our work with teachers and students thus far has convinced me that computer algebra is a very useful tool to help people bring the objects of mathematics, especially formal mathematical expressions, into their realities.

The examples given in the previous sections are just that: examples. There are many other examples of modeling opportunities that have little to do with $x^n - 1$: Chebyshev polynomials, Lagrange interpolation, Newton's difference formula, and generating functions, just to name a few. All of this beautiful and classical mathematics is now accessible to many more students than in previous decades, and all of it becomes "real" in a CAS environment.

—Center for Mathematics Education, EDC, Newton MA, 02458
acuoco@edc.org

References

[1] Education Development Center. *CME Project* (2009). Pearson, Boston, MA.

[2] Cuoco, A. "Mathematics as a Way of Thinking about Things". In Steen, L. (ed) *High School Mathematics At Work* (1998) National Academy Press, Washington DC.

[3] Cuoco, A., Goldenberg, P., & J. Mark. (1996) "Habits of mind: an organizing principle for mathematics curriculum" J. Math. Behav. 15(4):375-402.

[4] Cuoco, A. (1997) "Constructing the Complex Numbers." IJCML, 2(2):155-186.

[5] Cuoco, A. "The Case of *CME Project*". In Hirsch, C. (ed) *Perspectives on Design and Development of School Mathematics Curricula* (2007). NCTM, Reston VA.

[6] Goldenberg, E.P. "Algebra and Computer Algebra." In *Computer Algebra Systems* (J. Fey et al, eds) NCTM:Reston, 2003.

[7] Cuoco, A. and Levasseur, K. "Classical Mathematics in the Age of CAS." In *Computer Algebra Systems* (J. Fey et al, eds) NCTM:Reston, 2003

[8] Cuoco, A. and P.Goldenberg. "A Role for Technology in Mathematics Education." *Journal of Education*, 178:2, 1996.

[9] Suzuki, J. "On Coefficients of Cyclotomic Polynomials." Proc. Japan Acad., Ser. A,1987.

[10] Harel, I. and S. Papert *Constructionism* 1991 Ablex Publishing Corporation Norwood, NJ

[11] Lappan, G., Phillips, E., et.al. "Prime Time", a unit in *CMPII* (2008). Pearson, Boston, MA.

Building tunes block by block: Constructing musical and cross-cultural understanding through Impromptu

Michael P. Downton, *mdownton@indiana.edu*
School of Education-Learning Sciences Program, Indiana University, Bloomington, IN, USA

Kylie A. Peppler, *kpeppler@indiana.edu*
School of Education-Learning Sciences Program, Indiana University, Bloomington IN, USA

Adena Portowitz, *adenap2@gmail.com*
Bar-Ilan University, Ramat Gan, Israel

Abstract

Using a constructionist framework in music, specifically through an emphasis on composition, is revolutionizing the field of music and education by bridging the gap between the novice and professional. Much of the research has been spearheaded by Jeanne Bamberger and others, who noted the computer's potential to highlight what it means to be a composer and facilitating those with no musical background to express their musical 'intuitions' through the use of the computer (Bamberger, 1972, 1975a, 1975b, 1991). Her close work with Seymour Papert at MIT allowed her to develop MusicLOGO and *Impromptu*, which allows users to manipulate small blocks of melodic and rhythmic patterns, employing mathematical ratios, finding that people with little to no training in music, knew more than they could verbalize. Through their active constructions of tunes, they were building and developing intuitions about music. Despite her work and the work of other leading scholars in the field, constructionism is still a framework largely overlooked and understudied in the field of music education. However, music, specifically composition, is well aligned with the major tenets of constructionism and there is little known about how learners form a social, cultural, and historic identity through music composition. This is now an apt time to investigate how we can begin to use these tools to study how the sociocultural context changes learners' intuitions about music. The current study investigated how children develop musical understanding through cross-cultural activities composing music. 60 youth, equally divided, from the United States and Israel ranging from 8 to 12 years of age, reconstructed familiar and unfamiliar tunes, remixed tunes, and composed their own music using *Impromptu.* Each exercise built upon itself to help youth gain a better understanding of important musical concepts and allowed us to better understand what youth know about music from their own culture as well as others through their active construction of music compositions. Data is currently being triangulated (Lincoln & Guba, 1985) using three qualitative data sources, including artifact (music composition) analysis, reflections of artifacts, and discussions of shared music compositions. Preliminary findings suggest that through the active construction and reconstruction of tunes, youth refined their intuitive musical understanding as well as becoming more aware of the cultural differences reflected in other styles of music.

Keywords

Technology in music education; cross-cultural understanding; sociocultural constructionism

Introduction

Music composition was thought to be, at one time, classical in nature and left to those with many years of training, practice, and performance experience (Wiggins, 2009). The notion of what a composer is and who specifically can be a composer has become less distinct over time. Popular musicians are now relying on more sociocultural practices—playing in bands—rather than formal, conservatory education (Green, 2002). The lines are becoming more blurred since the personal computer and music have merged to offer users with little to no training or experience to become composers of music. These technologies have now moved from the confines of professional recording studios to homes and classrooms throughout the world (Savage, 2005; The´berg, 1997). Most software packages allow users to compose songs via pre-made loops—small bits of melodies and/or rhythms—record their own instruments, utilize MIDI, and even use traditional notation; becoming composers overnight.

Recognizing the importance the computer could play in music was Jeanne Bamberger who, in the 1970's was a researcher at the M.I.T. Artificial Intelligence Laboratory, worked alongside Seymour Papert and developed *MusicLOGO*, a computer program that allowed users to manipulate 'blocks' of tunes using mathematical ratio's. This later developed into what is now known as *Impromptu. Impromptu* allows users to reconstruct, remix, and construct tunes using 'tuneblocks'—virtual blocks that contain portions of melodies and/or rhythmic patters—all while building an understanding of important musical concepts such as form, melody, pitch, rhythm, and structure (Bamberger, 2000). What makes Bamberger's work important, and still relevant, is the high importance placed on the learner reflecting on the decisions they make in the construction process. This reflective process, not available on commercial software packages, is built into the software and now the composition and the thought processes in the construction of the composition become the artifacts available to the community at large. This is now and apt time to investigate how we can begin to use these tools to study how the sociocultural context changes learners' intuitions about music.

The current study investigates how children develop musical understanding through cross-cultural activities of composing music. While *Impromptu* was initially developed to help college-aged students become aware of their musical intuitions and was never meant as a composition tool for young children, our study takes advantage of the compositional tools available in *Impromptu* and gives younger learners a chance to manipulate and compose music from their own culture and cultures they are unfamiliar with. Youth, comprised of approximately 60 youth, equally divided, from the United States and Israel ranging from 8 to 12 years of age, will be reconstructing familiar and unfamiliar tunes, remixing tunes, and composing their own music using *Impromptu.* Each exercise builds upon itself to help youth gain a better understanding of important musical concepts (Bamberger, 2000) and will allow us to better understand what youth know about music from their own culture as well as others through their active construction of music compositions. Data will be triangulated (Lincoln & Guba, 1985) using three qualitative data sources, including artifact (music composition) analysis, reflections of artifacts, and discussions of shared music compositions. Preliminary findings suggest that through the active construction and reconstruction of tunes, youth refined their intuitive musical understanding as well as becoming more aware of the cultural differences reflected in other styles of music.

Background

While the research on music has been numerous, the research on music education in the areas of performance, teaching, pedagogies, and attitudes toward music education (see Duke, 2000; Colwell, 2006; Jorgenson, 2002; Allsup, 2002; Asmus, 1986; Goolsby, 1999) have helped further the notion that music and the arts, is a seemingly untapped area to further develop learning environments to foster knowledge development, social growth, and efficacious learners. The arts, and music specifically, have been slow to move away from a more information processing approach to learning to more knowledge constructing environments with the teacher and students co-constructing their understanding. The use of the computer in this construction is viewed as little more than a performance enhancer (drill and practice software) or tool for consumption rather than creation (c.f., Webster, 2007 for full review of technology in music education). Even in her thoughtful views of teaching and learning music in a constructivist framework, Wiggins (2009), devotes very little to how the computer can help children develop their musical understanding. This is not to say that all music educators and researchers feel this way, but that they have not considered the important role the computer and its accompanying software can play in developing a learners understanding through constructing and reflecting of musical artifacts. Most artifacts and assessments in music are performance based (Goolsby, 1999); what can the learner do on a particular instrument with little to no regard for what the learner actually knows. Fortunately, in recent years, researchers have begun to investigate music composition (Swanwick and Tilman, 1986; Brophy, 1996; Burnard, 2000; Strand, 2005). The concern now is the computer's place in music composition. If the computer can be viewed as an extension of the learner instead of from a performance or music consumption view, than it becomes an "object to think with" (Papert, 1980).

One researcher who recognized the computer's potential in musical understanding in the early 1970's was Jeanne Bamberger. Bamberger argues that people know more than they can actually talk about through peoples' construction and reflection on music compositions. She argues that since music has it's own rule sets, people who are not exposed to it often, make their own sense of the sensory phenomena that happens in music; hence the notion of an intuition. If people have intuitions about music, and then are taught something that conflicts with these intuitions, it confuses the learner and makes learning more difficult (Bamberger, 1972, 1975a, 1975b). Wiggins (2009) agrees that young learners especially should not be exposed to the traditional notation system and favors allowing youth to create their own musical representations.

What makes Bamberger's work important, and still relevant, is the high importance placed on the learner reflecting on the decisions they make in the construction process. This reflective process, not available on commercial software packages, is built into the software and now the composition and the thought processes in the construction of the composition become the artifacts available to the community at large. Using these reflections, we can also reveal aspects of the learner's cultural identity. Little is known about how systematic reflections (with tools like what's built into *Impromptu)* reveal a sociocultural understanding of music and composition. The current investigation begins to use these tools to study how the sociocultural context forms the learners' intuitions about music.

Guiding Theoretical Framework

The theory learning based on constructionism builds on Piagetian frame of constructivism—making sense of the world around us through assimilation and accommodation of schemas—and adds that this happens when learners are actively engaged in constructing an artifact that is personally and epistemologically meaningful to them (Papert, 1980, 1993; Kafai, 2006; Bers, 2007). Building on this, sociocultural constructionism (Pinkett, 2000; Peppler & Kafai, 2007) argues that both individual and community development are better understood when the artifacts are an expression of the individual and the community as a whole and our understanding of the artifacts changes because of the sociocultural nature of the activity. While constructionism and sociocultural constructionism has been taken up in the world of math, science, robotics, and game design (c.f., Kafai, 2006; Bers, 2007; Kafai and Resnick, 1996; Peppler and Kafai, 2007), little attention has been given to music learning.

Research Approach

This research sought to understand how children, while constructing familiar tunes, re-mixing familiar tunes, and constructing their own tunes, developed their understanding, capabilities, and reflection of musical concepts.

- What concepts or musical ideas do children learn about while constructing their musical artifacts?
- What do the reflections reveal about the child's cultural context and how does this vary cross-culturally?

Settings and Participants

To investigate these questions, we have coordinated with a elementary school classroom at a school located in a mid-sized, Midwestern city in the United States and a school and afterschool program in Ramat-Gan, Israel. Working with Israel provides a unique opportunity to investigate how the cultural context, as it relates to music, changes when the learners are engaged in the constructive, composition process. Youth in Israel are acculturated at an early age with traditional Israeli folk tunes, which differ from Western music in many ways; most notably in the use of semitones, a predominance of minor modes, and non-conventional phrase structure.
Participants, approximately 60 total and equally divided amongst the two sites, range from 8 to 12 years of age. Each location is equipped with a computer lab, consisting of Windows based computers with *Impromptu* installed on each computer and headphones for personal listening.

Methods

Research took place between February and July 2010. The 40-hour curriculum began with three exploratory exercises that introduced them to how *Impromptu* works. Each exercise was meant to be an introduction into how to use the *Impromptu* interface. Important in these exercises is the process of reflection. Learners were asked to write about each of the decisions they make during the composition process and why they made those decisions. Learners began to reconstruct and remix music from unfamiliar cultures to their own such as Chinese, Arabic, and American folk tunes. Once the exploratory exercises were complete, learners then began to compose their own piece of music using *Impromptu.* Compositions were shared both locally and cross-culturally and others were encouraged to reflect on each other's compositions.

Data Sources and Analysis Strategies

There were three data sources using three qualitative data sources, including artifact analysis, reflections of artifacts, and discussions of shared music compositions. Music compositions were analyzed and coded for development of the learners' intuitive understanding of musical concepts. Professional composers were used to identify certain concepts such as melody, rhythm, and form; essential functions of music composition. Learner reflections were also coded to note utterances of development of musical concepts mentioned such as "it sounds familiar to the people listening and that pulls a song together". Written reflections were also coded to identify how the learners view of themselves individually as well as socially, culturally, and historically, as a composer of music. We looked at utterances that would point to changes of their identity and compared it within and between groups such as "my music sounds happy" or "this music seems to not have an ending and I can fix it by adding this note". Finally, we coded the written and verbal reflections to point out how learners' cultural understanding of music changes over time by identifying utterances such as "there are too many notes and it's hard to follow" or "this doesn't sound like anything I've heard before, but it's interesting"

Findings

At the time of this proposal, data is still being collected with an expected end time of May 2010. Preliminary findings suggest that, through the active construction and reconstruction of tunes, youth are developing an understanding of musical concepts such as pitch, melody, and rhythm as well as the cultural differences in other styles of music.

One such instance of musical understanding comes from Ella, a 12-year-old female from the United States. Ella, like most of her classmates, participates in a weekly general music class. However, her exposure to more formal music (e.g., private lessons) outside the class is non-existent. As part of the 're-mixing' exercise, Ella was asked to remix the given tuneblocks in *Impromptu.* This particular tune, "Austrian", was unfamiliar to her. She was instructed to reflect on every decision she made and why. While her reflections may be brief, she is clearly thinking about pitch and the structure of her composition. Below are her reflections on remixing the tune "Austrian":

> *I did what I did first because it sounded different and interesting.*
> *I did that because they went well together.*
> *It kind of works with the feel of the song.*
> *I did the next thing because it went well with the last one.*
> *I put this one next because it sounds familiar to the people listening and that pulls a song together.*
> *I did this next one because it feels out of place and surprises people.*
> *This one was because it sounded like something was ending because it went down.*
> *The next one was because It made the impression "This isn't over yet".*
> *This one just to repeat it one more time.*
> *This one to pull the other one together.*
> *To end it nicely.*

While Ella's reflections are short, they clearly point to her thinking about the songs melody (*"it sounded different and interesting"*), the form (*"…it sounds familiar to the people listening and that pulls the song together"*), and the structure (*"…it made the impression "This isn't over yet"*). These are all components of a composition that professional composers think through when they are writing using traditional notation (Swanwick and Tillman, 1986).

One other such instance is from Pia, a 12-year-old girl from the United States, that has some music exposure outside her school environment that is mostly driven by parental encouragement. Like Ella, Pia was also asked to 're-mix' an unfamiliar tune and keep a journal of her decisions as she constructed her new tune. This particular tune was a in the style of a traditional Arabic folk song. Pia (see Table 1) noticeably moves from thinking about the functions of music to the properties that make these functions possible. Also noted was her level of listening and how it developed from a less to more critical disposition.

Table 1-Pia's reflections and researcher comments on "Arabic" tune

Pia's Reflections on"Arabic" tune	Researcher Comments
1. First I did the gray one because I like how it gives the music a mysterious start to it, I like how it makes it song sort of creepy.	*Using terms like "creepy" speaks to the function of the music. Also trying to explain the cultural differences in the music.*
2. Then I did a purple one because I like how it makes the music sound like it ends, because the purple one has the notes that make it sound finished.	*Points out resolution in the tuneblock and how it sounds like it ends the music. Again, speaking to the function of the music*
3. After that I did a blue-green one because after I did the gray and purple one it made it sound sort of like and ending, and then the blue-green one comes up and it makes you think again.	*More function related talk here. Also using compositional functions like form in her composition.*
4. Then I did a green one because I like how it goes really high because after the blue-green one it sounded as if I needed to go higher, and the green one did that.	*Moving from function to what brings about these functions (e.g., the properties) and how that can help her composition*

5. Then I did a purple one again because I think it flows smoothly with after the green one, it makes it sound sort of like a scale going down. After that I did a blue-green one because I think it is the only one that sounds good after the purple one because it gives an ending feel to it, and the blue-green one it the only one that gives me a beginning feel to it	*Again, talking about functions and what the properties of the functions are.* *This also highlights her level of listening (e.g., critical) and the 'to-and-fro' between listening and creating. Very important when composing music.*

By comparison, the next table presents reflections of a 10-year-old boy at the Israeli site, named Moshe. The original melody of the tune "Arabic" featured the following order of blocks: 2 triangles – 2 triangles – 1 red - 1 triangle – 1 green – 1 purple. In comparing the original tune with Moshe's, we can see that both tunes opened with a repeat of the same block; both tunes featured a middle section that included the same two blocks, however, in a different order; and both tunes ended with the same green and purple blocks. However, Moshe's melody repeated the last two blocks, perhaps reflecting a more Western need for balance between the three sections.

Table 2- Moshe's reflections and researcher comments on "Arabic" tune

Moshe's Reflection on "Arabic" tune	Researcher Comments
1. I chose the grey one with the 2 triangles because it sounded like a nice beginning.	*The student was probably relating to the contour of the melody.*
2. I repeated this block again because this sounded like a stronger beginning and reminded me of the repeats in Frere Jacques.	*The student used repeats as a means of strengthening the beginning, middle and end parts of his song. He also remembered that we had studied a song which had highlighted repeats.*
3. I then chose the grey block with one triangle because I wanted the melody to go higher	*Beginning the middle section of the song, the student chose to rise in register.*
4. Then I chose the red block because its melody was also high	*As in the opening, here, too, the student was working in units of two.*
5. After this, I chose the green block because I wanted the melody to go down	*Feeling that it was time for a change, the student chose to balance the rise in melody with a fall in the melody.*
6. The purple block sounded like an ending	*The purple block sounded like a closing unit because of its melodic direction, which pointed down.*
7. I then repeated the green and the purple blocks so that I would have a strong ending	*Once again, the closing section was repeated twice, complimenting the opening and middle sections.*
8. I don't know other melodies that sound like this one, and I think that the melody sounded sad.	*Moshe was not familiar with Arabic songs, having come from an African background. However, he felt that the melody sounded sad, perhaps relating to its modal character.*

Discussion

Music and the arts is an area largely ignored by the learning sciences (Peppler and Kafai, 2008; Peppler and Davis, 2010) and constructionism as a framework in music is non-existent in the literature. The purpose of this study is to apply a sociocultural constructionist view to music learning by allowing youth to engage in music composition activities that builds individual and cultural identity. While the data presented represents our early findings, it clearly shows a direction of musical understanding and it is our intention to show further music learning and well as the role music plays in developing cross-cultural understanding. As youth articulate their ideas and assumptions about how music operates, this opens both an inner-conversation with the learner as well as classroom dialogue about the cultural differences found in various musical forms. Impromptu and the embedded design features that support reflection and deeper listening, forces the learner to articulate their intuitions and begin to articulate the foundations of the theoretical underpinnings of the cultural roots of music -- building a bridge to some of the big ideas of ethnomusicology. Preliminary findings suggest that through the active construction and reconstruction of tunes, youth refined their intuitive musical understanding as well as becoming more aware of the cultural differences reflected in other styles of music.

References

Allsup, R. E. (2002). Crossing over: Mutual learning and democratic action in instrumental music education. Thesis (Ed. D.)--Teachers College, Columbia University, 2002.

Asmus, E. P. (1986). Students beliefs about the causes of success and failure in music: A study of achievement motivation. *Journal of Research in Music Education, 34*, 262-278.

Bamberger, J. (1972). Developing a musical ear: A new experiment. *Massachusetts Institute of Technology A. I. Laboratory. Memo No. 264.*

Bamberger, J. (1975a). Development of musical intelligence I: Strategies for representing simple rhythms. *Massachusetts Institute of Technology A. I. Laboratory. Memo No. 342, Logo Memo 19.*

Bamberger, J. (1975b). Development of musical intelligence II: Children's representation of pitch relations. *Massachusetts Institute of Technology A. I. Laboratory. Memo No. 401, Logo Memo 43.*

Bamberger, J. (1991). *The Mind Behind the Musical Ear: How Children Develop Musical Intelligence*. Harvard University Press: Boston, MA.

Bamberger, J. (2000). *Developing Musical Intuitions: A Project-based Introduction to Making and Understanding Music.* New York: Oxford University Press.

Bers, M. U. (2007). *Blocks to Robots: Learning with Technology in the Early Childhood Classroom.* Teachers College Press: New York, NY.

Brophy, T. S. (1996). Building music literacy with guided composition. *Music Educators Journal, 83,* 15-18.

Burnard, P. (2000). How children ascribe meaning to improvisation and composition: Rethinking pedagogy in music education. *Music Education Research, 2,* 7-23.

Colwell, R. (2006). *MENC handbook of musical cognition and development.* Oxford University Press.

Duke. R. A. (2000). Measures of instructional effectiveness in music research. *Bulletin of the Council for Research in Music Education, 143,* 1-48.

Jenkins, H. (2006). *Convergence Culture: Where Old and New Media Collide.* New York University Press: New York, NY.

Jorgensen, E. R. (2002). The aims of music education: A preliminary excursion. *Journal of Aesthetic Education, 36,* 31-49.

Goolsby, T. W. (1999). Assessment in instrumental music. *Music Educators Journal, 86,* 31-

35+50

Lincoln, Y. S. and Guba, E. G. (1985). *Naturalistic Inquiry.* Sage Publications: Newbury Park, CA.

Kafai, Y., and Resnick, M. (1996). *Constructionism in Practice*: *Designing, Thinking, and Learning in a Digital World.* Lawrence Earlbaum: Mahwah, NJ.

Kafai, Y. (2006). Constructionism. In R. K. Sawyer (Ed.) *The Cambridge Handbook of the Learning Sciences* (p. 35-46).

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas.* Basic Books: New York, NY.

Papert, S. (1993). *The Children's Machine*. Basic Books: New York, NY.

Peppler, K. and Kafai, Y.B. (June, 2008). *Literacy and the Learning Sciences: Creating a Framework for Understanding and Analyzing Youths' Media Arts Practices.* Proceedings published in the 2008 International Conference of the Learning Sciences (ICLS) held at the University of Utrecht, Utrecht, Netherlands.

Peppler, K. & Davis, H. (June, 2010). *Arts and Learning: A Review of the Impact of Arts and Aesthetics on Learning and Opportunities for Further Research.* Proposal for a paper to be presented at the International Conference of the Learning Sciences, Chicago, IL.

Pinkett, R. D. (2000) Bridging the Digital Divide: Sociocultural Constructionism and an Asses-Based Approach to Community Technology and Community Building: Paper Presented in the 81st Annual Meeting of the American Educational Research Association. New Orleans, LA.

Savage, J. & Challis, M. (2001). Dunwich revisited: Collaborative composition and performance with new technologies. *British Journal of Music Education,2*, 139-149.

Savage, J. (2005). Working towards a theory for music technologies in the classroom: How pupils engage with and organize sounds with new technologies. *British Journal of Music Education, 2,* 167-180.

Strand, K. (2007). 5 steps for leading students in classroom composing. *Teaching Music, 5,* 42-47.

Swanwick, K., and Tillman, J. (1986). The sequence of musical development. *British Journal of Music Education, 3,* 305-339.

Th´eberge, P. (1997). *Any Sound You Can Imagine: Making Music/Consuming Technology.* London: Wesleyan University Press.

Torff, B. (2002). A comparative review of human ability theory: Context, structure and development. In MENC Handbook of Musical Cognition and Development. Ed. Richard Colwell. Oxford University Press.

Webster, P. (2007). Computer-based technology and music teaching and learning: 2000-2005. In L. Bresler (Ed.) *International Handbook of Research in Arts Education, 16*, 1311-1328.

Wiggins, J. (2009). *Teaching for Musical Understanding.* (2nd Ed.). Center for Applied Research in Musical Understanding: Oakland, CA.

Can there be a Science of Construction?

Micheál Ó Dúill, *mikedoyle@logios.org, logios.org@googlemail.com*

Abstract

What is Constructionism? What is Instructionism? How are they related? Is either idea well enough defined to be tested scientifically? Or are both philosophical postures adopted by people who have differing views of children and education? If both are more philosophical than testable, they have little relevance for classroom teachers. The craft of the teacher has a deep history. It has employed "talk and chalk" for over five millennia. The only novel element is the computer.

The art of science is simplification – and surprise. The current formulations: Constructionism as making physical objects; Instructionism as verbal transmission, are inadequate. However, two aspects of Logo, Turtle Graphics & Talk, suggest a route forward. Were these to represent two distinct human capabilities, we might reduce capacity to Construct to the human ability to draw. Instruction may then be reduced to the human capacity to talk. We contrast drawing with talking.

A review of behavioural science reveals a massive hole in psychology. We do not know how human beings have the capacity to draw – in contrast to quite detailed knowledge about speech. It follows that proponents of verbally-based instruction have a scientific basis Constructionists cannot match, The advent of a constructive medium, the computer, highlights this deficit.

The task in this paper is to assemble evidence that might offer Constructionism some scientific support. Our species is considered in its evolutionary setting. This includes mapping the sequence of hominine evolution and considering genetic foundations for society and technology. Two important concepts are: reciprocal altruism; and the extended phenotype. Aspects of the large brain that characterises our species are considered, particularly the executive function of prefrontal cortex, and its connectivity and maturation in relation to primary education.

A hypothesis about how humans are able to draw is presented. The theory is that the human brain is more extensively interconnected than that of our precursors. Uniquely, prefrontal cortex accesses information about how the brain processes primary information. Support for this hypothesis is provided from primary school years, the archaeological record of our species, and a small psychological experiment. Herein are found the elements of our technological capability. The word 'technicity' is borrowed from philosophy to denote this evolutionary adaptation.

The idea is applied to the media used in primary education, specifically in literacy and numeracy. t raise the question of whether a constructive medium would help children to learn more easily. There is also an implication that primary education in practice has an overall technicity focus.

Given the evolutionary importance of verbal communication, research is needed to clarify the role of "talk and chalk" in the classroom relative to construction. We are unconscious users of language and view literacy from a utilitarian perspective: a skill for learning and life. Is it perhaps time to ask the question: In what manner has technicity enhanced and refined language?

In summary, the claim is made that available science may not only support the constructionist position but also its stance on instructionist method. Technology, the hallmark of humanity, is at the heart of education. The new computer technology is arguably our most powerful constructive medium. The analysis in this paper suggests that increased use of the full potential of the new medium for active processing in teaching method with bring major cognitive benefits.

Keywords

Computers, learning, talking, drawing, constructivism, evolution, teaching method, technicity

Introduction

I am a teacher. I have taught children for many years, at both primary and secondary level. However, most of the children I taught had learning difficulties. Throughout my teaching career I have been beset by theorists and so-called educational innovators. I have been asked to be modern and then to be traditional. When, in the 1980s the computer arrived. I thought, “Here is a new educational medium with huge potential.” (Doyle 1986). I was excited by the consequences for literacy of a machine that could read and write; excited by a machine that could do sums; and I was excited by a machine that children could teach to do things. That was three decades ago.

I have enjoyed being involved with Logo, although opportunities to use it with my children were, and remain, limited. But the Logo community and the biennial EuroLogo Conferences provided an opportunity for me to think outside the classroom box. I met people who had a view of the computer not dissimilar from mine. I didn’t really mind that these conferences tended to be mathematically oriented. There was a delightful mix of practical innovation with innovative practice. I was made uncomfortable, however, by the small amount of such innovation that transferred to school. Comenius Logo, when it arrived in the 1990s with graphics as primary data, seemed better suited to the everyday classroom. I am pleased that Logo-based microworlds introduce children to computers in at least one country (Ilieva and Ivailov 2003).

I was not too concerned when the Eurologo name was dropped for “Constructionism,” and an extra year skipped to avoid clashing with WCCE2009. However, when I was asked by a very talented primary school teacher, what “constructionism” was I found myself in some difficulty. I (just) survived the behaviourism of the ‘60s and ‘70s and avoided the traditionalism of the ‘80s and ‘90s. Am I to be seduced by another “ism” In the twenty-first century?

No! A classroom teacher with boisterous children, I need effective techniques at my fingertips. But, with the computer this was, and still is, not possible. The computer is a new medium. I want teaching methods that fully realise its potential. However, the formulation of Constructionism has forced me to take a step back from the computer and to consider what it is to construct.

Towards science

Constructionism is defined through philosophical discussion and illustrative parable (Harel and Papert 1991, Papert 1994, LCSI 1999). That is, verbally. A science of construction is seen as premature. Reduction to a simple catch-phrase is resisted. Not unreasonable, but science works by simplification. So that, in the words of an English idiom, the wood can be seen for the trees.

A good starting point is the wording on the Constructionism2010 website:

> Constructionism shares constructivism’s connotation of learning as “building knowledge structures” irrespective of the circumstances of learning.
>
> It then adds the idea that this happens especially felicitously in a context where the learner is consciously engaged in constructing a public entity.

The first paragraph provides a part of our foundation. It restates the Piagetian position, which is now well accepted scientifically (Bransford et al 2000). The second paragraph is our starting point. What does “constructing a public entity” mean? Also, what is the “instructionism,” which Papert contrasts with constructionism?

It is easy to find a simplification for Instructionism. Papert prefixes it with “verbal.” Let us reduce instruction to “**speech**” (the “talk” of school parlance).

Constructionism has been stereotyped as "making things," but this does not capture its essence. I propose to reduce construction to "**drawing**" (the "chalk" of school parlance). Is it not said that "A picture is worth a thousand words."?

Reduction to "speech" and "drawing" brings immediate clarity:

A. Speech is an evolved biological adaptation unique to extant humans (Pinker 1995). It is a genetically determined part of the phenotype. Specific anatomical, physiological and neurological structures have evolved for speech. A normal child born into any culture will become a competent user of its language by the age of four. The languages of all humans, though different in sound and structure, are equally expressive.

B. Drawing is technology. Anatomical, physiological or neural adaptations for technology are not apparent. Children draw from the age of about four, but children's drawings are unreal assemblages of geometric-like forms. Technology is unique to modern humans. It extends our phenotype in myriad increasingly complex ways. Sophistication, function, power, construction, and materials of technology vary across cultures (Diamond 1998).

It is seems that we may have isolated two separate human capabilities. Burling (2005) captures their mutual isolation in the following anecdote:

> The only technical instruction I could ever elicit would come when a man would reach for my tool and materials, demonstrate the manner in which the job should be done, and then hand them back along with the injunction: "Do it like that."

Our technological capability needs a name as concise as "**language**." I will use "**technicity**", a word derived from Heidegger's (1977) philosophical enquiry into the essence of technology

A hole in knowledge

For a summary of scientific evidence, it is useful to turn to student texts. "Atkinson and Hilgard", (Nolen-Hoeksema et al 2009), the introduction to psychology of my student days, has now reached its 15th edition. There is a chapter entitled "Language and Thought." There is no chapter on construction or technology and less reference to drawing than in the original. There is no consideration of how the beautiful illustrations of visual illusions in Gregory's (1966) "Eye and Brain" come about. Books on children's drawing development can be counted on one hand. Kellogg (1969), Gardner (1980), Cox (1992), and Anning and Ring (2004), treat drawing as art. Only Goodenough (1926) considers drawing from a cognitive perspective. Otherwise: nothing.

Verbal instructionists can call upon a huge literature to back their claims. One text influential in education is "Thought and Language" (Vigotsky 1962) which also emphasises social interaction, equally well researched. The school curriculum puts language first; in the UK it is as "English."

Constructionism has a problem.

Evolutionary context

The following section contains the essential points that are relevant but includes sources which can provide additional background or clarification if required.

Human evolution

Darwin's (1859 1968) theory of evolution has proved scientifically more fruitful than those based on mythology (Kramer 1972). The modern synthesis with genetics (Dawkins 1989, Jones 1994) provides a powerful tool for studying human origins (Lewin 1998, Stringer and Andrews 2005, Mellars et al 2007). It seems that human evolution was a three stage process:

1. Evolutionary split from the apes to form the hominine lineage about 2.5 million years ago;

2. Rapid expansion of brain size about 1.8 million years ago. This was accompanied by reduction in sexual dimorphism; extended childhood; range expansion outside Africa into Europe and Asia; a complex tool assemblage including a characteristic bi-facial hand-axe; and increased behavioural complexity. The species identified are Homo ergaster followed by Homo erectus. Evidence suggests that speech originated in these species.
3. A second burst of brain expansion happened 0.5 a million years ago. It is associated with archaic Homo sapiens. Two well documented species are the Neanderthals and modern Humans (us). Cold-adapted Neanderthals were confined to Europe. Genetic evidence suggests our direct ancestors emerged in Africa by 200,000 years ago. There is evidence for 'modern' behaviour (technology) from 300,000 years ago (McBrearty and Brooks 2000). Some 70,000 years ago we spread out from Africa, reaching Australia (by boat) 50,000 years ago. We were living in Europe 10,000 years later, where we influenced and out-competed the Neanderthals. Both we and the Neanderthals had a fully developed suite of speech adaptations. So, although we differed in our technological capability, both species (and our common ancestor) may have had genetically modern language.

The balance of the evidence supports the proposition that spoken language evolved before our species did. Conversely, there is no evidence that technicity evolved in any species prior to us.

Genes and society

The survival of the fittest has come a long way since Darwin. Hamilton (1964) showed that nepotism facilitates family gene-pool survival, which may explain, for example, the menopause. More powerfully, Trivers (2000) demonstrated that cooperation between strangers can enhance the individual survival of both. This life-style is called "reciprocal altruism" (RA). It is the way we live. The prerequisites include the ability to recognise other individuals (we recognise faces) and good memory for events. These favour species with a large neo-cortex. The problem with RA is cheating: attractive in the short term but disastrous – mathematically modelled as the "Prisoners Dilemma" (Axelrod 2006, Cosmides and Tooby 1992). The loophole for cheats is the need to cooperate on first meeting. Hence, a stranger is more likely to be a cheat than a neighbour. Nettle (1999) demonstrated that certain characteristics of language, including accent and language diversity, are powerful determinants of successful reciprocal altruism in terms both of stranger detection and group identity. Dunbar (2004a 2004b) suggests that gossip and theory of mind (ToM) both assist this life-style. The levels of intentionality (ToM) at which people operate when gossiping is commensurate with levels of recursion (embedded clauses) used in language.

We are the only species to live in huge city communities with specialised services and roles, which we trade with strangers. The relationship between reciprocal altruism, large group living, and language suggests that this biological adaptation is prerequisite for our life-style.

A large brain

A big neo-cortex processes more information, so aids survival in complex environments (Ashby 1971). The first cortical expansion (above) was adaptive in complex natural environments. Byrne and Whiten (1988), Whiten and Byrne (1997) suggest that the later expansion was an adaptation to a complex social environment (see Barrett et al 2002). A larger cortex entails a greater range of capabilities (Deacon 1997). New brain areas are created and there is increased connectivity (Streidter 2005). In the human, the greatest expansion occurred in prefrontal cortex. Prefrontal cortex is massively reciprocally connected to all parts of the brain. It has an executive function, providing working memory, selective attention, and planning functions (Fuster 2008). The orbito-medial part is mainly connected to the limbic system and is concerned with motivation and long-term planning (Damasio 2006). Lateral prefrontal cortex is connected to pre-motor and sensory association areas. It is largely involved with cognition. No area of this cortex can be isolated as a language-module. Language processing, like other working memory tasks, appears widely distributed. The role of prefrontal cortex is to "invent futures from the past;" that is, to access and reassemble memory to offer the choice of a range of alternative action-scenarios in a given circumstance. In other words, it is the source of creativity for humans and other mammals.

In the human, prefrontal cortex matures rapidly between the ages of 6 and10, reaching the adult stage by about 12. Lateral (cognitive) prefrontal cortex continues to mature into the third decade of life. Phases of education in industrial societies appear to run in concert with this maturation.

Genes and tools

Biological (phenotypic) adaptations are built by genes. Dawkins (1999) powerful idea of the “extended phenotype” brings construction into the genetic realm. A bird is more likely to survive if nest ‘design’ is built into its brain as a genetically determined behaviour – a template. This is because generational transmission is unreliable, even in chimpanzees (Matsuzawa et al 2001). Learned behaviour is viable only if not critical for survival. An indicator of genetic determination is stability over time. The tool assemblages of all hominine species, other than modern humans, were stable for very long periods: Homo erectus, 1.5 million years; Neanderthals, 300,000 years.

We are the only species to let the genetic tool-template atrophy. For us, the risk inherent in generational learning is mitigated by the RA lifestyle and speech. But there is no benefit unless tools become technology. I.e. there is a mechanism for generational learning to improve design.

Whence technology?

I offer pointers to a process by which one hominine developed a lifestyle of such complexity that it can support the plot of a Shakespeare play with all the levels of interpersonal intentionality therein (Dunbar 2004b). It is not surprising that Deacon (1997), a neuro-scientist, argues that our symbolic capabilities emerged from a co-evolution of language and brain. Default to Language is attractive but it cannot be sufficient; and neither can the socio-sexual driver that Power (1999) perceives in the evolution of art. Other species spoke and had complex social relations, but they didn’t build a computer, even after 200,000 years. The literature does not illuminate the mental ‘how’ of human technology. Brain research offers little enlightenment. I hope education will.

Technicity: a hypothesis

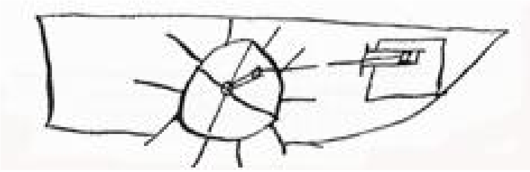
Hubel (1995) described feature detecting neurones in primary visual cortex. There are neurones that react to lines of specific orientation, and of specific length; colour neurones distinguish between blue and yellow, red and green, and dark and light; and motion neurones that detect movement in a particular direction. In primary auditory cortex, neurons respond to notes of specific pitch. Here is elemental data from which more complex entities might be constructed.

Prefrontal to primary sensory connection is feasible. Some neurologists presume a connection. Though, such connectivity is not apparent in the primate (Crick and Koch 1995). Let us suppose there is direct prefrontal connection to primary sensory cortex. Elemental feature information could be accessed, assembled and combined in prefrontal cortex, as if it were a stored memory. Here is a neural mechanism for the technology improvement-cycle, geometry and mechanisms.

Drawing

Unlike the words of language, elementary sense-data are not arbitrary symbols. They are reality abstracted. This is why I have made drawing stand proxy for our technological capability.

No other animal draws. Children and chimpanzees scribble, but even a chimpanzee that had symbolic language failed to learn reliably to join dot to dot (Iverson and Matsuzawa 2001). Children’s drawings cover classroom walls. This indicates that drawing relies on some peculiar organisation of the human brain. During child development, drawing follows language. But unlike language, graphic development is gradual with competence not appearing until puberty. This suggests co-development with prefrontal cortex. Primary visual cortex provides the line-elements needed, prefrontal cortex the planning and pleasure. Below, figure 1, are two drawings.



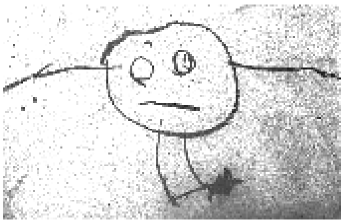


Figure 1a. (Left) Trevithick's drawing of a steamboat with paddle wheel amidships, 1806.
Figure 1b.(Right) A 'tadpole' drawing from the Kellogg online collection (age 3 to 4 years).

On the left is a sketch by a professional engineer, on the right an infant's figure. Is there not an essential similarity? They use the same graphic elements; and both show only relevant features.

Goodenough (1926:12) noted that:

… a child draws what he knows, rather than what he sees …

To which we might respond – and so does the engineer! If the child is not drawing what s/he sees: from where within the nervous system does the knowledge come? The human brain has a specific area for recognising faces (Carter 2000:196). So, facial-feature knowledge might be sourced from the facial recognition system. But where do lines and circles reside in the brain?

Childhood evidence

What is the evidence from the kindergarten? Below, figure 2, are examples of infant material:

Figure 2. Kindergarten equipment showing elemental colour, line and pitch features.

Do they not reflect the fundamental features into which our brain decomposes the visual image? Is it conceivable that the toys we give to toddlers actually help the brain to connect to primary sources of sensation? The colours are right. The shapes are right. They are simply and regularly combined, even in the three-dimensional building blocks and the wheel – archetypal technology.

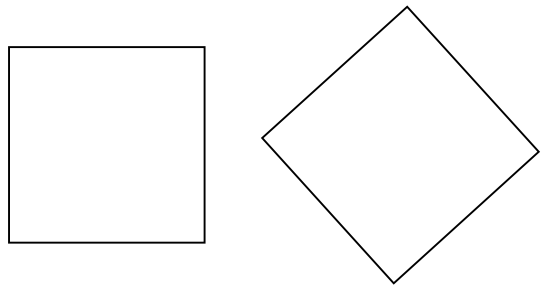
Whilst this evidence is indirect, it supports the notion that the way we structure learning in early years education helps human infants to develop skills in abstracting elemental information.

Archaeological evidence

McBrearty (2007) is insistent that modern human behaviour began to appear 300,000 years ago. She cites presence of bright red ochre and grindstones at living sites. Its use is thought to be like that in some modern cultures – symbolic representation of menstrual blood (Power 1999). But it also suggests 'knowledge' of pure red, i.e. a prefrontal-sensory connection. More persuasive are flints knapped into simple geometric forms to make compound tools, from 200,000 years ago.

Experimental evidence

The earliest mathematics appears to have been geometry. Below, figure 3, are two squares.



(a) (b)

Figure 3. Object-constancy breaking forms

We can do a neat experiment with a square. Show someone drawing (a) and they will name it as a square. Rotate it by one eighth turn to (b) and the word 'diamond,' previously unthinkable, will come to mind. This does not happen with a picture of a cat. Object constancy is a perceptual mechanism that keeps the world the same from different viewpoints. The experiment reveals our capacity to construct forms that by-pass this mechanism. I suggest that humans, uniquely, can create shapes within prefrontal cortex from data available at the neurones of the primary visual cortex. The square is particularly interesting, because lines separated by 90° excite totally different neurones (Hubel 1995). Therefore, there is primary visual data to construct a precise rectangle. I.e. we carry around the Platonic 'ideal' square in the structure of our nervous system.

Technology

I suggest prefrontal to primary sensory connectivity began to develop in a direct ancestor and that it led to a speciation. A process of generational learning accompanied by the capacity to make design improvements could lead to increasingly sophisticated technology. The capacity for 'design improvement' resides in the function of the prefrontal cortex, particularly the lateral convexity. It makes practicable the atrophy of the genetic tool-template through an ability to visualize modified tool designs. This is the evolutionary adaptation for which I use the word technicity. We, alone amongst animals, can imagine a different environment. But our conception is far simpler in form than is nature. Houses, tools, and fields are geometric. Colour is uniform. Music has notes. Children's drawings are not just art precursors; they reveal a capacity for engineering, for graphic technologies, for simplifying and extracting the essence. Surely, surprisingly, may not the foundation of science be discovered in this abstraction of simplicity?

School matters

We are literate and write what we mean. Mathematics and technology march in step (Hawking 2005). The earliest known schools taught the 3Rs 4,000 years ago in Sumer (Kramer 1981).

Two modes of learning

It should now be clear that human beings have two learning modes: a) a robust speech/memory combination, of evolutionary depth serving reciprocal altruism; b) a recent (risky) constructive technicity, expressed in technologies that extend our physical and mental phenotype. Both use a prefrontal capacity for creativity. Only technology progresses – through generational learning.

The computer as an educational medium

Some (Papert 1980) see the computer as a revolutionary agent; others as a means of better teaching the existing curriculum. The latter prevails. Alexander (2009) is fearful of ICT, whilst Rose (2009) sees 'technology' as a skill. Neither sees the computer for what it is: a medium. This is surprising because it has unique characteristics, compared with the oral and textual.

Oral methods rely on human memory. Mnemonics, chant, and ballad aid retention and recall.

Writing is an external memory store. Books are an advance on oral method because they can contain drawings and other notations, as well as words. Euclid took advantage of this.

The computer constructively extends our phenotype by processing information. It is a far larger medium-transition than was writing. Writing and drawing externalized human memory – beneficially. Our new capacity to emulate mental processes, many made necessary by writing, should similarly be beneficial. But conflict has arisen because the computer can mechanically perform “mental” operations – of numeracy and literacy – that are basic in the extant curriculum.

Literacy

Speech fades to imperfect recollection in a breath of air. The meaning of words can be denied by tone of voice. But, by drawing the words of speech we make them open to public scrutiny. We construct writing systems that extract the essence of speech (Robinson 1995). Once words are concrete, a wordsmith can combine them is novel ways, to tell of verbal duplicity (Chang 1991); a psycholinguist can analyse them as a window on thought (Pinker 2008). Literacy gives us a measure of control over speech (Oppenheim 1992). Why is literacy not taught as a technology?

The thrall of speech leads to methods that map spelling to the sounds of a language (Gupta 2001); and to the computational absurdity of speech from text (Taylor 2008). Yet, the English alphabet refines articulatory complexity to 26 letters, simplifying dialect vowel variation to 5: very close to the essential sounds for speech intelligibility (Jenkins 2000). Why does literacy method not now use auditory technology that lets children construct what they see, not what they say?

Numeracy

Accounting with numbers is not natural. We prefer to distinguishing features and name things. Gallistel et al (2005) found that, in common with primates, we naturally use real number. Societies that do account appear to construct number vocabulary according to Miller’s (1956) “Magic number 7±2.” We mentally ‘bundle-up’ number concepts at the level of hands-full.

Language expresses mental chunking at the count of ten. Grouping objects does not. The technology of graphic number representation has improved in design over time. Roman numerals, for example, are closer to physical grouping than to the Latin language. The later Hindu-Arabic place-value numeral system is fully in step with language, including Latin. But it is not congruent with physical object counting. Consider the counting square below, figure 4:

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	2	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

Figure 4. Hundred square

Figure 5. Gear-wheel counter

If good mathematics is elegant, then there is something anti-mathematical about the hundred-square, popular in schools: The numerals don’t fit. (Roman numerals do – try them.) The final column does not model ‘bundling up’ at each ten. So, it is probably out of step with thought: A source of confusion for children? A physical number representation congruent with both modern numerals and language is available, figure 5. It meshes with the gears of Papert’s (1980:viii) childhood. And it nicely illustrates “borrowing” and “carrying”. This, a simple adding machine, is more in tune with children’s minds than error-prone counting. Why is the constructive capacity of the computer to represent number operations not now a route to number understanding?

Method

Teaching has helped children to construct meaning from books – from external memory. Is there not now a pressing need for education to step up to the challenge of a constructive medium?

In summary

The absence of any scientific basis for our constructive and technological capability is the largest inhibiting factor in developing a sound species-level theory of education. I hope the argument I have outlined has enough detail to convince you that there can be a science of Constructionism.

I suggested that language evolved in a precursor species; as a prerequisite for a reciprocally altruistic life-style. An excellent memory is similarly prerequisite. Is it possible that instructionism, with its language basis and memory cramming tendencies, employs primitive evolved features?

Prefrontal cortex connectivity and maturation is at undoubtedly at the heart of primary education, and developing individuality. I argue that Technicity, our species-specific unique constructional capability, is rooted in neural primary sensory data / lateral prefrontal connectivity.

Research implications include language as a window on thought, as well as communication.

Implications for teaching method include a more constructional focus from kindergarten through primary education. The stored program digital computer is arguably humanity’s greatest physical and intellectual construction. As a medium it carries out processes that were previously mental. This challenges traditional method in primary education, particularly in literacy and numeracy.

The technicity hypothesis suggests that making learning easier should be natural and beneficial. progression. I offer a species focus rather than a cultural view of primary education, eschewing philosophy. The view of technology and learning presented is novel. I looked at its pointers to teaching method, raising questions. I hope that it offers a basis for furthering constructionism.

References

Alexander, R.J. (2009) *Towards a New Primary Curriculum: a report from the Cambridge Primary Review. Part 2: The Future*. University of Cambridge Faculty of Education, Cambridge. Pdf downloaded from http://www.primaryreview.org.uk/ October 30 2009.

Anning, A. and Ring, K. (2004) *Making Sense of Children's Drawings*. Open University Press, Maidenhead Berks.

Ashby, W. Ross (1971) *An Introduction to Cybernetics*. Methuen, London.

Axelrod, R. (2006) *The Evolution of Cooperation* Revised Edition. Perseus Books Group, New York.

Barrett, L. Dunbar, R. and Lycett, J. (2002) *Human Evolutionary Psychology*. Palgrave, Basingstoke.

Bransford, J.D. Brown, A.L. and Cocking, R.R. (eds.) (2000) *How People Learn* Expanded Edition. National academy Press, Washington DC.

Burling, R. (2005) *The Talking Ape*. Oxford University Press, Oxford.

Byrne, R.W. and Whiten, A. (eds) (1988) *Machiavellian Intelligence*. Oxford Science Publications, Oxford.

Carter, R. (2000) *Mapping the Mind*. Phoenix, London.

Chang, J. (1991) *Wild Swans*. Flamingo, London.

Cosmides, L. and Tooby, J. (1992) *Cognitive adaptations for social exchange.* In The adapted mind (pp. 163–228), J. Barkow, L. Cosmides, and J. Tooby (eds.). Oxford University Press, New York.

Cox, M. (1992) *Children's Drawings*. Penguin, London.

Crick, F. and Koch, C. (1995) *Are we aware of neural activity in primary visual cortex?* Nature 375 121-123.

Damasio, A. (2006) *Descartes' Error: Emotion, reason and the human brain.* Vintage, London.

Darwin, C. (1968) *The Origin of Species*. Penguin, London.

Dawkins, R. (1989) *The Selfish Gene*. Oxford University Press, Oxford.

Dawkins, R. (1999) *The Extended Phenotype*. Oxford University Press, Oxford.

Deacon, T.W. (1997) *The Symbolic Species*. Penguin, London.

Diamond, J. (2005) *Guns, Germs and Steel*. Vintage, London

Doyle, M.P. (1986) *Computer as an Educational Medium.* MPhil thesis Manchester University

Dunbar, R. (2004a) *Grooming, Gossip and the Evolution of Language*. Faber & Faber, London.

Dunbar, R. (2004b) *The Human Story*. Faber & Faber, London.

Fuster, J.M. (2008) *The Prefrontal Cortex*. Fourth Edition, Academic Press, London.

Gallistel, C.R. Gelman, R. and Cordes, S. (2005) *The cultural and evolutionary history of real numbers.* In Evolution and Culture, S.C. Levinson (ed.), MIT Press, Cambridge, MA.

Gardner, H. (1980) *Artful Scribbles*. Jill Norman, London.

Goodenough, F.L. (1926) *Measurement of Intelligence by Drawings*. World Book Co, New York.

Gregory, R.L. (1966) *Eye and Brain*. World University Library, London.

Gupta, A.F. (2001) *Sounds and letters in English*, The Committee for Linguistics in Education, http://www.leeds.ac.uk/english/staff/afg/phonics.html (accessed May 31st 2010).

Hamilton, W.D. (1964) *The Genetical Evolution of Social Behavior.* Journal of Theoretical Biology 7(1): 1–52.

Harel, I. and Papert, S. (1991) *Constructionism.* Ablex Publishing Corporation, Norwood NJ.

Hawking, S. (ed) (1988) *God created the Integers.* Penguin, London

Heidegger, M. (1977) *The Question Concerning Technology*. Harper Torchbooks, New York NY.

Hubel, D.H. (1995) *Eye, Brain, and Vision*. Online at http://hubel.med.harvard.edu/index.html.

Ilieva, V. and Ivailov, I. (2003) *Компютърът в началното училище*. Regalia 6, Sofia.

Iversen, I.H. and Matsuzawa, T. (2001) *Establishing Line Tracing on a Touch Monitor as a Basic Drawing Skill in Chimpanzees (Pan troglodytes).* In Primate Origins of Human Cognition and Behaviour, T. Matsuzawa (ed.). Springer, London.

Jenkins J (2000) *The Phonology of English as an International Language*, Oxford University Press, Oxford

Jones, S. (1994) *The Language of Genes*. Flamingo, London.

Kellogg, R. (1969) *Analyzing Children's Art*. National Press Books, Palo Alto CA.

Kramer, S.N. (1972) *Sumerian Mythology* Revised Edition. University of Pennsylvania Press, Philadelphia PA.

Kramer, S.N. (1981) *History Begins at Sumer: Thirty-nine firsts in man's recorded history*. University of Pennsylvania Press, Philadelphia PA.

LCSI (1999) *Logo Philosophy and Implementation*. Logo Computer Systems Inc. Montreal.

Lewin, R. (1998) *Principles of Human Evolution*. Blackwell Scientific, Malden MA.

Matsuzawa, T. Biro, D. Humle, T. Inoue-Nakamura, N. Tonooka, R. and Yamakoshi, G. (2001) *Emergence of Culture in Wild Chimpanzees: Education by Master-Apprenticeship.(Pan troglodytes versus).* In Primate Origins of Human cognition and Behaviour, T. Matsuzawa (ed.). Springer-Verlag, Tokyo.

McBrearty, S. and Brooks, A.S. (2000) *The Revolution that wasn't.* Journal of Human Evolution, 39(5), pp 453-563.

McBrearty, S. (2007) *Down with the Revolution*. In Rethinking the Human Revolution, P. Mellors, K. Boyle, O. Bar-Yosef, and C. Stringer (eds.). McDonald Institute Monograph, Cambridge.

Mellors, P. Boyle, K. Bar-Yosef, O. and Stringer, C (eds) (2007) *Rethinking the Human Revolution*, McDonald Institute Monograph, Cambridge.

Miller, G.A. (1956) *The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information.* Psychological Review, 63, 81-97.

Nolen-Hoeksema, S. Fredrickson, B. Loftus, G. and Wagenaar, W. (eds.) (2009) *Atkinson & Hilgard's Introduction to Psychology* 15th edition. Cengage Learning, Stamford CT.

Nettle, D. (1999) *Linguistic Diversity.* Oxford University Press, Oxford.

Papert, S. (1980) *Mindstorms.* Harvester Press, Brighton.

Papert, S. (1994) *The Children's Machine*. Harvester Wheatsheaf, Hemel Hempstead.

Pinker, S. (1995) *The Language Instinct*. Penguin, London.

Pinker, S. (2008) *The Stuff of Thought*. Penguin Science, London.

Power, C. (1999) '*Beauty magic': the origins of art*, In R. Dunbar, C. Knight, and C. Power, (eds) The Evolution of Culture, Edinburgh University Press, Edinburgh.

Oppenheim, A.N. (1992) *Questionnaire Design, Interviewing and Attitude Measurement* New Edition. Continuum, London.

Robinson, A. (1995) *The Story of Writing*. Thames and Hudson, London.

Rose, J. (2009) *Independent Review of the Primary Curriculum: Final Report.* Online at http://publications.teachernet.gov.uk/. Department for Children, Schools and Families, London.

Striedter, G.F. (2005) *Principles of Brain Evolution*. Sinauer Associates, Sunderland MA.

Stringer, C. and Andrews, P. (2005) *The Complete World of Human Evolution*. Thames & Hudson, London.

Taylor P (2008) *Text to Speech Synthesis*, Cambridge University Press, Cambridge

Trivers, R. (2000) *Natural Selection and Social Theory*. Oxford University Press, New York.

Vigotsky, L. (1962) *Thought and Language*. The MIT Press, Cambridge MA.

Whiten, A. and Byrne, R.W. (eds.) (1997) *Machiavellian Intelligence II*. Cambridge University Press, Cambridge.


A MARVEL OF OUR TIME: THE "MEMORY" MACHINE WHICH CAN SOLVE THE MOST COMPLEX MATHEMATICAL PROBLEMS.

Fourth graders' representations of time-related dance movements

Shlomit Ofer, *Shlomit.ofer@gmail.com*
School of the Arts Of Dance, Kibbutzim College of Education, Art & Technology, Tel-Aviv, Israel

Billie Eilam, *beilam@construct.haifa.ac.il*
Faculty of Education, University of Haifa, Israel

Abstract

The study focuses on 4th graders' self-generated representations of dynamic movement events as related to time units: (a) *Synchronous time-movement events*: one or more events transpire and are completed *within* a fixed time unit. (b) *Asynchronous time-movement events*: events maintained *across* more than a single fixed time unit.

Children learned basic dance movements, trained in their performance, and developed representations for short movement sentences. These in turn, were interpreted and performed by a decipherer, who participated in the learning and training – but not in representation development. Representations were improved in light of observed feedback perceived from the decipherer's performance of movement.

Findings suggested that in the synchronous case, distinct events were utilized for representing time units. In the asynchronous case, students brought forward in their representations either the movement events or the time units. The potential of such experiences for developing representational capabilities, observation and deeper understanding and conceptualization of representing and abstract concept like "time", is discussed.

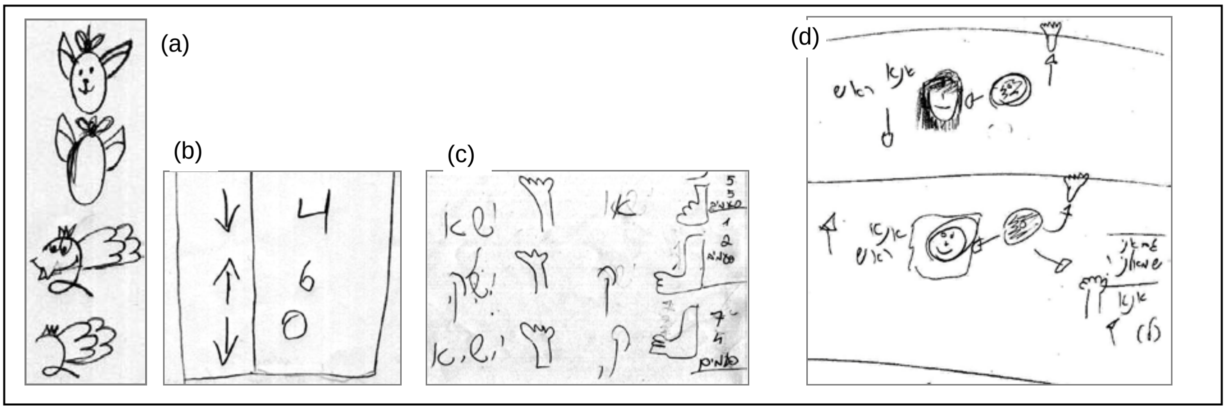


Figure 1. Examples of self-generated representations of dynamic movement events: synchronous time-movement events (a, b, c) and asynchronous time-movement events (d)

Keywords

self-generated representation; movement events; temporal aspect; elementary school children; constructionism

Introduction

"Papert's constructionism views learning as building relationships between old and new knowledge, in interactions with others, while creating artefacts of social relevance" (Kafai, 2006, p. 35). We embraced this view while designing a learning environment, that is based on young children's prior knowledge of representing and use of their own bodies, for engaging them in active learning; namely, an environment that facilitates collaboration and supports children's ideas and efforts in building new bodies of knowledge, while creating/inventing visual representations/language for communicating to others particular information regarding movement in time and space. Current studies suggest that children draw on cultural knowledge and past experiences for generating and creating visual representations. Among others, such prior knowledge may includes cultural norms of behaviour, cultural symbols and conventions, knowledge of drawing, sense of knowledge of one's own body or formally acquired knowledge regarding the use of symbolic languages for communicating information. Environments incorporating social interactions as well as explicit focused feedback, may increase children's meta-cognitive awareness of their own deliberations and the quality of learning products. We describe the designed environment and children's invented artefacts, while aiming to communicate movement events.

Theoretical Background

Dynamic movement events involve the temporal aspect, as well as the aspects of "space" – in which movement is enacted, and of "body" – its directions and the part/s that are moving (Ofer, 2001, 2009). These aspects' unique characteristics make them difficult to represent visually. We focus here on the representation of the temporal aspect only, in spite of its detachment from other aspects being artificial and frequently impossible. Another point to bear in mind is the spatial organization of symbols on the medium chosen for representing the movement.

Representing Time

Representing the entity of "time" is inherently difficult due to its abstractness and the inability to directly observe or feel it. Therefore, conceptualization of "time" evolves from individuals' daily embodied experiences, relative to experiences concerning moving and functioning in the environment. A sense of "time" is gained by comparing events or using successive iterations of specific events (Lakoff & Johnson, 1999). Hence, events' properties are projected onto "time", perceived as directional, irreversible, and continuous in nature, an entity that may be segmented and therefore also measured. Constituting a factor organizing representations into events, time may carry only a secondary role regarding representation and interpretation (Franklin & Federico, 2002)."Time" is referred to through the use of movement metaphors that map spatially conceptualized meanings onto their temporal meaning of expressions (Lakoff & Johnson, 1999). Changes occurring in time may be represented by: (a) a depicted series of objects or events at several time points, which enable inferences by comparisons, and (b) based on time equal units, fluidity and directionality, translation of "time" features into graphical spaces and the use of graphical means like time lines, or even timetables (Mackenzie-Taylor, 1999; Tufte, 1997; Tversky, 2005).

Spatial organization of symbols

Meaning of graphic-symbolic visual displays is conveyed by both the meaning of each of its symbols and all symbols spatial organization on the medium chosen for representing (e.g., a paper, a screen), which may be linear or spatial in nature. Since both are seldom random (Newcombe & Huttenlocher, 2005), interpretation is never easy or spontaneous, requiring the understanding of spatial relations among symbols as well as relations to relevant referents. The spatial display may be presented as viewed from an external point or as viewed from an internal referential point (Levinson, 2003). This consideration may affect interpretation, for example, by requiring the mental rotation of the scene (e.g., left and right) or taking an internal reference point of view. An additional difficulty evolves from the need to represent 3D phenomena on the 2D spatial display (Tufte, 1990). Frequently, symbols spatial organization is used for emphasizing a specific idea, the outcome of which are inaccurate or inconsistent representations (Taylor & Rapp, 2006; Tversky, 2005). Therefore, everyday graphic-symbolic representations of spatial phenomena are designed to accommodate a particular purpose by means of emphasizing, deleting or changing relations among its symbols and their relations to referents. Tversky and Lee (1998) suggested that among other reasons, interpretation is possible in spite of these difficulties due to individuals' awareness of the following rules: (a) continuity - if a specific description of a starting point in a certain segment is not provided, one should continue from the end point of the previous segment, and (b) a forward progression - unless specified differently. Some of the described knowledge was evidenced in children's self-generated representations as is described next.

Children’s self-generated visual representations of dynamic events

Engaging learners in processes of generating their own representations of a phenomenon is a practice anchored in the constructionist approach (Papert, 1980/1993). Studies demonstrated learners' difficulties to represent dynamic temporal events, and the relevant rich graphical solutions they come up with. Analyses of such solutions revealed learners' insight into the nature of representations and of issues related to representing "time" (*e.g.* Bamberger, 1991/1995; 2007; diSessa, Hammer, Sherin & Kolpakowski, 1991; Elkoshi, 2000; Nemirovsky & Tierny, 2001; Nemirovsky, Tierny & Wright, 1998; Sherin, 2000). For example, Sherin (2000), suggested that for representing motion, children draw on “constructive resources” like their accumulative experiences with drawing and with representation of temporal sequences (such as text), as well as the children's sensitivities to properties of figural elements (such as lines). Bamberger (2008) reported that children applied Gestalt law of proximity for grouping dynamic events (sounds) perceived as being adjacent. She (2007) described two styles of constructing meaning from musical events: “path-makers”, constructing meanings regarding the unique function of contextually situated objects or events, and “map-makers”, forming an outside, fixed reference structures, independent of the particular situation. Children's difficulties to represent motion occurring backwards in a graph were described by Nemirovsky, Tierney and Wright (1998). They suggested that in spite of individuals' natural awareness of "time" being irreversible and directional, the children's graph products exhibited their deficient understanding of the phenomenon. Another study reported that children who skipped a day of measuring a plant’s growth did not leave a space for that day on the graph, suggesting "this day did not exist". Graph’s homogeneity with respect to "time" was shown, and children's ability to bring forward those aspects they wished to emphasize (Nemirovsky & Tierney, 2001).

Objectives

We focus on children's representational expressions, modes of representing "time" units, as related to dynamic dance movement events. We examine these expressions as reflected in the symbols spatio-temporal organization, in the specific environmental design in which these artefacts have been generated.

Methods

Participants: Sixteen, medium SES, fourth grade girls, with normal spatial abilities and no movement limitations, volunteered to participate in a dance class and the study. At this age (9 to 10 years old) all have already formally encountered symbolic languages (e.g., mathematics, music notations or Hebrew), as well as constructed other resources to draw on for generating representations. Two roles were rotated among members of each group according to their wish:

(a) being a "Developer", 3-4 girls collaboratively developing scripts of dance notation, and (b) being a "Decipherer", one per developers' group, decoding the notation and performing its represented movement, thus providing "Developers" with feedback concerning the notation efficiency.

Study Context and Procedure: The conceptual framework used for developing the girls' knowledge about movement and for examining their representations included (a) the aspect of body directions (e.g., forward, right), (b) of the absolute directions of space (e.g., north, south), (c) of body parts (e.g., arm, head), and (d) the temporal aspect. This specially designed curriculum has been studied by all girls along the 29 intervention lessons. The girls acquired conceptual and practical knowledge through physically training, observing and analyzing the various aspects of movements, indicating the desired concepts. Following, the developers generated notations for a demonstrated (by teacher and video) short movement sentence (i.e., task: "put signs on the blank paper, so that your decipherer could understand the movements and perform the sentence"), and improved it in response to feedback cues perceived from the group's decipherer, who did not see the demonstrated sentence but performed it by interpreting developers' notations.

Data Collection: Two types of data were collected from all groups: (a) *video recordings*, of all lesson parts (instruction, training, notation development and accompanying discourses, decipherer's feedback and scripts improvement). These recordings exposed learners' understanding; and (b) *scripts,* learners' self-generated notations. These were scanned after each round of production and returned to learners' personal portfolios for continuous use.

Data Analysis*:* Video recordings were transcribed. The scripts, being *polysemic sign systems*, where the meaning of individual signs is driven from the consideration of the collection and combination of signs in which they are embedded, are subjective and debatable (Bertin, 1983). Therefore, our interpretation and analysis of scripts were based on and constrained by: (a) developers' recorded discourses, which revealed some of their considerations regarding the script generation; (b) knowledge of the acquired contents (conceptual and training), which constrained interpretation possibilities; (c) knowledge of the movement sentence to be represented, which provided clues regarding symbols chosen and their spatial organization; and (d) knowledge of the physical, social and cultural immediate context as well as knowledge of local and universal conventions regarding representations, symbols or their organization, that may have influenced script development.

Results

The aspect of time was exposed indirectly toward the end of the intervention, by the introduction and practice of movements involving the asynchronous enactment of several aspects. Till that point in time, all aspects of movement events were synchronous with time units. In spite of the fact that the aspect of time has not been represented explicitly, scripts concerning this synchronous time-movement events reflected developers' awareness of movement being changed with time. Later on, the representing of asynchronous time-movement events, exposed the temporal aspect as an element to be considered explicitly and directly. In both cases, expressions of the time aspect were found in: (a) modes of representing consecutive event sequences; (b) level of specification of movement elements as related to time units; (c) the selection and/or development of designated symbols for representing time; and (d) modes of symbol spatial organization on the script display.

We present first scripts representing synchronous time-movement events, transpiring ***within*** fixed time units and following those representing asynchronous events, maintained ***across*** more than a single fixed time unit.

Synchronous time-movement events, transpiring **within** a fixed time unit

Figure 2 presents a scheme of a synchronous time-movement sentence involving various events in all four movement aspects. Each column represents a single time unit.

Body directions	a	b	c	d	e	f	g
Spatial direction	A	B	C	D	E	F	G
Body parts	I	II	III	IV	V	VI	VII
Time units	1	2	3	4	5	6	7

Figure 2. A scheme of a synchronous movement sentence

A. Detailed representation of event sequences

Events were found to be represented linearly either from a point of view of an external observer or from that of an internal reference. The former is based on the common knowledge that events order in reality is linearly represented accurately by order of symbols in the script (Fig. 3).

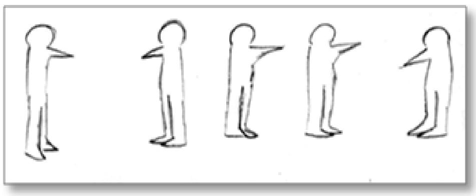


Figure 3. A linear consecutive representation of movement involving body directions "Left" and "Right"(pointed out by body parts) as viewed by an external observer

Interpretation of a script represented from an internal referential point is sensitive to the represented content (Fig. 4, body directions) and is based, in addition to the aforementioned factors, on the symbols chosen and their spatial organization rather than on external references. Sequence interpretation demands increased efforts as compared with the former case.

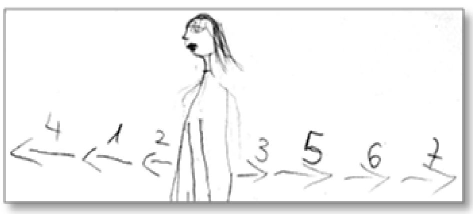
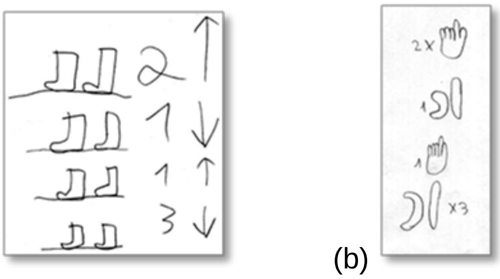


Figure 4. A linear referential representation of movement involving body directions "Forward" and "Backward", represented by arrows relating to the central figure; the order is indicated by numbers.

B. Abridged ("formulated") representation of event sequences

Whereas in the linear examples above, movement events are detailed for each time unit – even if repeated, some of the developers created "abridged" representation-type. This type is characterized by the grouping of repeated same events into a single symbol, indicating by numbers (or X and number) the amount of times they have to be performed (Fig. 5a and b).



(a) (b)

Figure 5. The "abridged" representation-type: (a) boots - representing steps to be performed, numbers – amount of steps, and arrows – directions of steps; (b) numbers showing the amount of times a movement has to be performed (1, 2X, X3), sequence order is represented top-down.

In the "abridged" representation-type, the girls perceived movement sequence comprehensively and were able to represent it schematically, while ignoring the representation of distinct time units. Lack of explicit direct relation between movement and time units may require the breaking up of representation into component events along time, for interpretation and movement enactment.

C. Selection and development of designated symbols for representing time

Different time unit representations were developed in the synchronous time-movement event framework. Each event coincides with a single time unit, represented as iconic, conventional, alpha-bet, verbal initials, and more. For example, the X symbol in Fig. 5b, constitutes a designated symbol that indirectly represents the number of time units along which the same event is enacted. Other examples (Fig. 6, 7) present arrows or a word designated for representing events enacted simultaneously – yielding several events in a single time unit in each line.

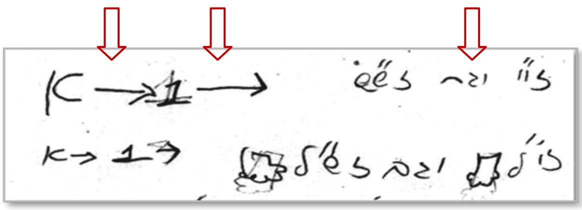


Figure 6. Arrows and a verbal symbol relate between events that are performed simultaneously (indicated by our added arrows).

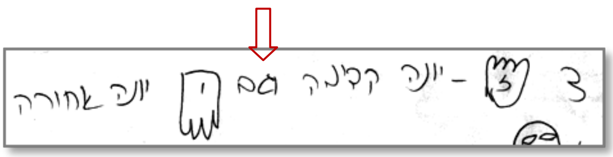


Figure7. A verbal symbol (see our added arrow) representing "simultaneous events".

When developers were required to represent movements in which more than a single aspect was enacted within a single time unit, the temporal aspect was represented creatively by either a table format or a compound format, with a small designated space between each event (Fig. 8a and b). In the table format, each dance aspect is represented in a vertical column of its own, in a manner that positions the two elements of these aspects (arrows and numbers) on the same line to denote their simultaneous enactment. In the compound format, different elements representing dance aspects are combined within each time unit (i.e., body directions, are represented by faces looking away from or at an observer, and spatial directions represented by disguised numbers), and sequential order is represented top-down.

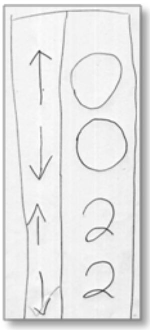
(a)  (b) 

Figure 8. (a) a table format; (b) a compound format

D. Modes of symbol spatial organization on the script display.

Time units were represented by parsing graphical units – either by specific organizations or by both organization and additives like lines. For example, in Fig 9a and b, each horizontal line contain a representation of simultaneous events and order of consecutive events is represented top-down by consecutive lines, resulting in a more complex format than the table one (Fig 8a).

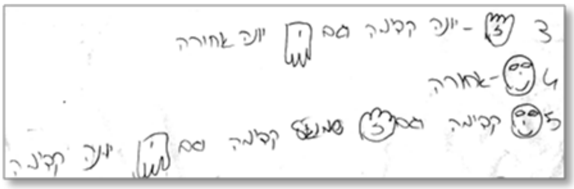
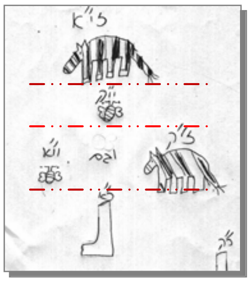
(a)  (b) 

Figure 9 (a) and (b). simultaneous events represented in horizontal lines

Asynchronous movement events, maintained **across** more than a single fixed time unit

In the synchronous movement events, time constituted a factor organizing representations into events. Things became much more complicated when movement events and time were asynchronous (see table on Fig. 10). Figures 11a and b present scripts representing the sentence presented in Fig 10; they represent two variations of parsing: (a) focusing on movement events (Fig 11a) in which the 4 "time" units (4 elements in the row) serve as a background, and the arm movement is brought forward – transpiring along two of these units. Time units are represented by parsing the sequential elements by small space; (b), time units are represented by horizontal lines (Fig 11b). The 4 distinct "time" units are kept as a main organizing device, representing twice the movement in each of the relevant units (3rd and 4th), indicating movement starting and ending points. Hence, solutions created for representing the movement across these time units are different, reflecting different considerations.

Arm movement			I	
Spatial direction	A	B	C	D
Body direction	a	b	c	d
Time units	1	2	3	4

Figure 10. Table representing a-synchronic movement; the arm moves across two "time" units

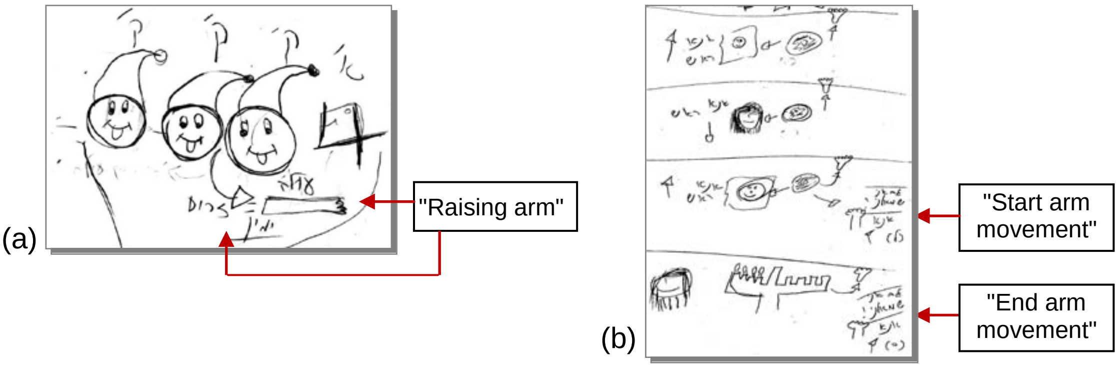


Figure 11. Two scripts representing arm movement across two "time" units: (a) bringing forward the continues movement; (b) bringing forward the separated time units

These differences in representing may be similar to Bamberger's notion (2007) of "path makers" who accounted for the event and "map makers" who accounted for the external "time" units references.

Concluding Remarks

Developers' spatio-temporal creative self-generated representations of movement reflected their perception and conceptualization of the aspect of time in dynamic movement event. Mostly, when possible, they used distinct events to represent time units. However, while representing movements transpiring across time units, either a continuous representation of the movement across the involved time units was used, or time units were emphasized, with a repeated representation of the movement in each of them. Many sources could be identified for developers' inventive ideas as reflected in their representations (choices of symbols and their spatial organization): use of universal symbols (e.g., arrows, ordinal numbers or numbers denoting amounts), use of local cultural elements (e.g., numbers and letters in costumes), use of conventions and elements of formally acquired symbolic languages (e.g., designating sequences order by numbers, by consecutive linear elements or top-down lines, using numbers, letters, words, verbal initials, abridged representation-type), etc. However, retrieved knowledge was adapted and transformed for the construction of new knowledge regarding representing dynamic movement events. For example, since participants were asked to avoid detailed verbal descriptions, manipulations on the verbal language were performed, using its basic symbols (e.g., letters, initials, verbal hints) for representing concrete aspect of an event (e.g., a direction) or an abstract idea like linking all events that transpire simultaneously.

Frequently, symbols received their meanings from the context they were embedded in. Arrows could represent a particular direction or represent the abstract idea that certain events are enacted simultaneously, or the use of numbers for representing events order or amounts.

Developers' interactions were social, affective and cognitive in nature, all influencing the resulting artefact. Previous representations, proven to be successful in communicating accurately the movement information to the decipherer, were sometimes neglected in favour of desires for new ideas, for showing off, for preferring and presenting one's ideas rather than another one's. Communication of information was frequently challenged by the desire to play and test decipherer's abilities. Criticism of other's ideas while praising one's own, constituted common components of the girl's discourse. However, many of their interactions were constructive in nature, increasing the girls' awareness of their own thinking (e.g., reasons for using certain symbols or the use of their own body for generating a solution to a problem), initiating imaging (e.g., taking another person's point of view), examining the quality of their products (e.g., flaws in the representations that may be misinterpreted or of ways to represent problematic referents) or of what representing means. Decipherers' feedback promoted the girls ability to observe a performance of the represented movement and compare it to the movement image held in their heads, identify specific differences and improve their representations by correcting relevant parts only for eliminating the identified differences. In this sense, the girls improved and greatly refined their representational abilities and their understanding of representations as communicational tools.

Hence, our environmental design enabled the girls to experience learning in which new bodies of knowledge have been developed, based on prior knowledge and linked to it. This new knowledge structure enabled the girls to represent complex information, while increasing and deepening their understanding of symbols, representing and representations.

References

Bamberger, J. (2008) *Noting time*. Colloquium presented at Swarthmore College Department of Music and Dance, September 18.

Bamberger, J. (2007) *Restructuring conceptual intuitions through invented notations*. In Notational knowledge, E. Teubal, J. Dockrell and L. Tolchinsky (eds), Sense Publishers, Rotterdam, pp. 81-112.

Bamberger, J. (1991/1995) *The mind behind the musical ear: How children develop musical intelligence*. Harvard University, Cambridge, Mass.

Bertin, J. (1983) *Semiology of graphics*. University of Wisconsin Press, Madison, WI. (Translated by W. Berg).

diSessa, A .A., Hammer, D., Sherin, B. and Kolpakowski, T. (1991) *Inventing graphing: Meta-representational expertise in children*. Journal of Mathematical Behavior, 10, 117-160.

Elkoshi, R. (2000). *Children's graphic notation as a representation of musical perception*. Ph.D. thesis, Tel-Aviv University, Tel-Aviv (in Hebrew).

Franklin, N. and Federico, T. (2002) *Organization of temporal situations*. In Spatial language: Cognitive and computational perspectives, K. R. Coventry and P. Olivier (eds), Kluwer Academic Publishers, Dordrecht, The Netherlands, chap. 6, pp. 103-120.

Kafai, Y. B. (2006) *Constructionism*. In The Cambridge handbook of the learning sciences, R. K. Sawyer (ed), Cambridge University Press, New York, NY, Chap. 3, pp. 35-46.

Lakoff, G. and Johnson, M. (1999) *Philosophy in the flesh: The embodied mind and its challenge to western thought*. Basic Books, New York.

Levinson, S. C. (2003) *Space in language and cognition: Explorations in cognitive diversity*. Cambridge University Press, Cambridge.

Mackenzie-Taylor, M. (1999) *Developing design through dialog: Transport tables and graphs*. In Visual information for everyday use: Design and research perspectives, H. J. G. Zwaga, T. Boersema and H. C. M. Hoonhoot, (eds), Taylor & Francis Group, London, Chap. 14, pp. 177-193.

Nemirovsky, R. and Tierney, C. (2001) *Children creating ways to represent changing situations: On the development of homogeneous spaces*. Educational Studies in Mathematics, 45(1-3), 67-102.

Nemirovsky, R., Tierney, C. and Wright, T. (1998) *Body motion and graphing*. Cognition and Instruction, 16(2) 119-172.

Newcombe, N. S., and Huttenlocher, J. (2005) *Development of spatial cognition*. In Handbook of child psychology (6th ed.), D. Kuhn and R .S. Siegler (eds), John Wiley and Sons, Chap.17, pp.734-776.

Ofer, S. (2001) *Movement literacy: Development of the concept and its implications for curriculum*. M.A. thesis, Faculty of Education, University of Haifa (in Hebrew).

Ofer, S. (2009) *Development of symbolic language to represent movement among fourth graders*. Ph.D. thesis, University of Haifa (in Hebrew).

Papert, S. (1980/1993). *Mindstorm* (2nd ed.). Basic Books, New York.

Sherin, B. L. (2000) *How students invent representations of motion*. Journal of Mathematical Behavior, 19s(4), 399-441.

Taylor, H. A. and Rapp, D. N. (2006) *Updating human spatial memory*. In Animal Spatial Cognition: Comparative, Neural, and Computational Approaches, M. F. Brown, and R. G. Cook (eds), [On-line], Available: www.pigeon.psy.tufts.edu/asc/

Tufte, E. R. (1990) *Envisioning information*. Graphic Press. Cheshire, Conn.

Tufte, E. R. (1997) *Visual explanations: Images and quantities, evidence and narrative*. Graphic Press, Cheshire, Conn.

Tversky, B. (2005) *Functional significance of visuospatial representations*. In The Cambridge handbook of visuospatial thinking, P. Shah and A. Miyake (eds.), Cambridge University Press, Cambridge, pp. 1-34.

Tversky, B., & Lee, P. U. (1998) *How space structures language*. In Spatial cognition: An interdisciplinary approach to representation and processing of spatial knowledge, C. Freksa, C. Habel, and K. F. Wender (eds), Springer-Verlag, Berlin, pp.157-175.

Modelling without Mathematics – Using JlinkIt modelling tool in educational settings

Fábio Ferrentini Sampaio, *ffs@nce.ufrj.br,* **Márcio Reis Teixeira,** *marcioreisteixeira@gmail.com* and **Marcos da Fonseca Elia,** *melia@nce.ufrj.br*
Programa de Pós-Graduação em Informática – Federal University of Rio de Janeiro

Description

For many years our research group on IT in Education (www.nce.ufrj.br/ginape) has been working with modelling in education in Brazilian schools. As part of this research we have developed a computer modelling tool called JlinkIt that allows any one (students and common people) to construct and simulate causal dynamic models without the necessity of knowing the mathematics that are normally used in analytical models (mainly calculus and differential equations). This modelling tool was developed in Java and runs in any browser (http://www.nce.ufrj.br/ginape/jlinkit/executa_jlinkit.htm). It also has a stand-alone version that can be run on computers not connected to the Internet. The software is free and can be downloaded from its website (http://www.nce.ufrj.br/ginape/jlinkit/download.htm).

The models constructed with JlinkIt are the type of cause-effect models. The software has a direct manipulation interface and it uses only two different building blocks (variables and links) to develop the models.

The graphical language of JlinkIt is based on the idea of causal-loop diagrams and it permits the users to construct, simulate and follow time graphs of different variables while the model is running over time. Also the software uses a semi-quantitative mathematics to relate the variables of a certain problem (Bliss & Ogborn, 1989).

The software permits the construction of a wide range of problems related to the syllabus of primary and secondary schools such as those in the categories of linear, exponential and oscillatory problems, in a disciplinary or interdisciplinary approach.

At the moment we are preparing a course based on Moodle LMS to introduce Brazilian teachers to the subject. The course will be launched in September 2010.

Keywords

Computer modelling in education, causal loop diagrams, literacy for computer modelling.

References

BLISS, J. & OGBORN, J (1989). Tools for Exploratory Learning. A Research Programme. Journal of Computer Assisted Learning, 5:37-50.

Young Children and Powerful Ideas: Snapshots of Creative Learning by Constructing from Early Childhood Education Settings

Chrystalla Papademetri, *C.Papademetri@euc.ac.cy*
Dept of Education Sciences, European University Cyprus

Andrea Eliadou, Maria Hadjiioannou, Chrystalla Loulli, Chrysanthi Meletiou, Maria Santis, Maria Siakalli, Maria Socratous
Early Childhood Educators, Cyprus

Abstract

Papert, in 'Mindstorms', unfolded a fascinating learning experience involving 'Children, Computers and Powerful Ideas'. This poster aspires to offer a fascinating learning experience involving 'Young children and Powerful Ideas'; a learning experience though, that doesn't involve computers. Through a collection of snapshots presenting young children (three to six-year-olds), involved in tasks of creative learning by constructing, from real early childhood education settings, this poster demonstrates how Papert's powerful ideas can be applied without the use of computers. It also demonstrates how when Papert is applied in the classroom little children gain access to big ideas.

This poster presents snapshots of young children involved in tasks designed and implemented by seven early childhood educators that were involved in in-service and pre-service teacher training courses on developing mathematical activities for young children. The main aims of the courses that were organised and facilitated by the first author of this poster, were to train the participants to deal with their practice as teacher-researchers and to support the participants in developing scientifically justified mathematical activities through a process of designing, implementing, reflecting and revising. At the beginning of the courses, the teachers were exposed through workshops to the constructionism tradition and Papert's conviction that mathematics education should aim not at teaching children mathematics but at teaching children how to think as mathematicians. Creativity, problem solving and mathematical literacy (presented as a combination of experiences, skills, attitudes, epistemological awareness and conceptual understanding) were some of the courses key themes.

The tasks presented in this poster, that were designed and implemented by the seven teachers, involve problem solving-based activities where children (a) had to construct shapes by composing other shapes, (b) create flowers with the use of different shapes angles and (c) construct triangles, quadrilaterals and circles with simple every day materials and objects. Through the snapshots presented in the poster, with the use of children's constructions and representations, photographs and conversations, one can detect the results of applying Papert in early childhood education settings. Thus, through the poster, Papert's conviction in (a) the importance of learning-by making and thinking-as-constructing, (b) providing children with objects-to-think-with, (c) accepting the validity of multiple ways of knowing and thinking and (d) acknowledging that 'when knowledge can be broken into mind-size bites it is more communicable, more assimilable, more simply constructable', comes to life through little children's involvement in creative learning.

Keywords

Young children, construction task, teacher training, creative learning

Demystifying Constructionism

Wallace Feurzeig, *feurzeig@bbn.com*
Department of Educational Technology, BBN Technologies, Cambridge, MA, USA

Abstract

Many teachers and educational researchers are unclear about the meaning of constructionism or its implications for learning and instruction. This paper discusses a number of frequently aired educational dichotomies in order to situate and clarify the constructionist perspective. An elaboration of the contrary positions expressed by proponents of both sides of these dichotomies offers insights into the sense and purpose of constructionist ideas and approaches. It argues for the unique and valuable benefits of constructionism as a powerful learning paradigm.

There are several false dichotomies in education: *tradition versus reform, structure versus freedom, knowledge versus creativity, instruction versus construction*. They are false because they are often viewed as diametrically opposed adversarial positions. It's as though if you are on one side you can't possibly concede anything to the other. These strongly biased one-sided oppositions remind me of Oliver Selfridge's characterization of much of the thinking in current artificial intelligence research—that things must either be true or false—as *binary heresy*.

This either-or kind of thinking can lead otherwise intelligent people to say unintelligent things. For example, a few years ago a highly respected Boston economist, assessing the effects of computer technology in education, wrote: "Computers have proved to be ineffective in education. How do we know that? Because we've had computers in schools for several years now, and schools are still terrible!" An obvious response: books must *really* be ineffective, because we've had books in schools for much longer, and schools are *still* terrible!

Constructionists want to build a critical mass of citizens who reject false and misleading educational dichotomies, who support instead the creation of learning environments that integrate the constructive ideas on both sides of tradition and reform, structure and freedom, knowledge and creativity, instruction and construction. More broadly, constructionists seek to develop a culture of learning. We would like to develop a national thirst for promoting intellectual curiosity and creativity. This paper suggests an ambitious thrust toward that end: an intensive and sustained political and marketing campaign to build a groundswell of support for developing a rich variety of learning opportunities, both formal and informal, infused with constructionist ideas and activities. That is an awesome challenge. Large-scale educational change may not be possible during our lifetime, but there are schools and learning places where constructionist ideas and culture can flourish even today.

Keywords (style: Keywords)

Constructionism, Learning, Educational Dichotomies, Logo, Mathematics

A False Dichotomy: “Tradition” versus “Reform”

Two often-impassioned views of mathematics education underly the math wars issue, those held by the traditionalist camp and those held by the reformist camp. The former hold that school mathematics should focus on acquiring knowledge of basic number operations and calculation skills. The latter hold that mathematics education should focus on the development of critical thinking and problem-solving skills. *This is a false dichotomy!* Children need to be able to do both kinds of things. Of course they should have computational competence. But they should also acquire competence in mathematical ways of thinking and their application to things that matter in their lives as individuals and as citizens. We need to help children develop and employ basic reasoning skills while they are developing basic computational skills. These goals are not inherently opposed. We need to go *forward* to basics—not *back*—by moving toward a more comprehensive and powerful set of mathematical skills, all of which can be fostered by appropriate Logo programming activities.

If the sole goal of school math is to help kids acquire the ability to do sums, long division, and square root calculations, it might be argued that, with the introduction of calculators and computers, school math is no longer necessary.

This suggestion is analogous to the one in Jonathan Swift’s brilliant 1729 essay “*A Modest Proposal: For Preventing the Children of Poor People in Ireland from Being a Burden to Their Parents or Country …”* Swift suggests that the Irish might ease their economic troubles and relieve their population problems by cooking and eating the children of the poor. In the same satiric fashion, though somewhat more benignly, we suggest that children should be removed from school math classes. Now that we have computers to do calculation, we no longer need school math. Children could be sent home sooner to do things that they find more enjoyable (and that are perhaps more intellectually beneficial than adding or dividing long strings of numbers.)

I’m being facetious. Of course kids need to learn calculation, but we can help them in new and better ways. In many schools, six or seven years are dedicated to teaching a superficial understanding of numbers and arithmetic operations. At the end of this protracted period, a large percentage of students fail to achieve even modest competence. What a terrible and unnecessary waste of time—all those years focused on calculation and the kids can’t successfully emulate a calculator! Today, appropriate computer programming activities can make an enormous difference in the ease, enjoyableness, and effectiveness of learning number ideas and acquiring number manipulation skills.

Part of the problem is that the standard arithmetic algorithms are taught as cookbook recipes, disconnected both from the world of real mathematics and the world of kids. Boring, repetitive drills stamp out any flicker of curiosity and generate an indelible perception of mathematics as the realm of lengthy, ritualistic calculations. Our present method of teaching the subject conveys to our students the unmistakable (and lasting) impression that mathematics is both difficult and deadly dull. Rarely in the course of thirteen years of pre-college education are students given to understand that mathematics can be *fun.* Of course, *you* know that it *can*. Those of us in the Logo programming community have very specific ideas about how children’s work with Logo can be used to motivate their development of mathematical ways of thinking while transforming their dislike of school math into fondness for the real thing.

Each of us has favorite areas for Logo-based interventions. Let me briefly share two of mine. One is the early introduction of combinatorics. This is the seminal area of mathematics that treats arrangement and ordering problems, the study of different ways of “jumbling” things. It’s not the process of enumerating a given set of objects—what children learn as counting, but a natural and powerful extension of that—inductive enumeration—appropriately called *counting without counting*. Students are introduced to mathematically rich “counting” problems such as matching, merging, and sorting that involve the exploration and investigation of different ways of representing objects and operations. Working with such problems gives concrete meaning to powerful ideas such as equivalence, uniqueness, and completeness.

Another favorite area, an elementary introduction to transfinite mathematics, builds on kids' love of big numbers and their fascination with the idea of infinity, starting from the realization that there is no largest integer. This branch of mathematics extends the concept of counting in yet another way, leading to the development of powerful ideas such as incommensurability, countability, and orders of infinity. These topics are a great deal more interesting to kids than the standard school fare. The basic ideas and proofs are accessible fairly early and, through their exploration, children are exposed to mathematically rich ways of thinking.

Could topics like these be part of a *new* math curriculum in the constructionist context of student programming projects? Of course, a constructionist curriculum is not enough. None of this is possible without teachers who know the underlying mathematics, who are able to learn from their students, and who are comfortable about sometimes relinquishing control and ceding it to their students. Finally, it requires a trusting school culture and a political and social environment that support the challenges and risks of project-based learning, a willingness to trade the predictability of lock-step, scope-and-sequence, lesson-plan structures for the development of kids who are mathematically more literate *and much happier* in their math classes.

Another False Dichotomy: Structure versus Freedom

Students can greatly benefit from guidance and direction on assigned instructional tasks on the way to acquiring new knowledge and skills. But they also need opportunities for designing and constructing artifacts that test and extend their understanding. That is the heart of the constructionist learning perspective. Constructionism focuses on making the creation and sharing of new knowledge a primary goal. Without structure there is no freedom. And without freedom there is no foundation for development of intellectual growth and creative expression. This is true not only for education, but also for research and practice at professional levels in all fields.

The music of Johann Sebastian Bach exemplifies the powerful synergy between structure and freedom. Bach's Art of Fugue, for example, is one of the most emotionally charged works in all music, a transcendent model of creative invention. Yet, its overall organization and the interrelations among its musical structures and devices can be described (post facto) as a sequence of formal mathematical algorithms. The beautiful drawings of M.C. Escher, the most mathematically inspired graphic artist of our time, show the same creative integration of structure and freedom. Where else is such patently obvious draftsmanship technique, a readily trainable skill, transformed so eloquently into great art before our eyes?

An educational philosophy that often extols freedom while abhorring structure is that segment of the progressive school movement exemplified by A.S. Neill's Summerhill School in England. Summerhill was noted for its philosophy that children learn best with freedom from "coercion." All lessons were optional, and pupils were free to choose what to do with their time. The school was founded with the belief that "the function of a child is to live his own life—not the life that his anxious parents think he should live, not a life according to the purpose of an educator who thinks he knows best." For some troubled kids, the unstructured school environment can provide a nurturing, and perhaps corrective, experience. However, for kids who don't already come with their own learning agenda, it can be an ineffective intellectual experience.

Another example of freedom without structure is from Logo: One of the middle-school students who was introduced to Logo in one of our early teaching experiments, came to us and said "Now I know how to program. *Tell me what to program*." Students need to be motivated, to have their own purposes and drives. They need to learn how to make their vague ideas clear and precise. Work with programming can greatly aid students to express, debug, and reformulate their thinking. Along the way, they often need to acquire content knowledge that they can draw upon to concretize their mental constructs and support their constructions. This goes against the instinctive tendency of some constructivist colleagues for whom the very idea of an explicit instructional agenda is anathema.

Work with Logo can also exhibit *structure without freedom*. I have seen curriculum materials designed to teach Logo programming that are so tightly prescribed and circumscribed, that they bring to mind the "do it by the numbers, dot-to-dot" paper drawing exercises for young children. In one case, in a Logo teaching sequence in a New York City school, students were told which commands to enter for a given procedure, line-by-line, from To all the way to End. They were then instructed to run "*their*" procedure, after which they were told, "Look what you've discovered!"

A Misleading Dichotomy: Instruction versus Construction

The distinction between knowledge instruction and knowledge construction serves to highlight the important contrast Seymour Papert draws between *instructionism* and *constructionism* (Papert and Harel, 1991). This dichotomy exposes the profound shortcomings of the all-too-common practice of school instruction that provides students virtually no opportunity for knowledge construction, e.g., for designing and building artifacts that test and extend their knowledge. Constructionism is not a rejection of instruction. Learning requires both instruction and construction. They are mutually supportive learning components, intimately joined throughout the learning process. Instruction is often both a useful precursor and a useful successor to construction.

Michelangelo, one of the greatest constructionists, learned the basic skills of his art by being exposed early to stonecutters and masons, and by apprenticing as a painter to the great artist Ghirlandao. He was initially assigned mundane tasks such as copying the works of his master. Even Picasso, the personification of originality and invention in art, trained himself by copying old masters at the Louvre, like generations of painters before and since. One learns to become oneself by the discipline of appropriating the knowledge of others as a prelude to forging new paths. Once a student has learned to replicate and assimilate the work of the expert he is better positioned to finding his own way.

Instructionism is the extreme case of instruction without construction, i.e., without enabling students to make the knowledge their own, to "*own*" it and move forward to take on new expressive challenges. Unfortunately, instructionism all too often focuses on *schooling* as opposed to *learning* (yet another dichotomy!) Constructionism—making knowledge construction a primary goal of instruction—is what those of us in the Logo community strive to achieve in our teaching and our own work. Along the way we learn from our mistakes. That's why debugging is so valuable. We try, as Thelonius Monk so profoundly put it, to make "the *right* mistakes."

There is, however, a distortion of constructionism that rejects instruction altogether and extols learning without teaching. For some educators, including many instructional technologists, "teacher-proof learning" seemed an attractive alternative to poor teaching and weak instructional methods. And, indeed, we admire autodidacts—"self-taught" highly accomplished persons—as exemplars of ostensibly instructionless learning. But the notion that most individuals can dispense with instruction as a key component of learning is wishful thinking: It simply doesn't compute! Learning requires shared interactions between (and among) children and their teachers. Sometimes, during these interactions, the children teach and the teachers learn. Programming experiences can provide a powerful mediating role.

Another False Dichotomy: Revolutionary Change versus Incremental Reform

Why is this a *false* dichotomy—surely, revolution is the opposite of reform? Aren't they polarities? For a long time, the Logo movement has been (somewhat simplistically) characterized as consisting of two warring camps—the reformers and the revolutionaries. Those in the BBN Logo group were labeled reformers, because we believed that Logo would ultimately make significant inroads toward school reform. The MIT Logo group, led by Seymour, called for a fundamental restructuring of education, a political and social (though non-violent) revolution.

I share the revolutionary perspective—that *is* the goal we should work toward. But I have somewhat different views about what should be done to advance that goal. When will constructionism become a standard component of educational practice in schools? Not significantly in our time, and certainly not in most schools as we know them. But let's be realistic. Schools are going to be around for the foreseeable future. De-schooling will not occur during our lives! Should we abandon working with schools? If not, what should we do to foster constructionist learning? Like many of you, I believe in continuing to work within the current school world with tools like Logo that support constructionism. Large-scale school change may not be possible during our lifetime, but there are schools and learning places where the constructionist philosophy and culture can flourish today.

We'd like to help children make a serious commitment to becoming good at something that they have to work at, that takes time, and that requires a significant investment of thinking. We'd like them to become practitioners, actively engaged in some discipline or craft, and sharing their work with others (Feurzeig, 1988). We'd like to bring the culture of practitioners into the classroom. Toward this end, we can promote initiatives for developing apprentice learning, not only in the arts but also in science and mathematics, under the guidance of practicing professionals. The effort could draw upon the potentially enormous resource of retired mathematicians, scientists, engineers, and teachers in these areas, who have the time and interest to participate, who love their subject, and who can engage effectively with kids in fostering shared learning activities and experiences.

It would mirror the pre-college music education model in which young students spend significant time at a conservatory—perhaps several hours on Saturdays over a period of years—not only learning to play an instrument but also participating in a comprehensive education program—performing in choral and instrumental ensembles, studying music theory, and engaging in composition—on the path to becoming complete musicians.

Constructionists seek to develop a culture of learning. We would like to initiate a major effort toward changing the negative image of learning and creating a groundswell of support for revolutionary educational change. We seek to develop a national thirst for promoting intellectual curiosity and learning, particularly in mathematics and science where the image problem is most notable (Goldenberg, 2007). We want to build a critical mass of citizens who reject false and misleading educational dichotomies, who support instead the creation of learning environments that integrate the constructive ideas on both sides of tradition and reform, structure and freedom, instruction and construction.

That is an awesome challenge. We need to champion a radical transformation of current perceptions of the nature and worth of learning—a transformation that will build a powerfully supported national demand for the development of a rich variety of learning opportunities, both formal and informal. The endeavor would require an intensive and sustained political and marketing campaign. It would employ the full range of broadband communications media, engaging as advocates leading national figures—celebrities with wide popular appeal and influence, including movie, music, sports, and television stars, as well as other nationally known icons. It would also require the participation of skilled media artists, working in close collaboration with constructionist researchers and educators to create new and compelling learning activities.

Ambitious thrusts like this pave the way to significant educational progress, perhaps not in our time, but for our children or grandchildren. Increasing numbers of Americans are concerned about the poor quality of early education. The concern is not only about the failure of early mathematics and science education. It is also about the failure to support the preparation of informed and intelligent citizens. We seek to foster the development of a generation of young people who are thoughtful about their lives, their intellectual development, and their social responsibilities, and who understand the sense and purpose of the learning ideas we hold dear. Constructionist learning activities can have a powerful impact on these developments and contribute greatly to advancing these goals.

References

Feurzeig, W. (1988) *Discipline-Oriented Learning Technology. Apprentice Tools: Students as Practitioners.* Book chapter in R.S. Nickerson (Ed.), Technology and Education in the Year 2020. Hillsdale, NJ: Lawrence Erlbaum Associates.

Goldenberg, P. (2007) Selling Intellectual Curiosity in Mathematics and Science: Increasing the Demand for Learning. Personal Correspondence.

Papert, S. and I. Harel. (1991) *Constructionism.* Ablex Publishing, Norwood, N.J.

Enhancing Science Inquiry Via Sound and Music

Wallace Feurzeig, *feurzeig@bbn.com*

Department of Educational Technology, BBN Technologies, Cambridge, MA, USA

Eric Neumann, *eneumann@alum.mit.edu*
Department of Educational Technology, BBN Technologies, Cambridge, MA, USA

Abstract

We describe the design of a software environment and a set of laboratory tools to enhance and enliven science learning through hands-on explorations, investigations, and student research projects in the domain of musical acoustics. Music is a powerfully riveting mode of expression and communication for high school students. A typical student's world is inundated with sounds: iPods, TV, CDs, radio, bands, rock, rap, funk, salsa. Students are receptive to music of all kinds. Many students perform and create their own. This contrasts with science, which fails to hold the interest of all but a few high school students. Approximately 80% of US students take first-year biology or earth science. The proportion drops to about 40% for chemistry and declines dramatically to under 10% for physics. Indeed, in areas where students are less concerned about college entry science requirements, these numbers are even lower.

Computer technology can provide a uniquely valuable way to build a bridge between the frequently disparate student interest worlds of music and science and to provide an alternate path to the serious study of computer science itself. Many kinds of natural, mechanical, and musical sounds that students find interesting are accessible to computers—heart sounds, birdsong, speech, thunder, ocean waves, polyrhythms. The rich sound-generation capabilities made possible by integrating computer-controlled synthesizers with software facilities for exploration and experiment can foster a lively introduction to scientific thinking, by creating an environment that enables students to generate and investigate an incredible variety of sounds and music through information processing techniques. Constructionist learning activities in the domain of sound have an extraordinary potential for providing a compelling pathway into inquiry, which is as an integral part of science education.

Computer visualization techniques have proven valuable for enhancing student interest and involvement in science investigations. Computer sonification techniques have received much less attention. Yet complex behaviors can often be better understood by associating sound with displays. The addition of sound brings an extra dimension to our experience and understanding of real world phenomena. Data originating outside the acoustic domain—such as seismic and meteorological data—can be transformed to audible range. Students are thereby able to *hear* natural data produced by phenomena such as the sunspot cycle and *listen* to long-term changes in barometric pressure or global warming patterns. We are certain that a significantly larger fraction of students will be drawn to science through introductory activities that exploit sound and music to enliven inquiry. We show vignettes of student work to illustrate the potential of such activities to augment and heighten student engagement in science exploration and inquiry.

Keywords

Science Education, Computer Sonification, Musical Acoustics, Auditory DisplaysEnhancing Science Inquiry Via Sound and Music

Enhancing Science Inquiry via Sound and Music

The sound-generation and acoustic analysis capabilities of computers offer very specific benefits for science education. The simplest is the sheer appeal of sound for students. In recent years, sonification methods have been effectively employed in science research (Shinn-Cunningham, 1998; Kramer, 1997; Kramer, 1994; Bargar, 1994; Scaletti & Craig, 1990). Thus far, however, their use in science education is virtually non-existent. Yet computer sonification, enabling science activities with sounds and music, can have the same power as computer visualization to capture students' interest and sustain their engagement. Sonic representations of phenomena can greatly aid students whose quantitative or spatial organization is less developed or who have difficulty acquiring concepts taught solely with textual or visual representations.

The use of sound is particularly valuable for investigating oscillatory systems, which are fundamental constituents of many physical and biological processes—and also of music. Since the concepts of sound and oscillation are nearly inseparable at a fundamental level, the use of sound is natural for studying such systems. Further, the physical product of a perfect linear oscillator is a pure tone, and the addition of pure tones leads naturally to an exploration of what makes a sound pleasing or musical. This introduces an aesthetic dimension to the science course, one that is often lacking for many students.

By coupling oscillators to MIDI devices such as synthesizers we can produce a rich variety of sounds. From heartbeats and cricket chirps to earthquakes and the periodic explosions of Old Faithful, the natural world offers many examples of oscillating systems that have nearly discontinuous behaviors. Phenomena like these are best modeled by percussive sounds—typically short-lived sounds that “go off” whenever their controlling oscillator reaches a pre-defined phase angle. When generated by a single oscillator, the sounds are repetitive and rhythmically uninteresting. When one couples oscillators together however, even in very simple ways, the resulting rhythms are often complex and fascinating—musically as well as mathematically and physically (Fergemann, 1993; Schneck, 1992.) Systems of coupled oscillators are natural candidates for sonification. By making their complex dynamic behaviors palpable as rhythmic patterns, sonification brings a new experiential dimension to the study of periodic and quasi-periodic systems and provides a powerful impetus to scientific inquiry.

An oscillation has three characteristics—amplitude, frequency, and phase. The concept of phase, which is surely the least understood of these, can be brought to life through the study of oscillatory systems. Sonic representation can strongly complement visual representation of oscillatory system behaviors. Our ears are sensitive to features that our eyes cannot perceive. Although the period of oscillation of light is incredibly short—on the order of 10^{-15} seconds—our visual apparatus is unable to perceive frequencies much in excess of 10 Hz. However, our ears are sensitive to frequencies three orders of magnitude greater—they enable us to experience complex processes that involve the superposition of multiple frequencies and that occur in less than a millisecond. Our ears are better equipped to extract meaning from these phenomena than our eyes are for deciphering an equivalent burst of light.

In addition to responsiveness across a large frequency range, our ears are extremely sensitive to phase shifts. For example, changes in the relative phases of signals presented to our ears are readily perceived as alterations in the direction of the source. Thus, the remarkable ability of humans to determine the provenance of a sound can be used to introduce the concept of phase.

In the course of sound-enhanced investigations of physical, biological, and musical processes, students’ understanding of the associated mathematical processes and concepts can be greatly deepened by sonic representations. Mathematical processes such as function generation, iteration, recursion, exponentiation, and algorithmic operation can be sonified as well as visualized. Through the use of audio displays, mathematical behaviors can be heard as well as seen. Such “in vivo” aural experiences can help students to gain new mathematical insights and advance their understanding of mathematical ideas they previously regarded as inaccessible or meaningless.

To foster mathematical and scientific inquiry, we are developing *SoundLab*, a computer system designed to enable students to create, explore, display, sonify, and analyze sounds of many kinds. SoundLab is a visual modeling environment for introducing the study of oscillators and oscillatory systems, through experiences coupled to sound. The system hardware includes a computer and a Musical Instrument Digital Interface (MIDI) connected digital sample synthesizer. We have demonstrated preliminary versions of SoundLab programs to science educators. Many have been excited by the realization that computer capabilities for representing biological and physical processes sonically as well as visually have significant potential for enlivening science exploration and inquiry activities.

The graphic tools under development in SoundLab enable students to “see” the sounds that they hear, decomposed into their fundamental frequency modes via fast Fourier transform methods. Using a computer equipped with an audio pick-up and appropriate software, students are able to sample and synthesize sounds of all kinds. The coincidence of aural and visual sensory inputs helps students learn the physical meaning of frequency, amplitude, phase, resonance, and linear superposition. They will be gaining a natural introduction to wave behavior and the fundamental concepts of acoustics. There can be few more powerful ways of learning about waves than to experience them both visually and aurally. Coupling sound to visual models of wave behavior provides a powerful experimental environment for engaging students in science explorations involving the analysis and synthesis of complex sounds.

SoundLab enables the development of new sonification-based techniques expressly designed for science education. Students can explore the structure of familiar sounds and identify their characteristic features. SoundLab tools enable users to hear sounds over a wide range of frequencies and through a variety of filters. Students can separate, view, and hear each component of a sound—even if its source is initially outside the audible range. Modeling labs in SoundLab enable students to generate sounds, investigate how sound interacts with the environment, and explore the effect of timbre, envelope, and pitch in music.

It is not enough, of course, to provide students with powerful tools for producing, analyzing, and manipulating sounds. They need to be guided in the use of the tools, presented with interesting and carefully sequenced challenges, and brought to the point where they can carry out their own investigations, and perhaps develop their own applications. Throughout, emphasis must be placed on the *process* that produces the sounds, whether these are data-driven or algorithmically generated. With appropriate preparation and guidance, students can take on problems and projects in various science and music domains, such as the following.

Biology

- generating waveforms from recordings of animal calls and songs
- analyzing the sounds of dolphins, bats, wolves, birds, monkeys, . . .
- generating bird songs that mimic those observed in nature
- modeling the behavior of coupled oscillators in biological systems (circadian rhythms)
- modeling the dynamics of the electrical control system of the human heart
- investigating the psychoacoustics of perception (what we “hear” when we hear)

Mathematics

- seeing sounds and hearing graphs: representing amplitude and frequency across time
- sonifying and studying the behavior of functions, sequences, and limiting processes
- investigating the harmony in trigonometry: sines, cosines, and phase

Physics

- creating audio displays of waves and oscillators (wave mechanics)
- masking unpleasant sounds (noise pollution) by filtering or by *adding* sound
- using audio signals to measure range and position of objects (sonar echolocation)
- investigating the behavior of physical oscillators, e.g. spring-mass systems
- using sound to analyze time-series data from earthquakes, weather, and sunspots
- generating racing car sounds and adding Doppler effects
- analyzing the waveform of a ping pong ball's impacts to determine its speed

Musical Acoustics

- creating “pleasant” and “unpleasant” music and characterizing key differences
- creating music sequences with specified rhythmic or harmonic patterns
- generating musically interesting polyrhythms algorithmically
- turning humpback whale songs into Fats Waller or Bono tunes
- creating complex tones by steadily increasing the rate of discrete rhythm patterns
- generating rhythmic tone sequences in music via coupled oscillators
- building virtual music instruments using filters and transfer functions

These investigative activities and projects are designed to provide students with the basic scientific knowledge and inquiry skills to prepare them to construct their own sound and music artifacts. The SoundLab software is designed to support their investigations and constructions. A brief description of some of the modeling tools that comprise SoundLab follows.

SampleView. This tool enables students to capture, display, and hear sound samples of all kinds—simple tones, music, biological sounds, data originating from non-acoustic sources that have been transformed to the acoustic domain. During data acquisition the sound stream will be viewable on a computer screen. The program has facilities for successive magnification of all or selected parts of the sample material, to enable one to see and hear the fine structure of the sound. Students are able to explore and investigate live and recorded sounds as well as non-acoustic time series data, animal sounds (dolphins, mocking birds, whales, wolves, bats), earthquake and underground detonation data, songs, musical instruments, and speech.

Harmonic Synthesizer. This tool enables students to construct sounds out of harmonic components that can vary in frequency, amplitude, and phase. A varied set of waveforms—square waves, sawtooths, chords, and vowel sounds—is provided. MIDI can be used to input the harmonics plus custom wave forms. Students are able to see and hear the results of varying these parameters, either in the sound components or in the composite sound. The system supports multiple linked representations of the waveforms. Student activities include creating or modifying one or more harmonic components so as to compose a sound that duplicates a given sound. Use of the program can enhance the students’ understanding of phase relationships among harmonics.

MidiPhasor. This tool enables students to design and control the operation of systems of coupled oscillators. Students can create complex rhythms and polyrhythms—close emulations of classical, jazz, and rock forms—as well as highly varied original rhythmic patterns. At the same time they can gain both aural and visual experience of some fundamental concepts of number theory, such as primality and incommensurability, and experience the utility of mathematics in aiding their rhythmic designs. Systems of coupled oscillators are fundamental components, not only in acoustic phenomena, but in virtually all natural systems. The program supports a rhythmic representation of the phase relationships between two oscillators. Users can vary the rates of each oscillator and the coupling strength between them. The oscillators can be coupled so as to produce complex and sometimes chaotic rhythmic patterns that are presented both graphically and aurally. The program outputs can be used with a synthesizer to generate a rich variety of timbral effects, producing compelling sound patterns and music.

Dynamics Construction Kit. This tool provides students an environment for building their own aural and visual dynamical systems. Systems can be constructed out of five kinds of objects—Inputs, Links, Clocks, Responders, and Composites. Inputs include keyboard and MIDI sound sources. Links have parameters such as propagation delay time and refractory period. Clocks (oscillators) have parameters such as period, phase, perturbation function, and coupling strength. Responders include MIDI objects (tones, and tonal sequences), graphics objects (music notation and events, e.g. pitch and duration as a function of channel), and number objects. Composites are higher-level objects constructed from the five basic objects and other composites. As a program runs, the currently active objects are highlighted and their links animated to show the passage of signal and control data. The other SoundLab tools complement the use of this facility in enabling students to develop sound and music generation systems of their own design.

Some of these capabilities are accessible via widely available software tools such as Audacity. However. their interfaces are not specifically tailored to facilitate use by pre-college students. SoundLab tools were crafted to exploit the enormous appeal that electronic music and sound generation have for high school students, in a way that provides educationally productive science learning experiences. SoundLab projects offer users bountiful opportunities for experiencing the spirit and sense of science inquiry. Some students will use SoundLab's sound processing capabilities in an indiscriminate fashion without learning any science along the way. Most, however, will find their work with SoundLab instructive as well as enjoyable.

The following scenarios illustrate the flavor of student interactions with SoundLab. The first shows the work of a tenth-grade biology student; the second, that of a ninth-grade general science student; the third, that of a scientifically sophisticated high school physics student.

1. Analysis and Synthesis of Mockingbird Song

Rachel wanted to investigate the song of one of her favorite birds, the mockingbird. She knew that these birds produce remarkable calls with many variations, and that they often imitate the songs of other birds. There was a highly vocal mockingbird near her house. She recorded the bird's song on cassette. After she had collected a few hours of song, she took the recordings to her biology class. She reviewed her tape to identify and mark the diverse song segments she found most interesting. Using the SoundLab SampleView module, she entered eleven different sound segments. Viewing the waveforms (Figure 1), she observed definite clusters as groups of waves. She selected those regions and played them back. She identified the groups as the various chirps of the mockingbird's song. Some sounded distinctly different than others.

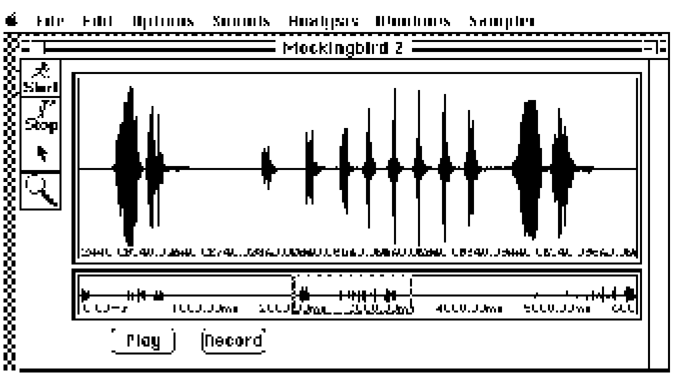


Figure 1. A set of varying mockingbird chirps, viewed across 1 second

She could not easily discern those differences visually from the sample, so she collected a few examples and grouped them by ear. She then used the SoundLab SoundScape tool to investigate the “composition” of these chirps. She displayed the pattern of one chirp (Figure 2).

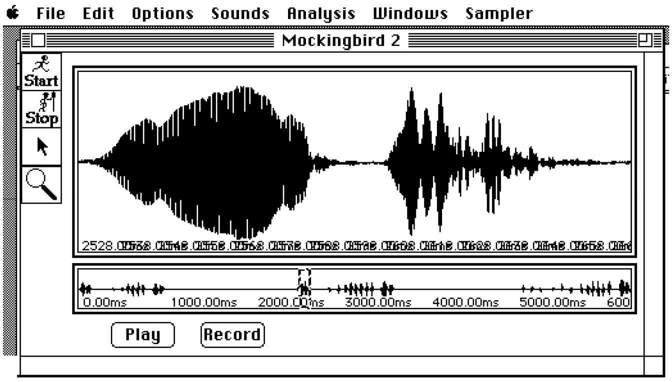


Figure 2. Close-up of the left-most mockingbird chirp from above

She used the SoundLab sonification tool to investigate the sounds associated with the graph’s regions. As she slid a cursor along the x-axis, a simple tone rose in pitch; for every doubling of the position of x, the pitch increased in a way that sounded like the chromatic scale. When she placed the cursor over the ridge pattern, the tone produced was near in pitch to that of the sample. The pattern clearly contained pitch information, but she wondered if the distinctive sound of the chirp could be clearly shown on the graph of the waveform. She used the zoom tool for a closer inspection of the waveform pattern (Figure 3).

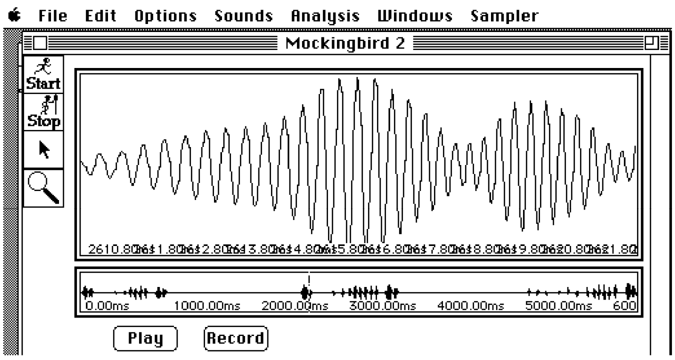


Figure 3. Highly enlarged view of the above chirp, illustrating the oscillating waveform

She now saw that it had a finer set of features; there were actually two ridges whose heights waver across a short time interval. She used the sonification tool to play multiple tones whose pitches followed the tops of both ridges. The new computer-generated sound was a great deal closer to the chirp—its attack and duration were similar. She repeated this waveform analysis and sound synthesis for similar and dissimilar chirps and annotated each chirp’s characteristics (e.g., its pattern of single or multiple pitches, and their attack and duration).

Rachel used her results to piece together the various chirps within a song segment (Figure 4). She saw the regular modulated waveform. She observed that the chirps formed patterns that were related to the various calls she was able to distinguish with her ears. She was amazed by the complexity of each call and the fact that mockingbirds can produce such a rich variety of songs.

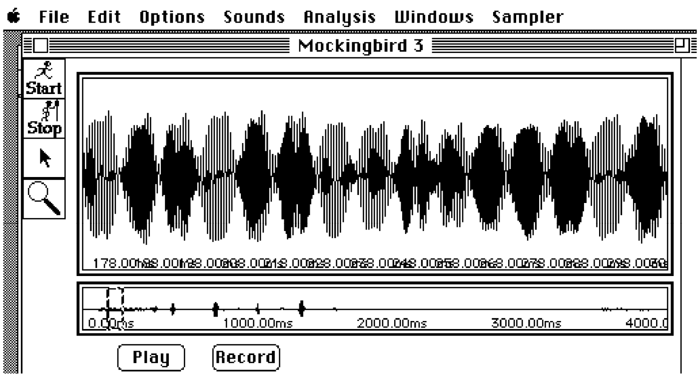


Figure 4. Close-up of a mockingbird warble

She compared these songs with other sampled bird songs obtained from the Cornell Ornithology Laboratory. After comparing the different samples by ear, she identified some of the mockingbird songs with the bird songs they appear to mock. She attempted to find the differences between the original and imitation songs first by ear, then by using the SoundLab analysis tools. After writing up her results for her biology project report, she wondered if she could elicit calls from other birds in the wild by using the mockingbird calls. She thought she might produce her own set of calls, put them on tape and play them back to the birds in the wild. She wondered if she could 'mock' the mockingbird song.

2. Using SoundLab to Create Music

Why should kids who are interested in drumming, rap, and rhythmic composition come to care about notions like phase and ratio? Rashad, a ninth-grade high school student, was turned off by math and science, but he was greatly interested in making and performing music. His music idols included M.C. Hammer, Michael Jackson, and Prince, and he was up with the latest rap hits. He would love to create his own electronic piece, but he was unfamiliar with the relevant technology. His science teacher, Mr. Owens, suggested that Rashad use SoundLab to put together a piece from Rashad's own music material for his science project. Rashad was excited by this idea.

He explained to Mr. Owens that he wanted to lay down some rhythm tracks and put music and various other sounds on top of that. He wanted to make a strongly rhythmic piece with sounds that repeat "though not always exactly." Mr. Owens asked Rashad what sort of sounds he wanted to include. Rashad replied, "I want some clapping, some shouts, and some wild big sounds!" His teacher lent Rashad a cassette recorder to collect and capture these sounds.

Rashad recorded some sounds directly from the street: car horns, sirens, laughter. He listened to some rap singing on his boom box and recorded the bits he liked. After a week of recording he asked to use SoundLab. Mr. Owens started SoundLab and activated the SampleView module. Rashad connected the tape output to the computer's sound port. He soon realized that he had several hours of recordings—a lot more than could be stored. So he reviewed the tape to select the segments he liked best. He used the record button in the SampleView module to input his selected segments. His sounds were sent to the SoundLab sampler and mapped to the various keys on the synthesizer. He heard each of his sounds by playing a corresponding note on the keyboard.

He commanded the computer to play the sounds on the synthesizer and arranged them sequentially using the MIDI control facility. After a few attempts at arrangement, he wanted to get rhythmic sound patterns that repeated indefinitely. Mr. Owens suggested using a MIDI periodic oscillator (a *clock*) in the SoundLab Dynamic Construction Kit to control the repetitive sequence of sound playback. Rashad created a clock and added the sequence of sounds around its circumference (Figure 5.) He started the clock and immediately heard the sound pattern.

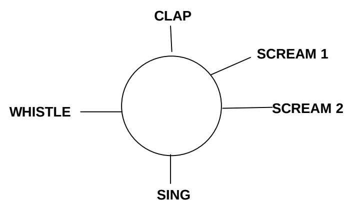


Figure 5. Rashad's Clock

“This is wild!” he exclaimed, “Can I move the sounds? How do I get the sounds to play a little sooner or later?” “You can grab the labeled sound symbols at any position around the clock and move them to a new position or phase.” “So the phase tells what time the sounds should be played every time the clock goes around!” He was enthusiastic. “Can I get another clock with different sound rhythms to play along with this one at the same time?” “Sure, as many clocks as you like” replied Mr. Owens, “but think about how you can get them to work together to do what you want.”

Rashad created another clock, placed some other sounds around it and started it playing. More sounds were being played, but the overall rhythmic pattern was not changed, and was not more intricate as he would like. He saw that the two clocks were running in synchrony. To change this, he used the mouse to grab the arm of one of the clocks to stop it from playing, and released it a second later. The rhythmic pattern shifted somewhat. f

“It's still the same beat!” he complained, “I want something more hip!” Mr. Owens tried to get Rashad to think about the problem differently. “Why don't you try changing the speed of one clock relative to the other?” He showed Rashad how to alter the clock speeds and phase parameters via a dialog box. The current rate value was 1.0. Rashad sped it up. He tried a value of 10. “That sounds crazy, it's too fast!”

Next he tried a speed of 3 and found the result more pleasing. After trying other values, he found that basic rhythms were produced by small integer values, but he could make more interesting ones using decimal values like 2.5 or 1.33. At times he needed to play with the hundredth decimal place to get the beat to sound “just right”. He also tried changing the speed of the other clock and began noticing similarities between rhythms whose clock values seemed to be quite different.

“ That's funny, I get the same pattern if the clock speeds are 1.0 and 2.5 as when they are 2.0 and 5.0, only it's twice as fast.” “Why don't you try comparing the ratio of the clock speeds for similar patterns?” “What do you mean by ratio?” Mr. Owens explained the concept of ratio.

Rashad began to understand the invariant rhythmic relationship between the two pairs of clocks. He wondered why this relationship is expressed by a mathematical operation, division. Putting that aside for the moment, he tried next to set the rhythmic patterns for his entire piece. He played some of his extended composition to Mr. Owens, who saw that Rashad was now using four clocks. “You know, if you connect the clocks to each other so they affect each other's phases, the patterns might get more interesting, even with just two clocks!” Rashad began coupling the clocks. This allowed him to generate more “fascinating rhythms” while raising new math and science questions about how to design “wilder” and more-pleasing rhythmic patterns.

In the constructive context of music synthesis, mathematics and science concepts such as variable, period, frequency, amplitude, phase, oscillation, phase shift, periodicity, steady state, and perturbation took on concrete meaning and became real for Rashad.

Through such work, the sense and value of science inquiry, even at a naive beginner's level, becomes apparent to "non-science" students like Rashad and has the potential to set the stage for further engagement and intellectual development.

3. Generating Music with Chaos in Coupled Oscillators

Other students, scientifically more advanced than Rashid, have used SoundLab to investigate the application of mathematics and physics for generating rhythmically complex and aesthetically interesting music. Two sophisticated high school juniors conducted intensive independent research employing the SoundLab MidiPhasor and Dynamic Construction Kit tools. (Fergemann, 1993; Schneck, 1992).

The work of one of these students is described next. David started his project by studying the prior research on dynamic models of physiological rhythms in organ systems. This body of work is comprehensively summarized in Glass and Mackey (1988). A paradigmatic example of these dynamic mathematical models is that describing the electrical behavior of the heart's cardiac rhythms under normal as well as pathological conditions such as arrythymias. Systems like this can exhibit their oscillatory behaviors as rhythmic patterns, ranging from periodic to chaotic.

David was interested in developing dynamic mathematical models of oscillatory systems with such behaviors in SoundLab in order to express their often-complex rhythmic patterns sonically. He sought to show that a sonic representation could significantly augment one's insight into the behavior of such systems, particularly when they exhibit chaotic dynamics. Sonification can provide more information than a graph alone can give, making it easier to recognize periodicity, quasi-periodicity, and chaos, while creating interesting musical effects at the same time.

Oscillatory systems with regular perturbations can be depicted as in Figure 6, which shows a rotating oscillator that is perturbed at regular intervals t by a stimulus at a fixed distance b.

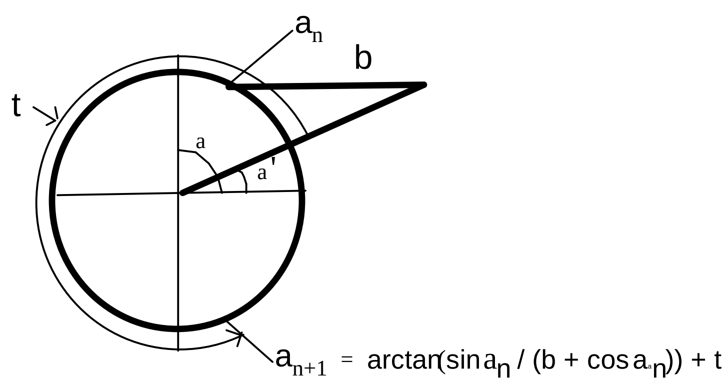


Figure 6. Graphical representation of a single perturbation and successive iterations

The result of a single perturbation is given by $a' = \arctan(\sin a / (b + \cos a))$ where a is the original phase of the oscillator and a' is the new phase. Successive iterations can be represented by the iterated function $a_{n+1} = \arctan(\sin a_n / (b + \cos a_n)) + t$. Each time the oscillator passes a = 0 an event, such as a heartbeat, occurs. This dynamic model lends itself well to sonification.

To create sound sequences, notes are placed at many points around the circle. A note is played whenever the oscillator rotates past it. Careful placement of the notes can generate interesting rhythmic and melodic sequences. With some values of t and b, the oscillator produces rhythms that are chaotic and form short, unstable patterns that change unpredictably over time. Subtle differences in t and b can cause a sequence of notes to be played, skipped, or repeated, making the changes in dynamics clearer and often producing sequences that are musically compelling.

David used a Yamaha SY-99 synthesizer with SoundLab to investigate the use of sound for representing the dynamics of coupled oscillator systems. He created two rotating clocks with MidiPhasor. The first clock behaved like the oscillator described above. It had notes of different pitches and volumes placed around it. The synthesizer played these notes whenever the clock rotated past them. The second clock rotated at a different constant speed determined by t and provided the stimuli to the first clock by causing a perturbation in it after each revolution. For any combination of t and b, the behavior of the oscillator system could be periodic, quasi-periodic, or chaotic. David showed how these behaviors can be sonified to produce a number of musical effects and how these effects can help clarify the differences between periodicity and mathematical chaos. He used the mathematlical appliation *Mathematica* to create bifurcation graphs showing the characteristic behaviors of the clocks for a wide range of values of b and t.

David concluded his project by conducting an extensive analysis of possible system behaviors. He showed that periodicity occurs when the oscillators "phase lock," i.e., when after M cycles of the first clock and N cycles of the second, the oscillators' behavior repeats exactly. The simplest form of periodicity is when b > 2. Here, the oscillators show phase locking when t > 0.5. Since M = 1 in these cases, a sequence of notes repeats the same way at each perturbation. When b is slightly greater than 1, however, the periodicity begins to change into chaos, and the sounds lose their regular rhythms though the music can still be controlled. Some regions of the circle tend to be repeated less often than others. Notes placed in these regions can add variety to the music. Short, frequently changing rhythmic patterns often form, offering greater variety in the melody and rhythms than phase-locking. At b = 1 some interesting rhythmic patterns occur. Different phase-locking zones form over a range of t values. These regions overlap in ways that produce very complex rhythms. In some cases the sound created resembles the irregular or changing meter frequently found in twentieth-century music.

These scenarios suggest how high school students at different levels of sophistication can use computer sonification tools in science explorations and investigations. There have been a few studies exploring the use of sonification by high school students (Upson, 2002; Stanionis, 1992.) These short-term interventions have not led to adoptions by schools. Although college courses employing sonification tools are offered at some institutions, (Berklee, 2010a; Berklee, 2010b; Sydney, 2010), few if any such courses are currently offered in U.S. high schools.

The educational potential of this unique approach to introducing science inquiry at pre-college levels has yet to be realized. The additional sensory contribution made possible by the use of sound as an investigatory medium is an innovation waiting to happen. Thus far, in the words of the bard, "The eye of man hath not heard, the ear of man hath not seen . . ." (Shakespeare, circa 1595).

References

Bargar, R. (1994) *Pattern and reference in auditory display,* in Auditory Display: Sonification, Audification and Auditory Interfaces, G. Kramer, ed. Santa Fe Institute Studies in the Sciences of Complexity, Proc. Vol. XVIII. Reading, MA: Addison Wesley, 161-165.

Berklee (2010a). *Course EP223. Modular functions and signal flow*. Berklee College of Music, Boston, MA.

Berklee (2010b). *Course EP225. Control systems in electronic production*. Berklee College of Music, Boston, MA.

Fergemann, D. (1993). *Generating music with chaos in coupled oscillators.* SoundLab Technical Report 2, BBN Technologies, Cambridge, MA, 1-11.

Glass, L. & Mackey, M.C. (1988). *From Clocks to Chaos: The Rhythm of Life.* Princeton University Press, Princeton, NJ.

Kramer, G. (1994). *Some organizing principles for representing data with sound.* In G. Kramer (ed.) Auditory display: Sonification, audification and auditory interfaces Proceedings of the First International Conference on Auditory Display. Santa Fe Institute Studies in the Sciences of Complexity, Proc. Vol. XVIII, Reading, MA: Addison Wesley, pp. 185-221.

Kramer, G. et al, (1997). *Sonification report: status of the field and research agenda.* Prepared for the National Science Foundation by members of the International Community for Auditory Display. http://www.icad.org/websiteV2.0/References/nsf.html

Scaletti, C. & Craig, A. B. (1990). *Using sound to extract meaning from complex data.* In E. J. Farrel (ed.), Extracting meaning from complex data: processing, display, interaction 1259, 147-153.

Schneck, R. (1992). *Dynamic construction set: sound as an educational tool.* SoundLab Technical Report1, BBN Technologies, Cambridge, MA. 1-8.

Shakespeare, W. (circa 1595). *A Midsummer Night's Dream. Act IV, Scene 1.* Stratford on Avon, England.

Shinn-Cunningham, B. G. (1998). *Applications of virtual auditory displays.* Proceedings, 20th Annual International Conference of the IEEE Engineering in Medicine & Biology Society, 1105-1108.

Stanionis, V. A. (1992). *Teaching physics using the MIDI and computer music.* AAPT Summer Meeting, American Association of Physics Teachers, Vol. 22 No. 2.

Sydney, (2010). *DECO1013: Sound Design and Sonification.* University of Sydney, Sydney, Australia

Upson, R. (2002). *Educational sonification exercises: pathways for mathematics and musical achievement.* Proc. 2002 International Conference on Auditory Display, Kyoto, Japan, 1-6.

Building understanding: Geometry for Design at the Community College of Philadelphia

Dorothy M. French
Dept. of Mathematics, Community College of Philadelphia

Abstract

Technology is radically altering the landscape, no more so than in the field of mathematics education. In particular, electronic learning resources lend themselves easily to a constructionist approach to teaching and learning mathematics. Along with traditional hands-on, activity-based methods, technology can help create productive environments for studying geometry. This paper compares outcomes for community college students studying two different college math courses, Geometry for Design and Intermediate Algebra. Students enrolled in visual design curricula -- CAD (computer assisted design), construction management, fine/applied art and interior design -- can study Geometry for Design, a technology-rich course that uses model-making, interaction and hands-on activities, along with traditional methods.

Geometry for Design students' achievement was compared with the achievement of students in a traditional lecture-based college algebra course that generally does not use classroom-based technology or hands-on activities. Geometry for Design students seemed to have a higher rate of success than students studying Intermediate Algebra. Geometry for Design students also seemed to hold more strongly positive opinions about hands-on methods for learning mathematics. Geometry for Design students were more likely to strongly agree that they had acquired strategies for thinking about problem-solving and that they had developed a better understanding of how math relates to the real world.

Figure 1. Students constructing models of Platonic Solids

Keywords
constructionism design education geometry mathematics technology

Introduction

This paper compares the achievement of community college students in a technology-rich geometry course with the achievement of students in a traditional lecture-based algebra course. Geometry for Design (Math 137) is an interdisciplinary, technology–enhanced geometry course that is taken primarily by students in the College's Construction Technology, Computer Assisted Design (CAD), Art and Interior Design programs. Intermediate Algebra (Math 118) is a traditional lecture-based course with no designated classroom technology component. Many students register for Intermediate Algebra because it is a prerequisite for most curricula at the Community College of Philadelphia. Outcomes for students in Geometry for Design courses were compared with outcomes for students in Intermediate Algebra courses. Geometry for Design students' attitudes towards learning geometry from a constructionist perspective were explored by questionnaires. Geometry for Design students seemed to have a higher rate of success than students studying Intermediate Algebra. Geometry for Design students also seemed to hold more strongly positive opinions about hands-on methods for learning mathematics. Geometry for Design students were much more likely to strongly agree that they had acquired strategies for thinking about problem-solving and that they had developed a better understanding of how math relates to the real world.

Geometry for Design at Community College of Philadelphia

More than half a century ago, in “Le Modulor”, Charles Edouard Jeanneret wrote:

Passée la porte des miracles …Mathematics is the majestic structure conceived by man to grant him comprehension of the universe. It holds both the absolute and the infinite, the understandable and the forever elusive. It has walls before which one may pace up and down without result; sometimes there is a door: one opens it --- enters--- one is in another realm, the realm of the gods, the room which holds the key to the great systems. These doors are the doors of the miracles.

More than half a century later, technological marvels that we take for granted—computers, air travel, ubiquitous cell-phones, medical imaging technology, and perhaps most interesting to Le Corbusier, tall buildings, to name but a few—would have been impossible to realize without mathematics. Excitingly, new technologies utilizing principles of visual design are emerging, creating jobs in architecture, animation, construction, engineering, applied art and interior design. These rewarding careers require strong mathematical skills and the ability to frame and solve quantitative problems. A solid foundation in geometry is essential for the development of problem-solving ability, and Geometry for Design is an important course for Community College of Philadelphia students in visual design and construction curricula.

The unfortunate reality is that, at many colleges today, students often begin their study of visual design without a strong knowledge of geometry. This impedes progress in many ways: students may be unable to make simple area calculations or conceptualize nets of three-dimensional solids. To address this problem, faculty from Community College of Philadelphia's Mathematics, Art, Architecture and Construction Technology departments created a new math course, Geometry for Design. Course development was supported by funding from the U. S. National Science Foundation and Geometry for Design was first offered 1998.

One characteristic of geometry is its immediacy. In contrast to other branches of mathematical study that rely heavily on abstraction, the study of geometry can begin with commonly observed and experienced phenomena from the physical world. Accordingly, the pedagogy of experience-based learning was adopted, and accordingly lecture lesson-plans include hands-on activities and computer applications and investigations. Geometry for Design combines a sound preparation in the basic concepts and techniques of plane and solid geometry with a thorough exploration of geometry's concrete applications in architecture, construction, art and design. Michael Serra's, "Discovering geometry: an investigative approach v. 4" is the assigned text and Geometer's Sketchpad software is used in the computer classrooms. Topics include: traditional straightedge-and-compass and computer-based construction methods; properties of triangles, polygons and circles; plane transformations, symmetry and tessellations of two-dimensional figures; area; 3-dimensional polyhedra; volume; the Pythagorean Theorem; ratio and proportion and similarity. Some study of formal proofs is included. Topics from art and architecture are studied, including perspective drawing, the design of arches and domes and an exploration of the Golden Mean and its relevance to architecture, art and science. Students create portfolios of manual and computer-based drawings and build 3-dimensional models. Classes meet in a computer-equipped classroom, and Geometer's Sketchpad is used extensively to explore ideas, make conjectures, literally draw conclusions and formulate proofs. Students also use Internet web sites that relate to geometry, architecture, construction, art and design.

Geometry for Design incorporates "hands-on" group activities to get students involved in "building knowledge structures". For example, the course begins with an ice-breaker "Building Blocks" puzzle that has multiple solutions requiring good visualization skills. Students are also introduced to the idea of working together in groups, which are used throughout the course.

Early on in the course students are exposed to the concept of inductive thinking and formula development. Given a set of regular polygons, the task is to try to find the number of diagonals of an 18-gon that has a somewhat daunting-looking diagram resembling "String Art" gone mad (Seymour and Beardslee, p. 42).

In the process of constructing the four points of concurrency of a triangle (incenter, circumcenter, orthocenter, and centroid) students practise measuring , observing and drawing using traditional compass-and- straightedge techniques. They also master the modern reincarnations of these time-honoured skills by recreating these constructions in the computer-based environment of Geometer's Sketchpad. These skills, evinced by the illustrations and drawing exercises in Lawlor's "Sacred Geometry " , are indispensable for students of design who will use geometry in the real world.

Much hands-on work is done on the properties of circles. In particular, students learn about pi as the ratio of circumference to diameter of a circle by rolling out the circumference of a wheeled object (e.g. a trolley or a bicycle tire) , measuring its diameter and thus discovering the approximate value of pi. This activity relates well to real- world applications such as oodometers and pedometers. The derivation of the area of a circle is explored by juxtaposing 12 sectors into a "rectangle" and positing its area as (radius) x (½ the circumference).

The study of isometry transformations -- translation, reflection and rotation-- is an especially fertile area for the use of hands-on activities using computer software. Students can relate to these topics, as they are used extensively in computer assisted design and graphic design. Students exercise their creativity by building their own tessellations using translation and rotation. Web sites and videos introduce students to geometrical aspects of the work of the artist M.C. Escher.

While exploring Pythagoras's Theorem, students use strings of holiday lights to investigate the condition for a right-angled triangle .Three students are designated as the "vertices" while a fourth is charged to count the number of "units" on each side and verify the right angle. This activity is a variation of the knotted ropes techniques used by Ancient Egyptians. Four different proofs of Pythagoras's Theorem are presented, including Bhaskara's proof, Leonardo da Vinci's hexagonal proof and President Garfield's trapezoidal proof. Students use a hands-on Geometer's Sketchpad computer activity to explore a "juxtaposing areas" proof of the Theorem.

Figure 2. Students exploring Pythagoras's Theorem

Rectangular solids and their volumes are explored and formulae are derived using small 1cm3 blocks. A set of small hollow translucent "Power Solids" can be filled with water to demonstrate that the volume of a cone is one third of the volume of a cylinder of the same radius and height. The formula for the volume of a pyramid is explored in similar fashion and students derive the formula for the volume of a sphere by comparing its two hemispheres' volume to the volume of an associated cylinder. This demonstration can be a bit messy with liquid being poured from solid to solid but students enjoy this hands-on activity and it is well worth doing. Students explore and discover the Euler formula V+ F – E = 2 (Seymour and Beardslee, p. 33) using cocktail sticks and mini-marshmallows (fig. 1) to make models of polyhedra. Using nets, students construct and creatively decorate cardboard models of the five Platonic Solids, including the famous "cheese-a-hedron".

Figure 3. Students' polyhedral models

Students access on-line web sites and they state that they really enjoy using artist/mathematician George Hart's web site georgehart.com to learn about complex polyhedra in the context of art/architecture history.

When studying similarity, students use traditional compass and straightedge constructions and computer-generated activities to investigate similarity criteria (AAA, SSS, SAS, ASA) for triangles. Students employ similar triangles to indirectly measure the height of a tall object (e.g. a ceiling sprinkler head) using a plane mirror on the ground between the sprinkler and the observer.

Finally, students explore the Golden Mean and its significance for geometry, design, biology, art and architecture. Golden rectangles are constructed using a compass and straightedge and students observe real word objects that are configured in golden rectangular proportions, such as credit cards and ID cards, packs of cigarettes, many containers of different products and the ubiquitous iPod.

To assess student achievement in the Geometry for Design course, seven computational homework quizzes, a midterm exam and a final exam, two short written papers and a portfolio of computer-generated drawings are used. Grades for the course are awarded as follows: A = 90-100%, B = 80-89% , C= 70-79 % , D= 60-69%, F= 0 - 59%.

Intermediate Algebra at Community College of Philadelphia

Most students at Community College of Philadelphia study Intermediate Algebra at some time during their college career. Its popularity is a result of compulsion, not inclination, as it is a prerequisite for admission to most degree and certificate programs, especially those in the fields of engineering, science and technology. As is the case with other college mathematics courses, Intermediate Algebra functions as a filter or "gatekeeper" course. Topics include the Real number system, systems of linear equations and inequalities, polynomials, rational expressions, radical expressions, and quadratic equations. Most instructors use a customized intermediate algebra text and students are usually taught in a traditional classroom using a "chalk-and talk" lecture-based format. Generally, students do not have access to classroom computing facilities, although there is a Math Learning Lab which provides some computer support. There is no departmental consensus on the use of technology in the classroom: some instructors encourage the use of calculators and computers, while others discourage and even prohibit their use. Meanwhile, students have embraced the twenty-first century digital age and want to use graphics calculators, computers and even cell phones for graphing and computation.

Assessment of student achievement in the Intermediate Algebra courses taught by the author is measured by seven computational homework quizzes, midterm and final exams. Grades for the course are awarded as follows: A = 90-100%, B = 80-89% , C= 70-79 % , D= 60-69%, F= 0 - 59%.

Comparing outcomes and opinions

To compare the achievement of students in a technology-rich geometry course with the achievement of students in a traditional lecture-based algebra course, records of students' final grades were analyzed. Data was obtained from five sections of Geometry for Design and nine sections of Intermediate Algebra taught by the author from Fall 2007 through Fall 2009. If "success" is earning an A, B or C grade in a course, the Geometry for Design students: chance of success was 95/135 ≈ 70 %, while for Intermediate Algebra students , the chance of success was 173/288 ≈ 60 % and so there was a small but statistically significant difference between these proportions. Geometry for Design students had a slightly better chance of earning an A, B or C than Intermediate Algebra students.

In addition, there seemed to be a qualitative difference in the students' opinions about the two mathematics courses. Short exit questionnaires with Linkert scale response options were used. When asked at the outset whether they found mathematics interesting and enjoyable, there was very little difference between the two groups' responses, with some strongly agreeing, some strongly disagreeing and some with neutral opinions. However, there was a discernable difference in the two groups' responses to questions that asked about their attitudes towards "hands-on approaches" to learning mathematics, towards understanding how mathematics relates to the real world and towards developing strategies for thinking about word-problems. Generally, Geometry for Design students strongly agreed that they liked hands-on approaches, strongly agreed that they understood how math relates to the real world and strongly agreed that they had developed strategies for problem-solving through the study of geometry. The Intermediate Algebra students generally agreed with the statements, but were much less likely to strongly agree. Furthermore, many of the Geometry for Design students wrote unsolicited positive comments about the course but hardly any of the Intermediate Algebra students wrote anything about their experience of learning algebra. For example, Geometry for Design students wrote:

"I really am not a math person but I did enjoy this course, more than any other math course I have taken," and " I had fun", and " Wow, a tolerable math class".

This is a statement by Chin, who studied Geometry for Design and transferred to art college:

"I'm thinking more about how technology is (sic) made an impact on the art world. Today I was reading an article in the City Paper. It was about a guy who's currently the Multimedia Director at University of the Arts. He's used the technology to create work that has the same vision and insight as traditional styles of art. I think of the geometry class and how we integrated the computer along with the traditional methods of geometry to come up with a more in depth idea of how the world and geometry work together. I think the time is near where the technology becomes the standard by which we communicate ideas and information on a daily basis."

Figure 4. Students exploring Euler's Theorem

Another statement -- from Jennifer, an Art major who found success with Geometry for Design after failing Intermediate Algebra three times in a row -- reveals the exasperation many students experience with learning algebra in a traditional classroom environment:

"…I kept dropping (Intermediate Algebra Math 118) too late past the time when you could drop. I was receiving 'F's and this kept driving me to keep taking the class to turn the 'F's into an A. …But I was unsuccessful. Then I went to register for this semester with (Head of Art) when she told me that I needed to take a math to graduate in May . I almost cried. She suggested I take Geometry for Design and told me to give up on 118 after reviewing my transcript. I was really hesitant and nervous, but then I talked to other students who had taken this class last semester, and they gave it good reviews. I really enjoy this class. I don't even consider it a math class. I can actually say that I look forward to going to this class. I understand what I'm doing because I can see how everything that is taught relates to the world. I always liked geometry better than algebra; I was always interested in shapes and design….I loved that movie… on MC Escher…"

Thus, after a series of F's in Intermediate Algebra Math 118, Jennifer earned an A in Geometry for Design, finally graduated from Community College of Philadelphia, and also won a $10,000 college scholarship. Although Geometry for Design Students seemed much more enthusiastic about learning mathematics, future study is indicated to better understand the differences in students' attitudes towards learning geometry from a constructionist perspective.

Conclusion

Students who studied Geometry for Design seem to succeed at a higher rate than students who studied Intermediate Algebra. Students who have studied Geometry for Design volunteered that it was a relevant and even at times enjoyable experience; Intermediate Algebra students rarely seemed to respond so favourably. Further study is required before drawing any broader conclusions. Future studies might compare the achievement of students taught geometry from a constructionist perspective using technology and hands-on activities with the achievement of students taught geometry using traditional methods with no access to technology and hands-on activities.

Studying Intermediate Algebra can be a frustrating, potentially unsuccessful and generally terminal mathematical experience for many students. However, Geometry for Design students are immersed in contexts of art, architecture, construction and interior design, and seem to benefit from a geometry course that employs a constructionist approach with technology enhancement and hands-on, activity-based methods.

Echoing Le Corbusier, a Geometry for Design student wrote, "I think this course is very helpful with my understanding of the world around me since mathematics is the true language of the universe…". On a parallel tack, Schneider (p. xxvii) quotes Scottish zoologist and classical scholar Sir D'Arcy Wentworth Thompson who wrote," The harmony of the world is made manifest in Form and Number, and the heart and soul and all the poetry of Natural Philosophy are embodied in the concept of mathematical beauty.". Perhaps it is the nature of a constructionist approach to learning geometry that resonates for students in visual design curricula. More research into this question is indicated.

Thus, as they explore the fundamentals of geometry, students in visual design curricula are finding that they have a better chance of success by studying Geometry for Design, where they are learning some interesting, useful and powerful mathematics.

References

Jeanneret, C.E. Le Corbusier (1954). *Le Modulor.* M.I.T. Press, Boston MA.

Lawlor, R. (1982). *Sacred geometry: philosophy and practice.* Thames and Hudson, London UK.

Schneider, M.(1995). *A beginner's guide to constructing the universe: the mathematical archetypes of nature, art and science.* HarperPerennial , New York NY.

Serra, M.(2008). *Discovering geometry: an investigative approach, fourth edition.* Key Curriculum Press, Emeryville CA.

Seymour, D. and Beardslee, E. (1990). *Critical thinking activities in patterns, imagery and logic.* Dale Seymour Publications, USA.

Learning with Squeak Etoys

Cathleen Galas, *c.galas@uclalumni.net*
Squeakland Foundation, Galas Consulting

Rita Freudenberg, *rita@isg.cs.uni-magdeburg.de*
Dept of Computer Science, Otto-von-Guericke-University Magdeburg, Squeakland Foundation

Abstract

This paper serves as introduction and overview of Squeak Etoys, an engaging computer media-authoring environment, and describes how it aids in the constructionist approach to learning, thinking, and education. Etoys development was inspired by LOGO, the constructionist ideas of Seymour Papert, and Piaget, Bruner, and Montessori. It was developed to help student exploration and discovery in learning and thinking deeply about powerful ideas in math and science.

The design of Etoys includes a powerful user interface that will run on all platforms and allow users to author in multiple dimensions. Etoys is an object-oriented system that is built in Squeak, an open-source implementation of Smalltalk. The learning environment is an open world, a sandbox, ready for the creative visions of the user, and undisturbed by buttons and toolbars. Resources are readily available, a click away, stored in flaps, hidden windows that open to many new possibilities. Beginning activities involve creating and scripting objects, building collaboration between objects, and building, exploring, and sharing projects. The user interface includes modeless editing, so that users can edit anything and at any time in project work.

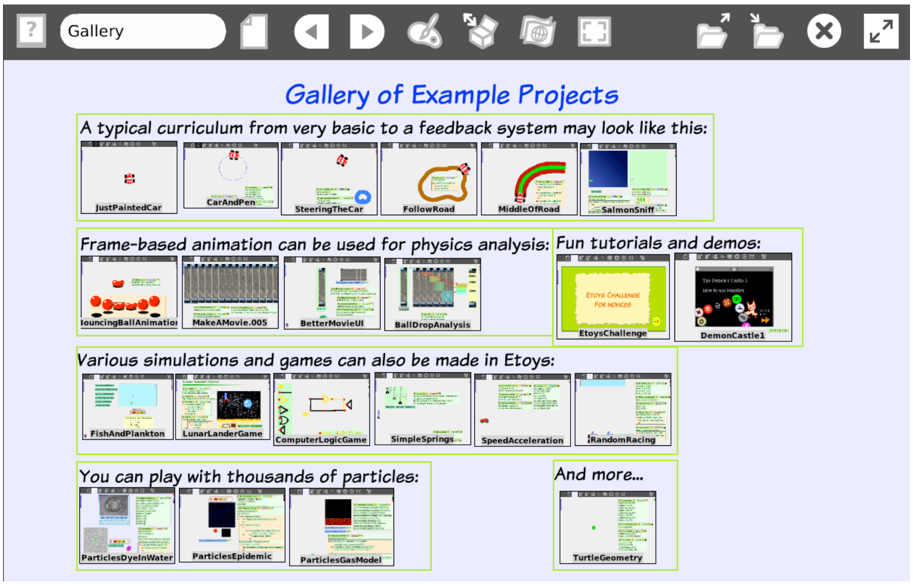


Figure 1. Gallery of example projects, which are shipped within Etoys

Keywords

Squeak Etoys, media-authoring system, constructionism, constructivism, constructionist application, objects-to-think-with, Seymour Papert, Alan Kay, Squeakland, powerful ideas

Introduction

Squeak Etoys is an engaging computer, media-rich authoring environment and visual programming system for children built to help student exploration and discovery in learning and thinking deeply about powerful ideas in mathematics and science. Squeak Etoys includes a wide variety of dynamic media that can be easily created, scripted or programmed, and shared publicly or used for collaboration with others. The “world” is a sandbox, a safe place for non-programmers, an open, flexible, environment in which to create, modify, script, and make objects interactive. It allows children to encounter and learn ideas by playing with them and constructing their own objects visually, kinaesthetically, and symbolically. Children’s representations can exceed their oral communication abilities, and Etoys lowers the threshold for exploring complex cognitive skills that are normally beyond children’s developmental stages, while also lowering the learning curve to work with powerful programming capabilities.

The basic idea behind the design of Etoys is the powerful user interface that will run on all platforms and allow users to author in multiple dimensions. Etoys is an object-oriented system that is built in Squeak, an open-source implementation of Smalltalk. Everything in Etoys is an object, the system comes with objects, and objects can be easily made, copied, manipulated, changed, and scripted with behaviors. The learning environment is an open world, ready for the creative visions of the user, and undisturbed by buttons and toolbars. Resources are a click away, stored in flaps, hidden windows that open to many new possibilities. Beginning activities involve creating and scripting objects, building collaboration between objects, and building, exploring, and sharing projects. The user interface includes modeless editing, so that users can edit anything and at any time. Projects can be published to keep locally, or to share directly to the Squeakland website.

Etoys is an educational tool for use in both classroom and informal learning environments. In the classroom, Etoys can be used on any point on the constructionism continuum, allowing purely learner-centered activities, or providing a context for teacher-directed instruction. Completed projects, available from the Squeakland website Showcase, classroom peers, teachers, or prior projects of the same user, offer opportunity for deconstructions, as children can open the viewer and see how the project is made, make their own changes to objects or scripting, and build upon others’ or their own work.

Brief History of Squeak Etoys development

Squeak is an open-source, object-oriented, Smalltalk language implementation that was derived from Smalltalk-80 at Apple Computer by Dan Ingalls in 1996, with Alan Kay and his Apple research group. (Ingalls, 1996) Squeak then migrated to Disney Imagineering Research as an open source project. Alan Kay was influenced by Seymour Papert and LOGO and directed the Etoys development to support constructivist learning. Etoys is an object-oriented application written on top of Squeak that describes object behaviors by tile-scripting and drag-and-drop operations.

Etoys influenced the development of another Squeak-based educational programming environment known as Scratch. Scratch was developed at MIT with Mitchell Resnick, and John Maloney of the original Etoys development team at Apple.

Squeak Etoys migrated to Viewpoints Research, Inc. in 2001, http://www.vpri.org/, a research foundation to improve education for the world’s children and advance the state of systems research and personal computing, when Alan Kay left Disney Research. Seymour Papert, Nicholas Negroponte, and Alan Kay worked toward furthering the constructivist learning in the developing world through the One Laptop Per Child, OLPC, initiative in 2005. In 2006-2008, Etoys was adapted to Sugar, the platform of the OLPC project. It is an integral part of Sugar and comes preinstalled on all OLPC XO-1 educational machines (B. Freudenberg, 2009).

In 2009, the Squeakland Foundation, www.squeakland.org , was created by Viewpoints Research, Inc., as an initial step in launching the foundation to continue encouraging development and use of Etoys as an educational medium. The Squeakland Foundation is a non-profit organization with a board of thirteen members, including educators, computer scientists, business, and researchers. The Foundation strives to build community around Etoys as free, open-source software, encourage deep learning worldwide using Etoys as the medium, and support, promote, and improve Squeak Etoys and related educational media.

The current version of Squeak Etoys was recently released as Etoys 4.0. This version contains three major improvements in the environment: Children can directly share projects from Etoys to the Squeakland website, "Etoys-To-Go" is a portable Etoys that works on any computer and is easily copied to different machines without installation, and other steps have been taken to make Etoys a completely free and open source system.

Educational Underpinnings

Etoys as an environment for education was originally inspired by the ideas of Seymour Papert (Papert, 1980), Piaget, Montessori, Dewey, Vygotsky, and Jerome Bruner. In 1968, Alan Kay developed the idea of the Dynabook for children after meeting Papert at MIT and watching Papert work with children in LOGO. At Xerox PARC, Kay and Dan Ingalls conceived and built Smalltalk, the first object-oriented language. When Ingalls later built the first implementation of Squeak in 1996, Kay saw this as an opportunity to build on the ideas of Papert and LOGO for a more sophisticated educational environment for children to explore deeply powerful ideas in mathematics and science.

Etoys is both playful and serious; it is not simply a tool to learn by making, but an environment providing opportunities for thoughtful, deep learning. Etoys allows users to create, publish, and share their creation with others, and to collaborate with users around the world.

Squeak Inheritance

Many systems influenced the development of Etoys, including LOGO, HyperCard, Smalltalk, and StarLogo. Once Squeak itself was developed, the development of a simpler authoring tool for object environments including tile scripting became possible. Squeak Etoys is similar to LOGO, but includes costumes and multimedia objects. It is similar to StarLogo, but includes levels of scale. It is like HyperCard or PowerPoint, but it is simpler to use and richer in the multimedia objects it contains. It is like Smalltalk because it is Squeak, a Smalltalk implementation, underneath (Kay 2007).

The LOGO language, especially turtle graphics, influenced Etoys through the idea of visualising algorithms while drawing vector graphics. Every newly created Etoys object owns a pen, which can be set up and down to draw a line when the object is moving. Turtle graphics is part of the Etoys system and creating an object and putting the pen down is a good beginning exercise to situate the user in the world.

The HyperCard system, released in 1987, introduced an easy-to-use programmable environment, described as “Object-Like Programming“. This system allows its users to author their applications without typing any commands. Users design new cards and model dependencies between cards using functions from a menu. HyperCard gives the user a lot of flexibility, freedom, control, and power (Shafer 1988). Etoys user interface allows users to author and edit anytime and anywhere in the project; it gives all the power to the user, since it allows them to completely reprogram the system.

StarLogo is an agent-based system developed by Mitchell Resnick at MIT. StarLogo allows users to explore decentralized systems like ant colonies or bird flocks through the modelling environment. StarLogo is a version of the LOGO language that allows the control of multiple turtles in parallel. (http://education.mit.edu/starlogo/) StarLogo was the inspiration for the particle system in Etoys called Kedama.

Basics: Drive a Car

Etoys is a constructionist tool built to help children construct, reconstruct, and deconstruct their understandings of the world in ways that incite curiosity, questioning, and amplification of learning big ideas in powerful ways. The simple, yet powerful user interface, provides and extends most personal computer needs, to: "runs everywhere on everything" and allows end-users "authoring and access to all things". (Kay, 2005)

In the traditional, Alan Kay "Drive a Car" Etoys curriculum, playing with a purpose to create and drive a car offers opportunities to interact with mathematics of the real world, an unfortunately rare occurrence in many traditional educational settings. Students don't necessarily reflect on their learning by describing the fact that they are programming on the computer, or that they are learning math, but rather, they are making a car and driving it!

The drive a car curriculum provides children with authentic problems to be solved so that something works better. The idea of using gear ratios to make a smoother car ride is an example. One must have an understanding of how the concept of gears is applied to the problem. By experimenting with different ratios, and likening it to the child's world using bicycles and gears climbing a hill, students develop visceral kinaesthetic understandings of big concepts. The problem that they can't control the car is an authentic problem within Squeak Etoys. Driving the car successfully is personally motivating and authentic to the user. (Galas, 2001) It is the car they have taken time to create and now they will be motivated to spend time solving the problem of how to drive it. The educational system or the teacher does not impose this problem; the user wants to make their own construction driveable. Solving these problems requires users to make decisions, make choices and test out their hypothesis to solve the problem. They spend time, thought, and energy in the pursuit of "hard fun". (Papert).

Creating Objects

The Etoys system comes with an assortment of objects the user can readily use, and users can create new, similar kinds of objects anytime and anywhere in the system. Users can take an included object and easily change the color, size, and costume, a scripting term used to describe the objects appearance. The objects scripts can be easily changed, as well as its interaction with other objects.

To create a new object, the user just makes a painting and clicks on "Keep". The painting tool provides a new, blank sheet of paper, to draw your idea. Each new object can immediately be scripted, and many objects can be scripted separately, and set to run separately, or at the same time. Existing objects can be easily redrawn; the user can "change the costume" of the object at whenever desired. Objects include everything in Etoys, from the supplied objects to interface objects, text, fonts, shapes, arrows, balloon help, sound recorders, movies, and menu items. This is one of the powerful ideas of the structure of Etoys, the idea that you can handle every object, including all media, in Etoys, in the same way, and with the same power.

In contrast to many other systems, the creations of the users are always the focal center. The entire screen, in Etoys called the "world", is available for creation. The user Interface is built so that the functionality of an object and opportunities for object manipulation will be revealed when right-clicking on an object. This allows Etoys to be modeless, so users don't need to switch between "editing mode" and "running mode".

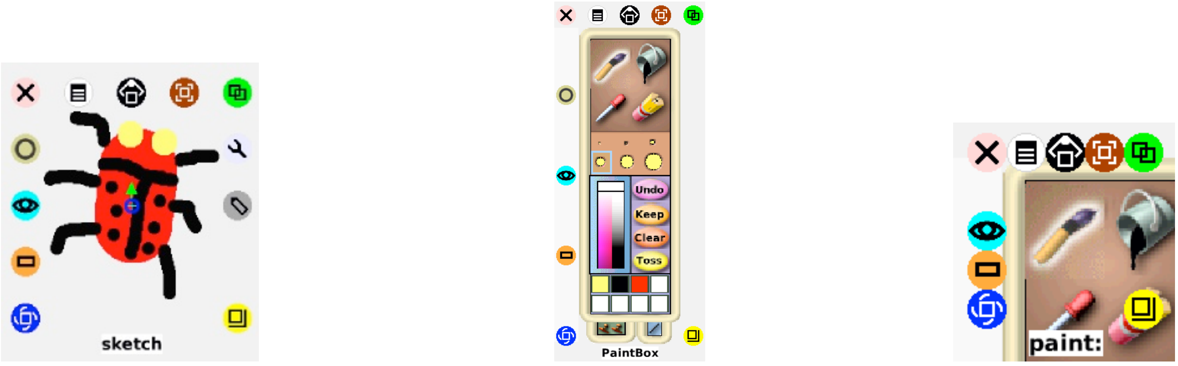


Figure 2. Halo for painted object, PaintBox and Brush

Scripting

Scripts bring the objects the user has created to life by telling the object what to do. Besides copying, rotating, resizing, embedding, or deleting, the most powerful way to manipulate the objects is to script them. Dragging the tiles onto the desktop, and then dragging the tiles into the script makes scripts. By scripting in this way with tiles, the syntax is always correct and will always work when the script ticks, or is started. The turquoise halo opens the viewer, which shows properties and behaviors of the object, and is always available for analysis or change.

Anytime the user wants to view object attributes, they can open the object's viewer; it will reflect the current state of its object. For instance, the property "heading" shows the direction an object is pointing, and the number in the viewer corresponds with the actual graphical direction of the object on the screen. Changing the number or rotating the graphical object will always update each other. The powerful idea here is to learn that there can be different representations of the same underlying data. Exploring with objects using heading is a powerful idea that bridges the known reality of the user to the computer, allowing direct manipulation and change in perception of the real world. In this process of exploring and discovering, users begin to comfortably represent their world symbolically.

To build scripts, the user arranges tiles from the viewer in a scriptor. The numbers in the tiles can be changed, and several tiles can be combined. The script itself is a Squeak Etoys object, so the user can get its handles and access the script's viewer, and use a script written for another object like the car. The user could make the scriptor, an object itself, leave a pen trail and make a circle or show any other path the user scripted.

There can be several scripts for every object, and all scripts can be executed at the same time, timed to start at different times, or manually started at different times. Child users often have an easier time with parallel processing than adult users. It is also possible to stop a script, make changes and start it again, while other scripts are still running.

When 10-year-old children create scripts similar to the example shown in Figure 3, they begin to gain fluency and appreciation of the power of symbols. The pen down function makes forward and turn visible to the user; they can visualize this familiar event in their world. The discovery process first involves the use of some known reality, moving into direct manipulation that transforms the learner's perception into meaning, and lastly moving into the symbolic mode of representation. Bruner's work also suggests that even very young learners can learn complex ideas as long as the instrucrion is organized appropriately. (Bruner, 1966) The repetition of visual manipulations, like drive a car scripts, over time, lead into the development of abstract ideas. Alan Kay expanded on Bruner's ideas to refer to images in the world representing and modelling the internal thoughtful, or the reflective symbols of the mind. (Kay, Google video) The

Etoys user interface gives young children the appropriate instructional environment to explore, build, and eventually reflect on their constructions and powerful mathematics ideas.

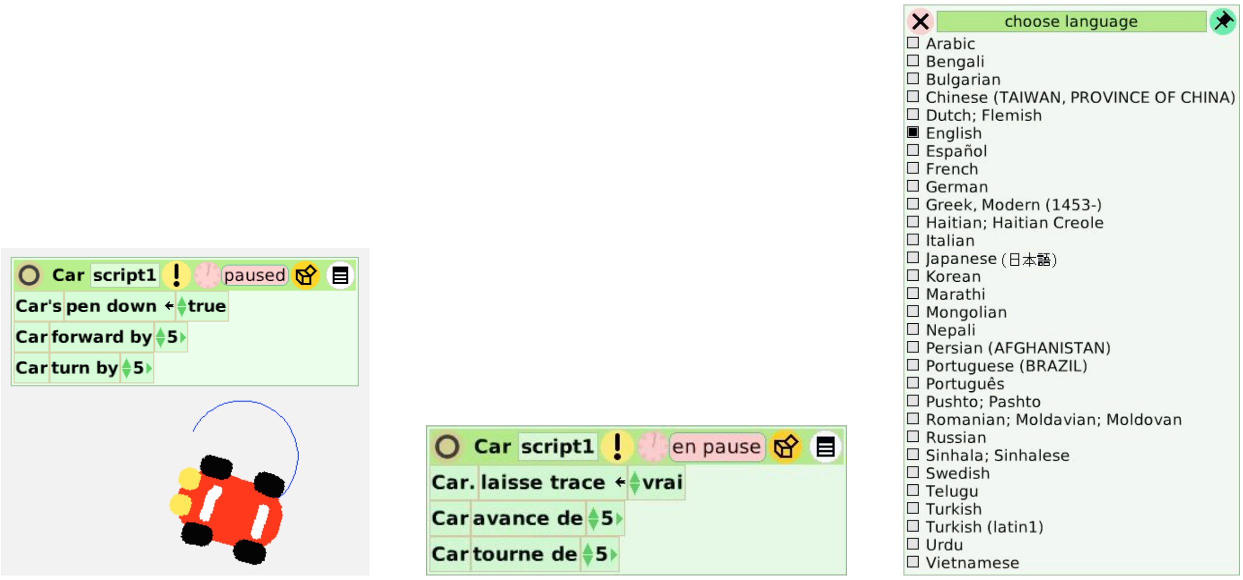


Figure 3. A script for moving the car with pen down and the same script when language is changed to French. A list of available languages in the current version of Etoys.

Etoys is a "live" system, allowing the user access to objects and scripts and possibilities to make new objects or script old or new objects while the system is running other scripts. This immediate and always available live system, makes variables and feedback more accessible to the user. Creating robotic car scripts that drive themselves on a path uses feedback. The powerful idea of feedback to an object is scripted using conditional statements, if-then statements, requiring the user to think again in new ways about their world.

Collaboration between Objects

While creating one object and scripting that same object is easily accessed, creating objects that interact with each other requires the user to think about more than one viewer while scripting. While a script belongs to one object, you can easily add tiles from other objects into each script. The user will need to understand how the movement of one object is related to the scripting in another object. In the Figure 4, you see a car and a steering wheel. The script says, that the heading of the car should result from the heading of the steering wheel. When you set the script "ticking" and use the blue halo of the wheel to rotate it, the car will also rotate. By exploring and playing with these objects and their interaction, users can discover how the feedback from the steering wheel's heading can steer the car. This activity again requires the user to apply their understanding of the real world through symbolic representations in the Etoys world. Numeric feedback, in the form of positive and negative numbers, is used to drive the car. Through continued exploration, users gain fluency in driving by manipulating the numbers to drive straight, or to the left or right, as desired. The computer representation is an extension and symbolic representation. Fluency in the manipulation of the car is a kinaesthetic manipulation of variables, feedback, positive and negative numbers, and the beginning of modelling, and reflecting on, these powerful ideas through computer play. (Kay, Google video)

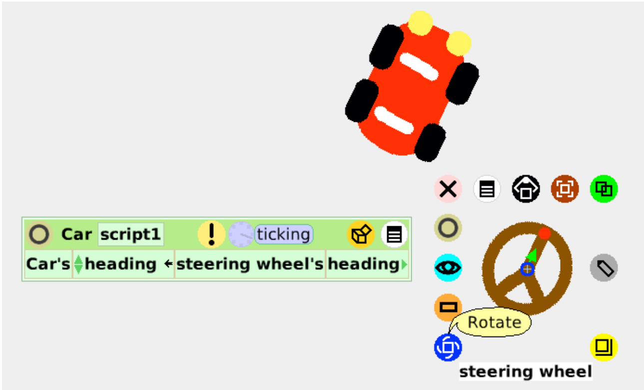


Figure 4. Car's heading is retrieved from steering wheel's heading

This is a simple example of collaboration; you can of course build scripts that are more complex and create simulations and games. Using a tile of another object instead of a number also introduces the concept of a variable: that a name stands for a number.

Animations

Animations are quickly and easily made. The user paints several "costume changes" for an object and drags tiles to a script to move the cursor and change the costume. Movies and videos in Etoys are just an implementation of the animated objects. Frames from a movie are dropped in a holder, the user drags the tiles for the script, and the movie will play! The Squeak Etoys book has pages that are just different costumes on the Etoys object! Books and power-point like presentations are just animations of objects in the system.

Extending Etoys: Kedama

Kedama is a system within Etoys that was inspired by work on StarLogo. StarLOGO is a LOGO dialect for the modeling of massiv parallel agent based simulations. It allows graphical programming and inspired the development of Kedama. The actual Etoys version of Kedama is called Particles and is the environment for massive parallel simulation within Etoys, which can be scripted in the same way as other Etoys objects.

Learning and Teaching

Constructionism and Constructivism

Etoys provides an environment for Piaget's constructivist and Papert's constructionist learning. Teaching for constructivist learning requires the teacher to view the learner as unique, and take into account the culture and background of experiences of the learner. The responsibility of the learning is the learners, although the motivation for learning is reflected in the student's confidence in their learning. Vygotsky's ZPD (zone of proximal development) is the area between the things the learner can do by themselves and the things they can do only with the aid of a peer, mentor or teacher. The larger the child's ZPD, the more they will be able to do. Constructionist learning further requires that students are actively engaged in some kind of product construction, such a LOGO project, or an Etoys project.

Learning Etoys

Earlier Etoys versions started with a blank screen and offered the same power as a blank page of paper and a pen – but also the same challenges. Now, Etoys starts with a pre-loaded project, showing an open script and a moving car, and links to tutorials and example projects, so users can start exploring existing projects. Users that have not had experience in open constructionist learning environments may have initial difficulty getting started. Learning through play and exploration is part of the Etoys experience and users can be encouraged to spend some time just exploring. However, complete freedom to discover doesn't give users the competence that creates the greater ZPD and confidence to pursue project construction.

Etoys provides scaffolds

Using the drive a car curriculum, users learn the basics of objects, scripting, turtle graphics, pen down visualizations, and interacting objects while learning to "pay attention" to some of the powerful ideas they are exploring. The drive a car curriculum is available to users in informal settings, in Etoys tutorials on the Squeakland website (www.squeakland.org), and can be used for beginning Etoys in a formal classroom environment. Etoys provides opportunities for Papert's "hard fun", but attention to scaffolding support can help build the user threshold for dealing with difficult problems, thinking through and exploring possible solutions, and generally building tenacity and resilience in thinking.

The user interface is "user friendly" in that it allows the user to access the program and the computer at all times. This provides support for the neophyte user in gaining competence. The user interface is representative of the system itself; the tiles that you drag and drop are visual representations that look like phrases or sentences and clearly show that the objects are getting the "messages" that instruct their behaviors.

Teachers in learner-centered classrooms can introduce Etoys by observing their students individually and in groups, and providing the "next step", or what is sometimes called "just in time" learning. Teachers in this setting should be ready to support learning through expert peers, expert aids or parents, or they provide the next step to give students in their Etoys experiences. (Galas,1999). Usually, students will ask for their next step because of their own deep engagement and motivation to continue to construct a project. Scaffolding learning and support for building projects is imperative, as different users have different thresholds (ZPD) for feeling competence and success. Extending the ZPD of users by scaffolding supports to students can be done with expert peer assistance, expert teacher assistance, apprenticeship learning set-ups, or encouraging and teaching to the use of the balloon help and the tutorials in the Squeak Etoys systems.

Squeak has no ceiling

Squeak Etoys is a powerful educational tool that can extend the range of user creations and the powerful ideas that users can explore. The underlying programming of Squeak Etoys is Squeak, a powerful, full-featured, object-oriented programming environment.

Etoys is developed in Squeak, an open-source Smalltalk implementation. This is immediately obvious for beginning users, who would be overwhelmed with an expert-level programming environment, but this is Squeak and Etoys. You can go beneath the Etoys interface, or as Kay and Ingalls say, "look under the hood", and access the underlying system using a special key. You then have immediate access to the full power of the Squeak programming language.

In Figure 5, you can see the similarity of the two representations of the script. There is an option to show the text code version of a script within Etoys, but to be able to write textual code, you need to actually open the system browser.

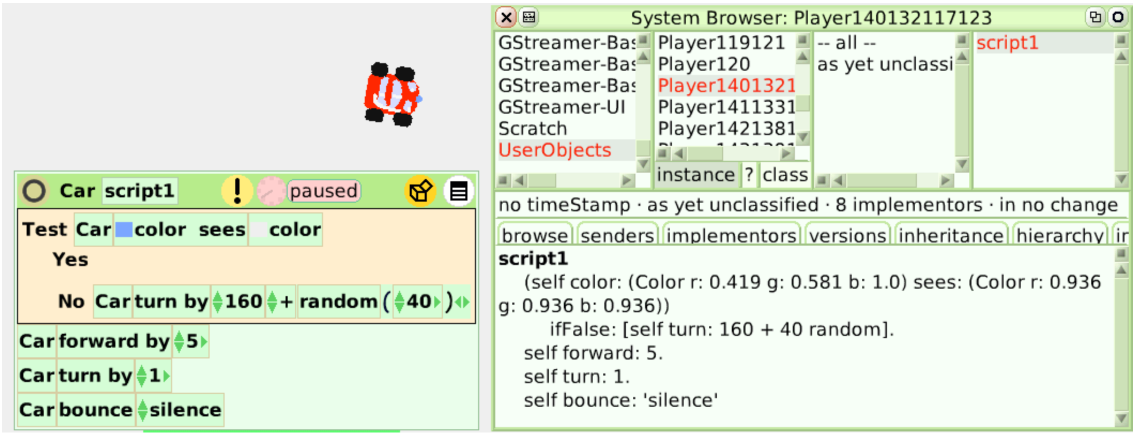


Figure 5. Tile version of Car's script and text code of the same script in the system browser

Construction and De-Construction

In Etoys, users can construct their own projects, and can explore and deconstruct projects from others to learn how these are built. You may need to view several individual scripts that are part of larger projects, and discover how the scripts work together. This ability is not only a powerful possibility of the system, but it also lets users know there is no “magic” in the computer; users can find out how everything works and work to understand or recreate it.

Another Etoys project use is exploration of prepared educational projects. There are projects built by educators or parents to teach specific concepts to children. These projects include the main idea, a simulation of a phenomenon, a description of the content, labels, explanations,, and, text fields to change parameters,. There are readily available objects in the Etoys supplies bin and object catalogue to enrich projects created by teachers or parents for these purposes.

In directed teaching environments, teachers can provide a “prepared environment” for students to experiment and explore specific topics, such as measurement (Pyramid Challenge, NASA Connect). Etoys provides teachers an easy-to-use tool for preparing lessons, thereby familiarizing teachers with Etoys use and possibilities. For students, these teacher-made projects are a resource to explore and expand their learning. (R. Freudenberg, 2009).

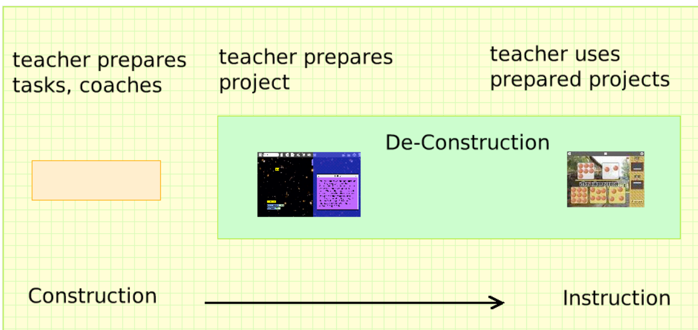


Figure 4. Different ways to use Etoys in the classroom

Conclusion

Squeak Etoys is a powerful constructionist tool that allows students on multiple platforms to publish and share projects and explore the world of powerful ideas by building and manipulating interactive simulations as active, not passive learners. Learning through play and exploration in the open world is an important part of the Etoys experience. The user interface is designed to openly allow the user access to play with powerful ideas in mathematics and science. Etoys is an educational tool for use in both classroom and informal learning environments. Opportunistic teachers can utilize learning opportunities as they occur in student exploration, using "just in time" learning and teaching practices. These teachers can question, help students notice powerful connections, mediate classroom discourse, and nurture the development of deep thinking about powerful ideas (Galas, 2001).

Peer projects and teacher projects or a users saved projects, are all available from the Squeakland website Showcase, and offer opportunites for deconstructions, as children can open the viewer and see how the project is made, make their own changes to objects or scripting, and build upon others' or their own work. Squeak, the underlying system in Etoys, is available for users wishing to move into programming modes involving greater expertise than the basic Etoys system. Squeak Etoys is a powerful system that can help young children learn mathematics and science powerful ideas that are not easily accessible in other learning environments.

References

Bruner, J.,Toward a Theory of Instruction, Harvard University Press, 1966.

Freudenberg, B., Oshima, Y., Wallace, S., "Etoys for One Laptop Per Child", Seventh Annual International Conference on Creating, Computing, Connecting, and Collaborating through Computing, Kyoto University, Kyoto, Japan, January 2009. (*www.vpri.org/pdf/tr2009001_etoys4olpc.pdf)*

Freudenberg, R., "Lernen mit Etoys", Proceedings 13th GI-Fachtagung Informatik und Schule, Berlin, LNI P-156, Bonn, 2009.

Galas, C., "School Squeaking", Squeak News, Volume 1, Issue 4, October 2001. (www.squeakland.org/resources/articles/article.jsp?id=1012)

Galas, C., "The Never Ending Story, Questioning Strategies for the Information Age", Learning and Leading with Technology, ISTE, April 1999", Learning and Leading with Technology, ISTE, April 1999.

Ingalls, D., Kaehler, T., Maloney, J., Wallace, S., Kay, A., "Back to the Future: the story of Squeak, a practical Smalltalk written in itself", OOPSLA, Atlanta, Georgia, 1997. (http://portal.acm.org/citation.cfm?id=263754)

Kay, A., "Children Learn by Doing: Squeak Etoys on the OLPC XO", VPRI, 2007. (*www.vpri.org/pdf/rn2007006a_olpc.pdf)*

Kay, A.C., "Squeak Etoys, Authoring and Media", VPRI Research Note RN-2005-002 (*www.vpri.org/pdf/rn2005001_learning.pdf*)

Kay, A.C., "Doing with Images Makes Symbols: Communicating with Computers", http://video.google.com/videoplay?docid=-533537336174204822, Apple Computer.

Papert, S., Mindstorms: Children, Computers, and Powerful Ideas, Basic Books, 1980.

Paper, S., "Hard Fun", http://www.papert.org/articles/HardFun.html.

Shafer, D., HyperTalk Programming, Hayden Books, Indianapolis, 1988.

Getting started with Squeak Etoys

Rita Freudenberg, *rita@isg.cs.uni-magdeburg.de*
Dept of Computer Science, Otto-von-Guericke-University

Introductory description and overall goals

Squeak Etoys (http://www.squeakland.org) is a free, multi platform authoring tool. It is tile based programming and is available in several languages at the click of a mouse. It was developed by a small group of people under the direction and vision of Dr. Alan Kay. Since 2009, a new non-profit organisation has taken on the task to continue develop, support and maintain Etoys and it's growing community of users.

Etoys development was inspired by LOGO, the constructionist ideas of Seymour Papert, and Piaget, Bruner, and Montessori. It was developed to help student exploration and discovery in learning and thinking deeply about powerful ideas in math and science. The workshop will show a variety of projects which can be created to give participants suggestions how it can be used in the classroom. In the hands-on part of the workshop, the participants can experience the Etoys constructionist learning environment and go through a typical Etoys lesson.

Method

The workshop will start with a brief introduction of Etoys and show example projects and how these can be integrated within a school curriculum. Based on the interest and background of the participants an suitable introductory project will be chosen.

The second and larger part of the workshop will introduce the basic ideas in developing Etoys projects: creating objects, describing their behavior and producing dynamic behavior. Using the example project, the participants will walk through the basics of painting an object, exploring it's properties, scripting it, controlling it and performing simple tests. The participants will receive hand outs with a basic description, that they can take away. Knowing the Etoys basics they will be able to create their own projects and to reproduce the projects shown at the beginning of the workshop.

Finally, the participants will have a chance to ask questions and exchange ideas on how they might take Etoys into their particular learning environments.

Expected outcomes

The participants will get an introduction to Etoys and an overview of how it can be used in the classroom. They will get a short demonstration of example projects from different geographic and subject matter areas, suitable for learners of multiple ages and then will start a hands-on experience using Etoys to create their own projects.

Keywords

Squeak Etoys, media-authoring system, constructionism, constructivism, constructionist application, objects-to-think-with, Seymour Papert, Alan Kay, Squeakland, powerful ideas

Exploring Elements of Linear Algebra through Experiments with LOGO

Karl Josef Fuchs, *karl.fuchs@sbg.ac.at*
Department of Mathematics and Informatics Education, University of Salzburg

Hans-Stefan Siller, *hans-stefan.siller@sbg.ac.at*
Department of Mathematics and Informatics Education, University of Salzburg

Abstract

In terms of constructionism knowledge and skills are internalized activities. Teaching following this approach has turned out to be efficient at any stage of life. This paper discusses the design of a course for the introduction of Linear Algebra at secondary level combining Mathematics and Computer Science. The way to gain basic elements does not head to traditional definitions and subsequent exercises but turns the path almost upside down. Constructing, experimenting and exploring are at the beginning. The implementations of LOGO functions should open up and produce deeper insights into basic elements of Linear Algebra by the students.

Keywords

Mathematics, Computer Science, Linear Algebra, Constructing - Experimenting - Exploring, LOGO Functions, Transfer, Analogy, Inner Differentation

Prelude to the Constructionistic Approach

If you are skimming over the titles of our paper you may receive the impression of huge inconsistancy. Apparently we want something from Elements of Linear Algebra like **vectors** and **matrices** which are largely regarded as extremly abstract objects. Otherwise we address exploring and experimenting as appropriate strategies to discover these abstract basic ideas. With the following concept we want to demonstrate that constructionistic thinking (Papert 1993; Papert, Solomon 1971) overbears seemingly even such opposed gaps.

We do not conceal on the fact that through this approach with the support of computers in particular through the use of LOGO, LISP and CAS Mathematica we have taught many highly motivated students in computer classes at grammar school from age 16 to 18 and teacher students in Mathematics and Informatics at age 19 at the university for a couple of years with the result of a profound knowledge of elementary data – structures on one hand and of Elements of Linear Algebra on the other hand finally.

All implementations in this contribution are made in MSW LOGO in terms of uniformity, adaptations in LISP and CAS Mathematica® can be set up easily.

Wading through the Course's Design including students' monitorings

Atoms, Lists and Functions – Assembling and Disassembling

At the beginning all the students are informed of the data structures used in LOGO. Easy examples carried out by students in groups deal with manipulations on **atoms** and **lists** using the **functions** word and sentence for assembling and first, butfirst (bf), last, butlast (bl) for disassembling. Additionally the functions wordp and listp are used for checking the outputs. All the inputs are executed immediately after entering the code into the **Command line**.

The following section shows a work – sheet of average level of difficulty the students have to work with cooperately.

Given: [[1 2] 23 [24,25]]

- Combose the given list!
- Extract
  - the element 2 as a list,
  - the element 23 as a word and
  - the word 2425 from the given list!

Possible solutions are:

```
show [[1 2] 23 [24 25]]
[[1 2] 23 [24 25]]

show bf first [[1 2] 23 [24 25]]
[2]

show first bf [[1 2] 23 [24 25]]
23

show word first last [[1 2] 23 [24 25]] last last [[1 2] 23 [24 25]]
2425
```

Some of the students solved the last problem in two steps

```
show last [[1 2] 23 [24 25]]
[24 25] and then
```

show word first last [[1 2] 23 [24 25]] last last [[1 2] 23 [24 25]]
2425

which is absolutely reasonable as they want to ensure themselves of the object which they will be disassembling further.

Vectors and Matrices

Definitions and Basic Attributes

In the following it remains to be seen that the elements of Linear Algebra can be obtained from the fundamentals in the first work – sheet by the strategies of Transfer (Schubert, Schwill 2004) and Analogy (Herber, Vásárhelyi 2002) which are important for Constructionistic Thinking.

Hence the outcome of a consequent implementation of the concept for the methodology is as follows. In one respect the teacher must namely be the guider of the course as he owns the professional and educational competences. He develops work – sheets for each step of the course presented in the following with permanent regard to the learning process of the students. In short: During the construction process of the students the teacher withdraws from the classroom union and gives support individually in the other respect (Fuchs 2007, p. 183).

We will call elements like [1 2 3] or [-2.3 0] **vectors of dimension three** respectively of **dimension two**. Generally we will call $[\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_\mathrm{k}]$ **vectors of dimension k**.

Immediately we have to cope with a new problem. What does the attribute **dimension** address? Very soon the students will find out that the dimension equals the number of elements of the vector.

The predefined LOGO function count satisfies our problem. The students can find out the dimension of a **vector** easily. For example typing in

show count [-2.3 4 5.75]

into the **Command Line** will yield 3 which is the correct answer.

Gradually we define elements like [[1 2 3][4 5 6][7 8 9]] or [[1 0][0 1]] as **3 × 3** respectively **2 × 2 matrices**. Generally we will call $\begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix}$ **m × n matrices**.

The next steps are up to the students' responsibility in a broader extent.

The first task will soon be done by the students namely to show the relation between the 'list – of – lists' – representation such as [[1.5 3.43][2.756 3.1]] and the common symbolic 'mathematical representation' $\begin{bmatrix} 1.5 & 3.43 \\ 2.756 & 3.1 \end{bmatrix}$.

But when using the count function again the students soon become aware that for example

show count [[2 3][4 5]]

yields 2. A result which is absolutely justifiable by the students – two lists are the elements in the given list – but it is not adequate for **matrices**.

So the second task will be more difficult. Analysing the definition the students will answer the question according to the **dimension** of a **matrix** by bringing in the number of rows and the number of columns for a new definition of the attribute. We agree to this statement but for this reason the dimension of a matrix must be splitted into a **row – dimension** and a **column – dimension** consequently. Now the students' big challenges consist in the construction of a dim – **module** for **matrices**.

Additionally we want to introduce the **Programming – (Editing) Mode** of MSW LOGO and the concept of the **parameter** or **variable** in LOGO.

Empirically this implementation necessitates some time for experimenting to become familiar with the **Editing Mode**. Most of the students will approach by 'trial and error' when modifying the code in the edit – window. Only few of them focus on structuring the design before entering the code. Never astonishingly these students reach the desired implementation more quickly.

```
to dim :mat
op sentence count :mat count first :mat
end
```

Further Attributes

We sustain our course by going on the explorations of further properties of the elements **vector** and **matrix**.

Forces, velocity and acceleration are terms the students know from their Physics lessons. The Natural sientist uses **vectors** to describe such concepts. We take advantage of these interdisciplinary aspects in our concept.

As a matter of course the students argue that all these vectors are characterized by two informations, one is the **direction** the other the **absolute value** of the term. We will focus on the second one.

We will solve the problem by transferring it to Geometry and then by stepwise refinement.

The two dimensional problem:

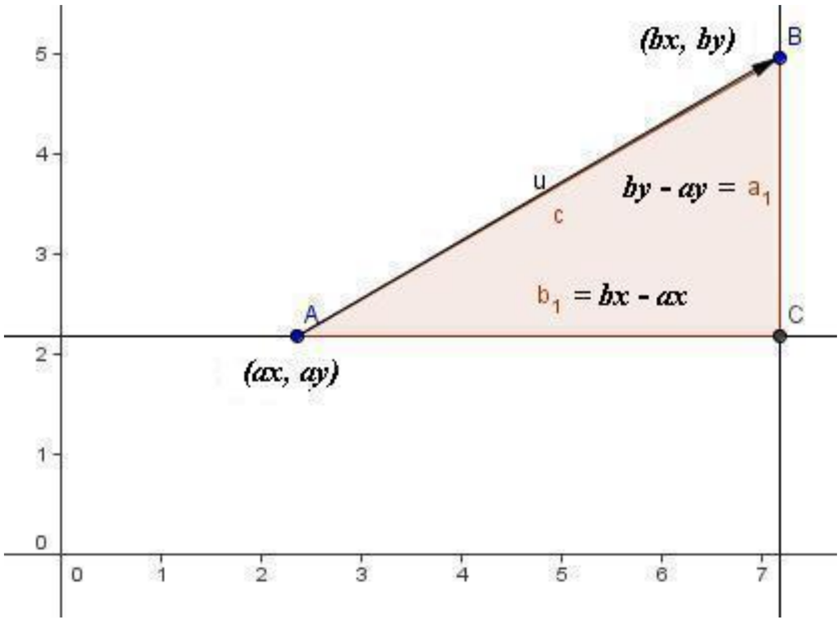


*Figure 1. Length of **vector** u*

Figure 1 suggests that the **absolute value** c of **vector** u can be found by the Pythagorean Theorem easily: $c = \sqrt{a_1^2 + b_1^2}$, where $a_1 = b_y - a_y$ and $b_1 = b_x - a_x$.

The solution for the three dimensional problem should be devolved by the students.

Although the generalization $c = \sqrt{u_1^2 + u_2^2 + \ldots + u_k^2}$ (with $u_1 = b_{x_1} - a_{x_1}, u_2 = b_{x_2} - a_{x_2}, \ldots,$ $u_k = b_{x_k} - a_{x_k}$) is beyond graphical representation the expression is accepted by the students willingly.

But now our interest concentrates on the LOGO implementation of the attribute.

Already from this stage on we use the widely unfamiliar LOGO functions MAP and APPLY for the manipulations of our **lists** (**vectors**). Once again we can avoid recurrent value assigments (Fuchs, Siller, Vasarhelyi 2008).

```
to abs_val :v
op sqrt apply "sum map [? * ?] :v
end
```

We will test our function to err on the side of caution:

```
pr abs_val [2 3 4]
```

yields

```
5.3851648071345
```

which can easily be checked as the right answer.

But we can also discover further attributes for **matrices**. We want to pick out the **symmetry**. On one hand we choose this property as it is of notably importance when teaching algorithms. More efficient strategies can be gained when adjacency matrices which represent the implementations of graphs are **symmetric**. On the other hand we settle on this attribute as we will come across with another interesting **module** namely *x* on the way to the final implementation of *symmetry*.

So let's go back to the problem. A **matrix** is **symmetric** when it fits in with its **transposed matrix** which evolves by mirroring the elements on the main diagonal $(x_{11}, x_{22}, \ldots, x_{kk})$.

Although this attribute sounds very repellent the students will have no problem with it as some of the students' comments like **symmetric matrices** must be quadratic give evidence to our statement.

Hence we will implement a LOGO transpose function first. Investigating a reduced problem – a **symmetric 3 × 3 matrix**

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}$$

Figure 2. symmetric 3 x 3 matrix

we will find out that the strategy will be to generate a new **matrix** where $x_{ij} = x_{ji}$ for *i, j = 1, 2 3*. We are satisfied with the **algorithm** and generalize it for i, j = 1, 2, .. k but the implemetation of this strategy makes a most interesting new module necessary. We will call it simply *x*.

x will be a selector function in the two parameters *:i* and *:j* using the predefined LOGO item function. In doing so *x : i :j* addresses the element x_{ij} in the **matrix** *:mat*.

```
to x :i :j :mat
op item :j item :i :mat
end
```

The implementation of the **module** to check the **symmetry** brings in some very challenging new ideas which must be explored in a dialogue between teacher und students. Thereby a main focus rests on the **control structure** of **conditional branching** that comes along with the **recursion**.

```
to symmetry :mat :row :col
if equalp dim :mat sentence :row :col [op "true]
if not equalp x :row :col :mat x :col :row :mat [op "false]
if not greaterp :row first dim :mat [if lessp :col last dim :mat [op symmetry :mat :row :col+1]]
op symmetry :mat :row+1 1
end
```

When discussing the LOGO source code the main focus will be on the two **conditions** *if equalp dim :mat sentence :row :col [op "true]* and *if not equalp x :row :col :mat x :col :row :mat [op "false]*. Such conditions are essential in programming recursive functions to avoid infinity in executing. Empirically they are often ignored by the students.

We are satisfied with the **module** as

```
show symmetry [[1 -2 3][-2 4 0][3 0 5]] 1 1
```

yields

```
true
```

whereas

```
show symmetry [[1 -2 3][-2 4 1][3 0 5]] 1 1
```

outputs

```
false.
```

We expect that our severe philosophy in implementing the code did not escape the attention of the reader as we strictly avoid value assignments using make. Our intention is to show the students that these commands which they know from courses in imperative programming very well are not necessary for a consequent functional programming style.

Discovering Operations

Exemplarily we will discuss some operations with **vectors** and **matrices**. Constructionistic Thinking in this case means that we will not define the operations traditionally but gain them by experimenting and playing.

Our first example is about the similarity of documents. Although the solution of the following problem is well – known in Information Theory this fact is not communicated to the students.

The modelling process will start with the question how to indicate the similarity of a document. After some discussions we agree upon the strategy to bring the absolute frequency of some relevant terms in the document into account for similarity.

We decide to describe each document by a **vector** with the absolute frequencies as its components. [1 5 0 0 1 2 8 1 0 1] will be an implementation for a document where the number of relevant terms *n* equals 10.

After some further continuative discussions with the students we decide to multiply the according elements of two documents (= **vectors**) and sum up these products finally. Hence the output sum is a rate for the similarity of the two documents.

Main arguments for this solution are:

- If the absolute frequency of a relevant item equals zero then the result will be zero regardless of the value of the other factor.

- If the absolute frequency of a relevant item equals one then the value of the other factor will be the value of the product.
- If both values of the absolute frequencies of the relevant items are bigger than one then the value of the product will be bigger than the value of each factor.

Now we are prepared for the LOGO implementation which should be done mainly by the students.

```
to similarity :v1 :v2 :pv
if equalp :v1 [][op apply "sum :pv]
op similarity bf :v1 bf :v2 fput product item 1 :v1 item 1 :v2 :pv
end
```

Finally we nominate this operation with **vectors inner product** or **scalar product** and test the **module**. The results are very satisfying.

show similarity [1 5 0 0 1 2 8 1 0 1] [0 4 0 1 1 3 5 0 4 1] [] yields 68 and
show similarity [1 5 0 0 1 2 8 1 0 1] [5 0 4 3 1 2 1 0 5 0] [] outputs 18.

Even if we only fly over the inputs we will consider the first two vectors as more similar than the second ones.

The second example will bring us to the **matrices** and it may be called metamorphoses.

We start with a square. It is a magic one as the sums of all its rows, columns and diagonals equals fifteen.

2	7	6
9	5	1
4	3	8

Figure 3. Magic Squares

The square will be implemented as **matrix**.

Now the students' problem is to create a new **3 × 3 matrix** by multiplying all the rows' permutations of the given square. For the further process it is indicated to name the **matrices** with A which is the original magic square and B, C, D, E and F for its rows' permutations.

A·A	A·B	A·C	A·D	A·E	A·F
B·A	B·B	B·C	B·D	B·E	B·F
C·A	C·B	C·C	C·D	C·E	C·F
D·A	D·B	D·C	D·D	D·E	D·F
E·A	E·B	E·C	E·D	E·E	E·F
F·A	F·B	F·C	F·D	F·E	F·F

Table 1. Magic Square Permutations

Back order the question 'What do we mean by **multiplying two matrices**?' is still open.

After some discussion we stick back to the knowledge that each row or column is a **vector**. In the example before we became acquainted with the **scalar product** expressed in the similarity **module**.

$$\begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix} \cdot \begin{pmatrix} y_{11} & y_{12} & y_{13} \\ y_{21} & y_{22} & y_{23} \\ y_{31} & y_{32} & y_{33} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}$$

*Figure 4. Identifying the **scalar product***

So let us implement the already existing **module** as a new function for the product of two (quadratic) **matrices**.

```
to metamorph :mat1 :mat2 :mat3 :i :v
if :mat1=[][op :mat3]
if greaterp :i 0 [op metamorph :mat1 :mat2 :mat3 :i-1 fput similarity last :mat1 pick_out :mat2 [] :i [] :v]
op metamorph bl :mat1 :mat2 fput :v :mat3 3 []
end
```

with

```
to pick_out :mat :v :i
if :mat=[][op :v]
op pick_out bl :mat fput item :i last :mat :v :i
end
```

Only the students with outstanding abilities in functional programming are able to implement this LOGO **module**. Nevertheless there is enough room for all the students to participate when recapitulating the given problem.

Now the additional task is not only to generate the permutations of the given magic square but to find out that some new generated **matrices** will be **symmetric** such as C · B with $B = \begin{pmatrix} 4 & 3 & 8 \\ 9 & 5 & 1 \\ 2 & 7 & 6 \end{pmatrix}$ and $C = \begin{pmatrix} 9 & 5 & 1 \\ 4 & 3 & 8 \\ 2 & 7 & 6 \end{pmatrix}$.

```
show metamorph [[9 5 1][4 3 8][2 7 6]] [[4 3 8][9 5 1][2 7 6]] [] 3 []
```

yields

```
[[83 59 83] [59 83 83] [83 83 59]]
```

Finally we use our already discovered symmetry **module** for automatic testing.

```
show symmetry [[83 59 83] [59 83 83] [83 83 59]] 1 1
true
```

Final Short Perspectives on the Constructivist Approach

The main intention of our paper was to put the design of a different course which is partly going beyond Mathematics and Computer Science in school up for discussion. LOGO looms large in the concept which is rigid in no case but very open. It leaves multitudinous possibilities of constructionistic acting to committed teachers namely be it the discussion of additional attributes and operations as well as the introduction of further basic elements of Linear Algebra such as the determinant.

Furthermore the course offers numerous opportunities for inner differentiation wherewith teachers are able to make the grade to different levels of achievements through activities adapted to the students' capacities.

References

Fuchs, K. J. (2007). *Fachdidaktische Studien*. Shaker Verlag, Aachen.

Fuchs, K. J.; Siller, H. – S.; Vásárhelyi, É. (2008). *Informatics With Casio CP 300+*. CASIO Europe GmbH, Norderstedt.

Herber, H. – J.; Vásárhelyi, É. (2002). *Das Unterrichtsmodell „Innere Differenzierung einschließlich Analogiebildung“ – Aspekte einer empirisch veranlassten Modellentwicklung*. In Salzburger Beiträge zur Erziehungswissenschaft 6, Nr. 2, pp. 5−19.

Papert, S. (1993). *Mindstorms – Children, Computers and Powerful Ideas*. 2nd Edition Basic Books, New York.

Papert, S.; Solomon, C. (1971). *Twenty Things To Do With A Computer*. MIT, A. I. Laboratory, AI MemoNo. 248, LOGO Memo No. 3.

Schubert, S.; Schwill, A. (2004). *Didaktik der Informatik*. Spektrum Akademischer Verlag, Heidelberg.

Developing Algorithmic Thinking by Inventing and Playing Algorithms

Gerald Futschek, *futschek@ifs.tuwien.ac.at*
Institute of Software Technology and Interactive Systems, Vienna University of Technology

Julia Moschitz, *moschitz@ifs.tuwien.ac.at*
Institute of Software Technology and Interactive Systems, Vienna University of Technology

Abstract

In many cases at school and at universities most of the learners consider the topic of algorithms as hard and not very attractive. Very often the focus of traditional courses is on learning specific algorithms that are considered as important in education or in practice. Often these algorithms are sequential algorithms. We show in contrast to these courses a way of learning principles and concepts of algorithms that is much easier to comprehend by the learners and makes them more fun. The idea is that we do involve as many students as possible in playing algorithms that are moreover usually proposed by them.

The task of the teacher is to give proper problem statements and to ask proper questions to keep the students thinking to create working algorithms that solve these problems. The teacher also motivates the students to improve their algorithms to find more efficient solutions.

Compared to a theatre play the students have the roles of the actors and the idea deliverers and the teacher has the role of the stage manager.

We give two examples: The first example is the calculation of a maximal value of a set of values, where each student represents a value. Parallel activities may improve the efficiency of the algorithm. Usually the students find good solutions and learn a lot about concepts of sequential and parallel algorithms that are usually learned in advanced algorithm courses.

It is a form of explorative learning, where the students can experience algorithms by playing them and they can determine the progress and invent algorithms that they play.

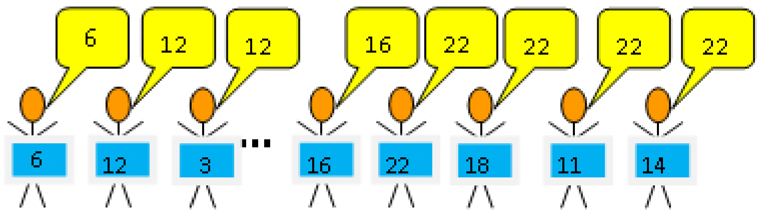


Figure 1. Sequential algorithm to calculate the maximum value. Each student passes the maximum_value_so_far to the next student in row.

The second example is named 'let's play robots', where students assume the roles of a robot and a navigator. By this way students learn more about basic algorithmic thinking while they are playing algorithms. A model of learning by inventing and playing algorithms is presented that proposes a cycle of 5 processes which the learners may perform for inventing algorithms.

Keywords

Algorithmic thinking, explorative learning, group learning, creativity

Motivation and Introduction

Algorithmic thinking is considered to be one of the main abilities that pupils may achieve in informatics education at school and university level. Algorithmic thinking as fundamental idea of informatics education is very complex and consists of a wide variety of abilities that can be understood at different intellectual levels, see Futschek (2006).

Furthermore finding and inventing appropriate algorithms is a necessary prerequisite of computer programming. Understanding algorithms is known as one of the difficulties students are confronted with when starting to learn programming, see Jenkins (2002).

To attract especially the younger and the newcomer to be more interested in computer science, it is very important to convey that algorithms are powerful tools that open a wide field of interesting activities where it is possible to achieve significant progress by providing new ideas.

There are several ways to help learners to understand principles of algorithms. Often animations of algorithms are used that are played by computer programs. Other approaches provide tasks and scenarios for learning principles of algorithms without using computers, see Bischof and Mittermeir (2008) and the Computer Science Unplugged activities from Fellows et al (2005). In these approaches the students play algorithms and get in this way a better understanding of given algorithms. In Futschek (2007) a learning scenario for playing counting algorithms is presented where a larger group of students can be involved in the play.

In this paper we present an approach where students invent algorithms to solve given problems. Because the students invent the algorithms they usually want to play their algorithms. While playing the algorithms the students find out the advantages and disadvantages of their algorithms and are highly motivated to discover necessary algorithmic concepts to improve their algorithms. A model for learning by inventing algorithms is presented.

Important Aspects of Algorithmic Thinking for Beginners

Algorithmic thinking is a special problem solving competence, which consists of several abilities, see Futschek (2006):

- analyze given problems
- specify problems precisely
- find the basic actions that are adequate to given problems
- construct correct algorithms to given problems using the basic actions
- think about all possible special and normal cases of a problem
- evaluate algorithms (correctness, efficiency, termination)
- improve the efficiency of algorithms

Algorithmic thinking consists amongst others of this wide range of abilities and is also influenced by many other human cognitive factors: abstract and logical thinking, thinking in structures, creativity and problem solving competence. This complexity makes algorithmic thinking not easy to learn and explains the need for a good didactic approach especially for beginners. However, this is not the only reason why beginners consider learning algorithmic thinking as hard.

Another reason is that algorithmic thinking at a certain point is an unnatural thinking type which has to be especially trained by a learner. In everyday life we often have to solve natural problems by algorithms where we have to find good solutions for having a better life. However, if we have to write a software program, we have to find a solution for a machine. This solution has to be made comprehensible for machines which usually like sequential instructions and have their own very basic instruction set. Humans in contrary like to cooperate and prefer parallel actions and have also a high level instruction set.

We think that especially for beginners the complexity should be reduced to that level where the concepts of algorithmic thinking can be learned in a natural way.

Therefore we need

- tasks that the learners know from daily life
- a natural description language for algorithms
- basic actions that the learners know from daily life
- a system that runs the algorithm
- a systems that allows the learners to experiment with the algorithm
- a system that gives immediate learning experiences
- a system that is flexible to run a variety of algorithms
- somebody who provides feedback

As system to run the algorithms we engage the learners themselves, they are playing the algorithms. The learners are intelligent processors that also run concurrent algorithms and can execute also natural high level commands. Therefore the tasks can be taken from daily life. As we will show in this article the learners can make fast learning progress in all abilities that constitute algorithmic thinking and they experience also advanced algorithmic concepts in a very natural and comprehensible way.

Important is that the problems to be solved are adequate to the pre-knowledge of the beginners. In the best way the given problems are from the students experience or from everyday life, because familiar examples can be comprehend immediately by students. These problems should be so general that they give way to a variety of different algorithmic solutions.

For students and their solution it is not necessary to know an exact language to describe algorithms (like programming languages). At the beginning the native language is adequate.

For beginners the knowledge of specific algorithms is not so important, but the ability to understand principles of algorithms and to find or create own algorithms for new problems. One main educational objective for beginners is to know that an algorithm prescribes exactly what to do in all possible situations. So, an algorithm prescribes not only the activities in the main situations but governs all possible situations. This can be experienced in a play that follows accurately the script of a self-invented algorithm.

The teacher has an important role, although he or she should take a back seat. The aim of the teacher is not to present solutions but to support students in their learning process, he should motivate students to make progress in finding solutions.

Learning by Playing Algorithms

Written algorithms are often too abstract for beginners to be understood. The students have to understand the syntax of the language that describes algorithms and also the idea how the algorithm solves the problem. Algorithms involve too many concepts, so a good learning approach needs a systematic stepwise strategy in a way the students understand why these concepts are necessary. We have the concepts of algorithm description languages, sequential and parallel algorithms, efficiency of algorithms, software agents, synchronization of parallel processes, broadcasting, shared variables, atomic actions, state transitions, correctness, etc. Playing algorithms is a good way to learn the principles more slowly and in a more effective way. We distinguish the following possibilities of playing algorithms:

- teacher is playing
- software is playing
- some students are playing
- all students are playing

We ordered these possibilities by its learning efficiency in learning. If the teacher demonstrates algorithms and shows what is going on, the students see that the teacher knows how the algorithms work but often it does not help them to get better understanding. The disadvantage is that students only watch the solutions and so they are in a passive role.

If a piece of software plays the algorithm the students can actively experiment with different inputs. Active learning is more efficient than passive learning. Students who play algorithms themselves get more insight in the algorithm details. The more students are involved in playing algorithms the better. Therefore our goal is to find problems where all students are involved in playing algorithms. If students are playing algorithms, they are in an active role. This method is time-consuming thereby effective learning is possible because “feelings, thinking, memories and physical sensations” are activated, see Siebert (2005).

The advantage of playing given algorithms is that the students see good solutions for problems and get inspiration for inventing other algorithms.

Model for Learning by Inventing Algorithms

Much more motivation and identification with the topic arises when the students get the possibility to invent own algorithms to solve a problem. Then playing the algorithms is more attractive for the students. A good choice of the problems to be solved is a prerequisite.

In this chapter we define a process of learning by inventing algorithms by dividing the process into five main steps based on the model of problem-solving thinking by Tümmers, see Seel (2005). The role of the teacher changes in this process from a traditional teacher to a coach taking a back seat. He is a supporter of the students. He motivates the students and states the original problem statement. By this way the students come to the front seats and are forced to be active. All five processes of figure 2 are done by the students. If necessary the teacher initiates the processes but he does not actively participate in finding, testing and improving solutions.

Important for this process are good problems according to age, previous knowledge and experience of the students.

Additionally, reflecting the learning process is a major part of this process. Often reflecting the learning process for students is something new and strange. However, on this way students learn to understand their programming problems by and by.

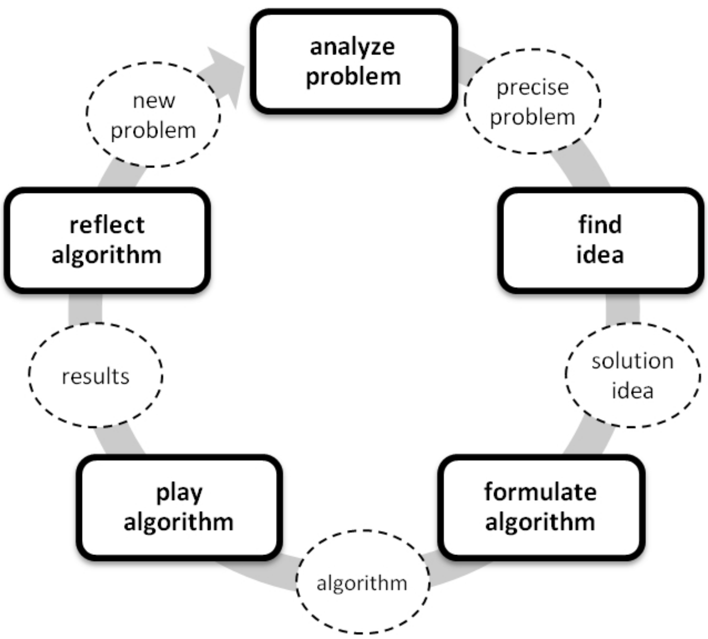


Figure 2. The process of learning by inventing algorithms

Analyze problem

The first step is to find out more details about the problem. In this process students try to find the main problem and to split it up into smaller problems.

Find solution idea

In this process the students have to be creative and should propose ideas, how they can solve the problem. If they have a problem, which is divided into smaller problem tasks, students can start to look for ideas for the smaller problems. They can write their ideas down or in a better way they discuss in groups their ideas. After the discussion they evaluate their ideas under the following aspects: Which idea can most properly solve the problem? Which idea can easily put into practice? Which of the ideas are efficient?

Formulate algorithm

In this step the goal is to write down or say a precise formulation of the ideas to solve the problem. For beginners the solutions can also be described orally. Advanced learners write down a precise formulation. At this time students have often problems to formulate specifically their solutions. Exchange of problems and questions with other students or the teacher can help solving this problem. Especially the basic actions should be clear.

Playing the algorithm

Main goal of playing the algorithm is to find out if it is working at all and how good it is working. Educational objective of this phase is to learn that not all ideas are working and to get a feeling for possible sources of failures.

Reflect algorithm

The purpose of reflecting the algorithm is to improve the solution. The results of the reflection are new problems to be solved. So the process starts again from the beginning. One possible reflection task is to find out the efficiency of the algorithm. Is it fast or slow, does it do unnecessary actions? In trying to give answers to these questions beginners can develop a feeling of the time complexity of an algorithm.

While learning in groups the students learn a lot from other students and their ideas. Furthermore motivation and assistance from the teacher help students getting ahead with their learning progress.

Examples

The following examples are taken from the authors teaching courses for students that are starting to learn about algorithms in tertiary education. The given examples show typical tasks and possible ways of teaching that were successful and may be taken to be used in a similar way in another learning context. The problem statement of the task to be solved should be very easily understandable and should give room for a wide variety of solutions. In Futschek (2007) the task of counting the number of a group of people was given and different algorithmic solutions were discussed. In our opinion this is the best first task for learning algorithms in a larger group (from about 10 up to some hundred students). All group members can be involved in counting activities and there is enough room for inventing efficient solutions. Here we want to give another example of this kind that is also very suitable for beginners.

Example: Maximum value

The task is to find the eldest or youngest student or to find the student with the highest student number or with the latest birthday within a year. We also tried this task where we distributed sheets with numbers on it. Very often some students know a sequential solution, where the students form a row of values:

> *"Beginning with the first value in the row one after the other compares its value with the maximum_value_so_far and if it is larger the maximum_value_so_far becomes this value. At the beginning the maximum_value_so_far becomes the first value in the row. After the last value in row has done his job the maximum_value_so_far holds the maximum of all values."*

Usually this solution is proposed if students know already the sequential solution and also if they are sitting in rows it is very natural to find a sequential solution.

Usually there emerge some minor problems while playing this algorithm. It is not always clear in what form the maximum_value_so_far is represented, so that the students that have to compare it with their own value have access to it. There are different possible solutions to this. First the maximum_value_so_far is always passed by the previous student to the next one. And the last one delivers the maximum value as result of the problem. Another solution is to involve a board, where the actual value of maximum_value_so_far is written and can be seen by all students. If these different solutions arise the teacher should discuss with the students the differences of these approaches. A board that is accessible and visible by all is not easy to handle if there are hundreds or even thousands of students. Who has writing access to the board?

Although this algorithm seems to be easy and clear, it is important to play the algorithm. The students can see that the activity moves from the beginning to the end and that there is a need to remember the maximum_value_so_far and that all players must have on their turn access to the actual value of the maximum_value_so_far.

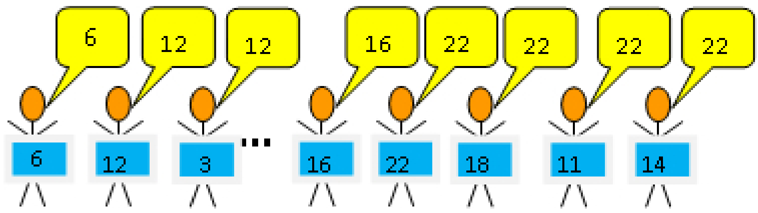


Figure 3. Sequential algorithm to calculate the maximum value. Each student passes the maximum_value_so_far to the next student in row.

The teacher makes the students to play their algorithms, and motivates them to write down the algorithms and poses questions to keep the learning process in progress. Possible questions of the teacher to induce an analysis of this algorithm by the students:

- Does each student run exactly the same algorithm?
- How do you know who is the next in a row?
- Does this algorithm always find the maximum value?
- How many steps does it take to find the maximum value?
- How would you solve this if you are standing in a group and there are no rows?

The last question leads to a nice solution where the students find ad hoc a next student who has not yet compared his value with the maximum_value_so_far. For this case the students have to indicate if they are still candidates for a next value or not, for example by standing instead of sitting. The following questions should motivate to find a more efficient solution:

- How many minutes does the algorithm take if there are a thousand students?
- Can the problem be solved faster?
- Are all players very active in this algorithm?
- Can the algorithm be improved by making the players more active?

Parallel Algorithms

A simple speed improvement often proposed by students is:

> *"Calculate in parallel the maximum of front half and the maximum of back half and deliver the maximum of these two."*

A question to clarify the improvement may be:

- How much is this parallel variant better than the sequential one?
- Are the students more active than before?

The teacher encourages the students to find a much faster solution. A possible idea to calculate the maximum much faster is as follows:

> *"Parallel calculation of maxima of rows. Then follows the calculation of the maximum of all row maxima."*

Usually this algorithm is suggested when the students are sitting in rows. It is nice to find out how much faster this algorithm is than the previous ones. If appropriate the teacher can address the problems of describing the efficiency of an algorithm. In this particular case the order of the algorithm is $\sqrt{N}$ if we have N rows and N columns.

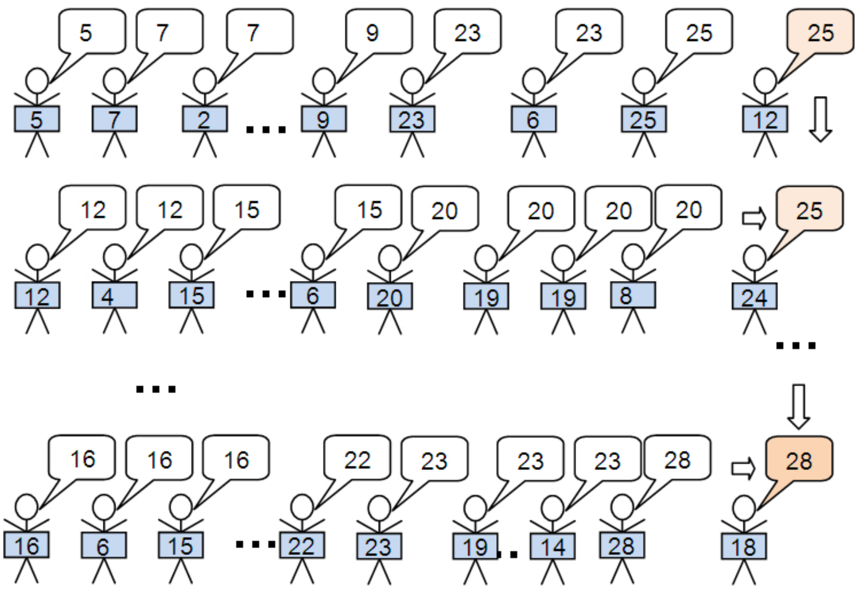


Figure 4. Parallel algorithm to calculate the maximum value. Each student passes the maximum_value_so_far to the next student in row. The students at end of each row calculate the maximum of three values: from side, from behind and its own value.

While playing this algorithm the students may find out that especially for the last in each row it is hard to coordinate getting values from the side and from behind that arrive at the same time. There arises the necessity to synchronize the two players that pass a value. So there is an agreement between the two players at what time both are ready to pass the value. A good synchronization technique is to look in each other eyes.

Sometimes students propose algorithms that need more clarification. For example the following parallel algorithm:

> "*All students compare in parallel their value with maximum_value_so_far and if their value is higher they replace the maximum_value_so_far by their own value.*"

Here it is necessary to clarify that comparing and replacing has to be a single atomic action to achieve a correct solution. While playing this algorithm the students can discover that replacing the maximum value cannot be done in parallel.

Another proposal of the students may be to exclude values that are smaller than other values:

repeat
choice of a value that is still in game
all values less than this value are removing themselves from the game
until only one value is remaining

Here a sort of broadcasting is necessary, since all students have to know the value that is selected.

Possible questions of the teacher to clarify additional problems:

- Does this algorithm always find a solution? Under what condition?
- How is it indicated that a value is still in the game (or is already excluded from the game)?

Another algorithm that may be proposed: All students have sheets with values that are readable also by other students.

All students show their sheet to the other students and all look in an arbitrary order at the sheets of the other students. When a student sees a sheet with a larger number than his own number, he removes himself from the game.

Questions:

- How many steps are needed to find the maximum at least, at most, in average?
- What are the basic actions?
- Do we need synchronization? If not, why?
- When does this algorithm terminate?

It seems to be important to involve all students in playing algorithms. Although the students have to follow accurately the given algorithm, they know better what is going on, how long it takes and where the problems arise. After each play they can analyze the play and can think about better or other algorithms.

The role of the teacher is not to write the script of the play. He proposes the problem to be solved and he is the play manager. Additionally he has teaching goals to fulfil, he can tell the students specific technical terms of informatics concepts of specific findings of the students.

Example: Let's Play Robot

The topic 'Robots' is fascinating for students, especially for young pupils. The topic "robots" is appropriate for projects at school, because it allows being creative in many ways. This fascination can be experienced by the students in the next example. This simple game is suitable for learners in small groups and is for learners from age 8 who have no or only a little experience in algorithmic thinking. By this game students learn describing algorithms exactly and in a specific order. Furthermore students learn to invent easy algorithms in small groups, they find out that not every solution works and they improve their already found solutions. The duration of this teaching method depends on the given exercises and takes between 15 and 60 minutes.

In practice this game is really simple and all students like assuming roles, even students at university level. Students assuming robots often think too active and show intelligent behaviour that is not part of the algorithm. Preventing this we give robots clothes to blindfold them. As a result of such games we observed also positive social skills effects on groups. The groups seem to be more communicative, have more confidence within the group and participate more active in the course.

The teacher asks the students to arrange in pairs. One of the pair assumes the role of the robot and the other one is the navigator of the robot. The teacher hands out papers to navigators with short instructions for the robot (forward, backward, put, …) and different exercises. This game can be played by all small groups at the same time. To begin this game the teacher gives a signal like “let’s play robot”. During the game students should be invited to change the roles. After the game the students present their solution to the others. Often beginners consider improving an algorithm as hard. To activate students thinking about improving their solution the teacher can ask questions like: Are you satisfied with your solution? Such questions help students reflecting their solution without anticipating a better solution (by the teacher).

Examples for tasks

- Task 1
  The first task is to navigate the robot through the room without touching something or somebody. Through this task the students learn the meaning of commands and to follow basic actions in a specific order.
- Task 2
  In the second task the navigator determines a point in the room and tries to navigate the robot to this point. Enhancing the level of the difficulty the navigator writes the algorithm before the robot starts. In the next level the algorithm should work for every point in the room. By this way students learn analyzing the problem, designing an algorithm and finding a solution for all possible special and normal cases of this problem.
- Task 3
  In this task the goal is finding the best solution for painting a star on a paper on the floor. The navigator tries to find an algorithm and gives the instruction to the robot. The robot paints a drawing according the instruction on the paper.
  In this task the students learn expanding their basic actions, because they have to find new instructions for the robot. Further they gain knowledge in analyzing a problem, finding solutions and get first experience in improving algorithms.

These examples of tasks show that the tasks can be adapted in many ways to the knowledge level of the learner. Ideas for other exercises can be found in books and publications about Logo or robots, because these exercises can be allocated also to the game “let’s play robot”.

Summary

Inventing algorithms is an effective learning method that can be done also with novices in algorithms. The students can play the algorithms to well chosen tasks of daily life and find out in a natural way even advanced algorithmic concepts like concurrency, synchronization, broadcasting, shared variables, etc.

In the proposed learning scenarios the students

- learn actively
- learn in groups
- govern the progress of learning
- are actors
- are script writers

and the teachers

- have to deal with unexpected proposals
- should have explored possible solutions before the lecture
- must be very firm in algorithms (analyse, create, not just replicate)
- are the play managers
- propose the original problem
- ask questions (that provoke often new sub-problems).

References

Bischof E. and Mittermeir, R. (2008) Informatik erleben (in German). Institut fur Informatik-Systeme, Alpen-Adria University Klagenfurt, see also http://informatik-erleben.uni-klu.ac.at/.

Fellows M., Bell T. and Witten I. (2005) Computer Science Unplugged, see also http://csunplugged.org/.

Futschek, G. (2006) *Algorithmic Thinking: The Key for Understanding Computer Science*. In Lecture Notes in Computer Science 4226, Springer, pp. 159 - 168.

Futschek, G. (2007) *Logo-Like Learning of Basic Concepts of Algorithms – Having Fun with Algorithms*. In Proceedings of Eurologo 2007. Edited by I. Kalas, Bratislava.

Jenkins T. (2002) *On the Difficulty of Learning to Program*. In Proceedings of 3rd annual Conference on LTSN-ICS, Loughbourgh, pp. 53-58.

Seel N. (2003) Psychologie des Lernens (in German), Ernst Reinhard Verlag München Basel.

Siebert H. (2005) Pädagogischer Konstruktivismus (in German), Beltz Verlag.

SLurtle Soup: a conceptual mash up of constructionist ideas and virtual worlds

Carina Girvan, *girvanc@tcd.ie*
Centre for Research in IT in Education, School of Education and School of Computer Science & Statistics, Trinity College, Dublin, Ireland

Brendan Tangney, *tangney@tcd.ie*
Centre for Research in IT in Education, School of Education and School of Computer Science & Statistics, Trinity College, Dublin, Ireland

Timothy Savage, *Tim.Savage@cs.tcd.ie*
Centre for Research in IT in Education, School of Education and School of Computer Science & Statistics, Trinity College, Dublin, Ireland

Abstract

In *Mindstorms*, Papert (1980) advocated "*the construction of educationally powerful computational environments that will provide alternatives to traditional classrooms and traditional instruction.*" (p. 182). At the same time he identified that the technology of the day was limited in its capabilities and functionality. Since then considerable work has been done to create tools ranging from Logo, Mindstorms, Scratch, ToonTalk, etc which embody a constructionist approach to learning. In this paper we introduce a new embodiment of constructionist theory, SLurtles (programmable Turtles in *Second Life*).

The virtual world *Second Life* provides a high floor high ceiling building and programming interface difficult for the novice to master. To address this problem, SLurtles leverage concepts from Turtle and Turtle geometry; *Mindstorms*; and *Lego*, and combines them with *Scratch for Second Life* (a low floor, high ceiling programming environment for *Second Life),* to create a low floor, high ceiling programmable building tool for *Second Life*.

This paper presents a use case example and description of ideas which underpin the SLurtle conceptual mash up. This is followed by an outline of our initial trial with learners using SLurtles to experience constructionist learning in a virtual world, with an overview of our research aims.

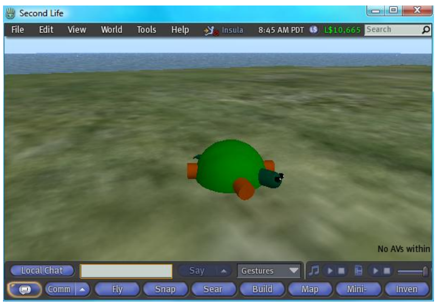


Figure 1. SLurtle, a programmable Turtle in Second Life

Keywords

Turtle geometry; Mindstorms; Scratch for Second Life; Constructionism; Virtual Worlds; Second Life

Introduction

In *Mindstorms*, Papert (1980) advocated "*the construction of educationally powerful computational environments that will provide alternatives to traditional classrooms and traditional instruction*." (p. 182). While he accepted that the tool he was using, Logo, was limited by the capabilities and functionality of the technology of the 1970s, it nevertheless went on to give rise to a rich vein of research based on the notion of 'objects-to-think-with', with *Scratch* (Maloney et al., 2004) and *ToonTalk* (Kahn, 2001) prominent examples. These 'objects-to-think-with' provide the learner with an easy to access, 'low floor', entry to programming, whilst allowing the more experienced user create highly complex algorithms, resulting in a 'high ceiling' (Sheehan, 2000).

Virtual worlds, such as *Second Life*, provide three-dimensional, persistent and flexible environments that can provide an alternative to traditional educational locations and approaches. While the automatic reaction to replicate what has gone before in a new technology (Winn, 2005) has been observed in virtual worlds with the creation of in-world lecture theatres, etc, some educators have begun to explore the potential of this technology and the new educational opportunities it can support.

Much like the Lego/Logo 'behaving machines' that could be created in the real world (Resnick, 1993), artefacts which are interactive and exhibit behaviours can be created in the virtual world. Content creation in the virtual world of *Second Life* begins with a simple object (prim), which can be manipulated in numerous ways and combined with others to create sophisticated structures. However the skills required to create even a simple artefact, such as a staircase, results in a high step to entry. Furthermore, in order for this simple artefact to exhibit behaviours requires the object to be programmed. Programming within *Second Life* requires the use of Linden Scripting Language (LSL), a high floor, high ceiling programming language with C style syntax.

Based on *Scratch* (Maloney et al., 2004; Resnick et al., 2009), Eric Rosenbaum (2008) developed *Scratch for Second Life* (S4SL) as a low floor programming environment for *Second Life*. S4SL provide the learner with an opportunity to programme and introduce behaviours to otherwise static objects in *Second Life*, much like the Lego/Logo 'behaving machines'.

This paper presents SLurtles (programmable Turtles in *Second Life*) as the embodiment of constructionist theory within a virtual world. SLurtles leverage concepts from Turtle and Turtle geometry, *Mindstorms* and *Lego*, combining them with S4SL within the virtual world of *Second Life*, to provide new 'objects-to-think-with' within the new learning environment of virtual worlds. S4SL provides the learner with a low floor, high ceiling programming environment to programme the SLurtles. The SLurtles in conjunction with S4SL provide the learner with a constructionist low floor, high ceiling tool for the construction of objects in *Second Life*.

Following a brief discussion on virtual worlds and the Logo heritage, this paper presents a use case example and description of SLurtles. This is followed by an outline of our initial trial with learners using SLurtles to experience constructionist learning in a virtual world, with an overview of our research aims.

Background

Virtual Worlds

Virtual worlds are typically characterised as persistent, three-dimensional, immersive environments (Castronova, 2005) which provide opportunities for users to collaborate and share experiences without the need for physical co-presence. Users are represented by avatars which can interact with their environment and communicate with others through a range of communication tools. This supports the user's perception of immersion and co-presence with other users. Some virtual worlds also provide the opportunity for users to create content. The construction of static objects which can also be programmed is available to users of *Second Life*, limited for users of *There* but unavailable to users of *Club Penguin*.

Replication

As Winn (2005) describes in his work on virtual reality, the natural reaction of the early adopters of a technology is to replicate what has gone before in the new environment. Thus in virtual worlds we see replicated lecture theatres and university campuses. While this may provide some advantages for distance learners, there is a need to move beyond what can be replicated and begin to explore and innovate in our use of virtual worlds for educational purposes.

Increasing numbers of learning experiences reported in the literature are making links between the features of virtual worlds and the learning approaches adopted. For example, role play can leverage the sense of immersion arising from the use of avatars, communication tools and the 3D environment (Jamaludin et al. 2009). There is also opportunity for experimentation without real-world repercussions (Dede, 1995; Burmester et at., 2008) as well as the creation of learning experiences which could not easily be achieved in the real world (Good et al., 2008).

Despite the increasing number of reported learning activities which are beginning to explore beyond simple replication, Savin-Baden (2008) notes that these learning experiences often lack pedagogical underpinnings. We posit that to design learning experiences that have strong pedagogical underpinnings we first need to identify those pedagogies which can strongly leverage the unique combination of affordances that virtual worlds can provide (Girvan & Savage, 2010). These pedagogies then need to be explored in action, as part of a carefully designed learning activity experienced by several groups of learners, to identify how the pedagogy manifests and leverages the affordances of a virtual world.

Affordances

Within the literature on virtual worlds, ‘affordance’ is a widely used but often undefined term. The affordances of an object are the actions that an individual can conceive of as possible when interacting with that object based upon a visual impression (Gibson, 1979). For example, based on the perception of a sharp edge an individual can envisage that a knife affords cutting. We specifically focus on Norman’s (1999) description of ‘perceived affordances’ which requires us to acknowledge that what may be perceived as an affordance may differ from person to person. Those affordances which may support educational activities are described by Kirschner (2002) as ‘educational affordances’. Thus, those affordances, as perceived by users, which could be leveraged for educational purposes can be described as ‘perceived educational affordances’ (Girvan & Savage, 2010).

Our current exploration of pedagogies within virtual worlds focuses on *Second Life*. Based on the literature and our experience as educators within virtual worlds, we can identify that the three-dimensional landscape and representation of avatars within *Second Life*, combined with the opportunity for interaction through communication tools, affords a sense of self and presence. This sense of self and presence can in-turn result in immersion and support socialisation and collaborative learning (Kemp & Livingstone, 2006; Cross, O’Driscoll & Trondsen, 2007; Minocha & Roberts, 2008). *Second Life* also provides tools that afford build and rebuild opportunities for the construction and programming of objects and environments (Delwiche, 2006) which are persistent (Castronova, 2005), interactive and flexible. However, these tools, while providing a high ceiling, present the learner with a high floor to access.

Second Life and constructionism

With regard to the perceived educational affordances of *Second Life* outlined above, constructionism appears to be a potentially appropriate pedagogy to underpin the design of learning experiences in virtual worlds. Central to constructionism is the designing and construction of shareable artefacts as an opportunity for the learner to actively explore, experiment and extend their understanding. Hoyles et al. (2002) also note that programmability remains an essential facet of constructionism. These core features can leverage the building and scripting tools in *Second Life* which afford build and rebuild opportunities in the construction and programming of objects and the environment. As virtual worlds are persistent, any object created by a learner can remain in the space it was created, without requiring the user's presence. This provides an opportunity for learners to create artefacts which can be returned to and shared.

The sense of self and presence, resulting in an opportunity for socialisation and collaborative learning, can also be leveraged to support the sharing of artefacts. We suggest that immersion may support the creation of personally meaningful artefacts and result in a sense of interacting and observing artefacts more directly. However, there is little understanding of how the use of an avatar or the variety of communication tools and sense of presence and immersion could impact the pedagogy in action.

Based on our current understanding of the literature on virtual worlds, there have been few studies which have placed constructionism at the heart of the learning experience. Both Dreher et al. (2009) and Good et al. (2008) report on learning experiences within *Second Life* that use constructionism but lack a description of how constructionist learning was supported during the learning activities. There is limited discussion as to how the specific features of the technology support the constructionist learning activities and a lack of results or findings that enhance our understanding of the pedagogy in action within virtual worlds.

Within the literature on tools which facilitate constructionist learning, Kahn (2007) considers the potential of virtual worlds such as *Second Life*. His particular focus is on constructing programmes within these environments similar to *ToonTalk*. The potential additional benefits he identifies are based on the opportunity to leverage the impression of a three-dimensional environment, realistic physics, collaboration and 'inhabited spaces'.

From Turtle to Scratch

Logo was designed to be an easily accessible (low floor) and at the same time, powerfully expressive (high ceiling) programming language providing opportunities for learners to construct, explore and investigate (Feurzeig, 2007). Before Turtle, Logo was used to write poetry, create translators or construct strategy games (Papert, 1980). The Turtle provided an opportunity to engage learners across a much wider age spectrum with an 'object-to-think-with', whether sharing a physical presence with the learner or as an image on the screen which produces colourful lines. 'Objects-to-think-with' provide a focal point for the "intersection of cultural presence, embedded knowledge, and the possibility for personal identification" (Papert, 1980, p.11), whilst being shaped and even split due to differing cultural and political contexts (Agalianos et al., 2006)

Turtle geometry was proposed as a computational style of geometry, unlike Euclid's logical style (Papert, 1972). Within Euclidian geometry there are a number of concepts, one of which is a 'point' which has no properties other than 'position'. Instead of a point, Turtle geometry uses a 'Turtle', whether physical or on a screen, to draw lines. Similar to a point, a Turtle has a position but it also has a 'heading' resulting from the direction it is facing (Papert, 1972; 1980). The Turtle 'object-to-think-with' thus provides the entry point to Turtle geometry which is dependent upon both position and heading.

Since the early days of Logo and Turtle geometry, the concept of low floor, high ceiling programming languages has continued to result in the development of languages and micro-worlds such as Scratch (Maloney et al., 2004). Scratch provides a good example of how these later instantiations of constructionist 'objects-to-think-with' began to widen the walls, supporting a variety of projects that could be realised dependant on the interests and learning styles of the user (Resnick et al., 2009).

The design of a constructionist tool for a virtual world: SLurtles

SLurtles were designed and created to provide a constructionist 'object-to-think-with' for use within *Second Life,* which leverage the perceived educational affordances of virtual worlds. SLurtles are designed as a low floor, high ceiling tool for the creation of objects, used in conjunction with *Scratch for Second Life* (S4SL).

This section presents a SLurtle use case example to provide context to the following detailed description of the appropriation of Eric Rosenbaum's (2008) S4SL and modification of 'lineSegment' in the design and creation of SLurtles.

SLurtles in action: A use case

SLurtles can be programmed to create and place virtual blocks in patterns and sequences to construct sophisticated artefacts within the virtual world of *Second Life*. This programming is undertaken using the low step interface of *Scratch for Second Life* and imported into the SLurtle. A range of SLurtles have been constructed each of which contains and lays a unique style of block

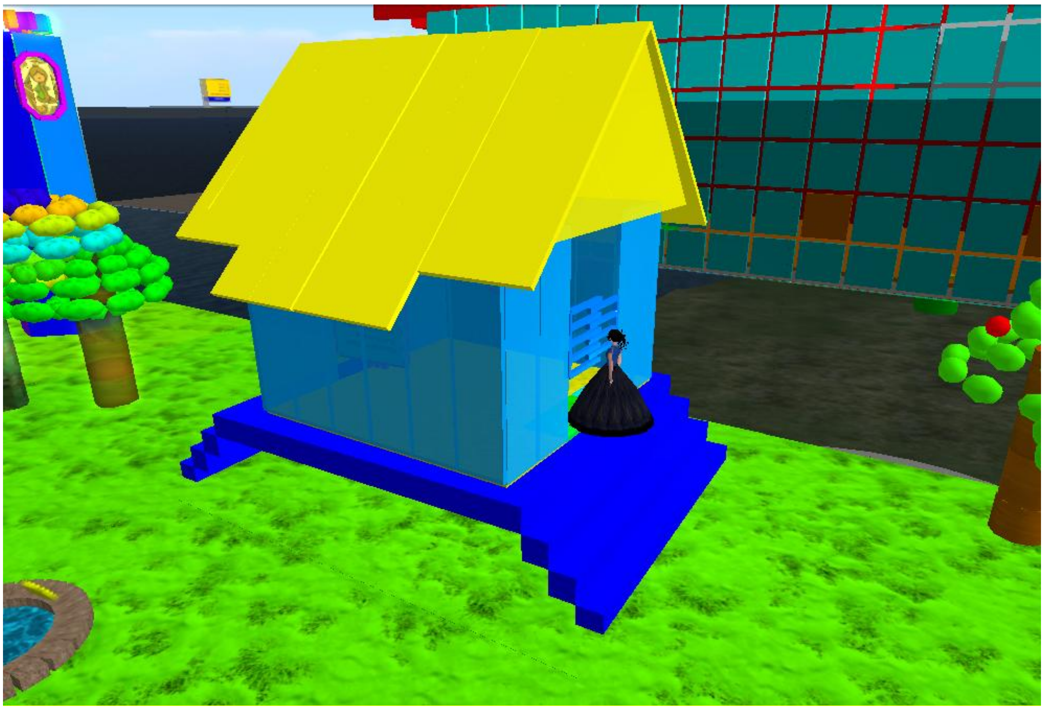


Figure 2. Witch's house in the Enchanted Forrest installation

The witch's house and trees shown in figure 2 were each created using programmable SLurtles. The house was constructed using six SLurtles, each which created different shaped blocks. Each SLurtles was programmed using S4SL to position, colour and place each block, with the length of each block determined by the distance the SLurtle travelled. In a similar way each of the trees shown was created using two SLurtles. The first SLurtle created cylinders and was programmed using S4SL to create the trunk of the tree. The second SLurtle created spheroids and was programmed using S4SL to create the green leaves and randomly place red spheres. Both the witch's house and trees, once created were then programmed to exhibit behaviours.

For example the red spheres, which represented apples on the trees, were programmed using S4SL to shrink and change colour when touched, accompanied by the sound of a witch's cackle.

Scratch for Second Life (S4SL)

As previously stated the tools which support the creation and programming of objects in *Second Life*, while powerful, can be difficult to use for those without previous experience. Object creation is essentially an exercise in 3D modelling and the Linden Scripting Language (LSL), with its C-style syntax and structure, has a high barrier to entry, particularly for someone with no previous programming experience.

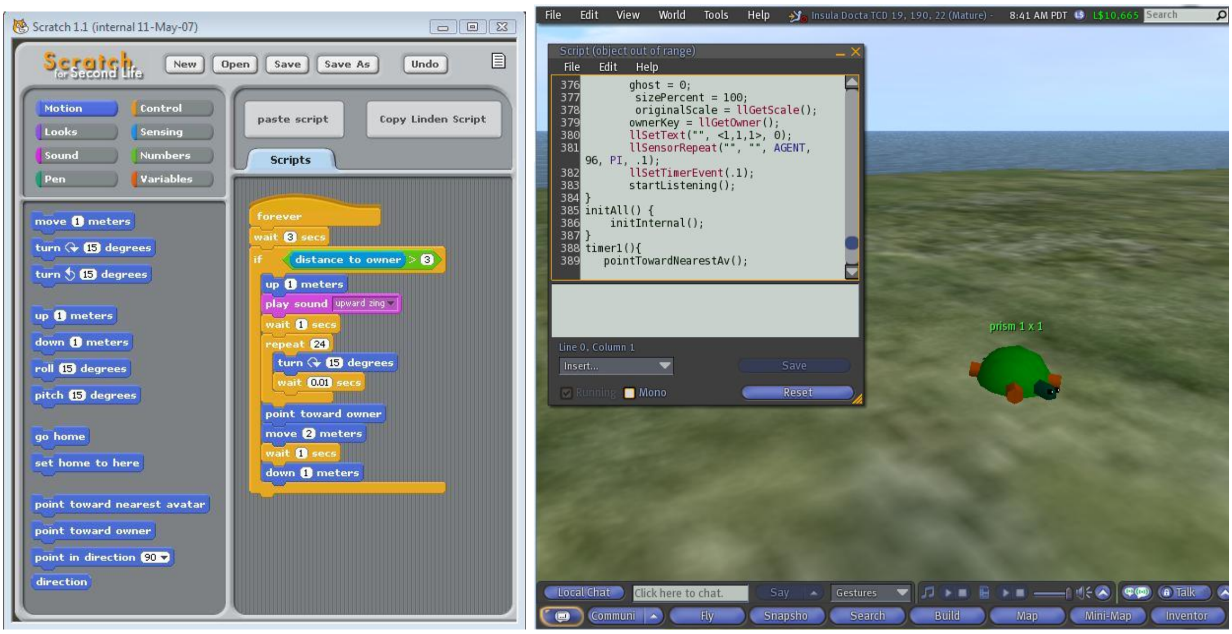


Figure 3. Scratch for Second Life interface and Second Life client with open LSL script and SLurtle

Based on the graphical programming language, *Scratch* (Maloney et al., 2004; Resnick et al., 2009), *Scratch for Second Life* (S4SL) was designed by Eric Rosenbaum (2008) as a low floor programming environment to help lower the barrier to adding behaviours and interactivity to otherwise static objects in *Second Life*. Unlike Kahn's (2007) focus on constructing programming tools within the virtual word, S4SL is a separate programme which runs outside of the virtual world. It provides the traditional Scratch visual programming environment in which graphical blocks are snapped together to create a programme which is then exported into equivalent LSL code. All that is required of the user is that they select the "Copy Linden Script" button to generate the LSL code and then paste that into a new script within an object created in *Second Life.* Figure 3 shows the S4SL environment and a section of the LSL code generated which is placed inside a SLurtle.

Pen down

The S4SL blocks which control behaviour originate from *Scratch,* are fairly intuitive and include categories for 'motion, 'control', 'looks', 'sensing', 'sound', 'pen', 'variables' and 'numbers'. The following discussion will focus on the ones that control the "pen".

In the traditional *Scratch* programming environment the 'pen down' command results in a 2D line drawn on the stage, behind the sprite as it moves. This feature in turn is influenced by Turtle geometry (Papert, 1972) and the ability to create programmes which direct the actions of a physical or on-screen Turtle to create line drawings. In S4SL the 'pen down' legacy continues however LSL and *Second Life* have no 'pen' equivalent. Instead, Rosenbaum has created a work-around for S4SL which leverages the affordances of the building tools in *Second Life*. A 'lineSegment' object, a 0.1 meter wide by 0.1 meter high cuboid, can be placed within a *Second Life* object. This containing object can be scripted using S4SL and when 'pen down' is called this causes an instance of the 3D 'lineSegment' to appear (be drawn) within the virtual world. The location and length of the new object are determined by where the parent object was when the 'pen down' command was issued and how far it travelled in one action. For example, figure 4 shows some S4SL code and what happens in *Second Life* after an avatar has clicked on the object (the wooden cube) which contains both the LSL script generated by S4SL and a 'lineSegment'.

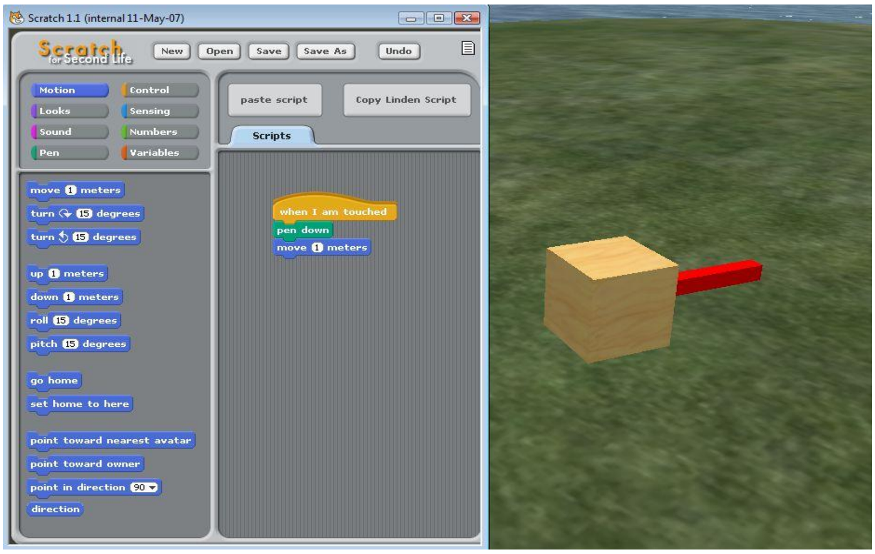


Figure 4. S4SL blocks and result of the programme in Second Life, object producing a 'lineSegment'

Each time the cube is clicked it will create a new instance of 'lineSegment'. Unlike the *Scratch* or Turtle pen marks which merely draw lines, each shape created using the pen in S4SL is an object, although only temporal.

SLurtles

To the learner, the cube in figure 4 when first created is much like the Euclid point. It can be observed to have a position but no obvious heading. When programmed to move forward it will move forward but to the learner it is not clear what direction this will be until the programme is executed. In addition, S4SL users must explicitly embed a 'lineSegment' in an object so that those objects can respond to 'pen up/down' commands. We have taken S4SL and the notion of 'lineSegments' in objects one step further by making more explicit the link to Turtle and by drawing on ideas from Lego building blocks and *Lego Mindstorms*.

We have created SLurtles (programmable Turtles in Second Life, shown in figure 1) which behave in a similar way to Turtle. A SLurtle has an explicit notion of heading and responds to 'pen up/down' commands by placing a SLurtle block. The learner does not need to know about the embedded object but by programming SLurtles using S4SL, sophisticated objects can be created in the virtual world.

The SLurtle block is an adapted 'lineSegment'. As stated above, the original 'lineSegment' was temporal lasting a short time before disappearing. However, by leveraging the persistence affordance of *Second Life* the SLurtle block is a permanent object, thus SLurtles provide an opportunity to lower the barrier to object creation in *Second Life*.

Borrowing from the idea of Lego bricks a variety of different SLurtles are provided which create different shaped blocks including cuboids and spheroids of different height and widths.

Current work

By combining S4SL with concepts from Turtle geometry and Lego bricks, SLurtles become an 'object-to-think-with' (Papert, 1980) within a virtual world. They also provide an opportunity to explore constructionism in action within a virtual world.

An initial user trial of SLurtles has recently been conducted with postgraduate students on a technology and education course. Students on the course have a wide range of backgrounds encompassing computer scientists with an interest in education and school teachers with an interest in using computers more effectively in their own classroom. As part of the course learners are introduced to a range of pedagogical theories and technologies.

This year, as part of their work on virtual worlds, learners were introduced to SLurtles following a brief introduction to *Second Life* as well as lectures and workshops on constructionism, using *Mindstorms* and *Scratch*. Following a workshop session with SLurtles which had an emphasis on using Kolb's (1984) learning cycle to support the explorative process, students collaborating in pairs, used SLurtles to create an installation which in-turn was programmed using S4SL to be interactive as part of their course assessment.

The user trial has provided us with an opportunity to garner feedback on the learners' experiences with SLurtles and their thoughts on its design. It has also provided us with an opportunity to begin exploring our research questions about constructionism in action within a virtual world. During the learning experience participants recorded their text based conversations using the chat logging feature in *Second Life*. Following completion of the assignment participants completed a questionnaire and were invited to take part in a semi-structured interview. Finally their interactive installations and reflections submitted for the assignment were collected.

Qualitative data analysis is currently underway and a number of positive and unexpected outcomes are beginning to emerge. Overall participants appear to have enjoyed using SLurtles to construct their installations within the virtual world, achieving much more than they had had first expected possible. The Witch's house and trees shown in figure 2 comprise part of one group's installation of an enchanted forest. Other groups created a piano, obstacle course, an interactive animation of the story of the Three Little Pigs, a bowling alley and an abstract optical illusion.

Discussion

The following discussion briefly considers some of the potential applications, opportunities and constraints of SLurtles as presented in this paper.

Potential applications

As presented in the previous section, SLurtles provide an opportunity for learners to experience constructionism in action for themselves. There is potential, through the use of closed access virtual worlds such as OpenSim, for using SLurtles to teach geometry within the K-12 education sector. Finally, SLurtles may provide a approach for learners to explore and learn about the abstract concepts of an initial computer programming course through creating concrete objects rather than focusing on the syntactic complexities of code.

Persistent objects

The original 'lineSegment' for use with S4SL was a temporal object once placed. This limits the opportunity to share artefacts and significantly restricts opportunities to observe and reflect on constructions and part of a constructionist learning experience. Leveraging the persistent nature of *Second Life*, we have altered 'lineSegment' so each instance is no longer temporal. The persistent instances of 'lineSegment' provide an opportunity for learners to observe and reflect on their constructions. They are able to log out of *Second Life* and their creations remain for others. In addition they can alter their S4SL code, return to *Second Life* and create a new object which can be compared side-by-side with their previous construction. As a result SLurtles can be programmed and reprogrammed to create persistent artefacts which can be shared with others.

Lowering the ceiling

SLurtles lower the floor for construction within virtual worlds. However, while it leverages the affordances of *Second Life* and S4SL it is also constrained by them. LSL, while a complex programming language for novices does provide a very high ceiling with a wide range of functions available. S4SL only provides the user with a selection of these functions, limiting the functionality that can be obtained and thus lowering the ceiling. It would, however, be possible to overcome this with further development of S4SL.

Widening the walls

While SLurtles lower the floor for construction, they may also widen the walls as evidenced by the wide range of installations created by learners in the initial user trial. The variety of shapes and forms of those shapes that can be created by the SLurtles, supports the creation of a wide range of objects in the virtual world by the novice user.

Acknowledgements

The authors would like to thank members of the Centre for Research in IT in Education, Trinity College, University of Dublin, as well as the first year students of the Masters in Technology and Learning, for their support during this work. This research is supported by 'Embark Initiative' funding from the Irish Research Council for Science Engineering and Technology.

For a copy of SLurtle, please contact Carina Girvan at the address above or Sleepy Littlething in Second Life.

References

Agalianos, A., Whitty, G. Noss, R. (2006) *The social shaping of Logo*. Social studies of science 36, 241-267

Burmester, A., Burmester, F. & Reiners, T. (2008) *Virtual Environment for Immersive Learning of Container Logistics.* In Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2008 (pp. 5843-5852). Chesapeake, VA: AACE.

Castronova, E. (2005) *Synthetic Worlds*. Chicago, The University of Chicago Press

Dede, C. (2004) *If Designed Based Research is the Answer, What is the Question?* The Journal of the Learning Sciences, 13(1), 105-114.

Dreher, C., Reiners, T., Dreher, N., & Dreher, H. (2009). *Virtual Worlds as a Context Suited for Information Systems Education: Discussion of Pedagogical Experience and Curriculum Design with Reference to Second Life.* Journal of Information Systems Education, 20(2), 211-224

Feurzeig, W. (2007) *Towards an culture of creativity: A personal perspective on Logo's early years, legacy, and ongoing potential*. In Proceedings of EuroLogo 2007. Edited by I. Kalaš, Bratislava, August 2007

Gibson, J. J. (1979) *The ecological approach to visual perception.* Boston: Houghton Mifflin Company

Girvan, C. & Savage, T. (2010) *Identifying an appropriate pedagogy for virtual worlds: A Communal Constructivism case study.* Computers & Education, 55(1), 342-349.

Good, J., Howland, K. & Thackray, L. (2008) *Problem-based learning spanning real and virtual worlds: a case study in Second Life.* ALT-J, 16(3), 163-172.

Hoyles, C., Noss, R. & Adamson, R. (2002) *Rethinking the Microworld idea.* Journal of Educational Computing Research, 27(1), 29-53.

Jamaludin, A., Chee, Y. S. & Ho, C. M. L. (2009) *Fostering argumentative knowledge construction through enactive role play in Second Life.* Computers & Education, 53(2), 317-329.

Kahn, K. (2001) *ToonTalk and Logo – Is TookTalk a collegue, competitor, successor, sibling, or child of Logo?* In Proceedings of EuroLogo 2001. Edited by G. Futschek. Linz, August 2001

Kahn, K. (2007) *Should LOGO keep going forward 1?* In Proceedings of EuroLogo 2007. Edited by I. Kalaš . Bratislava, August 2007

Kirschner, P. A. (2002). Can we support CSCL? Educational, social and technological affordances for learning. In P. A. Kirschner (Ed.), *Three worlds of CSCL: Can we support CSCL* (pp.7-47). Heerlen: Open University of the Netherlands.

Kolb, D. (1984). *Experiential learning: experience as the source of learning and development*. Prentice-Hall.

Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B. & Resnick, M. (2004) *Scratch: A sneak preview*. In Y. Kambayashi, K. Tanaka & K. Rose (Eds.), Proceedings of the Second Intyernational Conference on Creating, Connecting, and Collaborating Through Computing (pp.104-109). Kyoto: Kyoto University.

Papert, S. (1972) *On making a theorem for a child.* In Proceedings of the ACM annual conference, August, 1972, (pp.345-349) Boston, Massachusetts, United States

Papert, S. (1980) Mindstorms: Children, computers and powerful ideas New York: Basic Books

Resnick, M. (1993) *Behaviour construction kits.* Communications of the ACM, 36(7), pp.64-71

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastwood, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. & Kafai, Y. (2009) *Scratch: Programming for all.* Communications of the ACM, 52(11), 60-67

Rosenbaum, E. (2008) *Scratch for Second Life.* In S. Veeragoudar Harrell (Chair & Organizer), Virtually there: Emerging designs for STEM teaching and learning in immersive online 3D microworlds. Symposium in Proceedings of the International Conference on Learning Sciences – ICLS 2008. Utrecht, The Netherlands: ICLS. Abstract retrieved February 1, 2010, from http://www.fi.uu.ul/en/icls2008/144/paper144.pdf

Savin-Baden, M. (2008) *From cognitive capability to social reform? Shifting perceptions of learning in immersive virtual worlds.* ALT-J, 16(3), 151-161

Sheehan, R. (2000, 10-13 Sept.). *Lower floor, lower ceiling: easily programming turtle-graphics.* Proceedings of the IEEE International Symposium on Visual Languages (VL'00), Seattle, WA, USA.

Winn, W. (2005) *What we have learned about VR and learning and what we still need to study.* In Proceedings of Virtual Reality International Conference 2005. Retrieved March 14, 2008, from http://depts.washington.edu/edtech/laval.doc

Bringing "No Ceiling" to Scratch: Can One Language Serve Kids and Computer Scientists?

Brian Harvey, *bh@cs.berkeley.edu*
Computer Science Division, University of California, Berkeley

Jens Mönig, *jens.moenig@miosoft.com*
Enterprise Applications Development, MioSoft Corporation

Abstract

Scratch (http://scratch.mit.edu) is a computer programming language for children, with a graphical drag-and-drop user interface. It is a descendent of Logo, developed at the MIT Media Lab. A small but growing trend among universities is to develop computer science courses for non-majors using Scratch as the programming environment, because it isn't threatening—the same reason it works for kids. Also, the visible use of multiple threads in Scratch provide a simple introduction to parallelism. One such course was piloted this year at the University of California, Berkeley: "The Beauty and Joy of Computing."

But Scratch has weaknesses as a programming language. Most notably, it lacks procedures, so it can't convey the impressive phenomenon of recursion, one of the central ideas of computer science (and also one of the central ideas of early Logo pedagogy). Its support for data structures is also weak. These weaknesses aren't oversights; the designers of Scratch deliberately avoided cluttering the language with anything a child might find threatening.

To serve these two audiences, it has been proposed to split the Scratch community with two versions of the language, one for kids and one for advanced users. We believe that this is not necessary. By taking key ideas, such as procedures as first class data, from the Scheme language, we can add only a few features to Scratch and still make it powerful enough to support a serious introductory computer science curriculum. Furthermore, the graphical interface of Scratch makes the reification of procedures as data seem much less abstract and intimidating to novices.

Here is an example of writing a higher-order function MAP and using it with a reified procedure as an argument:

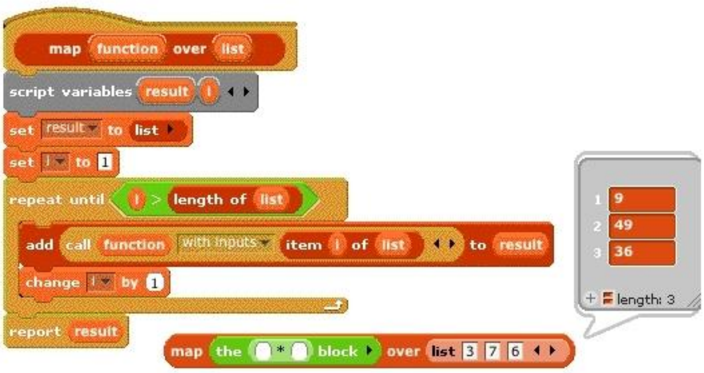


Figure 1. MAP function definition and an example of its use

Keywords

Scratch; computer science; lambda; first class procedures; education; programming

Introduction: Scratch

Scratch (Resnick et al., 2009) is a programming language for children in which no keyboard skill is needed, because the primitive program elements are available in drag-and-drop menus:

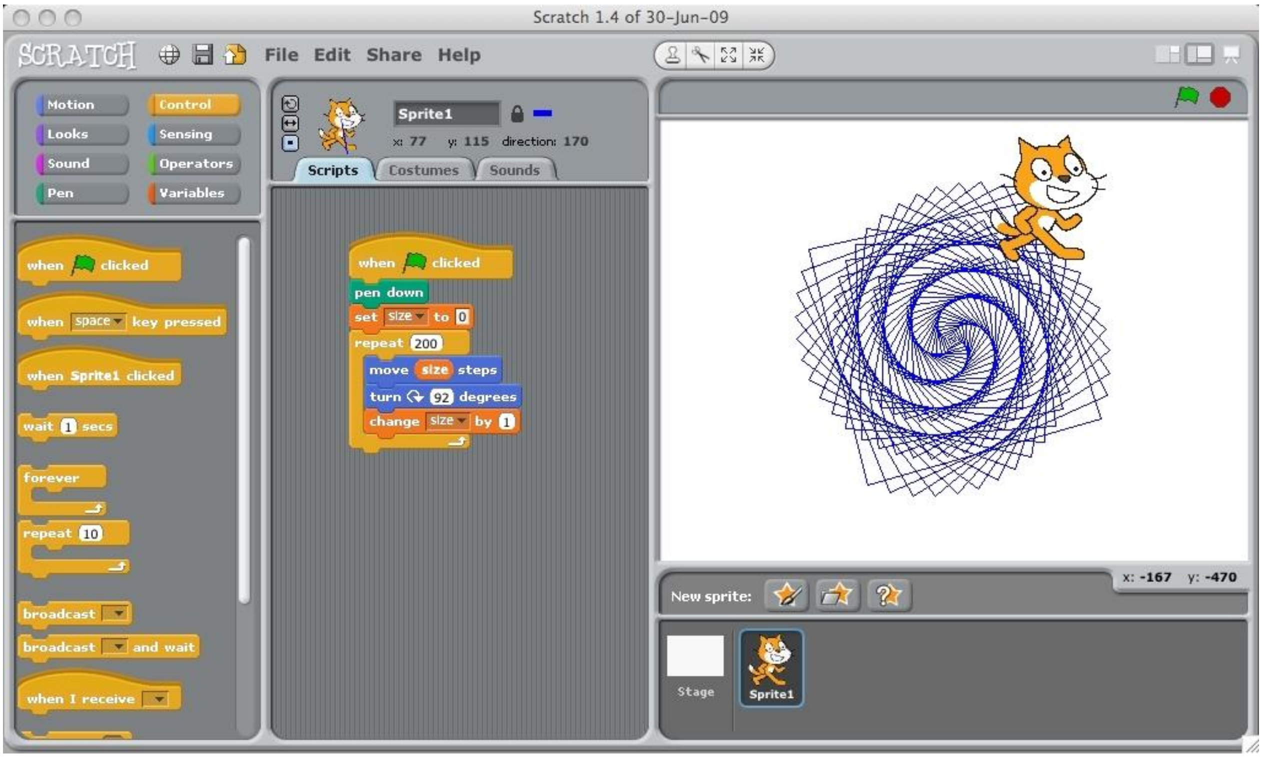


Figure 2. The Scratch menus, scripting area, and stage

Scratch illustrates object oriented programming in the form of multiple animated sprites with shapes taken from its own library or imported from any picture file. Each sprite has its own script area and its own local state variables. A “broadcast” primitive provides a rudimentary form of message passing, but with the limitations that messages can’t be directed to an individual sprite, and the corresponding method scripts can’t take arguments or return values.

These limitations are a deliberate part of the Scratch design, which has a a primary goal that every aspect of the language should be intuitive even to young children. As another example, until version 1.3, Scratch had no data aggregation mechanism at all; the lists added in 1.3 are, by default, visible on the stage, so that their behavior is apparent. Deep structures (lists of lists) are ruled out.

Scratch goes to the university.

In recent years, computer science departments at several universities have been making efforts to attract more students. One motivation for this has been the underrepresentation of women and minorities among computer scientists; in recent years a more general motivation has been a decline in the total number of computer science majors, even as job opportunities in the field have been increasing. Some countries, including the United States, have declared solving the shortage of students in scientific fields and computer science specifically to be a national priority.

The effort has taken many forms. One example is a shift from a narrow focus on programming techniques to a “breadth first” or “applications first” curriculum. But the form relevant to this paper is the search for a programming language that avoids the syntactic complexities that drive away beginners. Scratch is one of several languages trying to meet this need; others are Alice (Pausch, 1995) and Stagecast (Smith and Cypher, 1998).

Courses using Scratch have been developed at Harvard University (Malan and Leitner, 2007), and the College of New Jersey (Wolz et. al., 2008). One of the authors of this paper (Harvey) is developing such a course along with Daniel Garcia and a team of students led by Colleen Lewis at the University of California, Berkeley. A pilot version of the course was taught in Fall, 2009 (Shafer, 2009).

Berkeley's approach to introductory computer science has been profoundly shaped by the seminal text *Structure and Interpretation of Computer Programs* (Abelson and Sussman, 1996). In our new course, we wanted to preserve the big ideas we gleaned from their curriculum, including recursion and higher order procedures as organizing tools for flow of control. But Scratch, without the ability to define procedures, wouldn't support such a curriculum.

Build Your Own Blocks

Fortunately for the Berkeley effort, the other author of this paper (Mönig) developed an extension to Scratch called BYOB (Build Your Own Blocks) that solved the first problem (recursion) by allowing users to create new Scratch procedures:

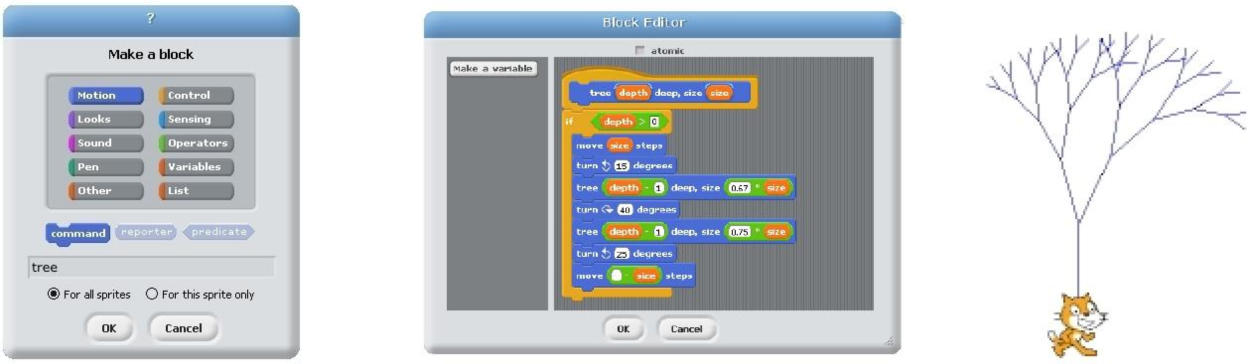


Figure 3. Initial block creation dialog, editing the block, and the final result

Using BYOB, the Berkeley course team was able to develop a pilot curriculum including recursion and teach it to an audience of non-computer-science major 18- and 19-year-olds.

First Class Data

Here is a motivating example for the software extensions discussed below: To introduce recursion, we wanted to use an old Logo demonstration developed by Paul Goldenberg. He begins by defining a few simple shapes:

```
to square
repeat 4 [forward 10 right 90]
end

to hex
repeat 6 [forward 7 right 60]
end

to star
repeat 5 [forward 12 right 144]
end
```

Using these, he presents students with a non-recursive procedure that draws a V shape with a randomly chosen decoration at each end:

```
to vee
  left 45 forward 100
  run pick [square hex star]
  back 100 right 90 forward 100
  run pick [square hex star]
  back 100 left 45
end
```

Since Logo instructions are just text, it's straightforward to take a procedure name and RUN it. Paul runs VEE several times until students are accustomed to the pictures it draws. Then he edits the definition of VEE:

```
to vee
  left 45 forward 100
  run pick [square hex star vee]
  back 100 right 90 forward 100
  run pick [square hex star vee]
  back 100 left 45
end
```

He asks students to predict what will happen when the PICK procedure chooses VEE; most students draw a two-level structure. Then he runs it:

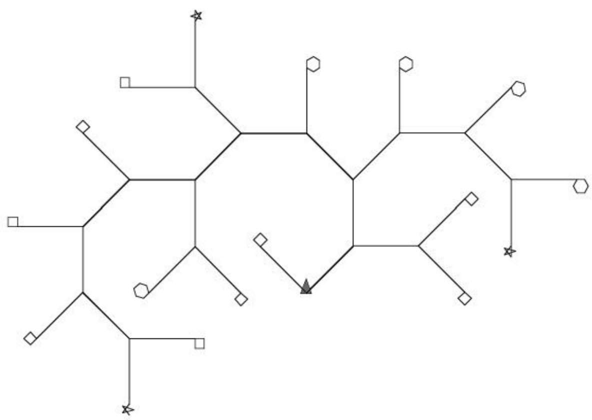


Figure 4. Result of running the recursive VEE procedure

Students' surprise at the complexity of the result leads into an understanding of the possibility of arbitrarily deep recursion. The use of randomness in selecting the decorations at the endpoints eliminates the need for an explicit base case in the procedure, making it easier to read and focusing students' attention on the recursive case rather than the base case.

The graphical nature of Scratch programming makes the VEE example both harder and easier in BYOB. Since a Logo program is just text, the same operations that manipulate sentences (lists of words) also manipulate programs (lists of instructions). The input to RUN is just text. In Scratch, a program is a combination of user interface elements that aren't directly representable as data. This difference required some awkwardness in the programming; the VEE procedure must broadcast a message, one of SQUARE, HEX, etc., and there must be scripts saying that when the sprite receives the SQUARE message it should run the SQUARE procedure, and similarly for the others. On the other hand, the graphical nature of BYOB program blocks *should* make it possible to show the repertoire of VEE decorations in a way that would make the program structure immediately obvious:

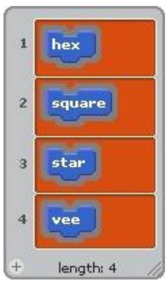


Figure 5. A visually apparent list of Scratch blocks (procedures)

This at-the-time-imaginary BYOB list of blocks is pedagogically preferable to Logo's list of *names* of procedures, because it eliminates the potentially confusing translation from the language's syntax (what you say in your program to refer to a value) to its semantics (the value itself)—in this case, the notation for a procedure versus the procedure itself. In BYOB, a program block shape is still a notation, but one that can't be confused with text; it has no meaning other than the procedure it represents. The pursuit of VEE's list of blocks led to the collaboration between the authors described in this paper.

The desire to put procedures into a list is an example of the general principle, due to Christopher Strachey, of *first class data.* A data type is considered first class in a programming language if instances of that data type can be

- the value of a variable
- a member of a data aggregate (array or list)
- an argument to a procedure
- the value returned by a procedure

As in most languages, numbers and text strings are first class in Scratch; procedures (blocks) are not. A procedure can't play any of the four roles listed above.

First class lists

Consider the following menu for an ice cream shop:

- Sizes: small, medium, large
- Flavors: chocolate pudding, pumpkin, root beer ripple, ginger
- Media: cone, cup

The natural representation of this menu in a computer program is as a *list of lists.* In Logo it would look like this:

[[small medium large] [[chocolate pudding] pumpkin [root beer ripple] ginger] [cone cup]]

A short Berkeley Logo program takes such a menu list and generates all possible combinations:

```
to choices :menu [:sofar []]
  if emptyp :menu [print :sofar stop]
  foreach first :menu [(choices butfirst :menu sentence :sofar ?)]
end
```

This program couldn't be written in Scratch 1.4, but it can be in the new BYOB:

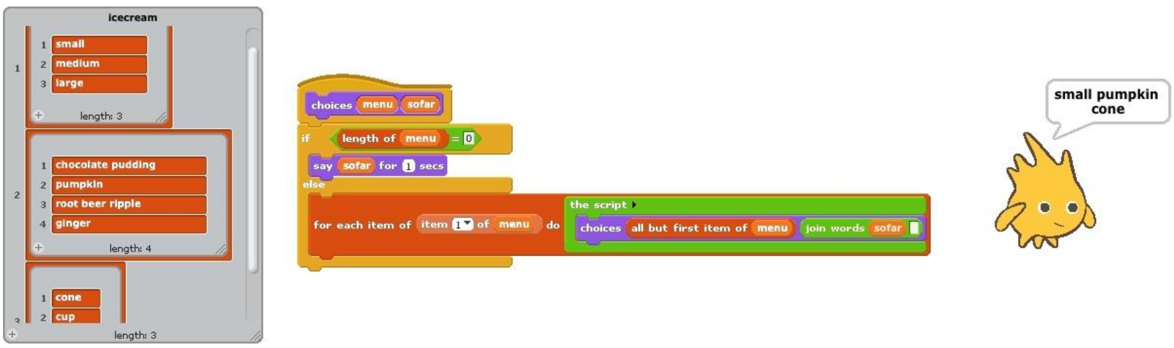


Figure 6. The menu as a list of lists, the CHOICES program, and the result of running it

Given lists of lists, any other desired data structures can be implemented: hash tables, trees, heaps, and so on. Thus, there is no need to clutter the basic Scratch menus with such derived types; they can be defined in libraries that can be loaded only when needed.

First Class Procedures

The pilot version of our new course included only half of the planned eventual topics. One that we left out not only for lack of time but because Scratch/BYOB wouldn't support it was higher order procedures: procedures that take other procedures as arguments. These have proven to be a powerful capability of Logo, and one that users can implement themselves. They are useful, for example, as an alternative to recursion or to looping with index variables when the programmer wants to process all of the elements of a list in a uniform way. For example, the MAP function takes two inputs: *a function* and a list. It returns a list computed by applying the given function to each element of the given list. In Logo, a function can be represented as text, either as the name of a defined function or as a list containing a Logo expression. But in the original BYOB, there was no way to encapsulate a procedure as data that can be input to another procedure—blocks are not first class.

To fix this, we have added to BYOB an equivalent to the Lisp LAMBDA, a way to turn procedures into data. In the Scratch context we need two versions, one for an individual block that's most useful for reporters (functions) and another for scripts, most useful for commands (action scripts):

Figure 7. Procedure encapsulation blocks, value of a block vs. its encapsulation

Once we have blocks and scripts as first class data, we want to be able to run them, the equivalent of Lisp's APPLY procedure. This, too, takes two forms in Scratch, one for commands and one for reporters:

Figure 8. The RUN and CALL blocks, without and with arguments

Figure 8 shows that the arrow at the right end of the RUN and CALL blocks can be clicked to expose as many slots as desired for providing arguments to the procedure being called. In almost all cases, BYOB can figure out where in the procedure the argument(s) should be used, but the arrow at the end of the THE BLOCK and THE SCRIPT blocks can be clicked to expose slots in which explicit formal parameters can be added and then dragged into the encapsulated block or script if necessary.

Encapsulating a block and then de-encapsulating it again by calling it may seem futile, but this combination allows us to write arbitrary higher order functions:

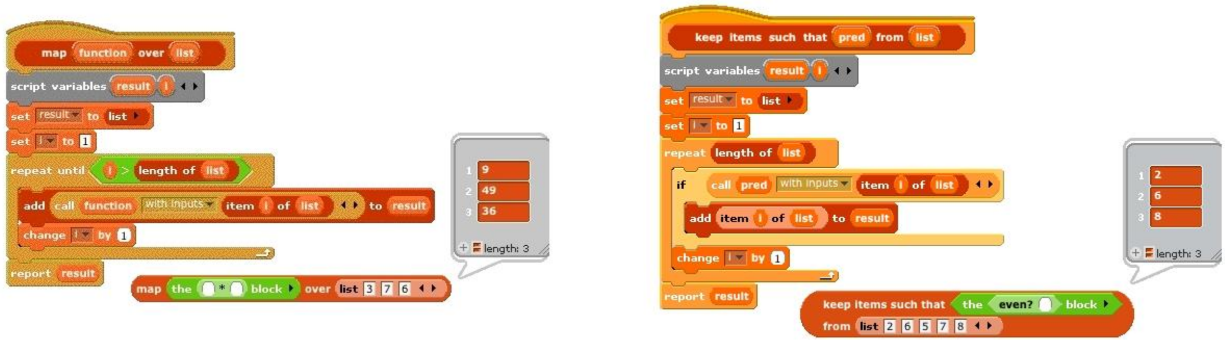


Figure 9. The MAP and KEEP higher order functions, with examples of use

Argument type declarations

Scratch blocks use different shapes and colors to distinguish among three data types in the argument slots:

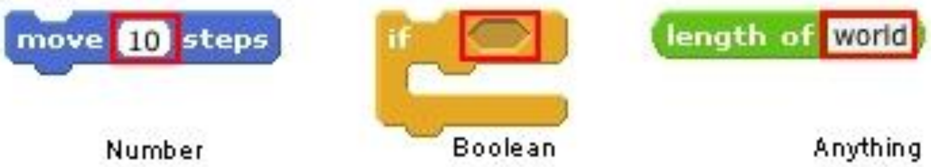


Figure 10. Scratch type shapes.

Also, the blocks themselves come in three shapes: the jigsaw-puzzle-piece commands such as MOVE, the oval reporters such as LENGTH OF, and the hexagonal predicates that fit into the hexagonal Boolean slots.

We have not added to the three block shapes, but because we've added lists and procedures as first class data types, we have created new input shapes for them:

Figure 11. Additional BYOB input shapes.

Since we represent reified blocks and scripts visually with grey borders, these shapes are similar to the shapes of the data that match them.

Our goal is that these input type declarations help the user, not bind the user. Also, this entire feature is optional; by default, the block editor makes all input slots be of type Anything, and the programmer must access a special menu to choose a more restrictive type:

Figure 12. Short and long input name menus

For input slots declared to be a procedure type (command, reporter, or predicate), we allow a special drag-and-drop technique that wraps an implicit THE BLOCK or THE SCRIPT around the input block or script, shown as a grey border:

Figure 13. Normal and implicit-script slot filling

Object Oriented Programming

It may come as no surprise that adding function encapsulation to a language allows it to support functional programming style. But this augmented BYOB also supports object oriented programming, in two ways. First, Scratch has a natural set of objects: the sprites that it uses to control animation. BYOB gives sprites object-like behaviours beyond those designed into Scratch. Second, first class procedures allow the explicit programming of classes and instances. Students who build objects themselves may arguably understand the nature of object oriented programming better than those who encounter an OOP language as a black box.

Central to the OOP paradigm is the idea of message passing. Scratch was designed with a very simplified version: The Scratch programmer can broadcast a message to all sprites, but can't direct the message to a specific sprite. Also, Scratch messages take no arguments, and the scripts that respond to a message cannot return a value to the sender. But both of these limitations are transcended once a script can be the value of a variable.

The Scratch "Sensing" menu includes a reporter block called "*variable* OF *sprite*." The original intent of this feature is, for example, to allow one sprite to find out the X and Y coordinates of another sprite's position. But when this block is used in the input slot of RUN or CALL, it allows one sprite to invoke a method in another sprite! (We have added a variant of RUN called LAUNCH that starts a new thread to run the method asynchronously.) The fact that this very useful OOP capability was automatically implied by a mechanism we created for a different purpose shows how powerful the idea of procedure encapsulation is. To prepare to accept a message, a sprite must merely create a local variable by that name, whose value is a method script for that message.

Here is the simplest illustration of the second approach to OOP in BYOB. We are going to use procedures to represent both the class and the instances for a COUNTER class:

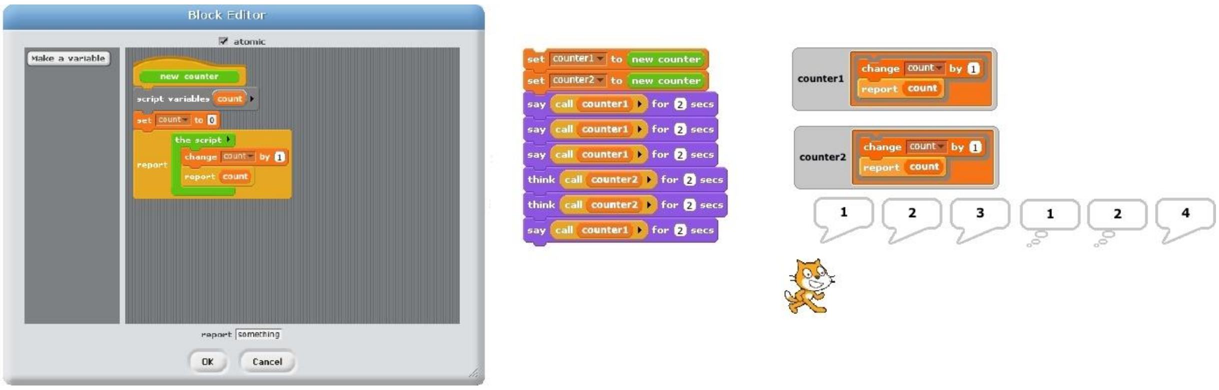


Figure 14. The COUNTER class, two instances, and the result of several calls to the instances

Procedure NEW COUNTER represents the class. Every call to any BYOB block creates new block variables. For the most part, these block variables are temporary; when the block's script completes and the block returns to its caller, nothing points to the block variables and their space is reclaimed. But this procedure returns a procedure! Since the latter refers to the COUNT variable, that variable is *not* temporary, but acts as a persistent local state variable for the counter instance. The procedure created by the THE SCRIPT block represents an instance. In Figure 14, we create two counter instances, each with its own COUNT variable. Each of them starts at 0 and is increased each time the *instance* (COUNTER1 or COUNTER2) is called. The figure shows that COUNTER1 remembers its count even after COUNTER2 is called.

Message Passing

The simple counter in Figure 15 shows how the ability of a procedure to return another procedure gives us the persistent local state variables that OOP requires. But to follow the object metaphor faithfully requires message passing—the ability to ask an object to do one of a repertoire of actions. It's not hard to extend the basic idea in Figure 14 to allow messages by representing an object by a *dispatch procedure* that takes a message (just a word) as its argument and returns a method (a procedure).

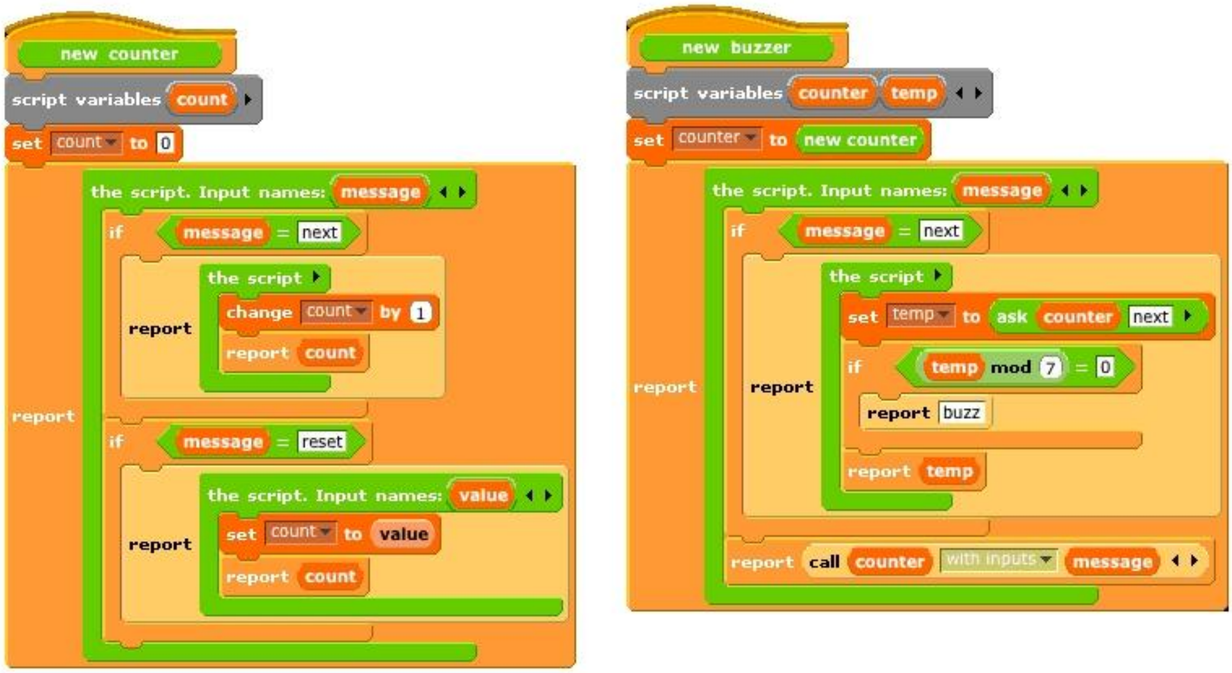


Figure 15. Message passing counter, and inheritance by delegation

The NEW COUNTER block in Figure 15 represents a class with two messages, NEXT and RESET. Each instance of the class is a dispatch procedure, created by the large outer THE SCRIPT block. Each of the two inner THE SCRIPT blocks creates a method. The NEXT method increments and reports the count. The RESET method takes an argument, and sets the count to that value. Sending an object a message is a two-step process: First call the dispatch procedure with the message to get the method, then call the method.

Inheritance

Message passing and local state variables provide the essence of the OOP paradigm, but to make it practical we need inheritance, the ability to reuse existing methods. The NEW BUZZER example in Figure 15 illustrates the easiest way to implement inheritance, namely delegation. A buzzer object is just like a counter object, except that if the new count is divisible by 7, it returns the word BUZZ instead of the number. Only the NEXT method must be changed; in all other ways a buzzer should behave like a counter. (In this simplified example there is only one other method, RESET, but you should imagine a class with many methods.) Every instance of a buzzer contains within it an instance of the counter class. (This is accomplished by the SET block near the top of the script.) The buzzer's dispatch procedure has an explicit script for the one message it handles differently (NEXT); if any other message is received, the buzzer merely forwards the message to its internal counter.

Conclusion

We believe that we can support a wide range of introductory computer science courses in Scratch with very few additions to its repertoire of primitive blocks:

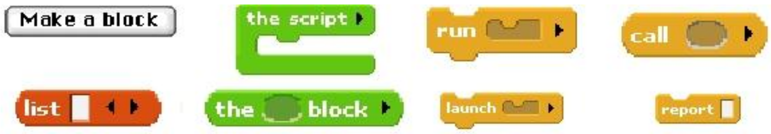


Figure 16. All we added

We had to consider and solve many technical problems, such as scope of variables, but our claim is that none of these will be visible to the typical Scratch programmer. The key is to begin with the ideal of first class data in mind, and to recognize in particular that procedures as data enable us to write *in Scratch itself* the many computer science examples that would otherwise have to be added as primitive blocks.

We are not suggesting that the children who are Scratch's main audience will start writing recursive procedures (at least, not the youngest of them), higher order functions, or object classes. What we argue is that the additional features we need to support computer science students will be inobtrusive. (Computer science students will, no doubt, use libraries of blocks written by their teachers, and some of those might be cluttered. But that won't affect the core of the language.) And there are many benefits to keeping the most expert Scratch programmers as part of the same community as the beginners; a separate Scratch for older users would lose those benefits. (As we are writing this paper in January, the software is just getting to the point at which we'll feel ready to try it out on kids. We hope to have some experiences to report by the summer.)

We gratefully acknowledge the ideas we borrowed from the Scheme language (Steele and Sussman, 1975) and from *Structure and Interpretation of Computer Programs* (Abelson and Sussman, 1996), as well as the brilliant inventions of the Scratch team at MIT.

References

Abelson, H., and Sussman, G.J., with Sussman, J., *Structure and Interpretation of Computer Programs,* 2nd edition, MIT Press, 1996.

Malan, D., and Leitner, H., *Scratch for Budding Computer Scientists,* SIGCSE Procedings, 2007.

Pausch, R., Burnette, T., Capeheart, A.C., Conway, M., Cosgrove, D., DeLine. R., Durbin. J., Gossweiler, R., Koga, S., and White, J., *Alice: Rapid Prototyping System for Virtual Reality*, IEEE Computer Graphics and Applications, May 1995.

Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., and Kafai, Y. (2009). *Scratch: Programming for All.* Communications of the ACM, vol. 52, no. 11, pp. 60-67 (Nov. 2009).

Shafer, R. *Oh! The Beauty and Joy of Computing,* Innovations Volume 3, Issue 10, UC Berkeley College of Engineering, December 2009. See also *http://inst.eecs.berkeley.edu/~cs10*

Smith, D.C., and Cypher, A., *Making Programming Easier for Children,* in Druin, A., ed., *The Design of Children's Technology,* Morgan Kaufmann, San Francisco, 1998.

Steele, G.L., and Sussman. G.J. *Scheme: An interpreter for the extended lambda calculus*. Memo 349, MIT Artificial Intelligence Laboratory, 1975.

Wolz, U., Maloney, J., and Pulimood, S.J., *"Scratch" Your Way to Introductory CS,* ACM SIGCSE Bulletin, Volume 40 , Issue 1, March 2008.

BYOB – Bringing "No Ceiling" to Scratch

Brian Harvey, *bh@cs.berkeley.edu*
Computer Science Division, University of California, Berkeley

Jens Mőnig, *jens.moenig@miosoft.com*
Enterprise Applications Development, MioSoft Corporation

Introductory description and overall goals

Scratch (scratch.mit.edu) is a graphical drag-and-drop programming language for kids from MIT. The same non-intimidating interface and ease of use that make it work for eight-year-olds also make it appealing as a language for courses to introduce computer science ideas to non-CS majors at the university level. But Scratch has weaknesses as a programming language, most notably the inability to define procedures, and therefore to explore recursion. As a result, several schools use Scratch for the first week or two and then switch to a "serious" programming language for the rest of the course.

Instead of that, we propose adding just a few capabilities to Scratch, so that it remains unintimidating for kids, but can also serve older learners. BYOB (Build Your Own Blocks) is an extended version of Scratch adding procedure definition, first class procedures, and first class lists. With these we can invent *in BYOB* additional tools for whatever data structures and control structures are needed. This workshop will introduce participants to BYOB (byob.berkeley.edu).

Method

After a brief introduction to Scratch, participants will explore writing procedures, recursion, creating control structures, using and writing higher order functions, and building data structures. There will be specific exercises to introduce each of these topics, but most of the time will be spent in free exploration of BYOB.

Expected outcomes

Participants will understand how some important computer science ideas can be expressed in the medium of a graphical programming language, especially recursion and higher order functions. They will be able to write BYOB programs.

Keywords (style: Keywords)

programming language; Scratch; BYOB; recursion; higher order functions

The Modelling4All Project

Ken Kahn (*toontalk@gmail.com*), **Howard Noble** *(howard.noble@oucs.ox.ac.uk)*, **Arthur Hjorth** *(arthurhjorth@googlemail.com) and* **Fábio Sampaio** *(ffs@nce.ufrj.br)*
Oxford University and Federal University of Rio de Janeiro

Description

The Modelling4All Project (http://modelling4all.org) has built a Web 2.0 tool that meets one of the biggest challenges facing computer modelling: Widening participation to include non-programmers.

At the heart of the Modelling4All is a browser-based tool that allows users to design agent-based models through a point-and-click interface: the BehaviourComposer. Just as a composer creates music by combining strings of nodes, users easily create agents and bring them to life by combining so-called 'micro behaviours' from a large pre-written but flexible library (e.g. 'move-forward-and-turn-randomly', or 'eat-nearest-enemy'). Even non-programmers are able to create complex models in hours or even minutes: The Modelling4All tool has successfully been used to teach subjects as diverse as epidemiology, flocking behaviour, and business on both bachelor's and master's level at Oxford University.

The Modelling4All tool allows easy sharing of models, combining a constructionist and a social constructivist approaches to learning and teaching: the Modelling4All models are stored on the web server and are compiled to NetLogo to be run as Java applets that can be easily embedded in any web page.

The Modelling4All tool is built on top of NetLogo in such a way that users can quickly build and run models without first learning NetLogo, but the NetLogo code is accessible and editable for more expert users.

Method

At Constructionism 2010 we will be showcasing the Epidemic Game Maker which enables users to build models and games of epidemics in minutes. It is built as an extension to the BehaviourComposer and was developed for the Royal Society's Summer Science Exhibition 2010 (http://royalsociety.org/Summer-Science-Exhibition-2010/). We will also set aside time for discussions of different approaches to teaching modelling and of what tools are needed to support it.

Expected outcomes

Attendees will develop an understanding of agent-based modelling using the BehaviourComposer, and the way the software has been used in teaching at Oxford and as part of an exhibit about epidemics at the Royal Society Summer Science event this year.

Keywords

Agent-based modelling, NetLogo, BehaviourComposer, Modelling4All, literacy for computer modelling.

Modelling4All and the Epidemic Game Maker

Dr. Ken Kahn, Howard Noble, Arthur Hjorth, Dr. Fabio Ferrentini Sampaio:
toontalk@gmail.com, howard.noble@oucs.ox.ac.uk, hermes.hjorth@medsci.ox.ac.uk, ffs@nce.ufrj.br
Oxford University Computing Services, Oxford University

Didactic philosophy

Modelling4All attempts to address one of the biggest challenges to the constructionist approach to learning: including non-programmers. We believe that while giving learners the skills necessary to build their own models may be the end goal, the learning curve is sometimes so steep that we lose learners on the way.

To address this issue, Modelling4All enables learners to "compose" models from blocks of code, called micro-behaviours. These micro-behaviours can be anything from the simplest behaviour (move-forward-every-t) that can be added an agent, to pre-defined agents that can be easily "plugged" into a model. Learners can even, by the click of a mouse, deploy hundreds of agents in complicated social networks. By enabling learners to build models 'middle-out' (as opposed to 'bottom-up') we believe that we get the best of Constructionism while avoiding the pitfalls.

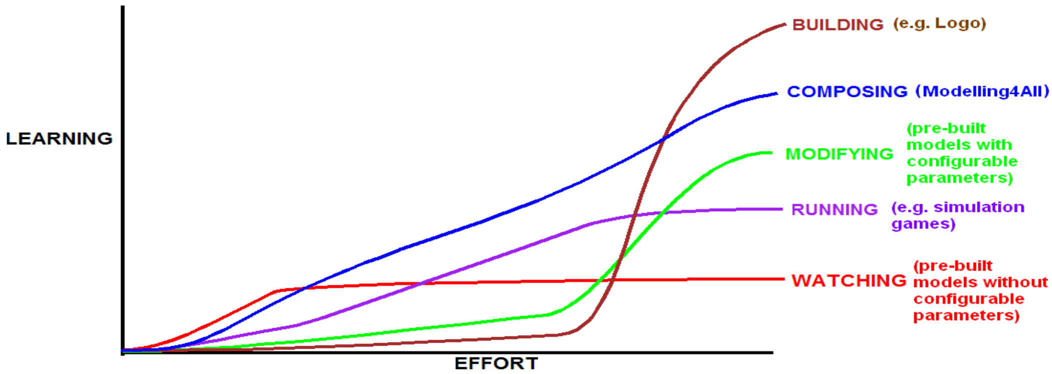


Figure 1. Learning vs. Effort for different interactions with models.

Epidemic Game Maker

Epidemic Game Maker was built for the Royal Society's Summer Exhibition 2010. It aims to teach children (10-15) about epidemics from the perspective of a public health official in 2 minutes (!) with an optional extension of up to 5-7 minutes.

The learner starts with a simple model showing children going back and forth to school. Just one child is infected with the flu. On running the model, the learner can see children moving around in the model-space, contaminating each other. By adding agents (e.g., more schools, adults, work places) and functionality (e.g., various policy interventions such as school closings) and by exploring the parameter space, children are able to build, run, and configure models the models in minutes. In the process students are earning about modelling, epidemics, and public health policies.

Keywords (style: Keywords)

constructionism, learning, interface, non-programmer,

LEGO® SERIOUS PLAY™ in Education

Anna-Sophie Trolle Terkelsen, Hasse Gedde Hoejland,
Anna-Sophie.Trolle.Terkelsen@lego.com; Hasse.Gedde.Hoejland@lego.com
LEGO® Education, LEGO® System A/S, Havremarken, 7190 Billund, Denmark

Introductory description and overall goals

Why are LEGO® bricks so pertinent to a 21st century classroom and what is LEGO® SERIOUS PLAY™?

The LEGO® System's robotics and engineering solutions lend themselves to STEM subjects and allow students to work in an authentic way with inquiry, design, measurement and control.

Within the LEGO® SERIOUS PLAY™ framework the brick is also a powerful tool which students can use to express their reflections and experiences as responses to e.g. poetry and literature, as well as to concepts within the humanities - such as citizenship and democracy.

Method

LEGO SERIOUS PLAY for education is a new concept developed for schools. The concept is designed to enable students to work together to build expressions of their understanding of the world around them. It is a method that teachers have credited for being remarkably inclusive and that helps to develop students' self-expression, self-esteem, and confidence as well as activate and enhance their creative thinking and collaboration skills.

This workshop is a chance to try out LEGO SERIOUS PLAY first hand.

Expected outcomes

The workshop is first and foremost an opportunity for participants to try a LEGO SERIOUS PLAY process hands on. The hands on experience of the workshop will give participants an understanding of the uses of LEGO SERIOUS PLAY within an educational context. The workshop will set an example of how the constructionist approach to learning processes with great advantages can be applied in the humanistic subjects, and participants are likely to find inspiration about how the teacher should handle his/her role as a facilitator in such learning processes.

Keywords

LEGO® SERIOUS PLAY™ for Education; Learning; Creativity; Humanistic subjects.

Bringing Constructionism to Action Game-Play

Nathan Holbert, *nholbert@u.northwestern.edu*
Learning Sciences, Northwestern University

Lauren Penney, *lauren.penney@u.northwestern.edu*
Learning Sciences, Northwestern University

Uri Wilensky, *uri@northwestern.edu*
Learning Sciences, Northwestern University

Abstract

As technology has become cheaper and ubiquitous, children are spending more time playing video games. Surveys suggest that that video game play is an activity that children participate in almost universally and that the amount of time spent playing games is enormous (Lenhart et al., 2008). While new research makes a compelling case for the educational potential of video games, some categories of games are rarely represented. Action platform games in particular, while incredibly popular among today's youth, are seldom mentioned in video game research.

Constructionism is a powerful design tool for transforming passive activities into highly engaging, thought-provoking, educationally rich experiences (Papert, 1993a). While constructionism has been utilized successfully in programs that encourage children to design video games (Harel & Papert, 1991; Kafai, 1995), we believe that constructionism has a place in the playing of video games as well. We propose that action platform games should be designed to incorporate a constructionist paradigm. By incorporating constructionism into action platform video games we believe that such games can become powerful spaces for identity formation and problem-solving skill development.

We believe a constructionist redesign of action platform games will include an opportunity for player-character construction, an open and flexible system for building objects in-game to overcome obstacles, and a medium for sharing game-play with other players. By allowing for the personally meaningful construction of unique in-game characters, players will be allowed to incorporate their own identities into that of their digital avatar (Gee, 2003; Harel & Papert, 1991; Kafai, 1996a; Papert & Harel, 1991). Designing levels so that components, rather than complete objects, are utilized in overcoming obstacles allows the player to systematically build various solutions to problems and to develop new relationships with their constructions as well as with the problem they're building to solve (Cavallo et al., 2004; Wilensky, 1991). Finally, by providing an integrated system that allows player to share their game-play, action platform games can become a space that nurtures a community of learners as players deconstruct one another's methods and construct new ideas and solutions (Kafai, 2006; Papert, 1993a; Papert & Harel, 1991).

We hope that these suggestions will serve as a starting point for a broader dialogue on a wider adoption of constructionism in video games.

Keywords

video games, informal, play, problem solving, identity

Introduction

Constructionist designs have been successfully applied to a wide range of domains including math education (Eisenberg, 2000; Feurzeig, 1989; Noss & Hoyles, 1996; Papert & Harel, 1991; Roschelle, Kaput, & Stroup, 2000; Wilensky, 1996), science education (diSessa, 1997; Sengupta & Wilensky, 2008; Wilensky & Reisman, 2006), computational literacy (Berland, 2006; Hancock, 2001; Harel & Papert, 1991; Kafai, 1996b; Wilensky, 1999), and engineering education (Blikstein & Wilensky, 2004; Martin, 1996; Resnick & Ocko, 1990). However, the design of video games, a domain currently being explored with much enthusiasm by educational researchers, rarely considers constructionism. Video games have always been of interest to constructionist, and many have designed constructionist environments that appoint children the role of game designers. Such research has shown game design to be a powerful way for youth of both genders and varying learning types to make personal connections to content and problem solving (Harel & Papert, 1991; Kafai, 1995, 1996a). While we are very excited by this work we believe that the *playing* of video games could also benefit from a constructionist design. In this paper we propose a set of level design strategies for transforming action platform games into constructionist environments where the player constructs her own solutions and paths through game levels. Such a design would encourage the player to construct sharable characters and artifacts as a means of overcoming obstacles and solving puzzles and provide a medium for sharing these designs. It is our hope that these design ideas will begin a conversation about how to infuse constructionism into game-play where it has traditionally been absent.

Motivation

Video games constitute an important part of the lives of children and youth in today's world. The PEW Internet and American Life Project claims that as many as 97% of all American teens (regardless of gender, age, or socioeconomic status) play video games in some way and 50% play games daily for an hour or more (Lenhart et al., 2008). Such numbers are commonly explained simply by assuming that video games are fun – of course kids like to play them! However, this off-handed dismissal neglects the reality that video games are generally difficult and require a very large time investment to master (Gee, 2003). Papert (1993b) suggests, "some forms of learning are fast-paced, immensely compelling, and rewarding. The fact that they are enormously demanding of one's time and require new ways of thinking remains a small price to pay" (p. 5). Like constructionism, video games are motivating and interesting, despite their difficulty, because they "empower children to test out ideas about working within prefixed rules and structures" (Papert, 1993b, p. 4).

A large body of work has shown that video games contribute to epistemic literacy (Gee, 2003), mimic proven and effective learning environments (Stevens, Satwicz, & McCarthy, 2008), positively impact learning motivation (Orvis, Horn, & Belanich, 2008), alter quantitative reasoning (Satwicz & Stevens, 2008), and can be effective at leveraging expertise in formal learning environments (Shaffer, 2006). Despite the positive nature of this literature, it is clear that different games have different strengths and that some games may be seriously deficient in educational value.

While children play a wide variety of video games, the action platform game is one genre with which nearly every child has experience. The basic structure of these games varies drastically, however, the defining characteristic of an action platform game is the need to overcome obstacles with quick reflexes. This basic game-play structure makes up a huge proportion of some of the most popular console games (Mario, Donkey Kong, and Sonic the Hedgehog are some classic examples) and leads to the quick and exciting play often preferred by younger audiences. Unfortunately, the literature rarely cites such games as having educational value. We believe that bringing a constructionist design to action platform video games that allows players to construct characters, construct tools and artifacts within the game, and provides a medium to share designs and strategies will transform passive reflex-driven interactions into an opportunity for thoughtful, reflective, and interesting game-play.

Character Construction

A key characteristic of action platform games is their linear nature. While these games have always allowed for some amount of exploration, players are always pushed towards a specific locational goal. For example, in the popular and extremely successful action platform game LittleBigPlanet (a game that we feel comes the closest to integrating constructionist design aesthetics) players can find additional "prize bubbles" by exploring out-of-the-way areas in a level. However, despite the slight bonus for curiosity, players are inevitably forced to get back onto the main pathway to finish a level. In addition, many action platform games have dramatically limited the way in which players can reach this goal. Even when offering various tools and powers as was done in the Super Mario Bros. series of games (flowers that allow the player to shoot fireballs, or a feather that gave the player the ability to fly), levels generally have a "best solution." This tendency to push players to a particular path is at least partially due to the inflexibility of the player character – when characters are designed only one way, there becomes only one logical path for level completion.

Constructionism places a very high premium on making learning experiences personal. In *Mindstorms,* Papert (1993a) suggests that learning (in this case physics) is about bringing content "into contact with very diverse personal knowledge" (p. 122). Constructionist environments allow learners to build artifacts that reflect their interests and goals – to take ownership of learning and to develop an "intellectual identity" (Papert, 1993b, p. 24). Whether it's turtles, gears, or LEGOs that are especially salient to the learner, constructionist designs generally allow the learner to "own" and customize this artifact. In addition, cultural and gender differences are not only supported by the environment, but also leveraged in artifact construction (Harel & Papert, 1991; Kafai, 1996a; Papert & Harel, 1991). The uniformity of player characters and consolidated game-play found in action platform games severely limits the possibility for personally meaningful construction.

One key difficulty in designing constructionist levels for action platform games is the limited abilities of the player character. In a game like LittleBigPlanet, where the player character can only push, pull, and jump, obstacle designs are severely constrained. We believe a constructionist design should allow the player to customize character traits and abilities. There are a variety of possible ways one might achieve this. One method, which we refer to as the "backpack design," is based on the classic game *Lemmings*. *Lemmings* allows the player to activate various character abilities in order to solve complex puzzles. Every lemming in the world is the same, however every level has a selection of backpack abilities (dig down, build stairs, climb up) the player can assign to any lemming. Most of these abilities are temporary; when the lemming runs out of materials or is unable to continue (digs through the wall or climbs up to the top), he turns back into a basic walking lemming. The player solves the level by assigning lemmings abilities that will remove obstacles and bypass hazards, allowing the remaining lemmings to travel safely to their home. We believe a "backpack design" which allows the player to obtain temporary special ability is a powerful idea to consider when building constructionist action platform games. The available abilities could change dynamically throughout the level or the player could preselect abilities before beginning levels. Classic action platform games such as *Super Mario Bros. 2* and more modern games such as *Trine,* have explored the notion of preselected or dynamic unique abilities to great effect. In these games predefined characters that each have different strengths or abilities allow players to move through the level and solve puzzles in *character-specific* ways. We believe that tweaking such a design to allow for flexible ability assignment would allow for puzzle and level completion that is *player-specific*. In addition, such an approach could lead to especially interesting results in multiplayer situations. In a multiplayer level players would select complimentary backpack abilities that they could then coordinate as they moved through the level.

Key to this design, and what distinguishes it from traditional action platform games, is the player directed nature of ability choice and the flexibility of level design to accommodate these choices. In a constructionist action platform game the player should be able to construct the character identity in a way that is personal and meaningful for them and obstacles and puzzles in the level should be designed in such a way that different abilities make for novel and interesting solutions.

Object Construction

Because action platform games are designed for a quick pace, elements encountered during game-play are already pre-fabricated. While a player might need to find a tool or utilize a vehicle in order to overcome some obstacle, there is usually only one correct way to solve these challenges. In LittleBigPlanet, complex vehicles that players could easily build in the game's highly constructive "create" mode," are simply there, waiting for the player to press the button to turn them on in the completely separate "play" mode. The tendency for action games to provide the player with complete objects to use allows for the possibility of speed, but removes a golden opportunity for construction.

Constructionism claims that by building and sharing personally meaningful artifacts learners become not only more aware of their own methods and style of problem solving, but also more aware of the nuances of the problem (Papert, 1993a; Kafai, 1996). By providing learners with the "pieces" with which to build solutions to problems, the learner is able to focus on different aspects and features of the design as they become necessary. As described by Cavallo, Papert, and Stager (2004), such an approach allows for "out of the box" thinking that can lead to creative and surprising designs. Constructionism's focus on components, rather than finished objects, allows the player to imagine a variety of possible endpoints. This increase in connections with the components and representations of the constructed objects increases the quality of the relationship the player will have with their final construction (Wilensky, 1991).

One way to make action games more constructionist would be to provide pieces of useful or necessary objects that could be put together in multiple ways, allowing each player the opportunity to build one of a number of different solutions – each of which could be used to solve the same challenge. In other words, rather than provide the player with the object used to overcome obstacles, the player would be provided with the materials to make such an object. One example of this would be to have the player build a vehicle rather than just providing it for her. A large variety of components would be provided and the player would have the opportunity to construct their vehicle in a way that is meaningful to her. In one case the player may create a small vehicle made of light material so that it can easily jump an obstacle of boxes, while another player may construct a metal vehicle with large wheels that can simply crash through the boxes. In this way players are not only encouraged to think about the design of their solution, but also to consider the many nuances of the obstacle for which they're designing a solution. In addition, the flexibility of design would likely encourage in-room interactions as other players or even non-playing friends and family members offered their suggestions and advice on the "best" object design. These in-room interactions have been shown to be an especially powerful aspect of the learning environment created by video games (Stevens et al., 2008).

Sharing

The public sharing of artifacts is a concept that is vital to constructionism and completely absent from the design of nearly every action platform game. Games and software that encourage players to create content – and these games are increasing in popularity – are often very successful at incorporating a public sharing component, but as mentioned previously, the "playing" aspect of video games lack this important feature. For example, when in LittleBigPlanet's "create" mode players have the opportunity to share object and item constructions as well as entire levels. However, once the player switches to "play" mode, nothing about the action is shareable. The absence of a public space or method to share the constructions that we propose should happen while playing action platform video games is a problem that must be solved before such games can be considered constructionist.

In constructionism, the words *public* and *sharable* are always present (Papert & Harel, 1991). Stemming from Lave and Wanger's (1991) idea of legitimate peripheral participation, and Papert's interest in Brazilian samba schools (1993a), public constructions give learners an entry point at all levels. Working side by side both novices and masters are able to participate at all levels of construction. Whether building a computer program or a tangible object, learners should have the ability to see other's ideas, borrow from them, deconstruct them, and to present one's own ideas. A constructionist design creates a community of learners much wider than the traditional model of only teacher and student (Kafai, 2006).

How does one share when playing an action game? One medium that players have adopted ad hoc to make their game-play public is online videos. A quick search on YouTube reveals thousands of videos recorded and cut by players to show game-play. These videos are often recorded to show off successes or to illustrate how one overcomes a particularly difficult obstacle. We believe this is one way action platform game-play could be shared publicly. Action video games could include a feature that would allow the player to record their actions at any given point in their play. This recording would then be tagged as relating to the player that created the recording and relating to a specific area of a level. Another feature would be available that would allow players to activate these videos when struggling with a construction or obstacle. Either a random video recorded of the obstacle could be played, or a player could choose specific videos made by a particular player. The previously recorded action would then be overlaid on the player's screen allowing her to see how other players have solved the challenge. Some of the actions depicted on the video may be particularly useful, while other actions might be irrelevant (perhaps the player who created the video has selected different starting abilities than the watching player making some actions impossible). This feature allows the player to deconstruct the actions of other players to find the useful bits, and it allows players to present in a public forum their own play.

Conclusions

In this paper we've tried to argue for the inclusion of constructionist designs in action platform video games. While there have been some interesting instances of constructionism in video game creation – which has begun to be included in some popular commercial games – the "play" aspect of games has been left without. We have argued that the opportunity to construct the player-character would allow for flexibility in game-play and variety in problem and obstacle solutions as well as a space for identity projection and experimentation. In addition, we believe that players should build objects, tools, and vehicles within game levels. While action platform games often create interesting opportunities to interact with such objects, we believe that providing players with the components to construct personal versions of these objects would necessitate systematic design thinking, highlight the power of emergent systems, and encourage in-room interactions. Finally, no constructionist design is complete without an opportunity for the public sharing of artifacts. We propose that action platform games should provide an opportunity to share one's game-play with other players and allow individuals to deconstruct and piece together other's strategies. Including these designs in the playing of action platform games will potentially transform a fairly intellectually passive game type into a powerful constructionist environment. We believe this is a starting point for a broader dialogue on a wider adoption of constructionism in video games.

References

Berland, M. (2006). Constructionist collaborative engineering: PVBOT. Paper presented at the annual meeting of the American Educational Research Association, San Francisco, CA.

Blikstein, P., & Wilensky, U. (2004). MaterialSim: An agent-based simulation toolkit for learning materials science. Paper presented at the International Conference on Engineering Education, Gainsville, Florida.

Cavallo, D., Papert, S., & Stager, G. (2004). Climbing to understanding: Lessons from an experimental learning environment for adjudicated youth*. International Conference on Learning Sciences* (pp. 113-120). Santa Monica, California

diSessa, A. A. (1997). Open toolsets: New ends and new means in learning mathematics and science with computers*. 21st Annual Meeting of the Psychology of Mathematics Education*. Finland

Eisenberg, M. (2000). Superposed turtle walks. *International Journal of Computers for Mathematical Learning*, *5*(1), 65-83.

Feurzeig, W. (1989). A visual programming environment for mathematics education. Paper presented at the Fourth International Conference for Logo and Mathematics Education, Jerusalem, Israel.

Gee, J. P. (2003). *What video games have to teach us about learning and literacy*. New York: Palgrave Macmillan.

Hancock, C. (2001). Children's understanding of process in construction of robot behaviors. Paper presented at the Annual Meeting of the American Educational Research Association, Seattle, WA.

Harel, I., & Papert, S. (1991). Software design as a learning enviornment. *Interactive Learning Environments*, *1*(1), 1-30.

Kafai, Y. B. (1995). *Minds in play: Computer game design as a context for children's learning*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Kafai, Y. B. (1996a). Gender differences in children's constructions of video games. In P. M. Greenfield & R. R. Cocking (Eds.), *Interacting with video* (pp. 39-66). Norwood, NJ: Ablex Publishing Corporation.

Kafai, Y. B. (1996b). Learning design by making games: Children's development of design strategies in the creation of a complex computational artifact. In Y. B. Kafai & M. Resnick (Eds.), *Constructionism in practice: Designing, thinking, and learning in a digital world*. Mahwah, NJ: Lawrence Erlbaum.

Kafai, Y. B. (2006). Constructionism. In R. K. Sawyer (Ed.), *The cambridge handbook of the learning sciences*. New York: Cambridge University Press.

Lave, J., & Wenger, E. (1991). *Situated learning: Ligitimate peripheral participation*. London: Cambridge University Press.

Lenhart, A., Kahne, J., Middaugh, E., Macgill, A. R., Evans, C., & Vitak, J. (2008). *Teens, Video Games, and Civics.*

Martin, F. G. (1996). Kids learning engineering science using LEGO and the programmable brick. Paper presented at the Annual Meeting of the American Educational Research Association, New York, NY.

Noss, R., & Hoyles, C. (1996). *Windows on mathematical meanings: Learning cultures and computers*. Dordrecht, The Netherlands: Kluwer Academic Press.

Orvis, K. A., Horn, D. B., & Belanich, J. (2008). The roles of task difficulty and prior videogame experience on performance and motivation in instructional videogames. *Computers in Human Behavior*, *24*, 2415–2433.

Papert, S. (1993a). *Mindstorms: Children, computers and powerful ideas*. New York: Basic Books.

Papert, S. (1993b). *The Children's Machine: Rethinking School in the Age of the Computer*. New York: Basic Books.

Papert, S., & Harel, I. (1991). Situating constructionism. In S. Papert & I. Harel (Eds.), *Constructionism*. New York: Ablex Publishing.

Resnick, M., & Ocko, S. (1990). LEGO/Logo: Learning through and about design. In I. Harel (Ed.), *Constructionist Learning* (pp. 121-128). Cambridge, MA: MIT Media Laboratory.

Roschelle, J., Kaput, J. J., & Stroup, W. (2000). SimCalc: Accelerating students' engagement with the mathematics of change. In M. J. Jacobson & R. B. Kozma (Eds.), *Innovations in science and mathematics education: Advanced designs for technologies of learning* (pp. 47-76). Hillsdale, NJ: Earlbaum.

Satwicz, T., & Stevens, R. (2008). Playing with representations: How do kids make use of quantitative representations in video games? *International Journal of Computers for Mathematical Learning*, *13*, 179-206.

Sengupta, P., & Wilensky, U. (2008). Learning electricity with NIELS: Thinking with electrons and thinking in levels. *International Journal of Computers for Mathematical Learning*, (accepted / in press).

Shaffer, D. W. (2006). Epistemic frames for epistemic games. *Computers and Education*, *46*(3), 223-234.

Stevens, R., Satwicz, T., & McCarthy, L. (2008). In-game, in-room, in-world: Reconnecting video game play to the rest of kids' lives. In K. Salen (Ed.), *The ecology of games: Connecting youth, games, and learning* (pp. 41-66). Cambridge, MA: The MIT Press.

Wilensky, U. (1991). Abstract meditations on the concrete and concrete implications for mathematics education. In *Constructionism*. Norwood, N.J.: Ablex Publishing Corp.

Wilensky, U. (1996). Making sense of probability through paradox and programming: A case study in a connected mathematics framework. In Y. B. Kafai & M. Resnick (Eds.), *Constructionism in practice: Designing, thinking, and learning in a digital world*. Mahwah, NJ: Lawrence Erlbaum.

Wilensky, U. (1999). NetLogo. http://ccl.northwestern.edu/netlogo/. Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL.

Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories--and embodied modeling approach. *Cognition and Instruction*, *24*(2), 171-209.

Creativity – An Emergent Phenomenon in Interactive Art

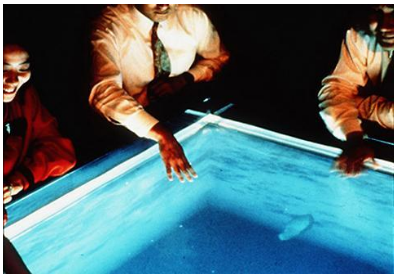
Jana Horáková, *horakova@phil.muni.cz* Department of Musicology, Faculty of Arts, Masaryk University, Brno, Czech Republic

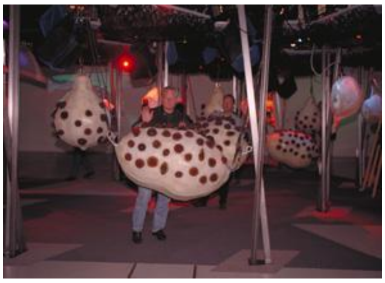
Jozef Kelemen, *kelemen@fpf.slu.cz* Institute of Computer Science, Silesian University, Opava, Czech Republic, and VSM School of Management, Bratislava, Slovakia

Creativity is usually considered as a phenomenon resulting from some mental and social processes: From the individual generation of new concepts or new associations between the existing concepts, so as the ability to find “new ways to look at things” (Minsky, 1986, p. 134), and the social process of accepting the result of the individual mental activity at least by a part of the society in which this activity runs. In the present contribution we continue in our contemplations concerning the creativity presented in (Kelemen, 2009). Using three examples taken from the field of interactive art we illustrate the phenomenon of emergence of creativity.

In the interactive art becomes, according our opinion, most evidently recognizable the fact that for emerging of the artistic affect of the artwork, the active roles both of the author of the “meta-piece” which provides the conditions for active interaction of the audience, as well as the activities of the audience with the “meta-piece” are necessary. So, the appearance of creativity in the case of interactive artworks is in fact an emergent phenomenon which results from interaction between the author’s (individual) work and the interactive activities of the audience (the part of the society).

The appearance of the emergence of creativity is tested applying the well-known (at least in the field of artificial life) emergence test proposed by Roland, Sipper and Capcarrere (1999). As the examples the project A-Volve by Ch. Sommerer and L. Mignonneau (1993, Fig. a), the Brain Opera project by T. Machover (1995, Fig. b), and the project Mnemeg by F. Diaz (2001, Fig. c) are used.





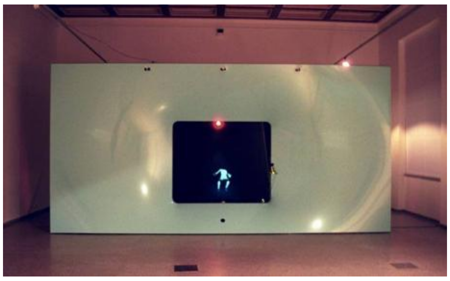


Fig.: a b c

References

KELEMEN, J.: A note on agents, societies, and creativity. In: *Proc. 7th IEEE International Conference on Computational Cybernetics, ICCC* (A. Szakal, Ed.). IEEE CD Publication (IEEE Catalogue Number CFP09575-CDR) 2009, pp. 81-84

MINSKY, M.: *The Society of Mind*. Simon & Schuster, New York, 1986

ROLAND, E. M. A., SIPPER, M., CAPCARRERE, M. S.: Design, observation, surprise! A test of emergence. *Artificial Life* **5** (1999) 225-239

Keywords

Creativity, interactive art, emergence, test of emergence, society

From a “Flap of a Butterfly Wing” to the “Wind of Change”

Eliza Stefanova, *eliza@fmi.uni-sofia.bg*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Nikolina Nikolova, *nnikolova@fmi.uni-sofia.bg*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Eugenia Kovatcheva, *epk@fmi.uni-sofia.bg*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Evgenia Sendova, *jenny.sendova@gmail.com*
Institute of Mathematics and Informatics, Bulgarian Academy of Science

Elitsa Peltekova, *peltekova@yahoo.com*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Nelly Hubenova, *nelh02@gmail.com*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Maria Miladinova, *mariamiladinova@yahoo.com*
Executive Environment Agency in Bulgaria

Ekaterina Kantardjieva, *ekaterina.kantardjieva@gmail.com*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Abstract

The paper deals with a specific case of the authors’ experience in combining new technologies with innovative pedagogical approaches inspired by the Papert’s constructionist ideas. During the *Audio-Visual and Information Technologies in Education* (ATIVE) course in the e-learning Master of Science program, the students and teachers started joint research on a topic “We’ll meet again in 10 years” with emphasis on the future of education. The main goal of the course leaders (the first four authors) was to put in action different approaches for effective learning. The constructionist idea for creating a meaningful product (containing something new) was implemented by means of ICT social networks where all intermediate products were shared and discussed.

The paper presents the activities and the first results of the *School-of-the-Future* learners’ team (the last four authors) - studying recent research, writing analytical reports on the current situation, designing and conducting questionnaires and video interviews with students of different ages, processing them and presenting the results).

The overall experience within the *School-of-the-Future* project made the course leaders optimistic of the potential of *collective intelligence* when harnessed in the development of a school model based on the lessons learned by the older generations and the dreams of the younger ones.

The work described is still in progress. Even if we think of the first steps being made as a “flap of butterfly wings” we surely wouldn’t like to cause an educational tornado. But we could hope for a “wind of change” at least…

Keywords

School of the future, ICT-enhanced skills, Teacher training, Team work, Work on a project

Introduction

If you do not design the future someone or something else will design it for you.
Edward de Bono

Nowadays the dynamic changes in economics, technology, political and social relationships have a crucial impact on the process of education. We shall focus on three aspects of these changes. The first one refers to the **requirements of the society** concerning the *products* of education. The business needs young people with good working-in-a-team and working-on-a-project skills. The company leaders are interested in workers with well-developed information- and presentation skills. They expect that schools and universities will provide them with such employees. At the same time the creativity-based society we are striving for puts emphasis on developing the creativeness of young people. In 2009 (The European year of creativity and innovations) the Institute for Prospective Technological Studies at the European Commission's Joint Research Centre started a large research project to identify if and how the educational institutions in different European countries contribute to the development of creativity. It focuses on the identification and development of gifted students, on the interdisciplinarity of education, on the application of ICTs in education, etc. The idea of lifelong learning has also matured and many European projects direct their efforts to its realization. The best way education can meet such expectations has been summarized by Seymour Papert (1999): *The choice we must make for ourselves, for our children, for our countries and for our planet is to acquire the skills needed to participate with understanding in the construction of what is new OR be resigned to a life of dependency*.

The second aspect of recent changes in the educational process is associated with **the educational environment**. The revolution in ICTs influences the infrastructure of schools and universities. Most learners have access to various electronic devices and to the Internet. Furthermore, research centers and technology oriented companies developed different types of educational software and Internet applications supporting lifelong learning, communities of practices, etc. Technology environments and tools for social networks cross the boundaries of personal life and enter school life.

The third aspect of resent changes in education is related to the **development of innovative pedagogical approaches** in accordance with society expectations for high-school- and university graduates**.** The approach appearing to be the most relevant with these expectations is *constructionism* (Papert, 1999) referring to everything that has to do with learning by making, with experiencing the construction of a meaningful product which could be presented and shared with others.

Some fundamental ideas of constructionism in action as experienced by the authors are presented further in the paper.

The Context

The research under consideration has been carried out in the frames of *Audio-Visual- and Information Technologies in Education* (AVITE) - a compulsory course for pre-service and in-service teachers at the Faculty of Mathematics and Informatics at the University of Sofia *St. Kl. Ohridski*. Since the AVITE course is a crossing point of technology and pedagogy it is natural that it has been dynamically changed in the last five years. The sessions (both lectures and the hands-on activities) are lead by a team of teacher educators striving to implement the best practices in integrating ICT in education as identified within recent European projects in the field.

The Actors

There were thirteen trainees in the course considered in the paper - six students in the e-learning MSc program and seven Bachelor students at the Faculty of Mathematics and Informatics, all working in parallel with their study.

The team of the course leaders was formed by the first four authors (further called *the educators*). In recent years, they have been involved in a number of national and European projects (IDWBL, TENCompetence, WebLabs, UNITE, ShareTEC, I*Teach, InnoMath) and their ambition has been to effectively merge different aspects of the educational process – technological infrastructure, pedagogical innovations, and creativity support. During this period, various aspects of the AVITE course (in which more than one thousand in-service and pre-service Bulgarian teachers have participated) have been used as a *live laboratory* in which the innovative teaching strategies based on the constructionist ideas are demonstrated at a meta level [Sendova et al, 2009a, 2009b, Stefanova et al, 2009a, 2009b].

The Approach

The educators' approach is based on a specific *I*Teach* (Innovative Teacher) methodology in which the notion of *ICT-enhanced skills* has been defined as a *synergy between the technical and the soft skills* expected to be transferable skills in the Life Long Learning society. Putting the emphasis on the development of *ICT-enhanced skills* has been addressed in the frames of Leonardo da Vinci *I*Teach* project (http://i-teach.fmi.uni-sofia.bg). The *I*Teach* methodology (Stefanova et al., 2007b) is based on active learning methods, viz. the student is in the centre of the learning process, the teacher is a guide and a partner in project work based on didactic scenarios encouraging the creative thinking of students.

Let us note that the soft skills expected to be developed during the project include team work (planning, task distribution, communication skills, conflict solving), information skills (searching for and selecting relevant information, critical thinking), presentation skills (selecting the most appropriate tools for a specific task, written and oral presentation of the project products). Furthermore, the project output is expected to be finalized ("put on the table") and sharable with others.

The Process

The general time-line of the course is presented in (Table 1) but the duration of each phase depends on the specific audience and the course duration.

As a rule the course starts with introduction of the participants followed by brainstorming on a specific theme chosen by the educators on the basis of a preliminary analysis of the students' profile.

This time the opening introduction of the participants was provoked by the following questions: *Are you an expert? In what? Why? Who was your teacher? What makes a great teacher?* Apart from the general amusement caused by the fact that hardly anyone considered himself/herself an expert the educators were impressed with some opinions concerning the features of the "good teacher" and the "good school".

- *My school was not what it should be, because the pupils were not challenged.*
- *The IT teacher of my son has no prestige because "she is not willing to develop herself; she is just reading from the text book and does not encourage the more enthusiastic pupils to share their knowledge with her and their peers."*
- *After a spelling mistake a teacher in English apologized to her students. "Don't worry, this is just a word, and you corrected it. As for us, we don't know so many things you could teach us…"*

Table 1. The phases in the course delivery and their objectives

Phase	Course Delivery Mode	Objective
Presentation of the participants	face-to-face	To capture the participants interests so as to address them later in the course
Brainstorming / lateral thinking on a specific theme	face-to-face	To define topics of possible projects to work on
Forming teams around the project topics being formulated as subthemes	face-to-face	To start building (to enhance) team work skills
Planning the work on the project	face-to-face	To start building (to enhance) skills for working on a project
Short (5 minutes) presentation of the developed plan	face-to-face	To enhance presentation skills
Working on a specific project in a team	distance	To develop ICT-enhanced skills (with emphasis on information skills, working-on-a-project skills, working-in-a-team skills)
Presentation of the work done and the projects' results	face-to-face	To enhance presentation-, evaluation- and self-evaluation skills

These observations were a starting point of a heated discussion about the role of the teacher in contemporary school. The participants shared the understanding that children are a great resource of ideas to be used by teachers.

Finalizing the participants' presentations and the discussions around the questions, the educators proposed the following formulation of the general theme: *We'll meet again in 10 years*.

The next session started with brainstorming (Figure 1) - *what sub-themes do participants associate with "meeting in ten years"?*

Figure 1. The brainstorming

Here are the first topics that came to mind:

- *Alumni reunion*
- *The dreams (now) and the realities (in 10 years)*
- *Today's students as future teachers*
- *The future school vs. the school of the future (pessimistic vs. optimistic visions)*
- *The most fruitful decade in our life*
- *The challenges we are ready to face*

Afterwards the participants clustered around the sub-themes and formed project teams accordingly. One of the groups (the last four authors) focused on the *School of the Future* and it is its work to be discussed below.

The Project Itself

The Organization

Given their extensive experience in using network tools the educators were happy to see that the *School of the Future* team organized its work by forming a Google group, ***fmi avito,*** and invited them to join in -

The first activity of ***fmi avito*** was to make a plan, to distribute the roles and the tasks among its members. The latter found the group tool to be very convenient for staying in touch and collecting new information in a single shared place. The social group **fmi avito** is being used by the educators and trainees to share new ideas, documents, and to work collaboratively on common tasks (Figure 2).

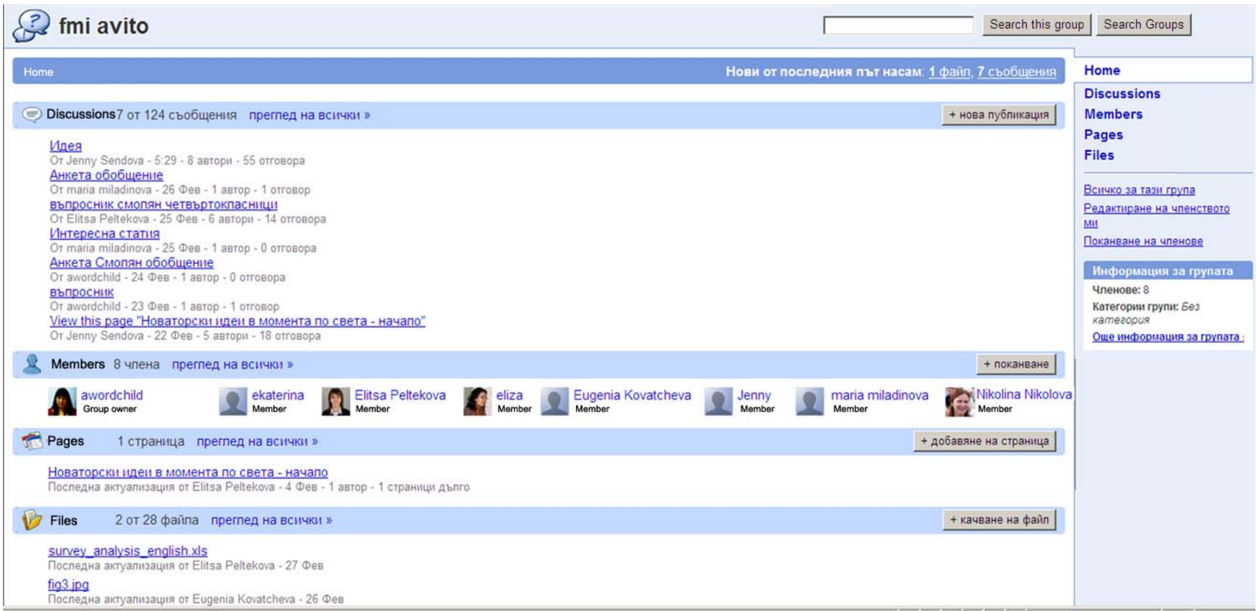


*Figure 2. The homepage with the main activities and participants of the **fmi avito** Google group*

When appropriate the educators would establish connection between experts in the field and the members of the *School of the Future* team. First, the trainees received in their "e-mail box" a link to the works of a well-known educator, Rachel Cohen from France, who is the initiator of the international literacy project *Mini web, Multilingue, Maxi Apprentissages* (MMM). The goals of MMM are in full harmony with the educational principles adopted by the learners' group and their vision about early education – *to enhance cognitive, social and technical skills from an early age, in order to allow children to communicate, to exchange, to have access to information, to construct their own knowledge, and to establish educational strategies allowing teachers to give a new power to their students: autonomous access to sources of information, construction of knowledge through distant cooperation and sharing* (http://www.mmm-ec.org).

Next, the trainees were stimulated to work collaboratively on an international study on a similar topic - *future of learning together*. They expressed their enthusiasm and satisfaction from such interaction with experts on international level which turned their project work into a component of genuine research.

The next milestone of the project was to design a questionnaire and conduct interviews with students.

The Questionnaire

The questionnaire on the *School of the Future* was proposed by the team leader but its final version was a product of joint efforts. The questionnaire consisted of seven questions focusing on three basic themes – what subjects should be taught in the school of the future, what should the teachers be like and what would the classrooms look like according to their expectations. Each of the topics included up to three questions with an open answer. The students were expected to provide up to three suggestions best matching their vision of the *School of the Future.* As for the classroom, it could have been drawn or described verbally.

The educators' team approved the questionnaire and helped the trainees to collect data from 12-graders (18-19-year-old students) - 30 questionnaires were filled in by these students one day later.

Several days later the learners continued to collect data from twenty one 4-graders (10-year- old pupils) and from 10 students having just graduated (25-38 years old).

The Interview

To arrange an interview with high-school students involves a lot of preparation and administrative authorisation. Thus, the educators were involved in providing permission from parents and assisting the *School of the Future* team in conducting the interview with four 7-graders (14–year- old students).

The Feedback

The most valuable part of the *School of the Future* project was its authenticity – the team was genuinely involved in presenting a vision of the school reflecting the opinions and the dreams of a variety of people – younger and older pupils, teachers, parents, researchers, and participants in educational experiments in the recent past. Thus, the final presentation and the discussion of the project were just a milestone (almost a side effect) of real research – this was the general feeling of the team.

What follows are some representative examples of the feedback provided by the participants in the team survey.

The Classroom of the Future

„Please, draw, paint or describe how you imagine classroom of the future" - this was the interview task that provoked the greatest varieties of suggestions – two examples are given in Figure 3.

As expected, technologies are present and the environment is stimulating – there are flowers and curtains (in the drawings of the younger students) and a non-conventional arrangement with a lot of corners providing facilities for working on your own and in a group (the 12-graders). Options for on-line communication with experts in the field of study are also envisaged.

It is interesting to note that although the prevailing answers to the question *What will not be present in the classroom of the future?* were of the kind*: the black/white board,* there were boards in their pictures. Another frequent answer (especially among the older students) to the same question was *separate desks.* This could be related to the wish of the students to work in teams (including their teachers and experts in different fields). These students' expectations are in accordance with researchers' expectations (Leis Miriam 2010) for globalization of education. A Common understanding of many researchers and pupils is also that *there will be no textbooks*.

Figure 3. "A Classroom of the Future" according to a 12-grader (left) and a 4-grader (right)

Although most of the younger students imagined the future classrooms rather conventionally (with desks in the usual type of rooms) there were still some who would prefer an open-air classroom. Some non-traditional elements in the imaginary classroom suggested by representatives of all age groups show that they would like the classroom of the future to be cosy.

There was even a counter-question asked by a 12-grader: *Are you sure that the classroom will exist?*. This question is in full harmony with some researchers' expectations that the future *classes will not be limited in terms of age, distance, etc.* With wider use of technologies in education the notion of the classroom will be changed dramatically and will be far from the traditional today's understanding. The main conclusion of the research was that the older the students – the braver their visions of the *School of the Future*, probably reflecting their awareness of how much the changes depend on them.

The Future School Subjects

According to the interviewed pupils (of all ages) the basic subjects in the school of the future (Figure 4) would still be mathematics and languages - a strong indicator of the general perception of the importance of these subjects in students' future education and careers.

As for some new subjects in the *School of the Future* most of the students in the senior classes shared their wish to study ecology and how to behave in society. The fact that these suggestions ignore reality of such subjects already being included in the school curriculum shows the irrelevance of the way they are being taught.

Such correspondence mismatch among existing school subjects and the expectations of the 12- and 4-graders was also observed in the case of Person&Society (studied in Grade 4) and World&Personality (studied in Grade 12. These subjects were mentioned as unnecessary in the school of the future. At the same time a subject expected to provide knowledge about how to live in contemporary society was included in most of the wish lists of subjects in the *School of the Future.*

The teams of both educators and trainees expected that technology would constitute a much greater part of the students' vision for the school of the future. It seems, however, that students do not necessarily link the acquisition of knowledge and skills with modern and future technology development, although they tend to acknowledge the increasing presence of ICT in their learning environment. Apparently students perceive their interaction with both classmates and teachers as being more important than the use of traditional or technological means to achieve it.

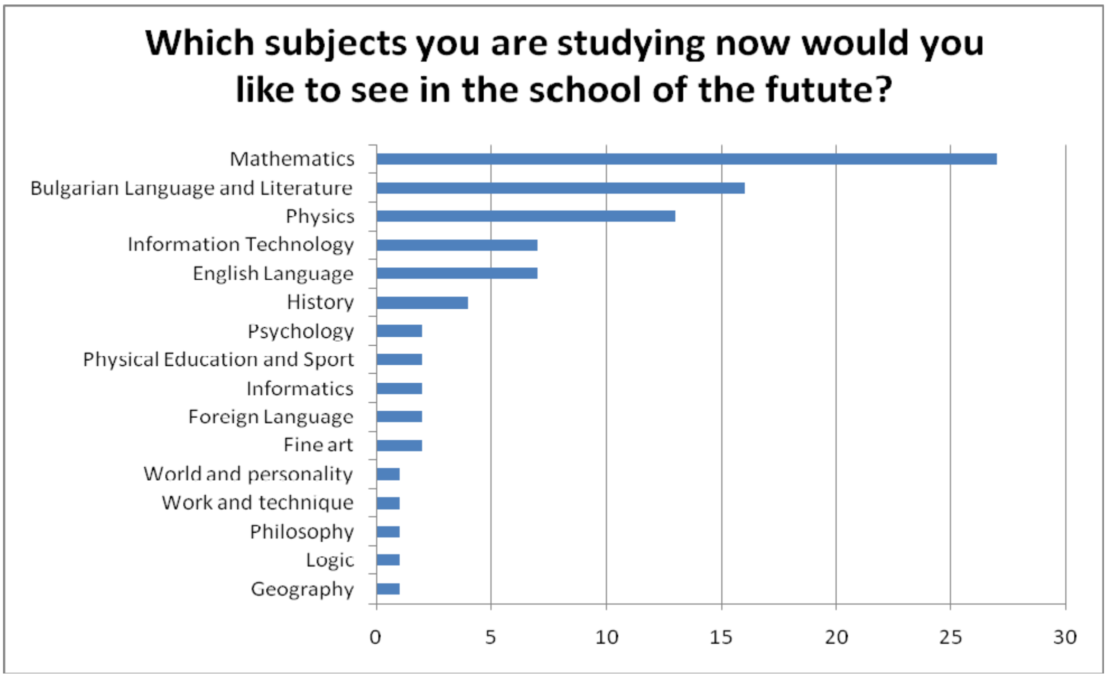


Figure 4. The subjects in the School of the Future

The Teachers of the Future

There is a serious overlap in the expectations (wishes) of the four and twelve graders in the answers to *What should the teachers of the future be*?. The most frequent answers were:

- motivated", „keen on their work"
- correct and fair
- thinking out of the box
- provoking students to think"
- encouraging all students
- fun, witty
- with good social skills
- kind, good, considerate

The respondents aged 25 – 38 think that the most important features the teachers should possess are: knowledge in the area of the subject taught, ability to communicate with students, being good psychologists and the ability to provoke the interest of the pupils.

Conclusion

The *butterfly effect* is a metaphor often used to demonstrate that insignificant changes in the initial conditions could cause dramatic effects, e.g. a flap of the wings of a butterfly in NY could cause a tornado in Tokyo. While the butterfly does not "cause" the tornado in the sense of providing the energy for the tornado, it does "cause" it in the sense that the flap of its wings is an essential part of the initial conditions resulting in a tornado, and without that flap that particular tornado would not have existed.

We made some modest steps towards constructing a model of the *School of the Future* in the frames of a teacher education course. The overall experience of the School-of-the-Future research team made us optimistic of what is possible to achieve when harnessing *collective intelligence* (Cornu, 2006) in the development of a model of education based on the experience of the older generations and the dreams and visions of the younger ones. The work described is still in progress but it gives an idea of how knowledge construction and sharing could be promoted not just as a primary goal in teacher education, but also as a good practice the teachers-to-be could apply in their future work.

Even if we think of the first steps being made as a “flap of butterfly wings” we surely wouldn’t like to cause an educational tornado. But we could hope for a “wind of change” at least…

References

Cornu B. (2006) *Collective Intelligence*, Conference talk, ISSEP 2006, November 7–11, Vilnius, Lithuania.

Leis Miriam (2010), Challenges for the Future of Learning Until 2030: Foresight on Learning, Innovation and Creativity, LearnTec 2010

Papert, S. (1999) *Introduction: What is Logo? And Who Needs It?*, in *Logo Philosophy and Implementation*, LCSI, p. IX

Sendova, E., Boytchev,P., Stefanova,E., Nikolova, N., and Kovatcheva, E. (2009a) *Creating a Natural Environment for Synergy of Disciplines.* In Proceedings of 4th European Conference on Technology Enhanced Learning - Learning in the Synergy of Multiple Disciplines, EC-TEL 2009. Eds.: U. Cress, V. Dimitrova, and M. Specht, LNCS 5794, Springer-Verlag Berlin Heidelberg, Nice, France, September 29-October 2. pp. 549-555.

Sendova E., Nikolova N., Stefanova E., Boytchev P., Kovatcheva E., (2009b), *Harnessing ICT for building a creativity-based society.* In Proceedings of the 4th IT STAR Wrkshop on ICT Skills, Education and Certification, G. Occhini and P. Nedkov (eds.), 27-28 November 2009, Rome, Italy. pp. 34-49

Sendova E., Stefanova E., Nikolova N., Kovatcheva E. (2008) *Like a school (of fish) in water (or ICT-Enhanced Skills in Action)*, LNCS Informatics Education - Supporting Computational Thinking, Edited by Roland T. Mittermeir and Maciej M. Syslo, July, 2008. pp. 99-109.

Stefanova E., Boytchev P, Nikolova N., Kovatcheva E., Sendova E. (2009a) *Embracing and enhancing ideas as a strategy for ICT education*, In proceeding of International conference on Multimedia and Information technologies in Education (m-ICTE), Volume 1, pp. 206-211.

Stefanova, E., Nikolova, N., Kovatcheva, E., Boytchev, P., Sendova, E. (2009b) *Myths and Realities about Technology Enhanced Learning.* In Proceedings of First International conference Software, Services and Semantic Technologies (S3T), D.Dicheva, R. Nikolov, E.Stefanova (eds.), Sofia, Bulgaria, October 28-29. pp.116-123.

Stefanova E., Sendova E., Nikolova I., Nikolova N. (2007a), *When I*Teach means I*Learn: developing and implementing an innovative methodology for building ICT-enhanced skills*, In abstracts and proceedings of the Joint IFIP Conference: WG3.1 Secondary Education, WG3.5 Primary Education: Informatics, Mathematics, and ICT: a 'golden triangle' IMICT 2007 , CCIS, Benzie D. and Iding M. (eds). Northeastern University, Boston, MA.

Stefanova E., Sendova E., v. Diepen N., Forcheri P., Dodero G., Miranowicz M., Brut M., et al (2007b) *Innovative Teacher - Methodological Handbook on ICT-enhanced skills*, Faleza-Office 2000, Sofia.

LEGO and LOGO in the primary school – a simple way for learning through creation

Vessela Ilieva, *vessela.ilieva@abv.bg*
Private Language School "St. Cyril and St. Methodius", Sofia, Bulgaria

Abstract

The introduction of "IT" in primary school acquaints the children with the possibilities of the contemporary computer systems and the variety of their applications. It provides the acquisition of basic knowledge and practical skills of working with computer which gives the children the opportunity to use computers for accomplishing their own ideas and projects.

For that purpose the IT education should provide:

- facilities accessible, understandable and attractive to the child;
- ensuring great variety of activities and permanently active role of the child in the learning process;
- the opportunity to work on topics and issues of interest to the child itself and being directly related to its actual life experience;
- the creation of a particular product that is valuable from the child's point of view and is able to "materialize" the invested skills and efforts.

Figure 1. LEGO and IT activities

In that context this paper deals with the inclusion of LEGO in the educational program in order to enrich the teaching of IT with new meanings and as an instrument of providing a larger variety of activities in the teaching process. Using the computer as a tool for managing and control of external objects (in this case LEGO models, created by the children themselves) reveals one more remarkable area of its application. That is also a natural motive of introducing the children to programming in a way attractive, interesting and accessible for them. The result of programming is very attractive and devoid of abstraction. It is easy to formulate as a prior expectation and clear to describe step by step, as it concerns behaviour that the child knows from the real life.

Keywords

Primary school, IT education, LEGO, LOGO, programming

Introduction

In Bulgaria, a non-mandatory national curriculum for learning to work with computers in primary schools was implemented in August 1998. It was introduced as a “free elective subject.” The curriculum, in 11 modules, was written by us and had been developed within our school over the previous year (Illieva and Ivanov 1999). Its main goal was to use the capabilities of the computer as a powerful new medium to challenge children in the context of normal activities for their age. This goal is attained by a project-oriented approach that we consider fundamental to the process of education. Children implement their ideas in projects that are based on personal experience and important events in their lives. Each project leads to the accomplishment of different types of activities, which are drawn from the school curriculum. In this way the children accept an active and creative role. This is why one module was “Working with computer systems for the control of models.” It was based on my work with children and LEGO-LOGO in the primary school over the previous six years. This and other modules were optional for technical and financial reasons.

In 2006 learning to work with computers became an “obligatory elective subject” in primary school. This subject was given the name “Information Technology.” National standards were set and a new curriculum was written by a team of which we were members. Both can be found on the Ministry of Education website, http://mon.bg.

The first curriculum remains an option for primary schools as a free elective subject. So, LEGO-LOGO, which was not included the new curriculum, is still possible.

The primary school is the specific element of the system of education where the child acquires initial knowledge and skills in a diversity of areas. At the same time the child also develops basic and enduring habits, concepts and attitudes to everything studied, to the teaching itself (including the place, the people and the approach related to it) and the learning as a process of individual intellectual activity. In this sense the primary school is a fruitful territory for any kind of novelties because they are perceived spontaneously and positively by a population free of fears and prejudice. It is at the same time a dangerous territory from the point of view that this is exactly a time when bad habits and attitudes can very easily be formed and reinforced.

The introduction of new technologies in the primary school provides many opportunities; and their application can serve the achievement of various objectives. The new technologies can influence both the educational environment in its complexity and any particular process of leaning and teaching. This is regardless of whether they are introduced as an independent subject of study, or used in a specific manner within other subjects of the curriculum.

Therefore regardless of the early age, new technologies should be presented in the full diversity of their multi-functionality. By this means the child may obtain a general overall, rather than deeper but disjointed, concept of the possibilities offered by the computer as a technical means. The child will become acquainted with most of the many applications that they may later use.

Considering the specific nature of the age-group – it is demanding for any such project to provide for:

- facilities accessible, understandable and attractive to the child;
- a great variety of activities with a permanently active role for the child in the learning process;
- the opportunity to work on topics and issues of interest to the child him or herself, which are directly related to their actual life experience;
- the creation of a particular product that is valuable from the child's point of view, and that is able to "materialize" the invested skills and efforts.

IT and LEGO activities

Ever since the creation of LOGO as a programming language and a pedagogical concept, numerous and varied LOGO-based microworlds have been developed and used worldwide. The purpose of each microworld was different. They were designed to organize specific pedagogical situations where the child acts and learns by using the tools and options of the microworld. In the proceedings of the Eurologo conferences since 1987 many such microworlds are documented.

To support the curriculum and provide teachers with materials that worked well with children we created a LOGO-based software package, “Tool Kid”, containing 48 small programs/microworlds divided into 7 groups, specially designed to introduce working with the computer to young children. With their help children are learning to handle the mouse and the keyboard. They investigate the properties of the computer and use the tools necessary for the treatment of various types of information – graphics, text, sound, animation and video, individually and in combination. Development took several years including rigorous school trials (Illieva and Ivanov 2001, Ivanov and Ilieva 2005). Tool Kid was published, with teacher and student books (Illieva and Ivanov 2003-2006, Ilieva and Ivanov 2004-2007). It is used by all primary schools that teach IT.

In the information technology classes I now teach, I use the microworlds in Tool Kid too☺. The key principle laid down in the creation of these programs is that in any of them the child should have appropriate environment, tools and options to act and learn, using them to solve a case or create a particular end product. The product can vary – it may be just a puzzle to solve or a picture to color, but it may be also a personal graphic project, a story, comic, multimedia card, slide show, film. The important thing in any such case is that the product is created by the child personally. She or he has to apply, in the process of creation, the knowledge and skills acquired; to improve them; and even acquire new ones.

Figure 2. Examples of children’s work in IT lessons

When acquired knowledge or skill can immediately be used in practice towards the achievement of a particular goal, one which is personally important, this generates interest. From this comes motivation to acquire more and more knowledge and skills; and put conscious, voluntary effort to this effect. This turns the child from an object of education into an active participant/subject in the process.

LEGO

If LOGO is a high-performance programming environment that has enabled us – the adults, to create a wide range of microworlds dedicated to one or other specific purpose and where children can learn by acting, LEGO is also a high-performance material environment, but one in which the children themselves create their own microworlds. Again it is of the greatest importance in this situation that they learn through action.

Working with LEGO constructional materials the children come to know the surrounding world by recreating it. The elaboration of any model places the child in a situation of very dynamic activity.

This is not merely the activity of manipulating elements and building a structure. The situation encourages the child to remember, examine, juxtapose and analyze the object in depth both as a whole and in its details; to seek and find the relations and interactions between parts; to realize functions, purposes and dependencies. The active action is not merely a complement - it is born and needed by the highly intellectual activity, and the great emotional attachment of the child to what they are doing.

LEGO and IT

In this context I have been using LEGO in my classes as a part of teaching about and with the new technologies. It initiates the children into the world of programming by showing them how to use the computer to control and operate the functioning of external objects.

The fact that these objects are actually the models they have created is almost guarantees the children's personal commitment to the activity. Basic principles are being learned in the course of the practical activity of model operation and control. Complicated explanations are avoided.

Both construction and programming require and develop the following skills:

- imagining the whole;
- analyzing details;
- forecasting consequences;
- realizing correlations;
- following sequences;
- seeking precise expression;
- seeking options

The result of programming is very attractive and devoid of abstraction. It is easy to formulate as a prior expectation and clear to describe step by step, as it concerns behavior that the child knows from the real life. The errors are quite flagrant and the disappointment – quite strong.

The emotional involvement of the child in the model emerging in the course of its creation is transferred to programming which appears in this case as the final stage of the construction event. This is a strong motivation to maximum concentration, to seek options, for error identification and troubleshooting.

Organization of the work

For twenty years I have taught LEGO and IT in primary school. I began this work in an experimental school and continued it in a private school. Both schools made it possible for what I wanted to do to be a full part of the school curriculum. So, in my present school, all pupils from 1st for 4th grade take two class hours of information technology and two class hours of LEGO construction weekly. To conduct the classes the school equipped a computer room with workstation for each pupil and a LEGO room with enough construction material to enable the common and simultaneous work of all the children in a class.

In the IT classes the children acquire basic knowledge and skills of working with computer systems and information technologies by creating their own products during their work on various projects.

In the LEGO classes they also become familiar with the structural material, the specific properties of each element and its possibilities, by gradually starting to create models. As models become more and more complicated there is a switch from individual to team work.

Like teaching computer skills, LEGO teaching breaks down construction into necessary skills. These are learned, applied, and refined through all four primary classes.

The integration of both subjects starts at the end of the second and the beginning of the third year. The knowledge and skills acquired in the information technology and English language classes, the latter being studied since the first year, find their application in the implementation and programming of their first controllable prototypes. This continues to the end of the fourth year. The prototypes become more complex in structure and function. The greater diversity also poses more challenging programming tasks.

Inserting this course to the regular curriculum and providing the necessary technical support to conduct it has enabled all the children to participate on equal terms. They all participate in a continuous and consistent training evenly spread over the entire four-year span of the primary school. This makes it possible to establish continuity between and relations among the year groups.

Themes and models

In her/his work the teacher is often facing a situation when she/he has to act as mediator between the new educational ideas, technologies, means and materials, and the child – the end user for whom they are intended. The teacher reduces the overall idea to a sequence of specific steps and actions in order to make possible its practical realization. S/he negotiates the interaction between the idea and the child to whom it is intended. In construction this is carried out through the topics on which the children work and the prototypes they are creating.

In my work with controllable prototypes I use the LEGO Data Control Lab. For the primary school age group this system provides me all the necessary devices and tools to work with the children in a wholesome way. The only problem is that the published technical schemes of models are not intended for primary school kids and therefore they are not appropriate for them. The models are too complex as structures, the objects to recreate are vague in terms of "behavior", and insufficiently attractive as ideas since children are very unlikely to meet them in their everyday life.

Therefore in thinking over the prototypes I bear in mind two things: a) the complexity of the structure into which the computer-controlled element is going to be integrated, and b) the complexity of the guidelines to be followed for its implementation.

Thus, I choose models that:

- Can be implemented by the children as a construction task with their available skills and materials;
- Naturally presume the presence of controllable devices – motors, lamps, sound elements, sensors;
- Will be attractive in both appearance and "behavior";
- Recreate objects familiar and interesting to the children;
- Recreate "behavior" that could be met in real life, and that is simple and clear enough to be described and then programmed.

The task is never reduced to the mere elaboration of an isolated controllable model. It should be an element of a situation reproducing as closely and realistically as possible the object's natural environment and its functioning within it.

The design of situations is more motivating, more challenging and more creative than the mere creation of a isolated prototype. It is richer in correlations, interactions and dependencies which should be sought, identified and recreated deliberately; which implies better cognition. The situation around the controllable model and its interlacing with other surrounding objects as it is in real life makes its "behavior" more authentic. The behavior is more understandable as it is somehow dependent on its ambiance and interacts with it. This helps the children to see more clearly the algorithm of functioning of the model and comprehend its purpose. In turn, this gives more sense to programming and reduces the degree of its abstraction. Moreover the modeling of situations is conducive to the organization of team-work. The objects are numerous but they are all elements of a large comprehensive project. Every child participates actively in creation, making his/her individual contribution. Another advantage is that there could be more than one controllable object which makes programming richer, more sophisticated and more open to variation.

Projects

Street lighting

I use street lighting first, to teach the elements of the operation and control system including: computer, interface block, model with integrated control device, and cables for the connections in between. The purpose and functions of each of these elements is explained.

The modes of cable connection to the different devices and the test port functions are shown. The operating algorithm, main interface of the operating program, introducing the control commands and their execution in direct mode of operation are demonstrated.

Figure 3. Staring to program their street lights using LEGO DACTA Control Lab Logo

Children's own experience – they have all seen the street lights and know that the lamps are switched on every evening and switched off in the morning and they are aware that this is being operated centrally and not by employees walking about in the streets to make this manually. The project includes the construction of a city with streets and street lamps placed along them.

Controllable devices – lamps.

Commands – *talk to*, *on*, *off*, *wait*, *repeat*.

The work on this project is continued into the third year, by including use of light sensor in order to associate the lamp operation to a particular condition. This entails the natural necessity to introduce the *waituntil* command. The work is no longer in direct but procedure mode.

Traffic light

The project incorporates the elaboration of a small village with one main street with a pedestrian crossing with a traffic light on it. One side of the street is bordered by the houses. On the other side are – the shop, the post office, the restaurant and the school. The traffic light is necessary to help people reach the place they live, work and amuse themselves without risk of accidents.

Figure 4. Building the street, and connecting and programming the traffic lights

The connection of the devices, the operating program opening and the service check are being performed without help from the teacher.

Children's own experience – they know the traffic rules, they have observed how the traffic lights work day and night, they can describe the algorithm of operation and associate it to their own behavior as pedestrians. Analysis of the traffic light operation in night mode – blinking orange light.

Refreshing the commands of the previous session, their meaning and the consequences of their execution.

Controllable devices – lamps.

Commands – Work in direct mode where the children try their own various hypotheses about the sequence of commands to put the traffic light in proper operation. The traffic light is in continuous daytime mode of operation. Introduction of the *onfor* and *forever* command.

A similar project is being worked on in the third year, this time in the city with traffic light for pedestrians and cars at a crossroad. It includes the use of *sound* element as a sound signal to the traffic light, intended for people with visual impairments. This makes it possible to associate the operation of a device, i.e. a lamp with another device, i.e. a sound element. In this situation it is necessary to think about not only the programming solution but also the scheme of connecting the devices to the interface block. The project can be further developed by including a light sensor and linking the traffic light functioning to a condition, according to which it is switched from daytime to night mode and vice versa, as was done with the street lamps.

The work is organized in procedure mode. The idea of a main procedure with sub-procedures is introduced. The fourth-year project work is focused on a crossroad with functioning traffic light and vehicles following the traffic rules: to move ahead when the light turns green. The vehicle operation programs include the use of touch sensor enabling the kids to operate them.

Windmills

Introducing a new controllable device – the motor.

Functions – to drive the windmill propellers.

Learning how the motor is incorporated within the windmill prototype structure. The project includes the construction of their own windmill by every child – before that a conversation is held on its purpose, appearance, functioning.

Figure 5. The windmills

The children's own experience – they have seen working windmills and know what they are used for. A situation including the participation of people and animals should be recreated around each windmill model.

Model operation – working in direct mode. Introducing the *rd* command for the direction of rotation of the axis driven by the motor. Every child creates their own program to operate their windmill. A condition is set for the windmills to work on different algorithm. All the patterns form an integral model through connecting pathways and landscape designed around them. This project is worked on in the third year. The *setpower* command is introduced.

Operation – the work goes into procedure mode. Every child first creates a procedure to operate its own windmill and thereafter creates one main procedure to bind together the functioning of all other mills in a common sequence. The "behavioral" idea of the models is suggested by the children.

Discotheque

A group project the purpose of which is to make a model of discotheque with air conditioning and programmed lighting.

The children's own experience – they know how a discotheque looks like, what do people do there, they have observed the light effects and are aware of the presence of air conditioning system and its purpose.

Introducing a new device – temperature sensor and associating it to the operation of the fans. The lights programming here requires maximum attractiveness and diversity. The *forever* command is introduced.

Figure 5. Ventilator control procedure for the sensor and motors, at the back of the discotheque

Operation – working in procedure mode. Separate small procedures are created for the various types of light effects operated by one main procedure. The main procedure also includes the fan control procedure. A project entitled "Circus" can be developed as a version of this project.

Police Action

This project is being worked on in the fourth school grade. It is intended to recreate a story with a plot. The situational model includes a street with several houses and a police station. One of the houses is equipped with alarm system. A light sensor is installed in the anteroom against the front door. There are signal lamps installed in the police building. A prototype of police car is made, which embodies an engine and siren. Another car prototype with an engine is made for the thieves.

The task is to make a program so that the thieves' car starts first and stops in front of the house front door. While the front door is open, the light sensor detects increased lighting. This causes a blinking signal light to go on in the police station. Five seconds after it starts blinking the police car should set off with siren sounding and stop right in front of the house.

Procedure mode is used with main procedure and sub-procedures. The class is divided into teams of 2-3 children: The team constructing the house is in charge of programming the sensor control; and the team designing the police service is in charge of the signal lamp programming etc.

Figure 6. Police – Action!

Christmas Town and Amusement Park

These are collective projects with the participation of all the children from 1st to 4th grade. The first is made before the Christmas holidays and the second – for the school year end.

Figure 7. Christmas town, with programmed lights on the tree and in the houses

The model is large and every class takes part in it by implementing some part of the common subject. The 4th grade pupils are in charge of the programming.

Figure 8. 4th grade watching their program work – and a close-up of another ride

In summary

Organizing the IT education as a work on personal projects, which leads to the creation of an end product with a personal meaning to the child, provides a different context in applying knowledge and skills; and guarantees the child's personal activity during that education.

The teamwork during these class hours and the group projects of the class in the IT and LEGO lessons is very important for the children themselves and the school life in general. Many ideas are generated in the teamwork, the communication is dynamic, the disputes which arise cultivate the ability to find arguments, to compromise. This environment is beneficial to multifaceted and diversified reasoning, to the formation of ability to take independent decisions and bear the consequences of it. The traditional teaching quite rarely provides natural opportunities and necessity to work in a team.

Regardless of the fact that I have discussed these school sessions mainly within the context of the IT classes I believe that they are influencing in a specific way the entire development of the children, the school environment and the pupils' attitude towards studying and education.

References

Ilieva, V. and Ivanov, I. (1999) *Informatics and Information Technologies in Primary School – based LOGO Environment.* In Proceedings of EuroLogo 1999. Edited by R. Nikolov, E. Sendova, I. Nikolova, I. Derzhanski. Sofia, August. pp. 227 − 234.

Ilieva, V. and Ivanov, I. (2001) *My First Computer Software*. In Proceedings of EuroLogo 2001. Edited by G. Futschek. Linz, August. pp. 277 − 285.

Ivanov, I. and Ilieva, V. (2005) *ToolKID – Logo language based software package for children*. In Proceedings of EuroLogo 2005. Edited by G. Gredorczyk, A. Walat, W. Kranas, M. Borowiecki. Warsaw, August. pp. 390 − 397.

Ilieva, V and Ivailov I, (2003 – 2006) *Компютърът в началното училище, Vols 1-5*, Sofia, Regalia 6

Ilieva V and Ivailov I (2004 – 2007) *Информационни технологии, 1-4 клас, Vols. 1-4*, Sofia, New Star

The crop circles – an inspiration for project-based learning in a Logo environment

Ridvan Isufov, *ridvan@fmi.uni-sofia.bg*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Bahsen Ismailova-Isufova, *bahsen@abv.bg*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Nikolina Nikolova, *nnikolova@fmi.uni-sofia.bg*
Faculty of Mathematics and Informatics, Sofia University St. Kl. Ohridski

Abstract

This paper presents a specific project (*Modelling the crop circles*) developed by the authors in the frames of a teacher training course on Logo and ICT. Because of the variety of crop circles being documented it is possible to formulate a rich set of problems addressing the development of both mathematics and informatics skills. During the course, together with mastering the Logo language, the authors experienced such important skills as team work, distributing the tasks, planning ahead, searching and selecting the relevant information and sharing the final product with an audience (all of these ICT-enhanced skills being crucial components of the constructionism). As part of the project they offered ideas for encouraging junior high-school students to look for Logo realisations of the crop models which could be qualified as "the best" according to different criteria – closeness to the original, readability, potential for generalizations, etc.

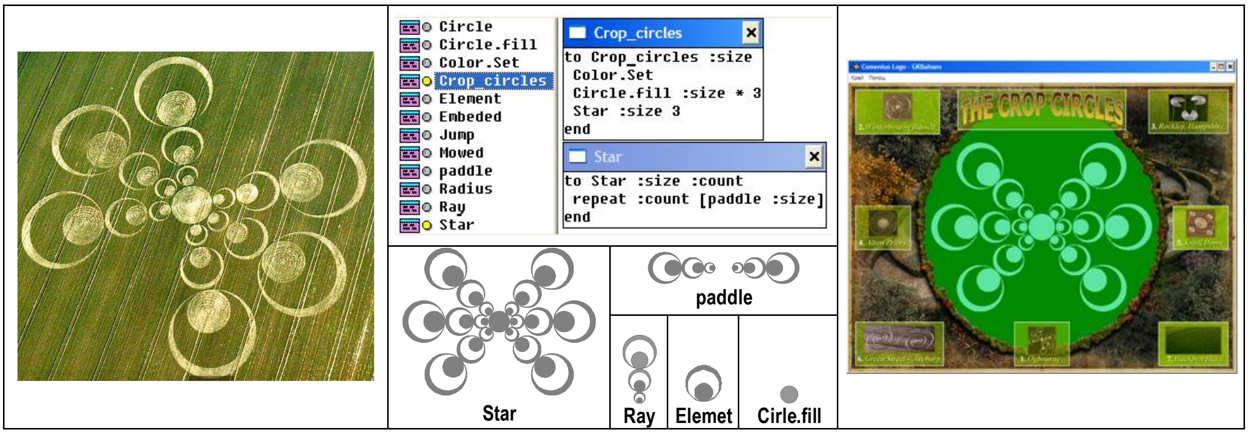


Figure 1. Modelling a specific crop circle configuration by means of Comenius Logo

As an example, the modelling of a specific crop circle configuration is considered together with the mathematics and informatics knowledge needed (Figure 1). The configuration itself represents a building block of a bigger project (a set of crop circle configurations chosen by a team of students).

Keywords

Project-based learning, ICT-enhanced skills, crop circles, mathematics, informatics.

Introduction

The idea of the project presented in this paper was born in the frames of the course *Programming languages and environments in education* (lead by Jenny Sendova) meant for in-service teachers in mathematics who would like to teach in addition informatics and information technologies. The participants are expected to prepare and defend a project that could be used as a model for project-based learning. One of the challenges the IT teachers-to-be are facing is to find motivating themes that correspond to the various interests of the students and could stimulate their research potential. New pedagogic strategies and approaches (such as the team work, project-based work, the teachers acting as partners in the learning process) have also been implemented in this course (Stefanova et al, 2007). In search of objects which could be easily modeled by means of Logo and at the same time provoking the interest of a larger audience we came across the crop circles.

The project design

The background

The crop circles are sizeable patterns created by the mysterious flattening of crops. These patterns appear in one night and usually there are no footmarks left around them.

The earliest recorded image resembling a crop circle is depicted in an English woodcut pamphlet published in 1678 called the "Mowing-Devil". The image depicts a demon with a scythe mowing an oval design in a field of oats. The pamphlet's text reads as follows:

> *Being a true relation of a farmer, who bargaining with a poor mower, about the cutting down three half acres of oats, upon the mower's asking too much, the farmer swore that the devil should mow it, rather than he* (http://www.rense.com/general39/mow.htm).

There are various theories about the origin of the crop circles. For the UFO supporters, the circles are signatures left behind by visiting spaceships. For mother-earth mystics, they're the manifestations of deep waves of natural energy. For psychics, they're the conscious results of remote-viewing experiments. For fringe physicists, they're the tracks of ionized plasma whirlwinds. But the most likely is that the pranksters or circlemakers are human that take fun in building such weird circles. As yet no conclusive evidence has been found for any of these theories (http://aliens.monstrous.com/crop_circles.htm).

Pictures of crop circles are presented in Figures 2-7 (http://www.lucypringle.co.uk/).

Figure 2. Ogbourne

Figure 3. Winterbourne Bassett

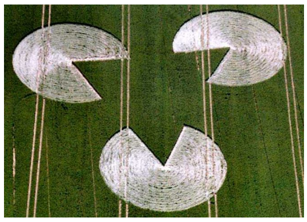


Figure 4. Rockley

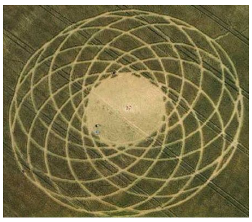


Figure 5. Alton Priors

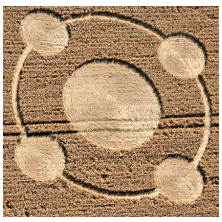


Figure 6. Knoll Down

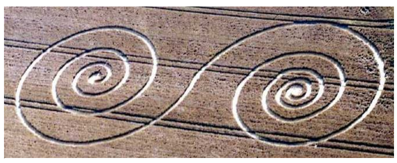


Figure 7. Hackpen Hill

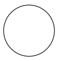
The whole mystery around the crop circles theme appears to be very intriguing for students with various interests and it grabs easily their attention. Furthermore the great number of publications on this theme and the variety of crop circles being documented could be used as a base for the formulation of a rich set of problems addressing the development of both mathematics and informatics skills.

Figure 8. Logo models of the crop-circle configurations in Figures 2-7

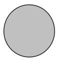
To illustrate this idea we are going to present the process of modeling specific crop circle configurations (Figure 8).by means of Comenius Logo (Blaho and Kalas, 1998).

Harnessing mathematics and informatics tools

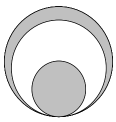
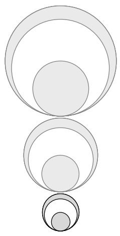
Let us consider now the modeling of the *Ogbourne* crop circle (Figure 2). This is a motive, which could be generated by means of geometric transformations (dilation, rotation and symmetry). The figure  could be considered as a building element which should undergo dilation (with ratio 2 and 3), then translation, symmetry, and finally - rotation. This makes it natural to create first a procedure for drawing a circle with a parameter for the radius, and then – a procedure for the building element , again with a parameter for the radius of the smallest circle. The first challenge for the students from mathematics point of view is to decide how to create a procedure for a circle (Sendova, Ivanov and Nikolov, 2002). If they decide to approximate it by a 360-gon with a parameter for the side length they might use the following procedures:

	**to** ***Circle*** :s **repeat** 360 [**fd** :s rt 1] **end**	**to** ***Circle*** :s :b **if** :b=0 [**Stop**] **fd** :s **rt** 1 ***Circle*** :s :b-1 **end**

This definition reflects better the intuitive movement along a circle. We fill it with color to imitate the real process of the crop of oats being mowed. It could be generated:

	**to** ***Circle.Fill*** :s **Let** “x **Xcor Let** “x **Ycor** **pu lt** 90 **fd** Radius :s **rt** 90 **pd** ***Circle*** :s **pu SetXY** :x :x **pd fill** **end**	**to** ***Radius*** :s **op** (180*:s)/3.14 **end**

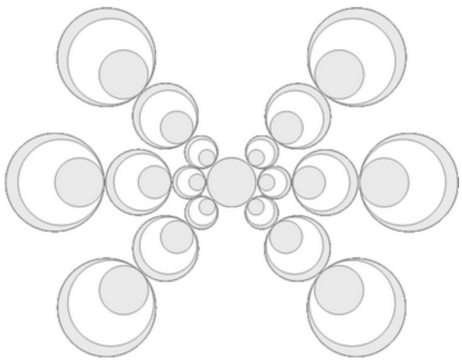
To move to the next circle so as to generate  it is not very natural to go along half of the circle instead of the diameter. Thus the next version of a circle procedure would be with the radius as a parameter.

	**to** ***Element*** :s **Let** "LR (**list** :s 1.75*:s 2*:s) ***Embeded*** :LR ***Mowed*** :R ; **end**	**to** ***Embeded*** :LR **if empty**? :LR [stop] ***Circle*** **first** :LR ***Embeded*** **bf** :LR **end**	**to** ***Mowed*** :R **Let** "x **Xcor Let** "y **Ycor rt** 90 **pu fd** :R **pd fill** **pu fd** :R * 2.75 **fill** **SetXY** :x :y **pd** **end**
	The procedure for this element is as follows: **to** ***Ray*** :s **Let** "b 1 Radius :s **repeat** 3 [**Let** “v :s*:b ***Element*** :v **pu fd** ***Jump*** :v 4 **pd lt** 90 **Let** "b :b+1] **pu** **end**		**to** ***Jump*** :s :b **Let** “r ***Radius*** :s **op** :r*:b+(:R/8) **end**

Then we reflect it:

	The procedure for this element is as follows: **to** ***paddle*** :s **Let** "x **Xcor Let** "y **Ycor rt** 45 **repeat** 2 [**pu fd** ***Jump*** :s 3 **pd lt** 90 ***Ray*** :s **SetXY** :x :y **lt** 90] **end**

Finally we rotate3 times it in 45 degrees to get the following:

	The procedure for this element is as follows:	
	to ***Crop_circles*** :size ***Color.seting*** ***Circle.fill*** :size*3 ***Star*** :size 3 **end**	**to** ***Star*** :size :counter **repeat** :counter [***paddle*** :size] **end** **to** ***Color.seting*** **setpc** [95 230 164] **setbg** [0 140 0] **end**

Various mathematical ideas could be explored in this context – there is more than one definition of the same notion, the computer circle is just an approximation of the mathematical notion of *circle* but could be close enough for practical goals. The final composition could be achieved by combining several geometric transformations and there is a variety of ways of doing this.

Similarly, we modeled the rest of the crop-circle configurations (See the Appendix for the code).

The final presentation

After finishing the Logo code, the next step in the project was to create a professional presentation. For the purpose we used the *Prezi* presentation tool (http://prezi.com/6qjnuyxuubkg/) in which it was easy to present dynamically the original pictures of the crop circles together with a documentary, their computer models, and the corresponding Logo code.

What was learned?

The project is very rich both from mathematics and informatics point of view. The following questions arose during its development: What programming style to choose – bottom-up, top down, or a combination of both? How many procedures and parameters to use, e.g. is it natural to represent the circle as a partial case of an arc and make a procedure for arc only? To use or not to use recursion? And if we decide to use recursion how to determine the best possible way to do this? To use an inbuilt procedure for a polygon filled with color? The next questions are related to when to stop – shall we just create a good enough approximation of the model on the picture, or rather create a whole class of similar figures (with an arbitrary number of circles, branches of circles, other polygons, etc.) The more sophisticated informatics tools the students learn the more elegant and simple their procedures will be.

Ideas for further development

As far as the future development of the project is concerned it could embrace modeling of more crop-circle configurations that have been documented, or extend and generalize the existing models in specific manners. In any case, the students would feel as real researchers who try to understand the nature of this phenomenon, to express their opinion on the existing hypotheses based on their personal modeling experience.

The most important for the children is to use research approaches in a logical order so a to get a final product that could be presented and shared – something crucial for the ideas of the constructionism (Papert, 1999)

Acknowledgments

We would like to express our gratitude to Jenny Sendova for sharing her love for Logo as a language, educational philosophy and culture with us.

References

Blaho, A., Kalas, I. (1998) *Super Logo: Learning by Developing,* Longman Logotron, Cambridge.

Papert, S. *Introduction: What is Logo? And Who Needs It?*, http://www.microworlds.com/support/logo-philosophy-papert.html. (February, 28, 2010)

Sendova, E., Ivanov, I, Nikolov, R. (2002) *FORWARD to the computer world*, Virteh OOD.

Stefanova E., Sendova E., v. Diepen N., Forcheri P., Dodero G., Miranowicz M., Brut M., et al (2007) *Innovative Teacher - Methodological Handbook on ICT-enhanced skills*, Faleza-Office 2000, Sofia.

Appendix

```
to Color_Seting :w
 setbg [0 138 0]
 setfc [95 230 164]
 setpc [95 230 164]
 setpw :w
end

to Circle.fill :s
 let "x xcor let "y ycor
 pu lt 90 fd ( Radius :s ) rt 90 pd
 Circle :s pu setxy :x :y pd fill
end

to Radius :s
 op ( 180 * :s ) / 3.14
end

to Circle :s
 repeat 360 [fd :s rt 1]
end
```

```
to StartUp
 cs Main .5
end

to Jump :r
 pu lt 90 fd :R rt 90 pd
end
```

```
to Main :s
 Color_Seting
 Circle.Draw :s 1 Circle.Draw :s * 2 0
 let "r Radius :s let "a 2 * :R * sqrt ( 3 )
 Kvadrats :a :s
 let "r sqrt ( ( :a * :a ) + ( 4 * :R * :R ) + ( 2 * :a * :R ) )
 let "s ( :R * 3.14 ) / 180
 Circle.Draw :s 0
end

to Circle.Draw :s :c
 let "x xcor let "y ycor
 Jump Radius :s ; pu lt 90 fd rt 90 pd
 Circle :s pu setxy :x :y pd
 if :c = 1 [fill]
end

to Kvadrat :a
 let "LR :a * sqrt ( 3 )
 repeat 2 [rt 90 fd :a]
 rt 60 fd :LR bk :LR rt 30
 repeat 2 [fd :a rt 90]
 lt 90
end

to Kvadrats :a :s
 let "x xcor let "y ycor
 let "r 2 * Radius :s
 repeat 3 [rt 45 pu fd :R pd lt 30 ~
           Kvadrat :a bk :a ~
           pu setxy :x :y rt 105]
end
```

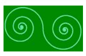
```
to StartUp
 cs Main 1
end

to Main :s
 Color_Seting let "r Radius :s
 pu rt 90 bk :R * 1.5 pd
 repeat 3 [CircleE :s 300 ~
  rt 180 pu fd :R * 3 pd rt 180]
end

to CircleE :s :l
 let "r Radius :s
 fd :R rt 90 repeat :L [fd :s rt 1]
 rt 90 fd :R pu fd :R / 2 pd fill bk :R / 2
end

to StartUp
 Main 3
end

to Main :s
 cs Color_Seting
 Arcs6 :s
end

to Color_Seting
 setbg [0 138 0]
 setfc [95 230 164]
 setpc [95 230 164]
end
```

```
to Arcs6 :s
 let "x xcor let "y ycor
 repeat 6 [Arc :s 3 Arc :s 10 pu setxy :x :y rt 60 pd]
end

to Arc :s :p
 setpw :P
 repeat 120 [fd :s rt 1]
end
```

```
За StartUp
 Main 2
Край
```

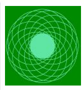
```
За Circles4 :s
 Circle.Draw :s 1
 pu fd 2 * Radius :s pd rt 90
 repeat 4 [Circle.Draw :s / 2 1 Jump :s ~
  Arc :s * 2]
Край

За Radius :s
 op ( 180 * :s ) / 3.14
Край

За Main :s
 cs Color_Seting
 Circles4 :s
Край

За Circle.Draw :s :c
 let "x xcor let "y ycor
 pu lt 90 fd Radius :s rt 90 pd
 Circle :s pu setxy :x :y pd
 if :c = 1 [fill]
Край

За Arc :S
 repeat 90 [fd :s rt 1]
Край

За Color_Seting
 setbg [0 138 0]
 setfc [95 230 164]
 setpc [95 230 164]
 setpw 5
Край

За Circle :s
 repeat 360 [fd :s rt 1]
Край

За Jump :s
 pu fd :s * 2 fill bk :s * 2 pd
Край
```

```
За StartUp
 cs Main .1
Край

За Cir :R
 Circle :R 1 1
 let "R1 :R
 repeat 14 [Spiral :R1 -1 let "R1 :R1 + repc + .618]
 let "R2 :R1 - ( 14 + .618 )
 repeat 14 [Spiral :R2 1 let "R2 :R2 - ( ( 14 - repc ) + .618 )]
 Circle :R2 -1 1
Край

За Main :s
 Color_Seting
 let "r Radius :s
 Cir :R
Край

За Circle :Rad :G :F
 repeat 360 [fd ( :rad * 3.14 ) / 180 rt :g]
 pu if :g < 0 [rt :g - 90][rt :g + 90]
 fd :rad / 2 if :F = 1 [pd fill]
 bk :rad / 2 pd
 if :g < 0 [rt :g - 90][lt :g + 90]
Край

За Radius :s
 op ( 180 * :s ) / 3.14
Край

За Color_Seting
 setbg [0 138 0]
 setfc [95 230 164]
 setpc [95 230 164]
 setpw 5
Край
```



```
За Spiral :R :g
 repeat 90 [fd ( :R * 3.14 ) / 180 lt :g]
Край
```

```
to StartUp
 cs Main 1
end

to Main :s
 Color_Seting
 Circle.fill :s
 Circles12 :s
end

to Color_Seting
 setbg [0 138 0]
 setfc [95 230 164]
 setpc [95 230 164]
 setpw 2
end

to Radius :s
 op ( :s * 180 ) / 3.14
end
```



```
to Circle :s
 repeat 360 [fd :s rt 1]
end

to Circle.fill :s
 let "x xcor let "y ycor
 pu fd ( Radius :s ) rt 90 pd
 Circle :s pu setxy :x :y pd lt 90 fill
end

to Jump :x :y :r
 pu setxy :x :y lt 60 fd :R rt 90 pd
end

to Circles12 :s
 let "x xcor let "y xcor
 let "r Radius :s
 repeat 12 [Jump :x :y :R Circle :s * 2]
end
```

Constructionism in the Era of One-to-One Computing: A Case Study from Thailand

Arnan Sipitakiat, *arnans@cm.edu*
Chiang Mai University, 239 Huaykaew Rd., Muang, Chiang Mai, Thailand.

Paron Issarasena, *paroni@cscoms.com*
Suksaphattana Foundation, 126 Prachautit Rd., Bangmod, Tungkru, Bangkok,Thailand.

Abstract

The purpose of this study is to investigate how new learning possibilities could take place in environments where students are saturated with laptops. Specifically, this paper draws upon in situ examples collected from three schools over a one year period to highlight the integral role of the technology, the people, and the institutions involved. This work has identified a significant difference in the nature of the learning activities observed between the schools. Among the three observed sites, one was a rural community-oriented school. The learning activities were primary driven by issues in the community and extended beyond the walls of the classroom (Figure 1 shows an example). The other two sites were urban schools and the learning activities there were governed mainly by the national curriculum. Despite the differences, this work has found that the core driving forces that govern the success or failure of the learning innovation remains significantly similar.

Figure 1. Students taking pictures of plants in a near-by forest using the laptop's built-in camera.

This paper presents a theoretical analysis of the case studies from Papert's comparison of Piaget's assimilation and accommodation process to how schools are embracing or resisting learning innovations. We further discuss the case studies from a conceptual framework where change itself is treated as a learning process. Thus, interpreting what is seen requires one to understand that the process is emergent, situated, and needs to be understood holistically.

Keywords

One-to-one Computing; Case Study; Holistic Interpretation; Thailand

Overview

This paper describes and analyses learning innovations collected from Thailand's first longitudinal study of learning environments with one-to-one laptop. Three schools are described, all of which have adopted the constructionist learning philosophy (Papert, 1980) in their own ways before the one-to-one laptop program. The goal of this paper is to describe case studies that took place and try to better understand the implications at a macro-level using recent, and perhaps less talked about, arguments given by Papert and his colleagues.

Papert has long advocated that schools are about to go through a radical paradigm shift (Papert, 1980). His early predictions about computers being ubiquitous in schools have been realized. Further more, the dream of having laptop computers in the hands of every student is not a distant reality. While the material aspect of Papert's vision has come true, his foresight of a fundamental change in the learning process has yet to take root in any clear way. This issue has been a subject of many studies and debates (See Cuban, 2001; Warschauer, 2006; Collins & Halverson, 2009). Papert has responded by positively suggesting that school's resistance to this inevitable event is similar to the assimilation process described by Piaget. In the end "assimilation eventually gives way to accommodation and in Piaget's view the temporary conservatism plays an essential role in preparing for the change" (Papert, 2002). This work uses this perspective to discuss the characteristics of the case studies.

Papert has also argued that changes in schools are better supported by an evolutionary process as opposed to a deliberate reform design (Papert, 1997). Cavallo calls this process "emergent design" and compares the transformation process to cultural changes such as "better eating in America" and "paradigmatic changes in manufacturing" where small and local changes collectively play a significant role. Cavallo points out that any effort to facilitate change in learning must be situated and take into account the sociological and institutional factors as a whole (Cavallo, 2004). This perspective is used to further articulate the case studies from a holistic point of view, taking into account the social and school context as well.

The cases described in this work come from only three schools and, thus, does not make any claims about school reform or paradigm shifts in learning. Rather, this work provides micro examples of Papert's macro perspectives. If changes in the educational system is itself a learning process, this work shows examples of one possible stage in this developmental path from schools that (a) computational materials (laptops) are ubiquitous; (b) a constructionist mindset is part of the school culture; (c) the schools have received some flexibility to introduce progressive learning approaches from their superiors.

Scope

This paper discusses three sites that have used the XO-1 laptop designed by One Laptop per Child non-profit association (OLPC). All the schools have used the laptops for at least one year. Although all the case studies were done on the XO-1 machine, the focus of this paper is to present an anthropological analysis of learning opportunities with one-to-one computing and not on the particular benefits or drawbacks of the XO machine offered by OLPC. The intention is to put the spot light on the people and the institutional factors that play a significant role in the fate of one-to-one laptop programs regardless of the hardware choices.

This paper does not aim to cover all the issues related to one-to-one laptop programs. Questions regarding assessment (either test scores or other qualitative measurements), machine repair and maintenance, side-effects from game addiction or inappropriate web content, establishing Internet infrastructure, machine theft, among others will be addressed in the project's forthcoming full report (Sipitakiat, A. et al., 2010).

Background: The Schools

There were three schools involved in this work. The first site, Ban Samkha, is a small rural school while the other two sites, Ban Sankhumpang and Tessaban-4, are large urban schools.

Ban Samkha is a primary school located in a remote part of Lumpang Province of northern Thailand. There are twenty eight students and three teachers. The school has a close relationship with the village and participates regularly in community projects. The Suksaphat Foundation, a non-profit organization promoting constructionist learning in Thailand, has been working with this community since 2001 facilitating projects such as a community retail store, community broadcasting, and water management. Ban Samkha received pre-production XOs in March 2007. The machines were given to every student from grades one to six, including the teachers. XO-1 machines were made available to them in August 2008.

Ban Sankhumpang is located fifteen minutes from Chiang Mai city. It is a primary school with more than one thousand students. The Suksaphat Foundation started their work at the school in 1999 providing support for technologically-rich learning innovations. Six classrooms, one from each grade level, received laptops in August 2008. A total number of three hundred laptops were given.

Tessaban-4 is located in the city of Lumpang province. It is also a large school with more than one thousand students. The school offers both primary and secondary education. The school became connected with the Suksaphat foundation in 2007. Forty seven laptops were given to two classrooms in August 2008.

Data Collection and Analysis

This work follows the development of learning activities from the participating schools over the course of one year starting in August 2008, which was when the schools received the laptops. The data collected for this work was gathered from three main sources: (1) the researcher's observation from site visits; (2) interview sessions with teachers and students; (3) written documents from student journals and teachers' monthly reports. During the study, bi-monthly half-day visits were made to Ban Sankhampang School. The other two schools, which are further away, were visited every two months. Three two-day workshops were held for all the involved teachers and school administrators to reflect upon their progress and difficulties. Information about activities in Ban Samkha that took place prior to this work was collected from interviews and teacher reports.

This work follows a qualitative research methodology (Bogdan & Bilken, 1992). During each school visit, we collected as much information as possible about the learning activities and anything related to the teachers and students. When new data was received either from a site visit or from teacher reports, they would be organized and broken down into groups. We would then synthesize the data to identify any significant themes that may have emerged (Patton, 1990). This approach allowed us to better steer our attention during the next school visit.

All the students that were involved in this research are primary students ages between eight and twelve years of age. The ages of students in the case studies from Ban Samkha are mixed as they do not divide their classes by age. Ages of students in examples from other schools are more specific and will be described in each case.

Case Studies in Community-Oriented Learning Activities

Prior to the laptop program, Ban Samkha School has already been involved with many "village projects" trying to involve their primary school students in thinking about local issues. The following examples show that this connection provided a fertile ground for project ideas when the laptops arrived.

Household Accounting

In the late 1990s, Ban Samkha was suffering from debt problems and villagers who cannot manage their finances were losing their homes. The village responded by creating a local debt-relief fund for those in trouble. There was a condition that those who which to receive help must build up a good financial habit by keeping a log of their income and expenses. Soon after, many villagers, not only those in trouble, were convinced to join this practice and it became a village-wide activity. Given that many villagers were illiterate, their children became helpful in keeping the account book up-to-date (see Figure 2 left). Each month they would sum up all the numbers and reflect upon their spending in comparison to their earnings. The school played its role by assuring that the students update their account book everyday. Teachers used this opportunity to teach students language (correct spelling, appropriate selection of words, etc), mathematics (summing numbers, using fractions, etc), organizational skills (categorizing items, designing the account layout), and discipline.

This activity unsurprisingly led to the idea of using a computer spreadsheet. The idea even predated the laptops. Spreadsheets can assist students with the calculations. Graphs and other visualizations can help parents reflect on their spending. However, it was the laptops that made the idea work. Before the laptops, the teachers tried using Microsoft Excel at the school's computer lab, which consisted of fifteen donated second hand computers. The attempt failed for two reasons. First, students usually update their account book at home where they could obtain financial activity from their parents. Doing the account book in Excel meant more work because they would have to write down the transactions on paper and re-type them into Excel at school. Second, parents would have to come to the school if they wanted to participate. Printing was not a viable solution for the school. Dr. Suchin, the school's pedagogical mentor since 2001, explained that although the computers were not physically far from their homes, there was a mental barrier that deterred the parents' willingness to participate.

Since the students were allowed to take the laptops home, the spreadsheet idea re-immerged. Khru Srinuan, a teacher at the school, described that the parents immediately became interested in the idea. Being able to utilize the technology at home made the activity much more personal and lowered the existing metal barrier (see Figure 2 center). However, there were technical difficulties that arose mainly from the XO platform. It was not possible to find a spreadsheet program that runs on the XO's operating system. Thus, the students had to rely on on-line spreadsheets, such as Google Documents as show in Figure 2 (right). As a result, this activity was limited only to the students who lived within the range of the school's wireless network. The success of the technological aspect of the project was, thus, yet fully realized.

Figure 2. (Left) An account book logged by students. (Center) A parent observing a laptop brought home by her children. (Right) Example of an account book created in Google Docs.

Laptops for Early Warning of Landslides and Flash-Flooding

Ban Samkha village is located at the base of a mountain with a summit of 600 meters. Deforestation in the past decades has increased the risk for landslides and flash-flooding during the rainy season. Such events pose a threat to villagers who often travel into the forest in search for food and other goods. Flash-flooding is especially dangerous for the school as it is located the closest to the mountain.

Responding to this danger, a weather station was installed at the village by the Hydro and Agro Informatics Institute (HAII). The autonomous station, shown in Figure 3 (left), collects information about rainfall, air humidity, temperature, and various other weather related parameters. The intention was to allow villagers to observe the amount of rainfall and determine the risk level. However, there was one problem. The information was not easily accessible. Getting data stored on the station requires one to connect it to a computer and manually download a log file. Alternatively, the weather data was automatically sent back to a server in Bangkok via a cellular network, which can then be accessed through the Internet as shown in Figure 3 (Right). However, both methods were complicated for the villagers to perform on a regular basis.

When the internet-connected laptops arrived, the students took on the role of monitoring the amount of rainfall. The teachers and the villager-head organized a system where students will be given a small credit when they alert the village of dangerous rainfall levels. Since the students normally spend a great deal of time online, this project worked out well.

Volunteers from the Hydro and Agro Informatics Institute have made regular visits as part of a routine check-up and evaluation of their equipment. This allowed the students to learn about the technology and the principles involved from the volunteers. For example, students were earlier informed by the teachers that rainfall greater than 80 mm/day is considered unsafe. But they were not able to understand what this number meant. Meeting with the HAII volunteers allowed them to learn directly from the experts. The students became connected and interested in a set of scientific ideas and there were opportunities for them to fulfill their curiosity.

Figure 3. (Left) The weather station installed at the Ban Samkha village. (Right) The rainfall information is uploaded to a website accessible by students.

The Laptop Band

This activity involved students forming a band mixing the laptops with traditional Thai musical instruments. The idea emerged after a group of students became fascinated using the laptops to create music. They have discovered a way, through a program called Tam Tam, to turn the keyboard into a musical instrument. A parent who is a vocalist from a traditional Thai band saw what the students were doing and engaged them to sing along. The group then came up with an idea that the laptops can serve as instruments in the village's band. Since the computer software could mimic many instruments, each laptop could play different sounds. This band (see Figure 4) became popular and they performed at many shows including a few in Bangkok.

From a learning perspective, the true value of this activity was revealed in the process of putting the band together. While tuning the local instruments to the laptops, the band discovered that the western musical interval is different from that of the traditional Thai instruments. That is, depending on the type of music being played, the eight notes in an octave are divided differently (Miller T., Williams, S., 2008). This mismatch was initially frustrating and it was not possible to decide which side is "out of tune". Both sounded "right" but they could not mix in harmony.

When the teachers discovered the cause and discussed it with the students, many students wondered how they could tune the sound on the laptop. This meant shifting the frequency of each key up or down. Although this process turned out to be too technical, the concept of sound frequency was widely discussed among the students and teachers. Although there was not an explicit attempt to teach students about the science involved nor was there any kind of assessment to prove what the students may have learned, there was no doubt that the students were highly engaged in the topic and collectively discovered something new. In the end, the band was able to identify a selection of Thai songs that were acceptable for the mixed band. Also, since the Thai instruments were tuned by ear, they were able to make some adjustments to compensate for the differences as well.

Because the laptop band became rather popular, the two other schools in this pilot program took on the idea and created their own version of the band. However, these later bands were different in two significant ways. First, original band emerged because of the existence of the traditional band. The later two were introduced mainly as a class activity. Thus, the original band in Ban Samkha was more authentic. Second, the songs selection of the later bands was mostly the same as those used at Ban Samkha. Thus, the miss match between the Thai and Western music intervals did not came up as an important issue for them.

Our intention of comparing the original and the later version of the laptop music activity is not to diminish the value of the later cases. The activity was adapted from its original context of a small rural school to an urban setting with a much larger class size. The activity was not a failure. In fact, feedback from students and teachers were highly positive. But the quality and authenticity of the activity were different from Ban Samkha. The differences observed of the same activity in different schools show how learning is tightly coupled to the local context at which it has taken place. Such learning activities do not transfer easily!

Figure 4. The Laptop Band integrated with traditional instruments.

Case Studies in Curriculum-Driven Learning Activities

Unlike community-oriented projects, the situation was drastically different when the learning environment was driven by the curriculum. The following shows examples of what took place when the learning activity becomes more school-like.

Modelling and Programming

Two months after the laptops were distributed to schools, a Thai-language programming platform was made available. The arrival of Scratch, a graphical programming environment for children (Resnick, M. et al., 2009), with a Thai interface created a significant difference in the kinds of projects student did with the laptops. Programming became a popular activity; being rated in the top three of the most used applications as reported by the teachers. Projects with programming elements allowed for more interaction and better integration of rich media than those created by specialized programs like paint or a word processor. Scratch projects incorporate drawings, animations, pictures, and sound all of which can be given dynamic behaviors. Consider the following case study.

Khru Tukta, a sixth grade teacher, has been using maps to teach students, ages tweleve, about geometry and other mathematical ideas for many years. One particular exercise is to ask student to draw a map showing the route from the school to their home. When this activity was first moved from paper-and-pencil (Figure 5 left) to the laptops, Khru Tukta appreciated the increase in students' engagement and participation. However, since the students initially used a paint program to draw the maps, the output was more or less the same as that done on paper (Figure 5 center). Thus, there were no significant intellectual benefits from the transition to the digital medium.

When students started to create projects with Scratch, however, she realized that her activity can now expand to cover a much larger range of activities and subjects. Students can record their voice to narrate the trip back home and add an animated character that moves along the path (Figure 5 right). Pictures taken from the laptop's built-in camera can be programmed to pop up showing important landmarks along the way. Such dynamic behaviors were not possible with paper-and-pencil. Thus, programming clearly enriched the map activity with more possibilities and creativity.

Figure 5. (Left) Maps drawn on paper. (Center) a digital map on the computer. (Right) Programming allowed students to enrich their maps with animated characters and story plots.

Rich Media and Storytelling

Storytelling is another popular learning activity observed in the pilot schools. It is common to see photo assays about family, local historic sites, weddings and other seasonal festivals. The activity fits well with what the teachers has already been doing, given that the Thai curriculum clearly dictates the development of language and communication skills. The teachers can also use the content of the story to assess the students understanding of a particular topic in the curriculum. In this case, the digital medium fits in nicely for the teacher by allowing an integration of rich media such as text, pictures, animations, videos, and sound recordings. Many teachers also find this activity more relaxing for them, especially for those who are less computer-savvy. With other kinds of activities, such as programming, there is a risk of the teacher being stuck and not being able to help students finish their work. But storytelling can go on even if all goes wrong with the technology.

Some teachers are able to go further and utilize the new possibilities to develop novel learning activities. For example, because students enjoy taking pictures with the laptop's built-in webcam and showing them to friends, Khru Srinuan in Ban Samkha School organized a photography fieldtrip along the nearby mountain. The assignment was for the students to take pictures of plants or flowers that they do not recognize in the forest as shown in Figure 6 (Left). They would then show the pictures to their friends and try to figure out what the plants were and write descriptions for others to later see. Other observed examples include video assays showing indigenous medicine using captured video interviews, stage acting using the computer screen for props and making sound effects (Figure 6 center and right), and making animated electronic cards for teachers and friends during the New Year's celebration.

Figure 6. (Left) Photography with laptops. (Center and right) using laptops as props and a sound source for plays.

Information Inquiry

It should not be a surprise that using Internet-connected laptops to search for information is an activity used by all the teachers. It fits well as an extension to the existing practice of the classroom. However, it requires a great deal of creativity to prevent this use from becoming over-taken by the traditional school routine. There have been complaints from some teachers saying that searching the web takes too much time and is a distraction from what the students need to learn. Consider the following case study.

A teacher requested the class to describe the internal organs of a typical fish. Instead of just looking this up in a textbook, the teacher asked students to search the Internet as well. Since the activity as assigned without the students' participation and it did not leave space for much creativity, students ended up with similar pictures and descriptions mostly from the same websites. In this case, searching the Internet meant extra work and made it difficult to complete lessons as planned. Difficulties connecting the laptops to the internet and other technical issues made it even worse. In this situation, teachers perceive the laptops as a distraction.

On the other hand, the response is positive for teachers who engaged students in creative open-ended projects and gave students more freedom to express themselves. For instance, Khru Jiraporn, a fourth grade teacher, engaged her students in finding out the nutrients contained in the food the students like to eat in a day. Then they would try to create their own healthy diet. Jiraporn passionately described how her students were able to learn much more about the given topic than that described in the school curriculum. "With the Internet, the scope of what the students could learn is equal to adults". The diversity among groups led them to discover different aspects of the topic and the information spreads quickly among the students when something "new" was found. "Students are enthusiastic doing their work on the laptop and you can really see their desire to produce good work".

Khru Jiraporn uses curriculum mapping to track students' progress, which alleviates her from the traditional rote teaching method. She admits that tracking students individually can take a lot of effort but the result is rewarding.

Outside of the classroom, accessing the Internet is a major attraction to the students. There are reports from all the participating schools of students who would "load" their laptops with web content before going home. The web pages could contain information related to homework. Or they could contain non-school related content. For example, students at Ban Samkha would load soap opera manuscripts to read at home. These manuscripts are common in Thai newspapers and give out the plot of episodes yet to be aired on television. Some students in Ban Samkha even come to the school at night to connect to the school's network.

Reflection:

This work has showed empirical examples from a longitudinal study of how learning with one-to-one laptop can take place and how they could differ significantly depending on the purpose of the learning activity. The following section attempts to synthesize the important foundations that allowed the activities to take place.

Innovation as a Developmental Process

From our experience working with the two urban schools, it was clear that the schools' primary responsibility is to deliver the content defined in the national curriculum. Any intervention or innovation must first address how this ultimate responsibility can still be fulfilled. Curriculum mapping has been the most commonly used technique to deal with this issue. Teachers would evaluate students' projects and map what was learned to items in the curriculum. This method satisfied the school system while giving room for project-based activities. It is important to note that teachers do not see this as a compromise. Instead, they usually promote this practice as a standard procedure. Although Papert probably would not reject subject guidance altogether, but following a single strict curriculum would severely contradicts with his notion of epistemological pluralism (Papert, 1992).

We believe this situation is an example of how assimilation is taking place in the schools. Curriculum mapping is a good example of how the new is assimilated into the old without requiring a major change to the system's foundation. This is a kind of assimilation less severe than many other cases where the innovation is transformed entirely to keep every aspect of the traditional schooling the same. The two schools have changed in many ways (i.e. long-term projects are possible), but some key aspects of traditional schooling are still kept the same. Thus, from a Papert's perspective, these schools are well in their developmental stages. It is essential, though, that they continue to evolve. Otherwise there is a danger of becoming too comfortable with the current practices that they may become stuck in an artificial stable state. For example, even though curriculum-mapping allows for project-based activities, project ideas often stem from the need to cover a certain topic in the curriculum more than the students' interests. Therefore, all parties involved must work together to keep pushing things forward in a positive direction.

The observations of community-oriented activities show a drastically different situation. In Ban Samkha, since the activities were mostly done outside of school and were integrated into real issues of the community, their origins were not driven by the curriculum. Learning that took place in the case studies were authentic and meaningful to the students and the laptops played an essential role. As a result, these projects have made Ban Samkha a model school and they have been well publicized in recent years. Ironically, there has recently been more emphasis on schooling in the traditional sense at Ban Samkha. Teachers have expressed that their students do not perform well enough in the national tests. They feel that the school should do well both in terms of community projects and test scores. In a way, becoming a model school has held them back due to the fear of not fulfilling the expectations of the current educational system. It is a situation where the existing schooling system can oppress an innovation even after the innovation has become successful. From a Piagetian's perspective, Ban Samkha is, too, in a developmental stage. Therefore, the most important thing for the school is not to "get it right", but to continue to evolve and move forward in their thinking.

Understanding Progress: A Holistic Perspective

The developmental process discussed above is not governed by a single entity. The school teacher, the school institution, the technology alone cannot be held responsible in isolation. Laptops will not make a difference unless there is proper facilitation from teachers. But teachers will lose their jobs if they do not fulfil their requirements no matter how convinced they are about Constructionism. School administrators may want to give teachers more flexibility but they have to stay within the limits given by the school's own superiors. We will always reach a deadlock unless we look at the system holistically. This work has presented a snapshot of schools in their developmental process. Their characteristics are defined by the following factors.

1. The Technology. One-to-One laptops have clearly created new possibilities for learning. This work has shown in situ examples of how learning activities can move beyond the limitations of paper-and-pencil. There is not question that a learning environment where every child has an internet-connected laptop brings about new ways of reaching content, communicating with peers, and expression through interactive rich media.

2. The People. Teachers hold the ultimate authority in the classroom. Developing learning activities like those shown in this paper requires the willingness and leadership of teachers. This work has found that teachers are capable of adapting to make the most of the new learning opportunities. However, teachers also have many questions and doubts. Teacher development takes more than teacher training. They need, among others, examples and guidelines, opportunities to experiment, and places to reflect upon the lessons learned. The process is developmental and happens over a long time period.

Students also create an impact on the learning process as well. Students' enthusiasm with technology and the increased interest in learning has inspired many teachers. For example, students' overwhelming interest in photography and programming have influenced the kinds of projects that teachers later chose to conduct. This force resembles the kind of irresistible outpour from learners that Papert calls "Child Power" (Papert, 1993).

3. The Institution. The School's support for flexible class organization and student assessment was essential for the progression of this work. If there was no support from the school managers, expecting teachers to innovate while keeping everything else constant would be unrealistic. Since schools are assessed by the municipal office, which is in turn governed by the Ministry of Education, schools do not always have the freedom to take action the way they want. In this work, the schools were able to justify their actions based on the fact that this is a pilot project working with a relatively small number of classrooms. Ban Samkha is a model school and, thus, now have the privilege to be unique. The situation would be different, and possibly much harder, for larger implementations of one-to-one computing.

Summary

Does this work with one-to-one computing illustrate a kind of end-point that Papert has envisioned about the future of learning? Not even close. However, we believe that the case studies provide a valuable example of the process of getting there. We have shown concrete examples of how one-to-one laptops can lead to novel learning activities both in a school and a community setting. Although these activities are still influenced by the existing school paradigm, we have presented a holistic view of the progresses that have taken place.

References

Cavallo, D. (2004). Models of growth: towards fundamental change in learning environments, *BT Technology Journal*, 22 (4), 96-112.

Collins, A., & Halverson, R. (2009). Rethinking *education in the age of technology.* New York: Teachers College Press.

Cuban, L. (2001). *Oversold and underused: Computers in the classroom*. Cambridge, MA: Harvard University Press.

Bogdan, R. C., & Bilken, S. K. (1992). *Qualitative research for education: an introduction to theory and methods*. Boston, MA:Allyn and Bacon.

Miller E. T., & Williams, S. (2008). *The Garland handbook of Southeast Asian music*. Routledge.

Papert, S. (1980). *Mindstorms: Chrildren, computers, and powerful ideas*. New York: Basic Books.

Papert, S. (1993). *The Children's machine: Rethinking school in the age of the computer*. New York: Basic Books.

Papert, S. (1997). Why School Reform Is Impossible, The *Journal of the Learning Sciences*, 6(4), 417-427.

Papert, S. (2002). The turtle's long slow trip: Macro-educological perspectives on microworlds. *Journal of Educational Computing Research*, 27, 7-27.

Patton, M.Q. (1990). *Qualitative evaluation and research methods*. Newbury Park, CA: Sage.

Resnick, M., Maloney, J., Hernandez, A. M., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y. (2009). Scratch: Programming for All. *Communications of the ACM*, 52(11), 60-67.

Sipitakiat, A., Junsuphachaisit, V., Tutiyapungprasert, N. (2010). *Learning with one-to-one laptops*. Bangkok, Thailand. Office of the Education Council. (Forthcoming)

Tyack, D., & Cuban, L. (1995). *Tinkering toward utopia: A century of public school reform*. Cambridge, MA: Harvard University Press.

UNESCO (2005). *Ban Samkha Community that learns*. Bangkok, Thailand: Ministry of Science and Technology.

Warschauer, M. (2006). *Laptops and Literacy: Learning in the Wireless Classroom*. New York: Teachers College Press.

Acknowledgments

We gratefully acknowledge the support from Suksaphat Foundation and Thailand's Office of the Education Council for organizing the pilot program and providing the necessary funding for the completion of this research.

Implementing the Dynamic Geometry Approach

Zhonghong Jiang, *zj10@txstate.edu*
Dept of Mathematics, Texas State University, San Marcos

Short presentation

This is a four-year research project funded by the National Science Foundation. The project compares effects of an approach to high school geometry that utilizes Dynamic Geometry (DG) software with standard instruction that does not make use of DG exploration tools. The basic hypothesis of the study is that use of DG software to engage students in constructing mathematical ideas through experimentation, observation, data recording, conjecturing, conjecture testing, and proof results in better geometry learning for most students.

The theoretical foundation of the DG approach and the theoretical framework of this project consist of the constructivist perspective and the van Hiele model. Building upon previous studies, this study will seek to answer the following research questions: 1) How do students in the experimental condition perform in comparison with students in the control condition on measures of geometry standardized tests and a geometry conjecturing-proving test? 2) How does the DG intervention affect student beliefs about the nature of geometry and about the nature of mathematics in general? 3) How does the DG intervention contribute to narrowing the achievement gap between students receiving free or reduced price lunch and other students? The research study follows a mixed methods, multi-site randomized cluster design. The population from which the participants of this efficacy trial are sampled are the 10th grade geometry teachers and their students at all high schools in Central Texas at which 50% or more of the students are eligible for free or reduced lunch. For determining the sample size, a power analysis has been conducted. Taking an attrition rate (20%) into consideration, 76 teachers are randomly selected from that population for the study.

The 76 teachers are then randomly assigned to two groups. Each teacher is represented in the study with measurements from only one classroom of students, and the classroom and teacher unit of analysis will overlap, yielding the design where the students are nested within teachers/classrooms, which are nested within schools. Teachers in both treatment and control groups receive relevant professional development. Fidelity of implementation for the experimental treatment is monitored carefully.

The study tests the basic hypothesis by assessing student learning using the tests indicated in the research questions. Data for answering the research questions are analyzed by appropriate HLM and qualitative methods. Results will provide strong evidence that can inform school decisions about innovation in that core high school mathematics course.

The implementation plan for the project is: Year 1: Preparation (All research instruments, DG instructional materials, recruitment and training of participants, etc.); Year 2: The first implementation of DG treatment, and related data collection and initial data analysis; Year 3: The second implementation of the DG treatment, and related data collection and continued data analysis; Year 4: Careful and detailed data analysis and reporting.

Keywords

dynamic geometry, experimentation, observation, conjecturing, proving, random assignment, fidelity of implementation

Eight years of journey with Logo leading to the Eiffel tower mathematical project

Jesús Jiménez-Molotla, *jj_molotla@hotmail.com*
Esc. Secundarias Diurnas #229 “L. Yivkova” T. Mat. & #139 “J. E. Rodo” T. Vesp., DF, Mexico

Ana Isabel Sacristán, *asacrist@cinvestav.mx*
Dept. of Mathematics Education, Cinvestav, Mexico

Abstract

Eight years ago, our school in Mexico was introduced to Logo as part of the government-sponsored Teaching Mathematics with Technology (EMAT) program. Since then we have increasingly become interested in developing long-term, interesting, constructionist (Harel & Papert, 1991) projects for our students, particularly in the last 5 years. The idea is that through these projects students become engaged and motivated, while they learn – in a fun and meaningful way – many mathematical topics in the official syllabus, but also have early access to other “powerful ideas” (Papert, 1980) such as “advanced” mathematical concepts that are usually not even considered for children of the age-groups we work with (12-14 years-old), as is the case of trigonometry. At Eurologo 2007, we reported our first long-term “Painless trigonometry” project (Jiménez-Molotla et al., 2007). Our latest project, the “Paris project”, inspired by the hosting city for Constructionism 2010, evolved from that previous trigonometry project, and had as aim the construction of the Eiffel tower (Fig. 1). Such construction (Fig. 2), done in 3D MSWLogo, has required an understanding and use of trigonometric ideas, such as the Pythagorean theorem, and mathematical analyses using various tools (including Google Sketchup and Scratch) of geometrical objects such as pyramids and prisms, which form the building blocks of the Eiffel tower representation. As in our previous projects, the children have been highly motivated and engaged, and their creativity and genius has been awakened.

Figure 1. The aim of our “Paris project”: the Eiffel tower

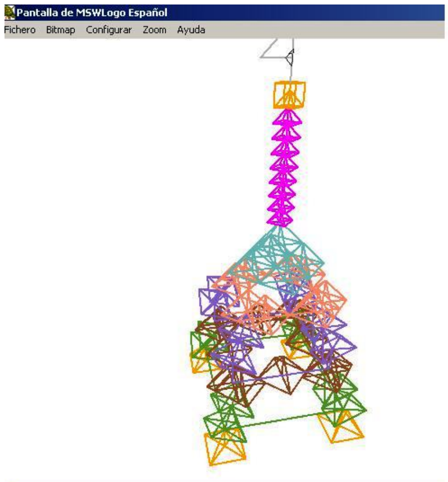


Figure 2. A student’s representation of the Eiffel tower in 3D Logo

Keywords (style: Keywords)

Pythagorean theorem; pyramids; 3D geometry; school project; Logo; Google SketchUp

Background: The evolution of the trigonometry projects

In 2001-2002, our junior secondary schools (children aged 12 to 15 yrs-old) in Mexico were introduced to the Teaching Mathematics with Technology (EMAT) government program, which promotes a constructivist use of open tools (where the user can be in control and have power of deciding how to use the software) such as Spreadsheets (Excel), Dynamic Geometry (Cabri-Géomètre), and Logo (MSWLogo). As we became more proficient in the use of the tools, particularly of Logo, we developed our own activities and projects. Thus, in the past 5 years, we have been working in developing interesting constructionist long-term mathematical projects for our students with an integral use of technological tools like EMAT's Logo, Cabri and Excel, but also with other creative and expressive software. These projects give students an avenue for learning many mathematical topics in the official syllabus, in a fun and meaningful way, while they also give them early access to "powerful ideas" and "advanced" mathematical concepts, such as trigonometry, which is usually not even considered for our Grade 1 and 2 children (12-14 years-old).

Trigonometry is a topic that is traditionally difficult to teach and learn. But computational tools such as Logo, allow early access to this important mathematical area through fun constructions. On the one hand, through these projects, young students can become familiarized with this topic, so that by the time they get to Grade 3 and have to formally learn some trigonometry, they will have experiences and useful intuitive ideas (diSessa, 2000) to build upon. On the other hand, through trigonometry, we can cover other mathematical topics in the curriculum such as: addition, subtraction, multiplication and division; powers and square root of whole and rational numbers; algebra (including constants, variables and polynomials); and geometry.

At Eurologo 2007, we reported on our first venture in this area (Jiménez-Molotla et al., 2007). We described how, in the academic year 2005-06, we took up the challenge to create a technology-based approach for the learning of trigonometry: the "Painless Trigonometry" long-term school project. In that project, we introduced our young students (12-14 yrs-old) to the Pythagorean theorem, basic trigonometry concepts and functions, and their applications using explorations and constructive activities with Cabri, Excel and Logo. This was a project that we carried in all our groups in grades 1 and 2 in two schools (approximately 250 students) for two academic years 2005-2006 and 2006-2007 (different students each year). Students thoroughly enjoyed the activities and gained interest in mathematics. They also developed problem-solving and collaborative skills. Furthermore, in written tests after the project, the students showed an understanding of the "advanced" trigonometry concepts, as well as of other algebraic ideas.

More recently, in Jiménez-Molotla et al. (2009), we reported how the painless trigonometry project had evolved in the academic year 2007-2008, into a project for constructing 3D pyramids. The new project was triggered by a question, from a student of the 1st grade (a group that was being introduced to Logo): "Is it possible to work in four dimensions in Logo?" This gave rise to the idea of a new school project for working in three dimensions, that was named for fun: "In search of the fourth dimension, while in three". We found a curricular topic for junior secondary grades 1 and 2 that could be worked in three dimensions, and that also gave access to the non-curricular theme of trigonometry: the pyramid. We started playing with paper-and-pencil in a geometry game to draw triangles and squares and whatever else was needed for a pyramid. We then transferred that activity to doing it with dynamic geometry (in Cabri) and used an Excel spreadsheet to help us in computing areas and perimeters. In the end, the children programmed pyramids in Logo, and some of them even achieved animations so the pyramids would rotate.

The new project: The Eiffel Tower

When at Eurologo 2007 in Bratislava we heard that the next conference would be held in Paris, immediately we thought that a theme for a new project could be the Eiffel Tower. But it wasn't until 2009 that the project could take place, since we needed to find ways for our students to have the necessary tools to fulfil what we had in mind. For example, students need enough competency in Logo programming, they need to develop an understanding of basic geometry (angles, triangles, polygons, circles, etc), they need to be able to solve arithmetic and algebraic problems, and, of course, they need to work in three dimensions. But the previous projects, particularly the one involving constructions of pyramids, showed us the path for the fulfilment of this new venture.

In this academic year 2009-2010, we have been working intensely for some 5 months on the project (as in previous years, we have worked with approximately 250 students of grades 1 and 2 groups in two schools), having only one 50-min. session per week in the computer room; the results have surpassed our expectations.

We like to think that this project considers the compulsory syllabus that we have to follow in our schools, as we are required to cover certain topics; and it does so, by all the mathematics (as listed above) that it involves. It also tries to use technology, as is recommended in the new official programs (which recommend explicitly the use of the EMAT tools) but it tries to do so in a more creative way than is usually seen in other schools. Our approach is constructionist, since students themselves build the project; we only give ideas or questions, but students themselves pursue them. For example, we may suggest that they build a procedure for constructing a general right triangle with equal catheti; students analyse the problem, share their solutions, and collectively pick correct procedures. This is how we work in general, and "children learn that the teacher too is a learner" (Papert, 1980, p.114). This contrasts with happens in traditional classrooms; in this sense we completely relate to what was also said by Papert (1980, p.115):

> In traditional schoolrooms, teachers do try to work collaboratively with children, but usually the material itself does not spontaneously generate research problems. Can an adult and a child genuinely collaborate … A very important feature of work with computers is that the teacher and the learner can be engaged in a real intellectual collaboration; together they can try to get the computer to do this or that and understand what it actually does. New situations that neither teacher nor learner has seen before come up frequently and so the teacher does not have to pretend not to know. Sharing the problem and the experience of solving it allows a child to learn from an adult not "by doing what teacher says" but "by doing what teacher does."

The development of the "Paris project"

We tackled the Eiffel tower project (which we nicknamed the "Paris project") using as a basis pyramids (which in itself is a challenging project, as we had seen the previous year). The project began with paper-and-pencil work (Fig. 3), using a geometry set, to think and get a clearer idea of how to tackle the design of the project. We also let students reflect on how to construct prisms and pyramids – we used the idea of prisms as building-blocks for building pyramids (we also studied volume formulas on the way). Google SketchUp was a good tool for visualizing the prisms and pyramids (Fig. 4).

Simultaneously, and since we have no resistance in taking advantage of new tools that come along, we started a blog (see Fig. 5) in Wordpress, with which we interact with our students and which their parents can see. In this blog we have been posting the progress of the classroom activities and of our project (with a table similar to the one shown in Table 1), as well as tips and tools that students can use. Through this blog students leave comments and participations, that help and enrich communication in the classroom, as well as with parents and authorities. In this way we are using social networking as educational tools, and the results have been very good.

Table 1 shows the sequence of activities that was followed with Logo (and Scratch) – a very similar table was posted on the blog. We introduced students to Logo and Scratch (as well as Cabri) only in October 2009 and in just a few months their progress was outstanding (those students who had participated in the previous year's Pyramids in 3D project had moved on to higher grades or had new teachers, so all the students we had this year were new to Logo, the other computational tools and the activities). We began by exploring quadrilaterals with paper-and-pencil and geometry sets, with Cabri (Fig. 6), and then with Logo as well as Scratch. This is an easy way to learn Logo and we let our students play with their constructions. For students, things are more meaningful if they can play with them or if they have a challenge. For example, when constructing squares, we challenged them to create a staircase but this soon became something else when they played with their Logo procedures (Fig. 7). Some people have told us to stop children from playing, but the answer is: if it's play to them, let them play.

Figure 3. Students' paper-and-pencil work.

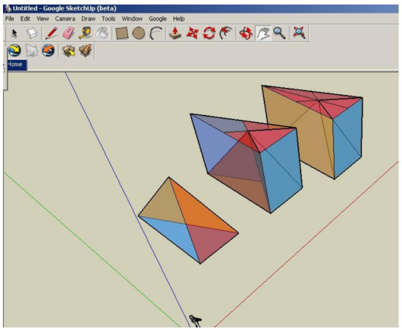


Figure 4. Prisms and pyramid constructions with Google Sketchup, as a visualization aid.

Figure 5. Image of our blog where we posted the "Paris project".

Table 1. Summary of activities in the "Paris Project":

Aim: construction of the Eiffel tower		
Activities	Logo	Scratch
1. Beginning with the construction of squares…	Getting to know Logo (Fig. 6)	Getting to know Scratch
2. … students will construct regular polygons with different number of sides	Regular polygons	Regular polygons
3. Through the construction of an equilateral triangle, students reflect on the inner, outer and supplementary angles of a triangle, and see that the sum of the	Equilateral triangle	Triangle

inner angles of any triangle equals 180°		
4. Squares and triangles…	…for building a house	
5. Squares, triangles and rectangles…	…for building castles	
6. Welcome to the third dimension	Building a cube (Fig. 8)	
7. First encounter with a very important tools: The Pythagorean theorem (with right triangles of 45° angles)	Half a cube: Trace the diagonals across two faces of a cube to split the cube in half.	
8. Construction of a rectangular prism and of a cube	Build a car using a rectangular prism and a cube (Fig. 9)	
9. Prisms and pyramids	Build a house using a rectangular prism and a triangular prism (Fig. 10)	
10. Triangle and segment…	… for building a flag. Challenge: animate the flag so that it rotates on its axis (Fig. 11)	
11. Rectangle, equilateral triangle and segment (tower with flag)	Place a flag on top of a tower (Fig. 12)	
12. Animations	Programming for creating animations of squares and polygons	
13. The Pythagorean theorem, triangles, rectangles and squares	Build the base of a pyramid and its height (Fig. 14)	
14. The Pythagorean theorem and right triangles with equal legs and 45° angles	Finish the pyramid using right triangles with two equal sides and 45° angles	
15. Students build theorems	Theorems are helpful in building different types of triangles.	
16. Building pyramid stacks	Note: Students ran into problems when they tried to build pyramids on top of each other. They realized they needed to begin and end the procedures in the centre of the pyramids.	
17. Building a new pyramid	Students refine their pyramid procedures for using less sub-procedures and so that they are easier to combine	
18. The pillar of the tower	Construction of part of the Tower	
19. Each student decides how his Tower project will be finished, and they share and express their difficulties and progress	Students decide how to finish their project based on their own reflections. They share amongst themselves their progress, and parts of their procedures, to help each other.	

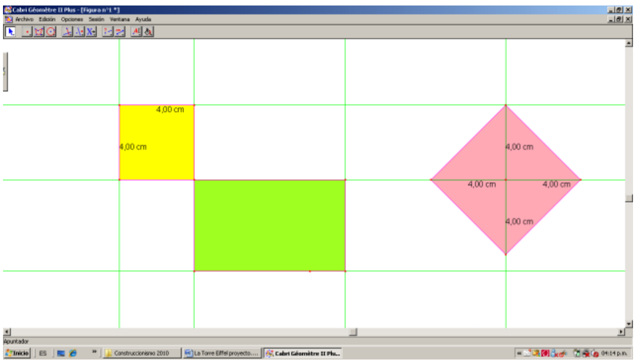


Figure 6. Cabri explorations with quadrilaterals.

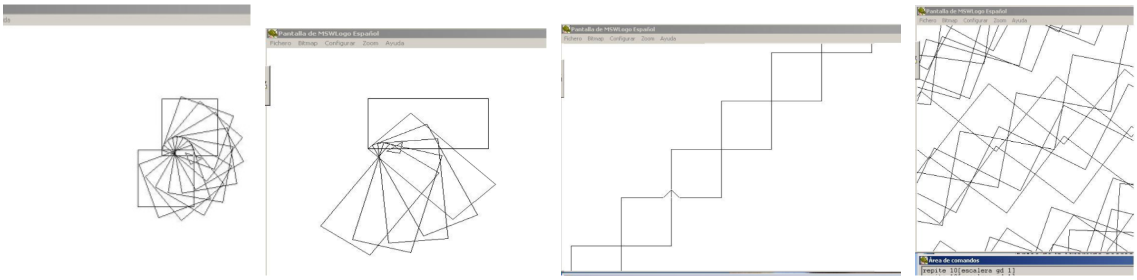


Figure 7. Logo play with quadrilaterals.

We continued with constructions of regular polygons with technology (activity 2 in Table 1) and without it. In the regular classroom we did paper-and-pencil analyses of the hexagon as a figure formed by 6 equilateral triangles; this was a way of introducing the importance of the triangle that will be very important for the project. We then would introduce the Pythagorean theorem and three-dimensional constructions. We began work in the third dimension with simpler exercises than the pyramid, like building cubes and prisms, which were first visualized in Google Sketchup, then constructed in Logo (Fig. 8). Students were introduced to 3D primitives (such as those for roll and pitch) during the activity to build a cube in Logo. In this activity they needed to reflect on how to combine squares and the movements and angles the turtle needs to do. Afterwards they continued building rectangular prisms.

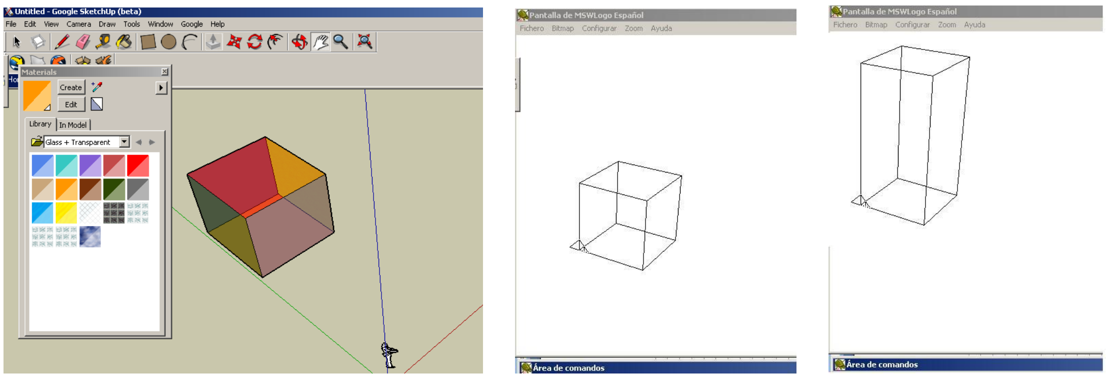
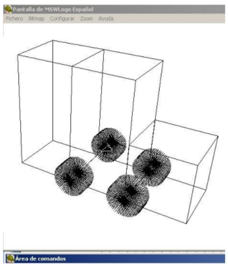


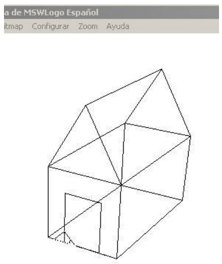
Figure 8. Visualization with Google SketchUp and constructions in Logo of 3D cubes and prisms.

We challenged them to use those previous constructions (cube and prism) to build a car (Fig. 9) and a house (Fig. 10). The results were fabulous and students had fun while they learned and reflected.

```
to prism
repeat 4[square fd 100 downpitch 90 rectangle fd 200 downpitch  90]
end
```



```
to car
prism
downpitch 90
fd 100
downpitch -90
prism
downpitch 90
fd 100
downpitch -90
cube
end
```



```
to house
prism
fd 100
downpitch 90
fd 200
downpitch -90
rt 30
prismtri
fd 200
rightroll 60
downpitch 90
fd 100
rt 90
fd 75
rt 90
door
end
```

```
to prismtri
triangle
downpitch -90
rightroll 30
rectangle
rightroll 60
rectangle
downpitch -90
rt 30
fd 100
downpitch 90
rightroll 30
rectangle
end
```

Figure 9. A 3D Logo car (and procedure) built with cubes and prisms.

Figure 10. A 3D house (and procedure) using prisms.

We then moved on to the construction of a flag, which would be needed to be placed on top of the Eiffel tower representation. A challenge here was to animate the flag by having it rotate around its axis. Children love to create animations. When the flag was placed over a tower, children realized that they couldn't erase the entire screen between "frames" like they had done before, so they used the *penerase* primitive.

When we felt ready to move on to the construction of the pyramid, which would be the basis for the Eiffel tower representation, we used Google SketchUp for visualization and analysis purposes, and to understand how triangles form a pyramid (Fig. 13). In the classroom we also used paper-and-pencil and geometry set activities for understanding further the construction of prisms and pyramids (we also calculated areas and volumes). Students then began the construction of pyramids in Logo, beginning with the base and height (Fig. 14).

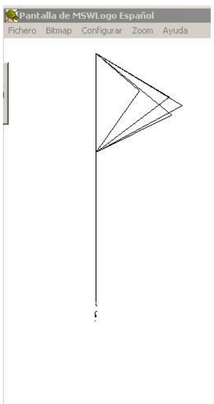


Figure 11. An animated rotating flag in 3D Logo.

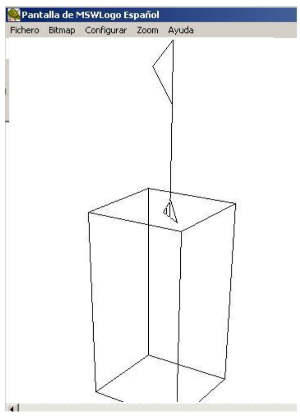
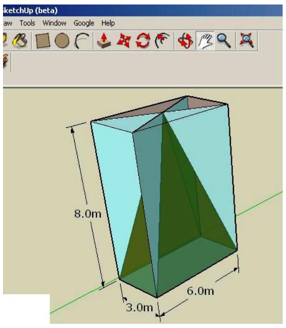
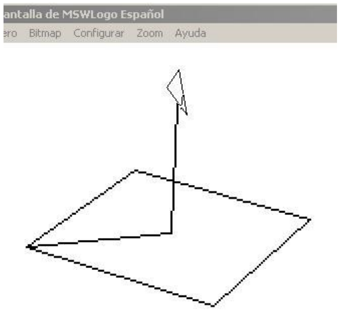


Figure 12. The rotating flag over a tower.





```
to pyramidbase
square
rt 45
fd sqrt (100*100+100*100)/2
downpitch -90
fd sqrt (100*100+100*100)/2
end
```

Figure 13. Visualization of the pyramid in Google SketchUp.

Figure 14. The base and height of the pyramid (with procedure) in Logo.

Finished pyramids would then be stacked to complete the Eiffel tower (see Fig. 2 on first page). A difficulty that was faced here was that for aligning the stacks of pyramids, the procedure for the pyramid needed to be changed so as to begin and end it in the centre of the pyramid.

Each student decided how to finish their own Eiffel tower; some children had real difficulties and challenges but they all shared their progress and those that were ahead shared parts of their procedures and ideas to help others.

Daniela, a 13 year-old student, was the first to finish the Eiffel tower representation (Fig. 2), and she added colours to make it nicer. She confessed to spending some 9 hours of work at home over three days to finish the project, showing the deep motivation that the project created in her. She wrote the following (translated from the original Spanish) on the project's blog:

> *"Logo is a computer software that really impressed me by its vast functions that one could modify by adding one's own created commands. During my experience with Logo, I was deeply impacted in discovering the things that one can do alone with just a little bit of mathematics, interest and Logo. I also discovered another way to learn and develop my spatial and geometrical abilities. The truth is that for my final programming creation, there were many obstacles beforehand, I had difficulties with the 4 pillars [of the tower], and diverse things like colours, but after these "trials" that some consider mistakes, you have another perspective on how Logo functions. For me it was the best experience in mathematics and computer science."*

These are some of Daniela's main parts of her finished Eiffel tower procedure:

```
to pyramid
repeat 4[triangle rightroll 90]
downpitch 90
back theorem
left 45
square
right 45
forward theorem
downpitch -90
forward theorem
end
```

```
to triangle
forward theorem right 135 forward
theorem2
right 135 forward theorem
right 90
end

to theorem
output squareroot (50 * 50 + 50 * 50)
end

to theorem2
output squareroot (theorem * theorem +
       theorem * theorem)
end
```

Some results

As the project progressed, we observed many changes in our students. Their interest in mathematics was awakened, their understanding of regular mathematical school work improved, and they even developed a defensive attitude of their work in the project against outsiders who criticized it.

A student commented:

> *"[It] was an adventure, in which one had to take risks and take different routes to reach the objective: the Eiffel tower, although I had a bit of difficulty with the bearings and I used as a compass, my previous knowledge; as a map, my classmates; and as a guide, my teacher, I explored different paths until finding the right one. A big adventure with Logo".*

There was also another student who was extremely aggressive at the beginning of the school year, and didn't do any work; even his father doesn't know what to do with him. This student became transformed during the computer work. Today he is a good boy in the computer room and is one of the first to finish the Logo activities (and though in the regular classroom he is still a bit undisciplined and doesn't pay attention, his overall work *has* improved). His classmates think he is a mathematical genius. As we have learned in our journey that began eight years ago, Logo helps us discover the genius in both the understood and the non-understood children.

The obstacles

However, as in the past, we have continued to face many criticisms (and obstacles) from some peers, authorities and even parents, on our technology-based projects, because they don't understand what we are doing, despite the fact that the use of the EMAT program (though governmental support for teacher-training in that program has been stopped) is still explicitly recommended in the new mandatory curriculum. The work is not easy, as there is no real support for the use of digital technologies and related projects. An example of this was last year when we asked for support from the school authorities for attending a conference: the answer was no and that we shouldn't send any more papers in the future to conferences because there was no support (and yet we are doing it again).

Another example is when a parent filed a complaint with the school authorities saying that we only played with computers and children were not being taught mathematics (he actually wanted his child to do repetitions after repetitions of operations). He objected to work that was fun; for him mathematics had to be tedious.

> The *kind of mathematics* foisted on children in schools is not meaningful, fun, or even very useful. This does not mean that an individual child cannot turn it into a valuable and enjoyable personal game. For some the game is scoring grades; for others it is outwitting the teacher and the system. … for school math … *despite* its intrinsic dullness, inventive children can find excitement and meaning in it. (Papert, 1980; pp. 61-62).

But mathematics *can* be interesting. Yet, working with technological tools like Logo carry great responsibility; it is not just a matter of doing a little bit and then abandoning it. That's why we like doing long-term projects. But most teachers object to doing this because they feel the curriculum and requirements are already too time-consuming that they leave no time to afford on these projects; they see only the parts and not the whole. We are reminded of Papert's words:

> Conservatism in the world of education has become a self-perpetuating social phenomenon. […] The computer revolution has scarcely begun, but is already breeding its own conservatism. … [the] conservative social system appropriates and tries to neutralize a potentially revolutionary instrument. (Papert, 1980; p.37 and p. 45).

Papert (2006) argues that it makes little sense to use digital tools to serve a curriculum that was created without them, and he challenged us to reserve at least 10% of our time to think of what new mathematics and mathematical practices can emerge from the use of these technologies.

For these reasons we believe even more firmly that it is important to continue with our efforts to break the barriers. We believe that if students learn to use technology in a thoughtful way (moving away from a mechanized use and developing projects they feel as their own) they can become better learners, while new mathematical ideas may emerge, and the teacher can share and collaborate in their discovery processes rather than being the traditional presenter of knowledge. We are a new generation of students and teachers creating experimental environments, computational microworlds and classroom dynamics that are very different from traditional school practices. The catalyst for change are the technological tools, but it is in our power, as teachers, to actually make the change, and use the tools in innovative ways that change mathematical teaching and learning. The tools don't bring the benefits; it's the *use* we make of them that does.

A further final comment on Logo

We would like to add here a comment that we consider important, with regards to Logo and to our project. When Jesús, the first author of this paper, went to Eurologo 2007 in Bratislava, he was very impressed by all the different versions and off-springs of Logo. Upon his return to Mexico, he felt temporarily that perhaps the version of Logo that was being used in our school projects (MSWLogo, which had been provided as part of the EMAT programme) was poor or outdated. But after a few days, he realised that it's not about the interface, and that the Logo language was in fact marvellous: Around that same time some officials from part of the Ministry of Education suggested that it was obsolete and outdated to not use other more modern tools, so he downloaded Scratch, with which we have also been working for a year. What we noticed by working with Scratch (which we also like) is that Logo is ageless and is far from obsolete: it is neither old, nor new, because Logo is Logo – a language – and it can be as current as we want depending on the use we give it, and we have found it invaluable in our Eiffel tower project.

This also reminds us of a comment made by Celia Hoyles in a visit that she and Richard Noss made to our school a few years ago; at the end of their visit, she said something like: "when I see your students working with Logo, my faith that Logo is still current, is reborn". What we can add ourselves is that our experience with Logo is incredible. We believe that Logo is a fabulous tool: a tool for constructing, developing abilities, and learning how to think, that not many other software have matched. Many more novel and spectacular tools have appeared, but some are only used because they are fashionable or because others are using them – some users only want the latest and most externally appealing tools – without considering their true educational potential. We have tried a variety of software, but we only use that which we see as potentially fruitful (such as those we presented here: e.g. Google SketchUp and our blog), though we also believe in the value and importance of using a variety of approaches and modes of representation with which students can engage and interact (Wilensky, 1991).

We are not against evolution, but we are against an indiscriminate and unthought use of new software or ICT tools. And amongst the sophistication of much modern software, our students still like Logo best. Another one of this year's 13-year old students said the following:

> *"Logo is a very cool and interesting software that teaches you to do different figures and you can create various procedures as if you were doing mathematics; you realize that with very little tools you can do lots of things and it really is all very intriguing all the things that you can do with these commands; it is fun and brilliant. And that is what I can say about this software that has taught me so much and the truth is that I would like to continue working with it."*

Last year, with the following critical statement, one of our 13 year-old students explained why he preferred Logo to other more modern software:

"…because [in Logo] I can express myself... [whereas] I think buttons make human beings obsolete(!), that's why I don't like most modern software, because it is just about pressing buttons… What should be done is to let users create their own tools… But [modern software] is not good, it is making human beings learn not to think, whereas with Logo one has to think…"

Wow to whoever thinks that Logo is just a programming language – Logo is a philosophy; and just like mathematics is not just about numbers and is a way of thinking, Logo is also a way of thinking. Borrowing some of Papert's (1980, p.18) words, Logo "is a particular way of using computers, of forging new relationships between computers and people".

Acknowledgement

We are grateful for the collaboration in the implementation of the projects of Alessio Gutiérrez-Gómez of the Escuela Secundaria Diurna #229 "Ludmila Yivkova", T. Matutino, in Mexico City, as well as to all the children we have worked with in the Mexico City schools Escuelas Secundarias Diurnas #229 "L. Yivkova" T. Matutino & #139 "J. E. Rodo" T. Vespertino.

References

diSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy.* MIT Press, Cambridge.

Harel, I., & Papert, S., (eds.) (1991). *Constructionism*. Ablex Publishing Corp., Norwood, NJ.

Jiménez-Molotla, J.; Gutiérrez-Gómez, A., Sacristán, A.I. (2007). Painless Trigonometry: A Tool-complementary School Mathematics Project. In *Proceedings 11th EuroLogo Conference: 40 Years of Influence on Education.* Edited by I. Kalaš. Comenius University, Bratislava.

Jiménez-Molotla, J.; Gutiérrez-Gómez, A., Sacristán, A.I. (2009). Pyramids in Logo: A School Project in 'Search' of the Fourth Dimension. In *Education and Technology for a Better World*, IFIP Advances in Information and Communication Technology series. Edited by A. Tatnall & A. Jones. Springer, Boston. Pp. 92-101.

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*, New York: Basic Books.

Papert, S. (2006) From the Math Wars to the New New Math. Keynote speech presented at the Seventeenth ICMI Study 'Technology Revisited' Study Conference. Hanoi, Vietnam.

Wilensky, U. (1991). Abstract Meditations on the Concrete, and Concrete Implications for Mathematics Education. In *Constructionism.* Edited by I. Harel & S. Papert, pp.193-204. Ablex Publishing Corp., Norwood, NJ.

Learning how to teach robotics

Martina Kabátová, *kabatova@fmph.uniba.sk*
Dept of Informatics Education, Comenius University, Bratislava

Janka Pekárová, *pekarova@fmph.uniba.sk*
Dept of Informatics Education, Comenius University, Bratislava

Abstract

For over three years we have been developing and implementing a curriculum for the pre-service teachers that would introduce them to both educational robotics and core constructionist concepts. Activities with robotic models, programmable kits and toys are attractive opportunity to organize lessons in the constructionist way. In this paper we **describe our robotics course**, we **compare two robotic kits**, we **observe how creative robotic principles work with our target group** and we **scrutinize how to provide instructions to boost construction of knowledge**.

We prepared and taught the seminar consisting of about 11 lessons each term. The seminar is regularly attended by 10 to 23 students from various study specializations. During the course they work both on small close-ended tasks and their own big projects that include: design of a model, construction, programming and solving mechanics problems. In this process we encourage partnership and dialogue. We have used two kinds of robotic kits: LEGO Mindstorms NXT and LEGO WeDo - designed for younger children with lots of simplifications in comparison to NXT. That's why while working with WeDo set, we chose less structured activities. A group using WeDo solved several smaller tasks until they got a grip on what the kit and programming language were like. The final project started with discussion and we introduced the principles of creative robotics for all as they are proposed in (Rusk et al., 2008): focus on theme, combine art and engineering, support storytelling, organize exhibitions. There are two different levels of applying these principles: (1) our students are learners and they should experience robotics same way as any other learners, thus we encourage them to experience every aspect of designing, building, programming themselves; (2) our students are pre-service teachers and they should reflect what they do with robotics kit in a teacher's perspective.

Figure 1. Four robotic models built using LEGO WeDo programmable kit

We consider these principles to be a good way to teach students of informatics education robotics creatively. Finally we propose basic framework for robotics course. In both proposed phases some instructions are in order, the amount and their nature depend i.a. on the robotic kit.

Keywords

robotics; constructionism; pre-service teachers; LEGO WeDo; LEGO NXT

Educational robotics and constructionism

For over three years we have been developing and implementing a curriculum for pre-service teachers that would introduce them both educational robotics and core constructionist concepts. We take inspiration from several existing robotics courses, e.g. (Alimisis et al., 2007). Future teachers should get at least basic education in this field since robotics is included in our national informatics curriculum and various researches reveal that learning with programmable toys and robotic kits is an attractive opportunity for **developing vast set of skills and knowledge**.

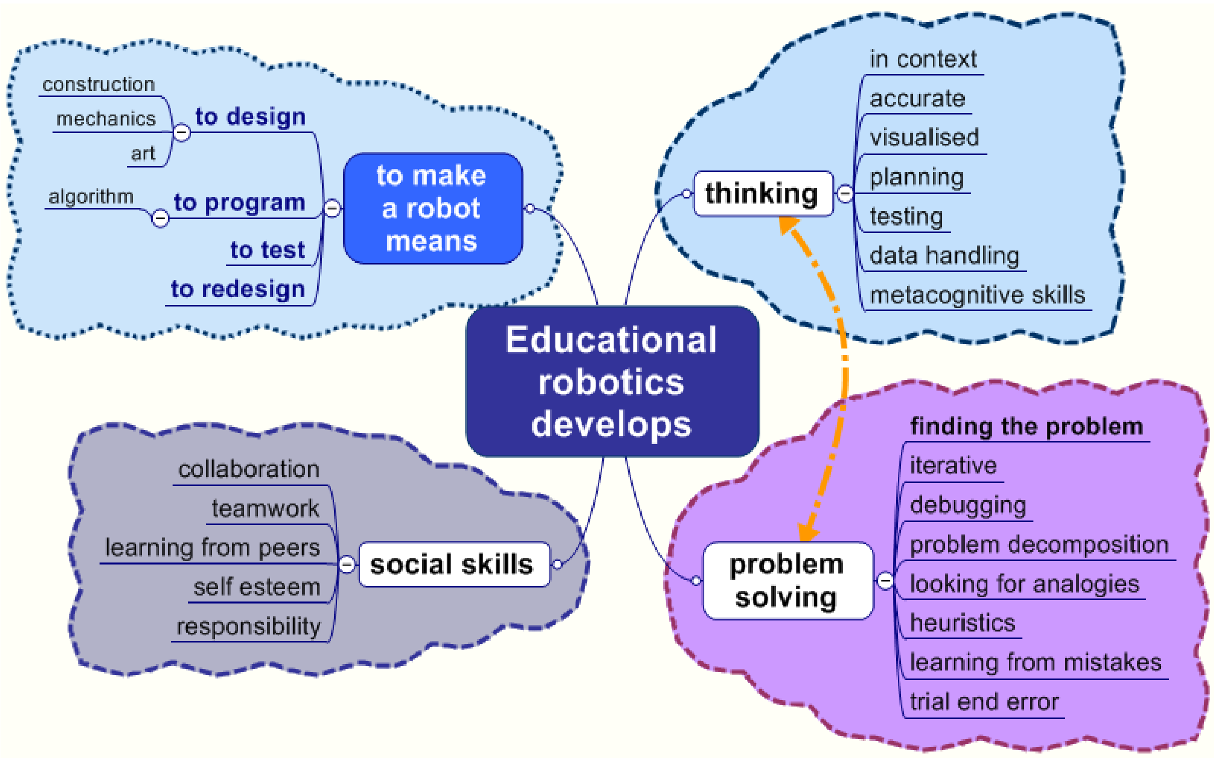


Figure 2. Activities within educational robotics may develop many competences

Activities with robotic models, programmable kits and toys are good opportunity to **organize the lessons in constructionist way**. The hands-on nature of learning with robots "*embodies a distinctly constructionist philosophy of learning*" (Evans, 2006). The constructionist ideas and principles (Papert, 1999; Rusk et al., 2008) we promote in our lessons:

- **learning by doing**, **hands-on** activity through **experience** - building a robotic model,
- **genuine achievement** and **own solutions**, **problem finding** - deciding what the model should do and how to achieve it, which theme to choose, exploring programming language,
- **hard fun** and **playful learning** - robotic kits are basically toys though making fully functioning model could be a hard task, the atmosphere at course is loose and playful,
- **learning through designing**, inventing and creating - creating robotic model involves this,
- technology as building material combined with artistic materials - exhibition settings, props,
- **taking time** - we don't have strictly given syllabus, we can freely explore within this course,
- freedom to **make mistakes** - we provide only limited instructions, students work on their own and they do make mistakes, we usually inquire what the problem is and help to fix it,
- **teamwork**, collaboration, sharing work and ideas - students learn how to manage their work in group, how to divide and assign tasks, some assignments (e.g. robots for contests) are not possible to solve by single person,

- **teachers learn too** - we are often in position when we have to solve unknown problems we are not prepared for, we have learnt quite a lot about robotics while helping the students.

Robotics course for pre-service teachers

We led a robotics course for pre-service teachers since 2006 to 2010. We prepared and taught the seminar consisting of about 11 lessons each term. The seminar is regularly attended by 10 to 23 students from various study specializations - informatics, applied informatics, informatics education, mathematics. It is primarily designed for future informatics teachers. Therefore even if there were none of such students, we constantly introduced problems and projects that were focused on educational issues.

At seminar students usually work in small teams of two or three members. We note that these students have already attended rather comprehensive courses on programming so they are not beginners. During the course they work both on small close-ended tasks and their own big projects that include: design of a model, construction, programming and solving mechanics problems. They also have to prepare a documentation for their project and fill out special worksheet (a lesson plan) that describes in more detail how to do the same model with children. Students are also encouraged to take part in robotics competitions – as volunteers helping with FIRST LEGO League tournament organization or as contestants in local robotics competition.

Last term we had an opportunity to use LEGO WeDo robotic kit with our students. We divided them into two groups – six pre-service teachers and one informatics student worked with this kit, the other students that don't specialize in education proceeded with LEGO NXT. In following section we try to compare our methodology of work with these two groups. We consider the processes running on the seminar as the constructionist ones. Let's have a closer look at them.

Work on seminar – constructionism vs. instructions

First four lessons focus at the basics of **LEGO Mindstorms NXT** robot programming. Using the set of close-ended tasks students should investigate what possibilities NXT robotics kit and its programming language offer. There are various reasons for that, some of them **practical**: at the beginning of the term we don't really know which students will really attend the whole course, the teams are switching members, some people quit and new ones can come. Therefore we can't let them do continuous bigger project. More important reasons are **didactical ones**: students get familiar with basic programming rules in NXT programming environment and learn to design simple programs for specific robotic model. They can experience data-logging by sensors, estimate the actions of the robot and analyze expected outcomes of their program in real conditions.

Smart cars are dream of each tired driver. Your robotic car also knows several tricks. Create following program:

- *If the car is out of petrol, it will slowly speed down to speed level 10. Afterwards it will beep and stop. There will be gas station image blinking on display.*
- *After car owner's whistling a car will move from its parking place to the restaurant.*

(Example of close-ended task)

Introduction to simple programs on the pre-built robotic model is the commonly used approach in robotics courses, see (Sklar and Eguchi, 2004) or (Lau et al., 1999). Still, there is great difference: we don't provide students with many instructions how to solve problems. We give them challenges containing little hints and let them proceed at their own speed. We assume that they should use their previous experiences as programmers to be able to understand programming language. If they ask for advice, we help them solve problems and we answer questions. Sometimes we work as catalysts for finding mistake – we try to invoke an idea what the students should change in environment in order to test their robot's behavior more effectively or help them read the codes they made from different point of view. Occasionally we also help students with deciding what their model should do and if it's possible to create such robot using this particular kit.

We encourage **partnership and dialogue** between students and us. If a team doesn't like the assignment, but suggests different idea what to do with robotic model, they can execute their idea. However, we inquire what they are doing and why, how it relates to their learning of robotics or how this could be used at the school in classroom environment.

A group of students was offered to program a golf player. At first, they have explored demo programs in Help section of programming environment. The idea of robot moving its arm has inspired them to program the fisherman – it could reel the fishing rod after the sensor is pressed.

While working with WeDo set, we chose less structured activities. LEGO WeDo is designed for younger children than NXT kit and there are lots of simplifications in comparison to NXT:

LEGO Mindstorms NXT

- Age recommendation 8+
- Programmable brick
- Three motors
- Four types of sensors (sound, light, ultrasonic, touch)
- More parts – bricks, gears, wheels etc.
- Autonomous device - not necessary connected to computer

LEGO WeDo

- 7 to 11 years old children
- Programmable USB hub
- One motor
- Two sensors (tilt sensor that detects 6 different positions, motion sensor that detects objects)
- Small number of classic LEGO pieces
- Necessity to be connected to computer

Programming languages for both robotics kits are **drag-and-drop** and **icon-based**. WeDo language is much simpler and contains cycle, wait command, motor motion commands, parameters (sensor input, number, text, random), sound replay, value display, background display etc. Both languages enable parallel processing of instructions.

Figure 3. LEGO WeDo kit - simple program that moves the motor according to sensor tilt

WeDo group started to explore the possibilities of the robotic kit directly – they were asked to build and program some of **demo examples** presented in the kit materials. Each team completed the task within one seminar, a team of two boys managed to build and program even two models. On next seminar they should have designed, built and programmed new **models linked by common topic**: playground equipment. All teams succeed. Still, we couldn't see much enthusiasm among the students. Later we found out it was not caused by too simple interface of the kit as we had supposed. We undertook some changes in project initialization which proved to be successful.

Figure 4. LEGO WeDo models built by students during first lessons

After completing Playground project the students tested their understanding of the language on the **set of close-ended tasks**. They should have also evaluated their skills – they felt they knew basics of the language and are ready to use it with children.

Both NXT and WeDo group proceeded their robotics education with planning **own big project**. We can find some differences between their work process: while NXT group spent most time with construction issues and some minor problems in programming couldn't be solved due to the end of the term, WeDo group prepared complex set of robotic models, programmed them and added some extra artistic effects too. Their success is partially subject to the simplicity of the interface. We suppose the other reason is the way how we organized the work on the project. Therefore we will describe it in more detail.

LEGO WeDo: The spooky castle project and creative robotics

After finishing the activities with WeDo playground models we introduced new big project that will include all four kits and seven students working together as one team. We opened this activity with discussion and we introduced the principles of creative robotics for all as they are proposed by MIT Lifelong kindergarten group (Rusk et al., 2008):

- **focus on theme,**
- **combine art and engineering,**
- **support storytelling,**
- **organize exhibitions.**

There are two different levels of applying these principles: (1) **our students are learners** and they should experience robotics same way as any other learners, thus we encourage them to experience every aspect of designing, building, programming themselves; (2) **our students are pre-service teachers** and they should reflect what they do with robotics kit in a teacher's perspective.

The theme

The students discussed two themes we have offered: Intelligent house and Spooky castle. Their discussion was focused on robotic kit - it's feasibility and programming language restraints. They reasoned against Intelligent house project: *"This kit is NOT intelligent. I can't code even an IF statement. It will be easier to make something that moves, makes noises."* This statement persuaded whole group to take on the Spooky castle project. The lector acted as supervising teammate and helped the students draw a map of their ideas while they were brainstorming and throwing in the ideas for individual models. After short revision of the map the group realized some of them were not possible to construct and program via WeDo means. They picked four that seemed possible to make and came up with the idea that the Spooky castle is in fact an amusement park attraction. We didn't provide the students with examples of premade models in this period, though we suppose it would be helpful.

Figure 5. The Spooky castle models: the tunnel, skeleton, jumping pyramid monster, paper ball shower

Artistic design and LEGO models

We asked the group to put down a list of resources they will need, we have hinted that we would bring even artistic material for some decorations. They asked for cardboard, paint, paper, scissors and other materials. After some experience with WeDo they knew they will need some longer cables to attach WeDo USB hub since it has to be plugged in order to work. One of girls brought her own LEGO train and the group built their models on its route. The students (especially girls) spent lot of time making the cardboard castle using all the artistic material we gave them. The girls also explored how to record and play custom sound effects. They recorded their own sounds (various screams, spooky ambient) and used them in WeDo program. All four models feature the motion sensor. Three of four models have motor attached. Students were discussing usage of the tilt sensor, but they were not able to come up with any reasonable use of it in this particular project.

Teamwork

We've observed closely the dynamics in group teamwork. The tasks were defined very soon and distributed among the students. Boys were assigned to build and program 2 models, girls made the other 2 models. The models were rebuild and repaired several times by the boys and the girls as well. Mostly girls made the castle prop and recorded the sounds. Whole group cooperated each seminar at assembling and disassembling the project set since we had to move it to another room. At one point when the group was told they need to finish the work and conclude the project they were in need of better work management. In this situation a dominant girl took the leading role and told everyone what needs to be done. After this the group quickly finished everything and prepared the project for exhibition.

Figure 6. The Spooky castle made of cardboard

Exhibition

The other group of students which have worked with LEGO NXT by then was invited to watch the Spooky castle presentation. We have also recorded whole event. WeDo group was proud to present their work and it met with appreciation of the NXT group. They even asked some question how the models were made.

Storytelling

This aspect of creative robotics have never had any success in our seminars. We suspect that the groups we work with are not average groups - they are students of various informatics specializations and thus it is possible they don't have a special like for storytelling.

Educational reflection of the project

We concluded the project with a discussion with WeDo group.

- The students valued the opportunity to work in bigger team, to work on common long-term project and to make props and work with artistic materials. They suggested that same experience could be suitable also for children.
- They labeled the WeDo programming language as fairly easy and potentially appropriate for primary kids, but they also pointed out several bugs and strange behavior.
- They haven't noticed we hadn't given them any instruction on how to create WeDo programs, they actually liked discovering on their own.
- They expressed a fear that untrained primary school teachers might have serious problems with teaching WeDo robotics because there occur various problems that are often solved only with broad knowledge of computers and programming.
- They stated that the biggest issue with robotics activities was that they needed lot of time to identify the problems first, prior to solving them. This could lead to organizational problems in classroom.

Conclusion

We have come to conclusion that these principles are a good way to teach students of informatics education robotics and creative robotics is a concept worth introducing to them. To give students the opportunity to create the setting, the props and use different kinds of materials is valuable experience. Final exhibition is a also good experience for pre-service teachers. Storytelling opportunity is not necessary for this particular target group.

We suggest that the robotics course should take place in two phases:

- first introduce the particular robotics kit via **smaller projects** that would reveal it's applicability and constraints. The amount of instructions in this phase depends on the robotic kit and programming language. We feel the need for more instructions while using NXT than while using WeDo.
- The next step is one **bigger project** (or more if there is a time) that is **based on teamwork**.

We think that **some guidance is appropriate** while discussing and identifying the problems that are to be solved, that involves also the discussion of project theme (we suppose that if pre-made model examples are provided along with theme names, this discussion might be less needed). **Guided discussion** about possible problems with robotic models can reduce difficulties students might have with execution of their ideas and can reduce time that they need to finalize what they want to create.

References

Alimisis, D., Moro, M., Arlegui, J., Frangou, S. and Papanikolaou, K. (2007) *Robotics & Constructivism in Education: the TERECoP project*. In Proceedings of EuroLogo 2007. Edited by I. Kalas. Bratislava, August. pp. 39.

Evans, D. (2006) *Teaching robotics with LEGO Mindstorms*. Futurelab. Retrieved online, 10.3.2010, http://www.futurelab.org.uk/resources/publications-reports-articles/web-articles/Web-Article495.

Lau K. W., Tan H. K., Erwin B. T., Petrovic P. (1999) *Creative Learning in School with LEGO Programmable Robotics Products*. In Proceedings to Frontiers in Education'99, IEEE CS Press, pp. 12D4/26 - 12D4/31 vol.2.

Papert, S. (1999) *The Eight Big Ideas of the Constructionist Learning Laboratory*. Unpublished internal document. South Portland, Maine. Quoted in Stager, G. (2005) Papertian Constructionism and the Design of Productive Contexts for Learning. Plennary Session Paper – EuroLogo X, Warsaw, Poland. Retrieved online, 10.3.2010, http://www.stager.org/articles/eurologo2005.pdf.

Rusk, N., Resnick, M., Berg, R., & Pezalla-Granlund, M. (2008) New Pathways into Robotics: Strategies for Broadening Participation. *Journal of Science Education and Technology*, vol. 17, no. 1, pp. 59-69.

Sklar, E., Eguchi, A. (2004) *Learning while Teaching Robotics*. In The AAAI Symposium Series: Accessible Hands-on Artificial Intelligence and Robotics Education.

The BehaviourComposer 2.0: a web-based tool for composing NetLogo code fragments

Ken Kahn, kenneth.kahn@oucs.ox.ac.uk
Oxford University, 13 Banbury Road, Oxford, OX2 6NN.

Howard Noble, howard.noble@oucs.ox.ac.uk
Oxford University, 13 Banbury Road, Oxford, OX2 6NN

ABSTRACT

The Modelling4All Project is building the *BehaviourComposer*, a web-based tool for constructing, running, visualising, analysing, and sharing agent-based models. These models can be constructed by non-experts by composing pre-built modular components called *micro-behaviours*. We are attempting to seed and nurture a Web 2.0 community to support modelling. Models, micro-behaviours, lesson plans, tutorials, and other supporting material can be shared, discussed, reviewed, rated, and tagged.

The *BehaviourComposer* supports a *middle-out* style of constructionist learning. Rather than begin with primitives of a programming system and build upward, learners browse for pre-built code fragments. They then customise and compose these components to quickly build rich agent-based models. For learners interested in particular scientific topics (e.g. how epidemics spread, how fish school, or how unequal wealth distributions can emerge) this provides a means to explore their scientific interests without first investing time and effort in mastering a programming language. For teachers in life, physical, or social sciences it provides a tool enabling their students to creatively learn in an exploratory fashion both the scientific topic being model as well as learning about the computer modelling more generally. Importantly the model components are transparent, in that they are small bits of readable NetLogo code that ambitious students and teachers can delve into and edit or author new ones.

The BehaviourComposer also supports *collaborative* model building in a similar manner to how Web 2.0 tools such as wikis and Google Docs support collaborative authoring. The models are hosted on the Modelling4All web site where geographically separated collaborators can make contributions to a shared model. They can enable real-time collaboration where changes any of them make are seen by the others in a few seconds. In addition to collaborating on model construction the web-based nature of the BehaviourComposer makes it easier to have collaborations where some author the models, others test or analyse the models, while others produce guides or video tutorials. Another form of collaboration is where some who are good NetLogo programmers author new specialised micro-behaviours while others without those skills explore the combinatorial possibilities of those behaviours to build a range of models.

Keywords

Agent-based modelling, NetLogo, simulation construction kits, micro-behaviours, BehaviourComposer, Web 2.0

SOCIAL SUPPORT FOR MODELLING BY NON-EXPERTS

The Modelling4All Project began by building upon the results of the Constructing2Learn Project [1, 2] also at Oxford University in which a modelling tool called the *BehaviourComposer* was designed, implemented, and deployed for use by students. The *BehaviourComposer* had a web browser component for browsing web sites of code fragments called micro-behaviours. These are bits of code that were carefully designed to be easily understood, composed, and parameterised. The *BehaviourComposer* user attached these micro-behaviours to prototype agents. In order to create models containing many instances of a prototype agent, a micro-behaviour for making copies was added to the prototype. When the user wished to run the current model, the *BehaviourComposer* assembled a complete program and launched it. The program assembled *NetLogo* [3] programs, but the framework could be adapted for other modelling systems such as *Repast* [4].

The Modelling4All Project has constructed the *BehaviourComposer 2.0* which is a complete redesign and re-implementation of *BehaviourComposer* in order to run in web browsers. There are many advantages to providing applications via web browsers. In many organisations, universities, and schools computer systems are “locked down” and only administrators can install or upgrade software. BehaviourComposer 2.0 allows users to save their work on servers, facilitating sharing and mobile use. Web browsers exist in nearly every operating system and on many mobile devices. The system is easy to use because the user interface builds upon the familiar web browser interface.

The Modelling4All Project has another reason for choosing a web-based approach. We are striving to build a web site (http://modelling4all.org) to support an online community as they design, build, analyse, validate, and verify models. We see great potential in using the Web 2.0 concepts that have been so successful in sites such as Wikipedia, flickr, YouTube, del.icio.us, and FaceBook. We have designed *BehaviourComposer 2.0* to facilitate embedding it and the models users create in other web-based tools. In this way a community of modellers can share, discuss, review, rate, and categorise the models, micro-behaviours, and supporting materials. Users can embed their models in their blogs, wikis, web sites, discussion forums, and email.

CREATING MODELS BY COMPOSING MICRO-BEHAVIOURS

BehaviourComposer 2.0 provides libraries of generic micro-behaviours organized into categories for specifying the initial state of agents, movement, appearance, attribute maintenance, scheduling, layout, copying, interactions, links, and social networks. In addition there are micro-behaviours for creating graphs, histograms, sliders, buttons, and event logs. Specialised libraries of micro-behaviours have been created for modelling epidemics, collective decision making, network formation, predator/prey ecologies, artificial economies, and low carbon ICT.

These libraries of micro-behaviours have been created by the Modelling4All team, but *BehaviourComposer 2.0* can use micro-behaviours hosted on any web site. A micro-behaviour can be authored by any web page creation software (including wikis). The *BehaviourComposer 2.0* processes the micro-behaviour web pages to add buttons to facilitate using or editing the micro-behaviour.

Users construct models in *BehaviourComposer 2.0* by adding micro-behaviours to prototypical agents. They can focus initially on getting a single individual of each “type” to behave correctly.

Then they can add a micro-behaviour to create the desired number of copies of the prototype. The fresh copies can easily be given additional behaviours to produce a heterogeneous population.

Micro-behaviours should not be confused with the software engineering concept of modules, components, or other programming language abstractions such as packages, classes, methods, or procedures. These modular constructs have interfaces that must be carefully matched in order to combine them. They represent program fragments that run only if another fragment invokes them. In contrast, micro-behaviours run as independent processes, threads or repeatedly scheduled events. They are designed to run simultaneously with a minimum (and in most cases zero) need to coordinate their execution order and interactions. Micro-behaviours resemble the structured processes in the *LO* programming language [5].

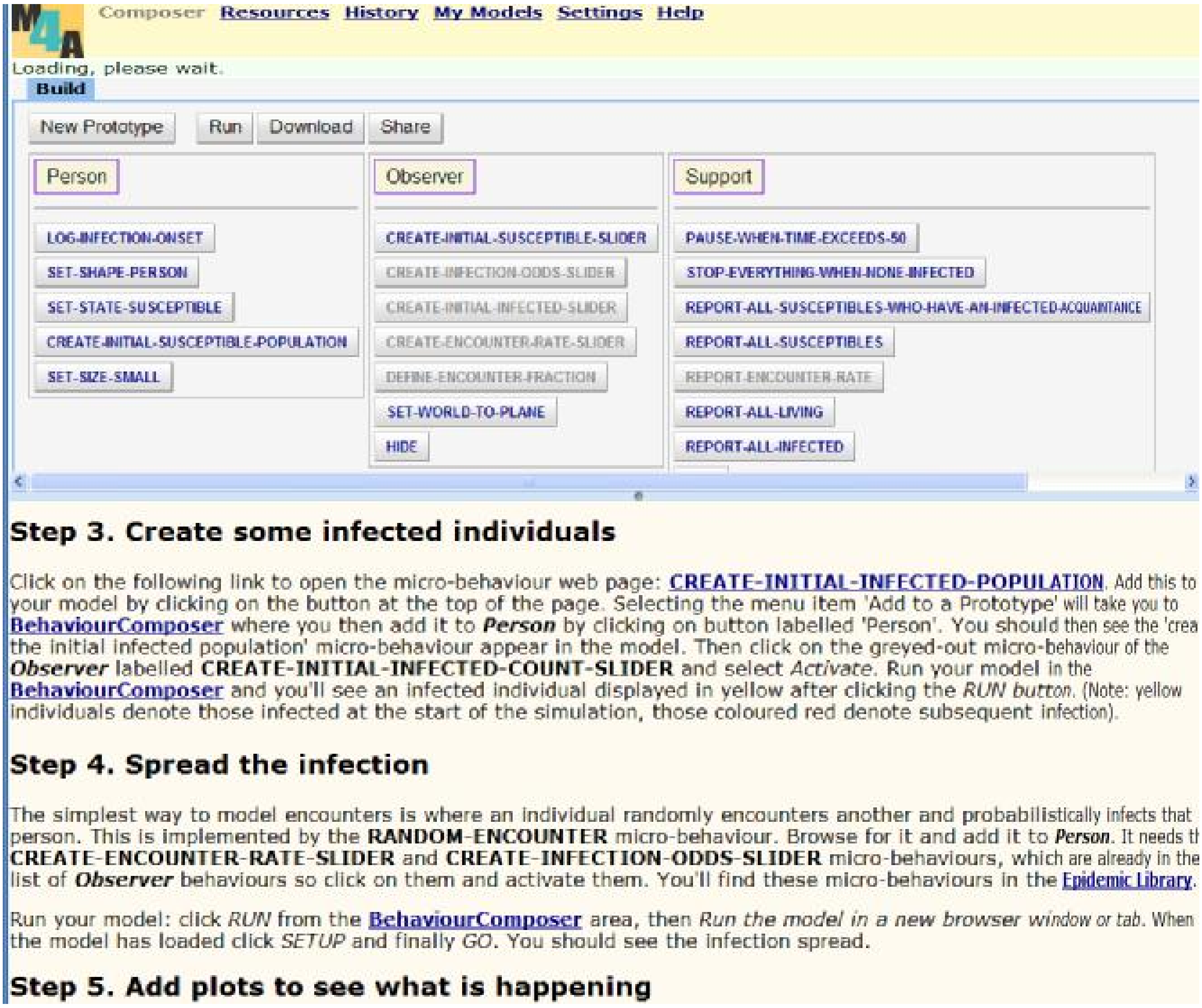


Figure1. Screen shot while constructing a model following a guide (in the bottom pane)

An illustrative example of a micro-behaviour is identified by the URL: http://modelling4all.nsms.ox.ac.uk/Resources/Composer/en/MB.4/RANDOM-ENCOUNTER.html. It contains the following code fragment:

```
do-every 1
   [do-if my-state = "infected"
       [do-for-n
           the-encounter-rate
           all-individuals with [my-state != "dead"]
           [set my-last-encounter the-other
            add-behaviours list-of-micro-behaviours "Encounter Behaviours"
                           [POSSIBLE-INFECTION.html]]]]
```

Our *NetLogo* extension `do-every` is critical for composing micro-behaviours. It repeatedly schedules an action that conditionally adds the POSSIBLE-INFECTION micro-behaviour. The reliance upon a scheduler associated with each agent greatly facilitates the composition of micro-behaviours without concern for component interfaces. This code fragment references another micro-behaviour POSSIBLE-INFECTION by providing the URL hosting the micro-behaviour. One source of name conflicts resulting from composing components is avoided by using the World Wide Web's global name space of URLs.

AUTHORING, CUSTOMISING, GROUPING, AND SHARING MICRO-BEHAVIOURS

It is relatively easy for a programmer to create micro-behaviours for specific uses. There are many challenges in creating *reusable* micro-behaviours including:

- Enabling non-programmers to easily specify parameters
- Enabling non-programmers to easily specify references from a micro-behaviour to other micro-behaviours
- Sharing attributes between micro-behaviours
- Properly scheduling micro-behaviours that depend upon other micro-behaviours

The first two challenges are largely met by the user interface of the BehaviourComposer. Web browsers support text areas where users can enter text (typically as part of the process of filling out forms). These text areas are used to provide easily editable parameter values (the '9' in Figure 2, for example). A micro-behaviour that references lists of other micro-behaviours can use the same interface that is used for collecting a list of micro-behaviours for prototype agents (see Figures 1 and 3).

There is no software support for dealing with conflicts between micro-behaviours that use the same attribute name for different purposes. Programmers need to trade-off between short simple names with their ease of reading, writing, and the clarity of references (e.g. in the history tab) with long names that perhaps even include the names of authors or projects involved. The default library of micro-behaviours uses names such as 'my-state', 'my-age', and 'my-acquaintances'. Perhaps as the community of micro-behaviour authors grows we will need to reconsider these names and use names such as 'my-flu-infection-state' instead of 'my-state'.

Early versions of the BehaviourComposer required that micro-behaviours be authored in HTML. HTML tags and attributes were used to identify the name and code fragment on a micro-behaviour page. References from a micro-behaviour to other micro-behaviours relied upon web page links. Editable parameters relied upon HTML *TextArea* elements embedded on the page. In addition to these essential uses of HTML, it is heavily used for rich text formatting and providing links to related or background materials. While anyone capable of constructing computer programs can easily master HTML we discovered this reliance of HTML authoring limited hosting possibilities for micro-behaviour pages.

Due to the popularity of tools such as wikis, blogs, virtual learning environments, and other online web page creation tools, we needed to provide plain text alternatives to HTML authoring. The Modelling4All server code was enhanced to accept micro-behaviour pages with special textual tokens and transform these pages to the necessary HTML. These pages can of course still use HTML for rich text and links and all the online authoring tools support this. Few online web page authoring tools support the ability to add text areas or attributes to elements. As example of avoiding reliance upon HTML consider the micro-behaviour for creating copies of an agent (). It contains this HTML fragment:

```
<font size="2" color="gray">Begin micro-behaviour</font>
<p><b>ADD-COPIES</b></p>
<font size="2" color="gray">Begin NetLogo code:</font>
<pre>
substitute-text-area-for number-of-copies 9
add-copies number-of-copies
           list-of-micro-behaviours "Additional behaviours"
                                    [SET-RANDOM-POSITION.html SET-RANDOM-
HEADING.html]
</pre>
<font size="2" color="gray">End NetLogo code</font>
```

The HTML elements are serving minor roles. When stripped of all HTML it still functions properly. Here is a plain text version (where the special tokens are depicted in bold face):

```
Begin micro-behaviour
ADD-COPIES
Begin NetLogo code:
substitute-text-area-for number-of-copies 9
add-copies number-of-copies
           list-of-micro-behaviours "Additional behaviours"
                                    [SET-RANDOM-POSITION.html
SET-RANDOM-HEADING.html]
End NetLogo code
```

The server proceeds by first extracting the name of the micro-behaviour ‘ADD-COPIES’ and the code fragment that calls the NetLogo procedure ‘add-copies’. The code is transformed to replace ‘number-of-copies’ with an HTML text area initially containing ‘9’. The list of micro-behaviours is replaced by a custom GWT widget that provides an editable list of “live” micro-behaviours. The resulting HTML is rendered as in Figure 3 (typically there is surrounding rich text providing documentation, variants, related behaviours, etc.).

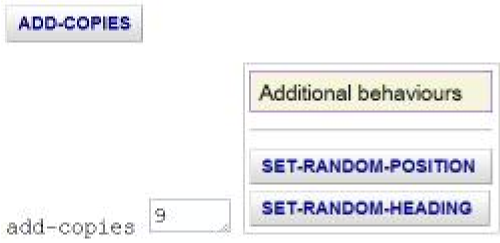


Figure 2. Screen shot of the essential part of the ADD-COPIES micro-behaviour

The only difference a user would see if the source page was stripped of all HTML would be that the name '`ADD-COPIES`' would not be in bold face. The ability to use arbitrary HTML in the name of micro-behaviours does provide a way to augment or replace names with icons or images.

Another problem with using wikis or blogs to host web pages is the hosting program adds additional material (navigation aids, help buttons, editor controls, and sometimes advertisements). We enable authors to select just a portion of a page to appear within the BehaviourComposer by adding special start and end tokens.

An additional benefit of defining micro-behaviours using unique tokens to identify elements is that search engines can find micro-behaviours using search terms such as "Begin micro-behaviour". If the search is carried out within the BehaviourComposer then the micro-behaviour pages are presented in their processed form. Any search engine can be queried from within the BehaviourComposer.

A WEB-BASED MODELLING TOOL

BehaviourComposer 2.0 is built upon the *Google Web Toolkit* (GWT) [6] and *NetLogo*. *BehaviourComposer 2.0* is a rich internet application (a web application with features comparable to desktop applications) using AJAX [7]. GWT supports interface elements such as tabs, panels, buttons, and editors as well as facilitating communication with servers. Users interactively assemble micro-behaviours into collections that represent prototypical agents. When the user clicks the *run* button, the server assembles a complete *NetLogo* program. The user can then run the program in their browser as a Java applet or download the program into *NetLogo*.

BehaviourComposer 2.0 supports micro-behaviours that use *NetLogo's* facilities for animating simulations, providing sliders for interactively exploring the parameter space, producing dynamical graphs, and interactively running experiments. Other *NetLogo* tools such as the *BehaviorSpace* for automating the exploration of the parameter space and gathering statistics are only available after launching *NetLogo* as an application rather than a browser applet.

Each micro-behaviour is presented as a web page which can be accessed via links, tags, or a search engine just like any other web page. Browsing for micro-behaviours uses the same tools and skills as web browsing for any other kind of information. New tools and skills do not have to be mastered.

A section of the web page is the program fragment itself (see Figure 3). A button is automatically generated when the page is loaded into *BehaviourComposer 2.0*. When the button is pushed, the code fragment is added to the desired prototype agent or list of micro-behaviours. By convention, the rest of the page includes sections that

- describe the behaviour
- describe how to edit the micro-behaviour to produce variants

- provide links to related micro-behaviours
- describe how the program fragment implements the desired behaviour
- a history of edits to the micro-behaviour

Some pages also have references to published papers and links to sample models using the behaviour. The addition of formal specifications of micro-behaviours is a topic of future research.

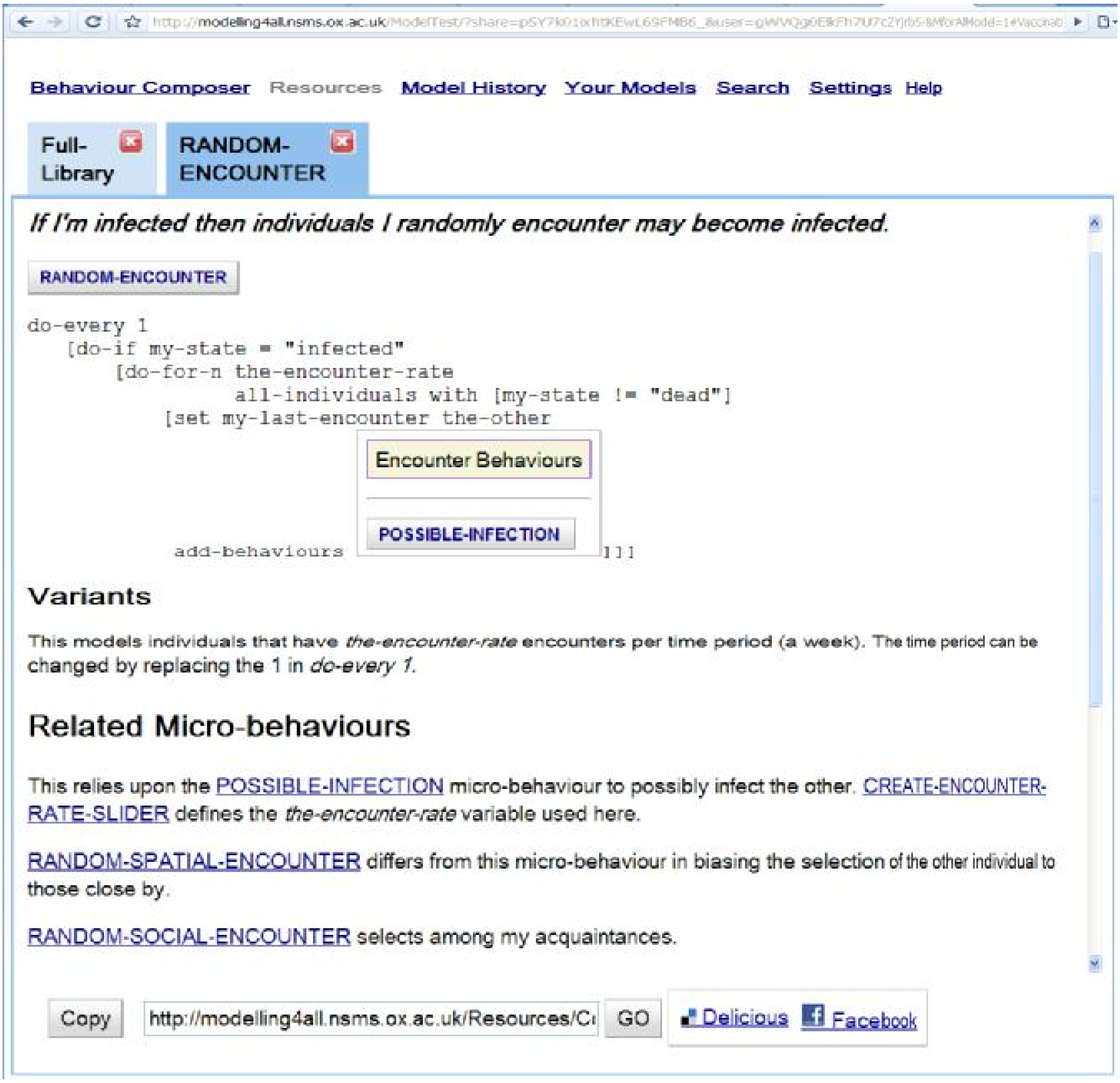


Figure 3. Screen shot of a micro-behaviour

The identity of a micro-behaviour is its URL. A micro-behaviour that references other micro-behaviours (e.g. adding new micro-behaviours to other agents) does so by providing web links to the referenced micro-behaviours. The "owner" of the URL can then update the contents to upgrade the micro-behaviour. Model makers, who instead want a snapshot of the current micro-behaviour, need to copy the contents of the page to another URL. The Modelling4All web site supports this copying (or editing) of micro-behaviours and produces new URLs with unique identifiers.

By relying upon URLs we avoid any possibility of name conflicts between micro-behaviours. It does introduce reliance upon the "good behaviour" of micro-behaviour web page authors. It also potentially introduces additional points of failure if a model relies upon web pages that are not currently available. The Modelling4All server does cache the code fragments of micro-behaviours but this introduces new issues of staleness of items in the cache. We are exploring policies that inform users when either the cached code differs from the current or the web page is inaccessible.

We have recently been exploring a style of usage of the BehaviourComposer that builds models from very generic building blocks. For example the RANDOM-ENCOUNTER behaviour in Figure 3 could alternatively be built by adding a DO-IF behaviour to the DO-EVERY behaviour. A DO-FOR-N behaviour could then be added to the DO-IF. UPDATE-ATTRIBUTE and ADD-BEHAVIOURS micro-behaviours could be added to the DO-FOR-N behaviour. These micro-behaviours are given more appropriate names and parameterised appropriately. We have found that most specialised micro-behaviours can be replaced by a composition of appropriate generic micro-behaviours in this manner.

Every change made to the model (adding or removing micro-behaviours, adding or removing prototypes, renaming, updating parameters, or loading sub-models) is communicated to the server. The server maintains a session history that is identified with a globally unique identifier. A user can resume a session either by relying upon their browser's cookie mechanism or by bookmarking a session URL. Small teams can share a session ID to facilitate collaboration. They can choose a real-time collaboration option so that changes made to the model are seen within a few seconds by all sharing the session.

Sessions are integrated with the browser's history facility. Any changes to a model can be undone by using the browser's *back* button. They can then be restored using the browser's *forward* button. *BehaviourComposer 2.0* has a history tab that lists descriptions of every model change. By clicking on entries users can restore the model to any point in its history.

When a user runs a model they are presented with new tabs that enable the user to execute it, embedded it in various ways in other web pages, or to export their model as XML. Models can be shared with others in several ways:

- as a *snapshot* that enables others to create a copy of the model at the time it was created and make changes to their copy
- as a *locked model* that enables others to create a copy of the latest version of the model and make changes to their copy
- as an *unlocked model* that enables users to access and make shared updates of the latest version of the model (all users of an unlocked model can roll back to earlier versions)

One advantage of providing a web-based model authoring tool is that the user is relieved of file and version management. Users need not concern themselves with transferring files in order to continue working on a different computer. Files are backed up automatically. Unlike desktop applications, users need not think about different versions of file formats since the server can automatically update internal files or databases. Giving others the opportunity to run, copy, or modify one's models is accomplished by sending them the appropriate URLs.

Another advantage of running our modelling tool within a browser is that it enables a tighter integration of associated resources. Libraries of micro-behaviours, tutorials, construction guides, lesson plans, and documentation can be HTML pages. *BehaviourComposer 2.0* can integrate these resources as tabs within the application. These pages can easily have "live" entities such as buttons for micro-behaviours or adding models and sub-models. It is particularly convenient to simultaneously read and access resources and build a model when *BehaviourComposer 2.0* is run in split screen mode. See Figure 1.

A web-based tool benefits from the tremendous world-wide efforts to improve the web and browsers. One example of this is cascading style sheets (CSS). CSS is used for all the user interface elements of *BehaviourComposer 2.0*. The styles can not only be changed to suit different tastes but also used to improve usability in special contexts such as mobile devices with small screens or visually impaired users.

AS A WEB 2.0 COMPONENT

Rather than build a large monolithic model authoring web application that also supports tagging, discussions, rating, usage summaries, and custom collections of creations we designed *BehaviourComposer 2.0* to be focused upon model authoring and to integrate well with Web 2.0 services provided by third parties.

The Modelling4All web site does not publish models. Instead models are always available via URLs containing unique global unguessable identifiers. These URLs provide privacy which is often desired for work-in-progress. No models are accessible unless their URLs have been published elsewhere. They become public only after a user references their model's URL in a blog, wiki, web site, email forum, or any other place where search engine spiders can find them.

The Modelling4All web site does not require a user to register and log in. Anyone can use it including spammers, vandals, and other troublemakers. However, since the site only produces unique URLs and does not publish anything created on the site, there is little harm they can cause and little that they can gain from doing so. This relieves us of much of the need to police user generated content for porn, copyrighted material, and other illegal material.

While we don't require login we still support a kind of authorisation that relies instead upon having unforgeable unique URLs. Only someone holding a session URL, for example, can access or change that session. At the Modelling4All site permission to use resources is not based upon identity but upon having obtained unique URLs. This approach builds upon the concepts of capability-based security. [8]

There is added value in supporting a minimal notion of identity. If the site can connect the identity of different authoring sessions then users could search for any of their past work. Collaborations are easier to manage if all parties agree to use the team's user identity. The Modelling4All site supports this weak sense of user identity. We rely upon unique unforgeable identifiers to represent users. The site does not know the identity of its users but can determine if the same user (or team sharing an identity) contributed to different sessions. The identity mechanism could be enhanced to give teachers access to the work of their students while the students only have access to their own work.

We believe that hosting models, sessions, and micro-behaviour edits in a private anonymous manner facilitates the integration of our services with third party services. The Modelling4All site hosts resources but does not make those resources accessible to those lacking the appropriate unique identifiers. Only if the holders of those identifiers make them publicly available on other web sites do the resources become available to the public. One of the problems with combining different Web 2.0 services is that each service typically has its own notions of identity and authorisation. The Modelling4All site does not contribute to this problem since it treats users and resources as anonymous.

We considered directly supporting discussion threads associated with saved models and instead have demonstrated how such threads can be hosted elsewhere (e.g. GoogleGroups). They can be embedded on the same web page as a Modelling4All model. The authors of the micro-behaviours and models hence decide where their creations will be discussed. We provide exemplars that point to the recommended practice for providing a discussion forum for micro-behaviours and models.

Social tagging has proved to be a useful way of categorising and organising large collections in a bottom-up fashion. The Modelling4All software provides buttons to add material to popular tagging sites such as del.icio.us. We are exploring stronger integrations with tagging sites using the site's APIs that would simplify the adding tags or using them for navigation. Because the tags are added to social bookmarking sites a folksonomy of micro-behaviours should emerge.

Users of Web 2.0 sites are guided by the ratings that earlier users have given to their pages. A rating facility will be added to the Modelling4All site so that users can find the highly rated models, micro-behaviours, and supporting materials.

In addition to ratings, users find it valuable to know the relative popularity of resources. We plan to add feeds that can be turned into user-friendly configurable gadgets (e.g. iGoogle gadgets) that can display lists such as models most frequently run, micro-behaviours most heavily used, and models most frequently copied and extended. These statistics will be produced in such a way that private models are not revealed.

We are investigating the possibilities of integrating forms for collecting data with model building guides. We constructed two guides which refer to forms that feed data into shared Google spreadsheets. The aggregated data is updated in real-time and can be made available in a classroom setting to the contributing students. It could also facilitate a teacher who wanted to collect the results each student obtained from running their model. The data collection, construction guide, and model authoring can all be integrated together.

AS A TEACHING TOOL

We have used the Modelling4All web site and tools in classrooms at Oxford University. We worked with instructors in producing micro-behaviour libraries tuned for modelling the desired subject matter and associated construction guides.

About 30 third year biology students constructed and ran a series of models exploring the dynamics of an epidemic spreading over a social network. In a single session they were able to build models with different kinds of networks and interventions. During the session they ran several variant models and each student contributed to a spreadsheet that automatically collected the reported results from a series of simulation runs.

Two groups of MBA and MSc students at the Oxford University Said Business School constructed and ran a series of Sugarscape models [9]. In a two-hour session most were able to build the models described in chapter two of the book *Growing Artificial Societies: Social Science from the Bottom Up*.

We have run a workshop with Oxford University academics and students where they built a predator prey model.

We have scheduled a session with economics students where they will use the site to build a series of models exploring network formation.

Very few of the biology or business school students had any computer programming experience, and yet they were able to build serious models in their field of study. They learned about the behaviour of a complex system in their subject as well as acquiring some understanding of the general process of model construction. They acquired what one of the faculty members we worked with calls *modelling literacy* – an understanding of how simulations work and the ways in which they are designed and constructed.

The students who built models of epidemics had an earlier session where they built a simple mathematical model of epidemics using other software. This modelled the dynamics of entire populations. When using the Modelling4All site they began with an agent-based model that mirrored this aggregate model. They then went on to explore the consequences of modelling a heterogeneous population. Agent-based models produce different dynamics of epidemics and outcomes for interventions than aggregate models do.

Another learning outcome is an appreciation for the differences between emergent phenomena and top-down control. The business school students, for example, saw how even very simple bottom-up models produced uneven wealth distributions that increased over time.

The micro-behaviours were designed to be engaging at different depths. A shallow understanding of a micro-behaviour is purely functional – what does it do and how can it be used. Some students were also concerned with how the micro-behaviours work and how they could be modified. The micro-behaviours by convention have a section explaining how they work. Additionally a good deal of effort went into making the source code readable by non-experts. In this way, the Modelling4All classroom session could be a first step towards learning to computer programming for building models.

AS A PUBLIC ENGAGEMENT IN SCIENCE TOOL

We are building a specialised version of the BehaviourComposer for the Royal Society 2010 Summer Science Exhibition. The challenge is to give visitors a taste of constructionist learning in only a few minutes. We will present visitors with the Epidemic Game Maker that contains a very simple model of an epidemic and a single intervention that players can take to stop the epidemic. Rather than expect the visitors to have the time and patience of browsing through a library of micro-behaviours for enhancing both the model and the game play we are providing customised buttons that enhance the game. Visitors can choose what entities to add to the model, what new behaviours, what things to measure, etc. by choosing which buttons to click.

ANIMATING MODEL EXECUTION IN *SECOND LIFE* AND MAPPING SERVICES

We have implemented a way to see the execution of a model inside of the 3D online virtual world *Second Life*. We provide a URL for each model that produces a stream of values for the position, orientation, scale, and colour of each agent in the model. The stream also records the creation or destruction of agents. Scripts within *Second Life* repeatedly read this stream and recreate the trace with *Second Life* objects representing the *NetLogo* agents. This is accomplished by running the model on our servers. By doing this model executions can be experienced in a social immersive manner.

We have also implemented a primitive manner for commands to be sent from within *Second Life* to alter a running model on our servers. We plan to add *Second Life* interface objects that play the same role as sliders and buttons within *NetLogo*. These interface objects will work by sending commands to the model running in *NetLogo* on our servers.

We have begun work on taking this same approach to visualising the execution of our models on mapping services such as *Google Earth* or *Google Maps*. The Modelling4All server code will be enhanced to produce standard mapping files [10] that can be loaded into a variety of mapping services. We plan, for example, to animate the execution of a model of pandemics on *Google Earth* enabling the viewer to see the spread of the virus on a globe they can view from any angle or height.

POTENTIAL PROBLEMS

One problem with providing a tool as a service is that users rely upon the service provider to maintain a robust stable service. This is a relatively minor problem when a large corporation such as Google or Microsoft provides the service but when it is provided by a small team in university project there is a greater concern. The problem is alleviated somewhat by providing a way to export one's data, by releasing the source code for the system, and by providing a way to copy models between different servers running the Modelling4All code.

Another problem is that the system is currently impossible to use without an Internet connection. Serious users can run a Modelling4All server locally to overcome this. A promising alternative we are considering is to integrate *Google Gears* [11] with *BehaviourComposer 2.0*. *Google Gears* is a browser plug-in that provides local storage. Using *Google Gears* the software could continue to work in some cases without a network connection, and then models will be uploaded when the connection is established.

We may discover difficulties with version management, especially for micro-behaviours. Software developers typically rely upon version control systems so that each build relies upon the appropriate version of components. In the *BehaviourComposer 2.0* references to micro-behaviours are by fixed URLs. The parties hosting those micro-behaviours can change the contents of the web pages, perhaps breaking models that relied upon the old contents. In contrast, Modelling4All models can be flexibly shared as a frozen version, as a read-only copy of the latest version, and as read-write access to the latest version. Perhaps we will discover that micro-behaviours need similar version control. It is possible to build web sites for micro-behaviours where the URLs specify the desired version policy.

Some are concerned about the public nature of HTTP traffic between the model maker and the servers. This could be alleviated by using a secure connection (HTTPS). Passwords could be automatically provided since here we are only trying to encrypt communications with the site for privacy reasons.

Models can be created and edited without the use of browser plug-ins. Many potential users with browsers lacking a plug-in will not, or are not allowed to, install a plug-in. Because BehaviourComposer 2.0 currently only supports *NetLogo*, the running of models requires either a plug-in to run Java applets or the prior installation of NetLogo (free for educational and research purposes). While the plug-in for Java applets is installed in the majority of browsers this remains a problem for a large minority of users. A sister project to Modelling4All has developed MoPiX [12] where the execution and animation of models is performed by the browser without the need for any plug-ins. BehaviourComposer 2.0 is, however, much more expressive than the equational programming supported by MoPiX.

There is concern that by giving users the freedom to choose the Web 2.0 tools that they integrate with their use the Modelling4All web site that the community will be much more fragmented than if a monolithic Web 2.0 site were provided instead. By providing guidance and exemplars we hope to guide community members towards shared tools.

The general issue underlying web applications is loss of control [13]. Students in a university or school have typically already lost control of the computers and software they use. This is more of an issue for long-term research projects using our services. Since Modelling4All is an open-source project, full control can be obtained by running the server locally. This might, however, fragment the community.

POTENTIAL USES AND FURTHER DEVELOPMENTS

The Modelling4All software currently only supports micro-behaviours constructed in *NetLogo*. The idea of browsing for program fragments that can be combined using a web browser can be applied to other modelling tools. If the language supports the expression of modular micro-behaviours then the server can generate complete source files that can be compiled and executed.

As users construct and run models, our servers accumulate data about how the site is being used. This data could be mined to focus further development efforts on those aspects that are most crucial. For example, analysis of the usage data may discover a common stumbling block where a significant fraction of users get stuck. We can then work to address this problem. Or we may discover that a very useful and powerful facility is being overlooked and we can then promote its use.

Data mining could also be useful to the Modelling4All community to acquire a crude level of self-awareness. Community members could learn what others are doing. Teachers could obtain summaries of what their students have built on the site.

To date we have focussed upon the educational uses of the Modelling4All site. We believe that researchers, journalists, and, policy makers could profitably use the site. It could also be valuable to the general public attempting to understand more deeply topical subjects such as causes of global warming or the spread of HIV. Some visitors to the site may only run a few highlighted models while some may follow the tutorials and construction guides to obtain a much deeper understanding of the underlying processes and mechanisms. We hope that our efforts to build and test the Epidemic Game Maker will address these concerns.

ACKNOWLEDGEMENTS

We are grateful to the Eduserv Foundation who has funded and supported this research. We want to thank the Oxford University Computing Services for their continued support. The JISC funded the Constructing2Learn Project that this work builds upon.

REFERENCES

Kahn, K., *Comparing Multi-Agent Models Composed from Micro-Behaviours*, Third International Model-to-Model Workshop, Marseille, France, March 2007

Kahn, K. *Building Computer Models from Small Pieces*, 2007 Summer Computer Simulation Conference, San Diego, CA, July 2007.

Wilensky, U., *NetLogo*, Center for Connected Learning and Computer-Based Modeling, Northwestern University, http://ccl.northwestern.edu/NetLogo/

North, M.J., Collier, N.T. and Vos, J. R., *Experiences Creating Three Implementations of the Repast Agent Modeling Toolkit*" ACM Transactions on Modeling and Computer Simulation, Vol. 16, Issue 1, pp. 1-25, ACM, New York.

Andreoli, J. and Pareschi, R., *LO and behold!* Concurrent structured processes, Proceedings of the European Conference on Object-Oriented Programming on Object-Oriented Programming Systems, Languages, and Applications, OOPSLA/ECOOP '90, Ottawa, Canada, ACM Press. Also published in ACM SIGPLAN Notices, Volume 25, Issue 10, Oct. 1990.

Google Web Toolkit, http://code.google.com/webtoolkit/

AJAX, http://en.wikipedia.org/wiki/AJAX

Capability-based security, http://en.wikipedia.org/wiki/Capabilities

Epstein, J. and Axtell, R., *Growing Artificial Societies: Social Science from the Bottom Up*, Brookings Institution Press and MIT Press, 1996

Keyhole Markup Language, http://en.wikipedia.org/wiki/Keyhole_Markup_Language

Google Gears, http://gears.google.com/

MoPiX, http://www.lkl.ac.uk/mopix/

Johnson, B. *Cloud computing is a trap, warns GNU founder Richard Stallman*, http://www.guardian.co.uk/technology/2008/sep/29/cloud.computing.richard.stallman, Monday September 29 2008 14.11 BST

Mathematics and Art: Thai Students' Design with The Geometer's Sketchpad

Asst.Prof. Dr.Krongthong Khairiree, *drkrongthong@gmail.com*
International College, Suan Sunandha Rajabhat University, Bangkok 10300, Thailand.

Abstract

The purpose of this research study is to explore the connection of mathematics, art and technology in the context of school mathematics. Data of the study were collected from 6 sample upper secondary schools in Thailand and the students were 16 year-old.

Research findings show that through mathematics project-based learning approach and the use of the Geometer's Sketchpad (GSP), the students were able to illustrate the connection of geometry patterns, functions and art.

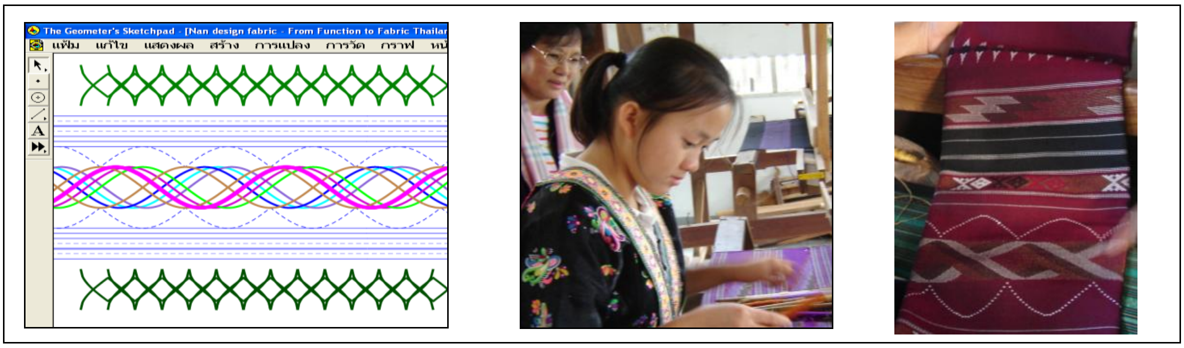


Figure 1. Thai students' mathematics and art designed, traditional looms and fabric

Based on the students' interviews they revealed that with GSP they were able to visualize and create graphical representations, which will enable them to develop their mathematical thinking skills, concepts and constructing mathematics knowledge. The students had fun in creating variety of graphs of functions which they can not do by drawing on graph paper. In addition, there are evidences to show Thai students' abilities in designing and connecting mathematics and art to real life outside the classroom and commercial products, such as weaving, patterning of broomstick, ceramics design, brooch and silver drinking bowl.

Figure 2. Trigonometric function designed for ceramic product

Keywords:

Thai Students' design, mathematics, art, the Geometer's Sketchpad, project-based learning, and commercial products

Introduction

The purpose of this research study is to explore the connection of mathematics project-based learning approach and the use of technology in the context of school mathematics in Thailand. Nowadays there are various information and communication technology which can be used effectively in mathematics class. One of them is a dynamic mathematics software named *the Geometer's Sketchpad (GSP).* GSP was introduced in Thailand since the year 2000. In year 2004 GSP was translated into Thai language and used widely in Thailand. More than 1,000 mathematics teachers were trained to use GSP as a tool in their mathematics classes.

Mathematics Project-based Learning

Mathematics project-based learning approach is employed in secondary schools in Thailand. This approach is one of the learning activities that shift away from the traditional classroom practices which are isolated, and teacher-centered. This approach emphasizes learning activities that are long-term, interdisciplinary, student-centered, and integrated with real world tasks to enhance learning. Students engage in project-based learning generally work in cooperative groups for extended periods of time, and seek out multiple sources of information (Oon - Seng Tan, 2003). According to Savin-Baden, M & Howell Major (2004) the project-based learning promotes collaboration among students, between students and the teacher, and between students and the community as well. Mathematics project-based learning approach provides opportunities for students to apply and integrate the content of different subject areas such as mathematics, arts and the use of technology to the production process. Thai Students have to design and develop their mathematics skills that relate to their daily lives. This idea was support by Masingila, J (1993), she said that it is her contention that the gap between doing mathematics in school situations and doing mathematics in out-of-school situations can only be narrowed after much is learned about mathematics practice in the context of everyday life.

Empowerment Through Tools: The Geometer's Sketchpad

The Geometer's Sketchpad is one of the dynamic mathematics software that provides opportunities for students to investigate and discover mathematics concepts in particular geometric patterns, functions and graph of trigonometric functions. GSP empowers students to use their abilities to create graphical representation, to enable them in developing their mathematical thinking skills, concepts, and understanding. In using GSP students learn by exploring, investigating and discovering.

From Mathematics Classroom to Commercial Product

We all know that mathematics is involved in every pieces of goods/product such as size, shape and pattern. Through mathematics project-based learning approach and the use of GSP, students are able to explore mathematics concepts in particular geometric patterns, functions and graph of trigonometric functions in more details and make mathematics learning fun and challenging. In addition, the use of mathematics project-based learning approach enhances students in exploring and creating mathematics content to commercial product such as weaving, patterning of broomstick, ceramics design, brooch and silver drinking bowl.

Research

This research is a case study that emphasizes mathematics project-based learning approach and the use of the Geometer's Sketchpad (GSP). The main purpose of the study is to explore mathematics art and GSP using project-based learning approach in the context of school mathematics in Thailand and connecting to real life outside the classroom and commercial products.

Research Process

Data of the study were collected from sample schools in Education school year 2006 until July 2008 from Srisawat Witayakarn School in Nan province, and Kalang Nakorn Lampang School in Lampang province Thailand. The 16 year-old students were at Upper Secondary level. Mathematics teachers in the sample schools implemented GSP as a tool in their classes. The students used GSP in mathematics projects and worked together in small groups of three to four members.

In this study, the researcher collected data from various resources such as classroom observations, students' project reports, newspaper, and commercial products. Semi-structured interviews with the teachers and students were also conducted.

Research Questions

1. In what way GSP can be used as a tool in designing mathematics, art and mathematics project-based learning approach?
2. What are the effects of mathematics project-based learning approach using GSP towards students' attitudes in mathematics?

Research Findings

The research findings were based on the students' mathematics project reports, the researcher's classroom observations, and interviews. The students' works show how they used GSP as a tool in constructing functions, trigonometric functions and geometric patterns. The students implemented their works designs in mathematics to the commercial product.

Research Question 1: In what way GSP can be used as a tool in designing mathematics, art and mathematics project-based learning approach?

The summary of research findings and the examples of students' works are described as follows.

1) From Functions to Nan Fabric Designed by Students of Srisawat Witayakarn School, Nan Province.

Thai textile weaving was designed by the students of Srisawat Witayakarn School, Nan Province Thailand. The students applied knowledge on graph of functions and arts to create the patterns of Thai textile weaving design. The students enjoy using GSP to create the graphs of trigonometric function especially graph of sinθ, cosθ, arcsineθ, and arcos θ. They constructed more than 50 graphs of trigonometric functions and came up with the beautiful fabric design. Examples of the students' designs are shown in Figure 3, Figure 4, and Figure 5. The students then provided the names of their designs, Figure 3 is Kleau-Klun design, Figure 4 is Samukkee-Klomklew design, and Figure 5 is Klun-Obe-Kao design.

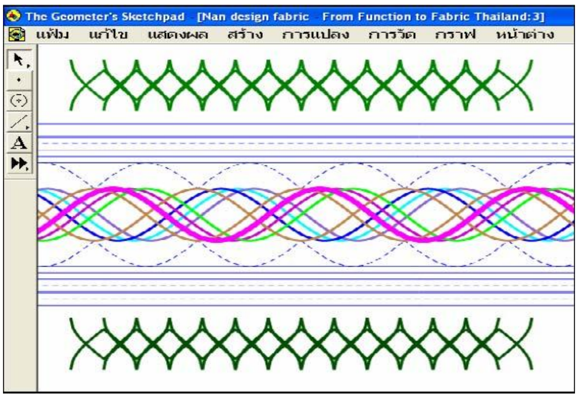


Figure 3. : Kleau-Klun design

The following functions are some examples of functions and trigonometric functions of Kleau-Klun design. The examples functions are shown as follows.

$$f(x) = b \times \cos^{-1}(|x+h|) + a;$$

$$g(x) = \cos^{-1}(|x+j|) + s;$$

$$h(x) = \cos^{-1}(|x+j|) + s;$$

$$s(x) = \cos^{-1}(|x+h|) + a;$$

$$w(x) = |\cos(x)|$$

Samukkee-Klomklew design is shown in Figure 4 below. The example functions and trigonometric functions of Samukkee-Klomklew design are shown as following:

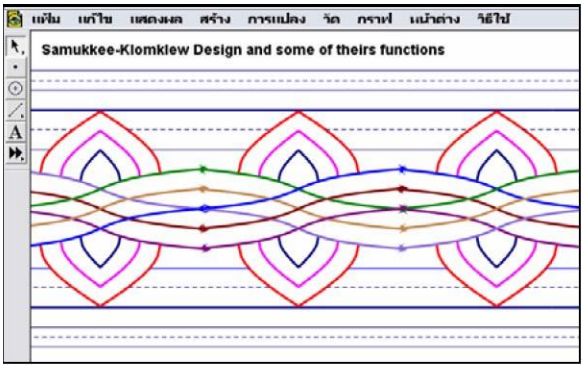


Figure 4. Samukkee- Klomklew design

$$f(x) = \log(x + \sqrt{1 + x^2}) \qquad g(x) = \cos^{-1}(|x|) + 2.4$$

$$h_2(x) = -f(x); \qquad h(x) = 1.5\cos^{-1}(0.5|x|) + 2.6$$

$$f_2(x) = f(x) + 1; \qquad q(x) = 2\cos^{-1}(0.33|x|) + 2.8$$

$$g_2(x) = f_2(x) + 1; \qquad q_2(x) = h_2(x) + 1;$$

$$r_2(x) = q_2(x) + 1; \qquad h_4(x) = -g(x) + 2;$$

$$u_4(x) = -h(x) + 2; \qquad h_5(x) = -q(x) + 2;$$

2) From Trigonometric Function to Ceramic Designed by Students of Kalang Nakorn Lampang School, Lampang Province.

There are quite a number of Ceramic Factories in Lampang province. The students were assigned to do mathematics project which relevance with a famous product in their province. They employed knowledge learned on trigonometric and transformation in mathematics classes to design a pattern of ceramic. The following examples show how students used GSP in their mathematics project.

Ceramic design 1:

The students designed a pattern of ceramic and used GSP to construct functions ***f(x) = cos (a. x)*** as following:

- Using Graph menu to construct a parameter ***a***
- Enter function ***f(x) = cos (a. x)*** by
  - Choosing ***New Function*** from the *Graph menu*;
  - Enter ***a*** by clicking on parameter ***a*** on the sketch and click on keyboard ***x*** in the *New Function* dialog box then click *OK;*

- Select f(x) = cos (a. *x*) on the screen, then choose *Graph menu* and select *Plot Function*;
- Graph of f(x) = cos (a. *x*) appears on the screen as figured on your right;
- Construct Point ***A*** on graph of f(x) = cos (a. *x*)
- Using point ***A*** as a center and construct a circle with radius about 1 cm, and construct point B on this circle;
- Construct line CD parallel to x-axis
- Reflect point B across line CD to create point B′.

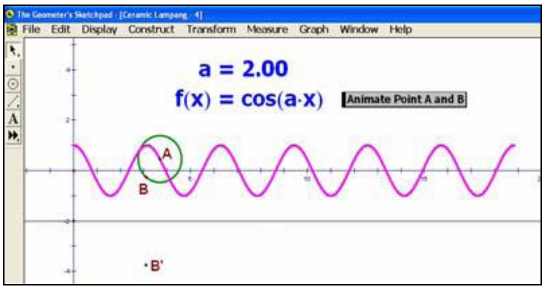


Figure 6. Graph of f(x) = cos (a. x)

- Select point ***B*** and point ***B′***, and Turn on *Trace points* menu
- Animate points ***A*** and ***B***.
- The traced pattern of point ***B*** and point ***B′*** will appear the same as figure on your right.

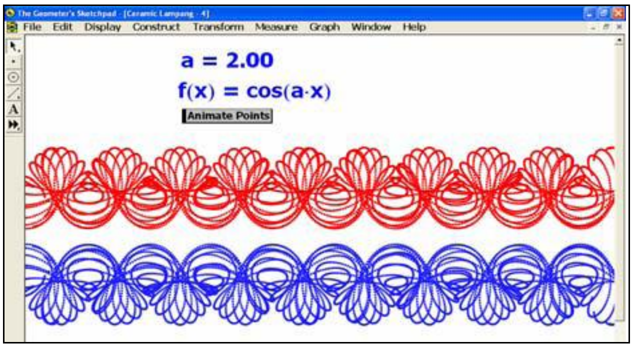


Figure 7. Pattern designed for Ceramic Product

Ceramic design 2: Sketch and investigate using new sketch of GSP

The students designed a pattern of ceramic and used GSP to construct functions ***f(x) =sin(a.x). cos (b.x) + sin (b.x)*** as following:

- Using Graph menu to construct parameters ***a*** and ***b*** and plot function *f(x) =sin(a.x). cos (b.x) + sin (b.x)* by choosing **Plot New Function** from *Graph menu*
- Construct Point ***A*** on graph of *f(x)*
- Using point ***A*** as a center and construct a circle with radius about 1 cm, and construct point ***B*** on this circle;
- Construct line ***CD*** parallel to x-axis
- Reflect point ***B*** across line ***CD*** to create point ***B′***.

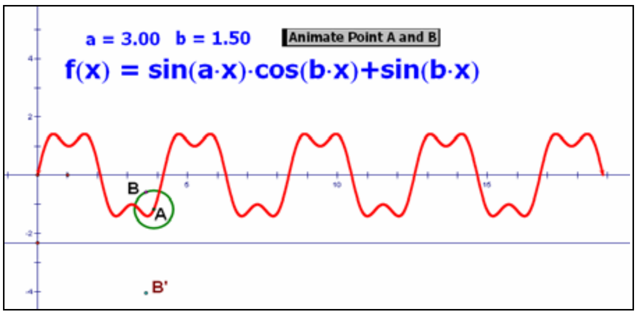


Figure 8: Graph of f(x) =sin(a.x). cos (b.x) + sin(b.x)

- Turn on Trace points for point ***B*** and point ***B′***,
- Animate points ***A*** and ***B***.
- The traced pattern point ***B*** and point ***B′*** will appear the same as figure on your right.

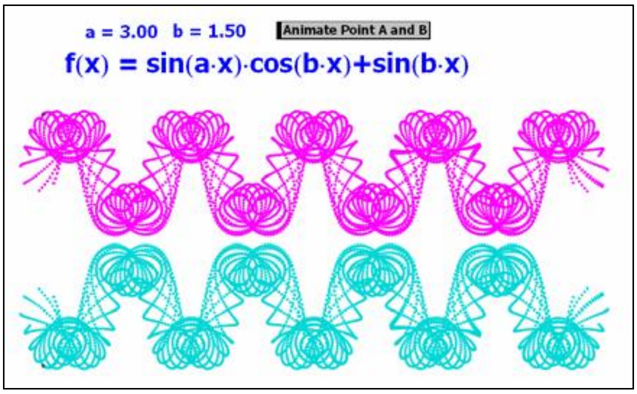


Figure 9. Pattern designed for ceramic product

3) From geometry in mathematics to brooch and silver drinking bowl designed by Students of Srisawat Witayakarn School, Nan Province.

The students of Srisawat Witayakarn School, Nan Province were assigned to do mathematics project which relevance to a product to be used in daily life. They applied knowledge learned on geometry, translation, reflection and symmetry topics in mathematics to create the patterns of brooch and design patterns for silver drinking bowl. The example of geometric and functions of design and photographs of these products are shown in Figure 10 and Figure 11.

The examples of functions of the design of brooch are as follows:

$f(x) = 1.5\sin^{-1}(x)$; $g(x) = -1.5\sin^{-1}(x-2)$; and $h(x) = -\sqrt{-(x-1)^2+4} - 2$

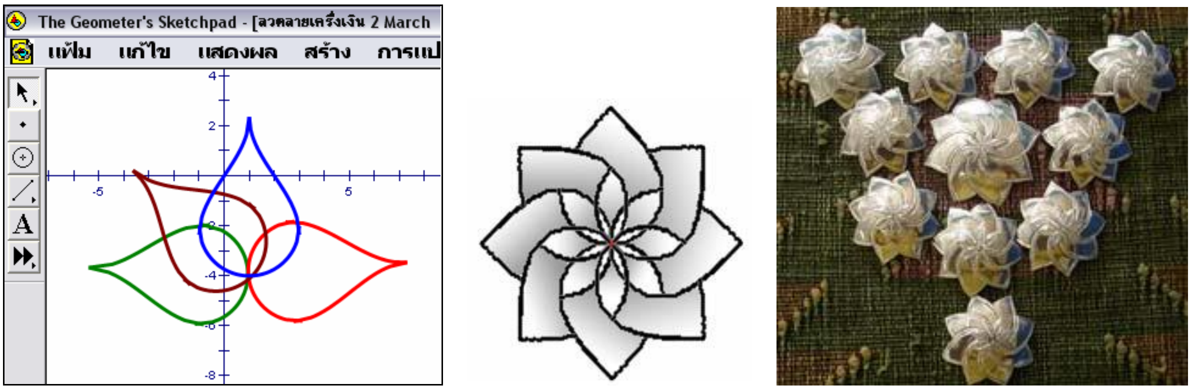


Figure 10. Pattern designed for brooch

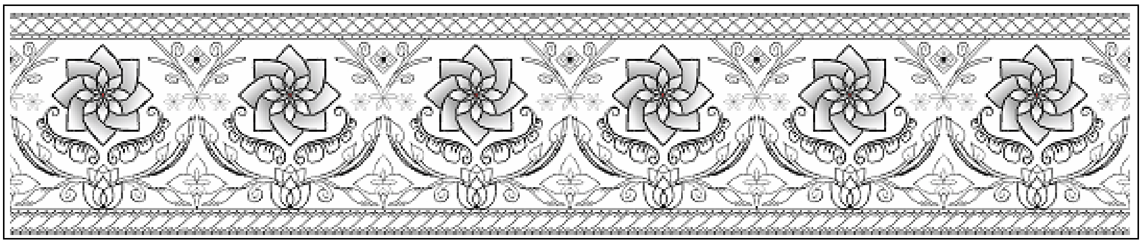


Figure 11. Pattern designed for silver drinking bowl

Figure 12. A silver drinking bowl commercial product

Research Question 2: What are the effects of mathematics project- based learning approach using GSP towards students' attitudes in mathematics?

Based on the findings, the researcher found out that the students in the sample schools can exhibit their relational understanding in mathematics which involves understanding structures and connections within concepts (Skemp, 1978). Results from students' semi-structured interview reveal that the after the students did their projects they liked to learn mathematics and they have more understanding on mathematical topics. The students can explain, know what to do and knew why they had to do.

In addition, the students revealed that with GSP they were able to visualize and create graphical representations, which will enable them to develop their mathematical thinking skills, concepts and understanding. The students had fun in creating variety of graphs of functions which they can not do by drawing on graph paper. The students explained that it was fun in learning mathematics by this method. It was better than work only from the exercises in mathematics textbook. Based on these evidences the students have acquired a positive attitude toward mathematics.

Conclusion

All illustrations are evidences to show the students' abilities in connecting mathematics and art with GSP to real life outside the classroom and commercial products, such as weaving, patterning, and ceramics design. It is clear that, with GSP students are able to visualize and create graphical representations to enable them to develop their mathematical thinking skills, concepts and understanding. This project can further facilitate learning among students establish useful connections in mathematics, particularly to life in the world outside classroom and to develop commercial products.

References

Bennett, D. (2002). *Exploring geometry with the Geometer's Sketchpad: Revised for use with Version 4.* CA: Key Curriculum Press.

Cuoco, A. & Curcio, F.(Eds.) (2001), *The roles of representation in school mathematics. 2001 Yearbook.* National Council of Teachers of Mathematics. Reston: VA

Kemp, Jerrold E. (1971) *Instructional Design: A Plan for Unit and Course Development.* Belmont, California: Fearon/Lear Siegler.

Khairiree, K. (2007). *Exploring Ethnomathematics with the Geometer's Sketchpad (GSP):* Thai Students' Weaving Projects. In Wei-Chi Yang, Tilak de Alwis & Jen-Chuan Chuan (Eds). Proceeding of the Twelfth Asian Technology Conference in Mathematics. (pp. 105-114). Mathematics and Technology, LLC: USA.

Masingila, J. (1993) *Connecting the Ethnomath of carpet layers with school learning*. In ISGEm Newsletter (International Study Group on Ethnomathematics) Volume 8 Number 2, May 1993.

Oon-Seng Tan (2003) *Problem-Based Learning Innovation: Using Problems to Power Learning in the 21st Century*. Singapore: Seng Lee Press

Savin-Baden, M. & Howel Major, C. (2004). *Foundations of problem-based learning.* England: SRHE and Open University Press Imprint

Scher, D.I & Rasmussen, S. (2009). *Spark! Teaching Mathematics with The Geometer's Sketchpad*. CA: Key Curriculum Press Emeryville

Skemp, R.R. (1978). Relational understanding and instrumental understanding. *Arithmetic Teacher*, 26(3), 9-15.

Students' mathematics project report (2008). *From mathematics and art to Nan: Silver commercial products*. Srisawat Witayakarn School, Nan Province Thailand.

Students' mathematics project report (2007). *Using GSP to design ceramic products*. Kalang Nakorn Lampang School, Lampang Province, Thailand.

Students' mathematics project report (2005). *From Function to Nan:Thai Weaving Textile* Srisawat Witayakarn School, Nan Province Thailand.

van Hiele, P.M., & van Hiele-Geldof, D. (1958). A method of initiation into geometry. In H. Freudenthal (Ed.), *Report on methods of initiation into geometry*. Groningen: Wolters.

[1]

[1] The author gratefully acknowledges the research support provided by International College, Suan Sunandha Rajabhat University, Thailand. I wish to extend my gratitude to the Principals, teachers and students of Srisawat Witayakarn School, Nan province, and Kalang Nakorn Lampang School, Lampang province Thailand for their contribution to this research. The author wishes to extend her gratitude to Mr. Steven Rasmussen and Key Curriculum Press for their support and contribution to this research.

Agent-Based Modeling and Complex Systems Concepts as Useful Prior Knowledge in Secondary School Science Students' Understanding of Evolution

Hal Scheintaub, *hscheintaub@govsacademy.org*
Governor's Academy, and Scheller Teacher Education Program, MIT

Eric Klopfer, *klopfer@mit.edu*
Scheller Teacher Education Program, MIT

Daniel Wendel, *starlogodaniel@gmail.com*
Scheller Teacher Education Program, MIT

Abstract

We have shown that agent-based modeling and complex systems concepts can be practical and effective tools for middle school science and high school physics learning (Klopfer, 2009). This study builds on that experience, extending the use of agent-based models and complex systems simulations from the physical sciences to the secondary school biology curriculum. To this end, we designed and implemented a sequence of interactive off-computer and agent-based model building activities that would enable students to experience and experiment with the mechanisms that drive the emergence of large-scale global phenomena from smaller scales of agent actions and interactions. By including the act of model building (in many cases through computer programming) in the learning experience, students were able to participate in the full spectrum of interactions with simulations (Klopfer, 2009). Our objective here was to determine whether these modeling activities would provide the right prior knowledge (Schwartz, 2007), that when coupled with appropriate scaffolding and learning resources, would help students overcome misconceptions and build a robust understanding of evolutionary processes

Analysis of early student programming strategies revealed a need for instruction in systematic design and problem solving skills. The trial and error process employed by most students was inefficient, but it was good enough to help them discover an important complex systems principle; that in some systems, small code (behavior) changes could yield dramatic changes in the resulting systems-level patterns. To assess how the concrete act of programming affected the understanding of emergence, students were asked to explain how patterns in their simulations happened. While it has been reported that a coherent understanding of complex systems eludes most students (Jacobson & Wilensky, 2006), more than 85% of the students in this study were eventually able to make connections across scale by describing how population patterns emerged from individual agent behaviors and interactions. Post-activity written and oral discussions revealed that the simulation experience proved to be the right prior knowledge to account for evolution.

Keywords

simulations, complex systems, programming, prior knowledge, evolution

Theoretical Framework

Both off and on-computer simulations have been used in classrooms to explore how complex dynamic systems evolve over time (Scheintaub, 2009). These simulations can support new forms of classroom interaction and can serve to catalyze the engagement with complex ideas, fostering the kinds of higher-order thinking and problem-solving skills that are called for in science and mathematics learning (Wilensky & Stroup, 1999; Collella, 2001).

Agent-based computer models are especially well suited for student inquiry and science learning (Jacobsen & Wilensky, 2006). Pre-built simulations can provide students with accessible visualizations, immersive learning environments, and opportunities to analyze data from virtual experiments. However, they do not give students the intimate knowledge that comes from building simulations, an important part of scientific practice; nor do they afford them the freedom to express their ideas or express their interest in a given phenomena, an important motivator of the sustained learning necessary for meaningful science learning (Edelson & Joseph, 2001). By including the act of model building (in many cases through computer programming) in the learning experience, students are able to participate in the full spectrum of interactions with simulations (Klopfer, 2009), thereby providing experiences that can serve as the 'right prior knowledge' to support future learning (Schwartz, 2007). The algorithmic thinking involved in programming emphasizes processes rather than facts (Cohen & Kanim, 2007) and programming provides students with a means of expression that is precise and compact (Sharin & diSessa, 1993). Subsequent use of the simulations proceeds from the experiential knowledge of its construction, providing useful prior knowledge upon which students can build new understandings.

Though recent advances in science and medicine, along with an abundance of observations and experiments over the past 150 years, have reinforced evolution's role as the central organizing principle of modern biology (Ayala, 2008), teaching evolution has proven to be extremely difficult and many misconceptions abound (Caldwell et al, 2006). Interactive lessons have been found to help students recognize many misconceptions and understand why evolution is considered one of the strongest of scientific theories (Flammer, 2006). To be most effective, interactive lessons need to be linked to and build on the foundation of a unifying idea. For evolution learning, the complex systems concept of emergence may be such a theme. It can facilitate the understanding of evolution by providing a unifying theme across scales of time, space, and size (Holland, 1996; Solé & Goodwin, 2000). Therefore, we designed and implemented a sequence of interactive off-computer and agent-based model building activities that would allow students to experience and experiment with the mechanisms that drive the emergence of large-scale global phenomena from smaller scales of agent actions and interactions. We wanted to study whether these activities could provide the right prior knowledge, when coupled with appropriate scaffolding and learning resources, to help students overcome misconceptions and build a robust understanding of evolutionary processes. Here we report on the development, implementation and outcomes of such a series of activities into ninth grade biology classes; and the effect of these activities on the ninth graders' understanding of evolution is documented.

Context and Methods

We used *StarLogo TNG (Klopfer et al., 2009)* as the modeling and simulation tool for many of these activities. *StarLogo TNG* has proven to be an effective tool for introducing agent-based modeling and programming to secondary school students (Klopfer & Scheintaub, 2008; Klopfer, 2009). *StarLogo TNG* builds on the tradition of Logo-based languages designed to facilitate the development and study of simulated systems in classrooms. It includes a graphical programming language which lowers the entry barrier to programming (Begel, 1996), and the game-like 3-D world helps provide the motivation necessary for students and teachers to experience the power of programming.

We report here on a sequence of programming and simulation activities and the resulting outcomes in terms of conceptual changes, as manifest in the curriculum of freshman biology classes over a two-year period in an independent school outside of Boston, MA. Additional implementation details for these activities can be found at “Biology Curriculum” at http://education.mit.edu/drupal/starlogo-tng/learn. Data sources in this study included: a) Multiple classroom observations and analyses b) Student-generated curricular products c) Student blogs, discussions and self-assessments and d) pre- and post-implementation student surveys.

Students used *StarLogo TNG* to program the actions of agents and then observed system-wide patterns in the complex-systems simulations they had built. In the first three activities, (See Table 1) students learn to build models and use simulations. They acquire a set of skills that are technical and general. Simulated experiments provide concrete examples of the abstract concept of emergence. While the activities are grounded in relevant content, that content is not their learning objective. The content objective doesn't come until the fourth activity where students apply their experience with simulations and their working knowledge of emergence to their efforts to understand the fundamentals of evolution.

Activity 1- Introduction to Programming in a game-like Environment: The first programming activity, Vants (Virtual Ants), introduces students to the idea that a program can define a set of rules of behavior for an agent (Figure 1). They are empowered to modify a program and motivated to create visually pleasing results. Through engagement in this activity, students get some first-hand experience with the important complex systems concept of emergence. They see how small changes in rules for agent behaviors can lead to large changes in observable systems patterns.

Figure 1. Small changes in initial conditions and movement of agents give very different looking patterns. The pattern in the figure on the right is generated from a very small variation in the code that produced the original pattern on the left

Activity	**Emergence – from simple rules to observable systems patterns**	**Biology Concepts**
Vants (Virtual Ants)- Programming and simulations introduction	Patterns in the landscape emerge from simple stamp and turn rules for agents	Interactions Agent/Environment Agent/Agent Environment/Agent
Population Growth -	Exponential and logistic growth	Birth rate, death rate,

off-computer, and in TNG	curves emerge from simple rules for birth and death	Recognizable patterns of population growth
Community Interactions – students build a sustainable bunny /carrot community	Cyclical, out-of-phase population graphs emerge from actions of and interactions among agents	Community as a set of interacting populations Predator/prey relationship Basics of ecosystem dynamics
Fishpond - a Sequence of TNG simulations for fundamentals of microevolution	Change in a population's gene frequency emerges from randomness and slight differences in genetically determined actions of individual agents in the community.	Population as the unit of evolution Evolution is the change in gene frequency in the population

Table 1. Simulations, Emergence and Curricular Concepts

Activity 2 – Population Growth: This section begins with a tabletop simulation where pennies are given a set of simple rules for reproduction (Collella et al., 2001). Students plot growth curves of their penny populations and discuss how rule changes might affect their results. When students build those rule changes into a *StarLogo TNG* simulation (Figure 2), they see how exponential and logistic growth curves emerge from rules for birth and death. As in the previous activity they see how small changes in those rules affect their population growth patterns.

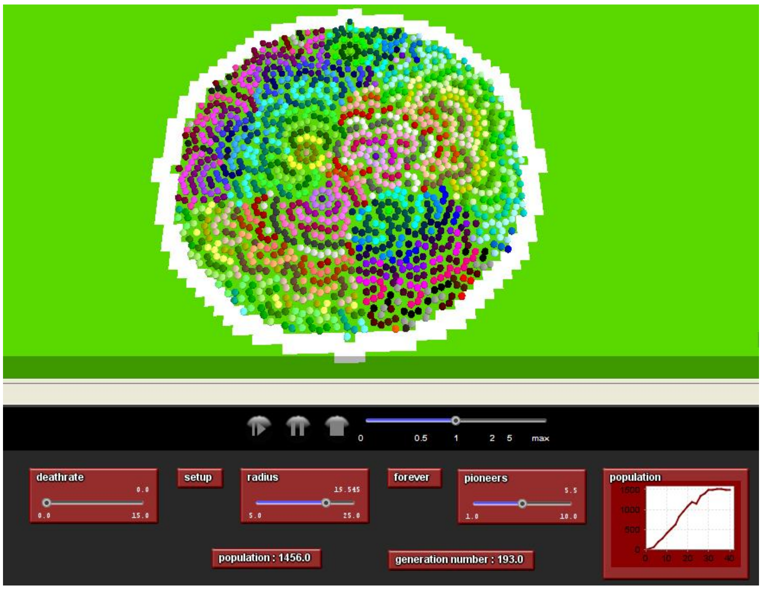


Figure 2. Students use sliders to change variables that affect population growth dynamics. Discussions reveal how the area of the circle could represent available resources.

Activity 3- Community Interactions - Building a sustainable producer-consumer community: This weeklong, self-paced programming unit builds on experience and the students' developing concept of emergence. Students build a virtual ecological community of producers (carrots) and consumers (bunnies), within which they can run experiments, by working through a set of instructions that progress from highly structured to more open-ended. Students program essential behaviors for bunnies (move, eat, reproduce, die) and carrots (spread seed, die). They focus on individuals and interactions; no population scale behaviors are defined. While the student’s attention was centered on the actions of agents in the programming panes (Figure 3) of their simulations, macro- scale organization emerged in the 3D world on the screen (Figure 4).

As in the previous two activities, they see how their programmed individual micro-scale actions and interactions caused ordered patterns to emerge at the population scale. Interacting in and with simulations in Activities 2 and 3 gave students an immediate experience with the concept of emergence that could be codified and brought to bear later in their study of evolution (Schwartz, et al, 2005).

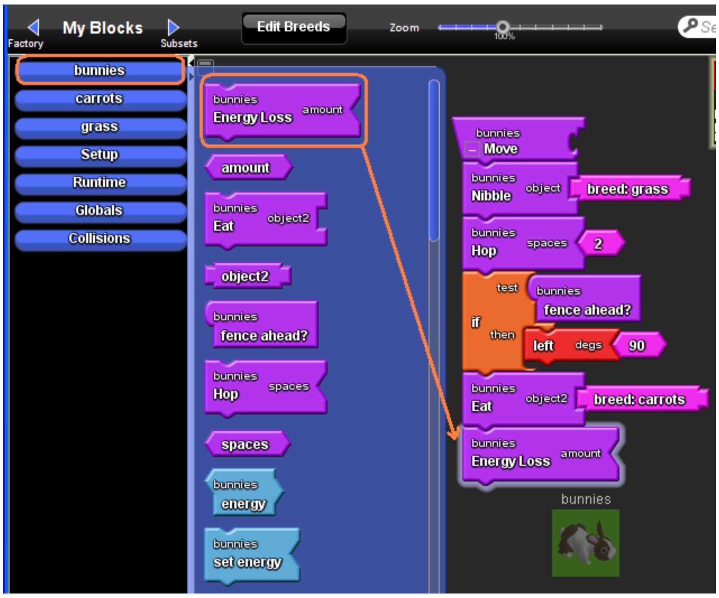
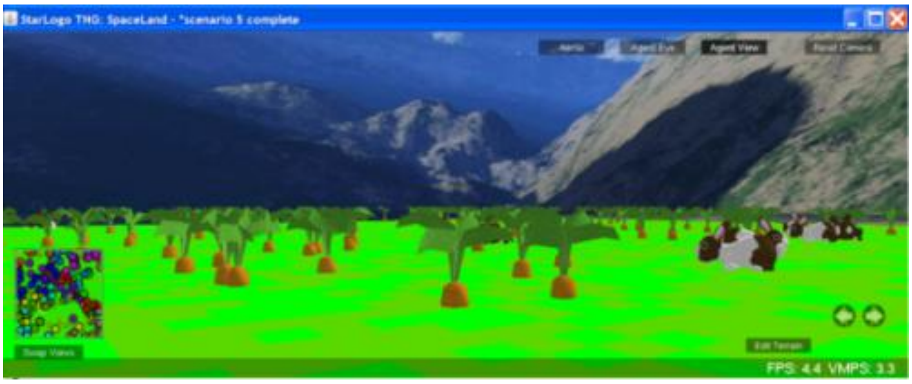


Figure 3. Screen shot of graphical programming language of StarLogo TNG used by students to build community interactions simulation.



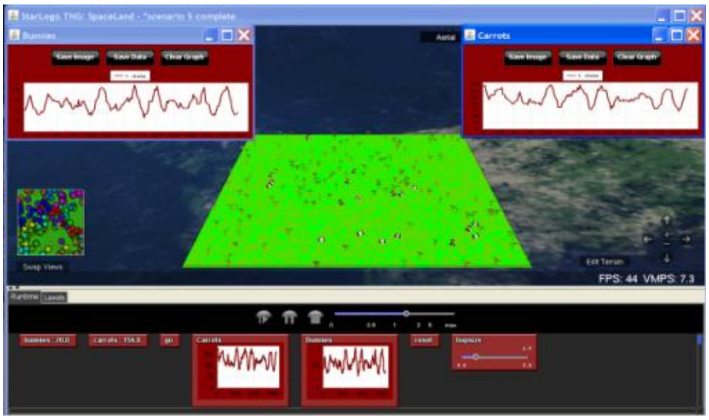
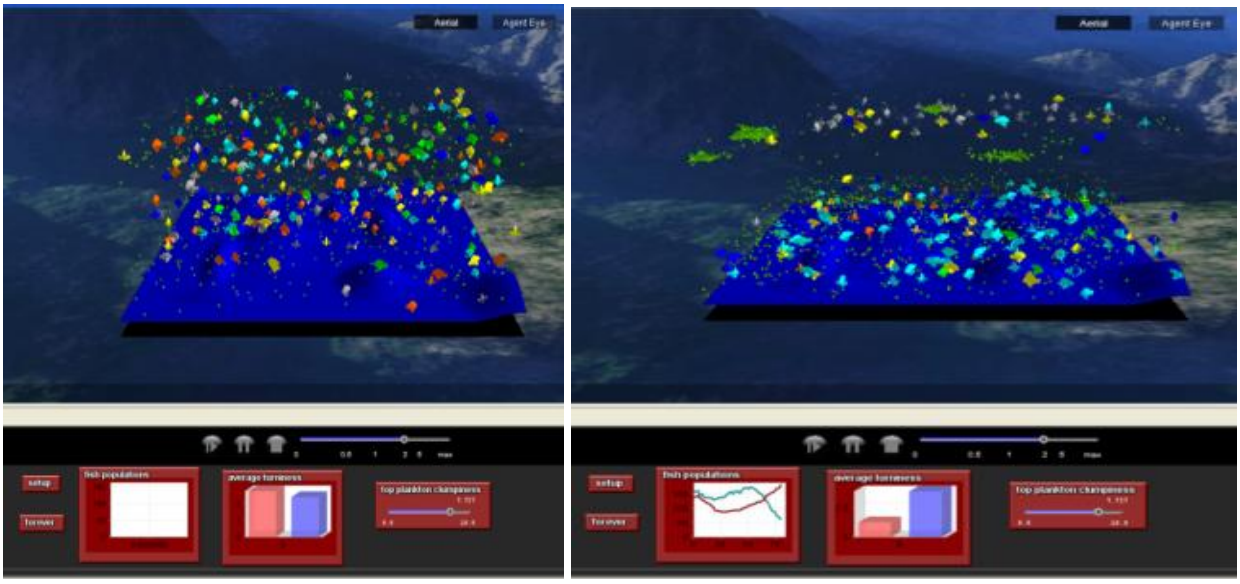


Figure 4. Screen shots of community interactions simulation from the first person (top) and third person (bottom) perspective. The two views let the students experience the simulation as both a participant and an observer.

Activity 4 – Evolution: Misconceptions about evolution are widespread (Caldwell, et al, 2009). The fact that evolution is a unifying theme in modern biology makes dealing with those misconceptions critical. To address one misconception, “individuals evolve”, we developed a series of agent-based evolution simulation activities that leveraged the students' concrete programming experience and complex systems learning from the previous activities. The evolution simulations focus on fish in a pond. The first of the series is essentially the same as the bunny and carrot simulation students built earlier. It serves as a review of how out-of-phase producer-consumer population cycles emerge from actions and interactions of fish and plankton in a pond. In the second of the series, two varieties of fish in an initial population separate over time, one variety lives near the top of the pond, one near the bottom. There are two varieties of plankton in the pond. One type of plankton grows best near the surface, one near the bottom of the pond. Experience looking to agent interactions to explain population patterns helps students eventually see that one variety of fish prefers the plankton that grows near the surface; the other prefers the bottom growing variety. The concept of evolution, as the change in the genetic composition of a population is introduced in the third simulation. In this simulation there is only one type of plankton and an initial population of fish of different colors. Fish and plankton are set up in random locations on the screen. Randomness is programmed into the movement of the fish. Fish color is a programmed ‘genetic’ characteristic that is passed on to offspring, but serves no survival value. Students recognize that the now familiar dynamics that lead to population cycling cause the elimination of certain color fish from the population when the food supply is low. When food is scarce, the survival of a particular color fish depends randomly on its position near plankton. They see evolution occurring, in this case without natural selection. In the final simulation (Figure 5) color is linked to a trait that affects survival, and evolution proceeds through the natural selection and the random processes seen in third simulation.



Figures 5a and 5b

Figure 5. One diverse population of fish (5a) evolves into two (5b). Fish at the top of the pond are better suited to forage on clumps of plankton while the characteristics of those fish near the bottom are better suited to eating the dispersed plankton found there.

Misconceptions about evolution often center on reasons why organisms change over time (Caldwell, 2006). To avoid such misconceptions we kept students actively involved with the concrete actions occurring in and the appearance of their simulations. They performed concrete tasks that helped them focus alternatively on agent behaviors and population patterns. Instructor-aided analysis was integrated into these activities to help students see how the evolution of the fish populations emerged from:

- randomness - in initial conditions and movement.
- genetic variation - as manifest in recognizable programmed behaviors of those fish.
- selection - that derived from understandable interactions between fish and plankton.

Data and Results

Both the process of student inquiry and the results of their learning were subjects of this study. A teacher/researcher and outside observers made multiple classroom observations over the entire school year. Observational data as well as written test results, student-generated curricular products, blogs, self-assessments and surveys served as the data sources. The data detail how the guided work, reported on here, allows students to make connections across scale, account for the emergence of population patterns in biological systems, and avoid the misconceptions so often seen in student explanations of micro-evolution.

Activity 1- Analysis of early student programming strategies revealed a need for instruction in systematic design and problem solving skills. This need was met with additional scaffolding and structured worksheets in subsequent activities. The trial and error process employed by most students was inefficient, but it was good enough to help them discover an important complex systems principle; that in some systems, small code (behavior) changes could yield dramatic changes in the resulting systems-level patterns. Students appreciated the freedom to experiment and reacted with excitement when one of their small changes in code changed observable patterns on their screens. The enthusiasm generated by this activity helped establish StarLogo as favorite classroom learning tool introduce modelling as a mode of learning.

Activity 2 - This basic population growth activity began as a physically concrete penny placement and counting exercise, but its analysis and extension to StarLogo made it mathematical and virtual. The bridges from physical model to computer simulation, and concrete counting to abstract mathematical analysis were designed to help establish modeling as a mode of learning for students. The personal nature of post-activity responses spoke to the effectiveness of this strategy. For example one student wrote, "The shape of my population graph is an "S". My graph compares to the other groups because it is exponential which means it has a steep rising point, and the whole graph is logistic, the "S" shape". Another student's insight into the modeling process is revealed here. "Some modifications that might enable this model to better represent living systems could be that pennies could get eaten by predators, and not (re)produce always the maximum amount."

Activity 3 – From the results of Activity 2, the authors knew that students could use modeling and model analysis for learning. They were ready to see how the concrete act of programming impacted students' understanding of emergence. Written explanations of how the cycling of simulated rabbit and carrot populations came about, was the test of that understanding. This was a difficult test because a coherent understanding of complex systems eludes most students (Jacobson & Wilensky, 2006). Detailed analysis of student explanations revealed that more than 85% of the students in this study made connections across scale by clearly describing how the cycling of the two populations emerged from individual agent behaviors and interactions. They noted that after the bunnies ate most of the carrots, most of the bunnies died. For the community to continue a few bunnies had to be lucky enough to be in the region where a few carrots remained. The high percentage of students making connections across scale is notable because often students have difficulties seeing the mechanisms that drive the emergence of large-scale global phenomena from smaller scales of interacting agents (Chi, 2005, Wilensky & Resnick, 1999). Analysis of explanations revealed the specifics of some of those difficulties. One student was not able to see the reciprocal nature of the interactions rabbit and carrot populations. He wrote, "The rhythm of carrots and bunnies is primarily controlled by the bunnies." Another student did not feel that the idealized simulation represented the real world. "Nature is nature! There is no systematic pattern as there is in our StarLogo simulations. In nature, what happens, happens. It is sporadic, and there is no pattern." These results are significant not only because the programming/simulation sequence yielded such a high percentage of students who demonstrated a good functional understanding of emergence, but also for exposing the strong beliefs about how the world works that endured in a few students even after participating in the activity.

In addition, the activity stimulated connections from the science classroom across experience. Highlights of a discussion in one freshman class revealed complex systems connections similar to those made by university students (Goldstone & Wilensky, 2008). One freshman student noted, "The pattern of supply and demand and the carrots and bunnies is relatively the same." Another added, "When there are a lot of bunnies the supply of carrots is low and demand is high." In that same discussion there was an exchange about the intentional, or unintentional nature of the interactions. When one student said, "It's all about multiple organisms working together." Another agreed, but added, "They subconsciously help each other out because they are only trying to help themselves, but they benefit each other by helping themselves"

Activity 4 – In Activity 4, students applied their experience with simulations and their working knowledge of emergence to the understanding of evolution. Fish and plankton in a pond was the setting of these simulations. Key aspects of the evolutionary process were developed systematically in a sequence of simulations. The series started at a familiar place, the interdependence of producer and consumer populations. In the second simulation, variety in the plankton population made that interdependence more complex. Inherited variability in the fish population was included in the third simulation and natural selection in the fourth. Worksheets and oral instructions helped guide students' attention to the actions of agents. Students ran simulated experiments, collected data and answered questions. These tasks made up the concrete experience base that students would go to when developed a systematic explanation an explanation for evolution.

One of the programmed variables in the fourth simulation was the behavior of a fish after eating plankton. In the initial population, some fish turned, while others continued swimming straight. Plankton behavior varied, too. Top-growing plankton grew in clumps, while plankton near the bottom were dispersed. Plankton growth and fish turning behaviors were passed on to offspring. Fish that turned after eating were better suited to survive at the top. Straight swimming fish did better near the bottom. Eventually, the initial fish population evolved into two populations, a turning population at the top and a straight swimming population near the bottom.

Post-activity written and oral discussions revealed that students had a good working knowledge of the micro-evolutionary processes involved in the simulated evolution. Students knew that some fish had a trait that directed them to turn more than others. Some fish had a trait that directed them to swim faster than others. They saw that after a time the initial mixed population evolved into two distinct populations.

In a guided debrief, the class was able to explain how this evolution occurred. A student scribe recorded highlights of that discussion. Quotes from those notes are used in this section. Students began their analysis with a concept developed initially in the carrot and bunny simulation of Activity 3. In that simulation they had noted that bunnies survived in regions of the simulation where there were carrots. Being trained to look at producers helped them see that, “Top food (plankton) is clumped, and the bottom is spread out.” This observation is important because it shows that the students realized that even though it was the fish that evolved, variability in the environment helped drive that evolution. Then their attention turned to the fish. They saw that the, “Fish separated into two populations; top and bottom. This was because fish survived where there was food. It *appeared* that they followed their food source.” Though it is only implied here, it is clear that the students recognized that fish survived where there was food that they could to eat. The feeding behavior came first, then the survival. The simulation helped students avoid the common misconception that the ability to eat a particular type of food evolved to help the animal survive. When pressed for specifics of the behaviours that affected survival, they replied, “Top fish turn more and bottom fish turn less.” When asked why, they reasoned, “Surviving fish are better adapted to their environment. Turning after eating works where plankton is clumped. Swimming straight and fast works where plankton is spread out.” The specific path of their reasoning was recorded as follows. “If you are turny and you are lucky enough to be on top you can survive and reproduce. If you are turny and you are on the bottom you most likely die. If you are fast and you are lucky enough to be on bottom you can survive and reproduce. If you are turny and you are on the top you most likely die. Over time, this results in fast fish on bottom, and turny ones on top. This is because survivors pass on their genes to their offspring.” This explanation reveals an understanding of the concept natural selection grounded in the concrete experience of their simulation. In addition, they see that to survive the fish have to be more than just fit; it has to be lucky enough to be where their inherited hunting strategy matches the food supply. This understanding of the role of randomness in the complex process of evolution is often overlooked in textbook presentations of evolution.

There are many other more naive interpretations that could be given for the evolution of the fish population, the most likely being that hunting strategy evolved to help the fish survive. The authors believe that students do not use this line of reasoning because of their experience building and using simulations. They know that code determines the behavior of agents and that the behavior of agents determines population dynamics. So they look to coded behavior for a cause of evolution. Understanding the programmed workings of the simulation leads to right reasoning and an avoidance of misconceptions. It proves to be the right prior knowledge for understanding the mechanisms of microevolution.

The debrief proceeded in a biologically and structurally sound cause and effect manner from plankton patterns through individual fish behavior to fish population patterns. It included fitness and randomness. The students saw evolution occur in their simulations, and they used their observations to account for that evolution. In contrast, students who are not able to see evolution occurring in simulations may use assumptions rather than observations to account for it. They may assume that evolution proceeds in a particular direction because individuals try to adapt (Caldwell, 2006). Students who use simulations have an opportunity to apply scientific reasoning to evolution, without the need to rely on previous assumptions.

Significance

This pilot shows that the integration of programming, simulations, and complex systems principles into freshman biology courses is both possible and promising. Students transferred the complex systems principle of emergence from a simple game-like system, into population growth models, through community interaction simulations, to the dynamics of the evolutionary process. They used their knowledge of micro-level events to explain complex macro-level phenomena like the out-of-phase cycling in a producer-consumer biological community and to provide plausible mechanisms for mysterious events (Jacobsen & Wilensky, 2006) like evolution.

In addition, the sequence of lessons employed here supported di Sessa's (2000) contention that programming can enable the individual discovery of important (biological) patterns by converting those abstract patterns into spatial and visible ones accessible to students. The activities proved to be the right prior knowledge to counteract a widespread misconception and support a sound functional understanding of evolutionary processes. Such a sequence may have wide application in the secondary school biology curriculum. To this end, we are developing "preparation for future learning assessments" to work backwards to identify the aspects of these activities that most directly prepare students to learn (Schwartz, 2007) the processes behind and the significance of evolution.

References

Ayala, Francisco, et al., 2008. *Science Evolution and Creationism*, third edition, National Academy of Sciences. http://www8.nationalacademies.org/onpinews/newsitem.aspx?RecordID=11876 Retrieved, 01/15/2010

Begel, A. (1996). *LogoBlocks: A graphical programming language for interacting with the world* (Thesis). Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science.

Caldwell, R, et al. (2006). *Understanding Evolution*,

http://evolution.berkeley.edu/evosite/evohome.html, Retreived, 01/15/2010.

Chi, M. T. H. (2005). Commonsense Conceptions of Emergent Processes: Why Some MisconceptionsAre Robust. *Journal of Learning Sciences*, 14, 161-199.

Cohen, E. W.,& Kanim, S. E. (2007). *Algebraic difficulties in physics*. Retrieved July 1, 2009 from http://spacegrant.nmsu.edu/NMSU/2004/cohen.pdf

Collella, V, Klopfer, E, Resnick, M. (2001). *Adventures in Modeling*, Teachers College Press, New York, N.Y.

diSessa, A.A.(2000) Changing Minds, *Computers, Learning and Literacy.* MIT Press, Cambridge, MA, p 17.

Edelson, D & Joseph, D (2004). Motivatinig Active learning: A Design Framework for Interest-driven Learning.http://www.designbasedresearch.org/reppubs/edelson-joseph.pdf. Retrieved 01/31/10

Flammer, Larry, (2006). The Evolution Solution: Teaching Evolution without Conflict, *American Biology Teacher*, 68, (3), 1-7.

Goldstone, R.L. & Wilensky, U., (2008). Promoting Transfer by Grounding Complex Systems Principles. *Journal of the Learning Sciences*, http://www.informaworld.com/smpp/title%7Edb=all%7Econtent=t775653672%7Etab=issueslist%7Ebranches=17 - v1717, (4), 465 – 516.

Holland, John. (1996) Emergence: From chaos to Order, Perseus Books, New York, N.Y.

Jacobson, M. & Wilensky, U. (2006). Complex systems in education: Scientific and educational importance and implications for the learning sciences. *Journal of the Learning Sciences, 15*(1), 11-34.

Klopfer, Eric & Hal Scheintaub (2008). StarLogo TNG – Science in Student-Programmed Simulations, *Proceedings of the 8th international conference for the Learning Sciences*, paper 388.

Klopfer, E, Scheintaub, H, Huang, W, Wendel, D, and Roque, R. (2009). The Simulation Cycle - Combining Games, Simulations, Engineering and Science with StarLogo TNG. *E–Learning*, 6, (1), 71-96.

Scheintaub, H, Klopfer, E, Scheintaub, M, Rosenbaum, E. (2009). Complexity and Biology – Bringing Quantitative Science to Life Science Classrooms. in Linking Mathematics and Biology in High School Classrooms, In Press for Fred Roberts, editor, Untitled.

Schwartz, D. L., Martin, T., & Nasir, N. (2005). Designs for knowledge evolution:Towards a prescriptive theory for integrating first- and second-hand knowledge. In P. Gardenfors & P. Johansson (Eds.), *Cognition, Education, and Communication Technology* (pp. 21-54). Mahwah, NJ: Erlbaum.

Schwartz, D.L., Sears, D.A., & Chang, J. (2007). Reconsidering prior knowledge. In M. Lovett and P. Shah (Eds.), *Thinking with Data*. Mahwan, NJ: Erlbaum.

Sharin, B., & diSessa A. A. (1993). Dynaturtle revisited: Learning physics through collaborative design of a computer model, *Interactive Learning Environments, 3*, 91 – 118.

Solé, R & Goodwin,B. (2000). *Signs of Life: How Complexity pervades Biology*. Basic Books, New York, N.Y., USA.

Wilensky, U. & Resnick, M. (1999). Thinking in Levels: A dynamic systems perspective to making sense of the world. *Journal of Science Education and Technology, 8,* 3-18.

Wilensky, U. & Stroup, W., 1999. Computer Support for Collaborative Learning, *Proceedings of the 1999 Conference on Computer Support for Collaborative Learning*, Article No. 80, http://portal.acm.org/citation.cfm?id=1150320, Retrieved, 01/15/10.

The Imagination Toolbox: Designing and Using Science Simulations and Games with StarLogo TNG

Eric Klopfer, *klopfer@mit.edu* & **Hal Scheintaub**, *hscheintaub@govsacademy.org*
MIT Scheller Teacher Education Program, Massachusetts Institute of Technology
& The Governor's Academy

Introductory description and overall goals

Much is made of the skills of the Millenials or "digital natives". They play games, create content online and network with each other. While many develop a full set of skills on their own or with their peers, others do not. There are opportunities for educators to help more students develop these skills, capture the interest of a generation, and connect interest in technology to the academic realm. Building this connection is the goal of The Imagination Toolbox (IT), an initiative built on the simulation and game-authoring tool, StarLogo TNG. StarLogo TNG builds on the tradition of Logo-based languages designed to facilitate the development and study of simulated systems in classrooms. This latest version provides several key advances including a graphical programming language and a game-inspired three-dimensional world. The blocks-based programming of Starlogo TNG puts programming aspects of the design and construction cycle into reach for most students and teachers, while the 3D world provides both motivation and perspective on simulation.

Method

This workshop will introduce participants to the IT approach that combines authoring and using games and simulations. This approach is based on the Simulation Cycle, a process that combines a scientific methodology with engineering design and play. Workshop leaders will demonstrate a variety of activities that provide different entry points into the Simulation Cycle. These activities are of two types. In Type 1, Game → Simulation, students explore a simulation by playing a pre-built game based on an academic topic. In Type 2, Simulation → Game, students explore a simulation by using the scientific inquiry processes (such as modifying parts of the model) and then applying the knowledge gained to solve game-like challenges based on that model. Workshop participants will work with partners do many of these activities, experiencing those activities as secondary school science students might experience them. In addition, leaders will offer the theoretical underpinning for these activities and consider practical concerns for their implementation.

Expected outcomes

The workshop assumes no prior knowledge of StarLogo TNG. It is open to novices, but will provide additional insights to those who have prior experience. It will start with some introduction to StarLogo TNG, including the programming paradigm and interface. It will then walk participants through one of the sample science units such as High School Physics (mechanics and projectile motion), Biology (evolution) or Middle School Earth Science.

Keywords

keyword; science, simulation, complex systems, games

Non-subject based education – methods of developing competences

Marta Kőrös-Mikis, *korosmarta@gmail.com*
Hungarian Association of Teachers for Informatics

Experiences of in-services teacher training courses

The latest versions of the 2008 legislation on public education and the 2007 National Core Curriculum called for non-subject based education for 25-50 pc of the classes in grades five and six (pupils aged 10-12). In the course of these classes emphasis needs to be placed on the elaboration of interdisciplinary topics encompassing several cultural domains employing constructivist theory and co-operative methods. According to the ministerial decree, only those pedagogues may conduct in these non-subject based classes who have completed a special, 120-hour accredited in-service teacher training course. The Ministry of Education has prepared an on-line, alternative, sample frame curriculum suited for non-subject based classes and textbook publishers have also provided methodology guidelines and corresponding workbooks. Our association (www.isze.hu) as one of the organizers of the preparatory courses, has published a manual too, containing relevant key information and practical suggestions.

Figure: Mirror project – Photos from our in-service teacher training courses

Initially, most pedagogues greeted the new educational policies and the 120-hour mandatory in-service teacher training with aversion. Many of them failed to grasp the need for having to obtain a new "degree" to teach the very same students whom they have hitherto taught mathematics, biology or history. During the training we employed the same co-operative methods aimed at acquiring new knowledge as expected of them in regard to their pupils. This involved a great deal of group effort, pair work and playful activities. On the basis of the questionnaires, we may conclude that our courses were a success, the teachers enjoyed participating and received much practical assistance from the lecturers.

In the course of the first academic year (2008-2009), the Hungarian Institute for Educational Research and Development (www.ofi.hu) conducted an on-line survey concerning the methods used in non-subject based classes. It turned out that co-operative techniques dominated over the whole class methods. ICT is an excellent opportunity for the interdisciplinary elaboration of curriculum topics. Computers and other digital means support the teaching of all subjects.

Keywords

non-subject based education, co-operative methods, Logo pedagogy, in-service teacher training

Intrinsic and extrinsic perspectives in 3d constructions

Maria Latsi, *mlatsi@ppp.uoa.gr*
Educational Technology Lab, School of Philosophy, University of Athens

Chronis Kynigos, *kynigos@ppp.uoa.gr*
Educational Technology Lab, School of Philosophy, University of Athens

Abstract

In this paper we report findings of a classroom research aiming at shedding light on 12 year-old students' construction processes as they worked with a 3d Logo / Turtle Geometry environment which we call 'MachineLab Turtleworlds' (MaLt). Illustrative examples of students' work are provided trying to examine in particular: a) the way the students used the software's functionalities of changing viewing angles throughout the construction processes, b) the interplay between the turtle metaphor and space visualisation through various viewing angles, c) the interplay between the perception of figures considered in relation to different viewpoints and in relation to their geometric properties. The analysis of the results brings in the foreground the dialectic relationship between the way the available viewing angle manipulation tools were used and the construction strategies followed, within the framework of particular tasks. It seems that the 3d space was experienced by the students through two distinct perspectives: an intrinsic and an extrinsic one. When the students were focusing on navigating and orientating the turtle, the 3d space was experienced through an intrinsic perspective, according to which the simulated space was viewed from inside, through the turtle's viewpoint. In this case the use of the body-syntonic metaphor was critical but yet not the same as conventional 2d turtle geometry. In contrast, when the focus was shifted from the management of the turtle's spatial movements to the construction of a graphic object, the 3d space was experienced through an extrinsic perspective, from the view point of an external observer who looked at the figural results of turtle's movement.

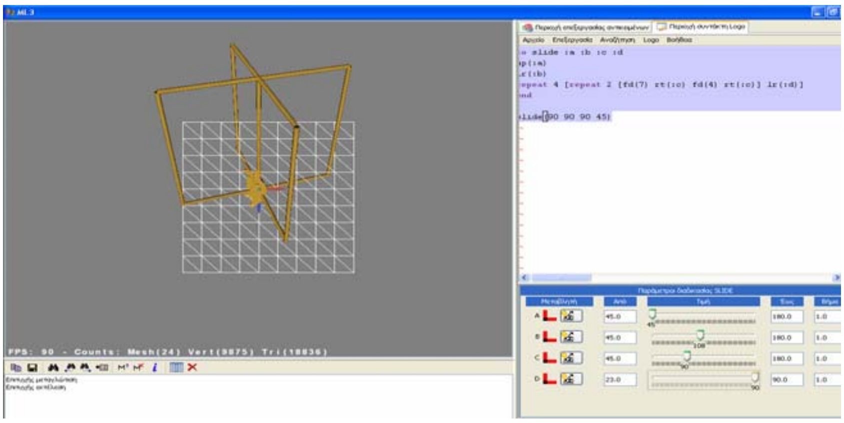


Figure 1: The 'MaLT' microworld

Keywords

3d logo, constructionism, space visualisation, construction processes and viewing angle

Theoretical background

The contribution of technology in the teaching and learning of geometry is perceived to be strongly linked with interactivity, multiple interlinked representations including symbolic ones, dynamic manipulations and dynamic visualisations (Laborde et al., 2006). However relatively little research has been carried out on the way the above distinct characteristics of digital media can be exploited so as to engage students in meaningful investigations especially as far as 3d geometry in concerned. Aiming at challenging students' intuitions and ideas concerning spatial visualisation and thinking (Presmeg, 2006, Arcavi, 2003) we developed a set of microworlds and a set of activities adopting a constructionist theoretical perspective (Kafai & Resnick, 1996). A distinct feature of the microworlds was that they were 'half-baked' (Kynigos, 2007), i.e. incomplete or buggy digital artefacts that the had to investigate how they work and to change and fix them.

Our pedagogical aim was to engage the students in navigating the turtle to construct graphical objects by means of Logo programming and dynamic manipulation of procedure variable values. Drawing with the turtle does not follow the conventions used so as to represent an object in a visually acceptable way e.g. depicting its external features in perspective. In contrast, it is based on analysing the visual characteristics and on explicitly addressing the geometric properties and structure of the object to be constructed along with the spatial relations of its elements. Students have to search for ways to reconceptualise 3d objects in terms that can be explained to the turtle through logo commands according to the distinct geometrical nature of turtle geometry. Turtle geometry is based on a different geometrical system to those usually associated with the learning of geometry and it has been characterised as differential by Papert (1980) and as intrinsic by Abelson & diSessa (1981). It's considered as differential as a given geometrical state of the turtle is fully defined by its relation to the turtle's immediately previous state. In a similar vein it is characterised as intrinsic in the sense that there is no need to refer to places outside the turtle's immediate vicinity when deciding on an input to a procedure to change turtle's state. Turtle geometry has been a field of long debate especially in relation to a)the degree children use the geometrical ideas embedded in it and b)the degree turtle geometry's intrinsic nature is related to other non-intrinsic geometries (Kynigos 1993). Researches seem to conclude that carefully designed Logo- based microworlds is an effective medium in offering rich mathematical experiences and encouraging inductive inferences from personal experiences which in turn can invite engagement in deductive thinking (Clements & Sarama, 1997, Kynigos, 1993). Moreover it seems that the turtle metaphor can be used to extend children's learning to include non-intrinsic geometry (Kynigos, 1992).However extending Turtle Geometry in 3d space offers a new perspective and raises new issues related to the way turtle metaphor is put to use and the way deeply rooted intuitions about experiencing space and locomotion can be exploited so as to make sense of geometric notions (Kynigos & Latsi, 2007).

MaLT is a 3d version of 'Turtleworlds', a turtle geometry environment with dynamic manipulatin tools for variable procedure values (Kynigos et al, 1997). MaLT integrates various tools for manipulating dynamically both 3d objects (in the sense of Turtleworlds) and the viewing angle of the simulated 3d space. In particular, the viewing angle manipulation tools are designed to support visualisation processes, e.g help users discriminate between what is represented and how it looks, imagine all possible diagrams attached to a geometrical object etc. Although the functionalities available are designed so as to help users abstract 3d geometrical objects' properties and structure and to acquire a sense of space and shape constancy, this should not be taken for granted. In contrast, it should be a point of concern that needs investigation as it seems that three-dimentionality and animation present special challenges and there is little support that they are beneficial to graph comprehension (Tversky, 2005). The new visual representations available must be considered not only in respect to their appearance but also from the perspective of their use, in respect to the manipulations that can be carried out on them as well as on the way other linked representations may be affected (Morgan et al., 2009). Viewing the images in the computer screen as 'signifiers' mediated by the conventional system in which they are created it is interesting to investigate how viewing an object from different perspectives would affect the actual process of getting to know the object and its spatial properties in the simulated 3d space. In particular building upon earlier researches (Kynigos et al., 2009) that took place within the framework of a European research and development project (ReMath, 2005-2009), the aim of our research was to investigate: a) the way students used software's functionalities of changing viewing angles throughout the construction processes, b) the interplay between the turtle metaphor and space visualisation through various viewing angles and c) the interplay between the perception of figures considered in relation to different viewpoints and in relation to their geometric properties

The Computational Environment

MachineLab is a programmable environment for the creation and exploration of interactive virtual reality simulations developed within the ReMath project (ReMath, 2005-2009). MaLT was conceived as a constructionist microworld environment within MachineLab that extends the 'Turtleworlds' Turtle geometry to 3d geometrical space. Thus an extension of Logo commands in 3d space is provided including the two conventional types of turtle turns (Reggini, 1985): 'UPPITCH/DOWNPITCH n degrees' ('*up/dp n*') which pitches the turtle's nose up and down and 'LEFTROLL/RIGHTROLL n degrees' ('*lr/rr n*') which moves the turtle around its trunk/vertical axis. However the distinct feature of MaLt is that the logo-based Turtle Geometry is integrated with the dynamic manipulation of interactive graphical representations - a functionality characteristic of Dynamic Geometry Environments'. In particular, the dynamic manipulation tools available can be divided in two categories:

a) dynamic manipulation of graphical figures by means of sequentially changing the variable values of the programs they create them through the use of specially designed variation tools (see the 1 dimension variation tool on the bottom right corner of picture 1).

b) dynamic manipulation of the viewing angle of the 3d space: a) by using toolbar's buttons where the user can pick among 3 default views (front, side, top-down) b) by manipulating through mouse a specially designed vector tool, called the active vector, where the user can define either camera's direction or camera's position

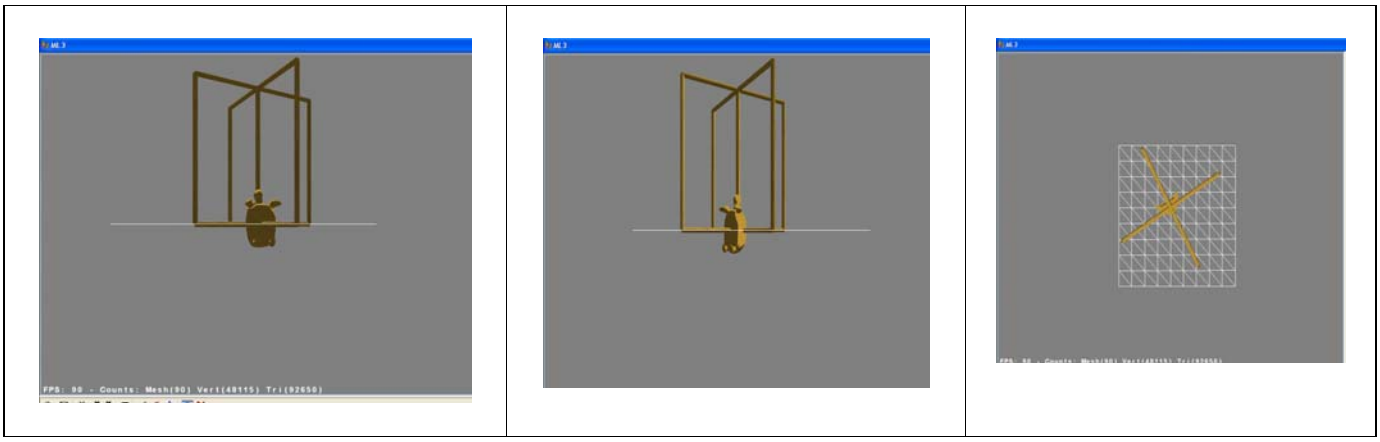


Figure 2: The 3 default views

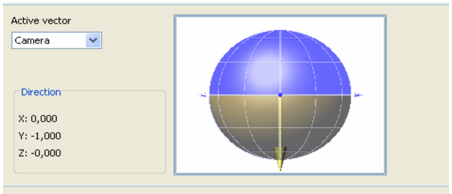


Figure 3: The active vector tool

Methodology

The work reported in this paper is part of a design-based experiment in the sense that Cobb et al. (2003) have described it. The research took place in the 6th grade of a public primary school in Greece. The class had totally 16 teaching sessions with the experimenting teacher over two months. The tasks were designed to bring in the foreground issues concerning the mathematical nature of 3d geometrical objects and how they may be dynamically manipulated and transformed in mathematically meaningful ways. In particular we divided the activity sequence in two phases and we developed for each one of them a strand of two tasks. In task 1 the students were asked to navigate the turtle in such a way so as to simulate the take-off and the landing of an aircraft. In task 2 the students were asked to construct rectangles in at least two different planes of the graphical space of MaLt simulating adjacent walls of a virtual room. In the second strand of activities students experimented with half-baked microwords. In particular, in task 3 students were asked to use the 1dimention Variation Tool to control and experiment with the three variables of the half-baked microworld ‘movedoor’ that corresponded to different turtle turns so as to create the simulation of door opening and closing. The procedure was designed to have on purpose more than the variables needed. The students had firstly to decide which the role of each variable was and which values could be given to them. Then they had to build upon the half-baked microworld so as to develop a procedure that creates the simulation of a door opening and closing with the least possible variables.

```
to movedoor :a :b :c
uppitch(:a)
leftroll(:b)
repeat 2 [forward(3) right(:c) forward(2)
right(:c)]
end
```

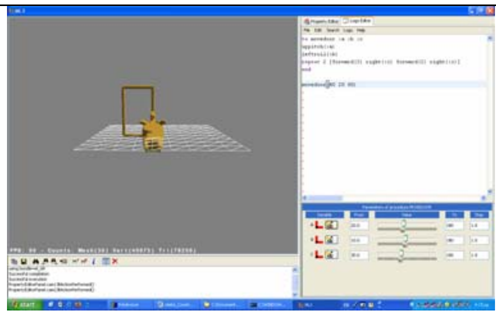


Figure 4: Simulating the opening and closing of a door and the respective logo code

In task 4 the students were asked to use the 1dimention Variation Tool to control the four variables corresponding to turtle turns in the ‘half-baked’ microworld ‘Revolving’ so as to create the simulation of a revolving door (see figure 5). The procedure was designed to have on purpose more than the variables needed. Students had firstly to decide which the role of each variable was and which values could be given to them. Then they had to build upon the half-baked microworld so as to develop a procedure that creates the simulation of a revolving door with the least possible variables. Finally the students were asked to extend the procedure of the revolving door in order to create a simulation of the fan of a watermill.

```
to revolving :a :b :c :d
up(:a)
lr(:b)
repeat 4 [repeat 2 [fd(7) rt(:c) fd(4) rt(:c)] lr(:d)]
end
```

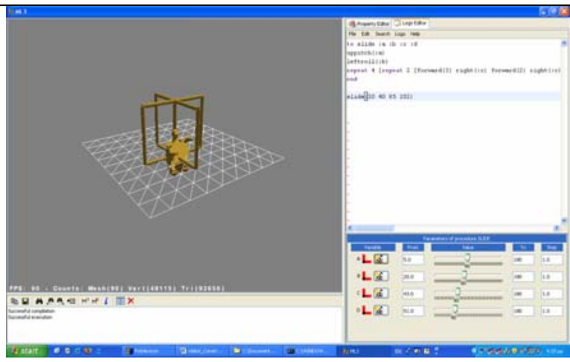


Figure 5: Simulating a revolving door and the respective logo code

In order to describe pupils' learning trajectories as they happened in real time we adopted a participant observation methodology while the main corpus of data included video-recorded observational data, researchers' observational notes as well as the sorting and archiving of the corpus of pupil's work on and off computer. Data were categorized in clusters of specific critical episodes that do not represent some quantifiable entity but are chosen to represent clearly the kind of activity that was going on in a specific time in the classroom. The analysis is still in progress while the results presented here are based on the work of one focus group focusing on the way viewing angle manipulation tools were used during the construction processes.

Construction processes through different perspectives

The analysis of our results has shown that students' construction processes could be divided in two categories: construction processes through an intrinsic perspective and construction processes through an extrinsic perspective, depended on the focus and the way the simulated 3d space was experienced. This division reflects the two dominant perspectives people take on space (Tversky, 2005), an external one when they observe space and manipulate objects in it and an internal one when they explore an environment and when they navigate in it.

Construction processes through an intrinsic perspective

Drawing with the turtle requires the formation of essentially novel methods of spatial orientation, where the reference point is not the position of the user's body but the turtle's body, relative to which the entire system of orientation may change. Thus during the construction process a critical point is the visualisation of space thought the different viewing angles of the turtle. Viewing space through the turtle's viewpoint is indirect and involves reflection about the turtle's viewing angle and orientation. However the degree of directness or not depends on the degree the body syntonic metaphor is applied, in other words on the degree personal knowledge of movement in space is applied to the turtle's action. The more body-syntonic turtle's motion is the more direct space visualisation and orientation become. Whereas in 2d Logo environments it is postulated that learning is aided by projecting one's own knowledge and experience of movement through space to the movement of the cursor on the screen (Fei, 1987), in the 3d logo environments body- syntonicity is questioned (Kynigos & Latsi, 2007). However the results of the present research underline the importance of syntonising one's body with the 3d turtle – vehicle of motion in the 3d simulated space. During the construction processes of task 1 students preferred 'flying' the turtle along the z axis, that gave the impression of depth, and at a plane vertical to the display plane defined by the 2d computer screen. Moreover they kept on working on the default front view (although slightly slanted through the use of the active vector manipulation tool) even though they didn't have a clear representation of turtle's journey (see picture 4).

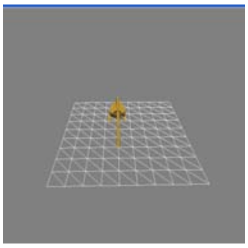
	Up(45) Fd(2) Dp(45) Fd(2) Dp(45) Fd(2) Up(45)

Figure 6: Simulating the taking-off and landing of an aircraft along the Z axis and the respective Logo code

It seems possible that children preferred flying the turtle along the z axis (that gave the impression of depth) while viewing the simulated 3d space from the default front view since this way they could more easily coordinate the various frames of reference (Wickens, 2005) present. In order to drive the turtle in a body syntonic way students had to coordinate the following frames of reference: a)the ego frame, defined in terms of the orientation of the trunk or location of the observer b)the display frame defining in terms of the standard way of referring to things presented in the computer screen, where the right/left up/down directions are fixed, c)the world frame defined in terms of the fixed directions of 'up' and 'down', as a result of the gravitational effect and d)the vehicle frame of reference, defined in terms of the place and orientation of a moving entity, here the turtle. Flying the turtle along the z axis, the orientation of the vehicle of the motion, the turtle, coincided both with the orientation of students' body in the lived in 3d space and with the standard way of referring to the orientation of information on the computer screen. Students' comments corroborate this result. When asked why they preferred this kind of flight they replied: '*If we wanted to turn turtle right or left, we could see from our hands. If we wanted to turn it right, let's say, we would think where our hand is and we would send it to the right. '.*

It is interesting that children are focusing more on body syntonicity while not being sidetracked by the visual effects even though only an inclined line - corresponding to the 'taking off' of the turtle- was clearly visible on the computer screen. This result comes in contrast to the findings of other researches in the framework of 3d computational environments that have noted students' preference in working in a plane parallel to the computer's screen display plane (Kynigos & Latsi, 2006, 2007). Working in a plane parallel to the display plane is considered closer to students' experiences with 2d figures in school textbooks or with 2d Logo and would eliminate the convention used in the representation of the 3d space. However it seems that the kind of task and the metaphor used was of critical importance: the aim was not to construct just a slanted line or a geometrical figure but to simulate the taking off and the landing of the turtle - aircraft. In this framework the use of the commands *uppitch/downpitch* as well as the motion of the turtle along the Z axis that gave the impression of depth was rather more easily syntonised with everyday experiences and representations of flying aircrafts.

In the following tasks the students used extensively both the default viewing angle tools and the active vector during their construction processes. It could be suggested that the various viewing angle manipulation tools were especially used a) when a bricolage construction strategy was adopted (episode 1) b) when students were experimenting with specific aspects of the half-baked microworlds (episode 2). In the following episode students are trying to construct 'a wall' during task 2, giving commands to the turtle while using visual cues without having a clear strategy in mind. Their trial and error strategy is evident in the number of commands given to the turtle while trying to construct a parallelogram. It seems that every command is related only to the turtle's previous position and not to the whole construction process and the figure's geometric properties. When it wasn't visually clear if they had constructed a closed figure, students did not resort to the geometrical object’s properties (e.g. that the opposite sides of the rectangular figure should have equal lengths) but to the viewing angle manipulation tools so as to check if the figure was closed. Then they proceeded again forwarding the turtle little step by little step.

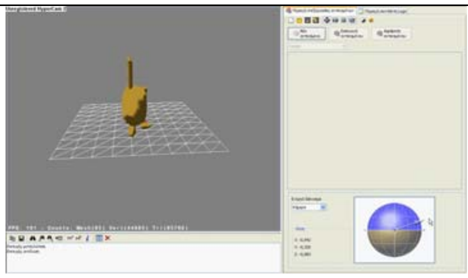
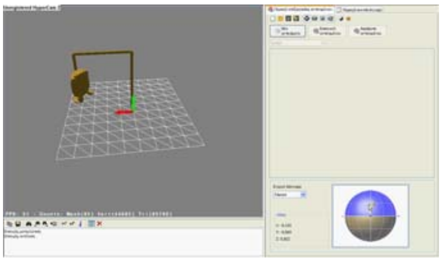
Episode 1: *S1: Fd, more Fd, more…we should go it forward 0.5. Does it have 0.5?* *S2: Wait, wait the turtle is here.* *S1: Yes, but it hasn’t touch the ground. Has it? Wait I have to see it.*(he changes the view from side to front through the active vector tool and continuous forwarding the turtle)	 	Up(180) Dp(45) Dp(45) Fd(4) Rt(90) Rt(90) Rt(90) Rt(90) Rt(90) Fd(4) Bk(1) Bk(1) Bk(1) Fd(1) Fd(1) Fd(1) Dp(45) Rt(90) Fd(1) Fd(1)

Figure 7: First column: Episode1, Second column: Changing viewing angles. Third column: The respective Logo code up to the point of the construction Episode 1 is referring to.

During task 4 students were initially experimenting with the values of the variables of the half-baked microworld ‘revolving doors’. They had extra difficulties in finding out the role of the :d variable, which determined the measure of turtle’s turning and respective position in the 3d space before drawing each successive door of the revolving door model. It follows that the d variable determined also the position of the four rectangle doors in the 3d space as well as their position in relation to one another. In the following episode students are conjecturing about the number of the visible rectangles (doors) if the value given to d is 720. However they do not find the front default view convenient and after testing all the available default views they choose to continue working with the top-down view active, where the number of doors created by the turtle was more clearly visible.

Episode 2:

S1: Lets see how many doors there are if the value is 720 (he plays with the 1d variation tool changing the values of the d variable). *Only one? This perspective is not convenient, I will change it* (he activates successively all the 3 default views and opts for the top –down one)

S2 Yes, exactly like in the case of 360. It turns two rounds.

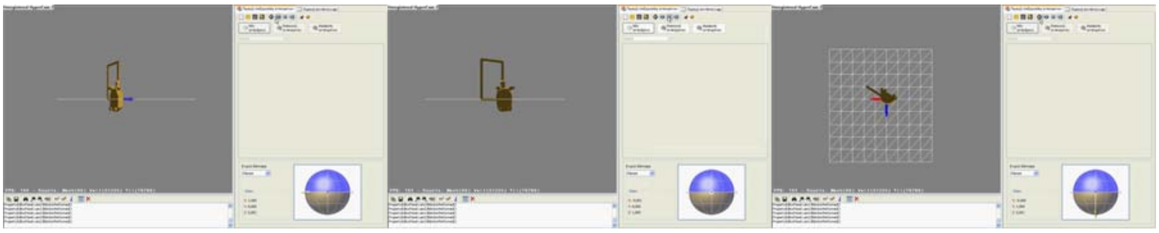


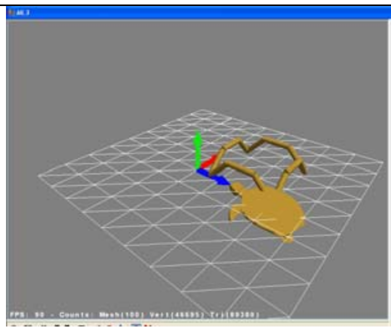
Figure 8: The 3 default views of the revolving door half-baked microworld when the value of the d variable is 720

In sum it could be argued that students have initially preferred a body syntonic way of navigating the turtle while opting for particular default views of turtle's constructions according to challenges faced, e.g. a front view (although a bit adjusted with the active vector) during task 1, a side view (a bit adjusted again) during task 2, a top-down view during experimentation with variable d in task 4. It should be stressed that the preferred default views offered students 2d representation that possibly helped them focus on particular aspects of their construction. In this phase it was more important for students to explore the environment and navigate the turtle command by command taking advantage of the body-syntonic metaphor and viewing space through an intrinsic perspective. However as students' construction strategies shifted to more analytic ones it seems that they ceased being so 'immersed' in the 3d space, a result that is treated in the next paragraph.

Construction processes through an extrinsic perspective

When using the turtle metaphor students have to pass from the management of turtle's spatial movements to the construction of a graphic object (Fein & all, 1987), while making a distinction between the agent and the object, between the navigation of the turtle and the result of this navigation, the geometrical object. In parallel students have to coordinate two different view points: the view- point of the turtle which must be moved in an appropriate way so as to draw a figure and the view point of an external observer who looks at the figural results of turtle's movement. The results of the present research suggest that as the activities unfolded, the students progressively adopted an extrinsic perspective of the 3d space, observing it as external viewers.

In the end of Task 2 there was some free time available and students spontaneously decided to try to construct a closed figure building upon their experimentation during Task1. Students were able to combine the flights they have previously constructed. Each taking-off and landing of the turtle was used as the building block of a 'peculiar' figure that came as result of four repeats of the initial turtle's journey while turning turtle 90 degrees before each reexecution. It is also interesting -as it is evident in episode 3- that students adopted a more analytic strategy, visualising the whole turtle's journey and explaining it to each other before entering commands to the microworld. Moreover when they returned to the microworld they did not execute the commands one by one but they inserted and executed a group of commands.

Episode 3: *S1: Would you like to make a design?* *S2: To make a circle?* *S1: To go this way and then this way and again so* *S2: Let's make a triangle. First it goes this way and then it comes back. No, I have an idea, to insert 45 so as to go this way and then*		Uppitch(45) Forward(2) Downpitch(45) Forward(2) Downpitch(45) Forward(2) Uppitch(45)

again 45 so as to go this way and then again 45 (they are showing on the screen and they are using their hands so as to simulate turtle's journey). *S1: Let's make a rhombus. So not right 45 but right 90* (So far they were talking to each other and now they return to the microworld inserting the commands)		Right(90)

Figure 9: The closed figure and the respective logo commands that were executed four times so as to construct it

Another interesting point was that before starting their construction, the students adjusted their viewing angle through the active vector so that there was a clear sense of perspective of the simulated 3d space (see picture 7). They then continued working on their construction keeping this viewing angle stable. However this was not an occasional choice as the students followed the same strategy during the construction processes of the fan of the watermill during task 4: they adjusted their viewing angle so as to have again a sense of perspective (see picture 8) and they kept it stable throughout the whole construction process. When asked why they preferred this view the students just replied: '*It is more convenient because we can view the whole object.'*.

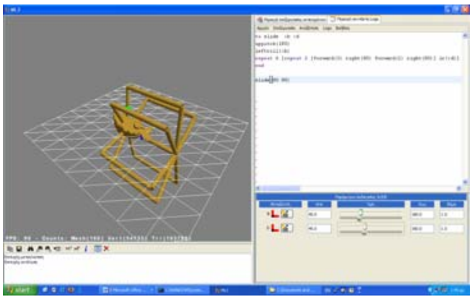


Figure 10: The view preferred during Task 4.

But the questions that arise are: Why students kept on working with a fixed view during 3d constructions that seem to necessitate a high degree of spatial visualisation and orientation. For instance wouldn't it be easier or more body-syntonic to change viewing angle in order to decide turtle's turning before each reexecution of turtle's flight during the construction of the closed figure (episode 3)? What were the reasons for this change as far as the use of the viewing angle manipulation tools is concerned as the activities unfolded? It seems that as students got progressively more accustomed to the 3d turtle's motion and the software's representational infrastructure they weren't so much concerned about body-syntonicity and that it was more important for them to have a clear sense of the threediness both of the simulated space and of the simulated objects. Constructing the simulation of a 3d object while viewing the simulated space in perspective was probably more realistic and familiar. However it could be also conjectured that students preferred a fixed view point during their constructions so as not to change position as observers and to have, thus, a stable point of reference which would probably be less cognitively demanding (Yakimaskaya, 1991). A fixed 3d view rather gave students a sense of space constancy, especially in cases that they adopted an analytic design strategy, as in episode 3, where they mentally visualised the whole turtle's journey before executing the relative commands so as to construct the figure. Thus it could be argued that as the construction process became more complicated students preferred to view space from an extrinsic perspective, as external observers, focusing more on programming and geometric properties while taking into account the whole 3d space.

Conclusions

The above analysis has tried to show that the way the available viewing angle manipulation tools were used was in a constant interplay both with task at hand and with the construction strategies followed. When the focus was on turtle's navigation and orientation in 3d space the body-syntonic metaphor came in the foreground while space was experienced through an intrinsic perspective: the user was immersed in space and was trying to view it from inside. In this case the students used various viewing angles which helped them face specific challenges and focus on particular aspects of their construction. The intrinsic perspective and the use of multiple viewing angles seems also to be adopted in case where a bricolage construction strategy was adopted, when the students had not a clear idea about the actions that should be taken and when the construction was progressing command by command through trial and error.

As the activities unfolded and as the students shifted focus from the management of turtle's spatial movements to the construction of a graphic object, they had started experiencing space through an extrinsic perspective, through the view point of an external observer who looked at the figural results of the turtle's movement. In this case a fixed 3d view was less cognitive demanding and offered students both a realistic effect of familiar objects, and space and shape constancy. Moreover a holistic/external view of the 3d space was in accordance with analytic construction strategies where the students were trying to visualise the turtle's journey taking into account the whole 3d space before executing commands on the computer screen. It was interesting that aspects of the distinction between the intrinsic and extrinsic characteristics of programming with the turtle metaphor in 3d space came in the foreground in a functional way through specific choices of viewing angles It goes without saying that there were not clear cut borders between the two perspectives and that there were a lot of instances that students oscillated between them according to their construction focus. This research was a tentative effort in appreciating an aspect of the large spectrum of the representational potential of a specific 3d microworld in the context of constructionist activities. However a lot of further research is needed in order to investigate the way mathematical concepts can be integrated with spatial navigation and orientation in virtual environments, as well as in order to investigate the way highly visual microworlds, such as MaLt, can be used in educational design.

References

Abelson H. and DiSessa A. (1981). *Turtle Geometry: The Computer as a Medium for Exploring Mathematics.* Cambridge M.A.: MIT Press.

Arcavi, A. (2003). The role of visual representations in the teaching and learning of mathematics. *Educational Studies in Mathematics*, 52(3), 215-241.

Clements D. and Sarama, J. (1997). 'Children's mathematical reasoning with the turtle programming Metaphor', in English, L (ed.), *Mathematical Reasoning, Analogies, Metaphors and Images* Lawrence Erlbaum Publishers, Mahwah, pp. 313–338.

Cobb, P., Confrey, J., DiSessa, A., Lehrer, R., & Schauble, L. (2003). Design experiments in educational research. *Educational Researcher*, Vol. 32-1, 9-13.

Fein, G. G., Scholnick, E.K. , Campbell, E.K. , Schwartz, S.S. , Frank, R. (1987). Computing Space: A Conceptual and Developmental Analysis of LOGO.. In R. E. Mayer (Ed.), *Teaching* *and learning computer programming: Multiple research perspectives* (pp. 55-74). Hillsdale, NJ: Lawrence Erlbaum Associates.

Gutiérrez, A. (1996). *Visualization in 3-dimensional geometry: in search of a framework*. Paper presented at the 20th International Group for the Psychology of Mathematics Education, Valencia, Spain.

Harel, I. & Papert, S. (1991). *Constructionism: Reseasch Reports and Essays.* Ablex. Publishing Corporation. Norwood, New Jersey.

Kafai, Y. and Resnick, M. (eds.) (1996). *Constructionism in practice: Designing, thinking and learning in a digital world*. Lawrence Erlbaum Publishers, Mahwah

Kynigos, C. (1993). Children's Inductive Thinking during Intrinsic and Euclidean Geometrical Activities in a Computer Programming Environment. *Educational Studies in Mathematics*, 24, 177-197

Kynigos, C. (1992). The Turtle Metaphor as a Tool for Children Doing Geometry. Learning Logo and Mathematics, editors: Hoyles C. and Noss R., Cambridge MA: M.I.T. press, 97-126.

Kynigos, C. (2004). A 'Black-and-White Box' approach to user empowerment with component computing, *Interactive Learning Environments*, 12 (1-2) 27-71.

Kynigos, C. & Latsi, M. (2006). Vectors in use in a 3d Juggling Game Simulation. *International Journal for Technology in Mathematics Education*, 13 (1),

Kynigos, C. & Latsi, M. (2007). Turtle's navigation and manipulation of geometrical figures constructed by variable processes in 3d simulated space. *Informatics in Education,* Vol. 6, No. 2, 359-372.

Morgan, C., Mariotti, M., A. and Mafei, L. (2009). Representation in Computational Environments: Epistemological and Social Distance. In *International Journal of Computers for Mathematical Learning,* 14 (3), 241-263

Newcombe, N. & Learmonth, A. (2005), Development of Spatial Competence. . In *The Cambridge Handbook of Visuospatial Thinking*, Shah, P. & Miyake A. (eds) (2005), Cambridge University Press, Cambridge, pp. 213 - 256

Papert, S. (1980). *MindStorms – Children, computers and powerful ideas*. The Harvester Press Limited, London

Presmeg, N. (2006). Research on visualization in learning and teaching mathematics. In A. Gutiérrez, & P. Boero (Eds.), *Handbook of research on the Psychology of Mathematics Education: Past, present and future.* Rotterdam: Sense Publishers.

'ReMath' - *Representing Mathematics with Digital Media* FP6, IST-4, STREP 026751(2005 – 2009)

Reggini, H. (1985) Ideas y Formas, Buenos Aires, Ediciones Galapago.

Tversky, B., (2005). Functional Significance of Visuospatial Representations. In P. Shah and A.Miyake (eds.) *The Cambridge Handbook of Visuospatial Thinking*. New York, CambridgeUniversity Press: 1-34.

Wickens C., Vincow, M. & Yeh, M. (2005). Design Applications of Visuospatial Thinking: The Importance of Frame of Reference. In *The Cambridge Handbook of Visuospatial Thinking,* Shah, P. & Miyake A. (eds) (2005), Cambridge University Press, Cambridge, pp. 383-425

Yakimanskaya, I. S., Wilson, P. S. & Davis, E. J. (1991). *The Development of Spatial Thinking in Schoolchildren*, NCTM.

Crafting activity plans as “improvable objects” as a constructionist activity for Greek language teachers

Katerina Makri, *kmakrh@ppp.uoa.gr*
Educational Technology Lab, Dept of Pedagogy, University Of Athens, Greece

Elena Vlachogianni, *elenavlaho@gmail.com*
Educational Technology Lab, Dept of Pedagogy, University Of Athens, Greece

Chronis Kynigos, *kynigos@ppp.uoa.gr*
Educational Technology Lab, Dept of Pedagogy, University Of Athens, Greece

Abstract

The work presented in this paper is an exploration of the potential of a constructionist rationale in the field of language teacher education.

The evolution of digital media and the advent of multimodal text has shaped a new textual reality in the Language Arts, questioning what was known as “text”. Digital texts are multimodal (distributed through more than one semiotic resource), hyperlinked (organised in different levels) and fluid (malleable and plastic), all three traits affecting both the understanding, and production of written speech. This change has found language teachers reluctant to use new habits of mind and new methods of work, consistent with new textual forms.

Our research with Greek language teachers was inspired by Papert’s influential idea on the importance of constructing and sharing a public entity. Teachers were engaged in designing, versioning and discussing activity plans, considered as public, improvable objects and as malleable entities, amenable to –and instigating- changes, through public negotiation (in the sense discussed by Healy & Kynigos, 2009). We therefore consider teachers’ design of activity plans as an activity with constructionist elements.

Our questions focused on the levels of teachers’ understanding of digital textuality and on the ways this understanding translates itself to their didactical engineering (e.g. if and how the teachers engage their students in activity including construction and public negotiation of digital text).

Through the corpus of successive versions of online discussion transcripts, emerged three levels of teachers’ understanding of the properties of digital texts: a) little or no understanding, b) restricted or superficial, c) complete understanding of concepts, impressed in student engagement in meaningful, collaborative production of digital text. This higher level of understanding is considered as appropriation (in the sense discussed by Kynigos 2007; Fuglestad et al, 2010). These three levels could serve both as analytical tools for further elaboration, and as design aids to the language teacher-designer.

Keywords

Digital text; language teaching; multimodality; design-based teacher education; improvable objects

Introduction: the new communicative order and digital textuality

It is argued that the evolution of digital culture, as well as the new socio-economic reality and emerging literacy practices contribute to the shaping of a new linguistic and communicative reality, which has been termed as a "*new communicative order*" (Street, 1998). It focuses on the literacy practices associated with screen-based technologies and recognizes that *"print-based reading and writing is now only part of what people have to learn to be literate*" (Snyder, 2001: 1). According to the linguist David Crystal (2004: 4), the linguistic reality made possible through broad use of digital media (especially internet based) is one of the changes that have altered the worldwide "*linguistic ecology*". The literacy young people need in this new context has been defined as "*the sustained and flexible mastery of a repertoire of practices with the texts of traditional and new communications technologies via spoken language, print and multimedia"* (Luke, Freebody & Land, 2000:20). In the following section, we discuss three important characteristics of the new communicative order, which, we argue, have a deep impact and implication on the design of the language curriculum and pedagogy.

Traits of digital textuality: multimodality, hypertext and fluidity

A crosscutting feature of the new digital textual reality is multimodality (Kress, 1997), that is, the distribution of meaning among different semiotic resources (written text, picture, sound, etc.), or, in other words, the construction of meaning through a system of multiple modalities, harmonically and not cumulatively coexistent.

The text hosted in digital environments is also characterised by its hypertextual structure, that is, its organisation in different levels, a fact that challenges the prevailing views and principles on textuality. This non-linear architecture of information allows for the connection of words or phrases on the screen to other texts and media representations, -what has been described as electronic "inter-textuality" (Lemke, 1997). According to Brody (2000: 145): "*A linear text, with specified start and end points, is a stable text. The matrix in which electronic text floats, is quite different – a flexible environment that allows multiple layers and n-dimensional reading variants*".

Finally, electronic texts are far more plastic and fluid than their printed predecessors. Thorne (1999) refers to the structure of text produced and consumed in digital and hypertext environments as "*changing and multivalent*". Through tools as simple as common word processing applications, digital texts allow for a number of changes to happen to them. In Brody' s (2000: 145) words, hypertextuality imposes *this polyvalent ability to enter, amend and exit the text in a non linear fashion*".

Implications on language curriculum and pedagogy

Several implications of the above characteristics affect language curriculum and pedagogy.

The first trait, multimodality, adds two new demands to the repertoire of the future citizen: *understanding* and *producing* multimodal texts. The appropriation of this hybrid form of discourse presupposes the acquisition of synthetic skills, as for instance, the understanding of the relations among meanings (Snyder, 2001). In their influential work "Multimodal Discourse", Kress and Van Leeuwen (2001) show how two kinds of thought processes interact in the design and production of communicative messages: "*design thinking*" and "*production thinking*," the kind of thinking which occurs in direct interaction with the materials and media used.

The second trait, hypertextuality, demands the cultivation of a new ability: that of juxtaposing passages of text or text and other media in a functional way, in order to convey a clear message. For second and foreign language education, this mechanism allows making background information, vocabulary definitions, grammar explanations, inter-linear translation, commentary, or cross-references immediately and conveniently available to language learners (Thorne, 1999).

The third trait, plasticity and fluidity, introduces the notion of malleability of written text, a notion not compliant with the "fixed" printed dominant impression on texts. According to Landow (2000: 166): "*Digital text is fluid because, taking the form of codes, it can always be reconfigured, reformatted, rewritten [...] infinitely adaptable to different needs and uses, and since it consists of codes that other codes can search, rearrange and otherwise manipulate, digital text is always open, unbordered, unfinished and unfinishable, capable of infinite extension*" (Landow, 2000, p. 166).

Implications on C.A.L.L. and the practices of schooling

As obvious from the dates of the cited literature, the traits of digital textuality are nothing new. In fact, about 20 years ago, many arguments were made for the necessity of readjusting language curricula and teaching practices, in order to comply with the emerging communicative needs. In 1991, Bolter talked about the transformation of writing into an "*electronic writing space*". Landow also argued for the need for a new paradigm in language pedagogies: "*we must abandon the conceptual systems founded upon ideas of center, margin, hierarchy, and linearity and replace them with ones of multilinearity, nodes, links, and networks*" (1992:2).

A reasonable guess would be that these predictions would, by now, have had a major impact on the field of C.A.L.L. (Computer Assisted Language Learning) as well as on language teachers' use of and attitudes towards technology in the language classroom. Indeed, "*digital and network technologies have helped to initiate a significant pedagogical shift, moving many language arts educators from cognitivist and disembodied assumptions about knowledge and learning as a brain phenomena, to contextual, collaborative, and social-interactive approaches to language development and activity*" (Hawisher, 1994; Noblitt, 1995; Ortega, 1997, in Thorne, 1999).

This shift, however, doesn't seem to have deeply affected current practices in formal schooling. Gunther Kress (1997: 58) comments that "*we are returning to a multimodal world after a period of some two to three hundred years of the dominance of verbal writing as the means of communication and representation*", having then to admit that, "*School, however, does not reflect that return, so that when a child enters school for the first time, there is a huge jump to be made from the "rich world of meanings made in countless ways, in countless forms, in the early years of children's lives, to the much more unidimensional world of written language*" (p. 10). Kress' s assertions are now adopted by several language and literacy educators, acknowledging the need to interrogate the emergent hybrid forms in which verbal and visual modes of representation are combined in new ways (Snyder, 2001). Nevertheless, teachers still appear reluctant to embrace new methodologies and tools related to the new textual forms (ibid). One of the reasons for this may be that the evolution of C.A.L.L. technology has always preceded the maturity of appropriate pedagogical approaches to support it. This causes an effect of reluctance in language teachers with regards to embracing new methodologies and tools related to new textual forms (Snyder, ibid). At the same time, some feel a degree of inadequacy and lack of preparedness for the challenges of the task. They are the product of a print generation: they were shaped, perhaps limited, by print-based understandings of literacy. Unlike the younger generation, they do not feel altogether at ease in virtual environments. For them, images are more often than not thought of as illustrations - even when they fill the entire page or screen and constitute the major mode of communication (Vincent, 2005).

A challenge, thus, for literacy and language teachers is to think of new ways to understand the "*information bricolage*" (Burnett, 1996: 71) needed, in order to be functional in the new communicative order. So a still open issue, as worded by Beavis et al (2009) is: "*What kinds of approaches, models and resources are needed to support teachers in the development and* *implementation of ICT-based curriculum that addresses both print and multimodal forms of literacy*?

The potential of design-based teacher education and the idea of "improvable object"

Our study addresses the issue identified in the previous section by using one of Seymour Papert's influential ideas, and framing it in the context of design-based teacher education.

In Greek language and literacy studies, multimodality and dynamic perspectives on text production are still depreciated as curricular values, the printed text still being prevalent in teaching and exam systems (Koutsogiannis & Mitsikopoulou, 2004). On this basis, we assert that, for language teachers in this context, training activities of added value should include: a) collaborative production of activity plans integrating the use of technology, b) public discussion on the traits of multimodality present in these activity plans and c) guidance on how to transfer this approach in their own teaching habits, e.g. engaging the students in understanding and collaborative production of digital texts. To support this, we adopt a broad perspective of constructionism, on the basis of Papert' s seminal idea that effective learning flourishes in contexts of collective negotiation of shared constructions *(*Papert, 1980). This idea was articulated with young children in mind, and was mostly exploited in science, mathematics and technology. At a later stage, it has been also used in the field of teacher education, for example, mathematics (Kynigos, 2007; Healy & Kynigos, 2009;). A similar approach has been adopted by Laborde (2001), who worked with different versions of mathematics teachers' activity plans, as they gradually acquainted themselves with digital tools. She characterised these versions as "*experimental teaching sequences*", opening a "*window to teachers' epistemologies*", and offering valuable information on the course of their thinking during this experience.

We use the above ideas in the context of Language Teacher Education, and complement it with a basic principle of design-based teacher education, according to which, teachers have to engage in the construction of artifacts (Mishra & Koehler, 2003). This process is collaborative and centres around the design of tangible, meaningful material entities, as end products of the learning process (Blumenfeld et al., 1991). Activity plans, as written texts, are not material artifacts such as models, simulations or other such "tangible" entities. They can be, though, viewed as "conceptual artifacts" (Bereiter & Scardamalia, 2003), embodying, reflecting, and using ideas. These "improvable objects", have a dual role: the center of the collaborative activity, and communicational tools, shaping a common language within a learning community (Bereiter & Scardamalia, ibid). In this sense, teachers' design activity is considered as bearing a constructionist element, that of publicly sharing and discussing specific objects and constructs.

Research questions and design

The first research question relates to if and how the design of digital environments for text production (DETP) induces or enhances the understanding of the three traits of digital text by teachers:

- How do teachers perceive the characteristics of digital text?

The second research question focuses on if and how this understanding translates itself to their didactical engineering, e.g. if and how they engage their students in constructionist activity, considering this second level of understanding as *appropriation –our emphasis-* (in the sense discussed by Kynigos et al, 2003; Fuglestad et al, 2010;). In other words, we suggest that teachers understanding of theoretical ideas is basic for their development. However, it is putting these ideas *in use* through designing activity plans that make students engage in constructionist activity is what differentiates theoretical grounding from enrichment of the teaching practice repertoire.

- If / if yes, how do these perceptions affect their professional identity and mindset whilst they craft instructional ideas and design teaching scenarios?

The tools we decided to use as DETPs were intentionally open-ended, that is, not drill-and-practice-style software, but environments such as office applications (word, powerpoint) and web 2.0 tools (blogs, wikis, media sharing applications). These, though not designed for educative purposes, are considered appropriate for authoring digital text, as they allow for any configuration of media and written information and free the user from predefined paths.

Teachers worked in the following order and processes:

- First, they were introduced to the three traits of digital textuality, both through reading relevant literature, and through whole class discussion, synchronous, and asynchronous.
- Then they were asked to work on a small but coherent teaching proposal, including the use of a DETP of their choice, publish it on the class forum and discuss on each others' ideas. Meanwhile, they would also have to work with their chosen digital tool and accompany their ideas with this artefact.
- This discussion' s output, was, in many cases, the versioning and re-adjustment of the proposals –and the artefacts- according to the peer feedback received.
- Finally, they were asked to prepare a longer text –as their final assignment- individually, called a "scenario of use of digital tools". This was seen as the final construct. Though an individual creation, we consider it a construct bearing evidence of collective understanding and negotiation, as it is the product of both individual work and study, and of the whole design process.

Our data corpus comprised of: a) the electronic discussion transcripts, from where we distinguished the different versions, and b) the final proposals (accompanied with digital artefacts, such as powerpoint presentations, videos, web 2.0 links etc.).

Our analysis was based on two axes, directly related to the research questions:

1. Teachers' understanding of the traits of digital textuality
2. Teachers' appropriation of the traits of digital textuality, informant of their future practice.

Findings

We identified three levels of teachers' understanding, represented as an evolving process: in level 1, teachers propose activity plans which neither depict their understanding of digital text, nor addresses students' respective understandings, as a learning objective. In level 2, teachers propose activity plans which indicate a certain level of their understanding of digital text, and an attempt to introduce their students to it, but this is restricted and exploratory. In level 3, teachers craft activity plans which indicate a deep level of understanding and a clearly stated aim of transferring this to students. The criteria used for this examination directly relate to the research questions and are the following:

- Relation of teaching aims and objectives to the traits of digital text
- Use of DETP functionalities supporting the functional and not cumulative combination of media modalities
- Support of such aims and objectives by the proposed flow of student activities: students are expected to *understand* digital text *by* collaboratively *producing* public and negotiable artefacts.

Each level corresponds to a set of indicators that emerged from the data and a description composed afterwards, based on the indicators (Table 1).

Levels	description	indicators
1	The proposal (and /or its versions) show no signs of **understanding** of concepts and as a result no signs of engagement of students in constructionist activity (digital text production)	- Absence of didactical aims related to the concepts - Aims restricted to technological literacy - The flow of activities doesn't include students' engagement with digital text production - DETP functionalities are not exploited or combined to support the concepts - Students are expected to understand, but not produce digital text
2	The proposal (and /or its versions) indicate a degree of student engagement in constructionist activity, but this is restricted and superficial.	- Presence of didactical aims related to the concepts but not adequately supported by the proposed flow of student activities - The flow of activities includes engagement with digital text understanding, and/or production, but DETP functionalities are not fully exploited or combined to support the concepts - Students are expected to produce digital text, but the production is either too restricted (all media provided by the teacher) or too superficial (cumulative, not functional use of different modalities) - Students are expected to engage in digital text production, but only to "announce" it to the class, not negotiate it as a malleable entity
3	The proposal (and /or its versions) indicate full understanding of concepts as depicted in student engagement in meaningful production of digital text	- Presence of didactical aims related to the concepts, adequately supported by the proposed flow of student activities - Students are guided to fully exploit or combine DETP functionalities to the concepts - Students are expected to produce digital text, taking full advantage of the range of media they have available - Students are given the opportunity to publicly negotiate their constructs

Table 1. Teachers' levels of understanding of digital text

This matrix was drafted from data derived both from the versions of the activity plans discussed by teachers, and by their final deliverables / products. 8 teachers presented three versions of their work and 11 presented only two.

With regards to teachers' evolution through the levels, as shown in table 2, about half of the students (10) didn' t show any progression and remained on the first (3) or the second (7) level of understanding. The rest of the class (9) indicate a progressive course, mostly from level 1 to level 2. Only 3 teachers made it to level 3, while 2 others digressed either from 2 to 1, or from 3 to 2.

fixation on 1	3
fixation on 2	7
progression from 1-2	4
progression from 2-3	2
progression from 1-3	1
digression from 2-1	1
digression from 3-2	1

Table 2. Teachers' evolution through the levels

The majority of teachers' work is characterised either by a gradual progression from the first to the second level, or a fixation on the latter. This is further certified by the class average of 1.8 (derived by counting the scaling for each version of each participant).

An indicative example of fixation at a certain level (2) is the case of John: his work was presented in three successive versions. In the first one (level 1), he states that:

"*The students, in groups, are assigned to create their own powerpoint presentation, with 150-200 number of words*".

It is obvious from this statement that he hasn't grasped the concept of functional combination of media –the essence of multimodality- and regards MS powerpoint, a DETP offering multiple configurations of media and written text, as just a word processor. However, in his second version (altered after the peer feedback he received), he starts referring to the concepts of collaborative text production, multimodality and fluidity, enriches his tool repertoire with an internet browser and MS paint, but still fails to connect these to solid teaching objectives. As he says in the forum discussion:

"*Students should acquire technological literacy, by learning the basic functionalities of word processing*".

His final version remains on this level. Though his proposal is now enriched with a much richer variety of digital tools (a blog, a web file repository, Ms Word, video and a database programme), his focus remains on just technological literacy. And though his proposal includes student engagement with specific traits of digital text, this is done in a subtle way, and is not clearly articulated, as shown in the proposal extract below:

"*Students are divided in groups of three and asked to collaborate on one presentation, as a digital story. This should include pictures, sound and text, and these should aid successful transmission of the message of communication. Finally, the groups upload their stories to the class blog*".

Here, there is an intention of engaging the students in digital text production, but the aim stated for this (*successful transmission of the message of communication*) indicates lack of full understanding of the potential of multimodal discourse.

An indicative case of progression (from level 2 to 3) is that of Maria, whose work was presented in two successive versions. In the first one (in the forum discussion), she presents a proposal combining the use of MS Word and powerpoint for the joint production of a story, and the manipulation of its beginning and end by different groups of students. Though this activity includes, in a high degree, a collaborative process of co-authoring a multimodal text, this is not what she verbally stresses. Instead, she focuses on the added value of the use of new media in general. However, in her final proposal, first she alters her toolset, by replacing office applications with web 2.0 elements (a blog, where a story is collaboratively authored and publicly monitored). Their use is supported by appropriate stated objectives, depicted in the proposed flow of activities. Joint production and public negotiation of digital text now seems one of her clearly stated pedagogical considerations:

"*Students are expected to produce joint products using social applications [wiki], and discuss on these on the class blog. The teacher will monitor this discussion and give constant feedback*".

Discussion

In this section, we discuss the following three points, drawn from our experience and conclusions:

1) The potential of constructionist ideas in the area of language teacher education

Constructionist approaches have been scarcely used, both in the areas of CALL (computer assisted language learning), and in language teacher education. We attribute this to the saliency of the concept of "construct" or "construction" in the field of literacy. Exploring the new landscape shaped by digital text induces the experimentation with what actually is tangible and constitutes a construct. We suggest that the idea of activity plan design bears constructionist elements. Further empirical research would allow for more robust theoretical grounding of constructionist principles in language teacher education.

2) The difference between understanding and appropriation

Our experience with the construction of activity plans and digital tools by teachers, and their versioning, discussion and negotiation made us think of the focus of our teacher training curriculum. If it is on understanding the traits of digital text, reading related literature and individually producing such texts should be enough to provide a language teacher with a basic theoretical background and one indicative experience with DETPs. There were enough activity plans in our corpus which indicated a certain level of understanding, though they weren't promising, in terms of the adoption of this understanding and its translation in future practice. If, though, we aim both at understanding, and at integrating this new skill in a teachers' professional practical repertoire, then a course should include both theoretical sessions, and experiential learning, through hands-on activities with actual DETPs, active discussion on their functionalities and the restrictions or potential they afford, and constant effort to relate these with an actual activity plan, addressed to real world students. This has not proven an easy task for language teachers in our context, as the focus on what the students will *do* with a text and a digital tool has never been an element of Greek language curricula, considering language learning as a cumulative acquisition of knowledge, and texts as static and fixed entities, only amenable to teacher feedback.

3) The importance of tool use in teacher appropriation and student engagement

We also noticed that teachers who reached a deeper level of understanding did this after having drafted more than two versions of their activity plans. These final products always include the use of a wide range of digital tools, functionally and purposefully combined to lead to specific learning results. The usual progression is from office applications to web 2.0, for example, co-authoring of text in word evolves in co-authoring of text in a wiki, thus exploiting the essence of collaborative text production a wiki offers. This, of course, demands a great deal of hands-on work with the tools, evident in most discussion around level 3 products. In turn, this intensive "bricolage" seems to have a deep effect on teachers' design rationale. Those who didn't present multiple versions –and consequently, didn't intensively experiment with technology themselves-didn't expect their students to do this, either. On the other hand, teachers who reached level three through multiple versioning and discussion, also expected their students to engage in such activity and proceed from just understanding new forms of text, to actually producing them.

References

Beavis, C., Bradford, C., O' Mara & Walsh, C. (2009). *Research Methodologies in Creative Practice: Literacy in the Digital Age of the Twenty First Century - Learning from Computer Games*. Paper presented at the Australian Association for Research in Education International Research Conference, Queensland University of Technology, Brisbane, November 30-December 4/ 2008

Bereiter, C., & Scardamalia, M. (2003). *Learning to Work Creatively with Knowledge*. In: E. De Corte, L. Verschaffel, N. Entwistle, & J. van Merriënboer (eds.), Unravelling Basic Components and Dimensions of Powerful Learning Environments. EARLI Advances in Learning and Instruction Series

Blumenfeld, P., Soloway, E., Marx, R., Krajcik, J., Guzdial, M., & Palincsar, A. (1991). *Motivating project-based learning: Sustaining the doing, supporting the learning*. Educational Psychologist, 26 (3 & 4), 369-398

Bolter, J. D. (1991) *Writing Space -- the Computer, Hypertext, and the History of Writing*. New Jersey: Lawrence Erlbaum Associates

Burnett, R. (1996). *A torn page, ghosts on the computer screen, words, images, labyrinths: Exploring the frontiers of cyberspace*. In G. Marcus (ed). *Connected:* Engagement with media, 3, 67-98. Chicago: University of Chicago Press.

Crystal, D. (2004). *The Language Revolution*. Polity Press Ltd

Fuglestad, A.B., Healy, L., Kynigos, C., Monaghan, J. (2010). *Working with teachers: context and culture, Teacher and Technology*. In Digital technologies and mathematics teaching and learning: Rethinking the terrain, C. Hoyles & J-B Lagrange (eds), Section 3, Chapter 13. Springer Verlag

Healy, L. & Kynigos, Ch. (2009). *Charting the microworld territory over time: design and construction in mathematics education.* ZDM Mathematics Education, 42, 1, 63-76

Koutsogiannis, D. & Mitsikopoulou, B. (2004). *The Internet as a global discourse environment*. Language Learning and Technology, 8, 3, 85-89

Kress, G. (1997). *Before writing: Rethinking paths to literacy*. London: Routledge

Kress G. & van Leeuwen, T. (2001), *Multimodal Discourse.* London: Arnold

Kynigos, Ch. (2007). *Using Half-baked Microworlds to Challenge Teacher Educators' Knowing*. In International Journal of Computers in Mathematics Learning, 12, 87-111

Laborde, C. (2001), *The use of New Technologies as a vehicle for restructuring teachers' mathematics*, in F. - L. Lin & T. J. Cooney (eds.) Making Sense of Mathematics Teacher Education, 87-109, Kluwer Academic Publishers

Landow, G. (1991). *HyperText: The Convergence of Contemporary Critical Theory and Technology*. Parallax: Re-visions of Culture and Society Series. John Hopkins University Press

Lemke, J. (1997). *Metamedia literacy: transforming meanings and media*. In D. Reinking, L. Labbo, M. McKenna and R. Kieffer (eds.), Literacy for the 21st Century: Technological Transformation in a Post-Typographic World (pp. 283-301). Mahwah, New Jersey: Lawrence Erlbaum.

Luke, A., Freebody, P., & Land, R. 2000 *Literate Futures: Review of literacy education.* Brisbane, Qld: Education Queensland

Mishra, P., & Koehler, M. J. (2003). *Not ''what'' but ''how'': Becoming design-wise about educational technology*. In Y. Zhao (Ed.) What teachers should know about technology: Perspectives and practices (pp. 99–122). Greenwich, CT: Information Age Publishing

Thorne, S. (1999). Chapter 3: *Educational and Foreign/Second Language Uses of Computer-mediation: A Review of Research*. In An activity theoretical analysis of foreign language electronic discourse. Unpublished Doctoral Dissertation, University of California, Berkeley

Vincent, J. (2005). *Multimodal Literacies in the School Curriculum: an Urgent Matter of Equity*. In Proceedings of ICTE 2009, International Conference on Multimedia and ICT in Education. Edited by A. Méndez-Vilas, A. Solano Martín, J.A. Mesa González and J. Mesa González

Snyder, I. (2001). *A New Communication Order*. Paper presented at the Joint National Conference of the Australian Association for the Teaching of English and the Australian Literacy Educators' Association, Hobart, Tasmania, Australia, July 12-15, 2001

Street, B. (1998). *New literacies in theory and practice: what are the implications for language in education?* Linguistics and Education 10, 1, 1-24

Visualization processes in a 3d tool designed for engineering activities

Foteini Moustaki, *fotmous@ppp.uoa.gr*
Educational Technology Lab, School of Philosophy, Department of Pedagogy

Chronis Kynigos, *kynigos@ppp.uoa.gr*
Educational Technology Lab, School of Philosophy, Department of Pedagogy

Abstract

The development of spatial and visualization abilities in engineering education has been considered as a process unrelated to the use of external representations as they are usually "monitored" through tests meant to define the level of accuracy in which the students create, store, recall from their memory and manipulate internal mental images. Professional computational environments, when employed, make visualization processes even more obscure for engineering students, as the boxed up mathematics they include restrain the potential for genuine explorations inside the 3d space.

Constructionist environments designed to offer features that support dynamic visualization inside the 3d space, on the other hand, provide a richer context for engineering students, as they seem to allow them to utilize and develop elements of their spatial and visualization abilities. Half-baked microworlds developed in such environments may challenge students to rebuild artefacts according to their own conceptualizations of the 3d space, while Logo programming components could be used to shed light to procedures that they appear black boxed in professional software.

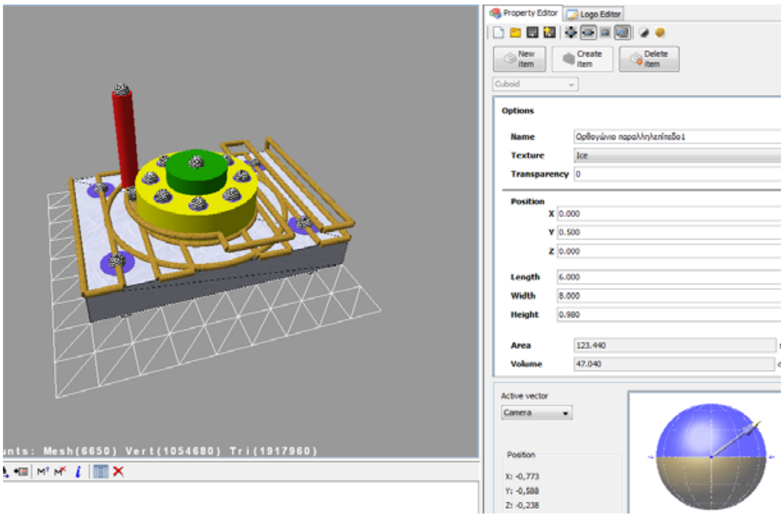


Figure 1. Representing an engineering component in a 3d space and creating Logo programs to define the cutting tool's path

Keywords

Spatial & visualization abilities, engineering education, constructionism, dynamic visualization

This work among others led to a European Commission project on collective reflection of collaboration in constructionist environments called "Learning to learn together: A visual language for social orchestration of educational activities". FP7-ICT-2009-5, Technology-enhanced Learning, N. 257872.

Spatial and visualization abilities in mathematics and engineering education

Spatial and visualization abilities have long been considered as being connected to mathematical learning and aspects of it, such as geometrical thinking and problem solving (Bishop, 1980; Presmeg, 2008; Clements and Battista, 1992). Apart from mathematics education, however, the development and improvement of spatial and visualization abilities have also been recognised as fundamental for the engineering students' education (Sorby and Baartmans, 2000). The main argument for this thesis is that the engineers through visualization bring forth the ideas they have in mind -using in most cases engineering drawings as a vehicle- and thus produce creative solutions to the design and technical problems they encounter (Ferguson, 1992). Therefore, several tertiary institutions urge engineering freshmen to take Spatial Ability Tests, so as to define if they possess these abilities and their level of elaboration, and then to get retested once again after attending "remedial" courses that are thought to accentuate those skills (Potter and Van Der Merwe, 2003). The perception of visualization from a cognitive psychology point of view as the process of producing, retrieving from memory and manipulating mental images in one's mind, with the minimal interference of external representations -including technology (for a discussion see Gutierrez, 1996)- has led to the development of various spatial abilities and mental rotation paper-and-pencil tests, that are designed to determine the level of accuracy and speed in which engineering students may create and manipulate internal representations of complex 3d objects, often unfamiliar to engineers. The use of computer environments, where available, has been mainly viewed as new –alternative to the paper and pencil- means for the students' spatial instruction through drawing and sketching and has been restricted to the use of CAD packages (Leopold et al., 2001). The question that raises here is what the mathematics education's view of visualization and the constructionist paradigm has to offer engineering education.

Although mathematics educators often use diverse definitions to explain terms such as spatial and visualization abilities, there is no doubt that they hold a long-standing research tradition in this specific field. Taking a different strand than the one presented before in the engineering education's case, visualization has been considered by mathematics educators as the ability to represent, transform, generate, communicate, document, and reflect on visual information (Hershkowitz, 1989, p. 75). The development of this ability, however, has not been regarded as independent of the use of external representations and especially technological artefacts (Gutierrez, 1996; Zimmerman & Cunningham, 1991). Particularly, computational environments that hold the potential for the creation and manipulation of dynamic images (dynamic visualization), may contribute to the development of the students' spatial and visualization abilities (Christou et al., 2007), as these often appear to be interwoven with the software's semantics and functionalities (Kynigos & Latsi, 2007). This kind of environments may not only empower students to build and observe inside them structures of 3d objects with certain properties, but also to transform those constructions in real time (Arcavi & Hadas, 2000), through dynamic manipulation. Real-time dynamic manipulation of 3d constructions could potentially enhance students' conceptualisation of the 3d space and enable them to work inside a three-dimensional frame of reference with the use of x, y, z Cartesian coordinates (Kynigos et al., 2007). Logo components in such computational environments that allow the manipulation of symbolic code -generating graphical effects on the screen- may also contribute to the students' mathematically driven navigation and orientation inside the 3d space, as they come to build bridges between the intrinsic turtle geometry and the Cartesian geometry (Kynigos, 1991).

Looking once again at the engineering education context, one may say that constructionism, as an idea, contains inside it what engineering students at the school labs usually do. They construct their knowledge about the engineering world as they collaboratively design, calculate, measure, program, shape and cut meaningful for them external artefacts. Tearing down something that they don't know how it works, just to explore its functionalities or repair it if broken and show it to others is regarded as a process belonging to their established habits of mind (Cuoco et al., 1996). However, when it comes to the use professional computational environments in which 3d objects can be designed and represented -such as CAD packages- engineers often find themselves to perform routine operations, such as pressing buttons on the environment's UI, unable to take them to pieces and find out what lies beneath. Thus, procedures that seem to be connected with visualization in the 3d space, such as changing a 3d object's scale or rotating it around an axis, remain boxed up when using professional computational environments, as their attainment is just a matter of selecting and clicking on the correct button (Kent & Noss, 2001). As mathematics become invisible and crystallized in black boxes inside sophisticated 3d CAD environments, visualization in the 3d space becomes a more and more obscure process for the engineering students.

Bringing mathematics educators experience in developing computational environments meant for mathematically driven visualization and adopting the constructionist perspective, we designed and developed a computational environment called MachineLab Turtleworlds (MaLT). The dynamic visualization of the 3d objects represented inside MaLT's 3d virtual space is achieved both through their direct, real-time manipulation and the execution of Logo procedures that define their properties and behaviours. The 3d manipulation of a camera's position and direction inside the 3d virtual space allows observation from multiple viewpoints, giving dynamic visualization a new powerful tool. The "3d Modelling & Cutting" microworld we developed in MaLT was specifically designed to challenge engineering students' conceptualisations of the 3d space as they attempt to represent inside it mechanical engineering components and generate for them shaping and cutting procedures using Logo programming. In this paper we report on a small-scale research conducted to study high school engineering students' visualization processes as they work with the "3d Modelling & Cutting" microworld. Analysing their reasoning activity while they explore, build and manipulate their constructions in the 3d space, we attempt to identify the kinds of spatial and visualization abilities they employ and monitor their development as the students interact with the computational environment.

The computational environment

The MachineLab Turtleworlds (MaLT) environment

MaLT is a programmable environment that allows the creation, exploration and dynamic manipulation of 3d geometrical objects, graphically represented inside a 3d virtual space. The objects visualized inside it are either *constructed* by the environment's Turtle, when running Logo procedures and commands, or *inserted* by the user, after selecting them from a library that contains numerous ready–made stereometric objects, such as cuboids, cylinders and cones. Inheriting elements from "E-Slate 2d Turtleworlds", MaLT integrates symbolic notation -in the form of Logo programs– with the dynamic manipulation of the 3d geometrical objects through the use of specially designed Variation Tools (Kynigos & Psycharis, 2003).

To observe her/his constructions inside the 3d space from different viewpoints, the user has at her/his disposition three different cameras: the Floor View, the Side View and the Main/Front View camera. However, building and manipulating geometrical objects in MaLT, is not restricted in solely looking at the 3d world form static 2d orthographic views. A 3d Camera Controller gives students the opportunity to navigate around, inside and through their constructions, offering the potential for new ways of visualizing the 3d space and the geometrical constructions inside it. The dynamic manipulation of the objects themselves in the 3d virtual space, along with the dynamic manipulation of the camera, can prove to be powerful tools for understanding mathematical concepts in the 3d space, and a resource for solving mathematical problems that require the use of spatial information.

The “3d Modelling & Cutting” microworld

Using MaLT as a platform, we built the “3d Modelling & Cutting” microworld in which the Turtle is replaced by a cylinder representing a milling machine’s cutting tool (Figure 1 and Figure 2).

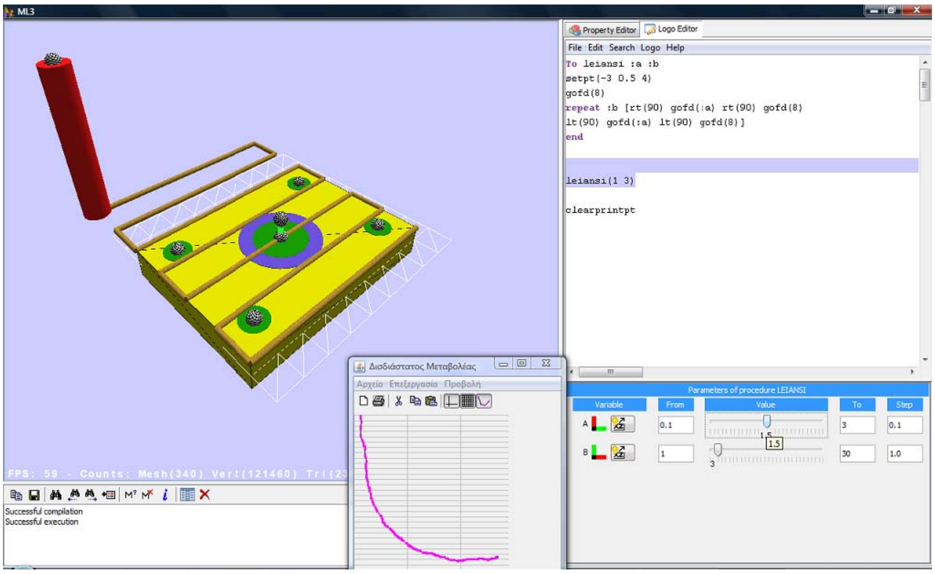


Figure 2. Planing an object’s XZ surface using a milling machine’s tool

The tool can be programmed through Logo commands to move inside the 3d space and perform in it several machining procedures (such as drilling and planing), leaving on its way a trail behind. This trail, when generated though the use of Logo procedures that encompass variables, transforms from static to dynamic and becomes connected to the environment’s Variation Tools. Manipulating the Variation Tools by performing dragging actions (Figure 2), the user attributes each time new values to the procedure’s variables, causing the tool’s path to change its direction, length, and/or position in the 3d space, which allows the parametric programming of several machining procedures.

Research design and methodology

Our research approach was based on the idea of studying learning in authentic settings through “design experiments” (Cobb et al. 2003). “Design experiments” aim to contribute to the development of grounded theories on “how learning works” and are conducted with the intention to shed light on the relationships between the material designed for the experiment (usually innovative technological artefacts having added pedagogical value) and the learning processes within a specific context of implementation.

Context and participants

The experiment took place in a Secondary Vocational Education school in Elefsina –an industrial town near Athens- with three 12th grade students, studying mechanical engineering and having a particular specialization in Programming Computer Numerical Control (CNC) Machines. As part of their two-year tuition, these students had taken courses in working with Computer Aided Design and Computer Aided Manufacturing environments and had operated real CNC milling and lathe machines with the help of their lab teacher. All of them had also been working at the time of the research at middle-scale mechanical engineering workplaces as inexperienced workers. Their school, as well as their workplace training, had given them several experiences in working with Cartesian coordinates, mostly, however, in two dimensions.

The experimentation process was held at the school's CNC lab for 15 school hours, with the students working together as one group. All three shared one PC, while they had at their disposal the computational environment's manual and a notebook for their ideas, remarks, sketches and drawings. Adopting a "participant observation" methodology, the researcher, who also was a Mechanical Engineering teacher, did not intervene to give out instructions or to provide the "correct answer", but chose to pose meaningful -often intriguing- questions at certain time points, so as to encourage students to continue their explorations, collaborate, share and discuss their ideas with each other.

Tools and Tasks

Drawing on the idea of "layered learning design" (Kahn et al., 2006), we divided the activity sequence in three distinct phases, the first of which also served as a "familiariazation with the computational environment" phase. In order to recreate a situation that could be experientially real for the students (Gravemeijer et al., 2000) and close to their professional life, we decided not to provide them any information regarding the features of the environment and ask them to work with it as if they were in their workplace and all three as colleagues had to explore and understand its functionalities, so as to represent in it objects that would be consequently cut in CNC machines. The only available help was coming from the environment's manual -with not much revealed about the 3d virtual workspace- challenging students to figure it out for themselves, if to work in this environment.

Phase 1: An unfamiliar 3d space

As we had decided not to provide the students any instructions regarding the environment's functionalities, we chose to develop for the first phase of our experimentations a relatively simple microworld. This microworld consisted of just one ready-made stereometric object, a rectangular parallelepiped sized 5x5x1, which, with the Top View being activated on the environment, looked like a 2d parallelogram sized 5x5. Working mostly with 2d environments, the students usually recognise as the "Width" dimension the one represented in the Y axis and the Y axis as being the vertical one on the surface of their screen. To induce the students to explore the computational environment's virtual space in all three dimensions, we decided to ask them to resize the object to 3x4x1 (length x width x height) and place it at the (0, 0, 0) point (XYZ). Possible visual mismatches between their established views of the 2d and 3d space and the feedback received from the environment could serve as starting points for new conceptualizations.

Phase 2: An arrangement of objects

For the second phase of the experimentations, we asked the students to represent in the environment's 3d space an engineering component for which we only provided a 2d drawing (Figure 4a). This was supposed to be the drawing a client had given them, requesting a presentation of a 3d prototype, so as to give his approval for the component's final production in C.N.C machines. As the 2d drawing represented the Top View of the 3d component, the only dimensions defined in it were the ones visible from this specific orthographic view. The dimensions of the 3^{rd} axis -and consequently the final shape of the component- were completely left to the students' choice. Since the cuboid they worked with in during the previous phase appeared to be now a main part of their component, we expected students to try to combine the cuboid's position in the 3d space with the positions of the rest stereometric objects necessary to complete the represented prototype. This process could foster the need to form specific spatial relationships between different 3d shapes, possibly performing at the same time rotations of the component in different directions so as to inspect it as a unified entity.

Phase 3: Moving the tool on the XZ surface

For the third phase of the experimentations we gave students a half-baked microworld (Kynigos,

2007) depicting, when running a Logo program, a machining procedure we presented as "planing across a cuboid's four XZ edges". However, after planning the first XZ edge, the tool was programmed to move in random ways inside the 3d space (Figure 5). Half-baked microworlds, being by their own nature incomplete, intrigue students to explore their functionalities deconstruct them and built on their parts. The "debugging" of a faulty machining Logo procedure could engage students in visualisation processes which may entail making sense of an object's position and displacements (the tool's) with regard to another one (the cuboid), both in terms of Cartesian geometry (3d coordinates) and Logo intrinsic geometry (distance from the previous position and heading).

Data collection-Method of analysis

A screen-capture software (HyperCam2) was used to record the students' interactions with the "3d Modelling & Cutting" computational environment, while a camera operated by the researcher was used to record the students' gestures and the process of generating free hand sketches and 2d drawings on paper. The corpus of data also included the researcher's field notes and the students' answers on worksheets we provided at each phase of the experimentations. The video-recorded data from the screen-capture software were verbatim transcribed, while the rest of the data were used for providing additional details. In analysing the data, we searched for verbal exchanges between the students and interactions with the computational environment that indicated that students brought forth, utilized and enhanced their spatial and visualization abilities as they attempted to make sense of the 3d space and specify the absolute and relative positions of the objects represented inside it.

Results

Tracking an unfamiliar 3d space

At the first phase of the experimentations the students were given a 5x5x1 object placed in a random position inside the 3d space and were asked to turn it to 3x4x1 and then move it to X0 Y0 Z0. Although it was quite clear that this was a 3d object to be consequently shaped in a milling machine, the students chose to represent it in paper using a 2d (XY) frame of reference (Figure 3). The shape of this object corresponded to the static 2d orthographic view activated at that time on their computer screen (i.e. a rectangle 3x4). It was only after trying to manipulate the object inside the 3d space and move it to X0 Y0 Z0 that they looked for a 3rd axis (i.e. the Z) and defined a new, 3d frame of reference (XYZ), also determining each positive half-axis's orientation in the 3d space.

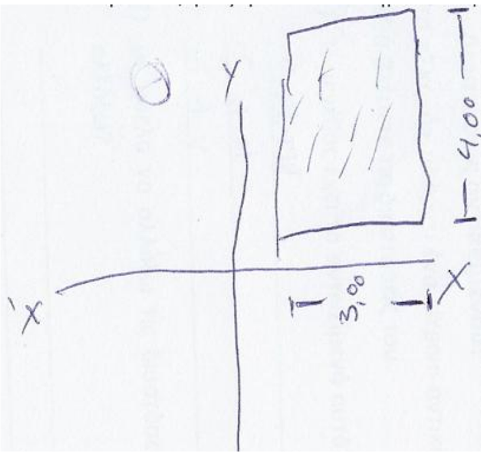


Figure 3. The 2d reference system the students originally came up with. The x' and y' are the negative half-axes

S2: Z = 0 is here [points vertically to the XY surface on the paper]. Z equals to 0 at 1.

R: Would you like to explain that?

S2: Now we can't see it [the Z axis]. I guess that's because we look at it [the object] from above

S1: Z is… comes from above [points vertically to the screen]

S2 decides to move the cuboid to Z=0.

S3: What happened???

R: How did it move?

S1: Upwards!

S2: At the Y axis's direction! At its positive direction! .... I'll move it again. Let's make X=1….[the object moves to the *right]. That's normal for the X axis.*

S1: I'd say this goes the other way around. That's the XZ there. Not the XY!

S2: Make Z -2!…. [after the feedback] That's rotated! [the reference system they had came up with]. Z defines upwards-downwards!

Although recognising 3d frames of reference and working with 3d coordinates was part of their curriculum, mapping the "3d Modelling & Cutting" environment's unknown 3d space seemed to challenge the students' spatial orientation and visualization abilities. Initially, the students seemed to devise an egocentric X-Y frame of reference that corresponded to the way they were looking at the object on the screen from their own viewpoint (Figure 3). Disregarding the fact that this was a 3-dimensional object (3x4x1) and its position was defined by an ordered triple of coordinates (X, Y, Z), the frame of reference they formulated was merely a 2d one. However, being able to *dynamically manipulate* the object inside the 3d space and observe the effect of their actions seemed to be enough of a spark to trigger new visualization processes that incorporated the interpretation of the object's changes of position in the 3d space. Translating the object's displacements inside the 3d space and interconnecting them to the X, Y, Z axes, the students defined the environments' frame of reference as being three dimensional and "rotated" with regard to the one they had originally come up with.

Representing a 3d object using a 2d drawing

To complete the construction of their engineering component in the 3d space, we gave students a 2d drawing that represented the component's Top View (Figure 4a). Although it was once again quite obvious that this was a 3d component that would be finally cut in CNC machines and that the "3d Modelling & Cutting" was a computational environment inside which 3d objects could be represented as such, the fact that we gave them a 2d drawing seemed to disorient and confuse the students.

S3: Now, to represent the rest of the component's parts, these drawings [pointing at the circles at the 2d drawing]….

R: What are these?

S3: Circles

R: What they would be in the real component?

S3: Holes

S1: Holes

S3: And we want to represent them, right? What will we do?

S1: Use circles!

With the Top View (XZ plane) being activated on the environment and looking at the 2d drawing, the students decided to represent the holes in the component not as cylinders (having some specific depth up to which they would have to move their drilling tool when machining them), but as 2d circular disks. The fact that the 2d drawing of the 3d component didn't include any information about the 3rd axis's dimensions, seems to have tricked students into transferring elements and properties of the 2d orthogonal projection into the 3d representation of the component (Parzysz, 1988). Even though this component was a quite common one in the engineering field (i.e. a flange) and the students were able to explain its function in specific machineries when asked, they kept representing with 2d disks all of its parts that the 2d orthogonal projection made them look circular. Finally, every hole or other cylindrical element on the component was represented merely as a 2d circular disk in the XZ plane (Figure 4b).

Taking about their client and the fact that he would be interested in the examining the component in detail before giving his approval for its production in CNC machines, the students looked for functionalities of the environment that would allow them to display the other two orthogonal views of the object, the Front and the Side one.

S1: [S1 moves the "Camera Controller" around and Figure 4b reveals]. What happened? The client won't be satisfied! [laughing]… This shows us everything. All the views we need, the front, the top, the side.

R: What about the parts we inserted?

S1: We have to state that these are holes.

R: What kind of objects we'll use for this?

S1: Solids, 3 dimensional. The circle is not one of them. Look how we got misled!!

R: What can we do?

S1: Let's give the circle some height! [make the circle a cylinder].

Manipulating the Camera Controller, the students navigated in the 3d space and observed the shapes and positions of the component's constituent parts, examining them form different viewpoints. The *dynamic visualization* of the component through the use of the Camera Controller enabled students realise that they had been "misled" by the static 2d drawing. It revealed the need not only to use 3d geometrical objects instead of 2d ones, but also to specify spatial relationships among the component's parts that would not differentiate as they changed viewpoints inside the 3d space.

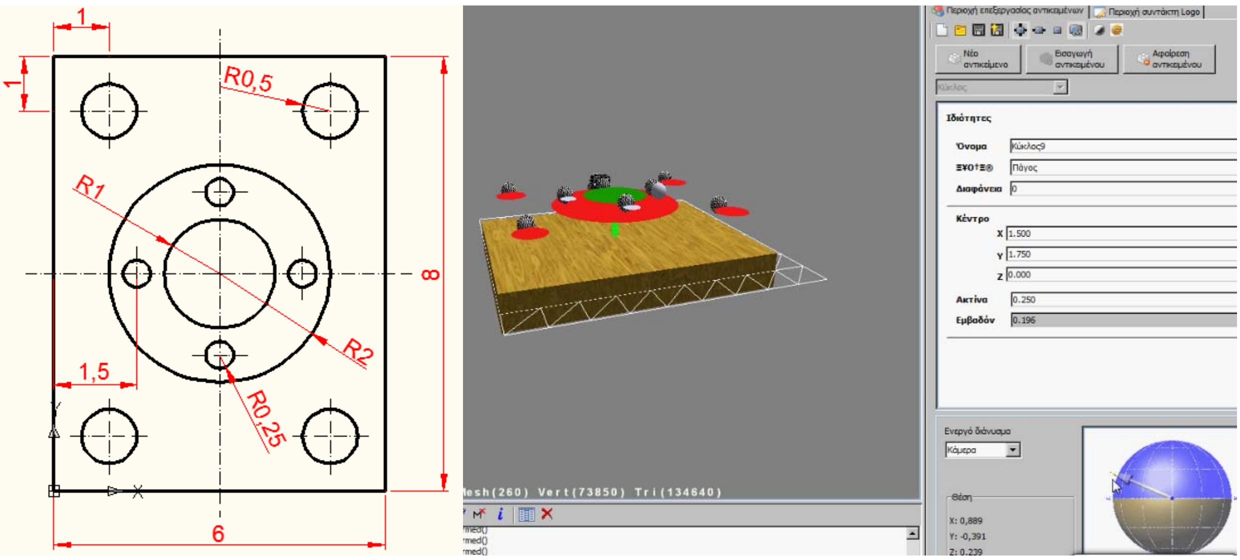


Figure 4. The 2d orthogonal projection of the component and the students' initial 3d representation of it

Using Cartesian and Turtle geometry to explain movements in 3d space

At the third phase of their experimentations, the students were introduced to a half-baked Logo microworld representing a "planing across the cuboid's XZ edges" machining procedure. Planing is a common machining procedure during which a cutting tool removes material from a piece of metal, moving in linear paths, and is usually performed with the intention generate accurate and flat surfaces. However, in this microworld, as Figure 5 shows, the cutting tool, after executing the first edge's planing, starts moving around in the 3d space in a random way, ending up plunged deep inside the component.

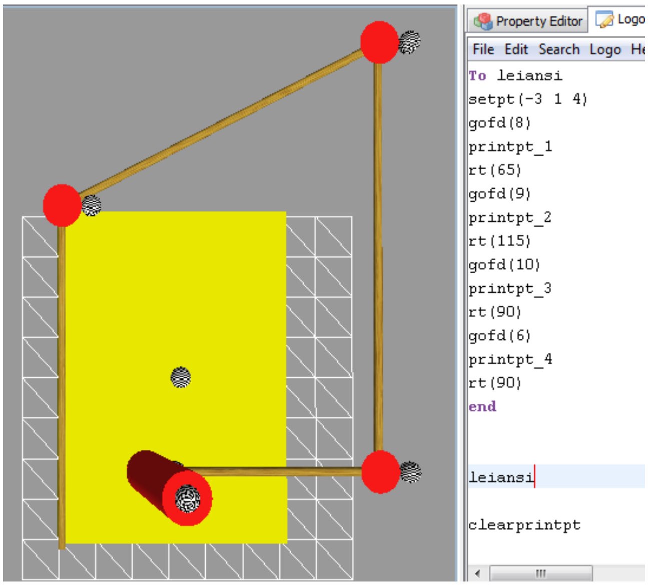


Figure 5. The graphical result of the half-baked "planing across the edges of the XZ plane"

S2: WHAT IS THAT? [Laughter]

R: It was supposed to plan all four edges.

S2: At least THERE ARE 4 edges! [more laughter]

S3: We'll have to start from the beginning!

S2: We should CORRECT it right from the beginning!

As they attempt to correct the Logo procedure the students start talking about Cartesian geometry coordinates and Turtle geometry commands:

R: Where does the tool begin its movement?

S2: [reading the setpt command] X= -3, Z = 4. Here! [points at the lower left corner]

R: What happens next?

S2: Go forward 8.

S1: It goes to X -3 and Z... [takes a moment to add 8]...12….? It should be minus 4!!!

S2: It goes here [points at the upper left corner]. That's X=-3 Y=0.5 Z=12. It's another way to say that from the zero point [the previous one] it moved 8 in forward direction.

S3: Yes, but didn't it move in the upwards direction? There is where the negative Zs are. Measure it from the centre. Minus 4. 8 is the whole piece.

Decomposing the Logo program to generate a correct planing procedure, the students use the "Turtle Geometry" commands so as to specify the tool's Cartesian coordinates as it moves inside the 3d space. Logo environments have been considered to promote spatial thinking (Clements and Battista, 1992) as Turtle Geometry provides a different way of measuring and moving (in the 2d plane or 3d space), complementary to Coordinate Geometry (Abelson and diSessa, 1980). The students, working with a microworld where manipulating an object (the tool) and moving it inside the 3d space is not matter of just pressing a button, recognize a global frame of reference ("X= -3 Y = 0.5 and Z = 12") and a local one ("8 in the forward direction from zero point") that they use interchangeably to navigate inside the 3d space.

Discussion

Mathematics educators for decades now develop and evolve dynamic computational environments that employ dynamic visualization of external representations as means for exploring and understanding mathematical concepts, both in Algebra and Geometry (CAS, DGEs, microworlds for constructionist learning). Visualization and spatial thinking in engineering education, however, are viewed as processes independent of the use of external representations and when computational environments are used, the black-boxes they contain slim down the opportunities for authentic 3d space explorations.

Computational environments, however, that incorporate the use of dynamic visualization and are developed under a constructionist perspective -along with purposefully designed collaborative activities- seem to provide a rich context for engineering students, allowing them to mobilize, enhance and develop elements of their spatial and visualization abilities. The dynamic manipulation of 3d objects inside these environments, as well as the dynamic manipulation of the viewpoint from which the students may observe their constructions, may trigger visualization processes connected with the spatial orientation inside the 3d space and the formation of spatial relationships among constituent parts of 3d configurations of objects. Half-baked microworlds (Kynigos, 2007), may foster genuine explorations in the 3d space, as their main characteristic, the fact that they are incomplete by design, challenges students to decompose them and built on their parts, so as to make them work "correctly" or to produce a different artefact according to their own conceptualizations of the 3d space. Logo procedures provide a symbolic notational system that brings to light obscure mathematical procedures professional environments box up under UI buttons, degrading visualization processes that could be fruitful for engineering students. It goes without saying that further research on constructionist dynamic computational environments designed to support engineering students' visualizations in the 3d space is needed to enrich these findings.

References

Abelson, H. and diSessa, A. (1980) *Turtle Geometry. The Computer as a Medium for Exploring Mathematics*. MIT Press, Cambridge MA.

Arcavi, A. and Hadas, N. (2000) *Computer mediated learning. An example of an approach.* International Journal of Computer for Mathematical Learning, 5(1), 25–45.

Bishop, A. (1980) *Spatial Abilities and Mathematics Education - A Review*. Educational Studies in Mathematics. 11, 257-269.

Christou, C., Jones, K., Pitta-Pantazi, D., Pittalis, M., Mousoulides, N., Matos, J. F., Sendova, E., Zachariades, T. and Boytchev, P. (2007) *Developing student spatial ability with 3D software applications*. Paper presented at the *CERME5*, Larnaca, Cyprus, 22-26 Feb 2007.

Clements, D.H. and Battista, M.T. (1992) *Geometry and spatial reasoning*. In Handbook of Research on Mathematics Teaching and Learning, D. Grouws (ed), Macmillan, NY, pp. 420-464.

Cobb, P., Confrey, J., diSessa, A., Lehrer, P. and Schauble, L. (2003) *Design experiments in educational research*. Educational Researcher, 32(1), 9–13.

Cuoco, A., Goldenberg, E.P., and Mark, J. (1996) *Habits of mind: an organizing principle for mathematics curriculum.* Journal of Mathematical Behavior. 15(4):375-402.

Ferguson, E.S. (1992) *Engineering and the Mind's Eye*, MIT Press, Cambridge, MA.

Gravemeijer, K., Cobb, P., Bowers, J. and Whitenack, J. (2000) *Symbolizing, modeling, and instructional design.* In *Communicating and symbolizing in mathematics*: *Perspectives on discourse*, *tools*, *and instructional design*. P. Cobb, E. Yackel and K. McClain (eds). Lawrence Erlbaum Associates, Mahwah, NJ, pp. 225–273.

Gutiérrez, A. (1996) *Vizualization in 3-dimensional geometry: In search of a framework*. In Proceedings of the 20th PME conference. Edited by L. Puig and A. Guttierez. Valencia, Universidad de Valencia. pp. 3-19.

Hershkowitz, R. (1990) *Some psychological aspects of learning geometry. In Mathematics and Cognition: A Research Synthesis by the International Group for the Psychology of Mathematics Education.* Nesher, P. and Kilpatrick, J. (Eds), Cambridge, Cambridge University Press.

Kahn, K., Noss, R., Hoyles, C. and Jones, D. (2006) *Designing digital technologies for layered learning*. In Proceedings of International Conference in Informatics in Secondary Schools – Evolution and Perspectives. Edited by R. T. Mittermeir. Springer, Berlin, pp. 267-278.

Kent, P. and Noss, R. (2001) *Finding a role for technology in service mathematics for engineers and scientists*. In The Teaching and Learning of Mathematics at University Level: An ICMI Study. D. Holton (Ed.), Kluwer Academic Publishers, Dordrecht.

Kynigos C., Alexopoulou E. and Latsi M. (2007) *Three dimensional constructions using an absolute frame of reference in a computer simulated 3D space*. In Proceedings of 5th CERME. Edited by D. Pitta – Pantazi and G. Philippou. Cyprus, Larnaca. pp. 1429-1438.

Kynigos, C. (1991) *Can children use the turtle metaphor to extend their learning to include non-intrinsic geometry?* In Proceedings of the 15th International Conference for the Psychology of Mathematics Education, pp. 269 – 276.

Kynigos, C. (2007) *Half–baked Logo microworlds as boundary objects in integrated design*. Informatics in Education, 6(2), 335–359.

Kynigos, C. and Psycharis, G. (2003) *13 year-olds meanings around intrinsic curves with a medium for symbolic expression and dynamic manipulation.* In Proceedings of the 27th PME. Edited by N.A. Paterman, B. Dougherty and J. Zilliox, Hawaii, U.S.A. pp. 165–172.

Kynigos, C. and Latsi, M. (2007) *Turtle's navigation and manipulation of geometrical figures constructed by variable processes in 3D simulated space*. Informatics in Education, 6, 359-372.

Leopold, C., Gorska, R., Sorby, S. (2001) *International Experiences in Developing the Spatial Visualization Abilities of Engineering Students*, Journal for Geometry and Graphics, 5(1), 81-91.

Parzysz, B. (1988) *'Knowing' vs 'seeing'. Problems of the plane representation of space geometry figure.* Educational Studies in Mathematics, 19, 79-92.

Potter, C.S. and Van Der Merwe, E. (2003) *Perception, imagery, visualization and engineering graphics.* European Journal of Engineering Education, 28, pp. 117 – 133.

Presmeg, N. (2008) *Spatial abilities research as a foundation for visualization in teaching and learning mathematics.* In Critical Issues in Mathematics Education. Major Contributions of Alan Bishop. N. C. Presmeg and P. C. Clarkson (eds). Springer, New York

Sorby, S.A. and Baartmans, B.J. (2000) *The development and assessment of a course for enhancing the 3-D spatial visualization skills of first year engineering students*. Journal of Engineering Education, 89(3), 301-307.

Zimmerman, W. and S. Cunningham (1991) *Editor's introduction: What is mathematical visualization?* In Visualization in Teaching and Learning Mathematics. W. Zimmerman and S. Cunningham (eds.), Mathematical Association of America, MAA Notes Series, pp. 1–8.

With Heart Upon My Sleeve

Robert W. Lawler, *LawlerEmail@gmail.com*
P. O. Box 726, Lake Geneva WI 53147 USA
for Seymour Papert, with appreciation and gratitude

Abstract

Case study analysis is a central method for those who understand development of mind as a process of self-construction. The credibility of the case method is held suspect, however, because study details often are kept secret “to protect the privacy of the subject.” Cynics wonder if it is not to protect the analyst from criticism.

We can resolve this issue directly. I intend to advance micro-genetic analysis of case study materials using public digital and communication facilities to share extensive case study corpora and existing interpretations. I hope as well to develop new interpretations in collaboration with others. I am building a web-resident archive of video and text materials based on three child development corpora created at the dawn of the personal computer era. My ownership of the materials, digital storage cost declines, and maturing internet technologies join in making this initiative possible. In summary, there will be text and video archives, organized around the three existing corpora; web log software will streamline communication with archives users, and wiki based interactions will begin more focused and intense collaborations.

Why should anyone be interested in these particular case studies?

First, the approach is more “anthropological” than experimental or clinical. Like Levi Strauss, I take seriously the concreteness of knowledge in everyday experience.

Second, the studies were Artificial Intelligence inspired explorations of how every day interactions through particular experiences changed what was in a child’s mind. The model came from Minsky’s famous “frames paper.” The cluster of ideas argued that one might look at the details of everyday behavior to characterize what was in the mind and then trace experiences in detail so that when a performance break-though appeared one could represent significant learning as the establishment of a single new link between formerly un-integrated elements into a system of frames. Such an interpretation of Learning Case 2 (the study of Miriam) was advanced in “Computer Experience and Cognitive Development.”

Third, LC3 was designed to explore another Minsky claim – that the structure of relations in language follow and reflect the structure of knowledge about objects and the manipulation of objects developed earlier. LC3 traces object knowledge development and language development. Interpretations will proceed separately, after which links and relations between the two streams of development will be examined.

Fourth, the historical context of the studies was novel. The learning of these three children was followed before and through the dawn of the personal computer age. There were no personal computers during LC1 and LC2 – yet the focus was how computing could affect learning. The computer used in LC3 was the TI-99 prototype Logo module development system. Coverage of the computing experience of these children was complete, because no one else had computers then.

The outcomes of these studies to date have been reported in books and articles, all of which are being made available at the web-site, http://www.NLCSA.org

Keywords

learning; case study archives; Artificial Intelligence; MIT Logo Project.

With Heart Upon My Sleeve

Robert W. Lawler

Common Sense Knowledge and Case Studies of Learning

The roots of my case study initiative were ideas and examples from Artificial Intelligence and Genevan Psychology. Minsky, MIT AI lab founder, focused on understanding the developing control structure of the mind; I followed his lead. Papert said Piaget's most profound work was in his case studies and that Psychology would divide eventually into separate domains of brain science and epistemology, the latter ultimately to be genetic in Piaget's sense. Robert White's case studies in **Lives in Progress** was a long familiar inspiration. An integrating idea was John Flavell's suggestion for a new research endeavor,[i] uniting Piaget's explanatory structuralism with the detailed "Ecological Psychology" idiographic studies of Barker and Wright.

Where Artificial Intelligence laboratories at Stanford and Carnegie Mellon focused on expert knowledge, Minsky argued that common sense knowledge was key to the flexibility and robustness of human intelligence. My focus became "natural learning," which I held were those processes creating common sense knowledge. The MIT Logo Project became my professional and intellectual home for six years, and Papert and Minsky have remained my colleagues since. In that lab, I developed computer learning environments and explored their appeal to my children and their impact on them. We called these environments computer based "micro-worlds."

My interpretation model derived from Minsky's "Frames Paper." He explained the quickness of human thought (despite a 300 mille-second brain cycle time) by the postulated existence of "systems of frames" (large scale data structures in the mind) switching between members of which permits rapid changes of perspective. I suggested that one might look at everyday behavior to characterize what was in a mind and then trace experiences in detail so that when a performance break-through appeared, one could represent learning as the establishment of a new links between formerly unintegrated frame-like elements into a system of frames. Learning Case 2 was so interpreted in "Computer Experience and Cognitive Development."

My case studies began as explorations in "AI-inspired Psychology," focused on using procedure oriented ideas to illuminate the nature of knowledge and its development: most especially, how one can understand such remarkable learning as humans exhibit. I collected and have preserved an enormous amount of material and case study observations. Advances of technology in the past thirty years, copyrights reversion, and my ongoing digitization of materials make it possible to share that archive. I will sketch the range of material, its use in theory development, and access to it.

Methodology Issues

Kurt Lewin argued[ii] that psychology can become a modern science only as researchers shift their focus from finding cross classificatory correspondences to developing fully explicit explanations for series of events in concrete cases; he recommends less abstraction and more "problem solving." Lewin argued as well for what he called "the pure case," as an ideal of individual study, a corpus with enough information at a sufficiently fine grain to resolve the issues it bears on. My case studies and the archive is in this spirit, aimed to:

- capture detailed information about individuals (in three separate cases)
- convert that corpus to an on-line database
- link related events and model development within the corpus
- offer access to that database of materials and interpretations for scrutiny by colleagues to enable criticism and envigorate development of alternatives.

The first benefit of such openness discriminating between the idiographic focus of the content and idiosyncratic interpretations. Further,case study has been less used than its potential warrants because materials are typically kept private “to protect the privacy of the subjects.” This is a legitimate concern, but secrecy makes the work suspect and inhibits legitimate criticism.[iii] For the method to become credible, **practitioners** need to open their entire corpora for examination by critics, they **need to wear their hearts upon their sleeves**, within constraints that do respect the privacy needs of their subjects. **My objective** in creating the Natural Learning Case Study Archives **is to practice what I preach**, hoping it may be of value to others who have today better opportunities, resources, even better ideas than I have.[iv] Figure 1 sketches the Implementation Structure using available web technology as a content management and collaboration base.

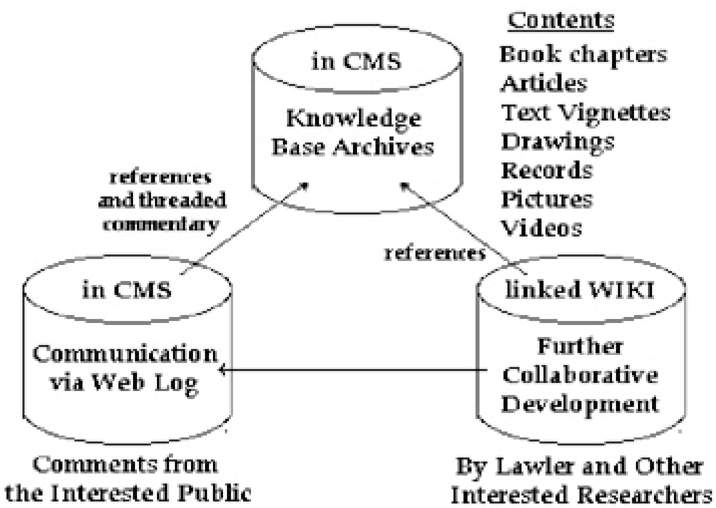


Figure 2. NLCSA implementation as a Content Management System & Wiki

NLCSA: An Internet-Accessible Case Study Archive

The **Natural Learning Case Study Archive** is built around three individual studies, although there is significant overlap of ideas, activity, and observational materials. **Table I** summarizes the studies as individual cases:

Table 1

Learning Cases	LC1 (NL)	LC2 (TIS)	LC3 (IPS)
title	Natural Learning	The Intimate Study	Infant Peggy Study
subject	Rob	Miriam	Peg
ages	6-8 years	5-8 years	18 weeks – 6 years
themes	natural learning; computing’s impact; mathematical ideas	natural learning; computing’s impact; learning arithmetic; Logo geometry; Programming Tictactoe strategies	natural learning; beginning and extension of the object concept; language learning; computing’s impact

books, articles	5 chapters, 2 articles, Logo ideas column	5 chapters, 2 articles, appendices	2 articles, 2 popular articles
text observation	21 protocols;+ TIS	133 vignettes	795 vignettes
video sessions	31 in TIS; also LC3	49 in TIS; also LC3	~225 in IPS
extent digitized	in process, most	in process, most	in process, most

Specific Outcomes of LC1: Natural Learning

Rob adopted computing as a new medium in which he could create things that satisfied his own interests[v] and in which I could make games[vi] that he would enjoy. Tracking his graphical constructions[vii] gave me confidence that I could, because of my access to and involvement with his life, trace his developing objectives in long projects and that a "cognitive anthropology" grounded in detailed case study was appropriate to understand natural learning.

- here will be presented a short video clip connecting LC1 with the well known turtle geometry activities of the Logo Project; specifically, Rob using Logo to create symmetrical inspis, whose angles of turning are prime numbers (from **TIS46**)

< estimated **duration**, with commentary, **5 minutes**.[viii]

Rob's speed of development and the depth and breadth of interests were expanding so rapidly I could not "keep up with him" in all areas, as I believed was essential. My solution to this dilemma was refocusing research on my second child, Miriam. With my wife carrying our third child, Rob and Miriam were in my care during much of 1977. Further, Miriam suffered from allergic asthma, which was relieved by air conditioning and the computer lab, which had reduced use between spring finals and fall startup. Very importantly, to avoid Miriam "displacing" Rob in our times together, I decided to collect data on the activities and development of both. We three plunged into "The Intimate Study" as a research team. While I provided resources, focus, and ideas, they contributed abundant energy, good humor, and their own ideas as data. They had considerable control over our activities.

Specific Outcomes of LC2: The Intimate Study

The breadth of material collected and interpretations offered ranged from the homely, anecdotal "Making Jokes and Learning"[ix] to the detailed and analytical studies of a child's introduction to Logo programming,[x] the intersection of computing experience and everyday life,[xi] multiple descriptions of "the same thing,"[xii] and the development of strategic thinking.[xiii] These were augmented with other data including a Binet test, a Piagetian profile, and school like materials.

- video clip from **TIS04**: Miriam; Piagetian "velocity experiment," (Papert)

< estimated duration three minutes

- video clip from **TIS47**: Miriam; Debugging her "Person" procedure, (Bob)

< estimated duration three minutes; jokes as alternative to analysis

- video clip from **TIS65**; Miriam; Debugging Bob's bugs in "jumping rope"

< duration 3 minutes: programming terminology for describing processes

These videos are available in the Case Study Archives as well as texts, graphics, and the remaining video materials listed in Table I. Though not fully analyzed, some of Rob's work in TIS which was especially valuable was published in the Journal of Mathematical Behavior. That material is in the Case Study Archive, as well as other articles I published in JMB.[xiv] and materials listed in Table I are available also.[xv]

Specific Outcomes of LC3: Infant Peggy Study

Discussing my data collection for the Intimate Study late in 1977, Professor “Mimi” Sinclair[xviii] lauded the effort but thought it would be impossible to complete the analysis to earn a doctorate for the work. Returning the next semester, she reported discussing my project with Piaget, who said he envied Papert for having a student with the taste and energy for such work. After others first raised the idea of studying the development of my infant daughter Peggy,[xvii] I discussed with Sinclair beginning a nonintrusive study of her language development, designed to place it in the context of developing interactions with the physical world (objects, space, animals, and people).[xviii] I wanted to develop, for this idiographic collection, a “spine of observations” to permit calibration of this child’s development with the body of developmental studies. Mimi proposed ramifications of the object concept, with a focus on inclusions within cavities of convex objects.[xix] I assembled a collection of toys Peggy then played with on camera, every week, for three years. Begun at 18 weeks, the Infant Peggy Study continued for six years, and ranged from social interaction to playing with blocks, nesting objects, and even computer microworlds, reading, and finger counting. Here are some samples:

Objects focus:

- IPS video clip P53E: infant Peggy putting objects “on top of” and “in” others

< duration, 3 minutes: without such distinctions, can her goals be specific?

- IPS video clip P146F&G: toddler Peggy inserting nesting cups and boxes

< durations, 5 minutes: climax of a long developmental sequence

Language focus:

- IPS video clip P26A1, infant Peggy with Bob

< duration, 2 minutes; vocal interactions with singing

- IPS video clip P65C, infant Peggy with Bob, “bring Hanky” (she does)

< duration, 3 minutes; verbal comprehension or situation analysis

- IPS video clip P104A; toddler Peggy conversation; her control of activities

< duration, 3 minutes; complex thoughts and simple verbalization

Symbol manipulation focus:

- IPS video clip: Paris TV: Miriam and Peg; introduction by Papert et alia.

< duration, 5 minutes of word worlds in French for “English” children

- IPS video clip G11B: child Peggy, addition, with fingers time permiting

< durations, 5 minutes; joining IPS study to Miriam’s behavior in LC2

The materials in Table I are now in or being added to the corpus.

LC3 continued despite my leaving MIT to work in New York and then in Paris, at Mitterrand’s Centre Monidal L’Informatique. After wonderful years in Paris, my family returned to New England, and Minsky guided me to work with Oliver Selfridge, forming a new AI group in GTE’s Fundamental Research Lab.[xx]

Access to the Archives

The web site at NLCSA.org is public. Navigation is straightforward. Though the video material is presented here as samples – small webstreamable Quick Time clips, behind these samples are the full digital videos, stored offline, on multi-terabyte hard drives.[xxi].

References

Lawler, R. *Computer Experience and Cognitive Development*. Chichester, England,and New York: Ellis Horwood, Ltd. and John Wiley Inc., 1985.

Lawler, R., DuBoulay, Hughes, and Macleod. *Cognition and Computers.* Chichester, England, and New York: Ellis Horwood & John Wiley Inc, 1986.

Lawler, R. “Making Jokes and One Child’s Learning,” in *Humor, the International Journal of Humor Research*, Vo. 2-3, Mouton de Gruyter, 1989.

Lawler, R. “On the Merits of the Particular Case,” in Lawler and Carley, *Case Studies and Computing*. Norwood, N.J.; Ablex Publishing, 1996.

Lawler, R. “Consider the Particular Case” *Journal of Mathematical Behavior*, 1998

Lewin, Kurt. The Conflict Between Galilean and Aristotelian Modes of Thought in Contemporary Psychology. In *Dynamic Psychology: Selected Essays of Kurt Lewin*, D. Adams and K. Zener (translators.) McGraw-Hill. New York; 1935.

Minsky, M. A Framework for Representing Knowledge, in *The Psychology of Computer Vision*, (P. Winston, ed.), McGraw Hill, 198n.

Minsky, M. *The Society of Mind*, Simon & Schuster, NY, 1986.

White, Robert W. *Lives in Progress*, Harcourt, Brace, NY, 1975.

Acknowledgements

Since my intellectual debts to colleagues are beyond measure, here I will direct your attention to the video sample “enduring colleagues” on the NLCSA web site and ask you to recognize that there are many others to whom I owe gratitude and recognition.

Help creating this corpus came in the first instance from Seymour Papert, who committed to my full time possession the video equipment I used. Not less important was Oliver Selfridge, who supported my first transfer of the LC3 corpus from reel to reel video to VHS format. With the oldest materials, recently I used the services of Rick Thomas, of Queen Creek, Arizona, who rescued videos that otherwise would have been lost. To him, thanks, and a recommendation to any who need video recovery help. (http://obsoletevideoservice.itgo.com/home.html)

Notes:

i. in an Appendix to *The Development Psychology of Jean Piaget*, (1972).

ii. Lewin (1935). The points are discussed in considerably more detail in "On the Merits of the Particular Case," chapter 1 of *Case Study & Computing, Lawler and Carley*, 1996.

iii. For further discussions of privacy see pp. 72-83 of Lawler and Carley, 1996.

iv. Such facilities could also provide an experimental workbench for advanced students to undertake a kind of apprenticeship in case analysis under tutelage of the case database developer.

v. One of his favorites then was a graphics components assembly environment (EEL) based on the children's drawing books by Ed Emberley.

vi. Rob spent a lot of time with "Ready, Aim, Fire," (RAF), in which I had superposed a gun sight and coordinate grid cross-hairs on the turtle geometry domain and a minor variation of Paul Goldenberg's Shoot program. When declared an "ace" after shooting down five planes, Rob made it is goal to get more kills than Baron Von Richtofen.

vii. In "The Development of Objectives," chapter 1 of *Computer Experience and Cognitive Development* (Lawler, 1985)

viii. A report of this work was published in the Journal of Mathematical Behavior, with the title "Extending a Powerful Idea."

ix. Published in the *International Journal of Humor*, 198n.

x. "The Equilibration of Cognitive Structures," chapter 3 in Lawler, 1985.

xi. "The Progressive Construction of Mind," chapter 2 in Lawler, 1985.

xii. "Cognitive Organization," chapter 5 in Lawler, 1985

xiii. "The Articulation of Complementary Roles," chapter 4 in Lawler, 1985.

xiv. "Extending a Powerful Idea," the *Journal of Mathematical Behavior*, date.

xv. This includes my four chapters in *Cognition and Computers*, (Lawler, DuBoulay, Hughes, and Macleod, 1986).

xvi. Hermine Sinclair deZwart, of the Faculty of Psychology and the Science of Education at the University of Geneva was "Piaget's Linguist." At MIT, she was a visiting professor in the Division for Study and Research in Education.

xvii. Meltzoff had recently used infant studies to attack Piaget's claims about the inception of imitation. After discussion with my wife, we agreed at first, but withdrew that agreement when we saw that trying to avoid affecting the results was limiting our interactions with the baby.

xviii. This followed Minsky's position that language was an aspect of human development profoundly affected by prior knowledge developed through interaction with objects and people and that linguistic structures were consequences of that prior knowledge.

xix. This was related to research by Stanback in Paris, in *Bebes et Choses*, 198n.

xx. Minsky said Oliver had the quickest mind of anyone he had ever known, that he had a genius for undertaking deep studies with simple computational models, and that his interest in children's learning was as committed as my own. Minsky was right on all counts, and I had the deep honor to become Selfridge's colleague for the rest of his life. For that work, see "Explorations in Experimental Epistemology, Constructionism 2010."

xxi. If you are engaged in a research project where the higher quality video would be of value to you, please contact me. We will make files available.

Explorations in Experimental Epistemology

Robert W. Lawler, *LawlerEmail@gmail.com)*
P. O. Box 726, Lake Geneva WI 53147 USA
for Oliver G. Selfridge, with gratitude and admiration

Abstract

This paper discusses the epistemology of learning in two different dimensions.

First, Learning through Interaction focuses on learning with concrete methods in a particular domain (learning strategies for playing Tic tic toe) with a machine learning study grounded in a human psychological case study (see "The Articulation of Complementary Roles," chapter 4 in Lawler, 1985).

The methods of learning in the computer modeling employ learning by example (Winston, 197n) and learning by debugging (Sussman, 197n). Codifying results and re-representing those results characterizes the learning (as a function of initial knowledge state, learning mechanisms invoked, and opponent actions) as a "network of genetic descent." This permits two novelties.

Although grounded in an individual's study, the modeling shifts focus to an epistemological question, "what kinds and paths of learning are possible." The outcomes are a new principle, "the learnability of a domain is the result of all the possible cases of concrete learning through particular experiences," and a new criterion, that the co-generability of related but variant knowledge forms is what makes learning possible in any particular domain.

Second, Escaping from Particularity, motivated by the inability of the machine learning model to encompass symmetry, confronts directly the issue of true novelty in learning in the "strategic aside," ***Nil Ex Nihilo***. Explaining the interaction of different modalities of mind as resulting in an effective abstraction by redescription, the proposed 'Multi-Modal Mind" is advanced as a context in which Redescriptive Abstraction serves as a precursor of Piaget's Reflective Abstraction and as a viable candidate for the human mind's general developmental mechanism. This may suffice not only for Piaget's "spiral of learning" but more importantly, may help us understand "the helix of mind arising."

Keywords

learning; epistemology; abstraction; machine learning; strategy learning; case study;

Explorations in Experimental Epistemology

Robert W. Lawler

During six years at the MIT Artificial Intelligence Laboratory with Minsky and Papert, I collected detailed case study material about the learning of three of my children.[1] On returning from several exciting years at Mitterand's World Center for Computers and Human Development, with Papert and Negroponte, at Minsky's urging I joined Oliver G. Selfridge in a new AI Research Group at GTE's Fundamental Research Laboratory.[2] There, I had the opportunity, with the guidance of Selfridge and some help from Bud Frawley, to construct learning models reflecting ideas and observations from my case studies.

That modeling permitted solving some long-standing problems in cognition and epistemology. Questions answered positively in the interpretation and modeling of the LC2 complete Tic tac toe corpus [3] are:

• how one can apply “learning by example”[4] and “learning by debugging”[5] to what people do.

• why some kinds of knowledge can be learned from concrete experience while other kinds of knowledge cannot.

• how one can escape from the particularity of experiential knowledge to more abstract and general knowledge.

But first "why TicTacToe?" you might ask, and "What does 'complete' mean?" A grand theme of the Newell-Simon AI initiative had been the nature of expertise, as realized in Chess Mastery. 3T (as Selfridge referred to TicTacToe) was a game of competitive strategy within the reach of an unschooled 6 year old. The corpus was complete in preserving every game she played in the study from its first introduction through its end.

Learning through Interaction

If learning is an adaptive developmental mechanism, that adaptation must comes from interaction with the everyday world. To follow the natural learning of common sense knowledge, a case-based approach is best to track unpredictable learning. Further, if learning is a process of changing one state of a cognitive system to another, based on interactions, then representations of that process in computing terms are most appropriate. But how can one use case material in modeling? As one uses boundary conditions to specify the particular form of a general solution to a differential equation. Such is the use made of the psychological studies here, as a foundation for the representations used in the models, and as justification for focusing on key issues: the centrality of egocentricity [6] in self-construction and the particularity of the naïve agent's knowledge.

The virtue of machine learning studies is that they allow us no miracles; they can completely and unambiguously cover some examples of learning with mechanisms simple enough to be comprehensible. Must we claim that learning happens in people the same way? No. Building such models is an exploration of the possible, according to a specification of what dimensions of consideration might be important. The computer's aid in systematically generating sets of all possible conditions helps liberate our view of what possible experiences might serve as paths of learning. When we can generate all possible interactions through which learning might occur, including some we first imagine are not important, we can explore alternate paths and the suite of relationships among elements of the ensemble.

Considering All the Possibilities

The model began as a verbally formulated analysis of one child's learning strategic play at tic tac toe (The Articulation of Complementary Roles, Chapter 4 in Lawler, 1985). It was continued in a constructive mode through developing a computer-embodied model, SLIM (Strategy Learner, Interactive Model. (See Lawler and Selfridge, 1986.) The latter was based on search through the space of possible interactions between one programmed agent (SLIM) having some of the characteristics of the psychological subject and a second, REO, a programmed Reasonably Expert Opponent. REO is "expert" only in the sense of being able to apply uniformly a set of cell preference rules for tactical play.[7]

Strategies for achieving specific forks are the knowledge structures of SLIM. Each has three parts: a Goal pattern, a plan of Actions, and a set of Constraints on those actions (each triple is thereby a GAC). I simulated operation of such structures in a program where SLIM plays tic tac toe against variations of REO. Applying these strategies leads to moves that often result in winning or losing; this in turn leads to the creation of new structures, by specific modifications of the current GACs. The modifications are controlled by a small set of rules, so that the GACs are interrelated by the ways modifications can map from one to another.[8]

To evaluate specific learning mechanisms in particular cases, one must go beyond counting outcomes; one must examine and specify which forks are learned from which predecessors in which sequence and under which conditions of opponent cell preferences. The simulation avoided abstraction, in order to explore learning based on the modification of fully explicit strategies learned through particular experiences.[9] The results are first, a catalog of specific experiences through which learning occurs within this system and second, a description of networks of descent of specific strategies from one another. The catalog permits a specification of two desired results: first, which new forks may be learned when some predecessor is known; and second, which specific interaction gives rise to each fork learned. The results obviously also depend on the specific learning algorithm used by SLIM.

Consider how SLIM can learn the symmetrical variation to one particular fork. Suppose that SLIM begins with the objective of developing a fork represented by the Goal pattern {1 3 9} and will proceed with moves in the sequence plan [1 9 3] (see Figure 1). SLIM (A) moves first to cell 1. REO (1) prefers the center cell (5), and moves there. SLIM moves (B) in cell 9. The plan is followed until REO's second move (2) is to cell 3.[10]

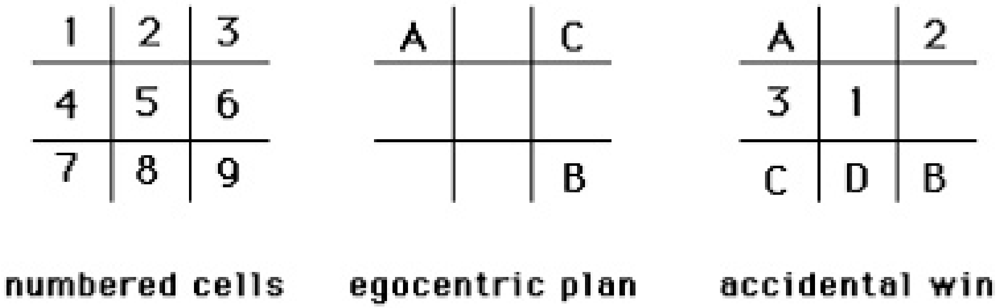


Figure 1: cells, a plan, and one played game

SLIM's plan is blocked. The strategic goal {1 3 9} is given over -- but the game is not ended. SLIM, playing tactically with the same set of rules as REO, moves into cell 7, the only remaining corner cell. Unknowingly, SLIM has created a fork symmetrical to its fork-goal. SLIM can not recognize the fork. It has not the knowledge to do so. What happens ? REO blocks one of SLIM's two ways-to-win, choosing cell 4. SLIM, playing tactically, recognizes that it can win and moves into cell 8. This is the key juncture. SLIM recognizes " winning without expecting to do so" as a special circumstance. Even more, SLIM assumes that it has won through creating an unrecognized fork (otherwise REO would have blocked the win). SLIM takes the pattern of its first three moves as a fork. That pattern {1 7 9} is made the goal of a new GAC. SLIM examines its known plans for creating a fork (there is one, [1 9 3]) with the list of its own moves, executed in sequence before the winning move was made [1 9 7]. The terminal step of the plan is the only difference between the two. SLIM modifies the prototype plan terminal step to create a new plan, [1 9 7]. SLIM now has two GACs for future play.[11]

The complete set of results involves consideration of all paths of learning, even those deemed unlikely a priori, and concludes with the complete specification of all possible paths of learning every fork given any fork prototype. For corner opening play, the first six GACs form a central collection of strategies. Their interrelations can be represented as trees of derivation or descent (shown in Figure 2). The tree with strategy three as top node may be taken as typical. Play in five specific games beginning with only GAC 3 known, generates the other five central GACs. For these six central strategies, the trees of structure descent can fold together into a connected network of descent whose relations of co-generativity are shown in Figure 3. The specialness of the six central nodes is a consequence of co-generability. Some of those are directly generable, can generate each other (such as GACs 1 and 2) ; they are reciprocally generable (solid lines). Some lead to each other through intermediaries (GACs 1 & 3); they are cyclically generable (dashed).

The form of these descent networks is related to symmetry among forking patterns. But they include more: they reflect the play of the opponent, the order in which forks are learned, and the learning mechanisms permitted in the simulations. These descent networks are summaries of results.

The experimental epistemology of SLIM begins with a focus on detail

• in the analysis of specific cases and

• in the analysis of the interaction of objects or agents with their context.

The basic principle applied is to try all cases and construct an interpretation of them. To predict the learning of a specific strategy by a human subject, one would need to know what strategies are already known, how the opponent's play would create opportunities for surprising wins for the subject, and what learning methods are in the subject's repertoire. Knowing these things in the machine case is what permits examination of the epistemological space of learnable strategies. In the analysis of SLIM, one begins with lists of games won without a plan. One then reformulates the relations between prototypes and generated plans into trees of fork plan transformations, which are the trees of descent. The learning algorithms are the functional mechanisms effecting the transformations. Aggregation is systematic and constructive though not formal: one pulls together the empirical results of exhaustive exploration (trees of descent in figure 2) into a new representation (the genetic descent network, figures 3).[12]

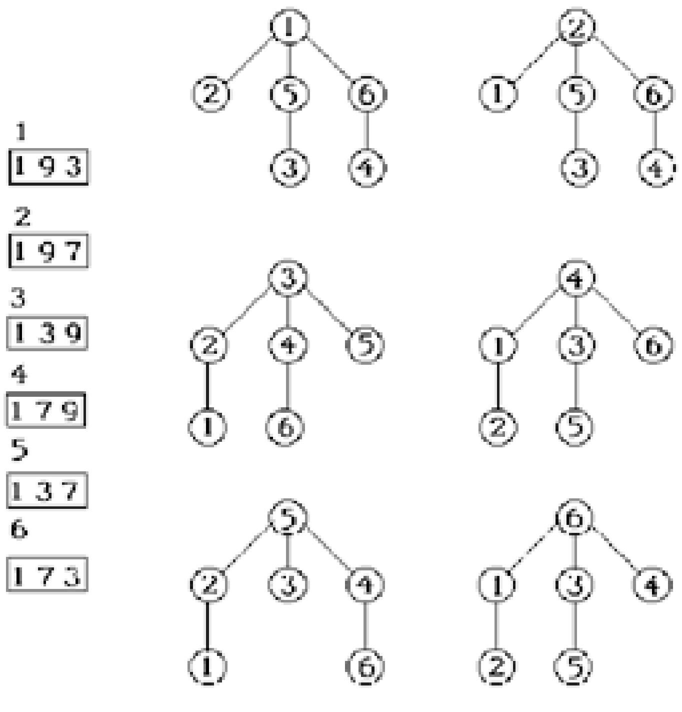


Figure 2: Plans Learnable from the Top Node Plan

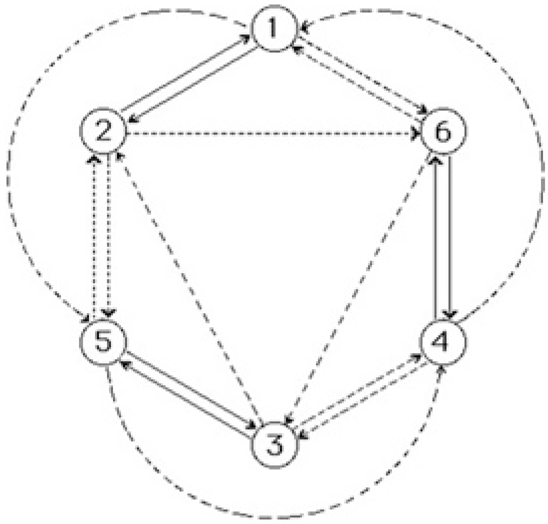


Figure 3: derivation tress folded into a network

The learnability analysis of this paper introduces two novelties: a new principle and a new criterion. Start with an epistemological stance instead of a psychological one. Here one is not so much interested in what a particular child did as an individual. Yet the individual case provides boundary conditions for modeling with a question "if the details of at least one natural case have such characteristics, what kinds and paths of learning are possible?" The question is of general interest if one admits that particularity and egocentricity are common characteristics of novice thought.

SLIM started with the general principle that learning happens through interaction. The model is “mental;” it represents the behavior of both the learner and the opponent in explicit detail with specification of representations and learning algorithms giving the notion a precise meaning. The new principle is that the learnability of a domain is the result of all the possible cases of concrete learning through particular experiences. Co-generativity permits each central strategy (1- 6) to be learned no matter which is adopted as a prototype fork. This suggested a new criterion, that co-generability of related but variant knowledge forms is what makes learning possible in any particular domain.[13] Such contribute significantly to the learnability of a domain because they are mutually reinforcing. Peripheral strategies[14] are rarely learned because they can be learned in few ways. Thus, one can characterize the learnability of a domain as a function of particular interactions among agents based on the connectedness of possible paths of strategy learning. That is what the genetic descent network does. Furthermore, these methods and representations even make it possible to judge that knowledge of a given domain is more learnable than another.

Escaping from Particularity

Most people seem comfortable with the symmetries of 3T, which greatly simplify their analyses and strategies. But the GAC representation, with its cell specific definition of pattern and plan elements, is entirely different and has no way of engaging with common sense notions of symmetry at all.

If we ask where symmetry enters such highly particular descriptions, the answer MUST involve abstraction, but which form of those kinds possible? Abstraction by feature-based classification is the most commonly recognized form, but there are others. Piaget emphasizes a kind of abstraction, focusing more on what one does rather than on qualities one attributes to external things. This reflexive abstraction is a functional analysis of the genesis of some knowledge,[15] as presented elegantly in Bourbaki's description of the generality of axiomatic systems:

"A mathematician who tries to carry out a proof thinks of a well-defined mathematical object, which he is studying just at this moment. If he now believes that he has found a proof, he notices then, as he carefully examines all the sequences of inference, that only very few of the special properties in the object at issue have really played any significant role in the proof. It is consequently possible to carry out the same proof also for other objects possessing only those properties which had to be used. Here lies the simple idea of the axiomatic method: instead of explaining which objects should be examined, one has to specify only the properties of the objects which are to be used. These properties are placed as axioms at the start. It is no longer necessary to explain what the objects that should be studied really are...."

N. Bourbaki, in Fang, p. 69.

Robust data argue that well articulated, reflexive forms of thought are less accessible to children than adults. The possibility that mature, reflexive abstraction is unavailable to naive minds raises this theoretical question: what process of functional abstraction could precede such fully articulated reflexive abstraction; could such a precursor be the kernel from which such a mature form of functional abstraction may grow?

Nil Ex Nihilo: a strategic aside

Genesis once told us "God said 'Let there be light,' and there was light." Cosmologists now say there was no visible light for 300,000 years after the beginning of our universe. They explain the novelty of light's appearance this way: after a period of 'inflation,' the expanding low entropy plasma was so hot that all particles were unstable and there were constant conversions of energy into matter and back again; the density was so great that everything always collided with everything else. As temperature fell through those 300,000 years, protons began to hold captured electrons, matter and anti-matter cross-annihilated, and the quantity of particles dropped enough that photons could escape the collisions and gravity of the still expanding plasma. THEN there was light.[16]

Photons existed from the beginning, in the parts and interactions that made up the whole. Light was revealed by no longer being obscured.

About human learning, we know that it begins with coordination of sensory and motor nerve impulses. On a later, larger scale, the need to coordinate systems of sensory-motor interactions, e.g. "eye and hand" is clear. Why should not the interplay of body-system related, interior representation schemes be invoked to explain processes of thought in pre-linguistic and even in language capable minds? Such is the strategy behind the "Multi-Modal Mind." Nothing comes from nothing. The truly novel is manifest when released from what previously obscured it.

The Multi-modal Mind

Let us discriminate among the major components of the sensori-motor system and their cognitive descendents, even while assuming the preeminence of that system as the basis of mind. Imagine the entire sensori-motor system of the body as made up of a few large, related, but distinct sub-systems, each characterized by the special states and motions of the major body parts, thus:

Body Parts	S-M Subsystem	Major Operations
Trunk	Somatic	Being here; being touched
Legs	Locomotive	Moving from here to there
Head-eyes	Capital/visual	Looking at that there
Arms-hands	Manipulative	Touching; changing that
Ears	Aural	Hearing sounds; Language
Mouth	Oral	Making sounds; Language

We assume the representations of mind remain profoundly affected by the modality of interactions with experience through which it was developed. One implication is that the representations built through experience will involve different objects and relations, among themselves and with externals of the world, which will depend upon the particular mode of experience. Even if atomic units of description (e.g. condition action rules) are shared between modes, the entities which are the salient objects of concern and action are different, and they are in relation to each other only through learned correspondences. This general description of mind contrasts with the more uniformitarian visions which dominate psychology today. These major modal groupings of information structures are imagined to be populated with clusters of related cognitive structures, called "micro-views," with two distinct characters. Some are "task-based" and developed through prior experiences with the external world; others, with a primary character of controlling elements, develop from the relationships and interactions of these disparate, internal micro-views. The issue of cognitive development is cast into a framework of developing control structure within a system of originally competing micro-views.[17]

Redescriptive Abstraction

I propose that the multi-modal structure of the human mind permits development of a significant precursor to reflexive abstraction. The interaction of different modes of the mind in processes of explaining unanticipated outcomes of behavior can alter the operational interpretation and solution of a problem. Eventually, a change of balance can effectively substitute an alternative representation for the original; this could occur if the alternative representation is the more effective in formulating and coping with the encountered problem. In terms of the domain of our explorations and our representations, there is no escape from the particularity of the GAC representation unless some other description is engaged. A description of the same circumstance, rooted in a different mode of experience, would surely have both enough commonality and difference to provide an alternative, applicable description. I identify the GAC absolute grid as one capturing important characteristics of the coordinated visual-manipulative mode;[18] other descriptions based on the somatic or locomotive subsystems of mind could provide alternative descriptions which would by their very nature permit escape from the particularity of the former.

Why should explanation be involved? Peirce argues that "doubt is the motor of thought" and that mental activity ceases when no unanswered questions remain.[19] Circumstances requiring explanation typically involve surprises; the immediate implication is that the result was neither intuitively obvious nor were there adequate processes of inference available beforehand to predict the outcome (at least none such were invoked).

We propose that a different set of functional descriptions, in another modal system, can provide explanation for a set of structures controlling ongoing activity. The initial purpose served by alternative representations is explanation. Symmetry, however, is a salient characteristic of body centered descriptions; this is the basis of their explanatory power when applied where other descriptions are inadequate. Going beyond explanation, when such an alternative description is applied to circumvent frustrations encountered in play, one will have the alternate structure applied with an emergent purpose. Through such events, the interaction of multiple representations permits a concrete form of abstraction to develop, one emergent from the application of alternative descriptions. Where does symmetry come from? The projection of body centered representations over the visual-maniplative grid.[20]

Emergent Abstraction

If alternative representations can serve as explanation for surprises developed through play, and if they can serve as a bridge to break away from the rigid formulation of the GAC representation, it is not impossible to believe they may begin to provide dynamic guidance as well -- exactly of the sort found useful by adults in their play. When this occurs, the alternative description, useful initially as an explanation for the more particular system of primary experiences, will become the dominant system for play. Then the symmetry implicit in the body-centric imagery will become a salient characteristic of the player's thinking about tic tac toe as the highly specific formulations of early experience recede into the background. Abstraction has taken place -- because the descriptions of the body mode are implicitly less absolute in respect of space than are those supposed to operate with the GAC representation. But the abstraction is not by features, nor is it by the articulate analysis of reflexive abstraction, as described by Bourbaki. This is an emergent abstraction via REDESCRIPTION, a new kind of functional abstraction. Redescriptive abstraction is a primary example of the coadaptive development of cognitive structures. As a kind of functional abstraction which does not yet require reflexive analysis of actions taken within the same mode of representation, but merely the interpretation of actions in one mode in terms of possible, familiar actions in another mode,[21] it needs bear less of an inferential burden than would the more analytic reflexive abstraction described by Bourbaki.

Redescriptive Abstraction and Analogy

One might say that emergent abstraction via redescription is "merely analogy". I propose an antithetical view: emergent abstraction explains why analogy is so natural and so important in human cognition. Redescriptive abstraction is a primary operation of the multi-modal mind; it is the way we must think to explain surprises to ourselves. We judge analogy and metaphor important because redescriptive abstraction is subsumed under those names.

Further, I speculate it is THE essential general developmental mechanism. This process can be the bootstrap for ego-centric cognitive development because accomplished without reference to moves or actions of the other agent of play.

References

Bartlett, Frederic C. ***Remembering***. Cambridge University Press, 1932.

Bourbaki, N. Excerpt from The Architecture of Mathematics, quoted on p.69 in J. Fang (1970), ***Towards a Modern Theory of Mathematics***. Hauppauge, N.Y.: Paideia series in modern mathematics, Vol. 1.

Fann, K. T. ***Peirce's Theory of Abduction***. The Hague: Martinus Nijhoff.

Lawler, R. ***Computer Experience and Cognitive Development***. Chichester, England, and New York: Ellis Horwood, Ltd. and John Wiley Inc., 1985.

Lawler, Robert W. & Oliver G. Selfridge. 1986 Learning Concrete Strategies through Interaction. The ***Seventh Cognitive Science Conference.*** U.C.Irvine.

Lawler, R. "Coadaptation and the Development of Cognitive Structures," in ***Advances in Artificial Intelligence-II***, Eds. DuBoulay, Hogg, and Steels, North Holland, Amsterdam, 1986. (Papers from ECAI, 1986.)

Lawler, R. "On the Merits of the Particular Case," in Lawler and Carley, ***Case Studies and Computing***. Norwood, N.J.; Ablex Publishing, 1993.

Lawler, R "Consider the Particular Case" ***Journal of Mathematical Behavior***, 1998.

Minsky, M. A Framework for Representing Knowledge, in ***The Psychology of Computer Vision***, (P. Winston, ed.), McGraw Hill, 198n.

Minsky, M. ***The Society of Mind***, Simon & Schuster, NY, 1986.

Peirce. Charles, S. ***The Fixation of Belief***. In Chance, Love, and Logic (M. R. Cohen, ed.). Harcourt, Brace, and Co. 1923.; also George Braziller, Inc., 1956.

Piaget, J. ***Biology and Knowledge***. Chicago: University of Chicago Press, 1971.

Piaget, J. ***The Language and Thought of the Child***. NY: New American Library.

Sussman, G. ***A Procedural Model of Skill Acquisition***, Elsevier Press, 198n.

Weyl, H. ***Symmetry***. Princeton; Princeton University Press, 1952.

Winston, P. "Learning Structural Descriptions from Examples" in ***The Psychology of Computer Vision***, (P. Winston, ed.), McGraw Hill, 198n.

Notes:

1. The materials of these case studies, published and unpublished alike, are now being assembled for public access at the web site www.NLCSA.org, as reported in the paper "With Heart Upon My Sleeve," at Constructionism 2010.

2. Minsky said Oliver had the quickest mind of anyone he had ever known, that he had a genius for undertaking deep studies with simple computational models, and that his interest in children's learning was as committed as my own. Minsky was right on all counts, and I had the deep honor to become Selfridge's colleague for the rest of his life.

3. During its construction, I referred to this corpus as "The Intimate Study." It served as the basis of two books (Lawler 1985 and Lawler et al., 1986, and other papers). Tic tac toe interpretation was completed as a post doc at MIT.

4. As in Patrick Winston's thesis "Learning Structural Descriptions from Examples."

5. As in Gerald Sussman's thesis "A Procedural Model of Skill Acquisition."

6. See Piaget, The Language and Thought of the Child.

7. REO's preferences are common in tactics: first, win if possible; block at need; finally choose a free cell: preferring the center cell first, then any corner, and finally a side cell.

8. Subject to rotational symmetry and prototypical strategy preferences of the psychological subject, the generation of games was exhaustive. Analysis focused on won-games in which SLIM moved first in cell 1. See figure 1.

9. This focus of the models is precisely where the egocentricity of naive thought and cognitive self-construction of the psychological subject are embodied. The models are "egocentric" in the specific sense that no consideration of the opponent is taken unless and until the current plan is blocked. The psychological subject played this way. When a plan is blocked, SLIM drops from strategy driven play into tactical play based on preferences for cells valued by type (center, corner) and not by relation to others.

10. The general commitment to egocentric knowledge representation has psychological justification in this specific case. Lawler's subject suffered the defeat above trying to achieve the victory of GAC 1 (the only strategy she knew), not attending to her opponent's move nor anticipating any threat to her intended fork.

11. This form of learning by modifying the last term of a plan is one of two sorts; the other involves generating two possible plans based on deletion of the second term of a prototype plan. We know the forks achieved by plans [1 9 3] and [1 9 7] are symmetrical. SLIM has no knowledge of symmetry and no way of knowing that the forks are related other than through descent, i.e. the derivation of the second from the first. This issue is discussed in the longer version of this text.

12. The complete Genetic Descent Network, with additional GACS, is shown in its fullness on page 19 of The Merits of the Particular CAse, in Lawler and Carley, 1996. In general, the process followed with these data is similar to Weyl's use of reformulation in his general description of the development of theoretic knowledge in Symmetry and Bourbaki's description of the genesis of axiomatic systems in The Architecture of Mathematics.

13. This principle would support stable knowledge in minds with reconstructive memories, such as Bartlett (1932) suggests humans have.

14. Such are those defined by other GACs discussed by Lawler and Selfridge (1986) and in “On the Merit of the Particular Case,” Chapter 1 of Case Study and Computing.

15. Piaget contrasts reflexive abstraction with classificatory or Aristotelian abstraction (p.320 in Biology and Knowledge), demeaning the latter by referring to it as "simple".

16. An elegant popular presentation by Stephen Weinberg of this speculative science is "The First 3 Mintues."

17. This view of mind is presented and applied in "Cognitive Organization", Chapter 5 of Lawler, 1985. A more extensive discussion of micro views appears in Chapter 7.

18. The GAC description is in terms of external things seen by a person referring to it. The absolute reference assigning numbers to specific cells preserves a top-down, left to right organization. Notice however, that even if one's internal representation were different -- based perhaps on a manipulative mode of thought and representation -- the essential points of following arguments remain sound.

19. Peirce's position (presented lucidly in "The Fixation of Belief" but ubiquitous in his writing) was the primary observation leading me to focus on this theme. He uses the term doubt because his discussion is cast in terms of belief; mine, cast in terms of goals, finds its equivalent expression as surprise. Doubts require evidence for elimination (but see Peirce on this); surprises require explanations. Surprise is accessible to mechanical minds as the divergence between expectation and outcome under a specific framework of interpretation.

20. See "Coadaptation and the Development of Cognitive Structures" in DuBoulay et al. ***Advances in Artificial Intelligence*** for a bit more detail of this argument.

21. The point here is that the process is more like Peirce's abduction than any inductive process of learning. See "Deduction, Induction, and Hypothesis" for Peirce's introduction to this distinction or K. T. Fann's "Peirce's Theory of Abduction" for an analysis of Peirce's developing ideas on abduction.

Encouraging Collaborative Constructionism: Principles Behind the Modeling Commons

Reuven M. Lerner, *reuven@lerner.co.il*
Department of Learning Sciences, Northwestern University, Evanston, Illinois

Sharona T. Levy, *stlevy@construct.haifa.ac.il*
Faculty of Education, University of Haifa, Mount Carmel 31905, Israel

Uri Wilensky, *uri@northwestern.edu*
Departments of Learning Sciences and Computer Science, Center for Connected Learning and Computer-Based Modeling, Northwestern Institute on Compex Systems (NICO), Northwestern University, Evanston, Illinois

Abstract

Constructionism argues that learning occurs best when constructing a public artifact (Papert 1983). NetLogo (Wilensky, 1999a) is an agent-based modeling language that has successfully been used in a variety of constructionist contexts. However, NetLogo lacks built-in support for making artifacts public, or for creating models collaboratively. Our research focuses on a Web-based tool known as the "Modeling Commons," which is designed to make NetLogo not only an effective tool for creating models, but also for sharing them with others and collaborating during the modeling process.

From our efforts and research spent creating the Modeling Commons, we have identified a number of design principles, which would appear to encourage the creation of collaborative online communities. These principles have provided us with guidelines during the design process, and have helped to focus our questions during design-based research interviews and analysis.

Briefly, our principles are: Focus on artifacts, provide multiple entry points, be forgiving, maximize findability, provide flexible permissions, and keeping users informed.

In this paper, we describe these design principles and explain why we believe that these principles are important and how they have affected our design of the system.

Keywords (style: Keywords)

Modeling; Constructionism; Collaboration; Design-Based Research; Computer-Supported Collaborative Learning

1 Introduction

Constructionism (Papert, 1980) argues in favor of learning through the creation and sharing of artifacts. NetLogo (Wilensky, 1999a) an agent-based modeling environment, has long supported individual constructionist learning (Jonassen, 2006; Reisman & Wilensky, 2006). However, NetLogo lacks built-in support for sharing models, let alone collaboratively developing them. Recent theory and evidence demonstrate the central role that social interaction plays in learning (Vygotsky, 1978; Lave & Wenger, 1991; Wenger, 1998) in general, and when modeling in paritcular (de Aennle, 2009, Bollen et. al., 2002).

In order to support and encourage interactions among modelers, we have created a Web-based complement to NetLogo, known as the Modeling Commons (Lerner, Levy, & Wilensky, 2010b). The Modeling Commons (see Figure 1) makes it possible for NetLogo modelers to share their work with others via a central Web site. The author of a model may optionally allow others to edit and update the model, providing a mechanism for collaborative modeling. Users may attach one or more supporting files to a model, such as input data, curricular suggestions, research papers, and PowerPoint presentations. Those users who visit the model in the Modeling Commons may participate in a discussion about the model, apply one or more social tags to it, or recommend it to their friends.

Anecdotal evidence indicates that many NetLogo users already share models with one another, and even collaborate with their peers while modeling. However, we believe that such interactions are less frequent than could be if collaborative tools were more easily accessible. For example, the netlogo-users e-mail list is the primary forum for discussing the NetLogo language. Over a period of seven years, containing 9,696 messages sent to this list, only 5.7% were found to contain models. NetLogo's Community Models library, hosted on the same Web site as NetLogo, has added only 274 models over the same time period.

At this time, the Modeling Commons is still being tested, with an expected launch in the coming months. It has already undergone three rounds of design research (Brown, 1992; Collins, Joseph, & Bielaczyc, 2004) with 72 subjects, and it has been used in courses at three American universities in the last 18 months.

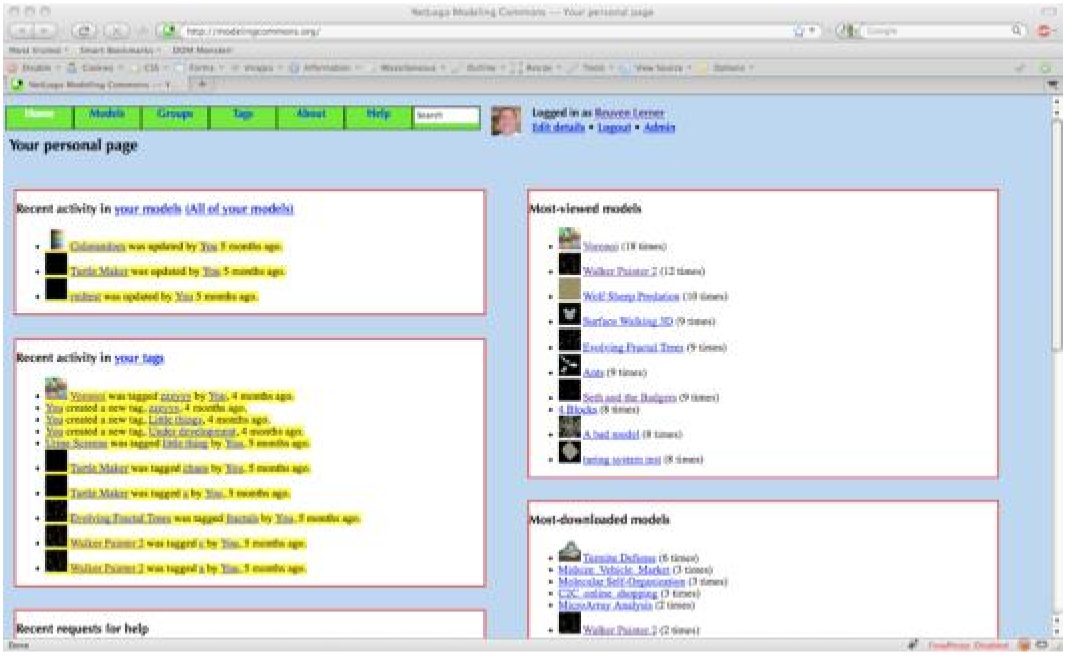


Figure 1: User's home page in the Modeling Commons

We believe that these trials have provided us with many insights into the fostering of public sharing and collaboration among NetLogo modelers. These principles may well be of interest to others creating similar collaborative constructionist environments (e.g., (Monroy-Hernández, 2007)). Our list was inspired by Bruckman (Bruckman, 1997), who enumerated principles that guided her design of the MOOSE language. By publicizing our list, we hope to engage in a discussion with researchers implementing similar environments, with an eye toward a set of "best practices" that encourage constructionist collaboration. There are already many descriptions of how to create an effective online community, and we have drawn upon many of their ideas, as well (Powazek, 2002; Spolsky, 2009).

1 Focus on artifacts

First and foremost, the social interactions are structured around artifacts – and more specifically, agent-based computer models. Models not only provides a focus for the user interface, but are also the core items to which users may attach documents (e.g., curricula, research papers, and NetLogo extensions written in Java), and serve as the foundation for both discussions and social tags (Smith, 2008). This model-centric focus reflects the constructionist philosophy of the NetLogo software. It also takes into account social learning theory (Lave & Wenger, 1991), and specifically Communities of Practice (Wenger, 1998), in which a shared repository of knowledge sits at the core of the community. By working on models in a variety of ways, new participants join the community.

Our experience to date indicates that this approach has been successful (Lerner et. al., under review). In the study, we have found that when looking at all communication taking place in a community, the models serve to create a fully connected social network.

The recent introduction of several oft-requested features — e.g., collections of related models ("projects"), and "families" of models derived from a common ancestor — has posed at least one challenge to this principle. For example, once there are 20 variations on the Fire model (Wilensky, 1998), should discussions be applied only to a single model? Or should they apply to all models with a common ancestry? We are currently considering a number of solutions to this problem.

2 Provide multiple entry points.

Constructionism has often used the example of "samba schools" (Papert, 1980; Zagal & Bruckman, 2005), in which participants take on a role within a community event. Part of the power of the samba-school example is its acknowledgement that there is no one, single path to learning and community participation. Rather, by providing a multiplicity of entry points, newcomers may choose a role that best suits them — or from a number of different roles, until they find something that seems appropriate.

We have tried to design the Modeling Commons such that it encourages users with different skills and interests to participate in ways that are most appropriate for them. They may browse or search through the list of models, reading about and experimenting with models that interest them. They may ask or answer a question, or simply post a comment, about a model. They may apply one or more social tags, such that other users will more easily find the model. They may contribute a document, providing additional documentation or curricular materials to accompany the model. Or they might read and modify the NetLogo code, enhancing the model itself.

One practical lesson that we learned during our design research was that many initial users wanted to be able to browse through the Modeling Commons without having to register or log in. Until that point, we had required registration, in part for easier tracking of user interests and activity. Removing this barrier to entry meant that browsing, or simply stumbling upon models in the Commons via a Google search, provided yet another flexible entry point.

Some initial findings (Lerner, Levy, & Wilensky, 2010a), showed that users did indeed take advantage of different facilities. The resulting social network from each of these individual communication links was a subset of the fully connected graph. Only after combining all modes of communication was the graph fully connected, demonstrating that different users preferred to use different aspects of the Modeling Commons.

3 Be forgiving.

A design principle of this same name appears in Bruckman (Bruckman, 1997), but we give it a different meaning: Whereas Bruckman designed the MOOSE language to forgive syntactic errors, we have designed the Modeling Commons to forgive editing and uploading errors. We did this by implementing a system similar to that used by Wikipedia (Wikimedia Foundation), storing all old versions and allowing users to download or revert to any of those versions.

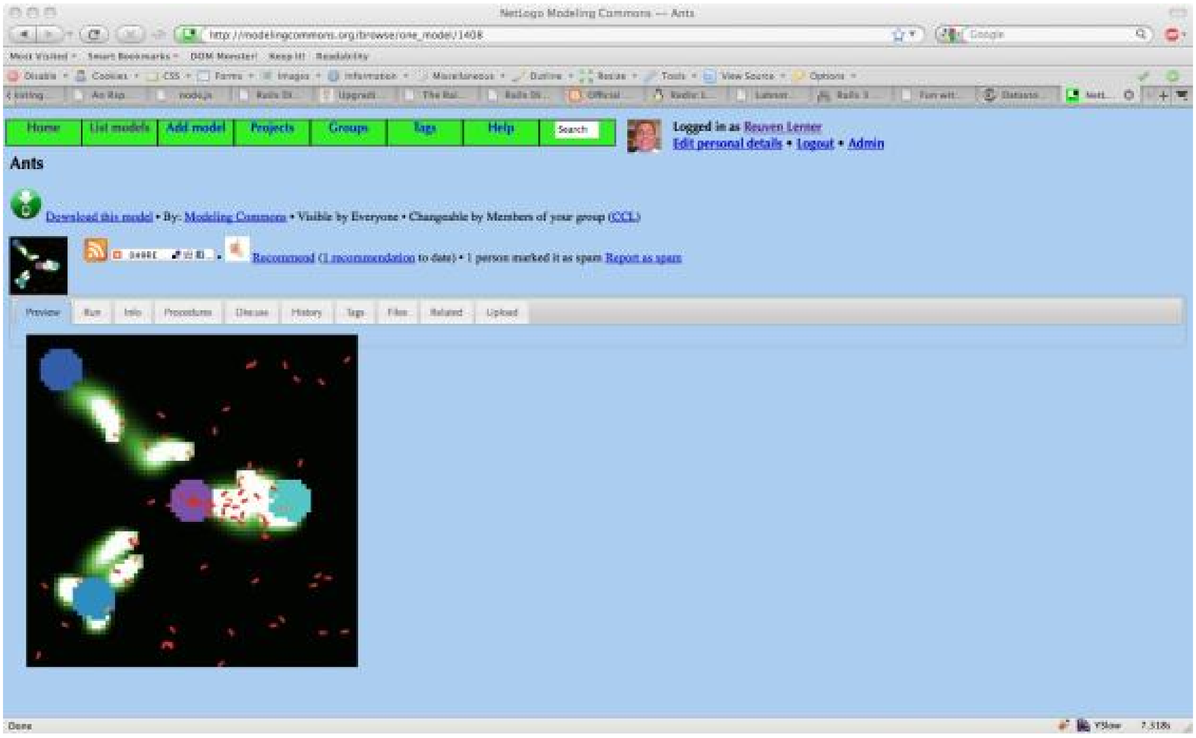


Figure 2: One model (Ants) in the Modeling Commons

Our design research found that even most experienced programmers, whose version-control systems provide similar facilities, failed to use such tools when modeling in NetLogo. By providing such capabilities, the Modeling Commons encourages users to experiment, without having to fear that previously saved changes will be lost. Experimentation is at the heart of constructionist learning, and we aim to offer modelers a safe environment for experimenting with their ideas.

4 Maximize findability.

Morville (2005) claims that information needs to not only exist, but be easily findable by someone interested in using it. Because many users of the Modeling Commons are new to NetLogo modeling, we aimed to structure the site for maximum findability, both from within the system and from without.

Within the system, we created a search system (see Figure 3) that looks at the model name, its authors' names, and the social tags that are associated with it. In addition, we made it possible to easily find models associated with a group of users, on a specific project, or to which a particular social tag had been applied.

Separately, we are employing techniques known collectively among Web professionals as “search engine optimization,” or SEO, to increase the chances of outside users discovering and then using the Modeling Commons. Already, before the official launch of the Modeling Commons, we have found that some users have found and downloaded models after searching for a specific model name in Google.

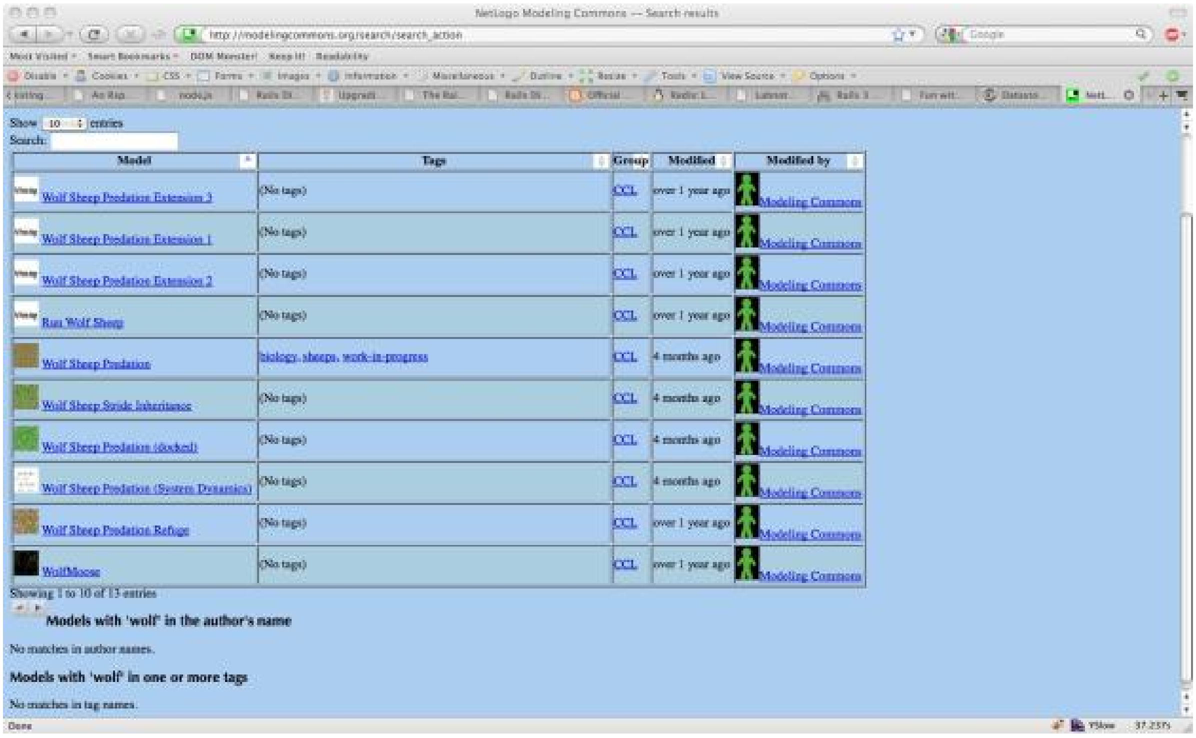


Figure 3: Search results in the Modeling Commons

5 Provide flexible permissions.

The original design of the Modeling Commons called for all models to be publicly readable and writable by all other users. Interviews revealed that many potential users were uncomfortable with such a policy, either because the models were still under development, or because the modelers were students, interested in sharing only with their classmates. Teachers were similarly nervous about using the Modeling Commons in their courses, if it meant that a students’ classwork assignment could be found and discussed, let alone potentially modified, by someone from outside of the class. At the same time, interviews showed that modelers were interested in eventually revealing their models and making them available to others in the Modeling Commons.

Protecting users’s privacy and security, while ensuring flexibility and allowing users to change those permissions down the road, thus became a key design principle. We distinguish between the "read" and "write" permissions that a user applies to a model (see Figure 4).

We should note that the question of permissions was complicated somewhat by the introduction of collaborators. If a user creates a model, and a second user modifies it, then should the original author be allowed to shut out the second one, by changing the permissions? We decided that any collaborator should be granted author status — meaning that once someone has contributed to a model, it is impossible to shut them out completely. Changing a model's permissions such that it is only visible to the author effectively allows all authors to still see it, as well as to modify it.

The issue of permissions is an important and sensitive one. We expect that it will require additional attention over time, and are curious to see how users will take advantage of these permissions, and how often they will indeed make previously closed models visible to the general public.

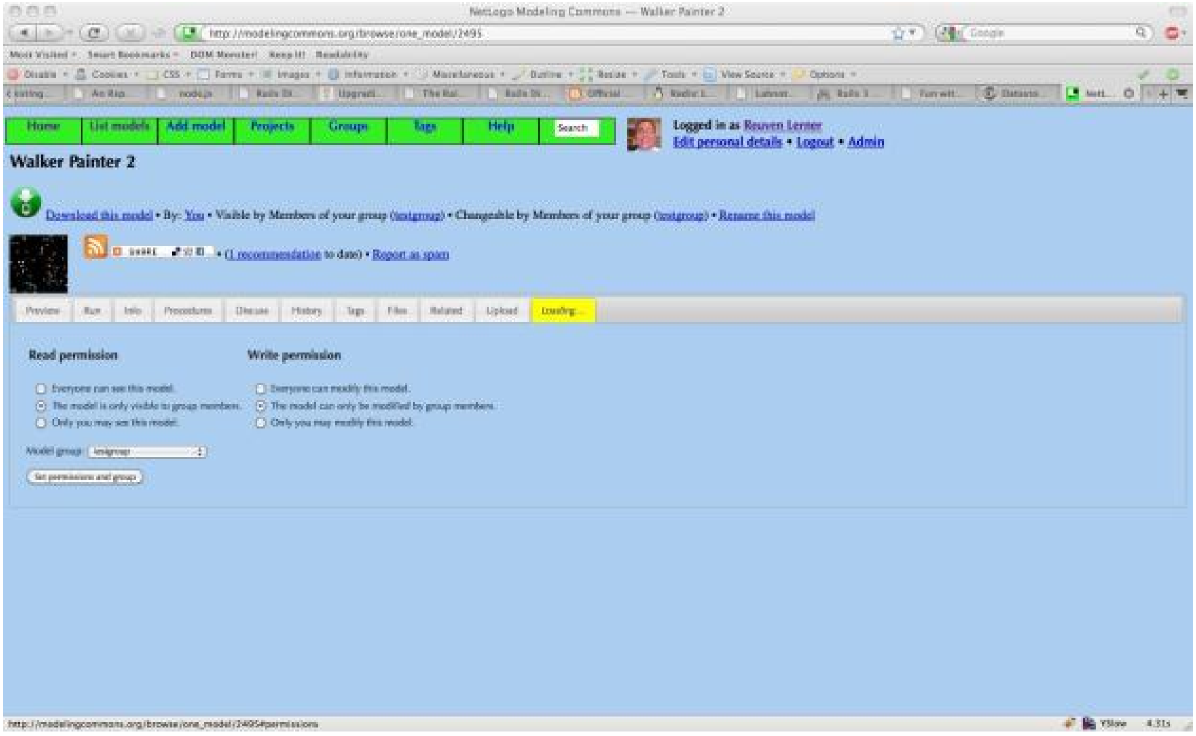


Figure 4: Permissions selection in the Modeling Commons

6 Keep users informed.

Collaboration requires participation, and participation requires an awareness of others in a community are doing. The Modeling Commons tries to keep users aware of what activities others are engaged in, both while they are using the site and when they are not. Many of these features were developed in response to users' feedback; each round of interviews provided additional ideas and suggestions for new features that should be added in order to make the system more social.

In the Modeling Commons itself, the user's default home page provides a bird's-eye view of the latest updates to the site, from the user's perspective. Every user sees nearly the same list of recently updated models, applied tags, and most-downloaded models. (The "nearly" reflects the fact that models to which the user lacks read permissions are removed from the list.) The user's home page also indicates what changes have taken place in the models to which the user has contributed, as well as those social tags that the user has created or applied.

The model pages also attempt to foster inclusion and participation. Users may indicate that they like a model, or recommend a model to their friends via a number of online networks and services, such as Facebook and Twitter. Users may participate in a discussion about a model; each comment in a discussion displays not only the contributor's name, but also their picture, much like blogs and recent social-networking sites.

Users may opt to receive updates about a particular model via "RSS," a popular protocol used by "feed readers," commonly employed by blogs. In this way, a user may "subscribe" to one or more models, learning about updates as they happen, but without cluttering their inbox.

If community members need to explicitly visit the site to find out what is happening, participation will be less frequent than if it is “pushed” to the users in some way. For this reason, an e-mail message is generated and sent to each of a model’s authors when a tag, discussion posting, or new model version is submitted. A user thus knows when other users users are improving or adding to a model. Each e-mail message contains a hyperlink back to the referenced model, discussion, or tag, making it easy to participate (and perhaps respond) immediately.

Principle	Applications of this principle
Focus on artifacts	•All discussions, and social tags are attached to a model •Secondary artifacts (e.g., documents) are also attached to a model
Provide multiple entry points	•Multiple ways to participate (e.g., adding and editing models, adding documents, taking part in discussions, and adding social tags) •No need to log in, if you only want to view models •Make it possible to run a model from within the browser
Be forgiving	•Easily revert to an earlier version of a model •Compare two versions of a model •Download earlier versions of a model (without reverting)
Maximize findability	•Internal search looks at multiple model attributes •Use of SEO to draw in people via Internet search engines •Links make it possible to find all models associated with a project, user, or social tag •Navigation and graphs showing relationships among parent and children models
Provide flexible permissions	•By default, models are visible to and modifiable by all users •Users may restrict read or write access to a model to themselves, to a group that they define. •Permissions may be changed at any time •Anyone who has contributed to a model is considered an author, and thus has the same permissions as the original author
Keep users informed	•Provide context and updates on the user's home page •Provide RSS feeds, for users to receive regular updates in a separate piece of software •Send e-mail alerts when a model on which the user has worked has been modified or updated •Send e-mail alerts when a user's discussion or social tag is updated

2 Conclusion

The Modeling Commons aims to provide NetLogo modelers with a method for sharing and collaborating that goes beyond the current norm among members of this community. These design principles have served us well throughout our design tests, and have helped to focus our development efforts. We look forward to providing further reports on our design principles, as well as the Modeling Commons itself, with the research community over the coming months and years.

References

Bollen, L., Hoppe, H., Milrad, M., & Pinkwart, N. (2002). Collaborative Modelling in Group Learning Environments. In P.I. Davidsen, E. Mollona, V.G. Diker, R.S. Langer & J.I. Rowe (Eds.), Proceedings of the XX International Conference of the System Dynamics Society (pp. 53-64), Palermo (Italy), July 2002.

Brown, A. L. (1992). Design experiments: Theoretical and methodological challenges in creating complex interventions in classroom settings. Journal of the Learning Sciences, 2(2), 141-178.

Bruckman, A. (1997). MOOSE Crossing: Construction, Community, and Learning in a Networked Virtual World for Kids. Massachusetts Institute of Technology, Cambridge, MA.

Collins, A., Joseph, D., & Bielaczyc, K. (2004). Design Research: Theoretical and Methodological Issues. Journal of the Learning Sciences, 13(1), 15-42.

de Aennle, C. (2009). In Simulation Work, the Demand Is Real. The New York Times.

Jonassen, D. (2006). Modeling with Technology: Mindtools for Conceptual Change (3rd ed.). Upper Saddle River, NJ: Pearson Education, Inc.

Lave, J., & Wenger, E. (1991). Situated Learning: Legitimate Peripheral Participation. New York: Cambridge University Press.

Lerner, R., Levy, S.T., & Wilensky, U. (under review). Connected Modeling: Design and Analysis of the Modeling Commons. International Journal of E-Learning and Learning Objects (submitted to IJELLO special series of Chais Conference 2010 best papers).

Lerner, R. M., Levy, S. T., & Wilensky, U. (2010b). Modeling Commons. http://modelingcommons.org/.

Monroy-Hernández, A. (2007). ScratchR: A platform for sharing user-generated programmable media. Proceedings from IDC.

Morville, P. (2005). Ambient Findability. Sebastopol, CA: O'Reilly Media.

Papert, S. (1980). Mindstorms. New York: Basic Books.

Powazek, D. M. (2002). Design for Community: The Art of Connecting Real People in Virtual Places. Indianapolis: New Riders.

Reisman, K., & Wilensky, U. (2006). T*hinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories -- an embodied modeling approach. Cognition and Instruction.*

Smith, G. (2008). *Tagging: People Powered Metadata for the Social Web*. Berkeley, CA: New Riders.

Spolsky, J. (2009). *Learning from StackOverflow.com. Google talks* http://www.readwriteweb.com/archives/anthropology_the_art_of_building_a_successful_soci.php.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes.* Cambridge, MA: Harvard University Press.

Wenger, E. (1998). *Communities of Practice: Learning, meaning, and identity*. Cambridge: Cambridge University Press.

Foundation, W. Wikipedia. Retrieved 11 November, 2007, from http://www.wikipedia.org/.

Wilensky, U. (1998). *NetLogo Fire model*.

Wilensky, U. (1999a). *NetLogo* [computer software]. Center for Connected Learning and Computer-Based Modeling, Northwestern University.

Zagal, J. P., & Bruckman, A. (2005). *From samba schools to computer clubhouses: Cultural institutions as learning environments*. Convergence, 11(1), 88-105.

A Case Study: Theatre as a constructionist tool for helping 6th graders build their own word meanings

Dinan Messiqua, *dmessiqua@carrollschool.org*
6th Grade Language Arts Teacher, Carroll School, Lincoln, MA, USA

Abstract

Constructionism puts forth the idea that individuals make sense of their world by building their own descriptive and meaning-making models of it. Most of the work done under the label "constructionist" is, of course, based on the use of tools that are computer-based or use some kind of technology. Probably, most of the papers at this conference will be of that nature. But, I believe that constructionist methods need not necessarily be technologically based.

Constructionists, I will argue, can use model-construction-tools from a variety of sources: written and spoken words, poetry, images, body movements, costumes, character roles, theatre sets, etc. In fact, I am going to present a constructionist example using theatre as the environment to encourage reading-disabled children to explore the meanings of new vocabulary words. I will describe how students construct their theatre world, the roles, the scripts and the action.

My paper describes my work at the Carroll School in Lincoln, Massachusetts. All students at the Carroll School are severely learning-disabled. All are dyslexic. I want to describe to you how constructionist techniques can be useful to students with special learning needs. I will suggest at the end of the paper that my activities with my students can offer lessons to those who work with students who are less handicapped.

Seven 6th grade students in a Language Arts classroom were participants. They were introduced to ten vocabulary words using a regular vocabulary textbook and then they were divided into three groups. Each group was assigned three vocabulary words and had to come up with a theatrical skit incorporating the words. Each group presented its skit to the class. The outcomes of this activity were that students: 1) Were able to construct their own meaning the words – one that made sense to them, 2) Were engaged in the activity since they had to create something, 3) Learned from each other since they were working in groups and, 4) Developed interpersonal skills since they had to work with one another. This case study shows that using constructionist approaches to work with students with learning disabilities is a valuable approach because it engages them in the learning process.

The study suggests that active student involvement in the learning process is key and seems to be more effective than traditional instructional methods. My hypothesis is that the success of this activity can be carried into the traditional classroom and that a constructionist approach can work well with students across the spectrum, not just students with learning disabilities.

Keywords (style: Keywords)

theatre, special education, drama in special education, vocabulary words and theatre

Introduction

Constructionist Approach

I work in a school for children who have severe reading disabilities and often attentional challenges. Due to their learning challenges, school has often been a torturous path for them. Sitting and listening to teacher speak for 45 minutes in a regular classroom can be painful for them. These students need hands-on activities, they need to participate in the learning and be active learners. For this reason, the constructionist approach is vitally important for these students. Constructionism stresses the importance of the artifacts, or tools, that we build, use and manipulate in our personal construction of knowledge. Theatre is a medium that that brings all the senses into play, (sight, touch, sound, smell and taste). For this reason, it is a wonderful media through which to build knowledge. In this paper I will give a specific constructionist example of how I encourage learning-challenged students to understand and use words through theatre.

Vocabulary Instruction

We live in a world where literacy skills are crucial to be able to function as a human being. Robust vocabulary instruction is vitally important to foster language and literacy skills in children. Reading disabled students struggle with all aspects of language, (reading, spelling, writing), except for speaking. Beck (2002) states that a common notion is that students naturally acquire vocabulary through context. She states that at an early stage words are learned orally when a child is young, but later as the child gets older the vocabulary learning is through written context – what a child reads. It is much harder to learn words through a written context since text lacks many of the supports of oral language such as intonation. Therefore, all children, (especially those with reading disabilities), need direct instruction in school to foster their vocabulary. Beck also debunks the myth that that students need to learn so many words to be successful in school that it is simply impossible to teach them all, therefore learning through context is the only way to go. Beck counters this argument by explaining that not all words need to be taught. She explains that only certain words need instructional attention. She divides words up into three categories: Tier I, Tier II and Tier III words. Tier I is made up of the most basic words such as *clock, baby, happy, sad, walk*, and so on. Children will most often learn these words in their home life, outside school and just through listening. Thus, these words do not require direct instruction. Tier III is made up of domain-specific words, meaning words whose frequency is quite low and are related to a specific subject. For example, *isotope, peninsula,* and *refinery* are all Tier III words. Beck explains that in general a solid understanding of these words is not of high utility for most students. These words are best learned when a specific need arises for them, such as in a science or social studies lesson. Tier II contains words that are of high utility for students and are found in many domains. For example, the words *coincidence, absurd, industrious* and *fortunate* are all words that can be used in many different contexts. Once a student knows the definition of these words, he/she can use them in many arenas and thus his/her verbal repertoire is greatly enriched. For this reason, it makes sense to focus on Tier II words when teaching vocabulary.

Since vocabulary instruction is key to fostering literacy skills in students with reading disabilities and these students learn through 'doing', this paper outlines a constructionist approach to teach vocabulary to students with reading disabilities using theatre.

Method

Student Profile

I taught this vocabulary activity as part of a Language Arts class to a group of 6th graders at the Carroll School. The Carroll School, located in Lincoln, Massachusetts, just outside of Boston, is dedicated to meeting the educational needs of children diagnosed with language-based learning disabilities, such as dyslexia, and to supporting the constituencies that serve them. At Carroll, all teachers adhere to the following principles, in all subject areas, in all classrooms, and in all instruction: cognitive approach, direct and explicit, structured and sequential, cumulative, multi-sensory, and alphabetic-phonetic. Grades kindergarten to 8th grade are taught and class sizes range from 6-8 students.

There were seven students in the class I taught. The range of scores for these students in each of the following categories was as follows:

Word Recognition: 3-5th grade. This ranged from students who needed to sound out each word to students who were good decoders.

Oral Reading: 3-5th grade. This ranged from students who needed to sound out each word to those who were strong decoders but had poor fluency.

Word Meaning: 4-7th grade. The students had uneven vocabulary knowledge. Most of them lacked Tier II words.

All the students had a much higher oral language comprehension than reading comprehension. For this reason, the unit was designed to teach to their oral vocabulary, (teach vocabulary at their cognitive level), to improve their comprehension.

Vocabulary Instruction

Students had a vocabulary book called Groundwork for A Better Vocabulary (3rd Edition). Each chapter in the student's vocabulary book contained ten words. For each chapter the following steps were taken:

1. Activator: Students' prior knowledge of the words is activated.
2. Meanings of words are directly taught
3. Fill in the Blank Activities
4. Yes/No Why questions
5. Theatre Skits (constructionist piece)
6. Application of words in their writing
7. Quiz

The following was done for Chapter 1 in the book. (see chapter 1 list attached at end).

1. Activator:

Words were shown on board. Students were asked to raise their hand if they could read a word. Once all the words were read, students were then asked if they could give one (or more) of the following for each word:

- a definition
- an example of the word
- give the word in a sentence
- part of the speech of word

This allowed me to gage what the students already knew.

2. Meanings of Words Directly Taught:

Any words that the students did not know were directly taught, followed by a discussion of the contexts in which the word and meaning could be applied.

3. Fill in the Blank Activities:

Since I begin with a set of isolated words, I know that I need to provide students with opportunities to use them in a variety of contexts and to receive feedback about their success in doing so. One of the ways in which I do this is via **fill-in-the-blank sentences.** Students are instructed to use each word only once in this activity, which can present them with much difficulty. This is especially true when more than one word could fit into a sentence. Through teacher modeling and small-group discussion, however, students quickly figure out the best ways to fill in the blanks. This format also provides students and I an opportunity to discuss inflectional endings (e.g., *-ed, -er,-s*).

In addition to the sentence task, each chapter includes two fill-in- the-blank paragraphs, which I have found most students with reading difficulties see as even more of a challenge than the sentence task. Their difficulty in recognizing clues about meaning in a text accounts for why their efforts to use context to figure out the meanings of unknown words helps so little. In working on the paragraph fill-ins, I show students how to complete the activity in stages, and model for them the kinds of decisions that skilled readers make as they process text. For example, to introduce the task, I first read the whole paragraph, showing students how to get an overall sense of the topic. Then, using a think-aloud procedure, I work through the blanks, drawing attention to the context clues that help narrow the possible choices. I also show how to skip blanks that are difficult to fill the first time through, cross out words as they are used, and pencil in possibilities when they aren't sure which choice is best.

4. Yes/No Why Questions:

This activity is modeled on one designed by Beck, Perfetti, and McKeown (1982). Yes/No/Why questions are constructed by pairing the words in the chapter, and students are asked to answer each question as well as to provide a reason for their answer. There are no right or wrong answers, and I provide a model for students of how to give support for their answers. By encouraging students to make their thinking explicit, additional relationships among the words and concepts can be discovered and discussed.

5. Theatre Activity or "Skits"

The seven students are divided up into three groups, (one group has three students). Each group picks 3 vocabulary words from the chapter, (words are chosen from a bag, they cannot see them), and one group, (the larger one), picks 4 cards. The students are given about 10 minutes to come with a skit that incorporates the vocabulary words. For the groups that have a hard time getting started, I tell them to pick location, (ex. a supermarket), and then choose characters that will be in their skit, (ex. salesperson, a customer, and the store manager). Students come up with a plot quite easily on their own. If I do this activity on a Monday, I ask students to pick an event from their weekend and make this the setting of their skit. With three in a group, each student might bring in their own weekend event into the skit and the skit can get quite wild! The 'weekend plot' also gives students a chance to learn about each others' weekend.

They have specific guidelines:

- In their skit they must show that they understand the meaning of the word (for example, they cannot have a skit's location be spelling class and just ask each other to spell the vocabulary words).
- They cannot use inappropriate language (swear words) in their skit.
- The location of the skit must be appropriate.

While the students are preparing their skit, I walk around to check that the contexts of skits are decent.

Presentation of Skits

To set the mood for the skits, the lights in the classroom are turned off and stage is created, (desks pushed back). The actors in the first group to present take their places. I tell the students who are not presenting that they need to listen for the vocabulary words in the skit and be able to tell the actors at the end what they were and in what context they were used. This way the students watching are active audience members versus passive. The students not presenting also introduce the skit by saying: 'Lights, Camera, Action" (there is a gesture that goes with each of these words). A student who is not presenting first is the lights person. He/she switches on the lights in the classroom after he/she hears "Action!"

The skit is presented, (it usually lasts about 2-3 minutes).

After skit has been presented:

Actors stay up on stage and the following happens:

1) Audience members raise their hands to say what words were used in the skit and how they were used. Actors pick on these people. They let the person know whether or not he/she was right.
2) Audience members ask questions to actors if they were confused by something in the skit.
3) I give immediate feedback to the actors regarding whether or not they used the words in the correct context.

6. Application of words in Writing:

Writing is one of the primary ways in which students are encouraged to process word meanings in an active and generative way (Curtis & Longo, 2001). Every other week I assign at one topic for students to write about, using at least 5 of the vocabulary words for the unit, along with any words from previous weeks they can incorporate. This encourages students to think about how to correctly use the words.

Conclusion

This theatre experience I had with my students taught me six important facts:

1) By making up the skits the students had to build their own method of understanding exploring the meanings of the words they were given. By doing this they gained a sense of ownership of the material. A month later, they would recall the meaning of the word by going back to their skit, which they remembered since they had created it.
2) Those students who lacked confidence in their vocabulary began to feel that they could make sense of difficult words. They gained confidence in their learning and they began using words these words in their oral language and in their writing.
3) The students learned from each other for two reasons: 1) as they created the skit together they had to talk about the words and refine their own meaning to make sure they really understood the words 2) students in the audience who watched each skit got to see how words were used and were able to ask the actors questions after the skit, for example if a word was used in confusing way.
4) I was able to see immediately if my students understood the meanings of the words and give them immediate feedback.
5) Students developed interpersonal skills because they had to work with each other.
6) For my students who learn in a non- traditional way, this is a wonderful way for them to access knowledge and to express their acting talents! Thus, to feel successful!

Summary

This paper describes a specific constructionist example of how I encourage learning-challenged students to understand and use words through theatre. I describe a classroom activity in which seven 6th grade reading disabled –students were taught a 10 vocabulary words using theatre. Once the meanings of the words had been discussed and understood in the classroom, the students had to apply the words in theatre skits that they made up. They were divided into 3 groups and each group was given three words. They were given ten minutes to prepare their skit. Once all the groups were ready, each presented. After each skit the students in the audience had to identify the vocabulary words that were used by the actors, and could ask clarifying questions to actors.

This study shows that using constructionist approaches, with theatre as the medium, engages students in ways that instructionist techniques do not. I've used both, by the way!

Finally, I would like to say that helping dyslexic students stay on topic every minute of the day helped me to interact with them in ways that might not happen in standard school environments. I had to stay on topic and watch and listen and talk with them about what they were doing and why. Teaching methods are never taken for granted at the Carroll School. Everyone must be on their toes and aware of what is going on, what is being said. The entire classroom becomes a theatre of learning and talking about learning. This surely relates to the public place that Papert described, where much of the meaning making of the constructionist project takes place. I hope that my experience, therefore, might be seen to be useful to those using other constructionist media and in other more teaching institutions.

References

Beck, I.L., McKeown, M., R., Kucan, L. (2002). *Bringing words to life: Robust*

vocabulary instruction. New York: The Guilford Press.

Beck, I.L., Perfetti, C.A., & McKeown, M.G. (1982). *Effects of long-term vocabulary instruction on lexical access and reading comprehension*. Journal of Educational Psychology, 74, 506-521.

Curtis, M.E., & Longo, A.M. (2001). *Teaching vocabulary to adolescents to improve comprehension*. Reading Online, 5(4). Available:http://www.readingonline.org/articles/art_index.asp?HREF=curtis/index.html

Johnson, B., Mohr, C. & Goldstein, J.M. (2004). *Groundwork for a Better Vocabulary (3rd Ed.)* New Jersey: Townsend Press, Inc.

Addendum

Chapter 1 Words from Groundwork for A Better Vocabulary:

challenge

dependent

fertile

perculiar

preference

principal

solitary

suitable

surplus

transform

Learning of Dynamic Data Structures – Having Fun with Algorithms

Ingrid Nagyova, *ingrid.nagyova@osu.cz*
Dept. of Information and Communication Technology, University of Ostrava

Abstract

One of the basic abilities students specialized in subjects focused on information technologies should cope with, is algorithmic thinking. However, we must admit that algorithms' training seems to be rather difficult for students therefore they may become unconcerned.

This article is an attempt to illustrate that algorithms training may be interesting and inspiring for students. They actively participate in the classwork; they seek inspirational tasks and their possible algorithmic solutions. When teaching dynamic data structures, we apply constructivist teaching methods. At first students primarily learn to create algorithms without using any programming languages.

The scope of dynamic data structures is relatively demanding. It requires a good understanding of computer memory organization. On the other hand, no extensive knowledge in programming languages is required, not even with most demanding projects.

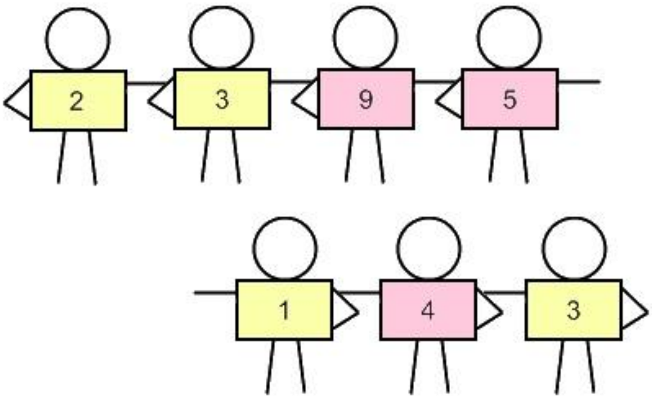


Figure 1. Students Form Dynamic Structures

At the beginning, each student puts one hand on his/her hip (this hand creates a linking loop (linking structure) and the other hand - the empty one - is "pointing" at nil. Getting linked by loops, students form trains. When the rules for work with "student trains" are defined, a lot of potentialities stimulating students to solve various tasks are created, for example a uncoloured train, an organized train, etc.

Keywords

algorithms; pointer; dynamic data structures, stack, queue

Learning Algorithms and Programming

The Pedagogical Faculty of University of Ostrava (Czech Republic) educates future teachers and professionals for all types of educational institutions. The field of study Information Technologies in Education is focused on the preparation methodologists and ICT coordinators on primary and secondary schools. During the course of the bachelor's degree study the students of that field complete four examinations focusing on algorithms and programming:

- basic algorithms and programming
- complex algorithmic structures – including the teaching of dynamic data structures
- Imagine programming
- object-oriented programming.

Teaching algorithms is one of the main curriculum topics of study fields focused on informatics, as well as the field Information Technologies in Education. Information technologies are based on the idea of algorithms. Just a few years ago, every common computer user had to make a practice of programming. Even today, this ability is current for experts working with information technologies.

Strategies and methods of programming and algorithms training are dealt with in quite a few scientific subject publications. In spite of all efforts that are made to apply these recommended methods, algorithms and programming teaching experiences difficulties. Demanding factor and the most abstract level of the subject-matter reflects into students´ lack of interest, and they tend to minimize their efforts to minimum level necessary for scraping through the exam, etc.

This contribution is based on experience gained in our courses on dynamic data structures, where we try to apply the constructivist teaching methods. First of all, students learn to suggest and to create algorithms without using any programming language, just through movement and activities in the classroom. Subsequently, they learn basic orders which are going to be employed in practical problems solution.

The scope of dynamic data structures (Wirth, 1986) is relatively demanding. It requires a good understanding of computer memory organization. However, as far as students cope with the subject matter, they will understand basic principles of computer architecture. Moreover, if the high demand factor and abstraction of the subject-matter those students perceive at the very beginning as an undefeatable barrier is compassed, it incites their enthusiasm. Students become involved in the study and develop their creativity in further research. And it is also included in the title of this paper – Having a Fun with Algorithms (Futschek, 2007).

Learning Dynamic Data Structures

Dynamic data structures' training consists of several coherent items:

- pointers
- linear dynamic structures and linked lists
- non-linear dynamic data structures.

We are going to concentrate on the first two items.

Pointers

The term "pointer" (as the index into the computer memory) corresponds to the Czech expression "signpost". Our university is situated at the foothills of Moravian-Silesian Beskids Mountains, and there are many marked tourist pathways crisscrossing one another in the surrounding area. The project is based upon these pathways, leading a passionate tourist from place to place (see figure 2). Project "Wandering through Moravian-Silesian Beskydy Mountains" has been developed from an original idea of students and thanks to their initiative. They created it in Imagine.

Over the project, students ponder on the term "pointer", and together we try to solve a range of questions: What stands behind the term "pointer?" What information will this signpost carry through? What shall we gain if we follow the message on the signpost?

A pointer is denoted by an arrow. The pointer carries information on the reference (address) of a certain place in computer memory, where a certain value can be stored. Further, teacher can point with his/her right hand (pointer) to individual students (values), and thus call them to come to the blackboard. On this call, each student can point with his/her right hand to another student, or s/he can even get linked to him/her – catching his/her left arm (see figure 1). Left arm represents the linking loop in this game, it is forbidden to use it for any other work.

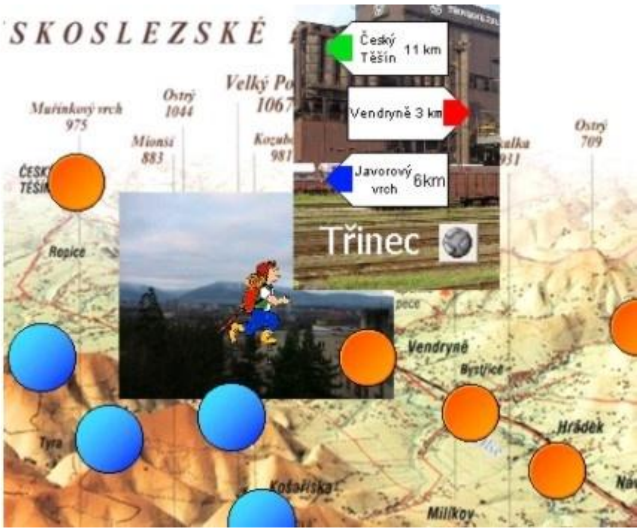


Figure 2. Project "Wandering through Moravian-Silesian Beskydy Mountains"

There are some other rules of the game:

- students, who are not held by anybody, must be given a name,
- each one is allowed to be holding at most one of the follow students,
- it is allowed to unbend from the follow student´s loop only in acute cases,
- if the person who is holding my arm moves, I must move with him/her.

Getting linked by loops, students form one or more trains. Then, "student trains" attempt to solve various practical tasks. They can, for example, couple on a "wagon" – a student from another train - either to the head or to the rear of their train. They can insert a student into the middle of their train. They can even cancel the whole train. More difficult tasks come after, for example reversing the train, whereas the first student becomes the last one - i.e. the one who is not held by anyone – or other tasks can be solved, e.g. reordering the train according to the colours of students´ T-shirts, or reordering it according to their height, etc.

For homework, we often use books or CDs instead of trains (see figure 3). These are to be stacked on one another. The upper CD (like the first student at the head of a train) must always be given a unique name, the name being labelled on it.

Through solving the tasks with CDs, students gradually familiarize with the problem and learn to work in the simulated environment of the defined game. Then, there is just one step to go to the definition of work with linear linked lists.

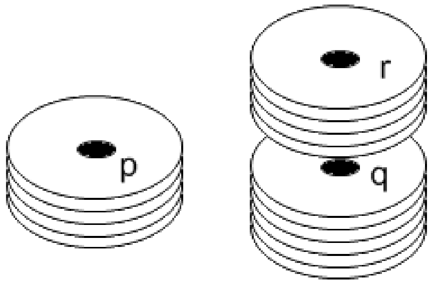


Figure 3. Tasks with CDs - Shifting and Labelling CDs

Linear Linked Lists

The ability to solve the tasks assigned in the environment of "student trains" game can be developed in a software simulator (see figure 4). It has been created again by students in Imagine. Individual "wagons" are represented by rectangles with a load - an integer value. Pointers are denoted by dots with arrows.

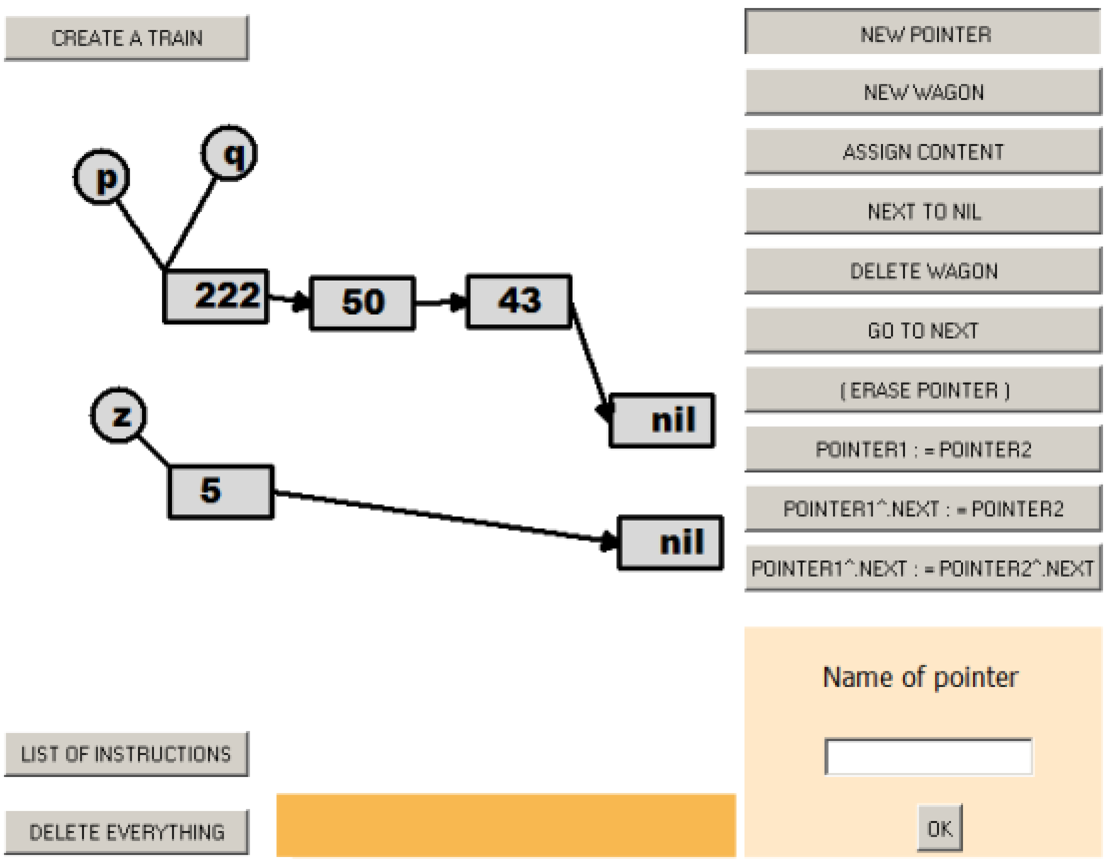


Figure 4. Software Simulator of Linear Linked Lists

Simulator provides:

- creation of new pointers,
- creation of new wagons which are assigned a value,
- deleting wagons, erasing pointers,
- shifting the pointer to the next wagon in the sequence,

- resetting (setting to nil) the value of the wagon pointer,
- mutual assignment of two pointers.

Tackling problems in a simulator is more complicated, since pointers are used as the names of individual wagons. Students are forced, in contrast to the preceding activities, to work with the names of wagons dynamically, to consider their creation and cancelling, to shift the wagons from one to another, etc.

The work of the student in the environment of a simulator is registered and written down step by step in the form of concrete orders of programming language into the computer memory. Whenever the student likes, s/he can look through the record of actions, which were made in Pascal programming language, and if necessary, it can be copied into the development environment of this language.

These way students learn to know linear linked lists and how to work with them. They are getting to know, that it is necessary to set the pointer of the last wagon of the train to nil. They learn that the wagon, which is not pointed at by any pointer, is irretrievably lost. They discover simple and more complicated algorithms for work with linear lists, which are coming up. As soon as these strategies are mastered, we can focus on particular dynamic structures - stack and queue.

Stack and Queue

"Historical architecture reminds us of lives of people and of long past events. Some of the oldest monuments of history are the buildings in Egypt, extant up to the present day. One of those is a well-known temple in Abu Simbel, which had to give way to the construction of Aswan Dam half a century ago. It was resolved that the precious complex of the temple would be moved over a few metres higher." These words are an introduction to the stack and queue theme. The huge complex of the temple had to be cut up into blocks, moved to a new place and put together again. Cars, waiting to be loaded, were standing in a line (queue). First of all, the blocks from the upper part of the temple, which had to be cut up first, were transported. However, it must have been the bottom blocks which had to be placed on the new place primarily. That is why the upper blocks were temporarily stored at a transfer area. The transfer area comes to be a stack.

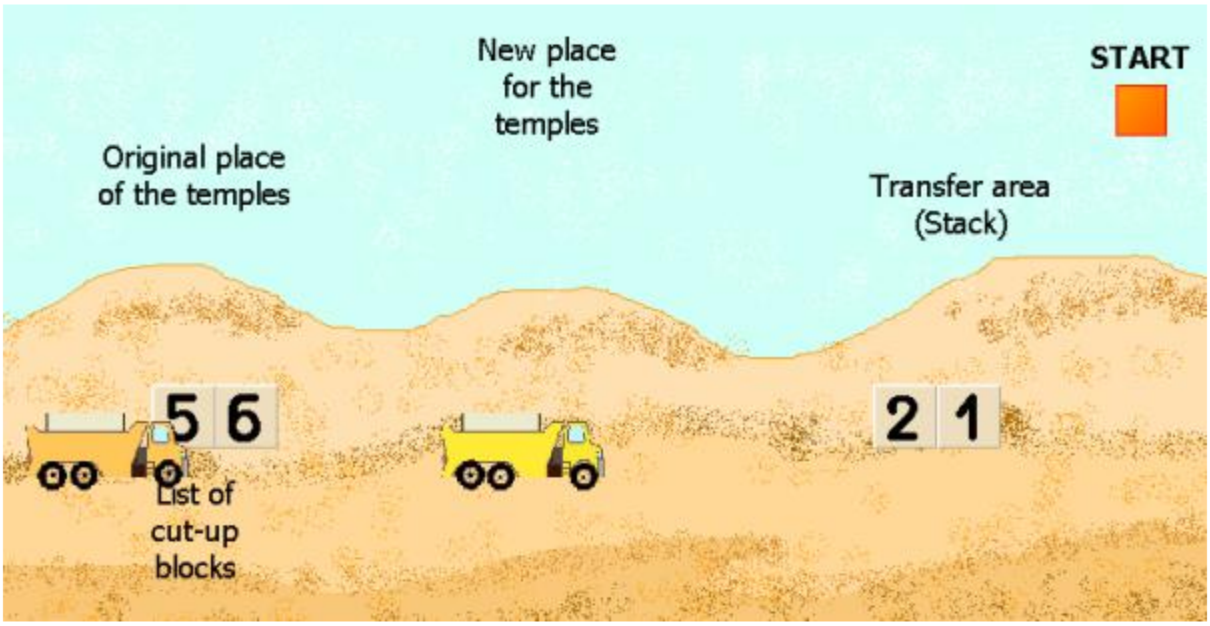


Figure 5. Transfer of the Temple in Abu Simbel

This story demonstrates a model created in Imagine (see figure 5).

Let us come back to the thought of "student trains". Students are lined up in one train and their task is to form a new train, formed by the same "wagons" (elements) in the same sequence as the existing train. We can choose only the first student from the existing train: This student becomes the first wagon in the new train. To solve the problem, we must use a transfer area again, where the train, made of the same wagons, is reversed. There are two steps for handling the task (see figure 6).

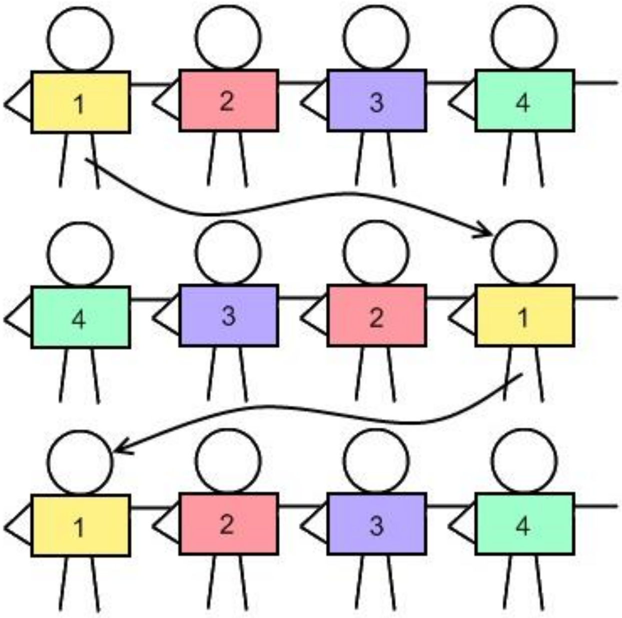


Figure 6. Transfer of Students into a New Train

Students learn the main principles of the stack and the queue operations through solutions of these model situations. Subsequently, they become able to distinguish and to define explicitly these structures. At the same time they come to know the corresponding programming codes by means of the described software simulator.

Conclusion

The scope of dynamic data structures is undoubtedly one of the most demanding. Dealing with algorithms seems to be most difficult to students. Curriculum tends to become tedious and arid for them, in particular on the grounds of the high extent of algorithmic abstraction.

Yet, experience and practice exemplify, that if we find a suitable presentation method of the given subject matter, students can master even complicated abstract themes. An advisable technique seems to be that kind of a teaching process, where students become active participants in class work activities simulating a microcosm with its rules, and students gradually gain required knowledge. The effect of such teaching method is even increased at the moment, when students, having an opportunity to participate in the activities, become a part of the game; they become one of the "wagons" of the "student train."

The field Information Technologies in Education only about 40 students pass out in full-time and combined forms yearly. The results of teaching therefore can be so difficult to evaluate quantitatively. The students' increased interest, discussions, creativity, as well as the focus of the themes of bachelor theses unambiguously confirm the positive evaluation of the described teaching methods, The process of teaching dynamic data structures is based on a yearlong cooperation with students, who, incited through our methods and techniques, brought new ideas, came with the view of enriching the learning process. Many of the above ideas and mentioned procedures are the result of students´ work. I would like to give acknowledgment and thanks to all of them. The educational aids were created in the universal educational environment Imagine (Kalas and Hrusecka, 2004).

The described teaching methods help students understand the structure and running of algorithms in action with dynamic variables. The students learn to design further algorithms and described their running. However, they are unable to use these skills for working with dynamic data structures in a particular programming language, for programme creation. These skills aren't included in the students' curriculum; we plane to find appropriate teaching methods in the future.

References

Futschek, G. (2007*) Logo-like Learning of Basic Concepts of Algorithms - Having Fun with Algorithms*. In Proceedings of EuroLogo 2007. Edited by I. Kalas. Bratislava, August. pp. 51.

Kalas, I. and Hrusecká, A. (2004) *The Great Big Imagine Logo Project Book: projects, worksheets and starting points.* Logotron, Cambridge

Wirth, N. (1986) *Algorithms & Data Structures.* Prentice-Hall, New Jersey

Reconstructing *Constructionism*

Richard Noss[1] **r.noss@ioe.ac.uk**
London Knowledge Lab, Institute of Education, University of London

It is more than a quarter of a century since the idea of constructionism was launched by Seymour Papert - the n-word rather than the v-word, constructivism. While the latter idea captured nicely the psychological substrate on which all learning (irrespective of teaching) is built, the n-word sought to develop a theory of pedagogy that could foster learning. More than that, while on the surface, the constructionist project seems like a pedagogical theory, it is as much a theory of epistemology as one of pedagogy, seeking to develop knowledge structures in the mind alongside physical or virtual structures external to the mind. Understanding the development of the structure of knowledge is part of and integral to the encouragement of an inclusive and powerful pedagogic theory and practice.

Constructionism symbolised a way of thinking about learning, a metaphor for the ways that human beings come to learn most effectively; building, debugging, sharing in ways that could at last be reasonably straightforward, or at least achieveable thanks to digital technologies - particular what Seymour called its "Protean" nature - its potential for being all things to all people. The Logo movement was, of course, emblematic of this approach and continues to leave a powerful mark on the educational world, a mark that is evident in the proceedings of this conference.

How odd, then, that so many believe that Logo, and more importantly, what Logo stands for, has failed in its essential mission to transform learning and teaching. Even learning settings that share Seymour's vision implicitly, seldom recognise their practice as an instance of the constructionist class. More often than not, it is confused with its psychological cousin, constructivism - a word that has all but lost its real meaning in the rush to embrace an alternative to behaviourism and its offshoots, a meaning diluted to the point that almost any pedagogy is routinely described as 'constructivist', as if a recognition of how humans learn is sufficient for prescribing how and what they should and could be taught.

I think there are many reasons for this, and they include political, sociological and philosophical as well as epistemological dimensions. I believe our conference here will touch on all these dimensions. The challenges are many: they include what it now means to collaborate, the idea of modelling, how to tap into young persons' cultures, new problems of design - the challenges are considerable. Here I want to focus on one difficulty that I think is particularly pertinent.

There is a fundamental difficulty of constructionism as an epistemological idea. As Papert says, the difference between instructionism and constructionism

> "looks like a split about strategies for education: two ways of thinking about the transmission of knowledge. But behind this there is a split that goes beyond the acquisition of knowledge to touch on the nature of knowledge and the nature of knowing".

In a world where the mismatch between telling and learning is all-too-evident - perhaps to all but the most regressive of educational policy makers - constructivism is an unproblematic alternative. Precisely because it is a theory of how the mind constructs knowledge, it does not threaten what needs to be known; the grain size of the constructivist description of human

[1] My thanks to Celia Hoyles for her contributions to this paper, and to James Clayson for his helpful comments on an earlier draft.

learning makes it homogenously applicable, it does not challenge who needs to know what; and since constructivism focuses its attention on the learning capability of the individual mind2, it does not require a rethinking of how to foster learning in the light of new tools that make the inexpressible possible to express. Constructivism is interpreted as constructionism minus epistemology.

At root, therefore, is the extreme reluctance of the educational enterprise to change what to teach and its preference for straightforward evolution of pedagogy (at most). Compared with reevaluating what can be taught and to whom, the switch from instructionism to some other ism that recognises the complexity and heterogeneity of learners, is unproblematic. Doubtless we will see many examples at this conference of genuine epistemological transformation, expressed as instances of what the constructionist project intends. It is time to gather these examples together, to develop a theoretical framework that can encompass the tremendous range of existence theorems that are instantiated in this conference, and turn them into a programme - of research and practice - that can do justice to Seymour's idea.

One possible starting point is a seminal paper by Andy diSessa and Paul Cobb. In it, they argue for the importance of theory in educational design experiments, and they survey different roles played by theory in design. In differentiating between four types of theory -- from 'grand' theories such as Piaget's constructivism (which they properly point out was not intended and largely fails to inform design) to "Domain specific instructional theories", which involve testable conjectures about learning processes and how to devise pedagogic situations that encourage them.

Constructionism, like 'learning by designing', Cobb and diSessa argue, falls into a category they call "Frameworks for Action", and they argue that while these frameworks do provide some heuristic power and structure to the design of learning environments, they typically:

> ... do not cleanly separate their scientific claims and validation from their suggested actions. That is, the theory or theories behind frameworks for action are relatively inexplicit, complex, and often involve multiple very diverse elements that cannot plausibly be brought under a single umbrella.

DiSessa and Cobb argue for the need to 'manage the gap', the failure of most frameworks to accomodate the complexity and interactions between the elements of instruction itself. It is a truism to note - as have so many others - that instructional effectiveness depends on many variables, not least the nature of technology - a field which is chaotic in the literal sense: tiny changes in, for example, the user interface can make massive changes in learning. The primary point is that in order to test theory, it is necessary - as far as possible - to maintain a gap between the pedagogical strategies at stake and the theories that motivate them. This is, of course, a difficult and mostly unattained task.

In contrast to these prevalent theoretical constructs, diSessa and Cobb propose the idea of ontological innovation, the idea, familiar from the realm of natural science, that it is necessary to 'develop theoretical constructs that empower us to see order, pattern, and regularity' in the settings under investigation. And here is the crux of a challenge confronting constructionism as a theory.

The tendency of most educational research - and by implication the producers and consumers of education research - is that the fundamental concepts remain invariant over time and technologies. It is tempting to take this observation as merely trivial: educational change is slow, it seldom takes account of the possibilities of knowledge transformation, and it is almost always concerned only with teaching more effectively, rather than learning within new epistemologies.

[2] I am aware that one of the flavours of constructivism is social constructionism. But here too, while the role of others in the development of individual cognition, and even technologies, is acknowledged, the nature of knowledge is largely uncontested. One exception to this is the French didactic tradition.

While this is true, it misses a key point about constructionism. When we build, we build with things - not just ideas. Of course, if we design properly, the things we build with have an epistemic foundation - of 'powerful ideas' say, that students are supposed to bump into. But the ways the things are connected, the relationships between them, and the behaviours they are given have to be expressed in the system of the things, not in the system of ideas. I could express the fact that the paragraph settings of this paragraph are contained in the final paragraph marker as a line or two of code; but as i am building this paragraph, it is much more natural to say (to myself), "If I merge this paragraph with this one, it will inherit the second one's properties. Note the informality of my expression: 'this' means nothing outside the situation of writing.

This particular property of construction systems (like programming languages) is both a powerful advantage and a difficulty. It is powerful because the complexity of an idea often inheres as much in the way it is represented as the idea itself, and being able to express the idea without learning a new language allows access to otherwise inaccessible knowledge. But it creates a difficulty that educational research has yet to confront: that of building an ontology for constructionist environments which, by definition, involve expressing relationships between objects.

Let us give an example of this problem. Some years ago, Celia Hoyles and I noticed a recurrent pattern in students building computational expression for mathematical and scientific ideas. We saw that while they seemed often clearly able to abstract from the particularities of the activity, as evidenced by an often implicit recognition of the relationships between variables, these abstractions did not resemble in their expression, the standard forms of algebraic or even quasi-algebraic representations. Naturally enough, they employed the tools - objects and relationship between objects - with which they had used successful in the activities. The tools-in-situation, in other words, acted as a means to express abstractions that might not have been expressible in standard forms: we called these 'situated abstractions' to try to capture this idea.

Situated abstraction is an ontological innovation in the sense that it identifies and organises a class of behaviour and expression that occurs in the context of activity in constructionist environments. Like any useful theoretical idea, its power lies in its application - in the potential of the idea that started as an observation of behaviour to influence and shape behaviour. The hypothesis is that there exist a range of ontological innovations that are yet to be identified and used, and that there exists some common element among them that derives from the constructionist setting in which they are idenfied.

Any ontological innovation worth its salt becomes natural over time: perhaps the most famous example is that of 'powerful idea' and its associated design principles (e.g. Papert's "Power Principle". And while 'powerful idea' works as an ontological element of educational discourse, the idea of ontological innovation is itself an ontological innovation in the domain of research. In that respect, we could ask how ontological innovations arise - and particularly, how can we use the idea here at our conference.

As I said earlier, I think we have at this conference a tremendous opportunity to study examples of genuine constructionist practice. Can we abstract from these examples some organising structures, elements of a new ontology for educational transformation? Can we develop elements of and relationships between those elements -- a grammar for constructionism?

This is the challenge for this conference: to reconstruct the idea of constructionism and transform it from a framework (or slogan) for action into a set of ontological innovations, ways of conceptualising what people do in constructionist environments, and to use this as in a way that can assist us in designing those environments.

Seymour often made the point that we don't have a language for discussing the kinds of radical transformation of learning (and by implication, schooling) he envisaged and most of us have inherited. It is time to invent that language as a step towards realising his and our radical transformation.

I think that, at last, there is a convergence of technological forces that will mean that we are cutting with the grain. It is the complete transformation of human interaction that has been occurring for about 10 or 15 years, and which shows no signs of abating. This is not technological determinism: the mobile phone has transformed the way people meet - that is obvious and while the phone itself doesn't of course have agency, it makes perfect sense to talk about a technological transformation as shorthand for a social transformation catalysed by technology.

Lenin once said that for revolution, only two conditions were necessary3. One it that people cannot continue in the old way. That is true for us now: educators cannot continue for much longer teaching what we have always taught, ignoring the possibilities and potential of technical transformation, and simply squeezing the last drop from the infrastructures that were set in place in the nineteenth century. Neither can we, ourselves, ignore any longer the importance of engaging teachers as agents of change, rather than something to be changed (or even, in some versions, abolished).

The other condition was that people could see some alternative to the old way. For that to happen, we need more than the instances we will hear about this week: we need the language, the grammar, the ontology and above all, the voices of students and teachers that will help us reconstruct that vision into a reality.

[3] I'm leaving aside how we *get* to our revolution: maybe Wally Feurzeig is right - evolutionary change can build to revolutionary change.

Dance & self-confidence

Ioana Cristina Ocnarescu, *ioana.ocnarescu@gmail.com*
PhD Student, Product Design and Innovation Laboratory - Arts et Métiers ParisTECH in collaboration with the User Experience Design Team - Alcatel-Lucent Bell Labs, France

Abstract

This paper proposes a living and interactive application that improves one's body language and dancing techniques. The work is the result of a close collaboration between designers, dancers and technicians. It shows how design can create a dance experience using new technologies in a constructionist approach (Figure 1). The interaction with this application is both physical and emotional. The system is a dance tool that therapeutizes the dancer; it cures his lack of confidence and his inner fears about dancing. The protagonist spends an enjoyable moment during the interaction with this application; he learns to dance and to better understand his body.

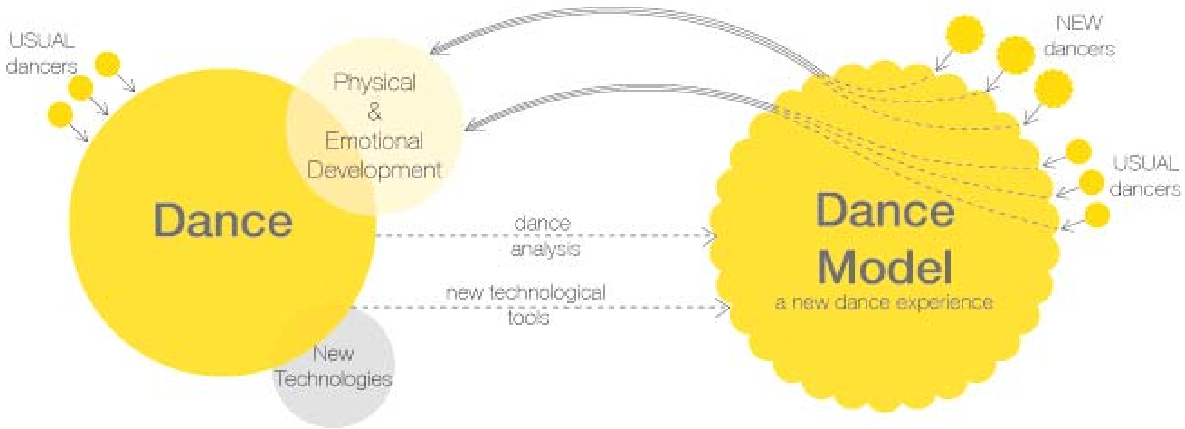


Figure 1. Project scheme

The core of this dance model is a shadow that is projected on the wall. It is this constructionist tool that the paper proposes. This silhouette has the user's body and once it is on the wall, it teaches him how to dance. Besides its role as a teacher, the system proposes other functionalities and characters for the shadow in order to create a close complicity with the protagonist. Words, music, images and movement are used in this interactive system to permit the protagonist to understand the dance practice and to develop himself, both physically as emotionally, leaving behind his fears and complexes about dancing.

The collaboration between dance and new technology reveals new tools that allow dance to be learned and understood differently. This system presented in this paper uses such tools in a constructionist manner. It proposes a new model for dance learning; it creates a subtle environment for a physical and emotional development and it creates even an extension of the dancer's physical body to new virtual dimensions.

This work was presented as a final diploma project of the Industrial Design and Innovation Management Masters Program (State College Designers) in October 2009.

Keywords

Dance and new technologies; interaction design; experience design; body representations; interactive shadow; dance teaching; dance complex and dance fear.

Understanding the context

From September 2008 to December 2009, I attended the Masters Program of Industrial Design and Innovation Management from Strate Collège Designers (www.stratecollege.fr). The year spent in this school made me realize the importance, the force and the responsibility of design in our everyday life. The academic cycle of this master's program was composed from three parts. The first part consisted in attending intensive courses on design methods, design thinking. I also learned different design expression tools like drawing, painting, preliminary sculpture and 3D modelling. The second part involved active participation in several design projects with industrial partners like Microsoft Surface (www.microsoft.com/surface) and IRI Centre Pompidou (www.iri.centrepompidou.fr). The last step of this education program was a final project on a personal topic. I chose my research field, I found a problematic question and, as a design thinker, I answered this question.

Project motivation

Impassioned by dance, the choice of my research field was almost instinctive: dance and new technologies. Dance is an activity that I have practiced from 5 years old. From ballet to ball dancing and from flamenco to step dance, this activity had a major role in my personal development and my social life. I considered that my project was an opportunity to write about my passion, to go deeply in the history of dance and to find out its importance in the history of humanity. I centred my research in a technological context. I questioned myself about the influence of the technology on dance and the benefits of this collaboration in our everyday life. Dance lost its major importance in humans' life. From animals dance to ancient civilisations, dance was considered a vector of life, a bond with the gods and an activity that connected body and mind. I decided to investigate how the technology development could help the dance to re-become a central activity. What is the new model to be created using the collaboration tools between dance and new technologies in order to reintroduce dance in human life? How the constructionist approach could help in generating this model and in bringing balance to a new equation where dance is a major parameter?

First step - The mémoire

The first six months were spent for documentation and research on the chosen subject. The result of this work was synthesized in a report named *Dance and new technologies*. This memoire presents the history of dance, its evolution and development within the new technological environment. The purpose of this report is to underline the importance of dance in human life and to analyse the benefits of this practise in a technological context.

Matos (1998) describes the challenges of the collaboration dance-technology: "Science and Technology stress the marvellous capacities of man, and their appropriation supposes the risks due to their power and presence, but also the risk of displacing their utilitarian codes in order to explore the constraints and freedoms they represent." How to use the power of the technology in order to strengthen dance and what are the results of this collaboration? "We need a connection with technology. There is no better art than dance in which to bring this about. Jean-Marc Matos, you are doing an important work" is written in the same journal, Matos (1998), by John Cage.

From Loïe Fuller to the birth of the cinema, and the early stages of the video, from the creation of the theremin passing by the beginnings of the interactivity, dance development reaches new dimensions. The creation of technological tools like Merce Cunningham's Lifeforms - that proposes a new way for transmitting and learning dance, the use of sensor systems - that permits the mapping between music and dance, and the creation of other systems of body perception, commit an important change of dance pillars, the body, the music and the environment. The virtual dancer, the hybrid bodies and the creation of environments where the artificial and the real dance come together are new ways of exploring dance.

Being a dancer that uses technology means exploring all these new tools and working with different specialists, in an interdisciplinary team. The time of the Laboratories has come. Lahunta (1997) explains this fact in one of his interviews with Jean-Marc Matos: “for the 21st century, the skills we have developed as homeless survivors may turn out to be just the tools needed to create a niche at the center of a future interdisciplinary, high-tech and body-centered project for researching, teaching and creating.” But how to use these skills and what could we imagine, create and improve?

For the moment these technological tools are used only by artists. In our everyday life, there are only timid collaborations between dance and technology. Dance is nowadays either a personal hobby or an activity practiced for time to time at special occasions. The collaboration dance-technology is materialised by products that come mostly from the game industry (Dance Dance Revolution, Wii). However, the existence of special events like concerts or art exhibitions shows to the grand-public some of the possibilities of the technological tools applied to dance universe.

Taking into consideration all these factors, a final question rose from my report: **How dance, using new technologies, could reaffirm the human being both physically and emotionally?** What kind of new model could be imagined in order to explore dance therapeutic capacities? A new dance dimension could be created that would give a simple, instinctive and fresh access to both the inner and outer world of the protagonist. And such model and dance extension belong to the constructionism philosophy.

Second step - The target analysis

The future dancer

How dance, using new technologies, could reaffirm the human being both physically and emotionally? To answer this question I had to understand who is going to be my target, my future dancer. Some of the people do not dance, do not want to know how to dance and are happy with it. However there are others that would like to know how to dance but, for various reasons, they fail to practice. After interviewing twenty persons, several problems related to dance practice were found. Sometimes, the dance vanishes from our lives with a simple sentence like “I have no time for this”, or it represents something that brings in fear and complexes. After all the interviews I succeed in determining two groups of people, with different concerns about dancing (Figure 2).

I called the target **“The discouraged”**. They like dance but they are afraid to practice it; they are too shy and don’t like seeing themselves dancing. They usually say: “We move but we do not dance; seeing others dancing feels better then trying to enter into the rhythm”.

The second group was called **“The got no time”**. These people are snowed under jobs, tasks and responsibilities. They say that they do not have the time and the perfect occasion to dance, even if they really enjoy this activity. They also do not like the dancing atmosphere from dancing-clubs or they do not know were to go in order to exercise this practice.

These persons lost their intimacy with dance; they lost their self-confidence with their bodies. They fear to put their bodies in movement and being seen by others perturbs them physically and emotionally. When they find themselves in a dancing situation, they may transform this fear into a total repulsion toward dance. And maybe by loosing dance, people lose a degree of intimacy with themselves. They lose a way of knowing and expressing themselves. They lose a language. How to get these back? How to re-create one’s personal dance? How could the constructionism and the design put hand in hand to create the perfect dance environment which easily allows the user to learn how to dance, to understand the capabilities of human body, to spend a pleasure moment and to gain self-confidence?

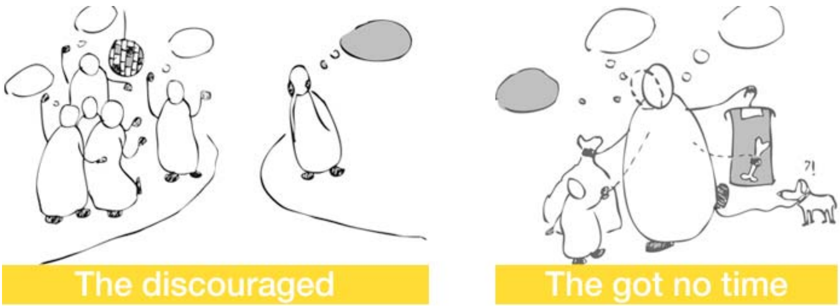


Figure 2.The target representation

Deeper into the dancer's mind: How do we see ourselves while dancing?

What fascinates us while dancing? What scares us? In order to generate a perfect environment for the chosen target, one has to answer these questions. "Man has always been fascinated by his own image and has attempted to reproduce his double, to extend and perfect that image", tells us Jean-Marc Matos, Matos (1997). And maybe by studying the way a dancer sees himself while dancing, by understanding the factors that perturbs the dance, one could imagine a new model that takes into consideration all these parameters and creates a new dance experience.

Supposing that the self-representation of the dancer is an important parameter of this model, how does this image transform one's feelings, behaviour and personality? I tried to understand how the dancer's body and its movements are projected into his own mind. For this I divided dance into three fields: individual dance, dance with a partner and group dance. This typology could be considered as related to the degree of intimacy of the dancer with himself and with his environment.

Individual dance: During the individual dance, the person has no sense of shame or fear; he is not limited by the eyes of the others. The movements are free. People express what they feel for themselves and with no restraints. Dancing alone means communicating with oneself, strengthening the bond between body and mind.

Even if we dance alone, we are dealing with different representation of ourselves. For example, the physical presence of our body makes a difference. Let's imagine a person who is dancing in front of a mirror. When a dancer sees himself in the mirror, he compares himself with an imaginary perfect body. Sometimes the fact of seeing your body moving and dancing is disturbing; it brings in fear and complicated feelings; it dissolves the enjoyment and the pleasure of dancing. On occasion the dancer in front of a mirror finds himself in a situation where he wants to impress himself. For professional dancers, the mirror is correlated to the idea of improvement and thus its use is extremely important.

On the other hand when we dance alone and without seeing ourselves, the self representation is completely changed. Usually the idea of improvement disappears when there is nothing around us that reflexes our body. Thus more pleasure is gained and we imagine ourselves dancing the perfect waltz. We are what we image. The fact that we do not see our body makes us free; we gain a degree of liberty. We do not consider ourselves ridiculous and imperfect. We just dance.

The dance with a partner: A question rises in this case. Does the other accept your individual dance? And do you perform your individual dance when someone is around you? If the individual dance shows your inner feelings, performing it shows the other who you are. That is why the relationship between the two bodies while dancing is the expression of the degree of intimacy of the two persons. In addition dancing with the other has its own magic: the communication between partners. Yet this communication is reduced by our fear, panic, and emotional complexes. That is why we choose to move and not to dance. We chose to reproduce some movements that we know by heart and not to communicate.

Group dance: During this type dance, all the protagonists share the same space. Each dancer sees himself in the eyes of the others. It is difficult to de-correlate oneself from the regards of the others, when they watch and judge. During this dance there are used societal rules that spawn complicated fears. Dance is influenced by the degree of intimacy between each participant. However the spontaneity and the intimacy of the dance are vanishing. Too many rules? Too many fears? Is this a cultural or a social problem?

What interested me most in these typologies, were the dancer's feelings during his dance. And also the fact that these feelings are correlated with the representation from the dancer's mind and with the environment that surrounds him. It is a two side factor, one from inside and the other from outside. They are combined in a very complex way. That is why I asked myself if I could find a suitable tangible representation of the dancer's body, a new constructionist model that could improve his self-confidence and teach him how to dance. This model could make the dancer's fears disappear and free his body by using the new tools of the collaboration dance – new technology.

The CONCEPT

The suitable representation

I had a QUESTION ("How dance, using new technologies, could reaffirm the human being both physically and emotionally?"), a TARGET ("The discouraged" and "Got no time"), and I chose to work on INDIVIDUAL DANCE. Thus the context of the application was established. In order to improve the protagonist's dancing techniques and increase his self-confidence, I decided to search for a suitable external model, for a tangible tool, that would incite the protagonist into his individual dance. Constructionists build mental models to understand the world around them; in the same perspective I was looking for something that would help the dancer understand his body, teach him different rhythms and stimulate repetitive practice.

In the first time, I was trying to find a perfect external image that would represent lightness, freedom, an unconstrained dance. I was inspired by the world of plants, by the dance of the dandelion seeds. I imagined an interactive application with this plant. A dandelion is projected on a wall and it is sensible to the user's gestures. At the beginning just a seed, the development of this plant depends on the user's movements. The body of the dandelion and its seeds copy the movements of the dancer's body. The purpose of this application is to make the person in front of the application forget about his inner-fears and complexes and feel the dandelion lightness. Trying to make the dandelion dance, the protagonist will make himself dance. Thus without noticing, the user frees his body and starts to dance.

In order to create a daily motivation, the dandelion is growing everyday. Each day it is different from the day before. Its physical development (number of the seeds, height) is influenced by the intensity, the speed and the amplitude of our actions. The person's lack of movement destroys the plant.

But what are the advantages and the disadvantages of such an application in relation with my first question? This application incites the user to move his body in order to keep the plant alive. The plant is the constructionist tool and by its interaction the person imagines himself differently, light and free. However there is not any choreography related to these free movements of the protagonist. The dandelion represents the dancer's body in a poetical way but it does not teach him any dance techniques or music rhythm. That is why the dandelion image did not respond entirety to all my requests; it is an insufficient model that does not take into consideration all the dance parameters.

Figure 3.The dandelion waltz - concept representation

I searched further for another representation that could better translate the user's body and his movements; something that would be also an extension of the dancer body. An idea came when I thought about my childhood and how I was training myself at the age of 10 years old. I did not use a mirror but I was putting a light on a wall and I was creating my shadow. I was dancing with my shadow for hours. The image of the shadow came in front of my eyes. That was the constructionist tool I was searching for. That could perfectly create my model! A pure body, an elegant movement, a perfect surface... But what is the magic of the dance with the shadow? My shadow is myself but it has something else. She is perfect. It could be taller than me, slimmer then me, but in the same time it represents me perfectly. What if this shadow teaches me how to dance? What if this body, which I recognize as my self image, becomes a part of my life, it dissolves my fears while dancing and it creates from dance a pleasant moment?

From all these memories, I imagined the application *Dance with me* (Figure 4). The dancer's shadow is projected on the wall. Once this silhouette comes alive it becomes another entity. The user interacts with his shadow, who is a teacher, a choreograph, a dancer, a partner and even a friend at the same time. Therefore dancing with this shadow means dancing alone, but also dancing with the other. It combines almost all the dance typologies presented earlier in a mystical way. This shadow will teach how to dance. It is a pure and elegant representation of the dancer's body. It is the constructionist tool that creates a new environment where the protagonist sees himself differently, feels differently and acts differently. The dancer sees his shadow, moving like a feather, so easily, so gracefully and this opens new doors for him, it takes him on a journey with new access to his inner and outer self; it gives the dancer the courage to free his body, follow the music, deliver himself from his fear and complex and spend a enjoyable dance time. But how could this model work? And how could it interact with the user both physically and emotionally? The next section will present a scenario of the interaction with this virtual environment.

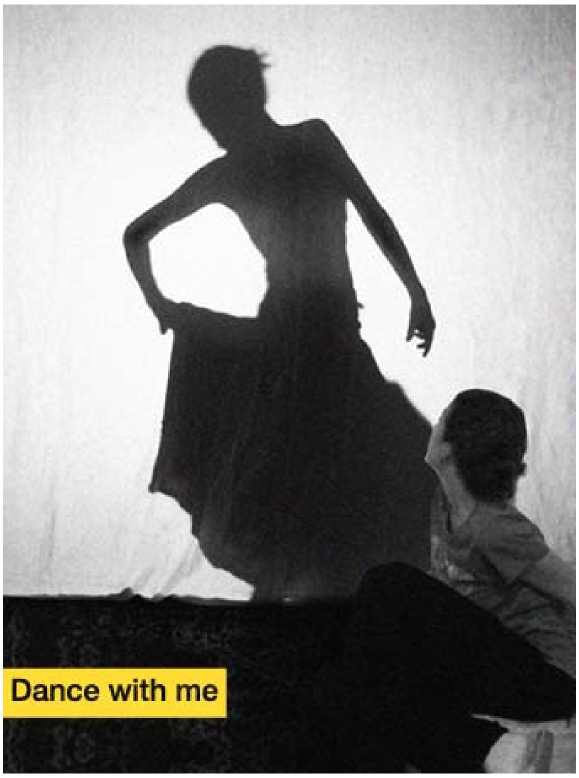


Figure 4.Dance with me - concept representation

A living creature with different characters

Dance with me is an interactive application that one could use at home. Either on our screens or by a projection on the wall, the system copies the human body and it represents it in the form of a shadow. A strip of light connects the protagonist with his shadow.

The shadow plays different roles in the user's life. Firstly, in order to start the application, the person has to call his shadow. This step is named "The waking up". By a gestures tracking unit, the system detects the intention of the protagonist while he touches the wall. The person waves his hand towards the place where the shadow will appear. Once the shadow comes into view, it recognises the face of the person or if this is the first time the person uses the application, the system registers new data (Figure 5).

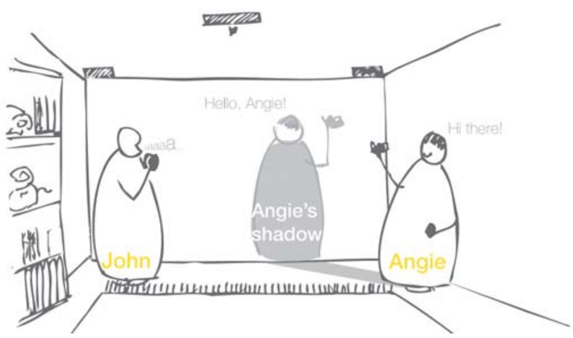


Figure 5. The waking up

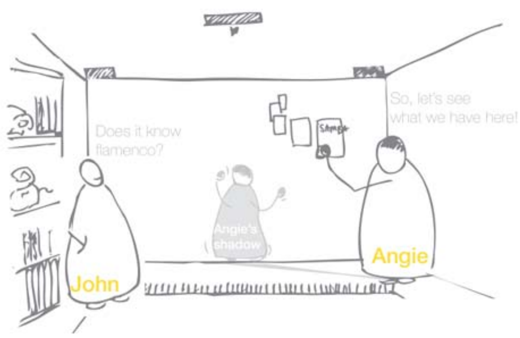


Figure 6. The dance lesson

Once this registration step is over, the shadow starts a short dialog with the user in order to find out the dance that suits him best at that moment. The use of an artificial intelligent program permits a fluent discussion and objective data analyses. The shadow proposes different dances; the protagonist chooses one proposal and “The dance lesson” begins (Figure 6).

The lesson starts with a short history on the chosen dance. This story introduces the person in the dance appropriate atmosphere. The next step is the warming-up. “Never dance without warming-up” explains the shadow. It proposes a personalised warm-up program, proper for the chosen dance. Thus the shadow stats to move on the wall and encourages the person to participate and follow its movements (Figure7).

Figure 7. The warming-up

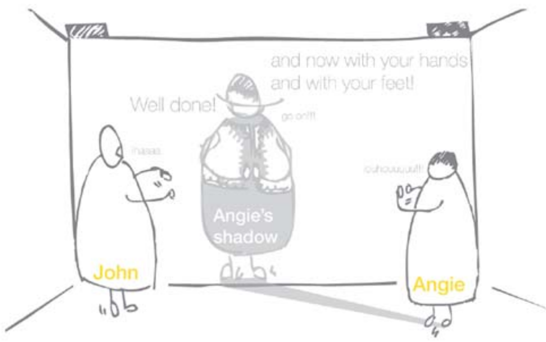


Figure 8. The dance

Once these preparations are finished, the dance lesson begins. First the shadow presents the music, its rhythm and step by step it shows simple moves of the hands, of the feet and then of all the body. It always encourages the user and it stimulates him to understand the music and the dance movements (Figure 8).

The lesson evolves depending on the energy and the interest of the protagonist. During the dance, the system detects the dancer’s motivation, his signs of enthusiasm, disappointment or fatigue. If a crises situation appears, the “Encouragements program” begins. The application proposes different supporting programs to surpass the difficult moments. Sometimes it shows the dancer’s progress from his first lesson or re-explains the basics steps. Other times it underlines the music rhythm and increases the volume. On occasion it uses images, objects or attitudes in order to explain a complicated dance move (Figure 9).

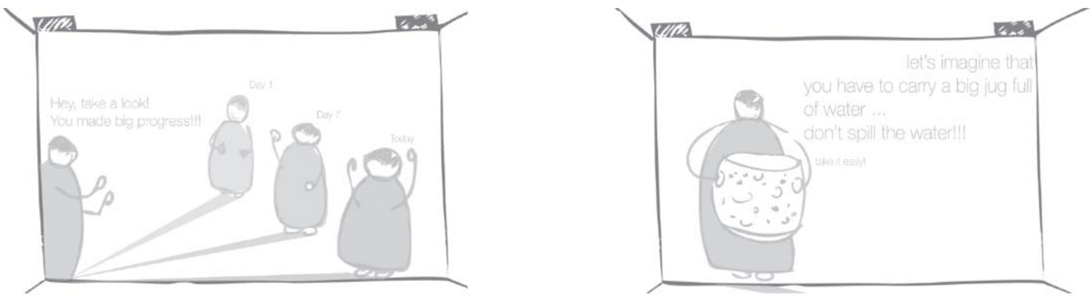


Figure 9. Encouragements program

Another stage of interaction is called “The real shadow”. The user has the possibility to add on the wall his real shadow next to the teacher shadow. This way the dancer could see his moves like in a mirror (Figure 10). However if this second image disturbs the user, he can immediately close his real shadow.

But above all, the shadow has a responsibility. It has to create from dance a pleasant moment and a frequent activity. What happens if the person, by commodity or by the lack of time and energy, does not practise for a long time? In this case the shadow suggests that a dance lessons should commence. In order not to disturb the person from his activities, a gentle ray of light will connect the feet of this one to the wall where his shadow is waiting for him. This way the real body and his virtual shadow are connected and the dancing lesson can begin (Figure 11).

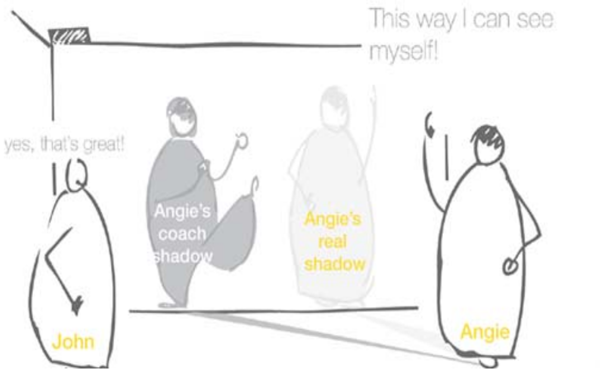


Figure 10. The real shadow

Besides all these features, there is another interaction stage named “The free dance”. If the person turns on some music and starts to dance near the application, the system detects his action and acts differently. The shadow appears; it detects the music and the dancer’s movements; it recognises the rhythm and it also starts to dance. It will not be a dancing lesson but a pleasant dancing moment. Sometimes the silhouette copies the person’s movements and it stretches them to the extreme (Figure 12). It detects the energy of the dancer’s bodies, the contractions of his muscles and it extends his actions. Seeing the shadow performing such impressive movements the protagonist is seeing himself improved.

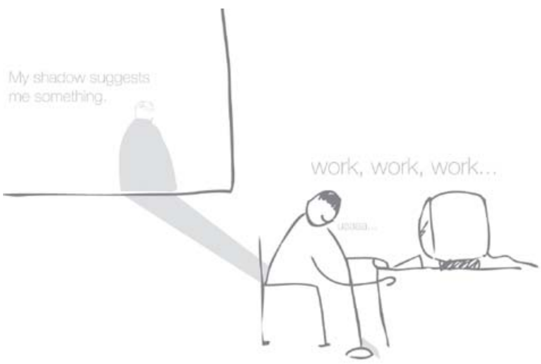


Figure 11. The responsibility

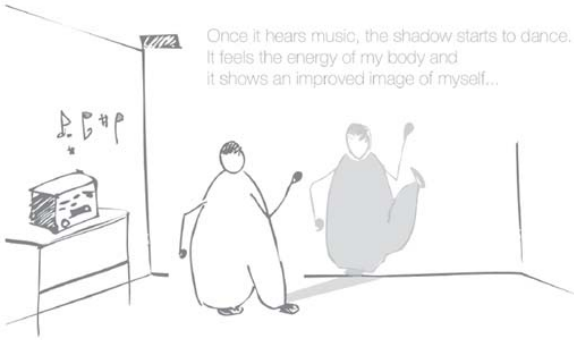


Figure 12. The free dance

The application offers as well the possibility to represent many dancers, each of them having their own shadow. The case of two persons in front of this application was imagined. Each protagonist calls its own shadow and a ray of light connects the users to their shadows. Both shadows start to talk and after this introduction step, they begin to dance. The persons watch their shadows dancing on the wall (Figure 13). They see their silhouettes waltzing. The invitation was made, the ball was opened. Maybe this way the persons will also start to dance. Maybe this way the imaginary barriers that exist between the two persons, their fear and shame, would be dissolved (Figure 14).

What is the impact of seeing your shadow dancing with another shadow? And how this representation would interfere with the human intimacy while dancing? Does such a representation facilitate the dance with a partner?

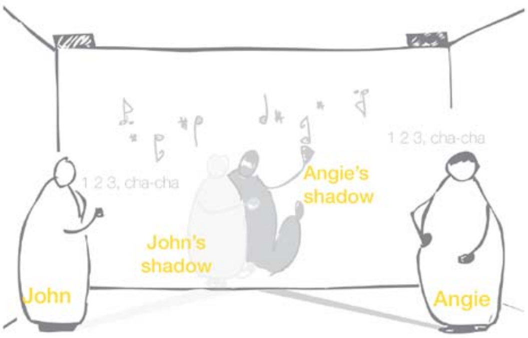


Figure 13. The dance of the shadows

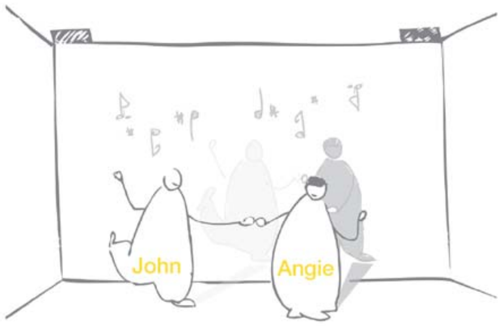


Figure 14. The double - waltz

Further on, the concept could be extended to the group dance. Such an application would be well suited in a dance class. In this case the real teacher would have a virtual assistant, a virtual help that would detect the students' motivation and their difficulties. What could be the impact of such interactive walls in schools, were children could discover their body, understand the rhythm of the music and learn how to dance? What kind of new activities could be developed in order to explore the human body's capacities? The development of body tracking systems, the amazing progress of artificial intelligence, the dancers' expertise and the designers' imagination give us a gleam over the force and utility of such applications. Once the tools are invented, the environment settled, the interaction with such an application could get bigger with any new scenario imagined. And as constructionist models are tools for exploration, they are never finished and always under construction. As for the presented model, it is hoped that a physical prototype would validate its scenarios and create others (Figure15).

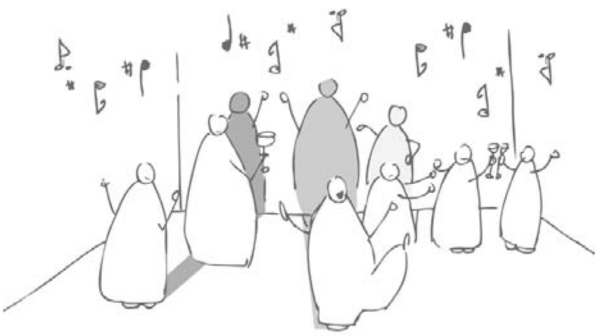


Figure15. Future scenarios – social event

Comments and analysis

Application analysis

When I started this project I had only the passion for dance and my personal experience. However in order to create the application *Dance with me* I met different people. Dance teachers, technicians, dancers and teachers helped me to find out the value of my concept and its utility. The first constructionist features were also underlined by discussions with these persons and by the end of this project I understood the major characteristics of this theory: the creation of a dance model accessible to a large group of people, a model that cures dance's fear and complex and that offers physical and emotional development while the dance practice.

My first report *Dance and new technology*, and the information that I gathered while writing it, was also very useful. It gave me a background, a technological context and the most important thing, it gave me a challenge: to make the real world dance with the virtual world and to extract the benefits of the collaboration between dance and technology.

The application *Dance with me* proposes an integration of the new technologies in our home environment. It is an application that we can use in our everyday life. It is a living and interactive application. Its main purpose is to make the user run over his frustrations and complicated fears about dance, to free his body and to spend a pleasant moment. In order to achieve this purpose, technology shows its possibilities: artificial intelligence, system projection, system gestures recognition. Every physical unit of this application has a vital role. The artificial intelligence makes the real and the virtual talk to each other, creates their complicity. The system projection captures the form of the user's body and projects it on the wall. Once it is projected, this silhouette is matched to a basic framework in order to create an independent entity. This way the movements of the silhouette on the wall, that has the same body as the protagonist but do not depend of the user's movements. It is a teacher, a choreograph, a dancer and even a friend for the user. Lastly, in order to see the person's motivation, a gestures recognition system was also included in this application. Thus the system captures the motivation of the user, his fatigue and satisfaction while dancing and adapts the dance program to his capacities and rhythm. All these tools are used in a constructionist aim, as constructionism encourages design that illustrates how technology can aid to learning, thinking and education; in this case it is imagined a new way of learning dance and an interaction both physical as emotional with the dancer for a self rediscovery. Going further, a constructionist system has the capacity to use the technological tools in an invisible way in order to live a life experience, to learn and to develop yourself. That is one of the important goals of this application.

In the game industry, the technology required in such application already exists or are the ready to be commercialize. For example Microsoft Project Natal proposes a similar the technology like the *Dance with me* application. Project Natal enables users to control and interact with the Xbox 360 without the need to touch a game controller through a natural user interface using gestures, spoken commands or presented objects and images. It is scheduled to be released during the summer of 2010.

Besides the game technology, in artistic world different laboratories experiment the results of the meeting between dance and technology. Their systems and solutions represent new opportunities for applications like *Dance with me*. Such a project is *Electric Shadow*, created by Naziha Mestaoui, architect, and Yacine Ait Kaci, multimedia creator. "With Electronic Shadow, architecture becomes a reality expansible to infinity, a support for interactive projection in which the interface or focal centre remains the inhabitant or the visitor...", Mestaoui and Kaci (2000).

Shadow, shadow on the wall, who in the land is fairest of all?

One could ask if the shadow is the suitable tool representation for a dance lesson. This silhouette may hide some details and it may introduce some uncertainties in the dance lesson.

But these details and the dance technique perfection is not the goal here. The technology permits to project on the wall a 3D body for the perfect comprehension of dance movements. However the shadow was chosen for the impact of its image, for its representation in human mind. This pure form induces new values over the user's body like lightness, freedom, easiness. In the constructionist approach this representation is the tangible object that sustains the model; it is the object that connects the real to the virtual and that triggers the self discovery: it shows the protagonist the physical capacities of this body and gives a mental equilibrium while dance practice.

Beside the benefits of its appearance, this silhouette plays different roles in the mind of the protagonist. Sometimes it is an entity that teaches dance, somebody different from the user. It has his body but the dancer will not identify himself with this image (Figure16a).

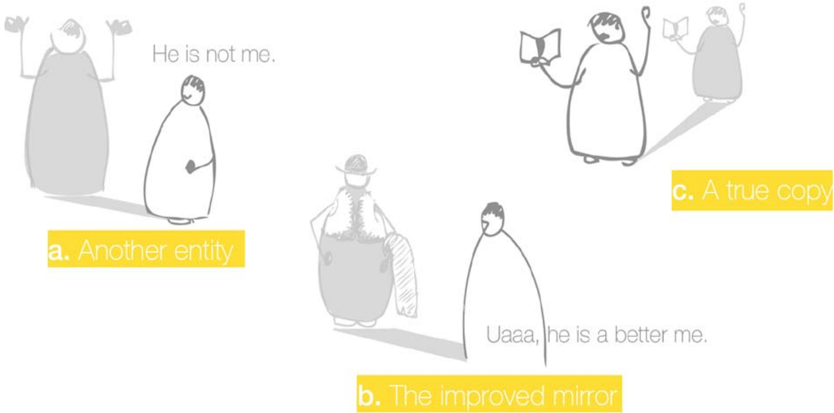


Figure 16. The shadow - another entity, the improved mirror, a true copy

Other time the shadow is an improved image of the dancer. The user could identify himself with what he sees on the wall (Figure16b) even if his action are different from the action of his shadow. And finally, in order to create a tight relationship and complicity with the person, the shadow copies everyday scenes of his life. In this case it a true copy of the user (Figure16c).

By these different characters the application creates a close collaboration between real – virtual and a perfect integration of dance and technology in the user's everyday life. An important point to underline about the projection of shadow is the fact that it is a digital data. Everybody could have a personal shadow and this data could be registered, exchanged, modified and even stolen. What is the impact of these personal representations on our network? This form opens up the doors of the digital words and plugs us into the virtual realm: "a territory – or a state – that should no longer be opposed to reality, but considered as a natural and enriched extension of it", Hémery (2010).

Conclusions

Dance with me application imagines a constructionist model for a new way of experiencing dance learning. The model proposed here is an interactive and intelligent environment whose principal tool is the user's representation, his shadow. This silhouette interacts with the user both physically as emotionally, in a magical relationship bathed in the complicity between real and virtual. While interacting with this application the protagonist surpasses his fear and complex regarding the dance; he improves his self-confidence by learning how to communicate with his body, by understanding the connection between body and mind and by spending an enjoyable moment. This system reduces one's inhibition which now is replaced by enchantment.

This virtual environment makes use of the technology in a subtle way. The system uses technological tools in order to strengthen the relationship of the user with this application; it proposes a personalized practice for each protagonist and acts depending on the user's motivation. From the physical point of view, the system offers dance benefits such as equilibrium, flexibility and stature. It rests however an educative application for emotional and physical improvement. Moreover, *Dance with me* is home incorporable and adaptable for a large public due to its playful and practical qualities. Furthermore, this model could be also used in schools during dance classes as a tool for discovering the capacities of human body and for taking dance comprehension to another level.

Acknowledgements

I wish to thank all my teachers from Strate Collège Designers who guided me during this project: Eloi Badoux, Etienne de Frileuze, Dominique Sciamma, Monique Touati.

I thank my friend Huieun Kim (student designer Strate Collège Designers) whose sensibility and design philosophy inspired me all along.

I am grateful to Anne-Aure de Frileuze (dancer and dance teacher) and to Natalie Ebenreuter (dancer and interaction designer) for fruitful discussions on their dance experience.

A big a note of thanks to James Clayson for the most interesting discussion on design, literature, and constructionism and for his encouragements that motivated me writing this article.

References

Matos, J-M (1998) *Dance and Technology: A General Statement*. Mindship International, www.mindship.com

Matos, J-M (1997) An interaction with Jean-Marc Matos, Scott Sutherland's diary, www.scottsutherland.com/dancetechnology/archive/

Lahunta, J (1997) *Statement on dance and technology*. Scott Sutherland's diary, www.scottsutherland.com/dancetechnology/archive/

Mestaoui, N and Kaci, Y A (2000) *Electronic shadows.* www.electronicshadow.com

Hémery, A (2010) *Electronic Shadow*, Extract from the heading Image, Intramuros n°146

From *Learning by Playing* to *Learning by Programming*

Katarzyna Olędzka, *katarzyna@oeiizk.waw.pl*
Computer Assisted Education and Information Technology Centre (OEIiZK), Warsaw

Abstract

Inspired by Papert's idea of learning and microworlds we try to put it into practice in our centre (Computer Assisted Education and Information Technology Centre - OEIiZK, Warsaw). We have prepared different workshops for teachers and children to spread this idea.

Firstly, it is worthy to examine computer games. Watching children playing games, one can observe that something important is affecting them. They learn skills such as data manipulation, strategic planning, making snap decisions or negotiating. Their emotions strongly affect motivation. So we can ask, what will children learn by making a game?

Secondly, a very popular activity which teachers do with children is storytelling. Creating animate presentation students gain new knowledge about the world, they improve their understanding of human nature and feelings. Moreover, they become fluent in using ICT.

Thirdly, storytelling and creating animations with descriptions and sounds are perfect ways to present some scientific ideas. Working in microworlds usually leads to a deep and broad learning experience. Students are ready to put much effort and spend a lot of time to prepare such projects.

Number division *Holiday adventures* *Which rectangle is longer?*

Figure 1. Some examples of projects

Keywords

playing, learning, programming, children, Imagine Logo

Introduction

Seymour Papert wrote in *The Connected Family:*

"I have seen many dozens of advertisements for software that make promises like: *Such Fun Your Child Won't Know She Is Learning*. I am horrified by the message. The suggestion is that learning is a nasty pill that must be sugarcoated with fun and games. It is true that learning has sometimes been given a bad name by poor practices in school and even by some parents whose constant refrain is *Do your learning. You can have fun afterwards.* But one of the great things the computer can do is turn this around and store the kind of enjoyment of learning you see if you watch an infant or a scientist. Both are learning all the time and they both know it and they love it."[1]

Inspired by Papert's idea of learning and microworlds we try to introduce some of these ideas in our centre (Computer Assisted Education and Information Technology Centre - OEIiZK, Warsaw). Together with Wanda Jochemczyk and Agnieszka Samulska we have prepared different workshops for teachers and children. The aim of this article is to present some ideas and experiences together with examples of students' and teachers' work.

Games in Imagine Environment

Computer games are a unique form of media. In strategic games players win by successfully navigating and meeting challenging tasks. They learn skills such as data manipulation, strategic planning, making snap decisions or negotiating. Their emotions strongly affect motivation. The first time children play, they usually fail, and then they have to do it over and over again, until they master the skill, they gain the knowledge or just win.

"When the reviewers like a software, they are full of gush such as: *easy to use*, *marvellous graphics*, *children love it*, *lots of learning*. When you talk, you raise more controversial questions. Half the time I can't agree with you but you get me thinking. Something important is affecting our children. What we need is not being told it's good, it's bad or it gets four stars like a restaurant. We need to talk more, argue more, think more about what lies behind it all."[2]

Although some of the educational games represent innovative approaches to learning, the majority aren't successful. Why? Because they are not attractive, children do not find them funny to play.

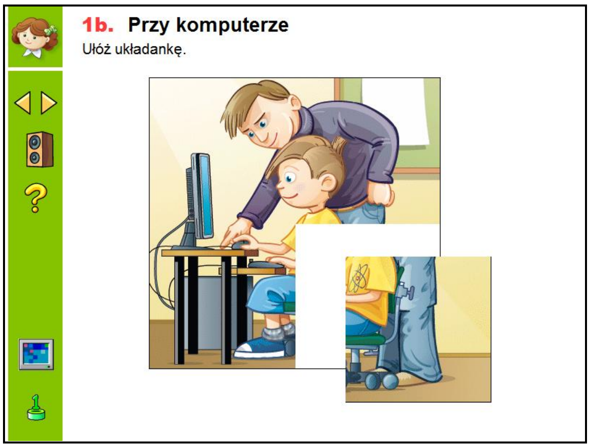


Figure 1. Jigsaw

[1] Papert (1996), p. 50
[2] Ibid., p. 7

There is a common believe that when children dislike schoolwork because it is too hard. “Nothing could be more wrong. Most dislike of school work comes from finding it boring, the exact opposite of finding it too difficult. Children, like everyone else, don’t want **easy** – they want **challenging** and **interesting** – and this implies **hard**.”[3]

An example of an interesting application is jigsaw where a child has to put many small elements together to make a large picture. To do this, analysing and synthesising is required. The difficulty level is carefully designed to gradually introduce more difficult problems. Both children and adults enjoy the puzzles and learn some sophisticated skills solving it. There are many other examples of puzzles which simulate thinking.

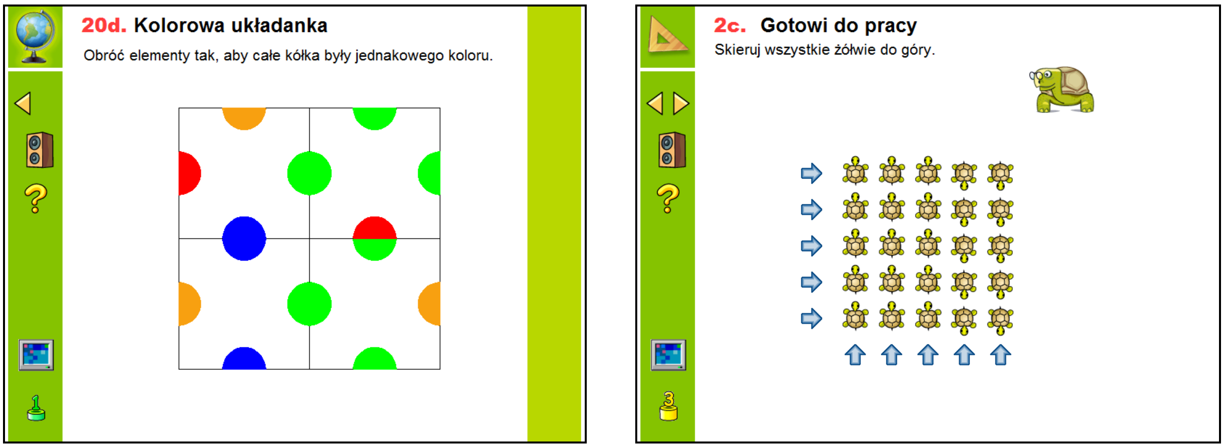


Figure 2. The examples of puzzles - Colourful Puzzle and In the Same Direction

Colourful Puzzle - You can turn right and left square pieces. The aim of this game is to join square peaces with the same colour of dots.

In the Same Direction - By clicking on an arrow near the row, you turn right all turtles in this row, by clicking on the arrow below the column, you turn right all the turtles in this column. The aim of this game is to direct all turtles vertically upward.

In Imagine Logo, object-oriented programming is applied. In a natural way, one can implement classes, objects and inheritance, and make event control objects. One can give new shapes to the turtles described in a drawing list. It is also easy to create animation. Fore example, in the *Colourful Puzzle* there is a class for squares with dots and each square is described in a drawing list. Students and teachers like to make such games. There are two more examples of games *Pentomino* (teacher’s project made by Anna) and *The Maze* (student’s project made by David). The first project was created during a workshop for primary school teachers, and the second one - during a distant learning course entitled “Programming in Logo”. David is very proud of creating this project and has his own secret connected with the way the board for the maze is remembered by a computer.

[3] Ibid., p. 52

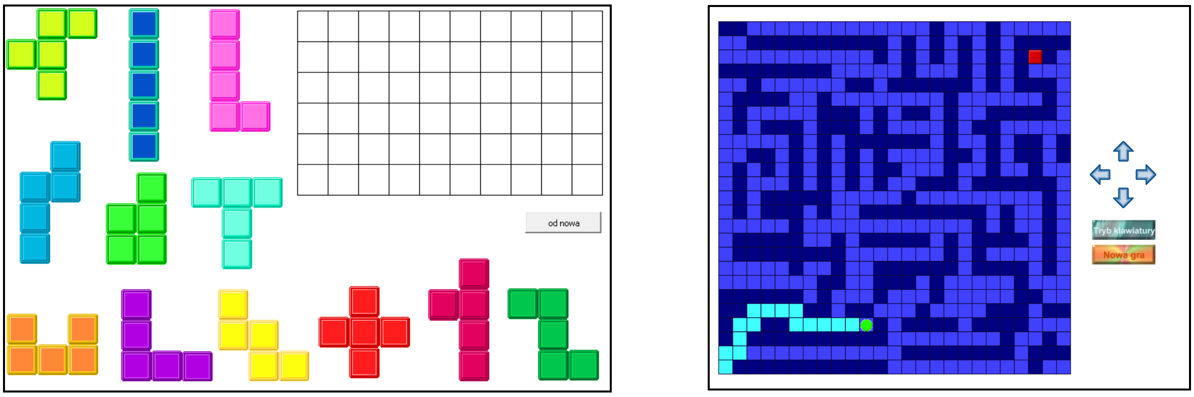


Figure 3. Pentomino and The Maze

Using Imagine Logo network features the implementation of a network game called *Treasure* was prepared. The goal of this game is to find three gold bars. There are two players and they have one board. The first person clicks on one of the squares. If a gold bar is there, the player gets a point and has another move, if not, the information about a distance to the nearest treasure is shown, and the second player has his turn. The game continues until all the gold bars are found. The one who has more gold bars is the winner. In this game you have to combine strategic issues with some psychological. When you try to find a treasure, you are also giving some information to the opponent where the treasure can be.

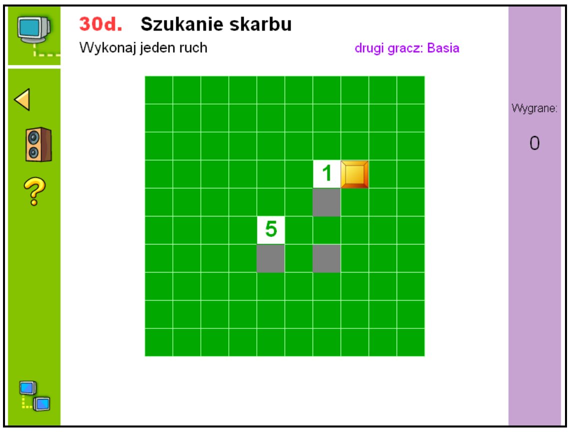


Figure 4. A network game - Treasure

We can ask after Seymour Papert “What will children learn by making a game? They will learn some technical things, for example to program computers. They will learn some knowledge traditionally incorporated in the school curriculum, for example in order to make shapes and program movements, they will have to think about geometry and about numbers. They will develop some psychological, social and moral kinds of thinking. Most important of all in my view is that children will develop their sense of self and of control. For instance, they will begin to learn what it’s like to control their own intellectual activity.”[4] We encourage teachers and students to work and play in Logo environment. It is a good instrument to create open exercises, stimulate thinking and make one’s own creative activity.

[4] Ibid., p. 47

Storytelling

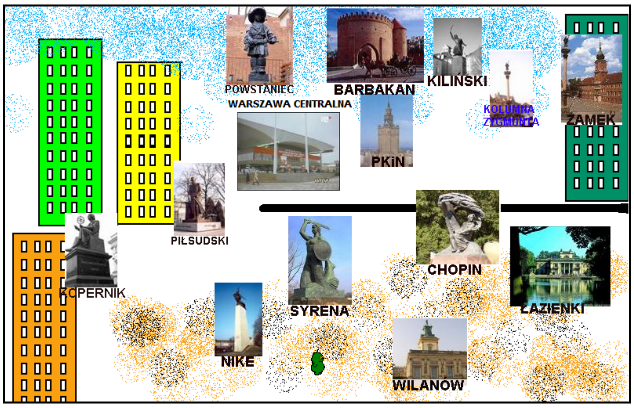
A very popular activity which teachers do with children is storytelling. During preparation of animate presentation students have to gain new knowledge about the world, they improve their understanding of human nature and feelings. Students also practise their language skills. They have to think what they want to say, choose a right medium – a written text, recorded voices, sounds, images or animations, and use their skills and imagination to prepare the entire story. What really counts when one has to create a good story is a concept and understanding of sequence. Students and teachers are really engaged in creating such projects. Though, it requires a major investment in time and effort.

"In my vision of this field its professionals will need special combinations of competences. Apart from a foundation in scientific knowledge and technological skill they will need high degrees of psychological sensitivity and 'artistic' imagination. For the ones who will make the greatest social contribution will be those who know how to mold the computer into forms which people will love to use and in ways which will lead them on to enrichment and enhancement...."[5]

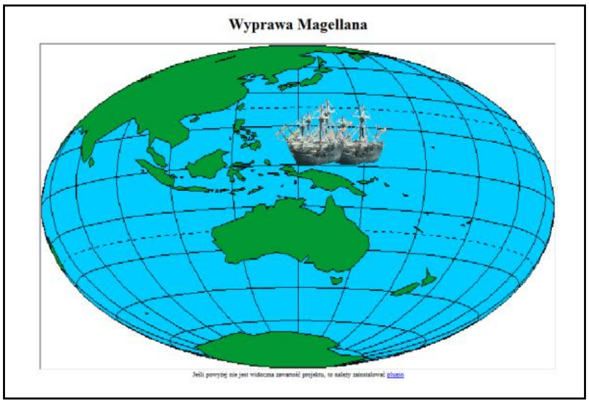
In terms of programming, one can start with very simple commands. As Seymour Papert suggests in *The Connected Family* the first step is to use HIDETEXT and SHOWTEXT commands, next you can name turtles and the work continues. If you are really engaged in making story, you will find that more programming skills are needed. Seymour Papert presents examples of operation on the picture that even a small child can do. They are as follows:

- make the picture disappear and reappear in response to a click on a colour or on a button or on an icon or on the picture itself,
- make the picture change size; for example shrinking it to stamp size and expanding it to full-screen (or any other) size,
- make the picture move.[6]

Below there are examples of the projects which were created by the teachers during our workshop.



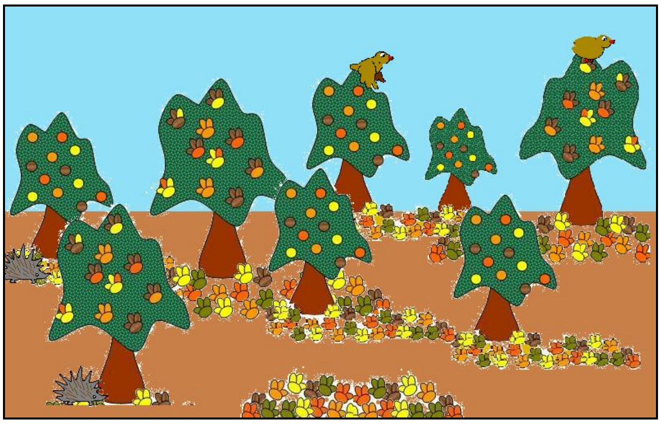
Walking in Warsaw - author Barbara



Ferdinand Magellan's Voyage - author Katarzyna

[5] After http://www.users.on.net/~billkerr/a/papert.htm, Solomon, Cynthia. Computer Environments for Children: A Reflection on Theories of Learning and Education. The MIT Press, 1987, after http://www.users.on.net/~billkerr/a/papert.htm

[6] Papert (1996), p. 133



Hedgehogs – author Małgorzata

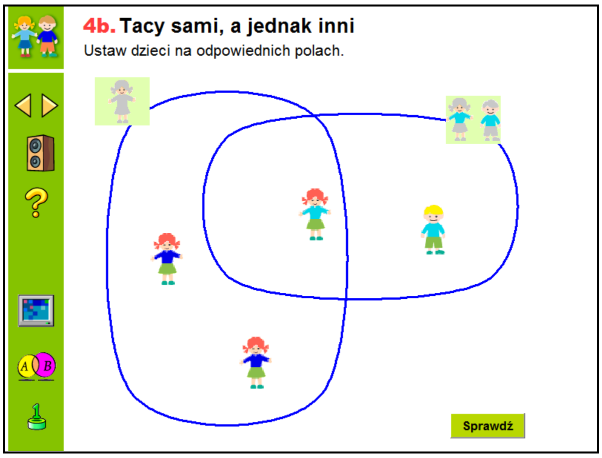
Existential problem - author Bożena

Figure 5. Teacher's story

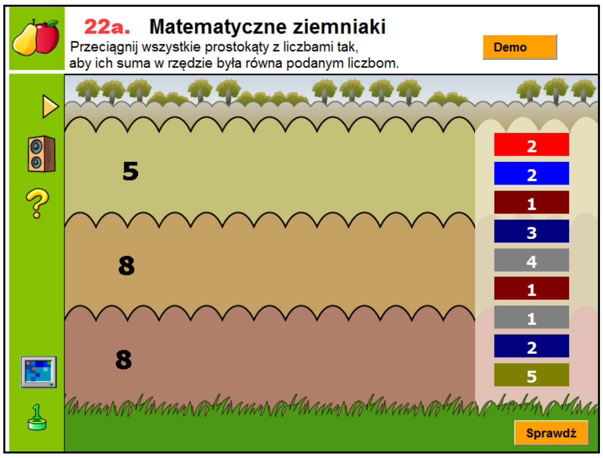
Making such animated stories leads to thinking about some moral issues and present them in an unconventional way. It is worthy to notice that storytelling and creating animations with descriptions and sounds are perfect ways to present some scientific ideas.

Powerful Ideas in Mind-Size Bites[7]

Working in microworlds it is like understanding scientific matters in a way we get familiar with another person. Children learn playing in microworlds and not just learning some facts by heart.

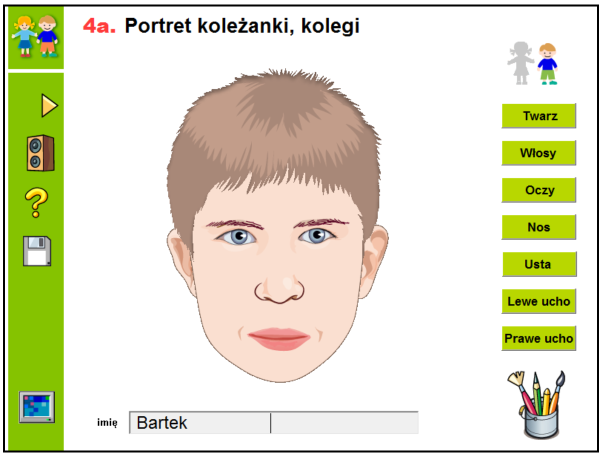


Sets

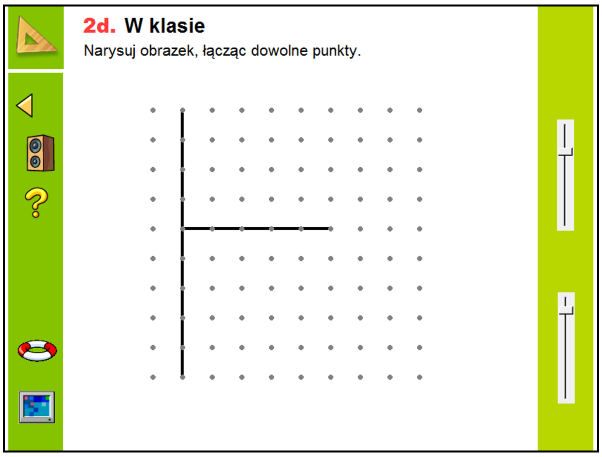


Mathematical potatoes

[7] This tittle comes from Seymour Papert "MINDSTORMS, Children, Computers, and Powerful Ideas"



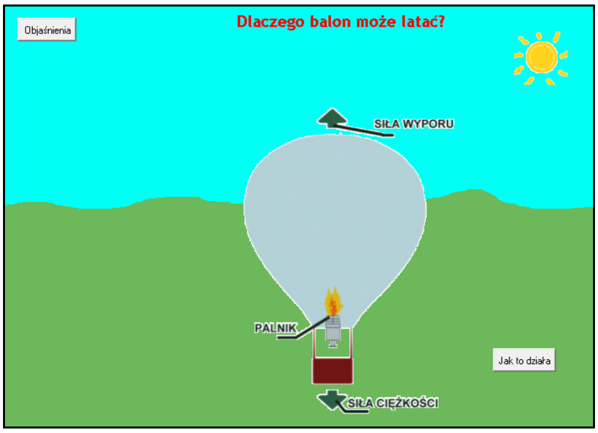
Friend's portrait



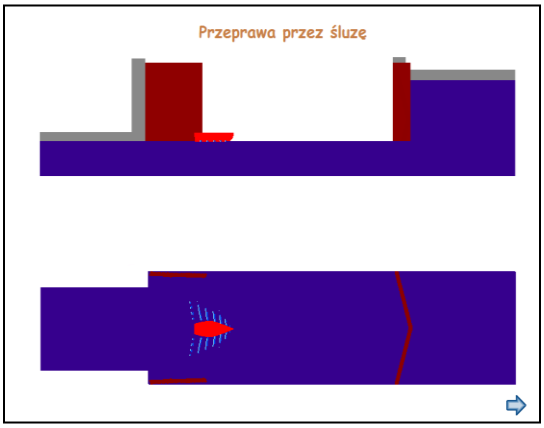
Dots

Figure 6. Examples of applications

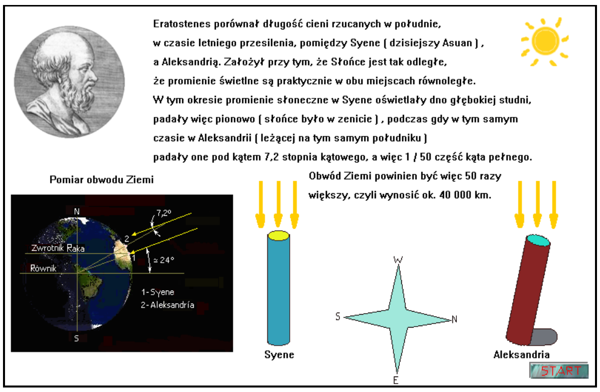
The learning process is deeper and more richly interconnected if playing in microworld is combined with creating. A creator needs to bring together programming skills and the projects' subject domain - like mathematics, computation, physics or other branch of science. Even a tiny part of making such project can give students a deep and broad learning experience. Above all, students are ready to put much effort and spend a lot of time to make their project interesting.



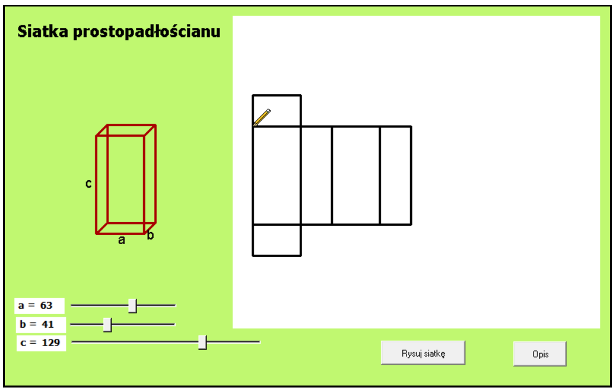
Why can balloons fly? Author Katarzyna



The Sluiceway author Sławomir



The Earth circumference – author Wiktor



Cuboids Nets – author Zofia

Figure 7. Teacher's applications

Making such projects one should define a model by identifying the major factors in a system and discerning rules that govern those factors. As a consequence, this process of hypothesizing, implementing and testing provides an excellent way to learning. It is a fertile ground for learning general thinking skills. These include problem decomposition, component composition, representation, abstraction, debugging, and thinking about thinking.

On the whole, multimedia projects presented in this article were created during workshops we organised for teachers and during courses for students. Computer Assisted Education and Information Technology Centre organised also the competition on creating multimedia projects. The competitors are primary and secondary school children. Every year a topic for the project is different. There were projects on environmental issues, on safety in the Internet, on the mystery of computers, a favourite book, on frogs during The Year of the Frog. Our teachers and students also took part in proposed activities during Eurologo 2007 in Bratislava (Logo images around us, Turtle's life, Logo in class, Imagine Logo Cup International Competition). We hope that by our effort we help children to learn by playing and to learn by programming. Last words belong to Seymour Papert "Across the world there is a passionate love affair between children and computers. I have worked with children and computers in Africa and Asia and America, in cities, in suburbs, on farms and in jungles. (…) Everywhere, with very few exceptions, I see the same gleam in their eyes, the same desire to appropriate this thing. And more than wanting it, they seem to know that in a deep way it already belongs to them. They know they can master it more easily and more naturally than their parents. They know they are computer generation."[8]

References

Kerr, B. (1991), *Papert's Ideas: Mainly From Mindstorms*, http://www.users.on.net/~billkerr/a/papert.htm

LOGO Web portal, http:/logo.oeiizk.waw.pl

Papert, S. (1980), *Mindstorms: Children, Computers, And Powerful Ideas*, Basic Books,

Papert, S. (1996), *The Connected Family: Bridging the Digital Generation Gap*, Longstreet Press

[8] Papert (1996), p. 1

Constructionism applied in early childhood mathematics education: Young children constructing shapes and meaning with sticks.

Chrystalla Papademetri, *C.Papademetri@euc.ac.cy*
Dept of Education Sciences, European University Cyprus

Abstract

Constructionism has been traditionally connected with computer-based research. This is probably why there is limited research within constructionism involving young children. Thus, we consider that the contribution of the study described in this paper relates mostly to the age of the subjects in conjunction with the tools employed. The aim of the study is to describe and analyze young children's understandings of shapes through an investigation of squares. A more focused purpose of the study is to investigate what knowledge young children have about the structure of simple shapes, how this knowledge is expressed, and how it is used in the process of constructing squares. The youngest child out of the 52 involved in the study was four years and ten months old and the oldest was six years and eight months old and for the construction process involved the tools employed were wooden sticks.

The consensus in existing literature is that children's limited, and often appearance-based descriptions of shapes, indicate that children view shapes as a whole and lack structural understanding. This study approaches children's understandings of shapes from a different perspective, based on an alternative and more dynamic interpretation of the van Hiele model and with the acknowledgement that there might be multiple ways of knowing and expressing mathematical knowledge. This study examines the understandings young children have about the structure of shapes. The methodology designed for the study is based on the constructionism idea of learning-by-making which leads to the need to search for tools that will act as objects-to-think-with and allow learners to communicate their thinking-in-change.

Fifty-two, five to six year olds, were engaged in three phase naturalistic task-based interviews. In Phase A (Description) the children were involved in classification and shape recognition activities. In Phase B (Construction) the children were asked to construct squares with the use of sticks and, in Phase C (Reflection) the children were asked to reflect on the construction process of Phase B. Even though during Phase A, the children, as supported by existing research, exhibited limited structural understanding about squares, through their involvement in Phase B, they exhibited much richer intuitive structural understandings. In Phase C, children tended to express structural understandings about squares in diverse and inventive ways.

The findings challenge the view that children's limited verbal descriptions of shapes indicate lack of structural understanding. In the process of the interviews the children articulated, through the 'language' provided, structural knowledge about squares that may be characterized as intuitive - if we share DiSessa's, definition of intuition (DiSessa 2000)- and at the same time they were able to situate their abstractions in the context of construction. Overall the findings indicate that, provided sufficiently sensitive techniques are employed, it is possible for children to express structural knowledge in diverse and often unconventional ways.

Keywords

Young children, shapes, construction, intuitive knowledge, non-verbal thinking

Introduction

Constructionism builds on the simple but powerful idea that with the appropriate tools learners can 'build things and ideas simultaneously' (Noss & Hoyles, 2006) and 'evokes the idea of learning-by-making' (Papert, 1991). This powerful idea that was added to the constructivist connotation that knowledge is actively constructed by the learner, has provided access to the construction of new meanings. Since, as supported by Noss and Hoyles (1996), it was the computer that allowed 'glimpses to new epistemologies' and 'opened new windows on the construction of meanings' it is not surprising that constructionism has traditionally been connected with computer-based research. This is probably why there is limited research within constructionism involving young children. The involvement in computer-based tasks requires a certain familiarity with the mean that might be considered as an agent which adds extra exertion that discourages researchers from investigating constructionism with younger children.

Thus, we consider that the contribution of the study described in this paper relates mostly to the age of the subjects in conjunction with the tools employed. The aim of the study is to describe and analyze young children's understandings of shapes through an investigation of squares. A more focused purpose of the study is to investigate what knowledge young children have about the structure of simple shapes, how this knowledge is expressed, and how it is used in the process of constructing squares. The youngest child out of the 52 involved in the study was four years and ten months old and the oldest was six years and eight months old and for the construction process involved the tools employed were wooden sticks.

Review of the literature

Young children's understandings of shapes

The consensus in existing literature (Burger, 1985; Burger & Shaughnessy, 1986; Shaughnessy & Burger, 1985; Clements et al, 1999; Clements & Sarama, 2000; Clements et al, 2001; Clements & Battista, 1992) which has formed the picture of what is mostly believed about young children's understandings of shapes is that children's limited and often appearance-based descriptions of shapes indicate that children view shapes as a whole and lack understanding of shape structure. It is interesting to search for the origins of this consensus. The attempt to discuss the origin of research concerning geometric thinking leads to the van Hieles.

Many years ago, when I found myself reading 'Structure and Insight' (van Hiele, 1986), at first I only managed to get to page 9 were van Hiele strongly claims that 'thinking without words is not thinking'. I rebelled; the van Hiele model was founded on a claim I could never accept. Because of my everyday experiences with young children I could never agree with van Hiele's claim. My experiences led me to the empirical realization that children are capable of, and understand so much more, than what they can express in words. My motive for getting involved in research on young children and shapes was exactly this realization which arose from the many opportunities I had of observing preschoolers constructing shapes when they 'had no knowledge' (if we measure knowledge by verbal language) of concepts like right angle, parallel lines etc. A realization which is highly supported by the consensus within early childhood education research that, 'the child has a hundred languages and a hundred hundred more. But they steal ninety-nine the school and the culture. To think without hands to do without head … (Loris Malaguzzi, poem translated by Lella Gardini, in Edwards et al, 1998). This poem came into my mind when I came across van Hiele's claim that 'thinking without words is not thinking'. The research culture which led to the consensus that children's limited and often appearance-based descriptions of shapes indicate lack of structural understanding, certainly fits the culture described by the poet and is highly connected implicitly or explicitly with van Hiele's claim that 'thinking without words is not thinking'. Within this research culture children were 'assessed' based on what they said.

According to van Hiele (1986) himself, at the first level of his proposed model of geometric thinking, children will simply say: 'This is a square', without any further explanation and without 'being able to mention even one property of' the shape (p.62). Van Hiele-based research has interpreted Level 0 as the level where children simply 'visually' recognise and describe shapes based on their appearance ('This is a square because it looks like a window'), see shapes as a whole, pay no attention to and have no understanding of shape properties. This interpretation of van Hiele's level 0 is widely accepted even though, if we carefully 'read' van Hiele's description of the level, he is not saying that children's inability to express in words excludes structural understanding. He is simply describing the nature of children's utterances. Similarly my experience in relation to children's attempts to construct shapes led me to the hypothesis that judging a shape by its appearance does not exclude structural understanding.

Even though my first reaction when I got to page 9 of 'Structure and Insight', was to close the book, I soon realised that I could not claim to be studying children's understandings of shapes without reading the whole thing. Almost every existing study on geometry refers to the van Hiele model of geometric thinking. I could, therefore, not ignore it and base my study solely on other researchers' interpretations. So I returned to the book and gradually realised that the theory unfolded by van Hiele had nothing to do with his opening claim that 'thinking without words is not thinking'. For instance van Hiele glorifies the importance of intuition in the process of thinking as he devotes a substantial part of 'Structure and Insight' to intuition which he defines as a way of 'seeing the solution to the problem directly, but without being able to tell' (p. 76). It is noteworthy to mention that van Hiele himself in a paper published in 1999 admitted that ……

> …… thinking without words is not thinking. In Structure and Insight (van Hiele, 1986), I expressed this point of view, and psychologists in the United States were not happy with it. They were right. If nonverbal thinking does not belong to real thinking, then even if we are awake, we do not think most of the time. Nonverbal thinking is of special importance; all rational thinking has its roots in nonverbal thinking, and many decisions are made with only that kind of thought. In my levels of geometric thinking the 'lowest' is the visual level, which begins with nonverbal thinking (p.311).

The fact is that van Hiele's positive stance towards non-verbal thinking was obvious in 'Structure and Insight' in spite of his negative claim regarding non-verbal thinking. 'Structure and Insight' was highly characterized by paradoxes which van Hiele-based research failed to detect. A thorough discussion of these paradoxes and an analysis on how they have influenced van Hiele-based research is provided in Papademetri (2007). Additionally, significant dynamic aspects of the van Hiele theory which were eliminated by van Hiele-based research are emphasized. This attempt to re-visit van Hiele in Papademetri (2007) led to the conclusion that van Hiele-based research failed to 'read' van Hiele, and as a consequence substantial pieces of research concerning geometric thinking are characterized by a tendency (a) to evaluate children mainly through their verbal ability, (b) to emphasize children's 'wrong responses', 'misconceptions'; this led to a downplaying of children's important, rich, intuitive understandings and (c) to evaluate the cognitive act with no reference to important aspects of the setting (e.g. child-adult interaction, activity) in which the cognitive act takes place; as an extension to this remark, young children were assessed in settings which were designed without taking into consideration the particularities of their age. A particular implication of these orientations is that they led to a restricted view of what children know about shapes. This restricted view has tended to degrade children's structural understandings.

In the next part of this paper I will describe how constructionism can provide alternative routes towards a more 'equitable' investigation of young children's understandings of shapes.

Construction, young children and shapes

As a starting point, this study values Hoyles (2001) insistence, on the conviction 'that studies in mathematics education should involve some discussion of mathematical activity' and that the knowledge constructed in children's heads is highly connected with the tools at hand. This conviction has its roots in the Vygotskian assumption that 'the activity in which knowledge is developed is an integral part of what is learned' and on the belief that 'treating knowledge as an integral, self-sufficient substance, theoretically independent of the situations in which it is learned and used' assumes 'a separation between knowing and doing', (Brown et al, 1989).

Noss & Hoyles (1996), within the framework of computer-based research emphasise the 'need to focus on tools and settings' as well as 'on the ways in which the understanding of mathematical ideas is mediated by the tools available for its expression' (p.50). Thus we have the constructionism principle that learning takes place in situations where learners are allowed to build and reflect on their own models (Kafai, 2006) which builds on the powerful idea of thinking-as-constructing which suggests that actual, physical construction can lead children to new understandings. 'Constructionism suggests that learners are particularly likely to make new ideas when they are actively engaged in making some kind of external artifact – be it a robot, a poem, a sand castle, or a computer programme – which they can reflect upon and share with others (Kafai & Resnick, 1996, p.1). Similarly, Noss & Hoyles (1996) refer to the importance of focusing on 'the ways in which the understanding of mathematical ideas is mediated by the tools available for its expression'. This leads to a search for 'objects-to-think-with', to borrow Papert's (1993) expression, but also tools that can be used by children as a language to express and communicate their 'thinking-in-change', to borrow an expression by Hoyles (2001). This study builds on the perspective that communicating is an integral part of thinking. Thus, in designing activities, there is a need for providing children with what Noss & Hoyles (1996) define as an 'autoexpressive' language; a language which acts both as a thinking tool and an expressive tool.

Even though Vygotsky (1962) states that 'to understand another's speech it is not sufficient to understand his words …we must understand his thought' (p.51), as we saw in the previous subchapter of this paper there is a tendency of restricting language to speech, to a verbal communication system. Within the computer age and Papert's revolutionary import of the programming language and the idea of 'thinking in images' (Papert, 1986), the discussion about language and thinking is elevated to a different level and context. Until before 'new technologies' opened new roads to thinking and understanding, children's use of words was analysed in order to unravel what children know or do not know, what children can or cannot do. But what the computer revolution showed was that language (the process of communicating and expressing) is not only about words. Words are not the only carriers of meaning and knowledge.

The study described in the remaining of this paper aims to add to this attempt to investigate the hypothesis that children might think in alternative ways, and challenge the idea that thinking depends on language (thinking in words). In addition, whereas existing studies which seem to have played an important role in formulating a picture of young children's understandings of shapes were build on an assumption of 'what thinking without words is not' the study described in this paper aspires to investigate 'what thinking without words is'. In addition it aspires to allow young children to 'look through' the same windows on the construction of mathematical meaning which were opened by constructionism. Thus, the effort was to design a study that would allow us to overcome the restrictions within most existing research as these were described in the previous section of this paper, in an attempt to better describe and analyse young children's understandings of shapes. To be more precise, the aspiration of the study was to investigate what knowledge young children have about the structure of simple shapes, how this knowledge is expressed and how it is used in the process of constructing shapes. The emphasis thus, was placed on the use of construction as a methodological tool.

Methodlology

In order to address the aim of the study, 52 children were engaged in task-based interviews, consisting of a three-phase framework (Description-Construction-Reflection). In Phase A-Description Task (DT), the children were involved in classification and shape recognition activities. This opening phase enabled subsequent data to be evaluated in comparison to those of existing research. In Phase B-Construction Task (CT), the children were given wooden sticks of various lengths and were asked to construct squares. This allowed the children to express their understandings of shapes in alternative ways. Finally, in Phase C-Reflection Task (RT), the children were involved in a process of reflecting on the construction process of Phase B.

Special attention was paid in relation to designing a research appropriate for the age of the subjects. During the research design, I had to constantly keep in mind the fact that this study was both a study within the domain of mathematics education as well as within that of early childhood education. As stressed out in literature on early childhood education research (Brooker, 2001; Donaldson, 1978; Grieve and Hughes, 1990; Tizard and Hughes, 1984; Wells, 1985) it is ‘commonsense’ knowledge (for people familiar with the nature of young children) that it is more likely to penetrate into children’s minds if you investigate them in a familiar environment, with familiar adults. That is why it is a striking experience for people with this kind of familiarity with young children to come face to face with research concerning young children, when such research ignores commonsense knowledge of how young children think and act. Therefore, in an effort to enable children to communicate in an authentic way, special arrangements were made that allowed naturalistic elements into the setting of the interviews.

Thus the interviews were conducted by a group of thirteen student teachers as part of a teacher training course they were attending at the University of Cyprus. The interviews were conducted in two public schools where the student teachers were ‘working’ as pre-service teachers. The student teachers attended a training program and were provided with a detailed interview scrip tool. The training program and the interview script tool were the products of an iterative piloting procedure completed in three cycles. The involvement of the student teachers allowed the inclusion of naturalistic elements in the research design and enabled the children interviewed to express their understandings while being involved in activities with familiar adults and in familiar settings. It also allowed for a far greater sample than would have been possible otherwise. All interviews were conducted in the children’s own school as part of their everyday involvement in free play activities where children are freely engaged in controlled activities. Given the many variations of play to which the children involved in the study were already accustomed within free play settings, the task-based interviews were considered by them as yet another usual activity.

All interviews were videotaped and transcribed. The coding scheme that was used for data analysis was developed in two stages. A preliminary coding scheme was developed with the use of the data collected from the piloting procedure. This was based on an initial open coding process in an effort to identify interesting phenomena and patterns among the data. This was then revised and advanced with the use of the data collected from the main study.

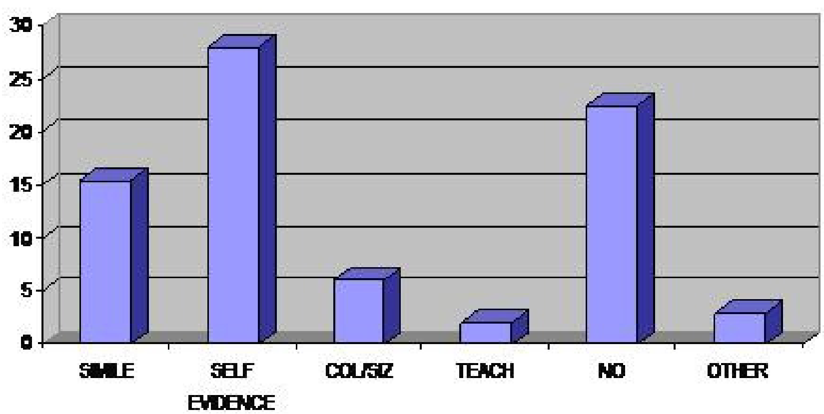
The findings of the study as presented in the following section of this paper aim to provide an insight on how the study’s participants used construction (a) to communicate rich, intuitive, structural understanding about squares and (b) as a tool-to think-with. In order to address these two issues a brief reference will be made to the findings of the DT and a more extensive reference will be made to the findings of the CT. In the CT the children had up to three attempts in order to construct a square. In this paper special attention will be given to the children’s first attempt to construct a square. Additionally we will refer to some of the children’s second attempt.

Findings

Children expressing their understandings through Description

Before presenting the findings in relation to how the children involved in the study used construction in order to communicate their understandings about squares, it is important to briefly describe the findings in relation to the children's involvement in the DT. One hundred and thirty-eight responses were collected from the fifty-two children that took part in the study. The information provided here provides a first sense of the 'quality' of the children's understandings as expressed in a setting restricted to classification, recognition and description tasks.

During the data analysis process it was considered essential to define a system which would allow the distinction between expression means which implicitly included structural elements ('reference' to the shape's structure) and other means of expression. It is interesting to note that within the setting of the DT, 77% of the children's responses did not implicitly include any structural elements. Graph 1 presents the findings in relation to the categories of responses which were categorized as non-structural answers. More than 25% of the non-structural responses included a self-evident justification ('because this is how a square is', 'because they have a square shape'). Besides the responses which included a self-evident justification, there was a significant number of responses which were categorised as 'NO' responses. In these cases, children would state that they didn't know anything about squares or they would simply not reply to the interviewer's question. There were 31 such responses all together (Graph 1, NO). Additionally, 16% of the children's responses in this study included a simile (Graph 1, SIM). The children would say that squares are like 'a house', 'a carpet', 'windows', 'the underneath of a house'.



All values are given in percentage rates.

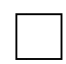
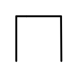
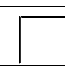
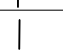
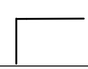
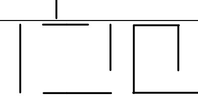
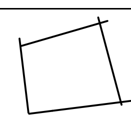
Graph 1: Results in relation to the sub-categories of responses which were not classified as non-structural

If this study was to follow the same methodological framework as existing studies on young children and shapes it could reconfirm the existence of van Hiele level 0. Shaughnessy & Burger (1985) claim that when younger students where asked what they would tell a friend so that s/he could pick up all squares from a sheet of paper they would answer 'I'd tell them to pick out all the squares' or 'look for the doors'. Consequently, Shaughnessy & Burger (1985) reconfirmed the existence of van Hiele level 0 as the level where descriptions are purely visual and no attention is given to shape properties. The difference between this study's methodology and the methodology followed by Shaughnessy & Burger (1985) lies in the fact that here the aim is to describe the ways children express their understandings of shapes within and in correlation to a specific setting and not to evaluate children in order to place them in levels independently of the setting in which they express their understandings. In conclusion, one can argue that within a setting restricted to simple classification, recognition and description tasks, children exhibited poor structural understandings of shapes. We will now describe children's understandings of shapes as these were expressed through their attempt to construct a square.

Children expressing their understandings through Construction

In my effort to identify an analysis tool for analysing the data collected from the CT, I faced a practical problem. There were some commonalities in relation to some actions in the children's attempt to construct a square, but the route each child followed was unique. The case of two or more children following exactly the same process was rare and thus, categorizing fifty-two children into 20-25 categories was not considered an effective way to categorise the data. So what could be the criterion for a meaningful categorisation? To answer this question, a very careful and repetitive examination of the data collected was considered essential. What became apparent from carefully studying the raw and transcribed data was that children would base their construction on a specific choice, **a foundational action** that had the attribute of stability. This specific action, which involved the choice of specific sticks and/or their spatial arrangement, was an action that remained intact until the end of their attempt and exhibited understanding of a specific property (or properties) of a square. This foundational action was the children's first action in their attempt to construct a square. The children would then proceed with other choices and actions in order to complete their construction. All of these other choices and actions during the construction attempt involved the element of experimentation and thus indicated that children were in a constructing process of building new knowledge on their original intuition. The identification of the children's foundational actions in the CT allowed the identification of nine square construction strategies among the fifty-two children that participated in the study. These nine categories are described in Table 1.

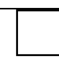
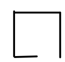
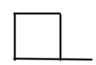
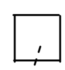
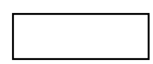
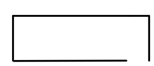
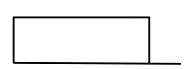
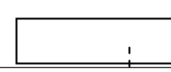
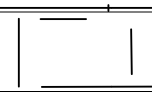
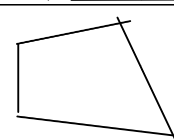
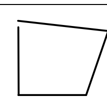
Table 1 Description of the strategies the children followed during their attempt to construct a square

Strategies	**Code**	**Foundational Action**		
		Verbal Description. The child ……	**Choice of Sticks**	**Spatial Arrangement**
1	**S1**	…… selects four equal sticks and places them one by one creating right angles and thus constructing a square.	4 equal sticks	
2	**S2**	…… selects three equal sticks and constructs an open shape with right angles.	3 equal sticks	
3	**S3**	…… selects two equal sticks and constructs a right angle.	2 equal sticks	
4	**S4**	…… selects two equal sticks and places them parallel and aligned.	2 equal sticks	
5	**S5**	…… randomly selects two (unequal) sticks and creates a right angle.	2 unequal sticks	
6	**S6**	…… randomly selects four (unequal) sticks and tries to construct a four-sided shape with four right angles.	4 unequal sticks	
7	**S7**	…… selects three equal sticks.	3 equal sticks	-
8	**S8**	…… selects two equal sticks.	2 equal sticks	-
9	**S9**	…… selects one stick at a time and tries to construct an irregular quadrilateral which somehow looks like a square with its sides not equal and its angles not right.	4 unequal sticks	

Besides the patterns identified within the data which allowed the identification of the nine stategies described in Table 1, patterns and commonalities were also identified in relation to the final product of the children's first attempt to construct a square. Eleven categories were identified in relation to this aspect. The products of the children's attempt to construct a square are described in Table 2 and are classified into three types (Table 2, Type A, B, C).

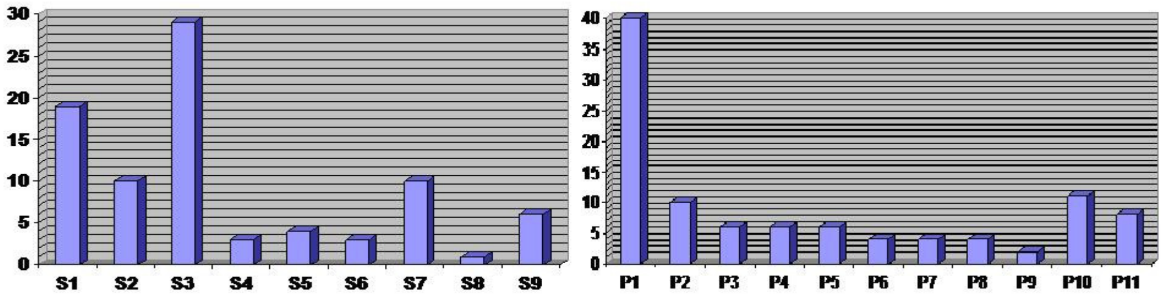
First, let us take a look at which Strategies (S) the children followed and what Products (P) they ended up with in their first attempt to construct a square. Graph 2 shows the percentage of children that followed each of the S identified and described in Table 1 and the percentage of children which ended up with each of the P as these were described previously in Table 2. A first significant observation that can be made based on Graph 2 is that the percentage of children that followed S1, S2 and S3 is much higher than that of the children that followed other strategies. As we can see in Graph 2, the most frequently used strategy among the 52 children was S3. 29% of the subjects used this strategy, in other words began their construction by selecting two equal sticks and constructing a right angle. A significant percentage of children (19%) followed S1 (selected four equal sticks and constructed a square with no experimentation required). The rest of the strategies were followed by smaller groups of children.

Table 2 Description of the products of the children's attempt to construct a square

Type	Product	Code	Verbal Description	Graphical Description
A	1	**P1**	Square with four equal sticks.	
	2	**P2**	Square with four sticks (gap).	
	3	**P3**	Square with four sticks (extension).	
	4	**P4**	Square with more than four sticks.	
B	5	**P5**	Rectangle with two sets of equal sticks.	
	6	**P6**	Rectangle with four sticks (gap(s)).	
	7	**P7**	Rectangle with four sticks (extension).	
	8	**P8**	Rectangle with more than four sticks.	
	9	**P9**	Rectangle with four sticks (gaps and extensions).	
C	10	**P10**	Irregular quadrilateral that resembles a square but has no right angles.	
	11	**P11**	Irregular quadrilateral with some angles right and/or some sides equal.	

The slightly oblique dotted line crossing through the side of a construction indicates that that specific side is constructed with the use of two sticks.

As far as the products of the children's attempt to construct a square, one first key observation is that there is a big difference between the findings in relation to P1 compared to other products. It is quite astonishing that 40% of the children involved in the study (21/52) successfully constructed a square by using four equal sticks. The rest of the products were the result of the effort of much smaller groups of children.



Graph 2 Results in relation to the Strategies (S) followed by the children and the Products (P) of their attempt to construct a square

In Table 2, the eleven products identified among the data were categorised into three groups: Type A, B and C. What is important to highlight in relation to Graph 2 is that the majority of the children (62%) constructed a Type A shape (shape with four equal sticks/gap/extension/more than four sticks). Thus, whereas in the DT only 35% (18/52) of the children gave structural responses, 62% of the children involved in the study (32/52) constructed a Type A product in their first attempt to construct a square.

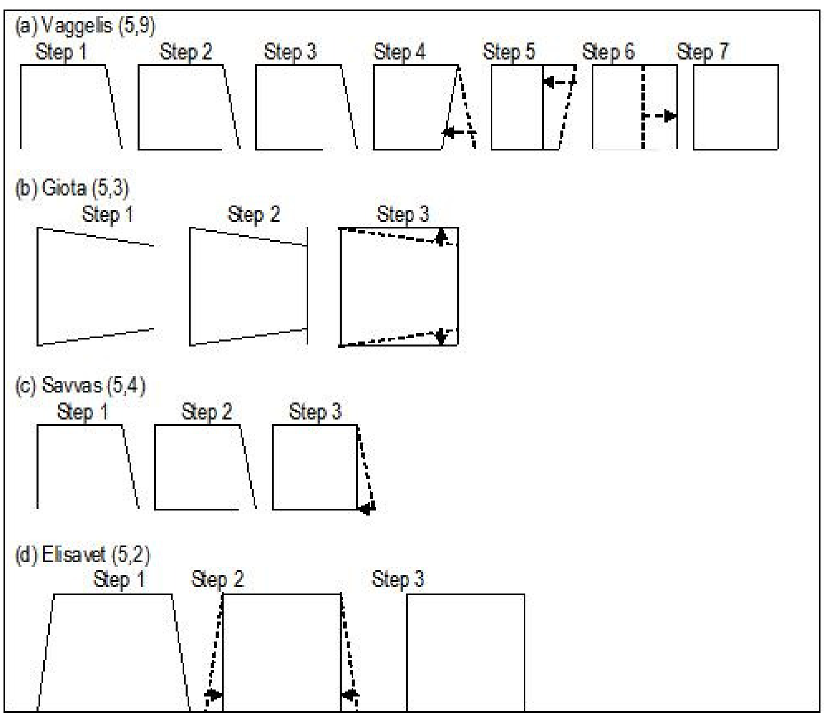


Figure 1 The construction routes of children who followed S7

During the process of data analysis (Chapter 5), two important observations were documented. First of all, that children following the same strategy did not necessarily end up with the same product and second, children that did follow the same strategy and end up with the same product did not necessarily follow the same route along the way. These observations reflect the variability that existed among the data. This variability is evident in the example routes in Figure 1. In Figure 1, we have an illustration of the construction routes followed by children which began their attempt to construct a square with S7. Even though all of these children followed the same strategy and ended up with the same product, followed different routes.

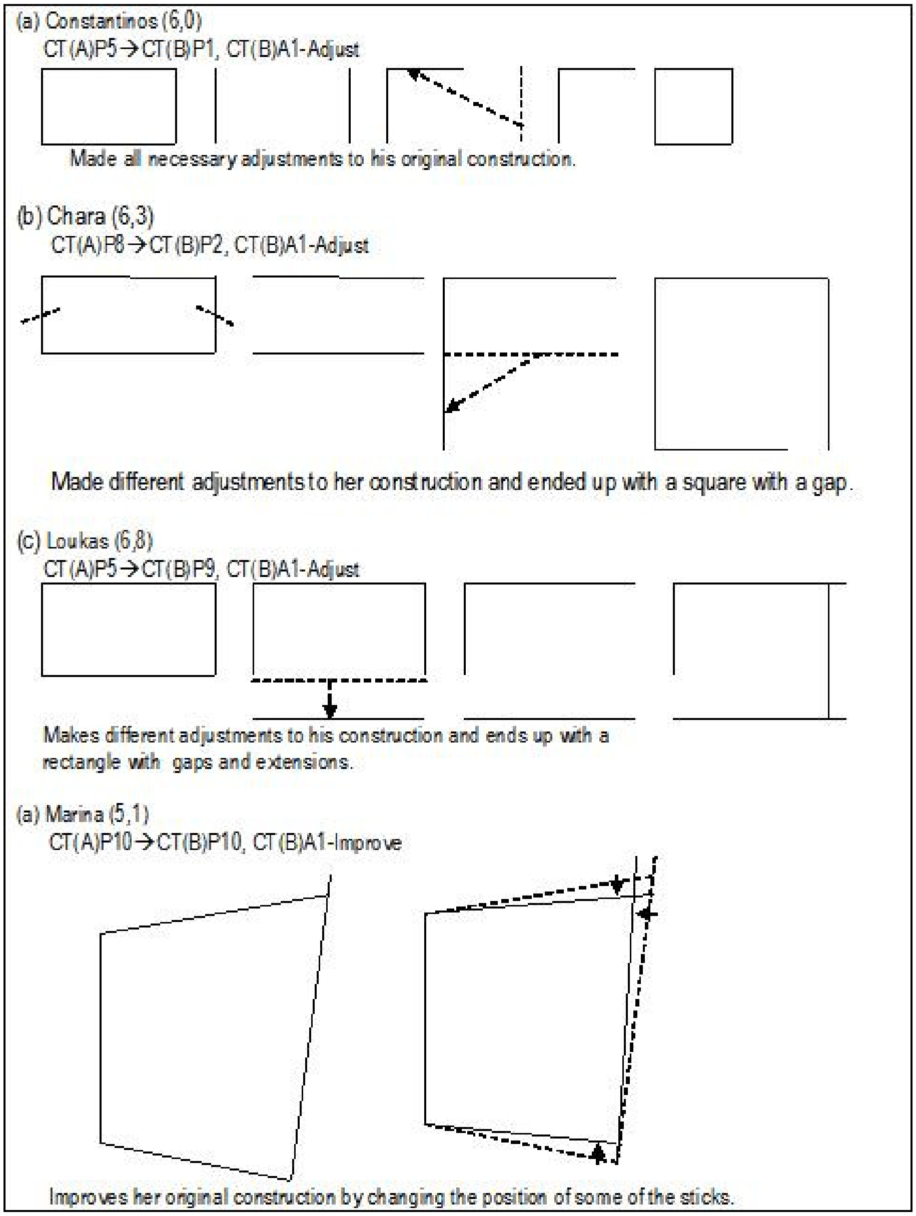


Figure 2 Some examples of the routes the children followed in their second attempt to construct a square

In the remaining of this section of the paper we will describe the routes followed by some of the children in their second attempt to construct a square (Figure 2) in an attempt to address the issue of how the children used the construction as an object-to-think-with and a tool for communicating their thinking-in-change. We had the cases of children like Loukas, for example. Loukas had constructed a rectangle in his first attempt to construct a square. After being encouraged by the interviewer to try again, he followed the route illustrated in Figure 2c. He ended up with a shape that looked more like a square than his original construction in the sense that the distance between the two vertical parallel sticks was more similar to the distance between the two horizontal parallel sticks. But the distance was still not equal. In many cases the children's actions allowed me to think of squares in ways I have never thought and look at square properties from a different perspective. Through his actions, Loukas gave a new perspective in relation to the square properties. Another way of expressing the equality of the sides is that the distance between the two sets of parallel sides has to be equal.

So Loukas acquired a vague intuition in relation to the distance between the sets of parallel lines contrary to other children such as Costantinos and Chara (Figure 2a,2b) who acquired a more clear intuition of the fact that it is not only the parallel sides that have to be equal but the adjacent sides as well. Similar to Loukas' was the case of Marina (Figure 2d). In an effort to make her original construction (quadrilateral with no right angles) look more like a square, Marina tried to close the angles. But the angles of her second attempt were still not right angles.

Construction versus Description

At first sight of the findings, one can argue that the children based their attempt to construct a square on specific structural understandings. These structural understandings unfolded through the children's strategies and products. Through the course of the children's attempts one can sketch the ways in which the children's understandings evolved and changed. The picture sketched in relation to the children's understandings as these were expressed through construction is rather different to the equivalent picture sketched from the children's involvement in the DT. Thus, at this point, we can support the point of view that even though within a setting restricted to description the children exhibited poor structural understandings about squares, through their involvement in the CT they exhibited rich structural understandings.

Discussion

As I claimed at the beginning, the aim of this study was to add to an attempt to investigate the hypothesis that children might think in alternative ways, and challenge the idea that thinking depends on words. In addition, the study described in this paper aspires to investigate 'what thinking without words is'. This last remark is an issue I would like to address in this last part of the paper in light of the findings as these were described in the previous section.

In order to address the question of what children know about squares first we need to address another important question. What should be used as an indicator of what children know: the strategy they followed, the whole route or the product of their attempt? One cannot ignore the fact that there was no linear connection between a strategy and a product. For example: the children that exhibited an understanding of the fact that a square has four equal sides and four right angles through the strategy they followed did not necessarily end up with a ('perfect') square at the end of their attempt. Should the children's 'failure' to construct a square at the end of their attempt erase the structural understandings they exhibited through the strategy they followed at the beginning of their attempt? Like in the case of Christoforos (Figure 3). Should his failure to construct a shape with for equal sticks erase the fact that he used three equal sticks to begin his construction? In addition, some of the children that exhibited an understanding of the fact that a square has four equal sides and four right angles at the beginning of their attempt and ended up with a square at the end of this attempt had to experiment. Again, does this experimentation imply that we should ignore the understandings these children exhibited at first?

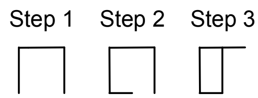


Figure 3 Christoforos (5,4) first attempt to construct a square

So here we are, as it shows, faced with a dilemma. Where should we focus on in order to determine children's existing knowledge: On their strategies, which included some 'correct' choices, on their need to experiment or on their sometimes 'faulty' products? Is this really a dilemma that needs to be resolved or is it a finding in itself? During the attempt to construct a square the children did not think in conventional ways. For example to think that '*a square has four equal sides and four right angles, thus I will select four equal sticks, place them in such a way as to construct right angles and thus construct a square'* in words, is a way of thinking in the conventional sense. It is a widely acceptable and recognisable way of thinking. If there was evidence among this study's data that children did think in such ways no one could deny that these children were thinking and it would be easy to identify exactly what they knew. But it is clear from the data that, in most cases, the children did not think in such ways. It is also clear that the children 'knew' specific aspects of a square's structure.

The question is what the nature of this knowledge is. The knowledge the subjects exhibited reminds us a lot of the way diSessa (1988, 2000) defines intuition. According to diSessa intuition constitutes 'little' pieces of knowledge, lack of systematicity and commitment, is very difficult to express into words, is 'rich', 'flexible' and 'diverse', and thus 'generative', is unstable, frequently effective and sometimes even correct. In light of diSessa's definition, it is quite safe to describe Christoforos understandings (Figure 3) as intuitive. In light of the study's findings we can support the point of view that through their involvement in the CT, the children exhibited a rich intuitive structural understanding of squares, and define intuition as the fragmented knowledge which children bring with them in a learning situation and are mostly able of expressing in ways which are contrary to what is formally acknowledged as correct.

Overall the findings indicate that, provided sufficiently sensitive techniques and 'languages' are employed, it is possible for children to express rich, intuitive structural knowledge in diverse ways. The search for such languages has been a main focus of computer-based research. But in this study, such a language was identified without the use of computers. Construction, the simple use of wooden sticks, became the language which the children could 'speak' and the adults could 'hear'.

References

Brooker, L. (2001). *Interviewing Children.* In G. Mac Naughton, S. A. Rolfe & I. Siraj-Blatchford (Eds.), Doing Early Childhood Research: International Perspectives on Theory and Practice (162-177). Buckingham: Open University Press.

Brown, J. S., Collins, A. & Duguid, P. (1989). *Situated Cognition and the Culture of Learning.* Educational Researcher, 18 (1), 32-42.

Burger, F. W. (1985). *Geometry.* Arithmetic Teacher, February, pp. 52-56.

Burger, W. F. & Shaughnessy, M. J. (1986). *Characterizing the van Hiele Levels of Development in Geometry.* Journal of Research in Mathematics Education, 17(1), 31-48.

Clements, D. H. & Battista, M. T. (1992). *Geometry and Spatial Reasoning.* In A.D. Grouws (Ed.), Handbook of Research on Mathematics Teaching and Learning (pp 420-464). New York: MacMillan Publishing Company.

Clements, D. H. & Sarama, J. (2000). Young *Children's Ideas about Geometric Shapes.* Teaching Children Mathematics, April 2000, 482-488.

Clements, D. H. Sarama, J. & Wilson, D. C. (2001). *Composition of Geometric Figures.* Proceedings of the 25th Conference of the International Group for the Psychology of Mathematics Education, Utrecht, 2, 273-280.

Clements, D. H., Swaminathan, S., Zeithler-Hannibal, M. & Sarama, J. (1999). *Young Children's Concept of Shape.* Journal of Research in Mathematics Education, 30(2), 192-212.

DiSessa, A. (1988). *Knowledge in Pieces.* In G. Forman & P. Pufall (Eds.), Constructivism in the Computer Age (49-70). Hillsdale, New Jersey: Lawrence Erlbaum Associates.

DiSessa, A. (2000). *Changing Minds.* USA: The MIT Press.

Donaldson, M. (1978). *Children's Minds.* Great Britain: Fontana Press.

Edwards, C. P., Gardini, L. & Forman, G. E. (1998). *The Hundred Languages of Children: The Reggio Emilia Approach.* Greenwich: Ablex Publishing Corporation.

Grieve, R & Hughes M. (1990). *An Introduction to Understanding Children.* In R. Grieve & M. Hughes (Eds.), Understanding Children (1-10). Oxford: Blackwell.

Hoyles, C. (2001). *From Describing to Designing Mathematical Activity: The Next Step in Developing Social Approach to Research in Mathematics Education?* Educational Studies in Mathematics, 46 (1-3), 273-286.

Kafai, Y. B. (2006). *Constructionism.* In K.R. Sawyer (Ed). Cambridge Handbook of the Learning Sciences. Cambridge:CUP.

Kafai, Y. B. & Resnick, M. (1996). *Introduction.* In Y. B. Kafai & M. Resnick (Eds.), Constructionism in Practice: Designing, Thinking and Learning in a Digital World (1-8). New Jersey: Lawrence Erlbaum Associates.

Noss, R. & Hoyles, C. (1996). *Windows on Mathematical Meanings.* Netherlands: Kluwer Academic Publishers.

Noss, R. & Hoyles, C. (2006). *Exploring Mathematics through Construction and Collaboration.* In K.R. Sawyer (Ed). Cambridge Handbook of the Learning Sciences. Cambridge:CUP.

Papademetri, C. (2007). *Structural Knowledge about Shapes: A Case Study of Young Children Describing, Constructing and Reflecting on Squares*. PhD Thesis, University of London, Institute of Education, London, U.K.

Papert, S. (1986). *Beyond the Cognitive: The other face of Mathematics.* Proceedings of the 10th Conference of the International Group for the Psychology of Mathematics Education, London, 1, 51-58.

Papert, S. (1991). *Situating Constructionism.* In I. Harel & S. Papert (Eds.), Constructionism (1-12). Norwood: Ablex Publishing Corporation.

Papert, S. (1993). *Mindstorms: Children, Mathematics and Powerful Ideas* (Second Edition). New York: Basic Books.

Shaughnessy, J. M. & Burger, F. W. (1985). *Spadework Prior to Deduction in Geometry.* Mathematics Teacher, September 419-428.

Tizard, B. & Hughes, M. (1984). *Young Children Learning.* London: Fontana.

Van Hiele, P. (1986). *Structure and Insight.* United Kingdom: Academic Press, INC.

Van Hiele, P. (1999). *Developing Geometric Thinking through Activities that Begin with Play.* Teaching Children Mathematics, 310-316.

Vygotsky, L. S. (1962). *Thought and Language* (E. Hafmann & G. Vakar). Cambridge: The M.I.T. Press. (Original work published 1939).

Wells, G. (1985). *The Meaning Makers.* London: Hodder & Stoughton.

Hyperbolas and chimneys in classroom

Alejandro Rosas, *alerosas@ipn.mx*
Program of Mathematics Education, CICATA-IPN, Mexico

Leticia del Rocío Pardo, *rociopardo2000@yahoo.com.mx*
Special Education Dept, Secretary of Education of Veracruz, Mexico

Abstract

In a second course of Mathematics, Engineering students must learn about integral calculus and its applications. Usually teachers show formulas and examples from the text book, unfortunately those examples have no meaning at all for students. So we have tried to design some didactic activities that let students to build objects with a specific form.

Figure 1. Model of a hyperbolic chimney (used in nuclear power plants) made by a student.

We asked students to build a chimney for a nuclear power plant. They did not have any restriction on the kind of materials they would use only for the form. Students used paper, wires, plastic, aluminium foil, plasticine and other materials to model the shape of a hyperboloid of one sheet. They had to calculate the equation by taking measures from a picture we gave them. In this work we show photos and videos of students' chimneys based on the equations they did.

Keywords

Solid of revolution, volume, area of surface of revolution, hyperbola, integral.

Introduction

In many universities Engineering Programs cover mathematics courses that include calculus in one real variable. Most times it is divided in two courses, the first for differential calculus and the second course that covers integral calculus. In the Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM) all engineering students cover three courses: Mathematics I - Ma1011 (Differential Calculus), the second course is named Mathematics II – Ma1012 (Integral Calculus) and the third is Mathematics III – Ma1013 (Multivariable calculus).

Our main goal is to involve students in the use of mathematics not just solve routine problems but to apply mathematics in dairy life and in an Engineering context. Teachers are encouraged to use different didactic techniques like Project Based Learning (PjBL), Problem Based Learning and Case Study Methodology.

Theoretical Framework

Our didactic activity was designed under the constructionism theory and following the PjBL technique. Papert (1996) said “Don’t worry if the questions are trivial and repetitive” (p. 38) talking about software and parents, we can talk about routine problems in a classroom and students *don’t worry if exercises are boring and repetitive if they are mathematics*.

In Martin (1996) we can read

> According to constructionist learning theory, people learn most effectively when they are involved in the creation of an external artifact in the world. This artifact becomes an “object to think with,” which is used by the learner to explore and embody ideas related to the topic of inquiry. (pp. 297-298)

Following constructionism we decided our students should build something to relate school with every day life, based on integral calculus and other mathematics courses.

In Project Based Learning web page we can find a precise definition:

> “PjBL is an instructional approach built upon authentic learning activities that engage student interest and motivation. These activities are designed to answer a question or solve a problem and generally reflect the types of learning and work people do in the everyday world outside the classroom” (Buck Institute for Education, n.d.)

PjBL characteristics can be summarized in a short list as

- Lerner-centered environment
- Collaboration
- Curricular content
- Authentic tasks
- Multiple expression modes
- Emphasis on time management
- Innovative assessment

Following PjBL technique we decided that our students should apply analytic geometry and integral calculus to adjust real data from a chimney in Spain and to find the equation of an hyperbola.

Joining both approaches we can say that our design involves the construction of a chimney (using s scale 1cm:10m) by using analytic geometry and integral calculus, letting students to freely choose any mathematical method or technique and any material for the model they need to build.

Activity

Justification

The main idea of this activity is to involve students in the use of "school mathematics" in an engineering context. Students usually learn about circumferences, parabolas, ellipses and hyperbolas and solve exercises like "Find out the center and radius of a circumference that…" or "Given the equation of a hyperbola find out the center, vertices…" and sometimes teachers solve an example of analytic geometry applied to engineering.

In ITESM we try to solve as many real world problems as possible, so the Chimney Activity is an open problem where students must build a hyperbolic chimney following the equation they find based on a photograph. We thought students would feel motivated to search, in internet and mathematics books, for different methods to solve the problem. The activity was used in two different groups with a total of 63 students in an online format; they had two weeks to finish all their work. We wanted the students to globalize their knowledge, that is, students should apply all the mathematical knowledge they have in order to solve the activity.

Activity in detail

We show students different kinds of chimneys and we discuss the many advantages and disadvantages that every kind of chimney has. In the case of a hyperbolic chimney we present the following diagram and photograph where some explanations are given.

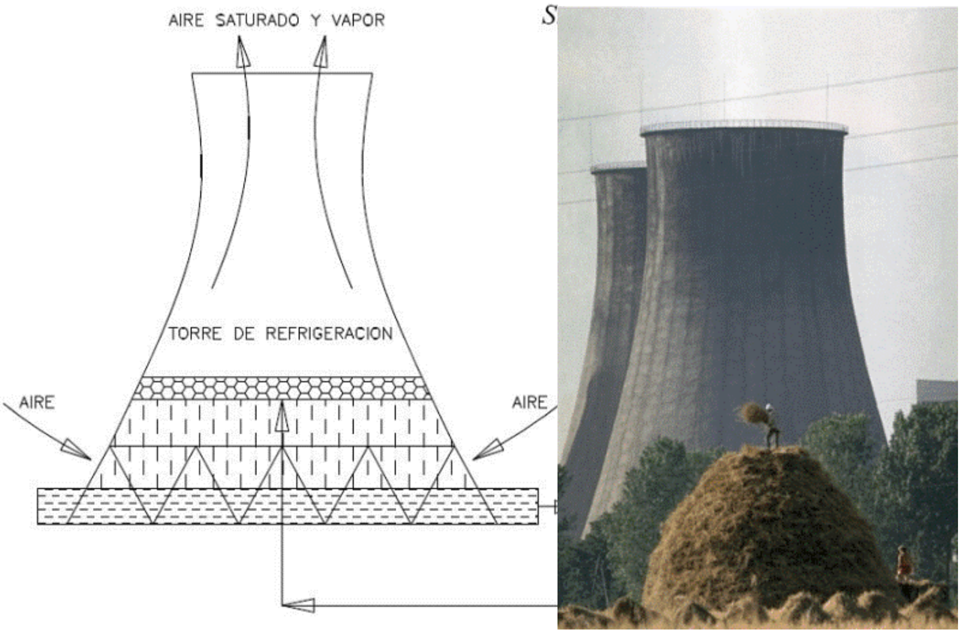


Figure 2. Diagram and actual photograph of a cooling system using a hyperbolic chimney.

Then we gave students the following schema with some measures of a real chimney:

Total height 119m (390.5ft),

Base width 108m (354.3ft),

Top width 75m (246ft),

Mid point width 65m (213.2ft),

Mid point height 88m (288.7ft)

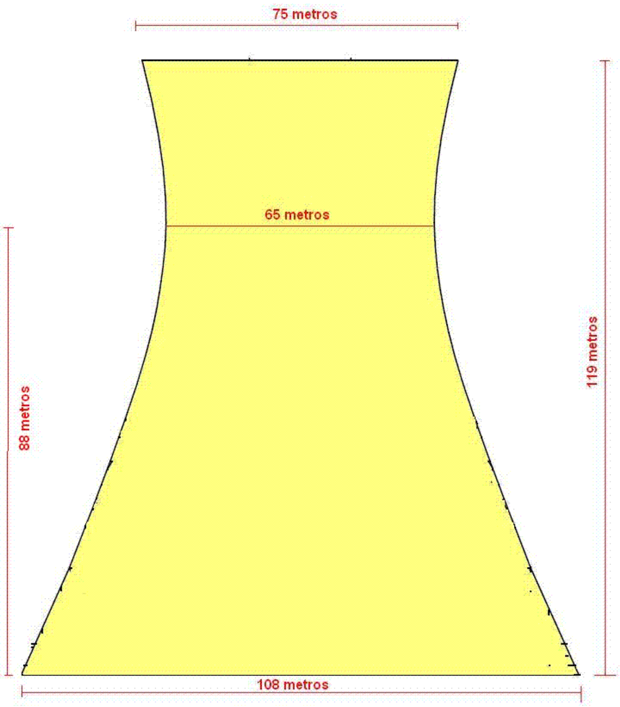


Figure 3. Measures of the chimney in metres.

Then students had the following tasks:

1_ Find out an equation for the hyperbola that generates the chimney.

2_ Find out the volume of the solid of revolution generated by the equation of (1)

3_ Find out the area of the surface of revolution generated by the equation of (1)

4_ Build a scaled model of the chimney using any material you want to use.

Students Answers

We analyzed the answers in two parts, we called Mathematical Part to the first one and deals with the use of mathematics to adjust chimney data with a hyperbola equation, the second one deals with the materials and technique students used to build the model, we called this one the Model Part.

Mathematical part

Some students applied analytic geometry, they drew Cartesian axes system with origin at the center of the hyperbola in figure 3, and then they calculated three or more points to get a system of equations involving the coefficients of the desired hyperbola equation. Solving the equations they were able to give the desired equation. One of these solutions can be seen in figure 4.

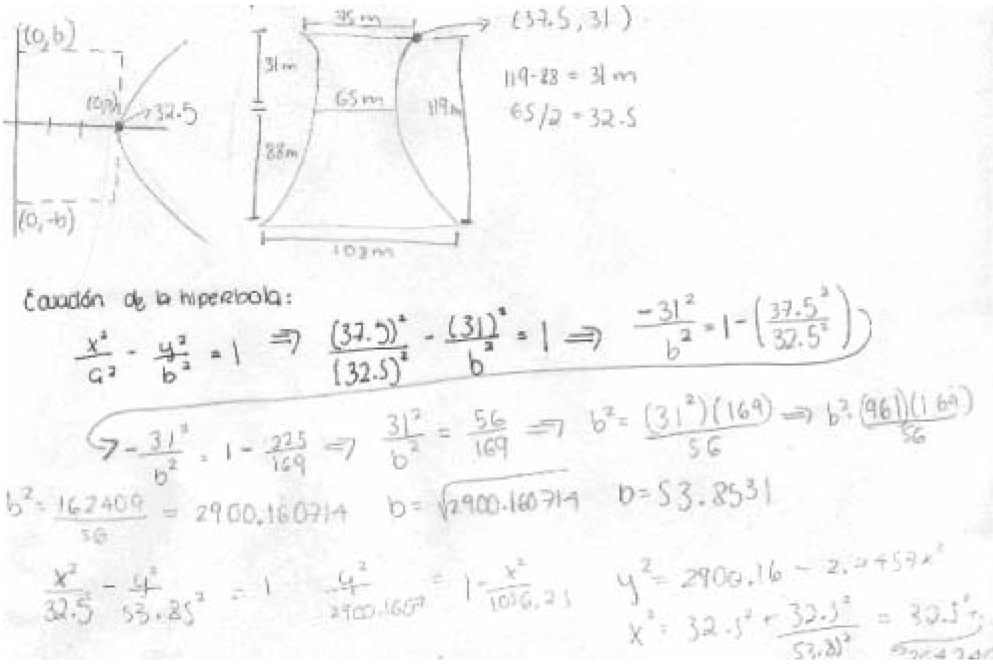


Figure 4. Student's procedure to find the hyperbola equation.

One student found his equation but he tested it and he discovered that the chimney should have to be built using two different hyperbolas. Then he calculated one hyperbola for the upper half and other equation for the lower half of the chimney. This was an unexpected result. Some students drew a vertical line in the middle of the schema in figure 3; they measured it and calculated many points. Then they used specific mathematical software to fix data with a polynomial function.

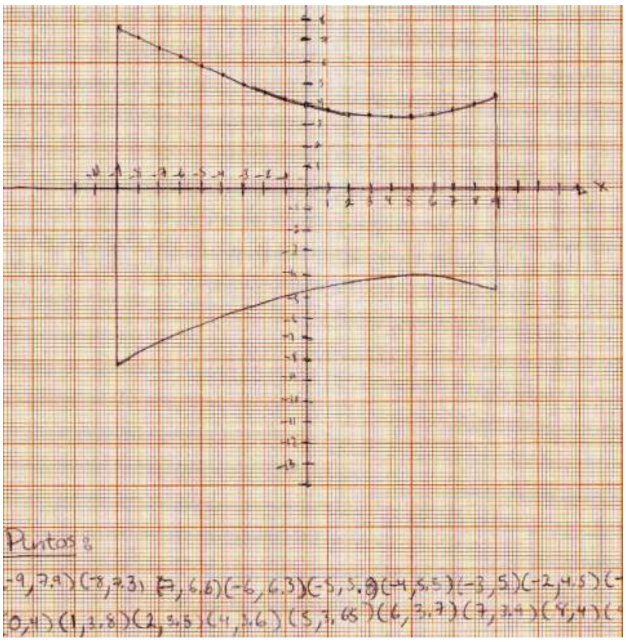


Figure 5. A student's solution included to measure many distances and use them in Mathematica® to fix data.

Every student applied the standard formulas to calculate the volume of the solid of revolution and the surface's area of the solid of revolution with integral calculus. Although they got different equations they got similar results for the volume and the surface's area.

The model

We found two important facts about the model construction, one deals with the material used in the model and the other deals with the shape of the hyperbola. Some students used paper and glue, some used wood, some used wire and paper and other students used a kind of plasticine to build the body of the chimney. Every kind of material had its difficulties.

To get the specific shape of a hyperbola was harder but students found many way. A few students cut cardboard rings with different radius according to the hyperboloid generated by rotating the hyperbola's equation then they covered the rings with plasticine (first chimney in figure 6). Some students used copper wire to cut the hyperbolas and then they welded with circumferences of different radius to get the correct shape, later they covered with paper or tape (second chimney in figure 6). One student searched bottles, cans and vases until she found a can with the correct form; finally she covered the can with plasticine (last chimney in figure 6).

Some students had many troubles to get the exact shape of the chimney, they tried to use only paper and glue but the structure was too heavy and the chimney collapsed. Other students tried only plasticine or other soft materials but chimneys also collapsed. Finally some other tried copper wire (not welded) and adhesive tape (like scotch tape) with the same results.

Figure 6. Chimneys made of cardboard, copper wire and wood.

Conclusions

Two of the main principles in ITESM are "students have to search the knowledge by themselves" and "students have to learn to learn". Our students searched in books and internet for different ways to find a hyperbola equation. We think this was a successful part of the activity because students faced a challenge and they could solve it; and the mentioned principles were fulfilled.

We got many interesting results but our main conclusion is that students really got involved in the construction of the model, they used many different materials for the body of the model, but the most amazing was the richness of different ways they used to get the perfect shape.

The answers we got lead us to think that students are capable to solve some real problems even if they think they can not. We think that this activity was very successful but we know it can be better and we are actually redesigning it.

References

Buck Institute for Education. (n.d.) *Project Based Learning. The Online Resource for PBL*. Retrieved on june 1st, 2010 from http//pbl-online.org.

Martin, F. (1996). Ideal and Real Systems, A study of notions of control in undergraduates who design robots. In Y. Kafai & M. Resnick (Ed.), *Constructionist in Practice: Designing, Thinking, and Learning in a digital world* (pp. 297-322). New Jersey: Lawrence Earlbaum Associates Inc. Publisher.

Papert, S. (1996). *The connected family bridging the digital generation gap*. Marietta, GA: Longstreet Press Inc.

Papert, S. (1996). A word for learning. In Y. Kafai & M. Resnick (Ed.), *Constructionist in Practice: Designing, Thinking, and Learning in a digital world* (pp. 9-24). New Jersey: Lawrence Earlbaum Associates Inc. Publisher.

The "computer" tells a story?

Márta Turcsányi-Szabó, *tszmarta@inf.elte.hu*
Dept. Media and Educational Informatics, Eötvös Loránd Univeristy, Hungary

Otília Pasaréti, *pasareti@gmail.com*
PhD student at Dept. Media and Educational Informatics, Eötvös Loránd Univeristy, Hungary

Abstract

Drawing and storytelling are essential activities of children in their early and elementary ages. These are best done by natural motoric and oral activities, however there are several ICT tools that were developed for this age group for supplementary activities using computers. All tools have different features that can be emphasized as beneficial for certain developmental activities.

The paper goes through discussing the types of storytelling tools TeaM lab developed during the past 25 years (*KIDLOGO* editor in Terrapin Logo, *Storyworld* in Comenius Logo, *LinoLiner* in "Imagine" during the MATCh project, *Interactive storytelling* in Creative Classroom and *TellingYouInPictures* editor in Colabs project using Imagine, and several other tools for expressing stories, poems and songs creatively, using interactive multimedia (within Imagine and Scratch), and describes their main features and developmental aims.

Figure 1. Interactive story made by children using TeaMstory builder at kindergarten

TeaMstory builder tool (created with Imagine) is the most recent development, which was introduced into three different kindergartens. The aim of the research was to deploy TeaMstory builder within the story creation and re-telling activity of each kindergarten, investigating how children are able to express their imaginations creatively and to what extent they are able to take part in the creation process collaboratively. The paper describes in more length how the features of each tool allows different developmental processes to take place.

Story related activities are perfectly suitable for kindergarten children, which reflects their fears and resolves them, creates an intimate sphere where internal ideas can be visualised and provides security due to the emotional binding towards the story teller person, thus a perfect form for internal visualisation takes place through the processing of experiences. TeaMstory builder encourages teamwork, interactions and a creative construction process.

Keywords

kindergarten, elementary, digital storytelling

Storytelling tools developed for early years

Providing tools for digital storytelling for the early and elementary years has been one of our constantly re-defined aims at TeaM lab (http://teamlabor.inf.elte.hu/) throughout the past 25 years, using the specific tools of each time period, looking for ways to develop cognitive skills. Drawing and storytelling are essential activities of children in their early ages. These are best done by natural motoric and oral activities, however there are several ICT tools that were developed for this age group for supplementary activities using computers. All tools have different features that can be emphasized as beneficial for certain developmental functions. Without giving any kind of global view on these tools, we point out some specific links: the pioneering work of Rachel Cohen (Cohen, 2005) - gave ideas to the development of several story tools and networks, who's main research goal was early age teaching of language(s); KidPix (http://pixelpoppin.com/kidpix/) - one of the first and most popular, that by now grew into a complex multimedia authoring tool; for current tools check out Alan Levine's collection of StoryTools (http://cogdogroo.wikispaces.com/StoryTools).

Earlier developments of TeaM lab

KIDLOGO (Turcsányi-Szabó, 1997a) was the first implementation in Terrapin Logo on the Commodore 64 (http://www.terrapinlogo.com/) that was introduced to kindergarten children using an icon driven environment to produce drawing in Logo style, adding music, animation and text to compose stories (See Figure 2.). These drawing activities, however, not only allowed the expression of creative figures, but were based on sound geometric principles (Turtle geometry) and also introduced the use of text (as visual element and also as name of sub-figure drawn, which could be redrawn from any position by evoking their name), visual problem solving (building figures from sub-figures) and programming (named sub-figures were subroutines after all) at an early age, which could be further cultivated in early elementary years. Even though the environment required the use of angles, kindergarten children had no problem with its understanding and fluent use. Children were also able to differentiate between the use of names as sub-drawings, as drawing processes (programming code) and finished pictures. Helping each other and working together was very common and they often shared their works as programming codes (Turcsányi-Szabó, 1997b). It must be mentioned here that this kindergarten project lasted for about 10 years (from 1984) and the only way the process could be handed over to the next generation is by the children themselves(!) constantly teaching the newcomers.

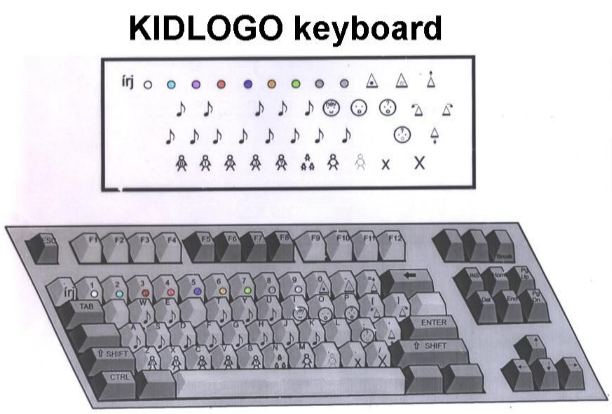


Figure 2. KIDLOGO icons on keyboard

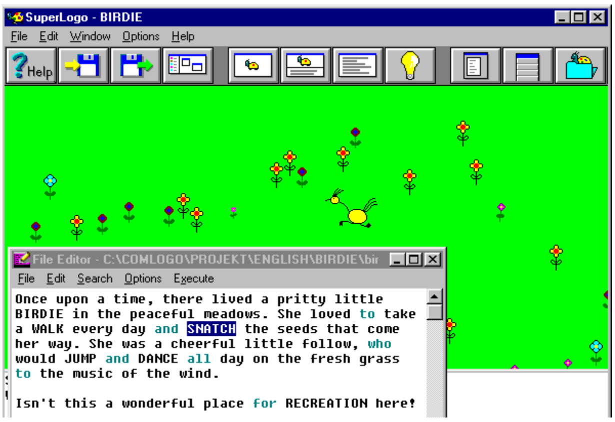


Figure 3. Comenius Logo Storyworld

With the appearance of Comenius Logo (1997), the new environment triggered the integration of special hyper-functions that would enhance the story environment. The introduction of complex animations, story editing and programming defined the environment to be used by elementary aged children, who not only like to play with interactive storybooks, but would prefer to create their own. Here, different animations (not only drawn {by changing shape} but also programmed {actually movement: by changing angle and position}) could be assigned to words, expressing the defined movements. These words could then be used within the story composition and when the words were clicked, the animations were evoked. These sub-constructions could then be further used to create more complex functions and the whole story could be developed into a start-up program (see Figure 3.). The **Storyworld** was included in the NETLogo e-Learning (Turcsányi-Szabó, 2001a) material designed for children and their teachers, where experiences showed that that these circular activities (story – animation – procedural programming) actually triggered each other with high motivation. The Hungarian NETLogo portal (http://kihivas.inf.elte.hu/halogo/) provided an environment for a community to develop, where both children and teachers were mentored by university students, updating the material.

Several other forms of media design were also introduced within these materials, introducing action poetry and text based animations that modelled language constructions, as well as game design. Several competitions were launched in which John von Neumann Computer Society provided prizes for winners and these environment became country-wide used tool in fulfilling the ICT related parts of the National Curriculum. We used this material also within in-formal education together with another material (Creative Communication), with which we mentored children attending telehouses, through individual and collaborative activities. Creative forms of expression were one of the main ideas that were addressed using various tools, for mindmapping (Turcsányi-Szabó, Pluhár, 2003), graphic design, typography, narratives, animations, ...etc. (Turcsányi-Szabó, 2001b).

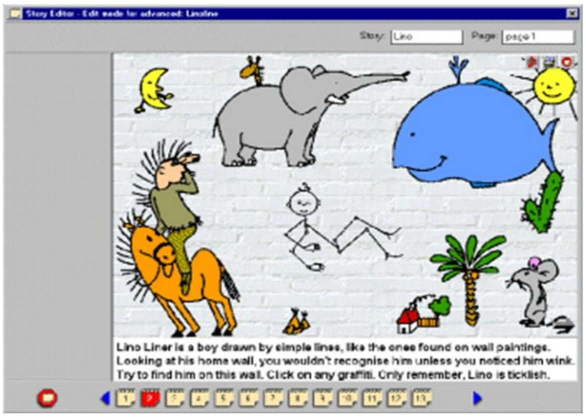


Figure 4. MATCh "Imagine" editor

Figure 5. Imagine Interactive storybook

The **MATCh project** (http://comlogo.web.elte.hu/team/match/) gave a wonderful opportunity to experiment further with storytelling tools (see Figure 4.) and the results of the experiments with children gave us lots on further experiences and ideas for later implementation (Turcsányi-Szabó, 1999) as well as the very idea for developing "Imagine". With the appearance of the Imagine authoring tool (2006), again the notion of storytelling was picked up (see Figure 5.).

Colabs project (http://matchsz.inf.elte.hu/Colabs/) investigated how children can learn through the internet collaboratively - among others - the "Picture Communication" portal (see Figure 6.) was born, where internationally understandable stories could be produced by using the "TellingYouInPictures" editor to compose complex thoughts and using a multilingual picture dictionary (containing nouns, verbs, and adverbs) to express stories in form of sequential pictures (Abonyi-Tóth, Bodnár, et al., 2005).

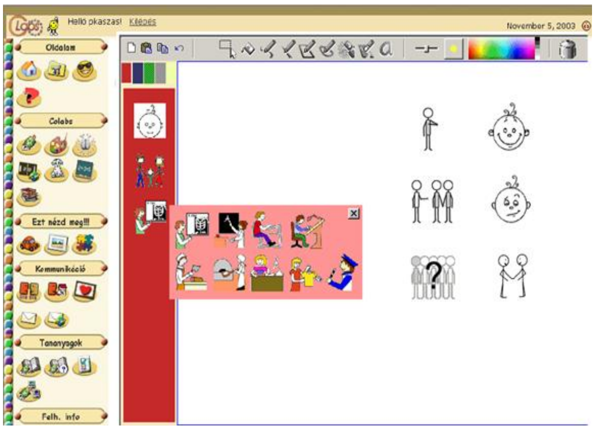


Figure 6. TellingYouInPictures editor

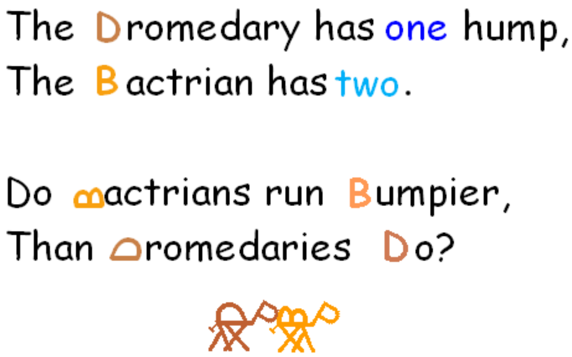


Figure 7. Different forms of interactive poems

Colabs project also stared out the basics of the **Creative Classroom** e-Learning material and activities that not only addressed digital literacy, but was an introductory course on programming in an object oriented environment and an inventory for different forms (see Figure 7.) of creative interactive expressions, which is freely available in both English and Hungarian languages and in Hungary it is frequently used in fulfilling parts of the National Curriculum (Turcsányi-Szabó, 2006a; Turcsányi-Szabó, 2006a;). Storytelling is again an essential part of the activities which has been emphasised by producing a Storyportal (http://meseportal.ini.hu/), which concentrates solely on the minimalistic direct manipulation and programming features of Imagine that allows the development of creative stories (Turcsányi-Szabó, Paksi, 2007). As an alternative tool for both storytelling and game creating, we developed the **Hungarian Scratch** portal (http://scratch.inf.elte.hu/), where children can go through a series of interesting bite-sized learning materials to learn how to develop complex expressions using LEGO-like code building structures in an object oriented environment.

TeaMstory telling

Features of TeaMstory builder tool

Throughout the past 25 years, kindergarten children were always in the main scope of attention. Thus a **TeaMstory** builder (http://teamese.inf.elte.hu/) was developed within Imagine, using an interface to suit early and elementary users in creating stories in collaboration.

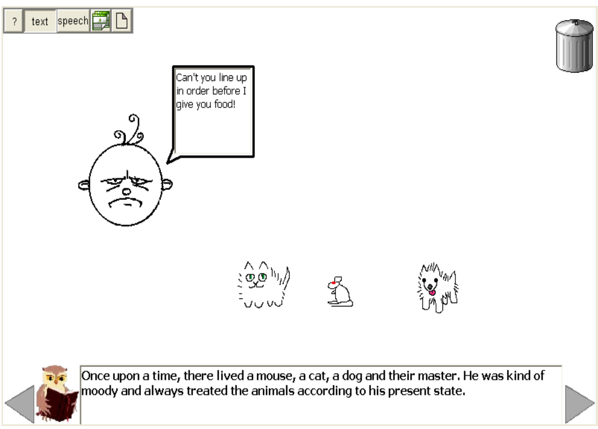


Figure 8. Editor configuration for kindergarten

Figure 9. Editor configuration for elementary

The user interface of TeaMstory builder can be configured, depending on abilities and needs:

- The "text" feature allows the use of text within the story-box. Clicking on the story-box, the text-to-speech reads out the text inside in the language installed. Feature can be disabled.
- The "speech" feature allows the use of speech bubbles. Clicking on the speech bubble, the text-to-speech reads out the text inside in the language installed. Feature can be disabled.
- Shift between small and capital letters were possible and sound can also be enabled or disabled. *Features listed so far were designed to be used by those able to read* (Figure 8).
- Clicking on the background, a panel will appear with pre-installed actors. Clicking on one of them will make it appear on the background itself and the panel disappears.
- Right clicking the actor changes its appearance; middle click allows recording narrative for actor; left clicking will make the narrative be heard. The owl is the story teller, so the main recording of the story is to be assigned to it.
- Actors could be dragged on the background, thus their movements could be played out
- Normal direct manipulative features of Imagine allows drawing on the background, creating new actor (not involved in story starter) or saving one's work.
- Actors and speech bubbles can be deleted by throwing them in the bin.
- New page button will create a new page with all the same functionality and actors involved in order to ease creation of the story-line.
- Clicking the right or left arrows, flips pages. *Remaining features for non-readers* (Figure 9).
- It is also possible to save the story on the web (when it is totally finished), but it requires a specific command to be written in the command line and clicking on the Enter key.

Kindergarten settings

The TeaMstory builder was tested in three different kindergartens (Pitypang, Meseerdő and Büköny) with different perspectives of childcare and technology use. ICT is since long part of Pitypang kindergarten's every days, which eases various administrative tasks, aids preparation of activities and provides enhanced forms of communication with parents, where several computers, cameras, printers and scanner are part of the openly used tools (by children too). Computers are also present in the homes of children and by showing proper attitudes with ICT use through the rules implemented at kindergarten (e.g. computers can be used for max 20-25 minutes at a time by usually about two persons together), teachers believe that these attitudes would be transferred into homes as well. Creating, recording, illustrating, producing story books and re-telling stories are essential part of the literature activities of Pitypang kindergarten (Méder, Varga, Knizner, 2005). TeaM lab has good working relation with them since five years by continuously developing edutainment tools for their specific use (http://pitypangovoda.hu). The other two kindergartens, Meseerdő (which is private kindergarten) and Bükköny (standard governmental kindergarten), though well equipped, do not have any specific relation with ICT.

The aim of the research was to implement TeaMstory builder within the story creation and re-telling activity, investigating how children are able to express their imaginations creatively and to what extent are they able to take part in the creation process.

The main research took part at the Pitypang kindergarten, where TeaMstory activities were integrated into the normal storytelling activities of the kindergarten. The sessions took place every Friday (lead by the researcher) for three months, 1-1 ½ hours in overall length, during which about 10-12 children took part with self initiated circulation.

In the first stage, the story books previously created on paper by children with their own drawings and story text (written on the pages by the teacher) were scanned and processed into interactive story books by adding animations and interactive elements. Children were thrilled to see their own creations and impressions become alive. They re-told lot of experiences while playing with their own interactive narratives and recalled a lot of events specific to them in relation to the story elements. They could move characters or click on them to invoke animations imposed by the story and listen to small sound elements too. They enjoyed very much these interactions; however the "closed" product had its limits for interaction, structure and implementation that were not mirrors of children's own fantasies, but that of the creators.

TeaMstory builder had an original version published in the SchoolNet digital repository by the developer (Turcsányi-Szabó, 2006b), which was used as an initial tool by the researcher (Pasaréti, 2009) and developed further through an action research cycle, which collected, analysed and advised further modifications in order to achieve aimed research goals, by modifying or implementing new functionalities, according to the experienced needs. The final configuration used in kindergarten switched off speech bubbles, alternatively used story-box with capital letters and switched of text-to-speech (though at times it was used for fun).

In the beginning, the researcher showed children the use of tools and how they can express their ideas with existing functionality, she helped children attaining their aims, later just acted on wishes of children and as times passed children slowly came forward to do the activities themselves, leaving the researcher in the background for "just-in-case" needs. The bigger ones quickly mastered all functions in order to build their stories and the smaller ones managed with more collaboration: one clicked on middle mouse button to start recording, while the other started talking holding the microphone and the third became responsible for the background (see Figure 10.). The computer provided instant feedback, no stress if at fault, so mistakes were realised by children and corrected by themselves or asked for help where needed.

Figure 10. Working in collaboration

Figure 11. Bewildered by story telling

Developmental experiences

Apart from proper body-build, children need to acquire several skills before entering school in order to be able to master reading, writing and arithmetic, the basics of which lie in attentive involved activities (137/1996 (VIII. 28.) Governmental regulation).

Developing skills for aural self expression

One of the requirements for school entry is that the child should be able to easily memorise short poems, nursery rhymes and stories. The repetitive listening to finished stories helps children with memorisation in the first place, furthermore the recorded enthusiasm in story telling proves the amount of motivation children possessed when mastering the story elements. Each child wanted to be Little Red Riding Hood, Piggy or the wolf in turn, to be able to express his/her version and children were eagerly helping each other to recover forgotten parts, discuss or debate happenings of a tale. It was quite evident how the amount of narratives developed and grew within children's memories.

The recognition and application of visual sequencing is also an important ability that TeaMstory builder provided by the sequential construction of story parts (which parts come first and which part follows), their circumstances (season and timing of the day or weather conditions).

A trivial requirement for children is to be able to express oneself in short compound sentences in which the freedom of story development within TeaMstory allowed children to develop such skills. At first the sentences were less fluent, but later one came after the other and whole story telling sessions evolved (see Figure 11.).

Developing skills for visual self expression and fine motoric skills

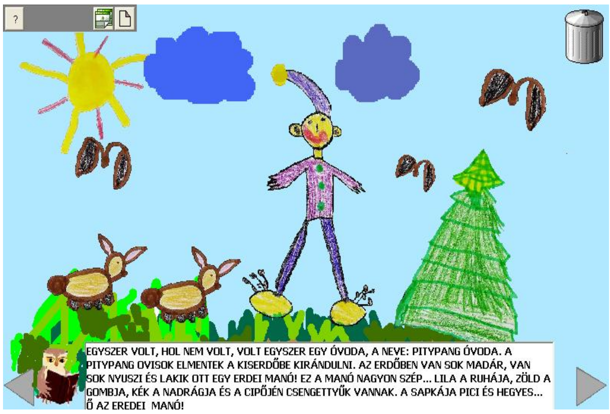


Figure 12. Editor configuration for kindergarten

Figure 13. Editor configuration for elementary

Keyboarding, the fluent use of the mouse for drawing and different manipulative activities develop fine motoric skills, but these were extended by off-computer activities, like drawing on paper (see figure 12) or folding Origami figures (see Figure 13), which are normal kindergarten activities. Children were happy to see the connection of their new activities with that of the usual, it gave them a break from computer use and something to take home after the activities. The manual works were scanned and imported within the TeaMstory world, which triggered further motivation upon recognising their own creations to go further in developing the story.

Children gave no preference in selecting figures (see Figure 9.) that were realistic (photo-like) or draft scratches created by themselves, they chose at random, but once their own products were involved, they made several proud comments on who’s work that was.

There were always more children awaiting their turn at the computer, so a relevant rooster emerged quickly: while some children were busy at the computer, others earned their turns by creating further visuals to be involved at the next stage. Thus stories often developed depending on the actual fantasies lived through on spot, the final creations and their sequences of involvements.

The use of the drawing tools was also a highly appreciated activity on its own. First they had to master the use of background drawing tools, in which choosing the proper colour was an important event in itself, then the creation of smaller figures and filling larger areas with certain colours. The bigger ones later found out how to create filled geometric figures in order to ease object creation. Later they learned how to modify drawings of actors within the Logomotion editor of Imagine, which very well illustrated the emergence of existing figurative schemes within children, e.g. they drew spots of skirt, apron, boots and wrinkles for a grandmother figure.

Developing mathematical skills

Some stories involved counting and sequential skills, like the Carrot story (see Figure 13.), which children played and recited several times dragging the actors into their right positions.

Spatial orientations were also important while playing with a story, as the places of actors were modifiable. The notion of: *inside, outside, under, over, behind, in front, beside, right* or *left* were easily practiced during story telling. Any number of actors could be created in a row and if actors were not needed any more, they could be thrown inside the bin in order to practice having more or less of them at a time.

Teamwork

TeaMstory (or any other story building or gaming tool for that matter) is not a “nanny” and was not developed to substitute educator/parents or caretakers, but rather provide a tool with which they could enhance their communicative interactions. At first children need help, advice and later an action triggers reactions as means of communication in collaborative involvement with a sibling, parent or caretaker that the child feels comfortable with and at ease to open up his/her full spectrum of fantasy. Children working together help each other, give advice, solve conflicts, take good care for everyone to take turns and learn how to socialise on the way. They are good at collaborative problem solving under such relevant situation and specific roles emerge on the way and prove to them the importance of teamwork to attain good results that everyone enjoys. Thus TeaMstory builder had a great role in forming effective production groups, exploring individual add-on values, adjust individual will to possibilities and acceptable morals, develop unselfishness and urge to help the other, heading the right way in the process of socialising in Vygotsky’s term of “*zone of proximal development*” (Vygotsky, 1986).

Reading and writing

There is no aim in kindergarten to teach children reading or writing, however it must be noted here, that more children master reading skills at the age of kindergarten then in the past decades as media is very much effecting the lives of children. Reading is the foundation for literacy as computer use the foundation of digital literacy. The TeaMstory builder allows the use of text for those children who wish to explore the world of words at their own space. But, more importantly the tool allows an enjoyable form of concentration for about 15-25 minutes, which is definitely a requirement before entering school.

There were no significant differences in use among the three visited kindergartens, except from the reaction time of children towards technology tools. While children at Pitypang kindergarten wanted to take part in story building from the very first session, children in the other two kindergartens were able to be actively involved only after a few more sessions. Apart from that, they were also deeply involved in all sorts of similar activities and developed their own skills likewise as fluently as the others.

TeaMstory at school

The features of TeaMstory builder allows it’s more sophisticated us for children who already know how to read and write, developing several key competencies (DeSeCo, 2005):

- Language studies: to express thoughts/emotions not only in words, but in writing as well – in one’s own mother tongue as well as some other foreign languages.
- Digital competencies: to interact with information using technology and the ability to communicate using media.
- Mathematical competencies: apart from the expression of quantities and location and times based relations, context based narrative problems can be composed easily.
- Art education: aesthetics and creativity can be further developed using visual multimedia elements and lots of imagination. Visual representation of stories, poems, special or just every day happenings. Music and singing activities can well be integrated too.
- Special education: the tool can produce interactive activity book for children with special needs, that can integrate media elements to better express the intended content.

Conclusion

Games are the most important developmental activities for early age children, allowing free association of ideas that are basic physiological needs at this age, which should be a continuous, long lasting recurring process in the everyday's of small children. The unstructured experiences accumulated from the outside as well as inside world of a child can become structures within acted games, thus it becomes an important activity. Especially around 5 years of age the social activities start to gain significance, providing: belonging, security, identity, binding, joy of doing something together, possibility of interactions and communication (Zsubrits, 2007). Story related activities are perfectly suitable for kindergarten children, which reflects their fears and resolves them, creates an intimate sphere where internal ideas can be visualised and provides security due to the emotional binding towards the story teller person, thus a perfect form for internal visualisation through the processing of experiences (137/1996 (VIII. 28.) Governmental regulation). However, computer related activities should not be used instead of other games, but as a tool for games; only developmentally appropriate activities should be chosen; only activities that are relevant for this age group can keep up children's motivation; it should not substitute human interactions, but should facilitate them; it should not take away much time from indoors or outdoors activities vital for this age (Turcsanyi-Szabo, 2004).

References

137/1996 (VIII. 28.) Kormány rendelet (1996), Az Óvodai nevelés országos alapprogramjának kiadásáról (Governmental regulation for kindergartens) http://www.okm.gov.hu/letolt/elektronikus_ugyintezes/ovoda_terv_090922.pdf

Abonyi-Tóth A., Bodnár E., Turcsányi-Szabó, M., "Telling you in pictures" – communication bridging languages, Proceedings of Eurologo 2005, pp. 307-312, 27-30 August, Warsaw, Poland, (2005). at http://eurologo2005.oeiizk.waw.pl/PDF/E2005AbonyiEtAl.pdf

Cohen, R. (2005), An early literacy telecommunication exchange pilot project: the MMM project, Educational Media International, Volume 42, Issue 2 June 2005 , pages 109 – 115, Routledge http://www.informaworld.com/smpp/content~content=a714023306&db=all

Comenius Logo: © Turcsányiné Szabó, M. & Kossuth Publishing Co., (1997). Hungarian Comenius Logo 3.0, original (© Andrej Blaho, Ivan Kalas, Péter Tomcsányi), Kossuth Publishing.

DeSeCo project (2005), The Definition an Selection of Key Competencies http://www.oecd.org/dataoecd/47/61/35070367.pdf

Imagine: © ELTE University TeaM lab: Abonyi-Tóth, A., Turcsányi-Szabó, M., Windisch, J. (2006). „Magyar Imagine" (Hungarian Imagine), original (© Blaho,A.., Kalas, I., Salanci, L., Tomcsányi, P.) downloadable from http://logo.sulinet.hu/

Méder Lászlóné, Varga Mónika, Knizner Anikó (2005): Pitypang Óvoda: Informatikai nevelés elmélete és gyakorlati lehetőségei, http://www.pitypangovoda.hu/docs/Informatika%20innov%e1ci%f3.doc

Pasaréti, O. (2009), Számítógépek kisgyerekkori alklamazása – Interaktív mesekészítés (Use of computers in early education – Interactive story telling), Diploma work, ELTE.

Turcsányi-Szabó, M., Approaching Arts through Logo, in Proceedings of the Sixth European Logo Conference, pp. 284-294. John von Neumann Computer Society, Budapest, Hungary, (1997a). at http://caesar.elte.hu/~eurologo/lectures/papartj.htm

Turcsányi-Szabó, M., Designing Logo Pedagogy for Elementary Education, in Proceedings of the Sixth European Logo Conference 1997, pp. 273-283. John von Neumann Computer Society, Budapest, Hungary, (1997b). at http://caesar.elte.hu/~eurologo/lectures/papthij.htm

Turcsányi-Szabó, M. (1999). IMAGINE a Tool to Learn, Express and Explore Stories, in Communications and Networking in Education: Learning in a Networked Society, Proceedings of COMNed'99 IFIP Conference, pp 346-356, Aulanko, Finland.

Turcsányi-Szabó, M., Why S.O.M.E. Logo Environments are Suitable for Broader Educational Purposes?, in Proceedings of Eurologo 2001 Conference, pp. 97-106, Linz, Austria, (2001a). at http://www.ocg.at/activities/books/volumes/band%20156/K61Szabo.zip

Turcsányi-Szabó, M., Learning in tele-houses – a perspective for development in underdeveloped regions within Hungary, Infovek conference proceedings, Stara Tura, Slovakia, (2001b). at http://matchsz.inf.elte.hu/TeaM/infovekpaperELTE.pdf

Turcsányi-Szabó, M., Pluhar, Zs., Modular mind mapping, in Proceedings of Eurologo 2003, pp. 158-167, Porto, Portugal, (2003). at http://matchsz.inf.elte.hu/Colabs/Porto/pubs/PZs_TSzM.pdf

Turcsányiné Szabó, M. (2004). Számítógépet az ovisoknak – Mozaikok a nagyvilágból (Computers for kindergarten – mosaics from around the world), Új Pedagógiai Szemle (New Pedagogical Survey), January, pp. 87-98. – also published at http://www.oki.hu/cikk.php?kod=2004-01-vt-Turcsanyine-Szamitogepet.html

Turcsanyi-Szabó, M., Creative Classroom CD Logotron Ltd, Cambridge, (2006a). at http://www.r-e-m.co.uk/logo/?comp=imagine&html=creative.html

ed. Turcsányi-Szabó, M. (2006b). "Digitális Írásbeliség" (Digital Literacy e-learning material), Sulinet Digitális Tudásbázis (Course material for Schoolnet Digital Repository), http://sdt.sulinet.hu/

Turcsányi-Szabó, M., Attila Paksi, Logo practice: from "turtling" to interactivity, 11th European Logo Conference, Slovakia, (2007). at http://matchsz.inf.elte.hu/team/PP-Turcsanyi-Szabo.pdf

Vygotsky, L.S. (1986): Thought and Language. Cambridge, MA: MIT Press.

Zsubrits Attila (2007): A gyermekkori kötődések motívumai. (Motives of early years' bindings) Új Pedagógiai Szemle (New Pedagogical Survey), July, - also pblished at http://www.oki.hu/oldal.php?tipus=cikk&kod=2007-07-ta-Zsubrits-Gyermekkori

Reconstructing an iconcostacis as a model for the constructionistic approach in education

Lilia Pavlova-Draganova, *pavlova.lilia@gmail.com*
Institute of Mathematics and informatics, BAS, Akad. G. Bonchev, bl. 8, Sofia 1113, Bulgaria

Evgenia Sendova, *jenny@math.bas.bg*
Institute of Mathematics and informatics, BAS, Akad. G. Bonchev, bl. 8, Sofia 1113, Bulgaria

Abstract

Introducing the information technologies (IT) as a separate school subject in the Bulgarian schools since 5th grade made it possible to include themes illustrating the Dewey's idea that *education is not a preparation for life but is life itself.* An example of such a theme is the restoration of ancient vessels − a project included in an IT textbook for 7th grade. When working on the project the students are faced with problems from real life and are expected to understand in a natural way when, how and which IT tools to apply so as to help a local museum.

Such an approach reflects the crucial ideas behind the constructionsim since the vessel-restoration project is to a great extent a miniature version of a real research project in the field. Still, to develop a more realistic project approximation it is worthwhile to examine the specifics of an authentic research-art project. This is done in the context of reconstructing the iconostasis of a Bulgarian Orthodox church from 18th c. partially destroyed by a fire (Figure 1).

Figure 1. Details of the iconostasis after the fire and their digital reconstruction

The reconstructing process involves building a hypothesis about the fundamental generating idea of the anonymous master of wood-carving. This hypothesis is based on a thorough analysis of the main characteristics of the iconostasis (the epoch, the place, the artistic style) including the study of specimens *close* enough according to the above characteristics.

Based on observations on various stages of the process the authors notice some implications for the educational process in the spirit of the constructionism, viz. that both in school and university setting the *learning is most effective when part of an activity the learner experiences as constructing a meaningful product*.

Keywords

reconstruction, iconostasis, project based learning, constructionism.

Introduction

Introducing a separate school subject on information technologies (IT) since 5th grade in the Bulgarian schools made it possible to include themes illustrating the Dewey's idea that *education is not a preparation for life but is life itself.* An example of such a theme is the restoration of ancient vessels - a project on which a whole section in the textbook for 7th grade is dedicated [1]. When working on the project the students are faced with problems from real life and are expected to understand in a natural way when, how and which ICT tools to apply so as to help a local museum restore ancient Greek vessels and guess their function. For the purpose the students are expected to study the shape of an artefact, to build and explore a virtual 3D model of it (Figure 2) by means of a specially designed software application *Potter's Wheel* [2], to decode a message with hieroglyphs, in a nutshell - to put together a great part of the subject knowledge and skills acquired during the school year and to work creatively in teams pursuing a common goal.

This theme is further elaborated by Boytchev [3] in the frames of *the Math2Earth* European project for developing educational scenarios with the idea of "bringing mathematics to Earth", i. e. to demonstrate to learners of a large scale that mathematics can be enjoyable and useful in many situation of everyday life.

Such an approach reflects to a great extent the crucial ideas behind the constructionsim since the reconstruction project is a miniature version of an authentic research project. The students

- learn a lot about the culture, the habits and the dreams of their predecessors by studying the ancient artefacts;
- experience tackling problems without obvious solutions and get an idea about the professional research;
- acquire ICT-enhanced skills [4] such as: searching and selecting relevant information, splitting the problem in tasks and subtasks, working in a team, constructing meaningful products, then presenting and sharing the products of their work.

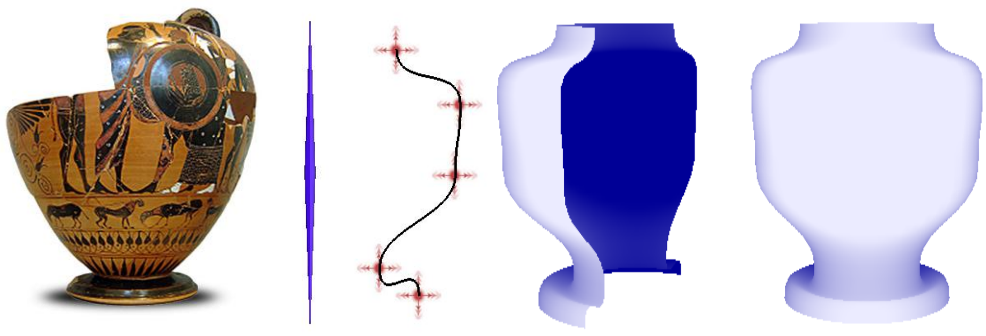


Figure 2. An ancient Greek vase and its virtual 3D model

Although the main idea behind the restoring-ancient-pottery project is based on authentic problems in archaeology many of the details (e.g. some of the hieroglyphs containing information about the function of the artefacts) are fictional. With this in mind the second author decided to learn more about the nature of a real restoration project in a slightly different context – reconstructing an iconostasis which is a woodcarving masterpiece of the Bulgarian Revival Period.

An authentic research-and-art project

The background

There are many Bulgarian iconostases that have been created through the ages (starting from 12-13 c. as stone altars). Although preceding the Renaissance they are in themselves magnificent monuments of art [5]. The earliest monument of this kind (St. Peter from Berende village) which is being preserved dates back to the 14th c.

In August 2009 a team of artists (of which the first author is a member) was given the assignment to reconstruct digitally the iconostasis of the *Dormition of the Holy Mother of God* cemetery church in Bansko so that the model could be used by wood-carvers to reconstruct the iconostasis as authentically as possible. The church was built in 18th c., and the iconostasis separating the naos from the altar space was made by an extremely talented anonymous artist.

In 1958 a fire had burst in the church, believed to be deliberate [6]. As a result one-third of the iconostasis (together with all the icons mounted on it) was scorched. A single picture of the iconostasis from a post card was preserved (Figure 3).

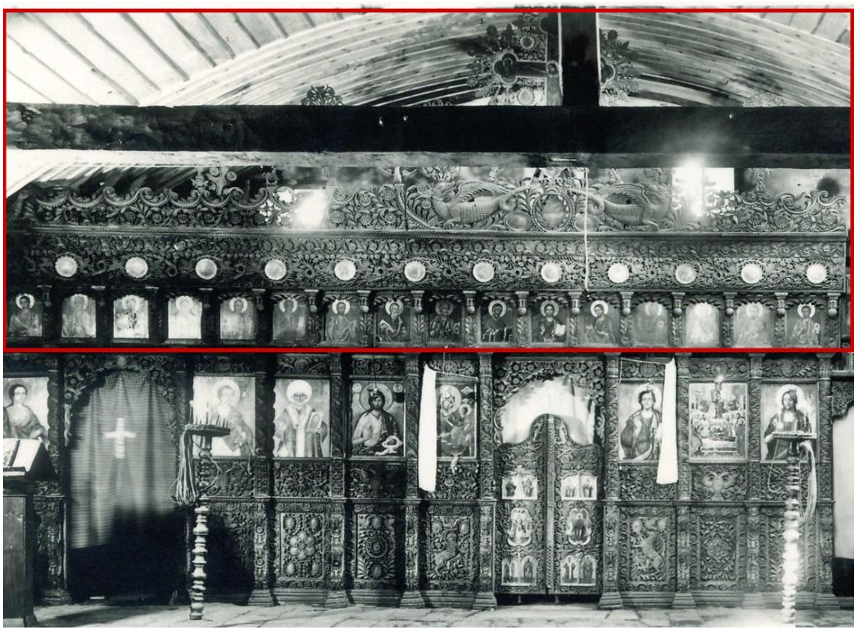


Figure 3. The iconostasis prior to the fire in 1958 (the part framed in red is to be reconstructed)

The reconstruction project through the eyes of a professional re-constructor

The floor goes to the first author now:

The main problems occurred were the insufficient information about the master and about the original iconostasis. Not only has the cross been charred by the fire but as seen in the original picture it is half hidden by the beam. Still there is a part of the iconostasis relatively well preserved after the fire which could help mainly for establishing the style of the woodcarving.

The main challenge the re-constructors faced was to build a hypothesis with the hope to extract the fundamental generating idea of the anonymous master. Such hypotheses are usually based on a thorough analysis of the main characteristics of the object of reconstruction (the epoch, the place, the artistic style) including the study of *close* enough (according to the above characteristics) specimens. The iconostasis under consideration contains a typical pattern showing that the master wood-carver has a specific mode of expression; the ornaments have clear and sharp contours, the flowers and the leaves stand out clearly against the background. According to the art expert Enev [6] *the woodcarving is very expressive, clear-cut, and sharp-lined. The major elements are flowers with birds and fairy-tale animals inserted among them.* As for the creator of the iconostasis, he is described as *skilled professional with a strong individuality, yet probably familiar with the work of his contemporaries from Athos*. Historical investigations show that the wood carver could hardly have been a monk from Athos but rather someone who had learned the craft from them. The presumed date of that iconostasis is 1801 and the experts establish a great similarity between the iconostasis to be reconstructed and the one of the churches in Golyamo Belovo. As for the woodcarving school the iconostasis creator belonged to certain schools (e.g. the famous Debar one) could be rejected based on the specifics of the elements (flowers and birds but not human figures typical for the wood-carved motives from Debar school). It is clear that the analysis should take into account the studies of experts in various fields (art, orthodox religion, particularly in Bulgaria − in the region of Bansko), the information provided by good intending local people (filtered appropriately).Then the practical work begins – taking highly professional pictures of the current status of the iconostasis in different levels of detailisation (Figure 4).

Figure 4. Details of the iconostasis after the fire.

One of the challenges for the re-constructors is to figure out the original appearance of the cross – at a first glance there are a series of potential candidates from other Bulgarian churches (more simple in terms of woodcarving, from the same region and period (Figure 5) but also very complex ones as the crown on the iconostasis (Figure 6).

Figure 5. Crosses from Bulgarian orthodox churches of the same period

Figure 6. The crown on the iconostasis in the naos in the Church of the Nativity in Arbanassi

An additional problem compared to the problem of reconstructing icons for instance is that the wood-carving could not be demounted after it has been charred.

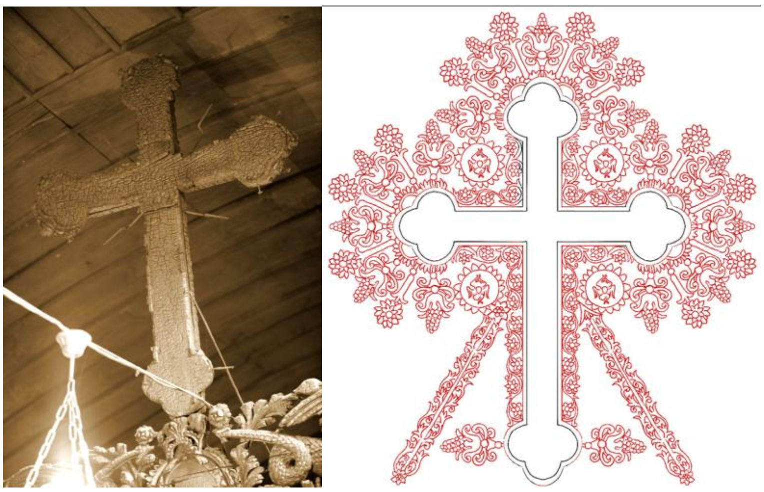


Figure 7. The charred cross and its digital reconstruction

After a careful analysis of the artistic line we can model the specific elements – flowers, leaves (Figure 7).This analysis is based on the logics from religion-symbolic point view, and from architect point of view.

The series of elements give an idea about the whole ornament, and from there – of the whole panel of the iconostasis.

The next step is to make a retrograde strategy by which to verify the hypothesis – how close you are to the reality in terms of size, distance, placement of the elements, etc. This verification is based on hundreds of pictures scaled in 1:10, fitting them in transparent paper, completing the missing parts of the picture by manual drawing (Figure 8). Then we scan the transparent paper, process it digitally to turn it into a vector graphics with a very high quality, scale it to its real size and print it on a plotter to be used as a model by the wood carver performing the final reconstruction.

The reconstruction of the icons requires more complex artistic and graphical analysis. For the purpose the research team will use an iconographic digital library with elements of semantic access [7, 8].

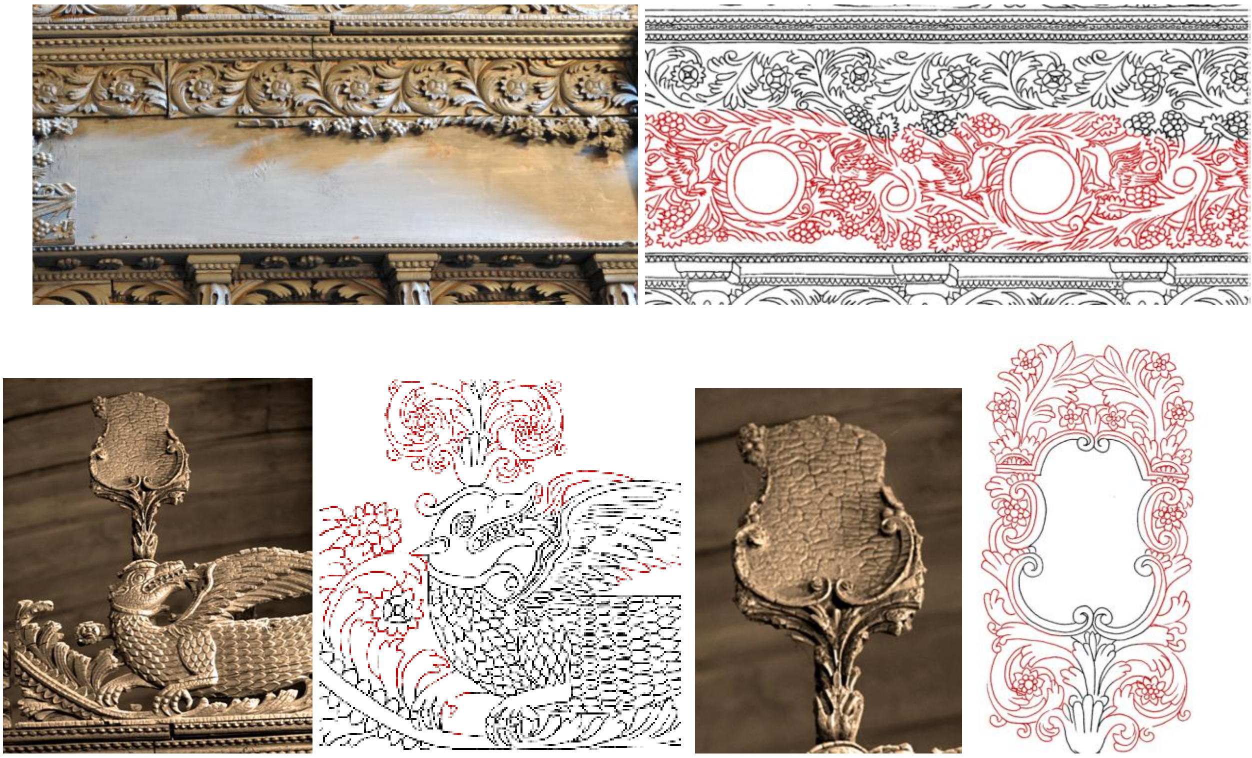


Figure 8. The digitalization makes it possible to model and reconstruct the objects with high fidelity

The main achievement of this project is the synergy between the extraordinary creativity of the anonymous master and the contemporary analytical view and knowledge about this masterpiece together with the new technological facilities. Without them the precise artistic reconstruction would have been extremely difficult.

Of course, being an artist yourself it is very difficult to suppress your ideas so as to remain faithful to the old master and to the canon. But again you learn so much... And I can't forget what my then 4 year old son said: *You don't seem to work, you are artists... the work couldn't be a joy.*

Back to the educational setting

What we realized after exchanging our experience in a project-based context is that it would be impossible not to simplify the realty in a school setting. Still there are many things in common. The interdisciplinary approach applied in the case of reconstructing the iconostasis included profound knowledge on the history of the orthodox art, digital photography, artistic skills, the use of specialized graphic software thus enriching significantly the palette of the project team with new tools and creative potential.

Although on a smaller scale the 7th grade students working on restoring the ancient vessel are expected to integrate knowledge from different fields in a motivated way. They have to realize from a personal experience that the participation in a project requires a successful implementation of ICT-enhanced skills including collaborative work, finding and using various resources, ensuring that tasks are complying their deadlines, transferring ideas and results from one domain to another, and finally, dispatching resourcing including personal efforts and time.

In conclusion, our "project-based" discussion reinforced our belief that both in school and university setting the *learning is most effective when part of an activity the learner experiences as constructing a meaningful product*. Of course, re-constructing is not less meaningful...

References

[1] Sendova E., Stefanova E., Boytchev P., Nikolova N., Kovatcheva E. (2008) *IT education - challenging the limitations instead of limiting the challenges*. In Proceeding of 6th International Conference for Informatics and Information Technology, CIIT, Bitola

[2] Boytchev, P. http://www.elica.net/site/museum/Dalest/dalest.html, 1.10.2009

[3] Boytchev, P. (2010) *(De-)Coding adventures for young researchers*. In Bringing Mathematics to Earth, J. Anderson et al (authors and eds), PRVOKRUH, Prague, ISBN 978-80-901470-2-7

[4] Sendova E., Stefanova E., Nikolova N., Kovatcheva E. (2008) *Like a school (of fish) in water (or ICT-Enhanced Skills in Action)*. In LNCS Informatics Education - Supporting Computational Thinking, R. Mittermeir and M. Syslo (eds), pp. 99-109, ISBN-13 978-3-540-69923-1

[5] Gergova, I. (1993) *Early Bulgarian Iconostases 16-18th century*, Bulgarski hudozhnik, Sofia

[6[Enev, M. (2002) *Bansko, A Center of Culture Through the Ages*, Sofia

[7] Pavlova-Draganova, L., Paneva, D., Draganov, L. (2007). *Knowledge Technologies for Description of the Semantics of the Bulgarian Iconographical Artefacts*. In the Proceedings of the Open Workshop „Knowledge Technologies and Applications", 31 May – 1 June, Kosice, Slovakia, pp. 41-46

[8] Paneva-Marinova D., Pavlova-Draganova, L., Pavlov, R., Sendova, M. (2008) *Cross-Media and Ubiquitous Learning Applications on top of Iconographic Digital Library*. In the Proceedings of the 14th International Conference on Virtual Systems and Multimedia (VSMM'08), 20-25 October, Lymassol, Cyprus, pp. 367- 371

Constructionism Applied

Pavel Petrovič, *petrovic@acm.org,* Dept of Applied Informatics, Comenius University

Abstract

Efforts within the constructionism concentrate around the platform used by the original author of the idea, Seymour Papert. Papert was working with LEGO educational division on what later became a known product LEGO Dacta, programmable in a parallel Logo-like language with commands to control the motors and sensors of a model connected through a wired interface. The current educational projects of LEGO are a follow-up of this good start, even though usually not using Logo language. Some efforts to compensate this include [1,2]. LEGO sets, however, always remain at the level of toy construction sets, and toy models. We argue, that the real constructionism starts where the LEGO ends, and we call it *constructionism applied*. This poster summarizes our applied constructionism experience, working with children of different age level.

FIRST LEGO League (FLL) – not only LEGO

Contests are a far greatest motivation for young people attending our robotics clubs (whether 10 year olds or 14-16 or 17+). In our region, four different contests are targeted at young people: FLL, RoboCup Junior, RobotChallenge, and Istrobot. In FLL, in addition to building a LEGO model, children work on a research project, put together their own creative idea, support it by their own designs, prepare, and show a presentation. Some of them show a deep excitement. For instance, the teams construct a model of an airship, show a little play to demonstrate and explain their idea.

Figure.1. Boat robot for Istrobot contest. Capable of navigating 100s of meters across a lake using compass sensor.

RoboCup Junior(RCJ) – category Soccer

The main RCJ category – robotic football/soccer is the core contest discipline, and LEGO robots are an excellent start. Using advanced LEGO IR seeker and compass sensors, children can put together a working robot in couple of hours. The real experience starts when they try to build their own robot. Two students from our club, have been building their robot for two years: they were able to design a 3D model of their robot using CAD software, have the parts cut using waterjet, design complex navigational strategies using omni-directional wheels, program specialized microcontroller boards with cameras and simple image-processing capabilities.

RCJ Slovakia – category Construction. Unlike the usual RCJ categories, and most of the other contests, where the task is (almost) the same every year, we maintain a traditional category labeled “Construction”. Here, the students learn the task only at the contest and spend several hours building and programming their robots to solve it. There are many advantages to this approach: the students can join the contest without any preconditions, they do not need to block the club's equipment for many weeks before the contest, and they show their real skills, as contrasted to skills of their team leader.

Acknowledgment. This work is supported by the EU European Regional Development Fund.

References

[1] Petrovic, P. (2007) Program Your NXT Robot with Imagine, Eurologo 2007.

[2] MicroWorlds Ex, Robotics Ed., LCSI, at: www.microworlds.com/solutions/mwexrobotics.html.

Robotic Educational Platform based on Ball Robots

Zuzana Zajíčková, *zajickova@st.fmph.uniba.sk,* Comenius University

Pavel Petrovič, *ppetrovic@acm.org,* Dept of Applied Informatics, Comenius University

Abstract

The two recent decades witnessed educational robots being developed, and evaluated in schools, kindergartens, free-time centres and clubs. Resembling cars, bees, drawing turtles, walking androids, insects, or construction sets that allow building anything within the scope of imagination. Examples include BeeBot, Probot, and Roamer from Terapin, Mavin and Robonova from HiTec, Yeti and Asure from AREXX Engineering, Pololu3pi from Pololu Robotics and Electronics, Solarbotics Mini-Sumo, Scribbler and BoeBot with extensions from Parallax, LEGO NXT and LEGO WEDO construction sets. The main idea is to move the constructionist playground out to the real world, the natural environment of the learner, where he or she interacts more directly, utilizes more senses, works in 3D, explores real forces, shapes, volumes, etc. This is in a sharp contrast to scenarios locked inside of the computer screen. We move further on: we propose a creative and non-conventional platform, which goes beyond the traditional wheeled or legged robot morphologies. Alternate morphologies bear unprecedented educational potential and entertaining educational experience.

The idea of an autonomous ball robot is not new. Successful robots were built and put on the market. Examples include the cleaning Robomop robot by Robomop International, Groundbot from Rotundus. Mono-wheel sphere can roll forward and backward in a straight or bent trajectory. Differently, a ball can shift its centre of gravity along the three axes, in order to start free-rolling movement in a desired direction. Independently, a study of a similar navigation type has been performed in simulation by [1]. We propose the mass to be concentrated in coupled pairs of co-centric points as shown in Figure 1.

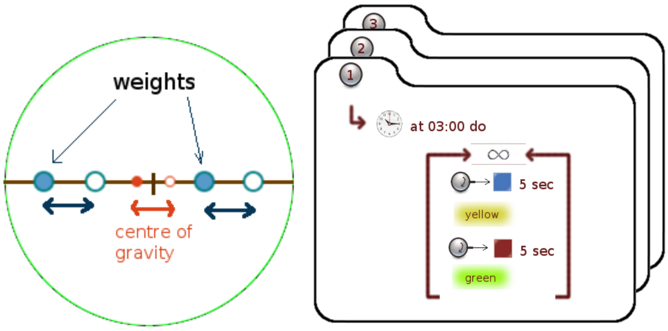


Figure.1. Principle of autonomous ball movement (left), concept of the programming paradigm: Iconic language, multiple balls, events, control structures (right).

Multiball Educational Platform

The set consists of 5 balls that can autonomously roll in an arbitrary direction. The balls can detect collisions with objects, and external displacement by a human. They sense the rotation along three perpendicular axes, emit light in changing colour (using multi-colour LEDs), produce sound, and detect the colours of light coming from all directions, typically from neighboring balls. The balls can communicate with each other, and receive program or messages from a master computer over BlueTooth. They will be programmed using iconographic programming language for children that is event-based, and contains simple control structures, timers, variables, integer arithmetics, motor and sensor control commands. Children can test their programs in simulation. Our position poster presents the platform idea and the early work on the prototype.

Acknowledgment. This work is supported by the foundation Nadácia Tatra banky.

References

[1] Der R,Martius G, and Hesse F (2006). *Let it roll – Emerging Sensorimotor Coordination in a Spherical Robot*. Artificial Life X, p. 192-198.

A constructionist approach to a contested area of knowledge

Dave Pratt, *d.pratt@ioe.ac.uk*
Institute of Education, University of London

Cristina Yogui, *cris.epl@freedom255.com*
Institute of Education, University of London

Abstract

When the media reports socio-scientific issues, there is routine reference to risk and in England risk has become part of the curriculum in Personal, Social, Health and Economic Education (PSHE), Citizenship, Science and to a lesser extent in Mathematics. It is therefore timely to consider the teaching and learning of risk to young citizens. Yet, risk is a contested concept, not only in the sense that risk is perceived in very different ways according to context, experience and perhaps personal disposition, but also by experts who disagree about what risk is. Thus, although the nature of risk is not yet well established or defined, it is vitally important that pedagogies of risk are developed in response to curriculum requirements and societal need.

This paper reports on one aspect of a study that is developing fundamental ideas about a pedagogy of risk. The approach is to iteratively co-design with teachers software tools that seek to perturb the teachers' knowledge about risk and about the teaching and learning of risk. Conjectures about the nature of the pedagogy of risk are embedded in successive iterations of the software design, according to design research methodology.

In conventional situations, designers have a clear starting point for their work. They are usually working with an established area of knowledge, often mathematical or scientific, and, by conducting an epistemological analysis of that knowledge domain, designers are able to imagine possible starting points. In addition, designers would also typically be able to draw on research about understanding of that area of knowledge. Because risk is understood in many different ways and is a concept that is contested by experts, designing tools to research knowledge about risk raises new challenges.

In this paper, we report on the design of the emergent software in order to tease out the underpinning rationale within which lay principles to inform the pedagogy of risk. We report ten design outcomes, most emerging directly as a result of the contested nature of risk. These outcomes are structured across four fields: (i) decision-making in complex scenarios; (ii) making personal model explicit and computational; (iii) fuzzy quantification; (iv) facilitating co-ordination of dimensions of risk.

We discuss the extent to which this design aligns or contrasts with more conventional microworld design. For example, the outcomes align closely with notions of purpose and utility, phenomenalising, quasi-concrete objects and the design heuristic of fusing control with representation. On the other hand, the approach when designing in a contested area of knowledge seemed to embrace expressive modelling than exploratory modelling as normally found in microworld design.

Keywords (style: Keywords)

risk; probability; constructionism; microworlds; design; mathematics education; science education

Introduction

The media routinely reports socio-scientific issues through reference to risk, often in exaggerated ways. There is a strong societal need for public understanding of risk. Yet, risk is a contested concept, not only in the sense that risk is perceived in very different ways according to context, experience and perhaps personal disposition, but also by experts who disagree about what risk is. Thus, although the nature of risk is not yet well established or defined, it is vitally important that pedagogies of risk are developed in response to curriculum requirements and societal need. In this study, we have been working with teachers to explore their knowledge about risk and its pedagogy. The aim in this report is to examine the design issues raised by the need to design a window on teachers' knowledge about a contested area.

When Papert (1982) talked about the designing of microworlds in his seminal work, Mindstorms, he offered a memorable metaphor in which the designer plants mathematical nuggets of knowledge for the learner to stumble across. But what does the aforementioned designer do when the knowledge to be addressed is contested and therefore loosely defined? This was the task that confronted the research team in the study reported in this paper, when trying to develop pedagogies for the teaching and learning of risk.

When designing for mathematical abstraction (Pratt & Noss, 2002), it is possible to carry out an epistemological analysis to reveal the key powerful mathematical ideas that lie at the root of the topic and to imagine a variety of trajectories that students might take that would imbue those ideas with utility (Ainley et al, 2006). Thus, very young children learn how angle and distance can facilitate drawing and animation through the use of turtle graphics and older children come to appreciate the power of variable when developing projects that involve Logo procedures. Papert's metaphor allows us to imagine this activity as children stumbling across the ideas of angle, distance and variable, not entirely fortuitously since the design of turtle graphics and Logo purposely positions those ideas as controls for the child to use in pursuing his/her objectives. Conventionally, students often struggle to understand the significance of powerful mathematical ideas such as variable in algebra but the constructionist approach promises to generate meaning through purposeful activity. In the above examples, the powerful mathematical ideas are clear, well-defined and entirely meaningful to those enculturated into mathematical discourse but the ideas are often rather less concretised (after Wilensky, 1991) to naïve learners.

In some areas of mathematics there is ambiguity in the mathematics itself though these are rare, at least at school level. Uri Wilensky (1997) has discussed the anxiety of students who are unsure of the epistemological basis for probability. More recently, alongside Dor Abrahamson (Abrahamson & Wilensky, 2007), approaches based around Netlogo have aimed to facilitate learning by bridging across the alternative epistemological interpretations of probability. Probability has in-built ambiguity and experts shift between classical, frequentist and subjectivist definitions according to the problem being solved or personal disposition. Nevertheless, each of those definitions is pretty well worked up and the design task becomes one of seeing the connections (and perhaps distinctions) between the competing views. In other words, an epistemological analysis of probability reveals clearly defined alternative views held by experts in the field and the designer is able to imagine ways of planting those ideas into an environment where they will be used and connected. For example, Abrahamson and Wilensky (2007) describe how their students made connections across alternative epistemologies for probability by tackling the same problem in different ways: by building towers of possible combinations and through collecting data from physical experiments and computer-based simulations.

This paper discusses our solution to a design task that involved knowledge that is fundamentally contested by experts and subject to many personal interpretations. One reaction might be to ask why should learners engage with such loosely defined knowledge. But it so happens that, in our view, the concept of risk, despite its lack of clarity, is of immense significance to citizens. Indeed, risk has been characterised as an integral part of the discourse of late modern society (Beck, 1992). The concept of risk has, in the last few decades, come to permeate real world decision-making, whether in everyday personal and working life, or in policy-making and politics. With the growing demand for awareness and participation at both individual and social levels, with people expected to act as citizens who are accountable for their decisions, an increasing number of choices relating to fields from health and lifestyle to transport to national and international politics involve the need to assess and take some risks.

In fact, the importance of the discourse of risk in public policy is now being reflected across various curricula in England. For example, Personal, Social, Health and Economic (PSHE) education becomes a compulsory part of the curriculum in 2011 and students will study risk as part of that curriculum. The PSHE education association asserts on its website (www.pshe-association.org.uk/):

"Risk-Taking introduces for students the distinctions between positive and negative risk; likelihood and severity… and risk perception… legal and illegal drugs, sex, gambling and anti-social behaviour."

In the Mathematics National Curriculum, 2007 (http://curriculum.qcda.gov.uk/uploads/QCA-07-3339-p_Maths_4_tcm8-404.pdf) in England, teachers are expected to consider situations that involve risk and uncertainty, portraying risk as an adjunct to probability as captured in the phrase: "applying ideas of probability and risk to gambling, safety issues and the financial services sector, and simulations using ICT" (p. 9). The curriculum in mathematics does not make a distinction between probability and risk. The Science curriculum in England is more ambitious and recognises the central importance of considerations of risk in socio-scientific issues and innovative curriculum programmes, such as Twenty-first Century Science (www.21stcenturyscience.org/) have developed extensive materials to support that area of teaching. The attainment target on "Making Science Work" refers to probability and consequence, cost versus benefit, the precautionary principle as well as relative and absolute risk. However, there has so far been almost no principled articulation of the pedagogy of risk to inform the further development of the content.

The contested nature and different perceptions of risk

The essential meaning of the idea of risk is contested and the subject of a diversity of epistemological viewpoints (Adams, 1995; Stirling, 1999). In the media, risk is often quantified and treated as being identical to likelihood (expressed as 1 in n). In other cases, risk is closely associated with the hazard in question: for example, in the London Underground signs reading "Danger: Risk of Death" call our attention to the seriousness of the hazard, making the consideration of likelihood redundant. In complex situations, however, both likelihood and impact need to be addressed simultaneously and trade-offs need to be considered.

In support of what has become standard theory in domains such as Economics, Campbell (2005) has demonstrated through a philosophical argument that a rational view of risk should incorporate both the dimensions of the likelihood of the hazard occurring and the impact should the hazard occur. Standard decision-making theory formulates risk as the product of the probability and disutility (a number quantifying the harm that might ensue from the occurrence of a hazard). However, authors such Tversky and Kahneman (2002) point to the difficulty of assigning a value for probability. The "correct" probability of events is not easily defined, and since "individuals who have different knowledge or hold different beliefs must be allowed to assign different probabilities to the same event, no single value can be correct for all people" (p. 19). The same difficulty applies to the quantification of impact.

The discourse of risk sometimes refers to *actual* risk, as measured and analysed by 'experts' using scientific and mathematical methods, and *perceived* risk, as articulated by the public at large and vulnerable to bias (Kahneman & Tversky, 1979; Kahneman et al, 1982; Stirling, 1999; Sztompka, 1999). Such apparent fallibility explains the experts' rather disdainful view of perceived risk in the discourse on risk. An alternative explanation is that individuals judge risk from a rational perspective in which the data are often tacit and personal (Slovic, 1986); Irwin & Wynne (2006) distinguish between rational expert and rational lay estimates. Estimations of impact are subjective and may be based on sources of information that are unknown to the expert and, in a rational way, lead to a different inference about risk.

On the other hand, the literature contains many studies, which show that judgement of chance is often guided by misleading intuitions, which are rooted in inadequate cognitive heuristics (Konold, 1989; Lecoutre; 1992). Peters (2008) calls attention to how individuals who differ in number ability perceive and use numeric information about risk differently, arguing that highly numerate individuals translate numbers into meaningful information and use them in decisions, in contrast with less numerate people who use other non-numerical sources of information, such as their emotions and their trust or distrust of science, policy-makers and experts. Slovic et al (2000) also highlight how people can read the same information differently when it is given in absolute numbers ("20 out every 100 cases will…") or expressed as a relative risk ("there is a 20% chance that…").

Many other factors also seem to affect how risk is perceived (Kahneman & Tversky, 1979; Campbell, 2006). People tend to perceive the risk of dying in an aeroplane accident as higher than the risk of dying in a car accident; the former can be classified as a more concentrated risk (since many people would simultaneously suffer the consequences should the accident occur). Additionally, risks that are voluntary (that are assumed following deliberation) or associated with benefit from the observer's point of view may often be perceived as lower. Peters et al. (2004) also highlight the importance of deliberation. Research by Finucane et al. (2000) has shown that less deliberation increases the inverse relationship between perceived risks and benefits.

Although citizens in industrialised societies are more affluent and longer-living than their antecedents, concerns arising from public mistrust of those institutions responsible for political decision-making (O'Neill 2002) have prompted personal anxieties and fears in the so-called 'risk society' (Beck 1992; Levinson, 2010), especially when the level of participation of the population is considered low. In this scenario there are pressures for a zero risk approach at the individual level, which reinforce demand for governments to apply the *precautionary principle* at the policy level. *Zero risk bias* can be attributed to a desire for *cognitive closure*, which Webster and Kruglanski (1997) describe as a desire for definite knowledge and the eschewal of ambiguity (p. 133).

Our approach

The fact that risk is regarded as part of PSHE, Citizenship, Mathematics and Science curricula indicates its fundamentally cross-curricular nature. School infrastructures are not well suited to handling issues that do not fall comfortably into one discipline. In our project, Promoting Teachers' Understanding of Risk in Socio-Scientific Issues (TURS[1]), we worked with inter-disciplinary pairings of Mathematics and Science teachers, each pair from the same secondary school. Our aim was to co-design software[2] about risk and about the teaching and learning of risk. We intended that the emerging software would act as a window on the teachers' thinking-in-change (Noss and Hoyles, 1996) by perturbing, making explicit and sharing ideas with each other and with the researchers in what was essentially a design research approach (Cobb et al,

---

[1] The funding of the Wellcome Trust is gratefully acknowledged (WT084895MA). Project website: www.RISKatIOE.org.

[2] The prototype was developed in Imagine Logo, an object-oriented parallel-processing version of Logo that allows the programmer many interface design options. It is published by Logotron: http://ns.logotron.co.uk/imagine/

2003). It is not our intention in this paper to present data from the teachers' activity but instead to reflect on the rationale for the design of the emergent software.

Deborah's Dilemma

The design research approach has so far resulted in software called Deborah's Dilemma. In this section, we set out a plain description of the software, leaving the explanation of the rationale for that design, based as it is on our interactions with the teachers, to the later sections of the paper. The reader might enjoy trying to anticipate the design rationale as they read this account.

Setting the scenario

We propose an imagined scenario in which a young woman, Deborah, suffers from a chronic back condition. There is available an operation, which might cure the problem or give rise to further complications, some of which might be regarded as relatively trivial and others, such as paralysis, that threaten Deborah's future quality of life. The teacher is challenged to judge how they would react in Deborah's situation and how they would advise Deborah.

A good deal of information is available, either through text or through talking-head videos, about the situation. For example, in describing the impact of the condition on her sporting activity, Debora explains: "I used to practice several sports which I can no longer pursue due to the stresses they put on my back. I am an adventurous person and enjoy kayaking, I have even tried hang-gliding. I like jogging but it is a high impact sport, which can jolt the vertebrae and cause even more damage. Gentle exercise such as swimming, yoga and Pilates actually helps to reduce the pain. Muscle-strengthening exercises help to keep the pain at a tolerable level." Similar descriptions are given about the medical condition and how it affects her working life.

Substantial information is also given about the operation that could be carried out. Deborah in fact is described as having three separate consultations and conducting personal research on the internet. This information yields different views about the likelihood of success and the possible complications that might happen. The teacher is expected to resolve these discrepancies and contradictions in discussion with other teachers.

The information about the condition can be used to model possible consequences of having the operation. The information about Deborah's attitudes and life-style can be used to model consequences of not having the operation. The modelling tools that are used in each case are described below.

Modelling the consequences of having the operation

The teacher is challenged to model the consequences of having the operation. Figure 1 illustrates the tool used to respond to this challenge. In Figure 1, the teachers have used the slider in the top left hand corner to set an overall probability of success for the operation of 0.7. They have used the 'Add Complication' button to include nerve damage, paralysis and superbug infection as three possible complications. In each case, they have used the corresponding sliders to set likelihoods for these complications. These likelihoods are overall probabilities and so cannot occur more often than failed operations. Two or more complications can occur in the same operation. Successful operations have no complications.

Towards the top of the Figure 1 the teachers have edited the number of times the simulation is run to 1000. We think of this as 1000 futures for Deborah. The results can be shown as a bar chart though we think the representation shown is more informative. The chart is colour-coded so that successful operation are shown in green, unsuccessful ones in red, and operations that are unsuccessful but have complications in stripes according to the range of complications. This representation enables the teachers to eyeball the whole set of Deborah's futures and gain a proportional sense of the various possible outcomes. The teachers might instead prefer to run the simulation once on the basis that Deborah has only one life and therefore we provide a ‘run once’ button towards the top of Figure 1.

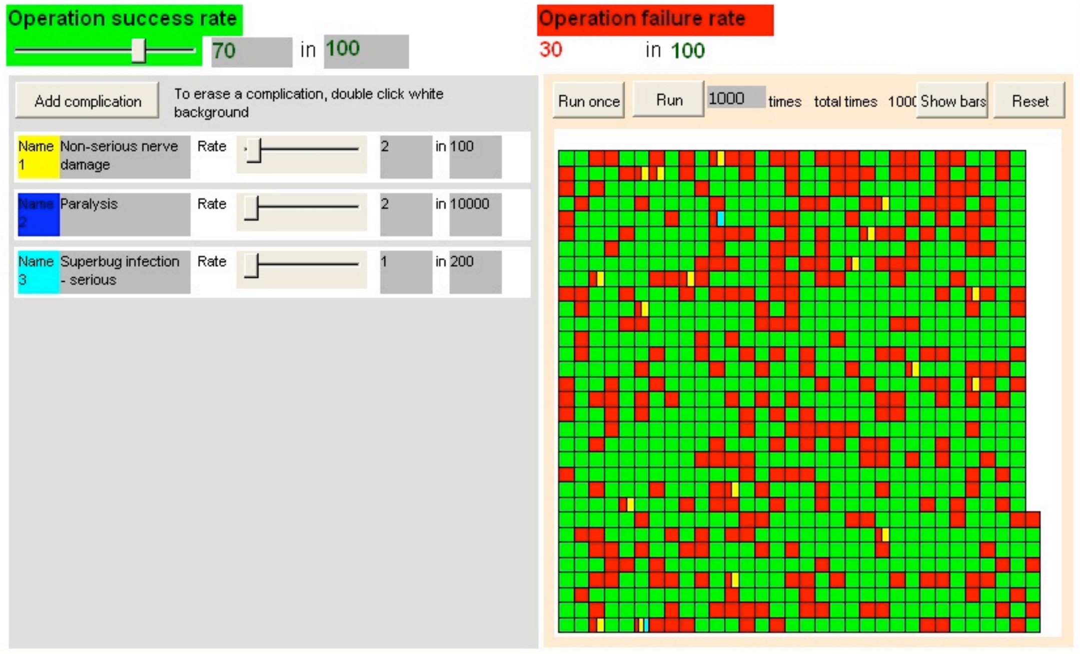


Figure 1: Two teachers model the consequences of having the operation

Modelling the consequences of not having the operation

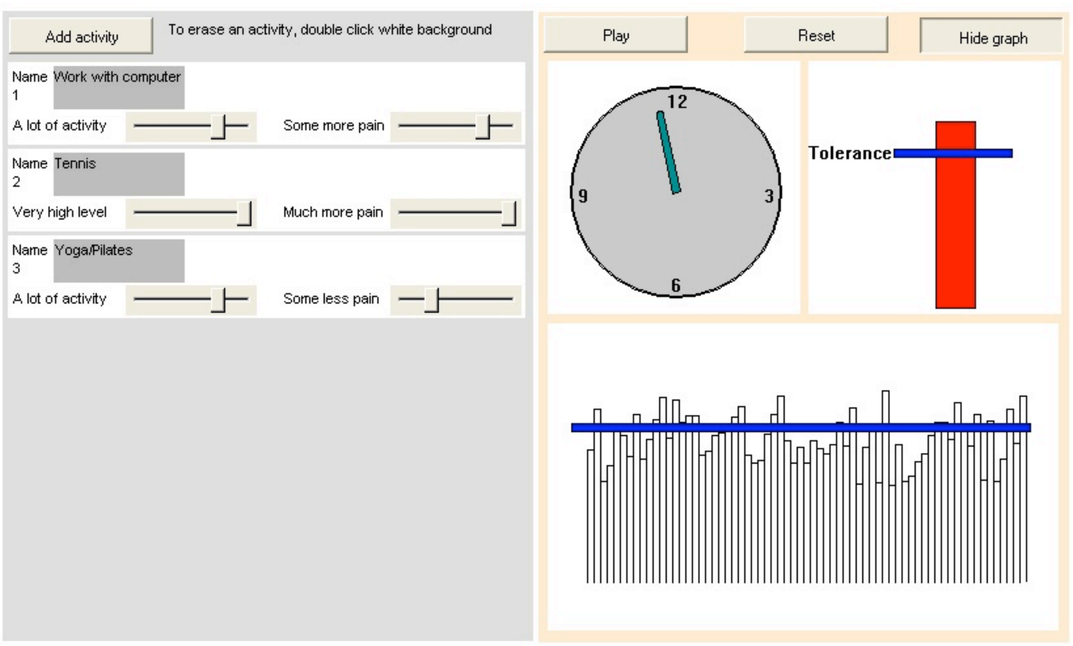


Figure 2: Two teachers model the consequences of not having the operation

The teacher is challenged to model Deborah’s lifestyle. Figure 2 illustrates the tool used to respond to this challenge. In that, the teachers have used the ‘Add Activity’ button to include computer work, tennis and yoga as three aspects in a model of her life-style. In each case, they have judged how much of the activity is or might done by her and how much pain the activity might cause. When the ‘Play’ button at the top of the screen is pressed, the clock rotates and the red bar (the *painometer*) oscillates, indicating in time how much pain she is suffering. Teachers can set the level of tolerance they think is appropriate. The graph shows a trace of the pain level.

Comparing risks

Throughout the process of modelling the consequences of having or not having the operation, the teachers have been encouraged to keep a map of what they see as the possible resulting hazards. In Figure 3, two teachers have built up such a map using the 'Add Hazard' button, and entering information such as likelihoods, impacts or other perhaps value-based information.

When the teachers press the 'Show Risk' button in the bottom right hand corner of Figure 3, the boxes will change colour. Boxes towards the left of the screen will become darker while those to right will become lighter on a continuous scale. The teachers will be told that hazards with darker colours have higher risks and those with lighter colours have lower risks. Inevitably, the teachers will now judge that some of the boxes are in the wrong position on the screen. They are able to drag the boxes to what they judge to be the correct relative position according to their risks. In doing so, they will of course refer to their judgements about impact, likelihood and other information as entered by them into the boxes.

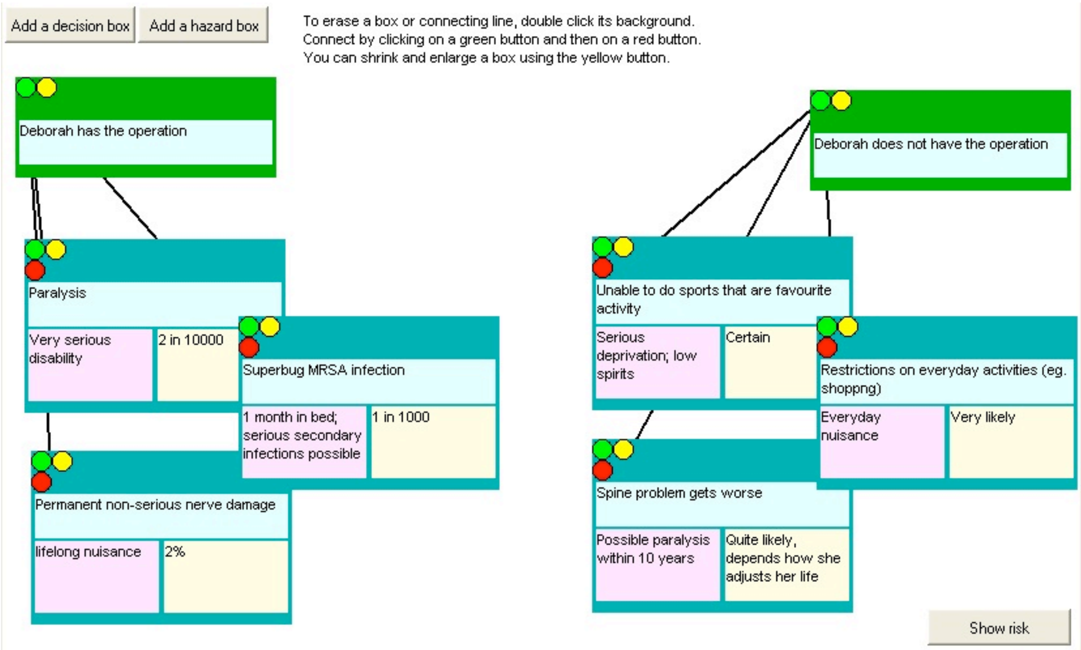


Figure 3: Two teachers map out key information before exploring the size of the relative risks

Design rationale

In the previous section, the software has been described in some detail. This section will now reflect on how the design decisions have been influenced by the need to develop pedagogical principles around knowledge that is not only uncertain but also contested.

Decision-making in complex scenarios

Our aim was to use the process of designing software to gain access to the teachers' knowledge about risk including its teaching and learning. We were though forewarned by the literature, as described above, about the deep sensitivity of knowledge about risk to context and it seemed that our aim of probing teachers' knowledge of risk would be undermined by an approach that separated the tool from a problem context. Indeed, we took the opposite view in which we intentionally embraced context.

Since risk is a tool that can be used in many decision-making contexts, it seemed appropriate to create a situation that was sufficiently life-like to provoke intuitive ideas that the teachers might bring to bear consciously or unconsciously in making such decisions. Pratt (1998) has referred to the use of context to stimulate relatively natural intuitions as *surface familiarity.* The attraction of such an approach to us as researchers was that the teachers might engage deeply with the scenario and that their activity might be driven by a sense of purpose and curiosity rather than by what they perceived to be the needs of the researchers. Our aspiration was that teachers' long-term commitment would facilitate the observation of teachers' utilities for risk (Ainley et al, 2006); in other words, we expected to identify how teachers thought risk might be used to make sense of such a scenario.

In fact, the real value of risk seems to be in supporting judgements in those contexts. We wanted to understand the nature of teachers' judgements. We therefore took the design decision to develop a complex scenario, which called for difficult judgement in the face of uncertainty and ambiguity. The scenario we developed sets up tensions between consideration of severe complications in the operation, such as paralysis, with low likelihoods of occurring, and life-style compromises that might be unacceptably debilitating, such as giving up work or sport. We tried to provide ambiguous and sometimes contradictory data, such as the conflicting opinions of various doctors and consultants and the outcomes from personal research on the internet about the types of complications that could occur and their likelihoods.

We have some limitations. In tying the exploration to one particular scenario, we are only able to observe the teachers' expressions as they relate to that particular situation, but nevertheless, we designed with the teachers software that:

1. Addressed a scenario with surface familiarity in order to expose an intuitive layer of knowledge about risk;
2. Was sufficiently complex that teachers would need to exercise judgement, exposing the nature of their thinking;
3. Incorporated ambiguous and conflicting information that would throw light on how the teachers weighed such evidence.

Making personal models explicit and computational

As designers, we faced our own dilemma. We described at the beginning of this paper the metaphor of planting nuggets of mathematical knowledge in the microworld with the intention that the user of the microworld would stumble across those nuggets. We noted the literature's ambiguous position on the nature of risk and pondered the design approach when there were no well-defined non-contradictory nuggets to plant.

We came to the view that the role of the software should be to expose what the teachers' imagined to be the nature of risk, what we refer to here as their personal models of risk. It became clear that our approach should emphasise expressions of risk as teachers interpret the scenario. However, the models that might be expressed by teachers could easily remain at a descriptive level. We aimed to push the teachers towards developing computational models that could be executed to generate feedback. Computational models require a level of explicitness, precision and unambiguity that is often not achieved by descriptive models that can be left unchallenged in their vagueness. We expected that feedback available from executing personal models might lead the teachers to re-evaluate their own thinking about risk. We also expected that by creating models, their ideas would be exposed to evaluation by colleagues, whose own models might also have a perturbing effect.

Thus, Deborah’s Dilemma contains tools to express personal models about the operation and its consequences. These are expressed through the creation of complications. Which complication to include in the model is for the teacher to decide. Each complication can be given likelihoods of occurrence. What level of likelihood is for the teacher to decide. The model can be run as many times as is felt necessary by the teacher. Similarly, Deborah’s Dilemma contains tools to express personal models about the consequences of not having the operation. These are expressed through the creation of activities. Each activity can be given levels of how much activity Deborah might do and how much pain is caused. Teachers make judgments about all of these factors. The model can be run to observe the fluctuating nature of Deborah’s pain resulting from those life-style decisions.

Inevitably the modelling tools we provided shaped and constrained the teachers’ expressions of risk, but nevertheless, we designed with the teachers software that:

4. Included tools to facilitate the expression of personal model of risk;
5. Provided feedback on the consequences of a personal model by making the models computational rather than only descriptive;
6. Encouraged sharing of personal models by making them explicit and open to scrutiny.

Fuzzy quantification

A particular aspect of making personal models explicit and computational was quantification. It was clear that risk should at least incorporate dimensions of likelihood and impact but how should we encourage their quantification without imposing a particular model of risk?

One unlikely but possible complication of Deborah’s operation was death as an unfortunate effect of the anaesthesia. Many teachers might regard death as the most severe of possible consequences and look to set a very large, even infinite, value for its impact. An infinite impact could really only be offset by an infinitesimal likelihood, a mathematically untenable situation.

We also observed in working with the teachers a good deal of discomfort in trying to set specific values on impact. Since we were committed to the idea of the teachers creating computational models, we decided to deploy techniques, which we have labelled *fuzzy quantification*. This approach is most evident when modelling Deborah’s life-style. Sliders are available to indicate how often Deborah might engage in any particular activity and what the impact on pain level might be. Categories like ‘much more pain’ and ‘less pain’ serve the purpose of fuzzy quantification. Pain itself is a vague notion, which it is claimed cannot be quantified in an objective manner as each person experiences pain in their own idiosyncratic manner. Nevertheless, we incorporated the notion of a painometer as a fluctuating bar with no specific scale and tolerance as a moveable threshold on how much pain Deborah could manage in order to partially quantify the consequences of the decisions made in modelling Deborah’s life.

In short, we designed with the teachers software that:

7. Enabled personal models to be computational through fuzzy quantification of impacts;
8. Encouraged simultaneous consideration of impacts and likelihoods through fuzzy quantification of impacts;
9. Facilitated an appreciation of the consequence of a particular personal model of Deborah’s life-style by the invention of a non-standard fuzzily quantified representations of pain and tolerance.

Facilitating co-ordination of dimensions of risk

Since risk contains at least the dimensions of impact and likelihood, it seemed important to consider both and at times to trade-off severity of impact against likelihood of occurrence. As we worked with the teachers, it became increasingly transparent that they struggled with attending simultaneously to likelihoods and impacts. For example, there was a tendency for them to come to one recommendation for Deborah when considering the low likelihoods of complications and another when considering the high impacts of some of those complications. We therefore needed to devise tools that might support the co-ordination of the various dimensions of risk without imposing a particular model of risk. The intention was not to lead the teachers towards a particular view of the nature of risk but to explore whether it was possible to devise tools that might enable risk to be seen as a single entity so that different hazards might be compared.

The mapping tool (Figure 3) was a late development in the software. Because there is a very large amount of information that described Deborah, her life, her condition and the operation, it was felt appropriate to provide a tool that the teachers could use simply to record what they regarded as key information. The provision of decision boxes, hazard boxes and connecting lines enables the teachers to structure that information. The innovation lays in the further provision of the risk button that colours the boxes according to level of risk, which can be altered by moving the boxes around the screen. The teachers' co-ordination of the dimensions of risk, as summarised in their notes within each box, is challenged and exposed by their actions and discussion stimulated by the need to move the boxes to an acceptable level of risk when compared to the other boxes. We observed how this activity threw up explicit mention of the need to balance impact and likelihood without the need for formal quantification such as when multiplying probability and impact. We believe that this concept-mapping tool is a first instantiation of a tool that supports the co-ordination of the dimensions of risk but we look forward to finding different solutions in the future. In short, we designed with the teachers software that:

10. Provided a tool for the co-ordination of the dimensions of risk through the fuzzy quantification of risk when comparing different hazards.

Conclusion

Although we have discussed the difficulty of designing a microworld when the focal knowledge is not only highly subjective but also contested by experts, we have been able to exploit the idea that one purpose of a microworld is to provide a window on activity for the users and for the researchers. In so doing, we are aware of various ways in which the Constructionist literature has informed the design process that has been summarised in the preceding section in the form of ten design outcomes.

The decision to construct a complex scenario was largely based on the need to offer a purposeful task that could lead to the exposure of utilities for risk. It is well established in the Constructionist literature that technology can afford a design process of phenomenalisation (Pratt et al, 2006) in which mathematical or scientific ideas become quasi-concrete objects (Turkle & Papert, 1991), capable of on-screen manipulation in ways that parallel the exploration of material objects in the lived-in world. The development of Deborah's Dilemma has involved the creation of on-screen instantiations of constructs of probability and impact in the form of sliders and risk as the colour of moveable hazard boxes. By utilising these quasi-concrete objects, the teachers grappled with Debora's Dilemma and exposed new understandings of risk. We see here Papert's Power Principle (1996) at work in the sense that the teachers were using risk even as their knowledge about risk was being transformed. Indeed, in developing the risk button in the mapping tool, we exploited the design heuristic that a window on a mathematical or scientific construct can be built by making a representation of the key idea, risk in this case, a central control over activity (Pratt et al, 2006).

The modelling that the teachers conducted in working with Deborah's Dilemma was primarily about expressing their own ideas for wider scrutiny. This contrasts to some extent with the common use of microworlds that embeds the mathematical or scientific idea for it to be explored. Although it is true that in Deborah's Dilemma, the notion of risk is phenomenalised, it is not unambiguously defined. The defining process that sensitises the teachers to the various dimensions of risk is more attuned to expressive modelling than exploratory modelling (see Doerr & Pratt, 2008) and we see this as a key consequence of designing a microworld where the key knowledge is contested. Papert makes a distinction between the nature of knowledge and the nature of knowing arguing that the first is a technical matter that belongs to educational school course and the second is epistemological. In exploring risk, it seems that both the nature of knowledge (contested and not tightly defined) and the nature of knowing are under scrutiny.

References

Abrahamson, D., & Wilensky, U. (2007). Learning axes and bridging tools in a technology-based design for statistics. *International Journal of Computers for Mathematics Learning, 12*(1), 23-55.

Adams, J. (1995). *Risk*. London: UCL Press.

Ainley, J., Pratt, D. and Hansen, A. (2006). Connecting engagement and focus in pedagogic task design, *British Educational Research Journal*, 32.1, 23-38.

Beck, U. (1992). *Risk society: Towards a new modernity*. Sage, London.

Campbell, S. (2005). Determining overall risk. *Journal of Risk Research* 8(7-8), 569-581.

Cobb, P., Confrey, J., diSessa, A., Lehrer, R. and Schauble, L. (2003). Design experiments in educational research. *Educational Researcher*, 32(1), 9–13.

Doerr, H. and Pratt, D. (2008). The learning of mathematics and mathematical modeling. In M. K. Heid and G. W. Blume (Eds.), *Research on technology in the teaching and learning of Mathematics: Syntheses and perspectives: Mathematics learning, teaching and policy v. 1*, pp. 259-285. Charlotte, NC: Information Age.

Finucane, M. L., Alhakami, A., Slovic, P. and Johnson S. M. (2000) The affect heuristic in judgements of risk and benefits, *Journal of Behavioral Decision Making,* 13, 1-17.

Irwin, A., & Wynne, B. (Eds.). (1996). *Misunderstanding science? The public reconstruction of science and technology*. Cambridge: Cambridge University Press.

Kahneman, D. & Tversky, A. (1979). Prospect theory: An analysis of decision under risk, *Econometrica*, 47(2), 263-292.

Kahneman, D., Slovic, P. & Tversky, A. (1982). *Judgment under uncertainty: Heuristics and biases*. Cambridge University Press, Cambridge.

Konold, C. (1989). Informal conceptions of probability. *Cognition and Instruction,* 6, 59-98.

Lecoutre, M. P. (1992). Cognitive models and problem spaces in “purely random” situations. *Educational Studies in Mathematics,* 23, 589-593.

Levinson, R. (2010). Science education and democratic participation: An uneasy congruence? *Studies in Science Education, 46*(1), 69-119.

Noss, R., & Hoyles, C. (1996). Windows on mathematical meanings: Learning cultures and computers. London: Kluwer Academic Publishers.

O'Neill, O. (2002). *Autonomy and trust in bioethics*, Cambridge: Cambridge University Press.

Papert, S. (1982). *Mindstorms: Children, Computers and Powerful Ideas*. London: Harvester Press.

Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning*, 1(1), 95-123.

Peters, E.M., Burraston, B., Mertz, C.K. (2004). An emotion-based model of risk perception and stigma susceptibility: Cognitive appraisals of emotion, affective reactivity, worldviews and risk perceptions in the generation of technical stigma, *Risk Analysis*, 24(5), 1349-1367.

Peters, E. (2008). Numeracy and the perception and communication of risk, *Annals of the New York Academy of Sciences*. 1128, 1-7. Retrieved January 28, 2008 from: http://www.uoregon.edu/~icds/Evolution_FG_files/Peters_Numeracy.pdf

Pratt, D, (1998), The construction of meanings IN and FOR a stochastic domain of abstraction, *Unpublished Doctoral Thesis*, Institute of Education, University of London, May 1998.

Pratt, D. & Noss, R. (2002). The micro-evolution of mathematical knowledge: The case of randomness, Journal of the Learning Sciences, 11.4, 453-488.

Pratt, D. Jones, I. & Prodromou, T. (2006). An elaboration of the design construct of phenomenalisation. Proceedings of The Seventh International Conference on Teaching Statistics, Salvador, Bahia, Brazil.

Slovic, P. (1986). Informing and educating the public about risk. *Risk Analysis 6*(4), 403-415.

Slovic, P., Mohanan, J., MacGregor, D. C. (2000). Violence risk assessment and risk communication: The effects of using actual cases, providing instruction, and employing probability versus frequency formats, *Law and Human Behavior*, 24(3), 271-296.

Stirling, A. (1999). *On science and precaution in the management of technological risk: A synthesis report of case studies*. Seville: Institute for Prospective Technological Studies (European Commission Joint Research Centre). Retrieved December 18, 2008 from: ftp://ftp.jrc.es/pub/EURdoc/eur19056en.pdf.

Sztompka, P. (1999). *Trust: A sociological theory*. Cambridge: Cambridge University Press.

Turkle, S., & Papert, S. (1991). Epistemological pluralism and the re-evaluation of the concrete. In I. Harel & S. Papert (Eds.), *Constructionism* (pp. 161-192). Norwood, New Jersey: Ablex.

Tversky, A. Kahneman, D (2002). Extensional versus intuitive reasoning: The conjunction fallacy in probability judgment, in *Heuristics and Biases – The Psychology of Intuitive Judgment*. Cambridge: Cambridge University Press.

Webster, D. M. & Kruglanski. A. W. (1997).Cognitive and social consequences of the need for cognitive closure. *European Review of Social Psychology*, 8,133-173.

Wilensky, U. (1991). Abstract meditations on the concrete and concrete implications for mathematics education. In I. Harel & S. Papert (Eds.), *Constructionism* (pp. 193-204). Norwood, New Jersey: Ablex.

Wilensky, U. (1997). What is Normal anyway? Therapy for epistemological anxiety. *Educational Studies in Mathematics*, 33, 171-202.

Music in Introductory Object Oriented Programming

Viera K. Proulx, *vkp@ccs.neu.edu*
College of Computer and Information Science, Northeastern University, Boston, MA , USA

Abstract

Our Java-based *idraw* library has been designed to give a novice Java programmer the tools to design a simple interactive animated graphics-based game. The programmer focuses on the design of the game behaviour and representation of the game scene in terms of simple shape-based graphics. It has been used by hundreds of students over the past several years. This paper presents the Java *isdraw* library that extends our *idraw* library by giving the programmer tools to add musical effects to their game.

The new library provides the opportunity for students to practice working with sequences of data, designing loops, and designing classes that represent musical phrases, melodies, chords, and effects. It also illustrates the connection between the information and its representation as data. Of course, the addition of musical effects to an interactive game is a great motivator, making the learning more concrete and challenging.

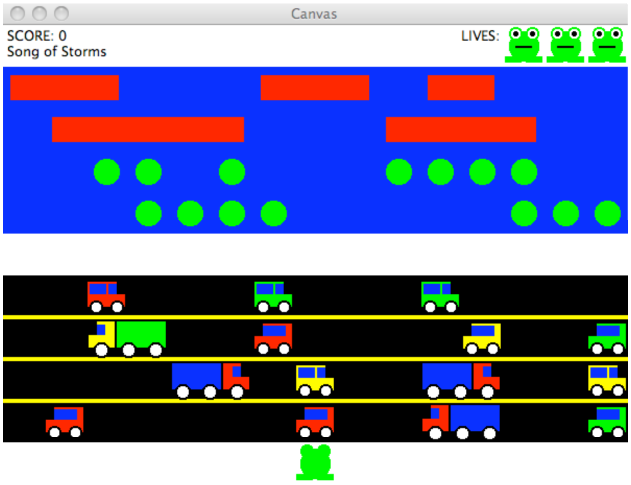


Figure 1. Frogger game designed with isdraw library.

Our pedagogy enforces systematic unit testing from the beginning. The design of the library that supports musical and sound effects makes it possible for students to play the tunes before embedding them in the game, and design tests for their sounds prior to playing them. Our goal is to combine constructive exploration with structured design discipline.

Keywords (style: Keywords)

Introductory computing; object-oriented programming; games and music; constructionism

Introduction

The *TeachScheme/ReachJava* curriculum focuses on introducing students to solid program design principles from the beginning. Several software artefacts that allow the student to focus on the key design concepts support the curriculum. At the beginning the teaching languages within the *DrScheme* programming IDE allowed student to begin programming without learning complicates language syntax, and with the ability to evaluate small program segments interactively. Over the years, the support for the first part of the curriculum, known as *TeachScheme!* has grown to include support for unit testing, support for the design of graphics-based interactive games, and, most recently, support for distributed program design with several clients communicating with a server. The most important feature of all these *teachpacks*/libraries is the ease of programming --- requiring only the basic programming skills.

Over the past eight years, we have been developing the next part of this curriculum, focusing on the program design in object-oriented style for class-based languages. Our curriculum starts with a simple Java-like language *NeuJava*, and progresses to the standard Java language as students see the need for language support for increasingly more sophisticated design of abstractions and libraries. Besides providing the language environment, over the years we have designed a *tester* library that supports unit testing within the constraints of the language knowledge of a novice student. This library is now used extensively in introductory courses, even when the instructors do not follow our curriculum. We have also designed and used extensively libraries that make it possible for students to design graphics-based interactive games, using only the very basic language skills --- working only in the mutation-free *NeuJava* language (the *draw* library). The more advanced version of these libraries uses imperative style and allows students to convert their games into Java Applets by adding with only a small wrapper class to initialize the applet (the *idraw* and the *adraw* libraries). But except for using the imperative programming style the library still asks the programmer to only provide the game model behaviour and the representation of the game's graphics as a simple shape-based graphics drawing.

Today's games nearly always include musical background and sound effects. We describe our new library that allows the beginner programmer add musical background and sound effects to their games. Additionally, the new sound library also provides a context for experimentation with musical phrases, and combines learning about music with learning to design programs that deal with sequences, and a variety of ways sequences can be manipulated and combined. The two versions, *isdraw* and *asdraw* are again targeted to Java Applications or Java Applets

The Tune Bucket and The World Library

The *idraw* library defines an abstract class `World` that builds a frame with a `Canvas` for the game display and declares several abstract methods that the student needs to implement. There is `onTick` method that represents the game action on each tick of the clock, the `onKeyEvent` method that represents the action in response to the different key presses, the `draw` method that defines the game display from the current state of the user's world, and a `bigBang` method that defines the initial world, the clock speed, and the size of the game `Canvas`. In a typical game, some objects move on each tick, some objects are controlled by the key presses, and the game ends when some object collide, or run out of lives.

The *isdraw* library adds support for sound effects and music exploration. An English saying describes musically inept people as those who cannot carry a tune. The whimsical response is that I can carry a tune - in a bucket. So, for the programmer with only a limited musical knowledge, we provide two buckets (instances of the class `TuneBucket`) for carrying the tunes: the `keyTunes` bucket for defining which notes are to be played in response to the key presses and the `tickTunes` bucket for defining the tunes to be played at each tick of the timer.

Originally, the library provided a list of constants that represents the pitches over three octaves. A more sophisticated musician would use the actual MIDI pitch codes directly. The two buckets get filled with an arbitrary collection of notes to be played during the event handling method. If the programmer wants to make sound when a key is pressed, he includes in the `onKeyEvent` method a call to `keyTunes.addNote` method that consumes as arguments the instrument to play and pitch of the note to play. Each key press plays the tune for one-quarter-note duration. In a similar manner, invoking the `tickTunes.addNote` method within the `onTick` method plays the given note when the `onTick` method is invoked. The notes added to the `tickTunes` bucket are played until the next tick event. We expected students to compose the tune as a list of notes and select the next note in the list on each tick.

Initially, this seemed to be a very primitive design, having no provisions for the duration of the tunes, little provision for repeating a theme, or for playing more sophisticated sounds. We let our students use the first prototype of the library in the fall 2009. A programming pair that included a musically talented student produced a *Frogger* game with an exciting jazzy background music that made the game quite impressive. Their code included several extensive lists of tunes; with one form each list to be played at each tick. They figured out how to represent the tempo, they defined a method that transposed the tune and/or scaled, it. Their work has been an inspiration to add tools and abstractions to our library so that both, musically talented, and musically challenged students would be able to add sound effects to their games.

During the Spring 2010, we were still using still the basic version of the library, but we gave the students several examples of what kind of sounds and music they can construct from the given building blocks. Many students explored the ways of generating polyphonic melodies, adding sound effects at the critical points during the game, annotating their code with auxiliary representation of the music data. A student with a serious interest in music constructed a sequencer application. The composer could add notes to a grid display where the height of a coloured marker represented the pitch, the horizontal axis represented the time, and the colour of the marker defined the instrument that should play the note. When the play mode was turned on, a thin vertical line moved across the grid and all notes that the line crossed were played.

This confirmed our prediction that a tool like this will inspire students to explore and construct interesting sound sequences and music --- all while learning the basic skills of writing programs that combine sequences of data into more complex structures.

Making Music: Evolution of the Library

Playing notes

The harpsichord music from the Baroque era had no provision for playing the note for extended duration. The playing of a sequence of quarter noted followed by a half note was accomplished by the timing of the initial key presses. Initially, we have adopted a similar technique for representing the note duration. We illustrate this on an example. The following sequence represents the first four lines of the Frére Jacques tune:

```
noteC,0,0,0,noteD,0,0,0,noteE,0,0,0,noteC,0,0,0,

noteC,0,0,0,noteD,0,0,0,noteE,0,0,0,noteC,0,0,0,

noteE,0,0,0,noteF,0,0,0,noteG,0,0,0,0,0,0,0,

noteE,0,0,0,noteF,0,0,0,noteG,0,0,0,0,0,0,0
```

Three silent ticks follow each quarter note; seven silent ticks follow a half note. etc. The constants `noteD`, `noteG`, are the names for the corresponding pitches. That means that the number of silent notes after the note is played represents the duration of the note. Surprisingly, this allows for constructing interesting and amusing musical sequences.

Representing notes

After our initial experiences with students it became clear that this tool could be extended to support extensive exploration of musical features and constructions without compromising our design-driven pedagogy of programming instruction. Additionally, we believe, the design of the library can serve as a model of different techniques of program design.

Our first challenge was to design a clean and robust representation of the MIDI notes. From students; point of view this is a wonderful example of multiple representations of information as data. One of the representations consists of the `pitch` and the `duration` of the note, another one specifies the note name (e.g. F), its `modifier` (sharp, flat, or natural), the `octave` on the piano keyboard, and the `duration` of play. Rather than defining methods that perform the conversion between the different representations, we defined a `Note` class with several constructors, each accepting a different representation of the note, but each of them initializing not just the `pitch` and `duration` field, but also the note name field, the `modifier`, the `duration`, and a field `snote` that records its representation as a `String`. So, a note representing the middle C, playing for 2 beats can be defined in either of the following ways:

```
Note c4n2V1 = new Note("C4n2");

Note c4n2V2 = new Note(60, 2);
```

The need for multiple representation of the same information and the use of constructors to make this possible is illustrated here in a compelling way.

Extending the Tune Buckets

Initially, the `tickTunes` and the `keyTunes` tick buckets limited the duration of each note to one tick. To make it possible for a note to be played over several time ticks, we needed to modify the design and the behaviour of these `TuneBucket`-s. Furthermore, in the initial version the programmer had to add the notes to the `TuneBucket` one at a time.

Music is a wonderfully complex time sequence. The MIDI synthesizer allows us to play up to 16 instruments at a time, and play on many of the instruments a polyphonic melody (a chord). Our goal in extending the library was to allow the musically gifted student to work with as many features of the MIDI interface as possible, yet keep intact the original simple setup for those that are musically challenged, or do not want to create elaborate musical structures.

Our new note representation includes the `duration`. We decided to limit the granularity to 1/16th note playing for one tick. So, a note of `duration` 2 is 1/8th note, note of `duration` 4 is a ¼ note. Of course, the actual time needed to play one note is set when we start the timer and specify the rate at which the time events should happen.

To capture the timing information and act on it, we added two new `TuneBuckets` to our `World`, and added method `nextBeat` to the `Note` class that simulates playing of the note for one beat by decreasing its `duration`. The `currentTickTunes` and `currentKeyTunes` `TuneBuckets` represent the list of notes currently playing. Instead of stopping the playing of all notes that started on the previous tick, only those whose `duration` has decreased to 0 are stopped. All notes added to the tune bucket on a tick or key event start playing, and are then moved to the current buckets. These are advanced to the next beat on each tick; the notes that fell silent are stopped and removed from the current buckets.

Designing Tunes

One can think of music as being a collection of scores, one for each instrument that needs to be played in a synchronized sequence. To model this, we designed a `Tune` class that represents one time event for an instrument. It includes a field that identifies the channel on which to play (or the instrument we wish to play) and a `Chord` --- a collection of `Note`-s to play. We then allow the programmer to add notes to the `TuneBucket` in any of the following ways: a single note, a single note given only by its name as a `String`, a single pitch (which then plays for one beat), a `Chord`, a `Tune`, or an `Iterable` collection of `Tune`-s, or an entire `TuneBucket`. This provides the flexibility for how the programmer organizes the musical sequences.

The `TuneBucket` now contains a collection of 16 `Tune`-s --- one for each channel in the current MIDI program. New notes, chords, and tunes are added to the `Chord` associated with the `Tune` for the corresponding instrument.

Playing the notes and instruments

To allow playing the music apart from the interactive game and to promote further exploration of the musical structure, we added a `MusicBox` class. It initializes the MIDI synthesizer to a default program, or to the program given by the programmer, and provides the method to play or stop the given `Tune` or a collection of `Tune`-s. The game `World` class then uses an instance of the `MusicBox` to play and stop the `Tune`-s in its `TuneBucket`-s.

Combining notes and instruments

Rather than providing a specific structure for the melodies students compose, we suggest exercises that gradually lead the student to understanding how music is structured and how various components can be combined and used to generate the next collection of `Tune`-s to play. So, the student may start with a simple sequence of `Note`-s, compose a canon as a sequence of `Chord`-s, create an inversion, transposition, or glide reflection from the original sequence, and compose those into a new sequence of `Chord`-s. The `Chord` sequence can then be played on several different instruments.

We use this to motivate the design of iterators that deliver the next collection of `Tune`-s to play at the next tick, and use circular iterators to create a musical sequence in the style of piano roll, that repeats a melody sequence indefinitely.

Unit testing

The programs students design define the behaviour of the program in response to the tick and key events. Adding the desired music data to the appropriate TuneBucket before each event is invoked generates all musical effects. During the program design stage, students can design complete unit tests that verify that the expected music data has been generated for each tick or key event. To make this possible, wee add methods that allow the student to examine the current state, and the effects of advancing to the next beat for each of the classes we designed. The students can then test that the `nextBeat` method modifies the note sequence appropriately, and that when building a collection of notes to be played on the next tick (e.g. a `Chord`), the collection consists of the expected notes.

We believe strongly that developing a solid design discipline that included systematic unit test design and test evaluation is essential to making students confident and competent programmers, Additionally, our tests-first approach forces students to think through the problem carefully and understand the underlying issues before writing the code.

The Pedagogical Perspective

Our *draw* and *idraw* libraries have been designed to give the students an environment in which they can construct their own worlds and practice on an open-ended problem the basic program design skills. Besides providing a motivation for learning and exploration, it serves a pedagogical purpose as well. Students learn to design fairly complex collection of classes that interact in a number of ways: colliding objects, object aware of the locations of other objects, objects whose behaviour depends on other objects. By focusing on the game model we can insist on proper design, code that is well organized, documented, and tested.

Students built games such as *Frogger*, *ConnectFour* board game, the classic *Snake* game, *Tetris*, traffic simulation, space invaders, and a number of others. However, while the class interactions were quite complex, only a few of these required extensive manipulation of multiple collections of objects.

Of course, adding music and sound effects to the game makes the game more exciting and motivates students even more. But that is not the main reason we decided to extend our libraries. Working with the music sequences provides a rich environment for practicing programming with arrays, `ArrayLists`, and loops. Our exercises ask students to combine several melody sequences together, generate chords, cord sequences, transpose the music to a different key, or construct a canon. They build new classes to represent the complex musical sequences with the built-in iterator to generate the next set of instructions for the `tickTunes` `TuneBucket`.

Our experiences have been great. Students are eager to add musical effects to their games and get motivated to manage complex loops so their music sounds just right. Furthermore, after working with similar games in their first semester, students start loosing motivation when the new game designed in the object oriented style differs little from the one they have designed in the functional style. Adding the sound components makes the whole game design more interesting.

Acknowledgements and Further Plans

This work has been inspired by Erich Neuwirth, especially his pedagogical use of music with spreadsheets, by Uri Wilensky and his team's use of music within NetLogo, and Jenny Sendova's exploration of the music phrase composition as examples for working with sequences of data. Erich Neuwirth's work on spreadsheets and music was especially influential. His spreadsheet language is used to create musical effects by manipulating music representation within the spreadsheet. Our project emulates some of this work in the context of a standard introductory object-oriented programming environment. The author wishes to thank Erich Neuwirth for his continued support and his encouragement for experimenting with music-supported pedagogy.

We plan to extend this work to provide tools for the display of the music in a variety of ways and to leverage the key event handling to allow students to enter the musical sequences by playing the computer keyboard. We will add the library to our website at *http://www.ccs.neu.edu/javalib/* and include tutorials, sample code, as well as downloads for both the library files and the source code.

References

Felleisen, M., Findler, R. B., Flatt M., and Krishnamurthi S. (2001) *How to Design Programs*. MIT Press.

Holbert, N., Penney, L., & Wilensky, U. (2010). *Bringing Constructionism to Action Gameplay*. Paper to be presented at Constructionism 2010, Paris

Neuwirth, E. (2010) *Music and Spreadsheets*. http://sunsite.univie.ac.at/musicfun/.

Proulx V.K. (2009a) *The pedagogy of program design*. In Proceedings of DIDINFO 2009. Brusno, Slovakia, pp. 65 − 74.

Proulx V.K. (2009b) *Test-driven design for introductory OO programming*. SIGCSE Bulletin 2009. 4191), pp. 138-142.

Sendova, E. (2001) *Modelling Creative Processes in Abstract Art and Music*, Eurologo 2001, Proceedings of the 8th European Logo Conference 21-25 August, Linz, Austria.

Mars & Sandrine: Two Success Stories from Constructionist Learning in Rwanda

Julia Reynolds, julia@laptop.org
Learning Field & Development Coordinator, One Laptop per Child (OLPC)

Abstract

I am a member of the One Laptop per Child (OLPC) Learning Team located in Kigali, Rwanda, where there are currently 10,000 XO laptops deployed with another 100,000 en-route. I have a background working to empower young women in urban communities. While I was new to the ideas of constructionism and the benefits of laptop-use in education, I was most curious about the ways constructionist learning theory could affect the lives of young women in Rwanda. During my ten months thus far in the country, my curiosity has been rewarded manifold, particularly through the lives of two young women, Mars and Sandrine.

Mars, a formally shy, reserved student in 6th grade in the Rwandan capital of Kigali walked four hours each week to use her laptop. There, a colleague and I showed her and several other students not only how to use the laptop but allowed her time to think deeply and develop a concrete understanding of things she had been taught in school, but never fully grasped or felt confident in her understanding. After just a few short weeks, Mars morphed into an outspoken, confident, young women, who used her newfound prowess to eventually teach her class and show her family the importance of continuing her education.

Sandrine, also a 6th grade student, and friend to Mars, used her participation in a journalism camp at her school, with the laptops, to elevate herself as a leader in her classroom. She interviewed the Headmaster of her school and offered opinions and edits to her peers. But for Sandrine, her greatest achievement was sharing her finished newspaper with her parents; their delight for what their daughter created greatly inspired Sandrine and to this day, nine months after the conclusion of the camp, her newspaper is prominently displayed for all in her family to see.

Keywords

One Laptop per Child, “play Turtle,” Logo, concrete understanding, creativity

Mars: the quiet girl turned teacher

Mars is a fifteen-year-old P6 (sixth grade) student from Kagugu Primary School, a co-ed school with 4000 students in Kigali. Mars' parents abandoned her and her older sister, Fiona (age 21 with a young daughter of her own), to live on their own with eight others in a small concrete house close to Kagugu. I met Mars through her outgoing friend who always came to talk with me when I visited Kagugu. Mars was quiet, reserved and did not appear to speak much English. In the beginning, I honestly did not notice her.

As a supplement to time in the classroom, a colleague and I asked a local newspaper to publish a weekly challenge based on various activities on the XO laptop called "XO Time." We also offered to any student who needed help solving the challenge, a meeting with members of our team each Saturday at a local café. We started this initiative with private school students in mind, assuming that parents could easily drop their children off at this popular café. On the first Saturday, much to our surprise, a small group of students from Kagugu, including Mars, dressed in their best clothing, arrived.

At Kagugu Primary school, the students are not allowed to take home their laptops nor have they been assigned ownership of any particular computer. The laptops are currently in storage in each classroom, until the teacher decides to use them. (It is OLPC philosophy that each child should have their own computer and should be allowed to take them home, but it takes some time for the schools and the teachers to become comfortable with this concept, especially in a school of 4000 students). The children from Kagugu who showed up to XO Time were eager to have more time to explore and use the laptops. Their excitement was palpable as soon as they saw the laptops on the table.

The first week's challenges were in Turtle Art, an activity based on Seymour Papert's computer language, Logo. We started by asking students to create simple shapes using Turtle Art's snap-together programming blocks. While the shapes were simple, the challenge was not. The participants did not know (or remember) which shape certain angles would create and, even after seeing one angle created, they could not replicate the same angle a second time (for example, when creating a square with multiple 90 degree angles). It was difficult for students to comprehend. This does not mean that Mars and others had not learned about angles and other geometric concepts in school. Quite the contrary, they had received adequate instruction, but they had trouble expressing and understanding this knowledge concretely.

As Seymour Papert discusses in *Mindstorms*, "the ability to articulate the processes of thinking enables us to improve them." (Papert, 1980) He goes on to elaborate on two of his major interests implicit in Piaget's work: "an interest in intellectual structures that could develop" in the child and "the design of learning environments that are resonant with them." (Papert, 1980) He further points to his belief that the "Turtle can be used to illustrate both of these interests: first the identification of a powerful set of mathematical ideas that we do not presume to be represented, at least not in a developed form, in children; second, the creation of a transitional object, the Turtle, that can exist in the child's environment and make contact with the ideas." (Papert, 1980). When my colleague and I saw that the students needed a more literal and concrete understanding of what they had been told in school we decided to proceed based on the Piaget theory that all learning is first physical. Papert and Cynthia Solomon exemplified this theory through their work on Logo, by having students "play Turtle" and we decided to do the same with these students. We used a traditional Rwandese spear to give the students a better grasp of "heading." The students held the spear as we called out different angles, which they had to replicate with their bodies.

Figure 1. Students play Turtle at a local café

Once the participants familiarized themselves with these physical actions, the challenges became easier for them; so we increased the level of difficulty. Mars, in particular, began to flourish. The changes in her demeanor were very apparent. I first noticed that as soon as the sheet with the challenge was passed out, the room fell silent; each child was completely focused on their task with no time for small talk. Even when I offered a hint (for which the students would beg), Mars and others would appear oblivious! Mars also began providing herself with additional challenges. When it was time for the students to draw their own names I thought that Mars would be happy knowing that her short name would be a fairly easy task, but instead she announced that she would draw her last name, UMUBYEYI. It took her the entire session and half of the next, but she was determined to draw her last name.

Figure 2. Mars uses Turtle Art to write her last name

In addition to her academic progress she had seemingly undergone a drastic personal transformation. The formerly “shy” girl who “didn’t know English” began speaking more fluent English than her peers (the best we had encountered in all of Kagugu). She was eloquent, nice, thoughtful and very smart and she became aware of her progression as well. On one occasion, while working with another student, I complimented the student on doing good work and on being “smart.” From across the room Mars overheard this and shouted “I am smart too!” It was surprising to imagine that this was the same Mars from just a few weeks earlier. Her growth continued. As new students arrived at the weekly sessions we needed more help. Some of the new students did not speak English (three of the newcomers were children of a taxi driver who saw the XO laptops and pleaded for his children to have the opportunity to work with us). Mars helped us to translate flawlessly. Additionally, since some of the new students were also new to Logo concepts we needed help getting them up to speed. Mars, again, rose to the occasion. I asked her if she would serve as the official teacher for new students. She shyly blushed and laughed at the idea of being a teacher, but she took charge, claiming a corner and an easel as her own. She used the same techniques we had used with her and taught her peers, confidently and correctly, something that just a few weeks ago eluded her.

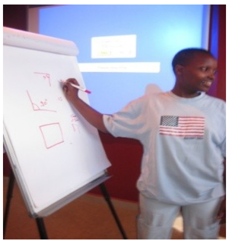


Figure 3. Mars teaches at XO Time

That was not the end of the surprises from Mars. A few weeks later at Kagugu, I met Mars and her friends in the school yard. Their teacher was absent and there was no substitute. Knowing how wonderfully Mars was teaching on Saturdays, I suggested, somewhat facetiously, that maybe she should teach the class. Much to my surprise and without hesitation, Mars said, “Okay, but first can you please make them all quiet?” I walked into the classroom, quieted the students and stood in the back of classroom and watched Mars conduct an entire class. Her peers were quiet and respectful as they grabbed laptops and followed along with her instructions. Two of her friends who attend the Saturday sessions walked around the classroom to help. One student even pulled out a folder with a copy of all the week’s challenges! Mars explained angles, calling on volunteers in the class to physically create the shapes as she called out various degrees in the same manner in which she learned.

Figure 4. Mars teaches her class

Soon, from the back of the room, I saw that all laptops were reflecting squares, triangles and circles. Mars was now not only a “smart girl,” but a teacher. At the end of the school year, Mars’ sister, Fiona, stopped by our OLPC apartment. She talked about all of the great things she saw Mars create and about “how good she is on the computer.” Fiona asked that I help her set up her own email address because now she also wanted to learn. Before leaving Fiona said “I may not have much opportunity for my life anymore, but Mars is very smart, and she is going to be something great, I will do whatever I can; so that she finishes school and succeeds.”

Sandrine: the student journalist

Sandrine is a twelve-year-old student also from Kagugu Primary. She is a friend of Mars and others who attend XO Time, so she began to attend as well. It was here that I could first tell that she was very bright and a leader, much like Mars. That is why, when I held a journalism camp at Kagugu, I wanted to provide a space for Sandrine to recognize her potential and to develop these skills on her own.

I chose to work in journalism because through journalism, students would guide and manage their own work while developing creativity and skills in analytical and critical thinking. Students could also freely express themselves, while working cooperatively in groups. While all decisions regarding the overall development of the newspaper as a whole would be a collective effort among the students, each individual student would be solely responsible for the content and development of their own articles. As journalists the children have the opportunity to be the voice of their communities and agents of change. Students and children are rarely called upon for suggestions as to what they would change or do to improve their community, neighbourhood or village. Now, by creating their own article topics, students can address issues and concerns important to them. This also provides the students with an opportunity to engage and learn more from their families, school, and community and through their writings to create change by focusing awareness on issues important to them. At the same time we wanted students to develop greater literacy and an appreciation for reading, writing and research. Students were divided into groups with Sandrine becoming the unmistakable leader of her group. Because the laptops were a transformative factor in her life, she lobbied her peers to focus their newspaper on technology in Rwanda. Her role as leader inspired the other groups toward more intimate collaboration. Group work is a very new idea to these students and some had trouble being inclusive and sharing ideas. Sandrine served as the ideal example of how this novel behavior could work beneficially. Soon the groups were reaching out to each other for ideas and editing help—creating very powerful learning moments.

The following week, after receiving her completed newspaper, I asked Sandrine what her dad thought about her newspaper. She smiled more broadly than I ever saw anyone smile before and said, “My parents love it! They say my newspaper is beautiful!”

After hearing Sandrine’s declaration and seeing her pure delight in her parents’ admiration, it became clear to me that a shift was occurring far beyond the walls of her school. As a student I remember school projects, as far back as first grade, as being thoroughly enjoyable. They were important to me because I could tailor them to my own interests and passions, but more importantly (which I only realize now), projects provided me with an opportunity to bring my work home and involve my family. I was able to share ideas with my parents and show them my progress in a way that was not possible through solitary study and contemplation. I believe my family also shared a reciprocal joy and gratitude for our productive and engaging time together. Of course, the best part was bringing home the fruits of my labor (diorama, poster board, etc.), with a good grade and having it displayed in our home. This was the first time something like this had ever happened for Sandrine. All she had to share with her parents prior to this was her report card with a grade based primarily on a year-end exam. Now she has a newspaper that to this day (9 months ago) she keeps preserved at home for all to see.

Here is the article that Sandrine wrote:

UBUMENYI IKORANA BUHANGA MU ITERA MBERE
Photo by Uwase Sandrine
IKIGO CY,AMASHURI ABANZA CYA KAGUGU

kiboneka mu karere ka GASABO. Mu mugi wa KIGALI Kikaba cyarabonetswe ho ubushakashasti Abana bakaba batagikenera amakayi. Kuko ibintu byose Bakaba babyigira kuri LAPTOP Abana bakaba bashimira abayobozi. Bazibage jejeho. Ikinyamakuru J,S,CandM tukaba twaregereye. Umuyobozi w,ikigo cyamashuri Abanza cya KAGUGU tumubaza ngo kuba abana. Basigaye b,igira kuri laptop. Babyakiriye bate yadusubije agira. Ati:<<Abana babyakiriye neza usibyeko bitari biboroheye kuko bwari ubwambere>>: None abana bakaba bazitahana. Kugira ngo bakore Imikoro. Tukaba twarabajije umwana. Ururimi bigamo aradusubiza ati biga m,ururimi rw,icyongereza. Ubwo namwe murumvako bageze. Ku Iteranmbere Iyo mbonye Iterambere risigaye Riba I KAGUGU binyereka aho U RWANDA rugeze

Figure 5. Sandrine's newspaper article

Sandrine's article translated:

Science and technology in development

Kagugu Primary School is located in Gasabo, District, Kigali City.

At this school it has been found that kids do not need to carry so many books, the number of books have been reduced by the use of laptops in studying science and technology. All students from this school thank the authorities for these laptops. In an interview, the Magazine J,S,C and M had with the Headmaster of Kagugu Primary School about how they receive the fact that students are using laptops in learning (classroom), he said: << Students have been excited and very happy, but it has not been easy for them because this was their first time to see laptops>>

Now, during the camp, kids take laptops home to use them for their homework.

We asked one student about which language they use in studying, and he said: we study all courses in English. I hope you understand how the development is advancing to this school, and the situation at Kagugu shows how Rwanda is developing.

Figure 6. Translation of Sandrine's Article

So, for me, while I have witnessed and learned much during my first year of field work in Rwanda, nothing exemplifies the importance of laptops in education and constructionist learning like the stories of Mars and Sandrine. Mars, once very shy and reserved, is now teaching her class and commanding the attention and respect of her peers. Children, who had never touched a laptop, walk 4 hours to work on a weekly newspaper challenge. Sandrine now has her very own "beautiful" newspaper hanging on her wall at home for all her family to see. As with most young women, Mars and Sandrine needed an opportunity and catalyst for them and their families to appreciate their intelligence and true potential. The laptops, the work they did and the goals they achieved empowered the children and elevated them both into leadership roles. They will now carry this knowledge and confidence with them the rest of their lives. I am happy to report that while advancing to secondary school was originally, not the course foreseen for them, both Mars and Sandrine have indeed continued on to secondary school. In the past few months I have received calls from both girls saying that "school is too easy for them now!" It is these moments that define the beginning of learning projects and these moments will continue and grow as the program scales and progresses. I look forward to meeting many more Mars and Sandrines.

References

Papert, S. (1980) *Mindstorms: Children, Computers and Powerful Ideas*. Harvester Press. Brighton.

OLPC website. Available at http://laptop.org (Accessed February 2010)

A Constructionist Journey: 42 years with APL - "A Programming Language”

Jean Rohmer, *jean.rohmer@fr.thalesgroup.com*
Thales RD 128, 91767 Palaiseau France

Abstract

This paper revisits the author’s long experience in using APL in various domains of research and industry. We explain how APL can be seen as a construction tool, rather than as a programming language. APL features are compared in this respect to current languages and methods of software engineering and object-oriented programming and modelling.

We advocate that a constructionist approach combined with a “liberal” tool like APL which empowers programmers yields better results, and makes difficult things possible. A detailed analysis is made to stress weak points of classical object-oriented languages, compared to languages like APL.

We emphasize the notion of architecture by layers of tools and constructions, and show why APL is a tool of choice in such an approach. We examine the managerial and cultural obstacles to be removed for a broad adoption of constructive programming.

Keywords

APL, Languages Interpretation, Software Engineering, Programmers Empowerment, Object Oriented Programming

INTRODUCTION

I recently attended a presentation by James Clayson, where he explained how his students in Humanities, at the American University of Paris, were able to program a computer after only a few weeks, and managed to do sophisticated things like painting landscapes of trees and forests, or generating pictures mimicking the style of contemporary masters. He explained that these students used the programming language Logo. These paintings were extremely impressive. During James Clayson’s conference, I discovered the concept of “constructionism” and the way he applies it with Logo, and I realized that my way of programming might belong to constructionism.

Personally, I am doing a lot of programming since 1967, in various scientific and industrial contexts. I experienced many different programming languages, but, when I have the choice, I prefer the one named “APL” (A Programming Language).

In 2006, I was visiting DFKI in Kaiserslautern, the main academic institution in Germany for Artificial Intelligence, and, during my presentation, I made a live demo of IDELIANCE, a comprehensive multi-users system to manage Knowledge Bases and Semantic Networks. Suddenly a bug occurred. Within 30 seconds, I could examine the guilty part of the system, diagnose, fix the bug, and resume the demonstration.

This was visibly the most impressive point of my presentation. None of the distinguished scientists in the audience could ever imagine such a “live” recovery from this kind of situation. They asked me the secret behind: I answered the truth: “APL”.

HOW I BECAME A RECURSIVE CONSTRUCTOR

I discovered APL in 1968 in Grenoble, when I was a student at ENSIMAG, a Computer Science School.

APL arrived there with the IBM 360-67 computer, inside its CP/CMS Operating System, the first large time-sharing OS with virtual memory and virtual machines.

It was the blessed times where the policies of standardization had not their present virulence, and where many interesting languages flourished all around the world.

APL was presented to us along with LISP, Algol 60, Algol 68, PL/1, GPSS, AED, Scratchpad, SNOBOL, … I do not mention here FORTRAN and COBOL, the true standards of theses times, since our founder and director Jean Kuntzmann wisely decided ***not to*** teach them, ***since they were standards*** *…("you will learn them yourself later in the industry if you need them")*

N.B. Today the situation is ***exactly the opposite***: academic institutions teach ***only*** standard languages.

In 1974, I joined a team at INRIA, the national French research institute in computer science, which was building an « APL machine », i.e. a processor specialized in processing APL programs.

For that purpose, we had not only to know how to use ***APL as a construction tool***, but to imagine how to ***design a machine which could construct APL***.

"A machine which constructs APL which constructs cuboids objects –as explained in the next paragraph"

At this point, we must remember that APL was initially proposed by Kenneth Iverson in the late 50's as a notation to help his IBM colleagues describing computer hardware, in other words, to construct machines. Indeed, APL was the tool we chose at INRIA, to describe the logics of our APL machine.

We used APL to construct a machine which constructs APL (which constructs cuboids)

In computer science, this kind of recursion is known as "layers of interpretation": a language at level N is used to construct a machine which understands another language at level N+1.

Language 1 -> construction 1 -> Language 2 -> construction 2 – Language 3 -> …

I conjecture that understanding and mastering several layers of construction is key to become a good constructor.

I conjecture also that the best way to teach and to understand a language L, is to

construct it from another language L-1

construct from it another language L+1

I then became a "recursive constructor".

Later, every time I needed to build some software, I used APL.

To write my thesis of « Docteur ès Sciences », I first wrote a text editor in APL, because IBM just launched in 1975 the 5100 computer, the first « portable » personal computer: 64 K memory, 55 (fifty-five) pounds, 20 000 dollars, 1.9 MHZ Clock)

In fact, APL was the main software environment (with Visual Basic, but most of 5100 were bought for APL). There was no other Operating System than APL itself.

The first personal portable computer was an APL machine.

I then joined Groupe Bull Corporate Research Centre, in Louveciennes, in 1980. I spent some months with a team building a parallel supercomputer. They were painfully using FORTRAN to simulate the hardware. I introduced them with APL. They immediately changed their mind and continued the simulation with APL. They needed only some hours of hands-on exercises. They were constructors and immediately felt the "constructivism inside".

They eventually built an impressive hardware prototype, and APL was key in this achievement.

A few months later, an important event happened in the world of programming languages: Japanese Ministry of Industry disclosed its "Fifth Generation Computing" program, aiming at developing a new kind of computers based on Artificial Intelligence, and totally relying on the PROLOG language. PROLOG was invented in 1972, in Marseille, by Alain Colmerauer, a former student of Ensimag.

Prolog was a true revolution: programming was presented as a process of solving logical equations on symbols. Colmerauer invented Prolog because he has to translate bilingual meteorological bulletins in Quebec. Remember that Colmerauer was not polluted by standards during his education in Grenoble.

Prolog was very difficult to understand for scientists like me, used to other languages. It took me several months to understand Prolog, and the definite way I found was to ***construct in APL my own Prolog Machine.***

APL was my Language L and Prolog my Language L+1

I was impressed by the power of Prolog, and I felt empowered by my construction of a Prolog machine with APL as a tool. I advocated for the creation of an Artificial Intelligence group inside Bull, and I had the opportunity to create and lead this group, known as CEDIAG, and grow it up to 200 people worldwide, from 1981 to 1994. We did research, products and many significant operational business applications in natural language, experts systems and constraint programming.

In 1993, Bull suffered a severe crisis, and most of AI group people, including myself, were invited to leave the company in 1994. In the small world of AI, became what was called the "AI Winter"

AI techniques allowed us to build a new generation of applications, which solved real-life problems, which were out of reach of classical software engineering, like monitoring complex industrial processes, crisis management, application of administrative regulations.

However, achieving such applications required sophisticated tools, expensive machines, skilled engineers, and a deep involvement of customers. In other words, this field of programming - known as **knowledge engineering**- was too demanding, was not ***sustainable***.

These limits gave me the idea of a more ecological AI: a ***personal*** one. Could we design a tool which would enable individuals to become programmers of their own knowledge?

But I was alone at home, without system engineers, C++ programmers and their costly Unix graphical workstations, up to 100.000 dollars each! Again, the solution was APL.

I started on my tiny portable computer to write the first line of APL code of a system I boldly named "Intelligence Amplifier". My idea was that, when performance problems would arise, I could find enough money to rewrite the code in a more standard and serious language like C++.

But, first, performance problems never arose, and, second, I never found enough money to rewrite anything.

I started a small company, and years after years, we were up to 6 people there, and the initial code became a comprehensive multi-users knowledge management system, on Internet and

Intranet. It was marketed under the name IDELIANCE, to large companies like l'Oréal, Air Liquide, Merck, and French Military Intelligence. Now this tool is with Thales Group, and is one of the few "intelligent" systems which has been used in military operations abroad.

The main reason for this achievement were the constructionist properties of APL:

self-sufficient: everything in Ideliance is programmed in APL; we use no database, no application manager, no HTML generator; even the multitasking scheduler is written in APL

easy to learn by doing: the new Ideliance developers were always operational in APL after two weeks, without any formal training. I just showed them what the Ideliance code was, and how they could extend it.

Power of productivity: the language is so concise and powerful that significant new features can always be prototyped in a few days, letting you observe, react, and adapt, -by throwing away everything and recoding if necessary

Transparency: everything is visible and touchable at any time: data and code. That is why I could fix a bug and continue during my presentation in DFKI

Generally, large software engineering projects are developed with too much money and not enough time. By contrast, Ideliance was constructed year after year, with not enough money but enough time. A sort of elegant and continuous epitaxy.

WHY APL IS A GOOD CONSTRUCTION TOOL

N.B. this paragraph is not an introduction to APL. No APL code is shown; we describe and analyse some properties of APL.

APL is good at something apparently[[abstract and virtual:

"construct rows, square or rectangular arrays, cubes or cuboids of numbers or characters"

(cuboid is a shorter word for rectangular parallelepiped)

This seems abstract, but:

- A word, a sentence is a row of characters
- A computer screen is an array of pixels of colours
- A colour is a row of 3 elementary colours
- A computer hard-disk is a cuboid of characters

With APL, you can build such objects, which are organized sets of elements.

More, you can create hyper cubes, hyper cuboids of any number of dimensions.

In the APL "workspace", each of these objects has the name and the content you gave them. Then APL lets you assemble these objects:

- You can add a row to an array, an array to an array, if the adjacency dimension is the same
- If you stack arrays of same dimension, you get a cuboid.
- You can put side-by-side two cubes if they have the same dimension: you get a cuboid.
- You can take a slice of an objects to get smaller ones.

- For instance, take the first three positions in a row, or the last 5 ones, or the positions numbered 3, 5 and 7 (this last operation yields a row of three elements)

At any time, you know the **list** and **name** of the objects inside your workspace, you know their dimension, you can **see** their content, and you can **modify** this content.

Any elementary operation on numbers and characters is generalized to these multidimensional cuboids, if they have the same dimension: the sum of two cubes is a cube. APL provides operators which construct new objects from old ones.

Classical mathematical operators are provided, like arithmetic operations, generalised matrix products, union, intersection, projection, comparison, permutation, rotation …

Less classical operators let you perform operations which transform the dimension of the objects (like flattening, putting side-to-side, expanding …)

A very special point is that APL represents most of these operators by a ***new*** letter of a ***new*** alphabet, invented by Ken Iverson. APL is a language written with special letters expressing operations on complex objects.

You compute new objects exactly as you compute new numbers with usual arithmetic operators. An APL expression looks like an arithmetic expression, with arrays or cubes instead of numbers, and juxtaposition or slicing instead of additions and multiplications. APL expressions are very concise, you can understand what they do just by looking at them.

Note that, in the same way young children learn arithmetic without any reference to programming, we have not yet used the word “programming” about APL.

APL lets you construct objects –rows, arrays, cubes, cuboids, hyper cuboids- in a multidimensional world, each elementary point of theses objects being a number or a character. In other words, APL objects are **sets** of elementary information, and APL operators operate on ***sets***. Whereas in other programming languages you have to program as many loops as you have dimensions to repeatedly execute each elementary operator on elementary information.

Above this powerful set of objects, operators and expressions, APL provides the classical features of programming languages:

- Assign the result of the calculation of an expression to an object
- Execute a sequence or loop of such assignments
- Define a function as the performance of a sequence
- Reuse this function as an operator in other expressions

Another characteristic of APL is that it works like a pocket calculator: you type some expressions, and it instantly creates and computes objects.

When you want to do something, you can do it immediately, and you can keep track of your work, to replay or refine it later.

The consequence is that you become a mighty constructor of objects:

- you can construct
- you can see what you construct
- you can see how you construct

If you can construct, you construct. Using APL empowers you. And when you ***can*** construct, when you ***do*** construct, you can ***think***, you do ***think*** differently. Doing and thinking work in a close loop.

This idea that doing and thinking could be the same in computer programming is unfortunately in total opposition with the current ***doxa*** of software engineering, which strictly isolate designing and doing.

About the etymology of "construction":

It comes from latin "strues", which means "heap". And in latin "structor" means "mason".

A recent theory about Egyptian pyramids construction, proposed by architect Pierre Crozat, is that they were built as heaps: a new layer of stones was laid on a pyramid "N" , simply by climbing it, producing pyramid "N+1". ***The pyramid was its own ramp.***

WHY APL IS SO UNKNOWN

If qualities of APL are so evident, why is it nearly totally unknown? Reasons are deep, they do not come simply from APL characteristics, they are ***cultural***.

When I started programming in the late 60's, ***programming*** was the most prestigious thing you could do with a computer. ***Using computers*** was reserved to clerks, female clerks who were employed for inputting data with punched cards, and male ones for loading and unloading cards racks and tapes, sorting printer listings, night and day. High-level white collars never saw a computer, except through the glass walls which let visitors gaze at the temple of the computer room, with great priests bustling around. After 40 years, things went exactly the other way round.

High-level management is today surrounded by the computer screens of their office and their smartphones. And programmers have become second-class corporate citizens.

High-level management have replaced clerks as major users –in the sense of "hands-on"- of computers. Programmers are relegated to invisible subcontractors, and programming is considered as a dangerous activity.

In fact, our technical civilization has still to understand what programming is. Industry and Academy made the choice of trying to reduce programming to engineering. But building an information system is quite different from building a bridge. (We should think of how constructionism may differ when applied to civil engineering or to computer programming).

Software engineering is a contradiction in terms.

Between management and programmers, engineering interposes **methods and standards**. The idea is to produce a cascade of deterministic transformations which will start from an "expression of needs" and end with an "executable machine code". This does not work, but the reaction of software engineering is to say "this is the proof that we need still more methods and standards".

With this culture in mind, it is clear that a tool like APL which empowers programmers is heretic and subject to the most severe inquisition. This has reached the point that most people cannot imagine that programming is a rich intellectual activity.

When I explain my vision of systems like Ideliance, people are in general very positive, until the killer question comes: "But who programs all theses nice things for you ?".

The hypothesis that Ideliance was possible **only** because I elaborated the concept **while** constructing it **with** an empowering tool like APL is just unthinkable.

In such circumstances, I incline to answer: **“Ich bin ein Programmer!”**

One of the key foundation of software engineering is that we must prevent programmers from making mistakes. It is an application of the precautionary principle.

Management believe that, since there exist methods and standards, all problems are solved by advance. In their eyes, methods and standards depreciate the work of their employees, which become interchangeable.

The true question should be to compare the mistakes induced by an empowered programmer and the mistakes resulting of the specification cascades of software engineering.

Note. Another reason why APL is not popular today is the **NIN** syndrome **: « Not Invented Now ! »**

COMPARING APL WITH OBJECT ORIENTED MODELLING AND PROGRAMMING

Object-Oriented Programming, through languages like C++ and Java, and Object-Oriented Modelling, with formalisms like UML, are today the official and unique way of programming taught in schools and universities, and the only practice accepted by Industry. This ***pensée unique*** has many drawbacks.

Indeed, methods and standards cannot be harmful by themselves. The problem is that they are currently proposed as ***the*** solution, ***independently of the tools*** –i.e. the languages- to which they are applied. And, precisely, they are applied today to languages like Java and C++, which are, in our opinion, very bad choices, because they are crippling tools rather than empowering ones.

We can even ask whether current methods are not around just to try to fight the crippling due to low level languages, like crutches.

Recently I listened to a conversation between two programmers in a bar. They just were exiting a course on methodology, namely “design patterns”. Verbatim: “We need not instructors or patterns telling us how to do our job, we need tools powerful enough to do our job the way we want”. I was happy for them, their clearness had survived the course.

We can fear that some managers think good methods achieve good results with bad tools.

Why are languages like Java and C++ bad construction tools ? Here are some reasons:

- R1: They manipulate very low level objects, and one element at a time: add two numbers, compare two characters, index elements one by one in a list.
- R2: They do not trust programmers: before writing any operation, programmers must make a declaration of what they want to do, when and how. They call it “typing”. A sort of bureaucracy, where you have to declare by advance everything you intend to do in the future, and conform to it.
- R3: When you have finished to “type” –in both senses- your program, it escapes you: the bureaucracy takes your program, checks if you obeyed your declarations, then “compiles” it, links it with other programs, runs the program
- R4: to understand whether your program is running as you expect, you have to add extra –low level- instructions to let you visualize, then imagine, its behaviour

- R5: since all the above-mentioned points are very painful, you try to spend a lot of time in minimizing the number of lines you write, by organizing your code in a modular way, by abstracting and reusing some parts (with, at each step, the obligation to declare your abstraction to the bureaucracy)
- R6: since all the above-mentioned points are very painful, you in fact do not use directly the programming language, but a "programming environment", which you must learn to use in supplement to the language itself (like IBM Eclipse environment)
- R7: the documentation which explains how to perform the above-mentioned points amounts to several thousands of pages, and much more if you include the variants and community tricks found on the Internet
- R8: since all the above-mentioned points are very painful, you are tempted to program less and less, (remember, it was the objective of the Methodists who do not trust you). You are invited to reuse programs written by others. They call it "libraries". But libraries mean still more thick books to retrieve, read and understand (see R7), and still more bureaucracy to declare how to use features typed and declared elsewhere by others.

What is by contrast the constructivist philosophy of APL ?

- R1: manipulate very high level objects, yet simple to understand, set-oriented, mathematics-oriented, with powerful operators
- R2: trust programmers: let them assemble objects as they want. Adopt a **liberal** rather than a **carceral** attitude towards programming. When programmers feel more responsible, they finally make less mistakes
- R3, R4: programs are interpreted instead of being compiled. High-level objects –cuboids- are visible at any time. Concise extra code can be written and executed directly at any time to experiment the current status of your objects
- R5: since code is usually 10 times shorter, since operators are powerful, you need not so many intermediate abstraction layers which hide what is behind: you just read what your code is and does
- R6: APL comes with its own dedicated environment which optimizes the above-mentioned points
- R7: a few hundreds of pages of documentation are enough, and more important, if you need to understand the behaviour of an operator, you just try it online
- R8: the more you program, the better you master the tool, the faster you program, the better you construct what you want. It remains feasible to go down to the very lines of code. What you have constructed is concise, its semantics is clear to somebody who would like to reuse it directly, or –better- transform it.

CONCLUSION

By writing this paper, I constructed some understanding of what I did with APL during the last 42 years.

Complex systems construction is difficult. A layered approach may help. Each layer must bring its own added value.

A layer is constructed with two things:

- a tool T
- a construction C, performed with T

The result of C is to yield a tool T+1, with which construction C+1 will be performed, yielding tool T+2, etc …

If we are in the following configuration:

T-1 -> C -> T -> C+1 -> T+1

Tool T has an added value, a ***raison d'être*** , if the sum of complexity of constructions C and C+1 is smaller than what would be the complexity of constructing T+1 directly from T-1.

APL is a good constructionist tool T because:

- It provides high-level set-oriented objects and operations (cuboids) to be used by C+1
- Its objects and operations embed a direct indexing logic which easily matches with low-level features used by T-1 tools, closer to the Von Neumann architecture

A good tool T is useful only if C+1 actually follows constructionism principles, among which freedom rather than methods, knowing by doing rather than doing after knowing.

Indeed, constructionism is not an enemy of knowledge and methods.

A recent TV document showed a wooden shipbuilder in the Syrian Island of Arwad. He explained that he builds **alone** 50 feet-long ships. He said: "I build them fast, because I use no blueprint, this way I do not loose time watching at blueprints. We are doing like this here for 4000 years"

We also must have constantly in mind the motto of Leonardo da Vinci, the prince of constructionists: **"Ostinato Rigore"**

Acknowkedgement

To the company Dyalog Ltd, www.dyalog.com , the company who produces and sells Dyalog APL since 1983

EasyLogo – discovering basic programming concepts in a constructive manner

Lubomir Salanci, *salanci@fmph.uniba.sk*
Dept of Informatics Education, Comenius University

Abstract

EasyLogo was designed for people with basic computer skills to make programming and problem solving as easy as possible for them. We wanted to use environments that would be much simpler than those in Imagine or Scratch. We, therefore, developed EasyLogo with a number of very unusual features, for example: EasyLogo works with grids in which the painting process occurs; the turtle rotates by 45°; the prog ram is constructed from simple cards, which the user then arranges with their mouse; the program is executed automatically meaning that the painting process updates whilst the user is making changes. The Logo language has been radically simplified – offering drawing commands, loops and procedures only.

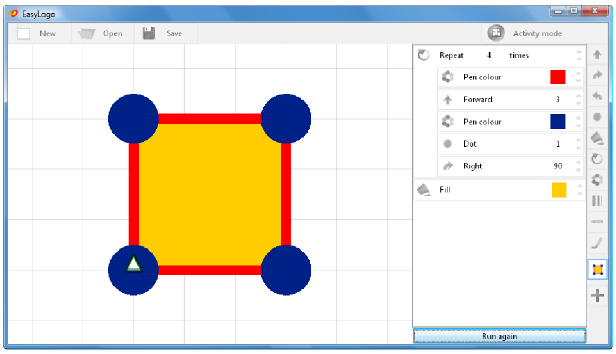


Figure 1. The environment of EasyLogo has an output area with a visible grid, a program editor that already contains a number of cards with commands, and icons of commands, which the user can drag into the editor. This is a so-called free programming mode where the user constructs anything that they like.

More importantly, the EasyLogo environment contains a collection of activities, which have been specifically designed to teach users basic programming concepts in a constructive manner. The consequence of such an approach is that EasyLogo blurs the boundary between computer games and programming. We believe that it is important to keep such educational environments open, in terms of activities, to allow the user to modify activities and to add their own.

Whilst testing the environment, we observed an interesting behaviour amongst users – a noticeable majority solved their assignments with loops. At first the user discovered a repeating pattern and consequently arranged commands for it; then a repeat construction was inserted and existing commands were moved into a repeat container; finally, the user changed the number of iterations for the repeat command. We revealed how very important it was that newly inserted repeat constructions had the number of iterations set to 1. For other default values, users were very confused. Such incremental processes of design, observation, analysis and improvement helped us to develop a comprehensive environment (note: this approach is close to design–based research).

Keywords

Logo, Programming, Grid turtle graphics, Didactics, Educational environment

Another Logo – Why and For Whom?

A new approach to Problem-solving had to be realized for a lot of teachers at primary schools. We knew that our participants were beginners in informatics and that they would never become programmers in the future, and many of them were of an older age group with mixed levels of digital skills. Conversely, our participants were very enthusiastic and creative and, as primary school teachers, they had a vast experience of social and natural sciences. As a result, we decided to approach our project in a very constant and fun way.

We began by considering, comparing and analysing various existing environments and our findings showed problems with all of them:

- Participants found the (semi) professional programming languages extremely complicated and frustrating.
- In traditional Logo environments (such as Imagine, MSW Logo), users have to learn/memorise commands and syntax rules. This leaves room for error, such as typos or missed symbols (eg. bracket in repeat command). Ultimately, these strict conventions detered our participants.
- Alternative environments (like Scratch) are too complex with a lot of commands – building blocks.
- Children's environments (like Thomas the Clown) are too simple and too closed in terms of their activities. So that adding new activities to them is difficult if not impossible.
- Flash games might be a lot of fun, but they were designed for other purposes with only a few of them directed at programming goals, and the majority closed in terms of adding new activities.

Despite these disadvantages, all of the mentioned environments have a lot of great features, which were a positive inspiration to us as we developed our project (see fig. 2).

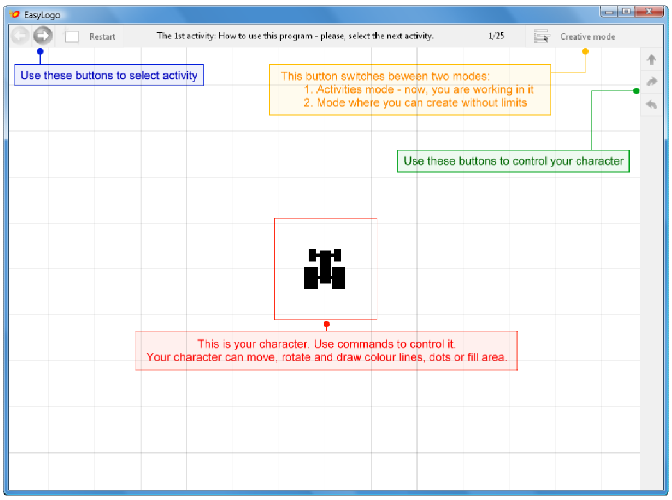


Figure 2. EasyLogo guides users, teaching them about how they can work within the environment. Such an approach is the norm amongst many of today's games.

We believe that primary education should be fun, that it should use positive motivation, and that it must respect the (constructivist) theory of learning. This led us to the development of EasyLogo – a new Logo-type environment. Our principal goal was to develop an environment for primary school teachers who were studying on our course. Our secondary goal was to make the environment so simple that it could be used with older children at primary or lower secondary schools during lessons on informatics.

These were our final conclusions:

- We must simplify the original turtle graphics.
- We must keep our environment open so that any user can add new activities to it.

What is EasyLogo?

The key features of EasyLogo are:

- A simple and very intuitive user interface.
- A reduced programming construction – the environment allows loops and procedures only (no variables or object-oriented programming are implemented – for now).
- A blur between the boundary of computer games and programming – a number of activities are offered to users.
- Unusual turtle graphics – the turtle paints in a grid and rotates by 45°.

The EasyLogo language has the following commands/programming constructions only:

- `Forward` – moves the turtle a constant number of steps.
- `Left`, `Right` – rotate the turtle by 45°.
- `Repeat` – works as a container that repeats commands a constant number of times.
- `Dot` – paints dots
- `Fill` – colour fills an area.
- `Pen color`, `Pen width` – change the properties of the pen.
- `Move`, `Draw` – disable or enable drawing (like the commands, `penUp` or `penDown`).
- It is possible to create new procedures to draw shapes to specification.
- It is possible to invoke your own procedure as a command to paint a defined shape.

Figure 4. It is possible to draw some stars with such simplified graphics.

The `Fill` command colour fills a specified area, but it has an unusual semantic. We can see this on the previous example: firstly, we draw a closed polygon, and then we give the command to fill the previously defined area.

The `Dot` command allows the user to draw around things like: road signs, balloons or flowers.

Figure 5. Dots are the marks in the corners of the square.

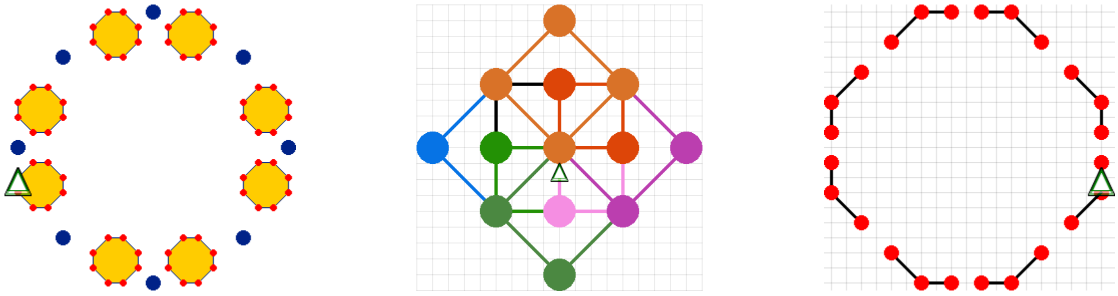


Figure 6. These are other examples of abstract pictures containing dots.

A program is constructed using cards and edited by the mouse, which means that users do not have to type commands with a keyboard. Similar approaches are used by several other environments, like Thomas the Clown, Baltie and Scratch. The precise design of the cards, alongside the mouse operations, ensures that no syntax errors are made by the user as they construct their program.

Why grid graphics?

We decided to put an accent on positive motivation from the beginning: a simple and easy to use environment, with fun activities and interesting drawings. We did not want to overwhelm our participants with complicated syntax rules, with a lot of varying commands, nor with mathematics (which, in many cases, is used improperly from a didactics point of view and complicates a beginner's understanding of basic informatics concepts). Our approach was to make the language, the turtle control commands, and the graphics as easy as possible.

We considered that it is much simpler to count steps or to calculate lengths when drawing shapes in a grid. This approach is very similar to what can be seen in some graphics editors – the grid helps users to align shapes and connect lines. Similarly, our proposed graphic pen always finishes a line at a grid point.

The 45° rotations are equally simple for calculatio ns, but are powerful enough to draw a triangle, the roof of a house, some flowers, crystals or simple stars. Thanks to grid graphics users do not need to calculate square roots...

We decided on a square grid, because a triangular (or hexagonal) one would not allow for the drawing of vertical or horizontal lines.

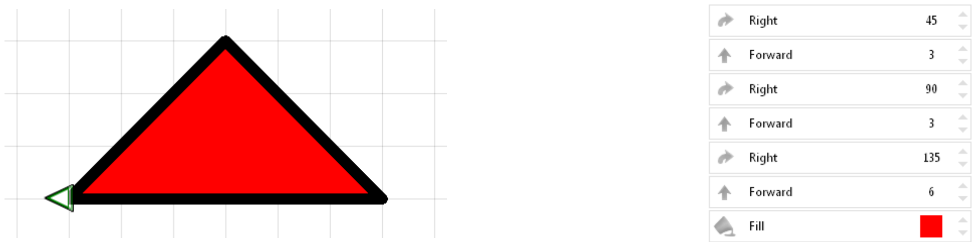


Figure 7. This picture shows the right triangle as a roof and the commands used to draw it. Although the triangle is right, the commands contain no square root calculations.

EasyLogo generates a vector drawing as the output of a program execution, so that any resultant drawing can be freely zoomed and the user can quickly change the size (and the size of the grid) via the mouse wheel.

Weaknesses of grid graphics

The grid turtle graphic does not have Euclidean metrics (measurements of distances). This means that some shapes with diagonal lines are not immune to rotations by 45 degrees. Let's see the next example:

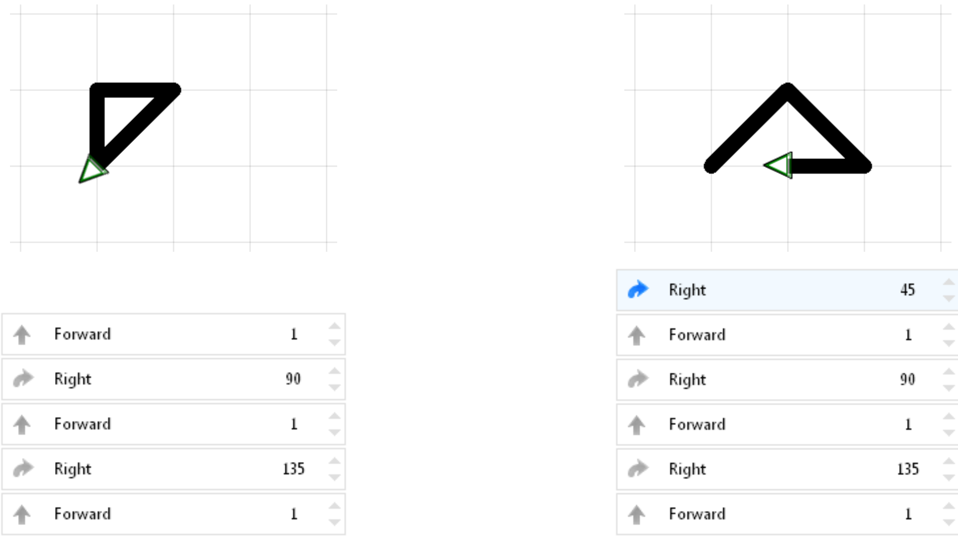


Figure 8. On the left side is a small triangle. We can see what happens when we rotate it by 45°on th e right side. The original triangle shape is broken.

This is the weak spot of grid graphics. But, if we learn to accept this fact, we can still produce nice drawings or realise interesting activities.

Procedures

In EasyLogo, procedures are represented as shapes that were previously drawn.

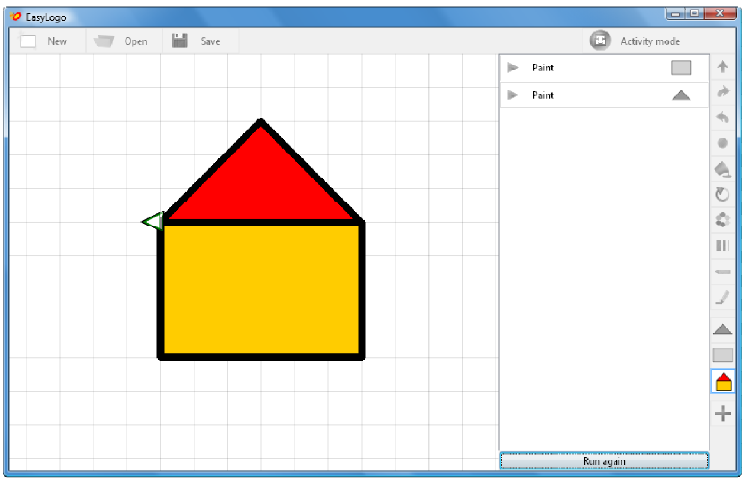


Figure 9. Picture of a house constructed from procedures – rectangle and triangle. We can see our own procedures in the right-hand corner of the window. Each procedure is represented by an icon of the shape which the procedure draws.

About activities

The important part of EasyLogo is the collection of activities, which are a series of small microworlds or tasks that need to be solved.

Different activities have various educational goals:

- To control the motion of a character (like a car, bee or girl) directly – by clicking on buttons.
- To control a character using a program – by constructing a sequence of commands.
- To construct a sequence of commands, which solve a simple problem.
- To recognize repeating patterns and design a loop.
- To construct procedures and use them to build a shape.
- To fix or to improve a program.

The user may arbitrarily browse, solve or skip activities. EasyLogo does not check a user's solution. A user may also switch environments from the Activity mode to the Programming mode where they can freely create any drawings that they like.

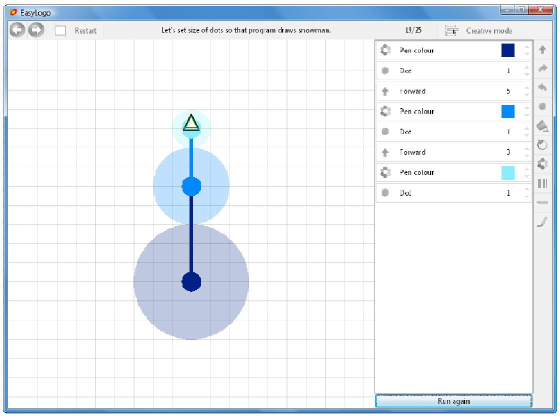
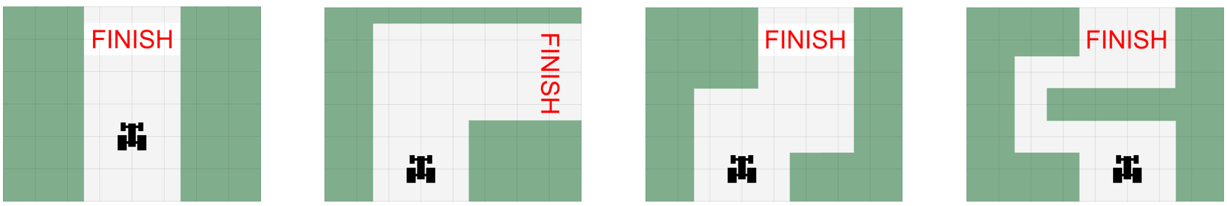


Figure 10. This activity requires the user to create a snowman by changing some parameters in the program. We can also see that EasyLogo displays a ghost image of the result below the grid.

When we were thinking about the concept of EasyLogo, we thought about creating activities that varied in complexity. For example:

1. The user clicks on buttons that directly move the Car. We designed these activities in order to make the user familiar with turtle motion and rotation.

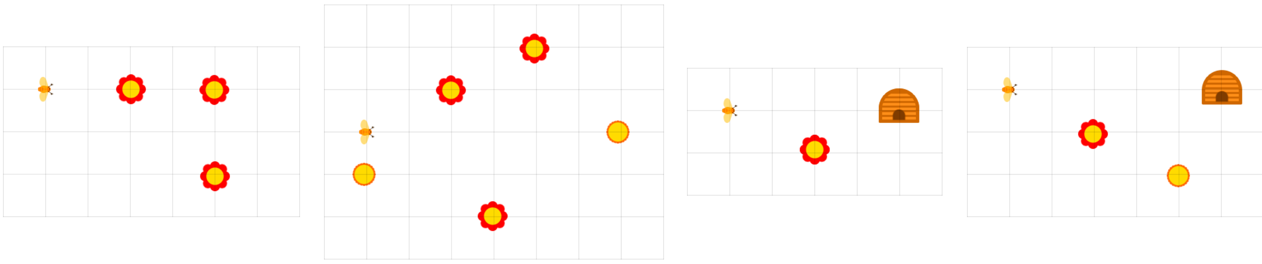


These activities are extremely simple but they are very important for developing knowledge of relative and local turtle geometry:

- The first activity could be solved just by clicking on the button.
- Consequential activities require a combination of rotations and movement .

The difficulty of each activity is defined by a road that the car must navigate, especially by the number of bends in each road.

2. While the previous series of activities required just clicking on buttons, the following series requires programming. These activities teach the user to construct a linear sequence of commands from `Forward`, `Left` or `Right`. We used the idea of a Bee that needs help to visit a number of flowers.

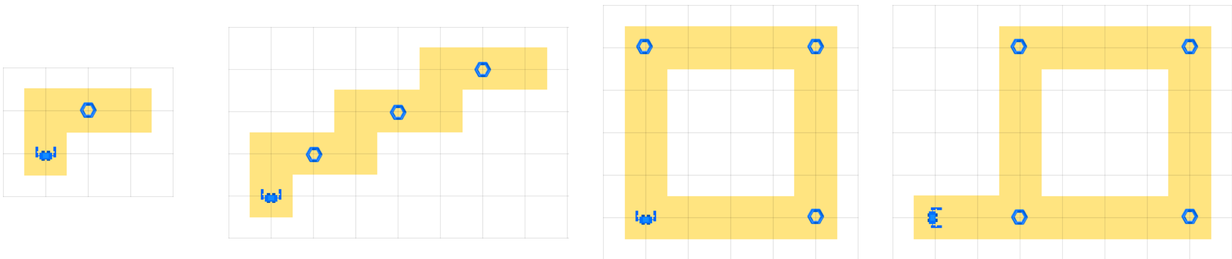


The activities are graded in levels of complexity:

- The bee must visit 3 flowers (in any order),
- The bee must visit more flowers (in any order),
- The bee must visit all of flowers and then fly into the beehive.

There are several skills that the user needs to master here: how to imagine a path for the Bee (which is also a very simple graph problem), how to drag and to order commands, or how to discover and change parameters for some of the commands.

3. Activities for `Repeat` commands were designed in order to teach the user to recognise repeating patterns.



In several activities, we used the idea of a robot that needs to collect messy gadgets:

- The solution to the first problem will be the body of the future loop.
- The user should (re)discover a repeating pattern and construct a loop with a repeat construction.
- In the more complex problem, the robot should realize something different before the loop itself (see the last picture in the series).

There are additional activities for loops that require the user to solve more complex problems. For example, they have a more complicated body of loops or they need to combine loops with procedure calls etc.

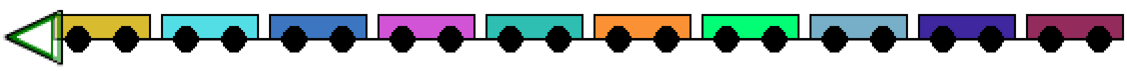


Figure 11. Example of a more complex activity: the train is constructed from several wagons.

In this way we could continue and describe educational goals and analyse our didactics approaches, for example, with other activities. But we then realized that we could use the existing scheme of activities, but organize them according to didactics principles:

- We always start with a very simple and elementary problem.
- Then we allow users to gain in (practical) experience.
- And only then do we introduce them to a slightly more complicated situation.

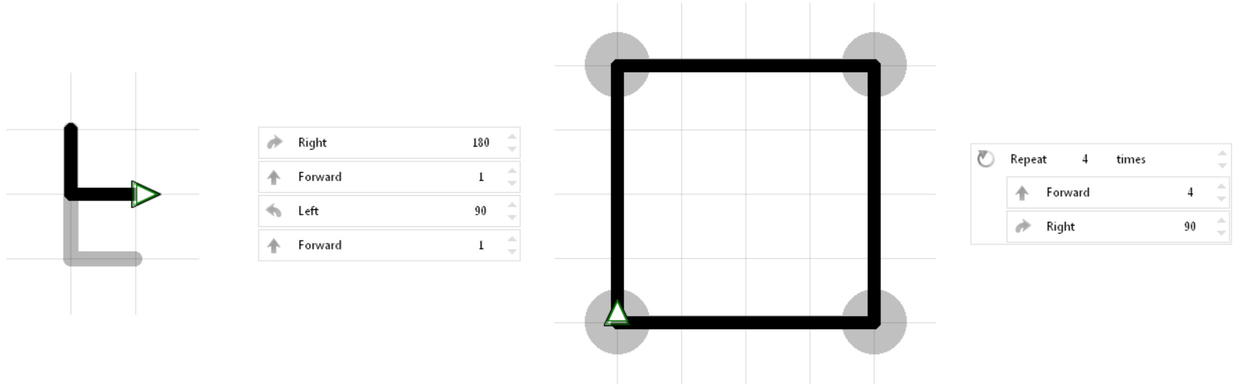


Figure 12. These are examples of activities in which the user has to fix programs – for example, to draw the letter L or to draw a square with dots. Gray drawings that are displayed behind black parts are requested results.

To create a graded series of activities we had to recognize and to qualify various levels of complexity. According to our empirical experience, this is the hardest problem for many people in didactics and the teaching of programming.

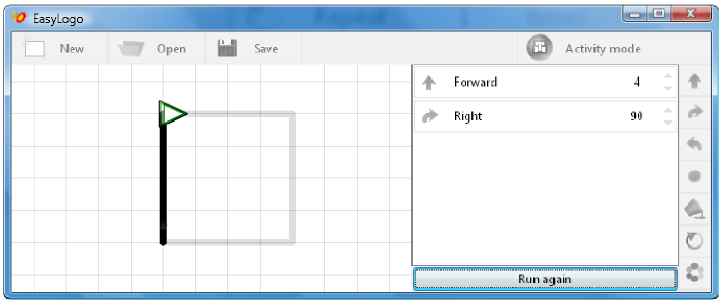
Interesting facts, observation, results

1. We believe that it is important to keep educational environments open in terms of activities. EasyLogo was designed in such a way that anyone can modify or create new activities. We decided to use simple textual files as a description of each activity and its properties. Optionally, each activity may display a bitmap as a background drawing.

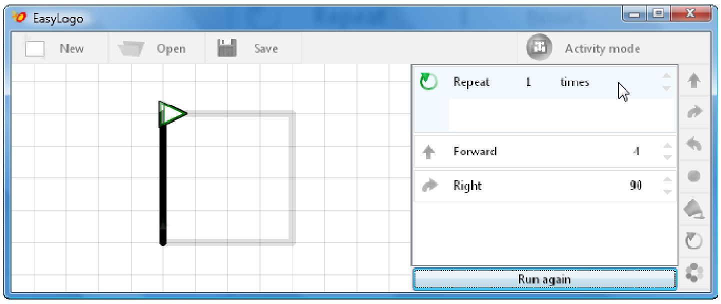
2. We developed EasyLogo in order to draw pictures immediately as the user made changes to the program. This means that EasyLogo always executes a program automatically as the user inserts/removes commands or sets parameters.

This was to become an incredibly useful feature, because it gives immediate visual feedback during the construction of a program. Sometimes, it is fun, interesting and edifying to see changes at once as we “play” with programs or parameters.

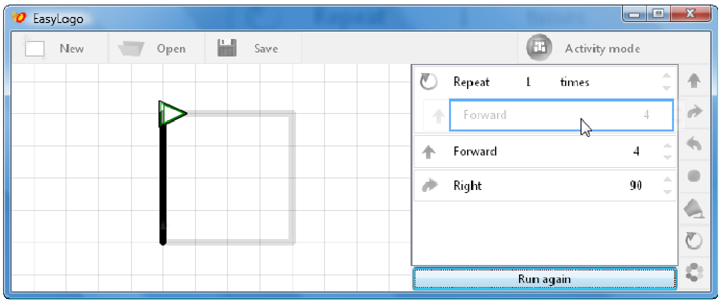
3. Thanks to the previous feature, we observed how people construct algorithms with loops. We will illustrate this process using the example of drawing a square – let’s have a look at the following images and comments below them:



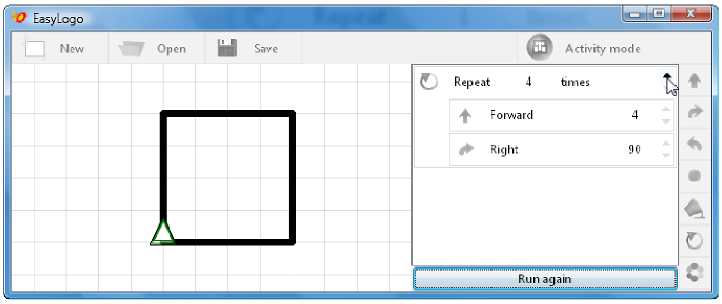
People find the repeating pattern first. Then they arrange commands which will become the body of the loop:



People insert the repeat command later. They usually put this command at the first position. Please note that, by default, the repeat command has set the number of iterations set to 1.



Then, they move previously arranged commands into the body of the repeat construction.



Finally, they change the number of iterations.

We saw this method widely used amongst young children as well as many adults.

4. There is an interesting story as to why the default number of iterations is set to `1` for the repeat command. In the early version of EasyLogo the repeat command had the default number of iterations set to `4` so that a newly dragged command said "`Repeat 4 times`". But we had to change the value from `4` to `1` in later versions.

Our colleague Monika Tomcsanyiova has a `6` year old son, Tomas, with whom she decided to test EasyLogo. She observed that Tomas worked exactly in the same way as we have just described. However, Tomas became very confused when he started to move commands into the body of the repeat.

As we discovered, he was confused because all of the moved commands were automatically executed `4` times, whereas others were executed only once. As a result he saw some strange drawings appear whilst he was working and this did not make sense to him. The program drew something other than what Tomas had expected, and this is reason why the repeat command now has its parameter set to `1` by default.

This experience shows that we must carefully consider every detail of an educational environment when using programming languages with beginners. Otherwise, the resultant product is counterproductive rather than useful.

5. We showed EasyLogo to another group of teachers; those who are teaching programming at lower or upper secondary schools. EasyLogo was introduced to them as an example of a programming environment that was primarily developed for educational purposes in contrast to Java, C# or other languages or environments designed chiefly for the software industry. The teachers were very positive and were absorbed with EasyLogo when using the program. To be correct, however, we must say that we do not, as yet, have feedback from schools where children have used EasyLogo – simply because this environment is too new.

Conclusion

Our goal was to introduce programming to beginners in an easy to understand way by designing an environment which blurs the boundary between computer games and programming. However, we still believed in keeping to fundamental informatics goals, such as: problem solving, the designing of algorithms, the use of programming constructions, the decomposition of problems into smaller parts, how to solve simple graph problems or how to work within a grid structure. For this purpose we designed a series of activities, graded according to their complexity in problem solving, to help users steadily develop their skills and knowledge.

We believe that EasyLogo combines several unique features:

- We implemented and analysed turtle graphics in a grid.
- We designed the fill command.
- We created activities that were an intrinsic part of the environment itself.

Technically, a user constructs a syntax tree by dragging commands using a mouse. The display list is generated from such a structure and is then rendered on the screen.

Our incremental process of design, observation, analysis and improvement is close to design–based research. For example, this approach helped us to make the repeat construction far more intuitive and comprehensible for users.

References

Kalas, I. and Blaho, A. (2001) *Object Metaphore Helps Create Simple Logo Projects*. Eurologo 2001, 55−66.

Correeia, T.and Correeia,S. (2007) Natural *Visual Programming Languages*. Eurologo 2007, 44.

Modelling without Mathematics – Using JlinkIt modelling tool in educational settings

Fábio Ferrentini Sampaio, *ffs@nce.ufrj.br* and **Márcio Reis Teixeira,** *marcioreisteixeira@gmail.com*
Programa de Pós-Graduação em Informática – Federal University of Rio de Janeiro

Description

For many years our research group on IT in Education (www.nce.ufrj.br/ginape) has been working with modelling in education. As part of this research we have developed a computer modelling tool called JlinkIt that allows us to construct and simulate causal dynamic models - from a System Dynamics approach - without the necessity of knowing the mathematics that are normally used in analytical models (mainly calculus and differential equations). This modelling tool was developed in Java and runs in any browser (http://www.nce.ufrj.br/ginape/jlinkit/executa_jlinkit.htm). It also has a stand-alone version that can be run on computers not connected to the Internet. The software is free and can be downloaded from its website (http://www.nce.ufrj.br/ginape/jlinkit/download.htm).

Method

During the workshop we will present the software and develop some ideas on how to use it in educational settings based on our experience with Science teachers and secondary schools in Brazil.

Expected outcomes

Attendees will develop an idea about modelling from a System Dynamics approach and possible uses in educational settings.

We are setting up a discussion group on the internet where participants (and others) could discuss further some ideas related to modelling in education and share models.

Keywords

Modelling in education, computer modelling literacy, computers in education.

Workshop on Mathematics and Dance

Karl Schaffer and Erik Stern, Co-artistic Directors, Dr. Schaffer and Mr. Stern Dance Ensemble, *schafferkarl@fhda.edu estern@weber.edu*
Schaffer: De Anza College Stern: Weber State University

Introductory description and overall goals

The presenters will offer a workshop for *Constructionism 2010* participants on how to integrate mathematics and dance in the classroom as well as on stage. They will incorporate several mathematical topics, including symmetry, counting principles, and the mathematics of rhythm. Participants will create, practice, and perform short dance phrases, and simultaneously explore mathematical principles and critique the work from the point of view of both the mathematics and the artistry involved.

Method

The purpose of this workshop is to give participants a palpable experience to help them understand connections between mathematical and choreographic concepts. Prior experience in either mathematics or dance is not required. The workshop alternates between creative problem solving and reflection/discussion. Workshop participants will (1) solve problems physically in groups of two to four, (2) discuss problems in smaller groups and as a class, (3) will first explore creatively the subject matter and then examine formalization of the concepts, (4) Discuss the use of these activities in K-12 and college classes.

Expected outcomes

Workshop participants will (1) solve problems physically in groups of two to four, (2) discuss problems in smaller groups and as a class, (3) will first explore creatively the subject matter and then examine formalization of the concepts, (4) Discuss the use of these activities in K-12 and college classes.

Keywords

Choreography. Open-ended Problem Solving. Symmetry.

Images versus Imagination : a destructive contribution to constructionism

Dominique Sciamma, *ds@stratecollege.fr*
Head, Interactive Systems and Objects Dept, Strate Collège Designers (France)

Abstract

This paper describes a pedagogic exercise given to designers to be (1st year), within a communication class. The exercise is only based on the use of 6 given images.

The exercise takes place after two sessions, the first of which being devoted to the participative construction of the famous communication diagram (emitter, coding, channel + noise, decoding, receiver).

In the second session, starting with this diagram, the emitter is then replaced by "the world" and the receiver by "you". With the help of this redefined diagram, "demonstrate" the students that the first limit of our communication with the world is to be found in our perception mechanisms, and that, in that respect, we are always abstracting, and that what we call "reality" is literally and technically an abstraction.

To free ourselves from this prison, we then "demonstrate" to the students that the only way to go beyond the limits of our perception is to build theories, through which it is then possible to discover all the hidden dimensions of the world.

But, by doing so, we are in reality building a new but even more wicked prison: the prison of our vision of the world. In that respect, we are de facto projecting on the world what we know about it: after being an abstraction, the world is moreover a projection.

The actual exercise demonstrates this second proposition in a very obvious and direct way.

With the help of 6 very simple images including characters and various other representations, we demonstrate, again, that the students are prisoners of theories. This is done by asking them to create a story of their own, using all 6 images, without any other constraints than the images themselves. We urge them to be creative.

When that is done, after one hour, the comparison between all the stories shows clearly and simply that they are projecting what they know about the "theory" of images and narration, thus limiting themselves to a common - and therefore not creative - domain of narration.

After this explanation is given, students are asked again to produce more stories, but this time by freeing them from the theory, and the results are simply astonishing.

Keywords (style: Keywords)

Images, Imagination, Creativity, Communication, Theories, Perception, Iconoclast, Design, Philosophy.

Preamble

This is not a scientific paper.

This is a paper about teaching, a paper about reality, a paper about teaching reality. This is a paper about creating, a paper about teaching creativity. This is a paper about design, a paper about teaching design. Eventually, this is a paper about all possible combinations between reality, creativity, design and teaching.

It does not intend to write some science, or even to theorize. It does not relate to any designated academic works, even if a lot of readers will connect it to known theories or researches.

No. It is a paper about my own and personal practice, within a French design school – Strate Collège Designers[1] – in which I have been giving so-called "communication" classes to the first year students for more than 9 years now.

This class is about destroying – quite an apparent paradox to speak about destruction in a conference devoted to Constructionism.

But, why speak about destruction in a communication class within a design school? Are we not supposed, as pedagogues, moreover in an applied art school, to have a constructive approach, helping our students to build a better future through a design process?

There is no paradox, just a necessity. Our human systems are based on some pereniality, permanencies, and persistence of systems, models, behaviors, or customs. And this is why they give some permanence to the world so that we can move and live individually and collectively in a relative order and security. This order, a necessity not only to live, but to survive, might also be a danger and turns against living and surviving.

This is what I would like to demonstrate in this paper. I will first define the general context in which I give my class, and build from that the two basic "theorems" about reality I share with my students. I will then describe precisely how I illustrate one of them through an exercise I created, that illustrates quite simply and immediately both some theoretical issues – philosophical and epistemological – and operational ones, that allow students to eventually have a pacified relationship to people, situation and creation.

From Shannon to Plato

The first thing I do with my students, in my communication class, is to interrogate them on what is communication. I actually ask them to give me keywords in quantity. And they do!

After half an hour, I have written a lot of words down on my board, and even messy, the set of keywords gives a very large picture of what 21st century youngsters think what communication is all about. From the Internet to Wi-Fi, from writing to reading, from adds to politics, a lot of concepts related to "communication" are more or less on the board.

I then ask my students to narrow or to constrain their answers by considering "Communication" in a very generic or universal way. Or, to say it differently, what is common to communication, whatever the actors involved? To help them, I ask them to adopt the point of view of an amoeba[2]: what is communication from a mono-cellular point of view? You can imagine that a lot of words/concepts then disappear from the board (e.g. politics…), while others are proposed.

---

[1] Strate Collège Designers is delivering a European Master in Industrial Design, and has been rated as one of the 60 top international design school by Business Week (www.stratecollege.fr)

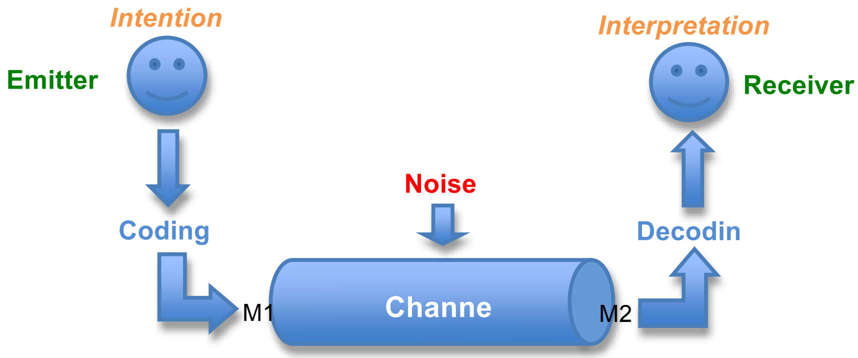
[2] The first difficulty being to explain to them, curiously, what an amoeba is.

After a while, I get some relevant words (more or less 100) that I ask them to categorize "freely" into 4 or 5 families, I actually more or less insinuate.

Very "obviously" and "naturally", they come out with categories that are labeled :

- Meaning
- Writing
- Coding/Decoding
- Noise
- Reading
- Interpreting

That leads us to draw on the board the classic communication diagram:



- The emitter has an intention;
- He codes his intention into a sequence of signs M1;
- M1 goes through a channel, and is possibly transformed due to some noise into a different sequence of signs M2;
- The receiver decodes M2,
- And interprets it.

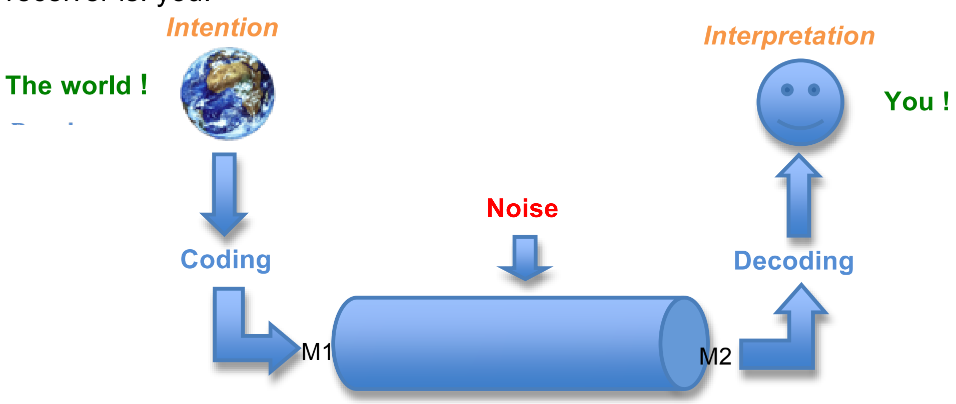
The diagram is a precious and fundamental tool. It helps me to explain what communication traps one must avoid : confusion.

Confusion between the emitter intention and my message M1, confusion between the message M1 and the message M2, confusion between the message M2 and the receiver interpretation. Those confusions are the origin of most human communication problems. Being aware of that is the beginning of a real communication wisdom.

Hello World !

If the hypothesis that the diagram is really universal, it is worth verifying the assumption. The next step is therefore to work on very special emitter and receiver:

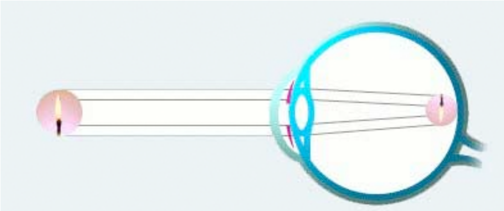
- The emitter is: the world!
- The receiver is: you!



It is then very interesting to verify, with the students, if and how the diagram is still functioning, starting – nonetheless – with the question: Is the world having any intention[3]?

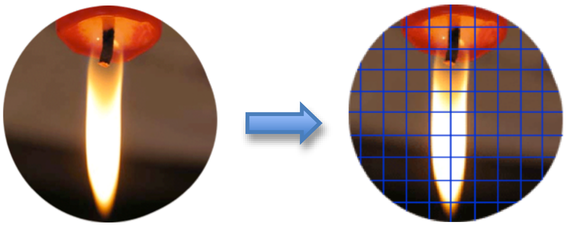
When it comes to Coding and Decoding (leaving aside the question of intention and interpretation, i.e. the question of meaning), the students very rapidly discover that we are speaking about perceiving the world through our senses.

Considering that all our senses are structurally functioning in a comparable way, we then focus on one of them – sight – to analyze "mechanically" the process of coding/decoding the world.



As in the preceding figure, we first illustrate the way the eye, as a machine, catches the image of a flame. We then focus on what is happening on the retina.

The reverse image of the flame is projected on the retina, as on a screen. The screen itself is composed of very little sensors (retinal rods)[4], which size defines the "resolution" of the retina, hence the smallest perceptible part of the image. Anything smaller than this retinal rod is literally invisible by the human eye, the given rod aggregating all the light information it receives into a single information.



The conclusion of this simple analysis is that the world our senses are perceiving is a pixelised world, a world where the order of magnitude of information sent is very significantly bigger than the order of magnitude of information perceived. The world, as we "see" it is nothing but a drastic simplification of the world per se: literally and precisely an abstraction.

This is the first theorem: the world is an abstraction[5]!

The lesson is: do not believe your senses, they are lying by omission, by constantly abstracting. Confusion, once again, is the ultimate sin: confusion between what we see and what is seen. As in the emitter/receiver general diagram we are facing the worst problems, with the world, with others, and with ourselves by ignoring this confusion.

The deep meaning of this theorem is that the world, in its completeness, is unreachable, unknowable, forever lost, if we have to rely just on our body. Our body is actually a cruel "reality diving suit", structurally and massively filtering information from the environment. We (me, all my students, and you) are moving in a world where an infinitesimal part of any situation is seen, where darkness is the rule, and where blindness is weakness.

---

[3] We will not answer that question today…

[4] I know that I am currently writing evidences that any distinguished reader already shares, but I am just describing the way things are said during the class. It is necessary to do this in order to bring this distinguished reader to the point of this paper.

[5] More exactly, the perceived world is an abstraction, but the first formula is definitely more provocative, hence pedagogically better!

From a prison to another

If we are prisoners of our senses, how can we free ourselves from this prison? This is the systematic and full of anguish question I get from my students as soon as they understand they are stuck into a “reality diving suit”. Through question and dialog I bring them to a known and simple answer: by building theories.

When our prehistoric ancestors, thanks to their observations, planned a hunt by taking into account the topography, the wind direction, the position of the sun, the vegetation, the systematic behaviors of aurochs, they are just theorizing, modelizing the reality, and anticipating the sequence of situation, by understanding and mastering causality.

When inventors of agriculture tried to plan all their campaign by understanding the time cycles – day, weeks, months, years – by observing the sun, the moon, and the wandering stars, they were building a total theory of the world, a cosmogony, thanks to which they were able to plan their action, for a greater good for the community.

When Isaac Newton, created his gravitation theory, he offered mankind the ability to predict with an incredible precision any movement of any object in the universe, either small or big as a planet, and hence to read the world as a gigantic mechanism. The industrial revolution is born from his theory, and from the Coca-Cola can to the “Eagle” Lunar Module, most human projects, either big or small, are based on his theory.

When Einstein, Schrödinger and others created the quantum mechanics they offered an unprecedented tool to understand (so to speak!), and use what is happening beyond our every day understandings. Our lives have deeply changed because of their discoveries, last but not least being the computer I am relying on today to communicate, learn, enjoy, live.

To summarize, Man has been able to go beyond the limits of his body and his senses by creating representations of the world, within which he can mentally play and anticipate events, and more than that create events, in order to not only survive but to reach given and wanted objectives. The ability to represent is at the heart of humanization, and has been the ultimate weapon to conquer the planet.

If primitive cosmogony is not comparable to general relativity, both are tools to read the world, navigate into it, and anticipate. They are representations of the world. They are “theories”, which etymologically means “look at things”.

With this ability to represent the world, and to use these representations, we might think that man has overcome the curse of the first theorem (*the world is an abstraction*), freeing himself from his reality diving suit, going far beyond the limit of his senses, no longer exposed to any limitation to his action and knowledge.

This is obviously false. If man actually freed himself from the prison of his sense, he has been creating a new and wider prison, even more dangerous because he cannot always see its walls. Theories are dangerous prisons as soon as we confuse them with the world they are supposed to describe: confusion again, confusion always. The danger of this confusion is even bigger because theories are so wonderful and efficient tools that we are pushed to identify the model and the world of phenomenoms they are supposed to modelize, because we hypnotized by this efficiency.

The fantastic and yet dangerous aspect of this confusion is that it is not only a mental and abstract one. It literally changes the way we “look” at the world. And by looking, I mean not only seeing through our eyes, but also reading it. Actually we tend to project what we believe, what we know onto the world.

This is the second theorem: *the world is a projection*.

This projection is a real limitation to our intelligence, because it caracterises, and even categorizes, every situation in order to fit in our vision of the world.

It has a direct effect not only on the way we think the world but even on the way we are looking at the world. Theory is actually not only a look at things. Very often it is a thing too! A chair, for instance, is mostly a theory of seating before being a material object made of wood and fabric. We never see it as the last, but always for the function it bears. Just try to give a chair to someone, holding it quite high, and ask him/her to put on the ground. The chair will always finish on its feet, rather than on any other position!

This can lead to the most violent and brutal situations, generated by the intolerance (in "good faith", and that is the problem) induced by the ultimate confusion between theories of the world and the world. Not only is it a great danger for man, but it is a sin for a designer. Indeed, how can a designer pretend to innovate if he cannot have a fresh look at the world?

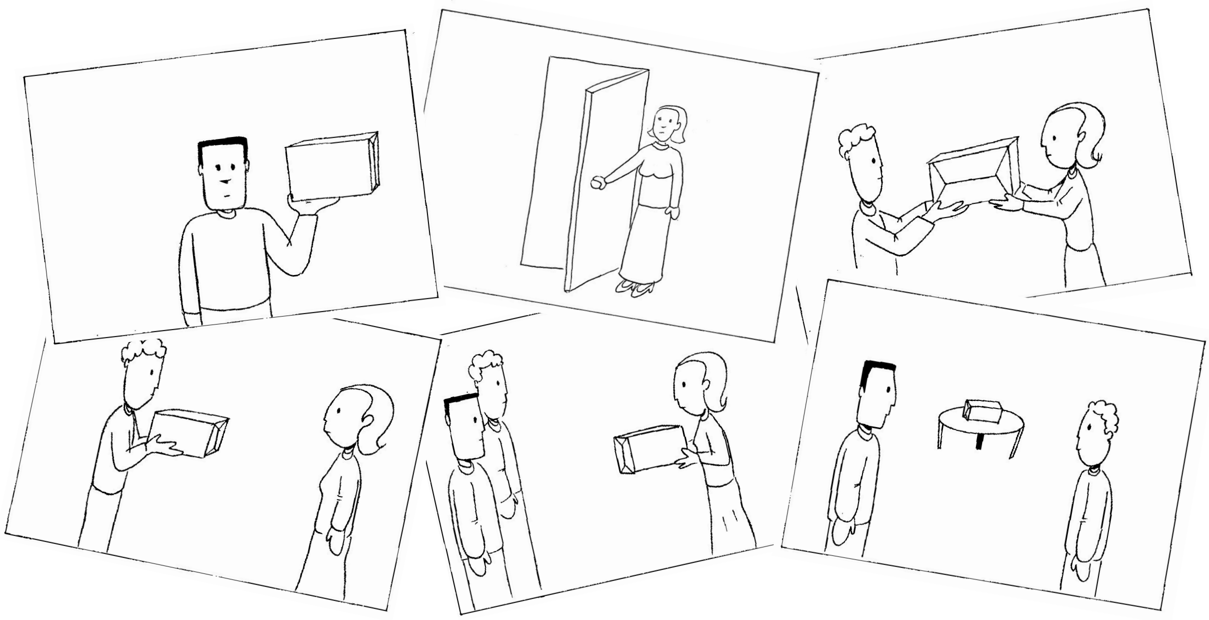
This is the conclusion I come to with my first classes: 2 "theorems" (The world is an abstraction – the world is a projection). And a promise: I am going to frame them whenever I want, even if they pretend to have understood the lesson.

And this demonstration is made using an exercise I created, and which I am going to describe now (at last!).

This is the story of six images…

Take a class of young people (but I am pretty sure that the result will be the same with any kind of participants) and give them the following brief.

1. Ask them to create groups of 3: it is necessary to create a group, because it induces naturally some behaviors related to social norms. Each group must have some intimacy in order to create with no disturbance from others.
2. Give them this series of 6 images[6]: (actually, it is 3 sets of 6 images that are given to each group).



[6] All images have been drawn by Laurent Sciamma, when he was 15. He is now a « Mad Man » to be….

At this point, it is very important to precise that the order of presentation of those 6 images is meaningless. When presented to the students, it must be emphasized that for now, these images are to be considered individually, i.e. not as a narrative sequence. They could actually have been presented in any other way.

3. Ask them to create 3 different stories (one funny, one dramatic, one free, using those 3 sets of images by strictly abiding the following rules:
   - Create one story at a time by working together:
     - No one is working on his own
     - Only one set of images is used at a time
     - Talk to each other! Nothing, yet, is done through telepathy.
   - Use all the images, in whatever order
   - Do not change any expression of the faces
   - Do not draw any object, any decoration
   - Do not cut, hide, destroy any part of the images
   - Use all the possible comics code (phylacters, onomatopoiea, etc.)
4. Each group has one hour to imagine the first story, produce it properly, title it, and sign it.
5. …Lets have some fun…..: before reading the following, please, take some minutes, yourself to imagine what kind of story you might want to create using those 6 images. When done, continue reading the paper.…

What's in a box?

While the first set of images is used, I am not doing much, not helping in any way the groups, just verifying that the rules are strictly observed.

It is very interesting to notice that the student are quite enthusiast about the exercise, and that rapidly they find ideas that make them laugh, ideas they are quite happy with.

After one hour, I stop my students. At that moment, they are eager to have me reading their stories. In the contrary, I do not read any of their stories; I just tell tem: "**Now, I am going to tell you what stories most of you have been creating!**".

The stories are :

1. A dead cat is in the box
2. Gran' Ma's ashes are in the box
3. A bomb is in the box
4. Some dirty material is in the box
5. The girl's things are in the box (she is being dumped by one of the guys)
6. A sex toy is in the box

At that moment, the students laugh by looking at each other, because it seems to them that I am a psychic.

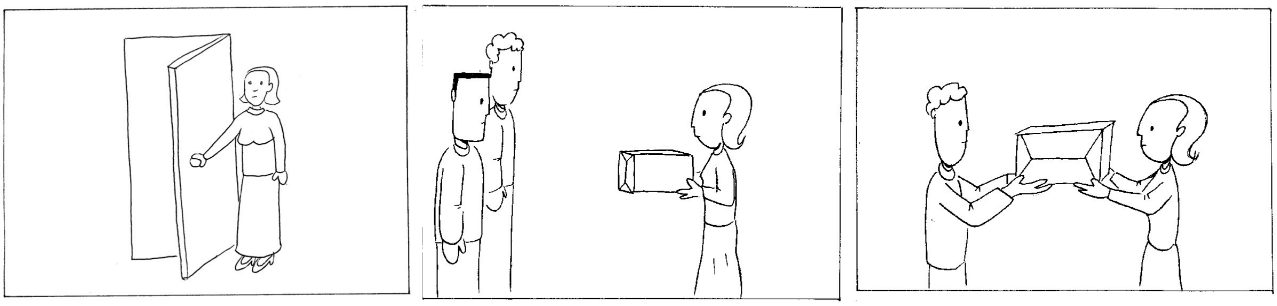
The question I ask them is: why are all of you – and your last year mates too - more or less telling the same story? Why is it so, since there are no others constraints than 6 images and the rules on how to play with them? Why are students, moreover future designers, and therefore supposed to be creative, not able to differentiate themselves in their creation?

To get the answer to that question, we go, with the students, into a short analysis of the images.

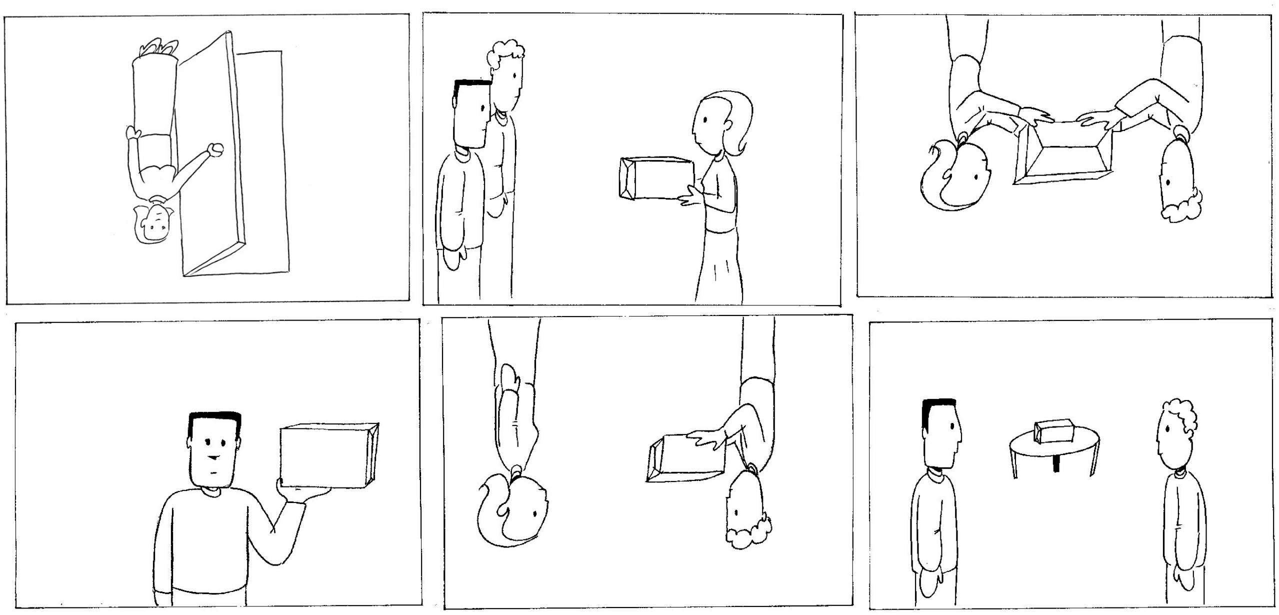
What's in an image?

To do it properly, I gather all images, by making sure - discreetly - that some of them are upside down. I then give the set to a student, asking him to put them on the table so that we can analyze them.

The result is systematically like that (order being meaningless again):



But never like this:



It means that the student who has been placing the images on the table has systematically put the images "right side up". I do not actually tell them directly that; I just ask them to analyze what their classmate did. Eventually, they tell me that he/she put the image in the "good" order (this is exactly the term they are using).

Good order? But what is the good order? The students' answer is: head up, feet down. But why is this the good order? Well, because this is how we live reply the students… My answer is: it depends! If you are an astronaut, there is no up and down. If you are working for UHU Sticks, you might test the quality of your glue the all day long by being stuck to the ceiling most of the day!

It is quite clear that the reflex of putting all images right side up denotes a projection of a vision of the world - our everyday world - where feet are on the ground. By doing so, it is clear that the students are cutting off all combinations of images with an "odd" orientation, hence forbidding themselves to imagine definitely funnier stories.

I then ask the students to analyze the content of the images. The answer is systematically:

- 3 characters :
  - 2 men (one brunette, one blonde)
  - 1 woman
- 1 box
- 1 table
- 1 door

3 characters?

Where the hell do you see 3 characters? I see **11** characters on those images!

The fact that you see 3 characters is related to the "theory" of sequential narration (comics, cartoons, movies), mostly based on characters pereniality. Since most of sequential narrations are telling stories about people, you are natively guessing that a similar representation of a character designates the same character. But it is not mandatory! We can perfectly imagine that there are up to 11 characters in this story. For instance, the story might hold on a clones' planet, where all the population has been cloned using 3 different human templates (kind of Jango Fet approach).

Actually, even if you consider that there are only 3 characters in your story, it is perfectly possible that the brunette in image *X* is the blonde from image *Y* with a wig.

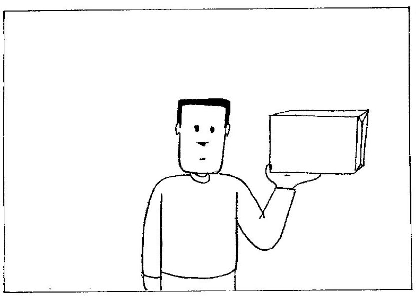
Once again, by doing so, you can access to a myriad of other possibilities, that would be forbidden if you unconsciously play with the rigid implicit rules of sequential narration.

A box?

In the same way, I might count 5 boxes on those images, and nothing can force me to think that it is always the same one.

And is it a box?

Why should I think it is a box? The rules of representation and perspective, mostly. It can perfectly be a wire whose shape gives me the illusion of a box, as in the following image:



The students' reaction to this last proposition is very interesting, and reveal more than anything else, the confusion syndrome we are all living in.

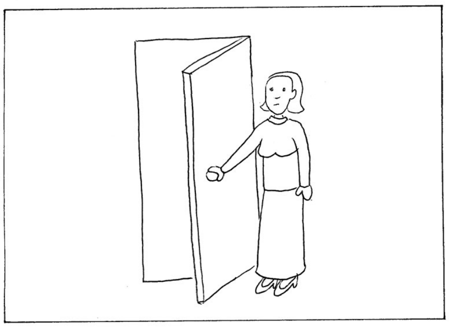
"*It is not possible that this a wire structure, because, if it was, we would see the fingers of the left hand of the characters!*".

But, what fingers are you talking about? This is just some ink put on a paper! And even if we consider the character as a human, he might perfectly have his left hand severed some time ago, and replaced by a special wooden device allowing to hold fake wire-based box!

Once again, we project onto the world what we know about the world, which lead us to see things that do not exist, and to interpret situation in a monovalent way, characters, boxes, and hands…

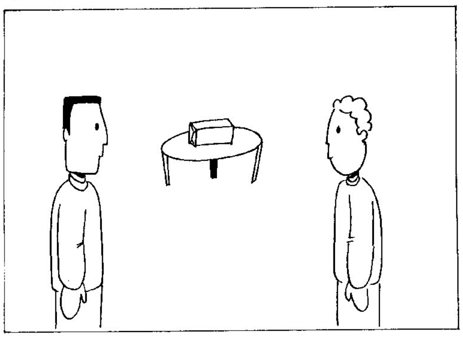
A door?

Really? A door? I don't see a door. But I might see a big vertical (and useless) guest book.



A table?

This is also an interesting illusion, and a composed one actually.



The reason we see a table is based on the conjunction of 2 facts:

1. There is "this" box (you know, THE box). The fact that we assume that there is a box makes us assume that the object we see in this image is this box.
2. This box being smaller in size, we assume that it is on a plan behind the character.
3. The object being a box, it seems placed on a surface of another object
4. The perspective makes us think that this object has 3 feet, one being in the shadow of the object itself
5. It is then necessarily a table!

But… It might be a small UFO (model Z-51, release 3.4), gently floating between the two characters, being in addition telepathically piloted by those 2 guys looking at each other hypnotically!

The dictatorship of images

This little exercise clearly demonstrates that, most of the time if not always, we are projecting theories on the world. Theories are always representations, and representations are images.

Indeed, in a game in which we are supposed to play with images, it is the images that are playing with us. In that case, it is the sequential narration theory that we project:

- Narration codes
- Representation codes
- Characters pereniality
- Ellipse

Ellipse is by far the most important projection we make. By filling the gap between two images, we create continuity in a discrete world, we give meaning to an empty space.

This is perfectly described in a wonderful book about comics, but which is definitely more than a book on comics: a book about narration, a book about communication, a book about semiology, a book about philosophy: **The invisible Art, by Scott Mc Cloud**. I highly recommend the book to any pedagogue.

Images vs Imagination

This is a lesson to anybody, any student, but what concerns me a lot, to any future designer.

The material world is full of theories, incarnated into objects. Objects, as such, are also representations and, literally, images. The fact is that objects – and more than objects, systems of objects - are created by designers. The way designers are looking at things is therefore critical. As soon as they forget that objects are materialized theories, they will not be able to reinvent them.

That's what creativity is all about. Playing freely with theories, which means, paradoxically, to know the theories. Actually, how can you break the rules of a game if you do not know its rules? By projecting what we know about the world onto the world, we confuse it with its representation, and we limit ourselves to what we know rather to what we can imagine.

Happy ending

If you remember, in the rules listed before, it was asked from the students to create one story at a time. The lesson on the dictatorship of images came after they wrote only one story.

So, after the explanation, I ask the students, now that they know how to break the rules, to astonish me in creating the two additional stories. And they do!

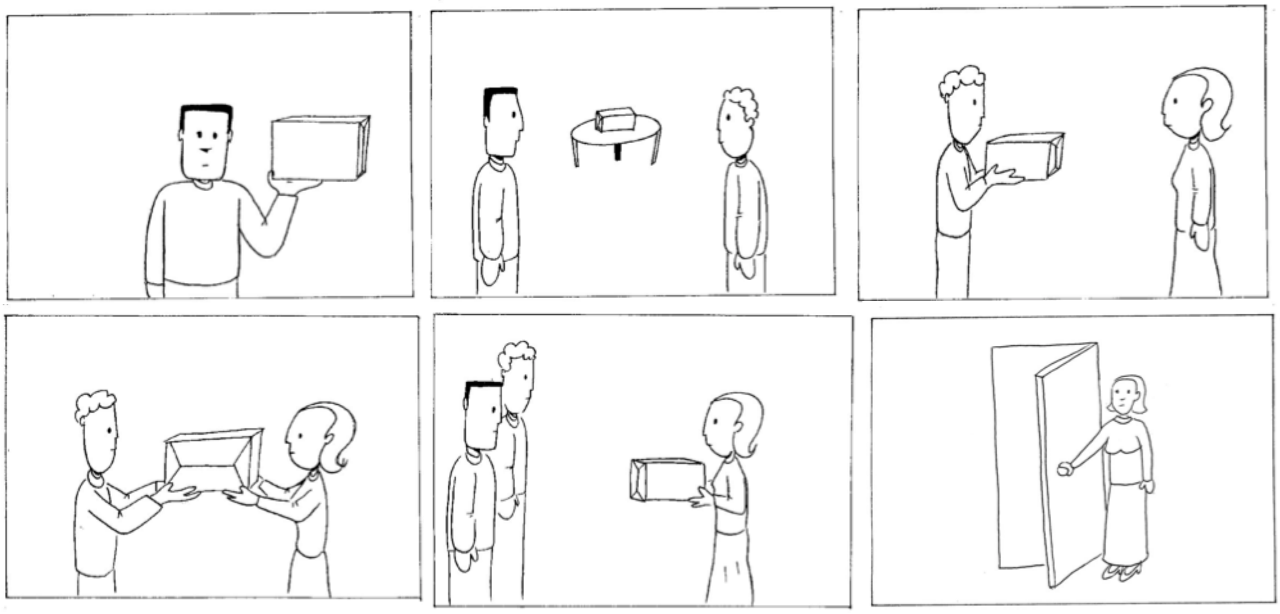
Every dimension of the theory might then be used:

- Think in 3D. Do not limit the space of narration to 2D. Images can perfectly be arranged in volume.
- Give new meaning to some drawing characteristics. For instance, it seems that all the characters are wearing mitten. This is worth using it!
- Change the situation with some comics code signs, such as some movement signs
- Etc[7]…

As an example, let me tell you a story of my own, where I "hack" some cinematographic concepts such as the medium close shot ("*plan américain*" in French).

[7] I am not going to give you all the « tricks » to tell brand new stories !

In a medium close shot, you cannot see the feet of the characters. As you can see in this sequence, all of the images, but the last, are medium close shots. This allows me to tell you this story.



It is a story happening on a planet where people have no feet. Two male friends have decided to offer their female friend a brand new Hi-Tec product: a pair of feet. This is exactly what is in the box. The girl is so happy, and the product so user friendly, that she decides to test her "feet" on the spot!

Conclusion: It is just the beginning

This important and simple exercise is one among a dozen others, all supposed to help the students to set them free from the prisons of senses and representations.

People that free themselves from the dictatorship of images have a name: iconoclasts, which literally means "Image breakers". In the 8th century, in what is today modern Turkey, Iconoclasts were destroying all images in churches, in order to avoid the adoration of images.

Considering all the "images/theories" we produce or consume, today more than ever, this is what teachers, students, professionals, citizens must be: iconoclasts.

Reference

McCloud, Scott.. (1993) *The Invisible Art* – HarperCollins

Thank you

Laurent Sciamma, James Clayson (American University of Paris), David Ferré (Strate Collège Designers), Sylvie Tosolini (Strate Collège Designers), Christine Chateauminois (Strate Collège Designers), Frédérique Pain (Alcatel Lucent Bell Labs), Arnaud Gonguet (Alcatel Lucent Bell Labs), Fabrice Poussière (Alcatel Lucent Bell Labs), Ioana Ocnarescu (Alcatel Bell Labs/Strate Collège Designers), Arnaud Lecat

Painting like Mondrian

Maria Skiadelli, *skiadelli@gmail.com*
Dept of Electrical and Computer Engineering, NTUA, Greek public school teacher

Short presentation

In this poster we present a LOGO made microworld that may serve as an interdisciplinary educational platform for Art, Maths and Computing. The microworld was originally created in a classroom of children between 12-14 years old, as part of the ICT curriculum. The aim of this activity was to help students find ways to produce their own drawings by imitating Mondrian's unique painting style. Since Mondrian's paintings are characterized by geometry (horizontal and vertical lines that form rectangles, some of them filled with colour), students have also the opportunity to study what is the impact of math functions and/ or geometrical shapes on the appearance of a painting and how they can change its various visual effects just by playing with the variables and parameters of these functions or shapes. Also important to the aesthetics of the painting is the idea of randomness, which may be applied to certain points of the drawing procedure (the distance between lines, the colours of the painting, etc.). From the programming point of view, children have the chance to take decisions at a technical level and see the results at the drawing level.

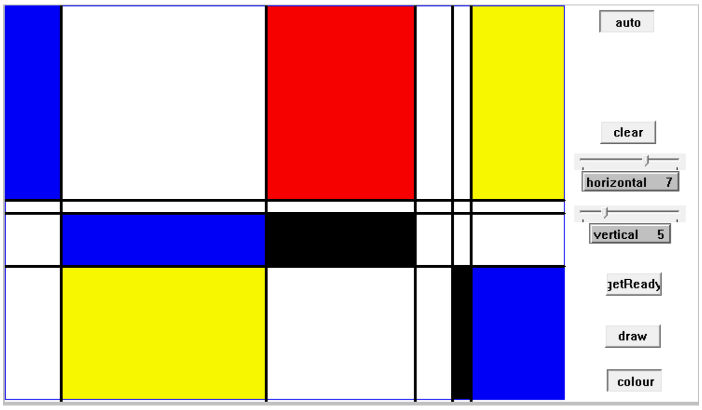


Figure 1. Painting like Mondrian microworld screen caption

Another interesting feature is that each student (and teacher) has adequate degrees of freedom to produce his/her one tool of painting depending on the decisions that he/she will take both at the mathematical and at the technical level. The same activity or scenario can lead to a variation of implementations (mircoworlds), according to the taste of each student or teacher. The microworld exists also in a second version to be used by children of younger ages (6-10).

The idea behind this microworld as mentioned earlier, was to create an interdisciplinary platform that could be exploited by teachers of ICT, Maths and Art, enhancing the dialogue between them and their disciplines, whilst giving the opportunity for co-teaching. Finally, the inherent rhythm that Mondrian's paintings posses, gives us the motivation to further develop this microworld in the future and expand its range of disciplines by including also music education.

Keywords

Microworld, Art teaching, ICT teaching, Math teaching, inderdisciplinary

A Constructionist Approach to Teaching with Robotics

Gary S. Stager, Ph.D., *gary@stager.org*
Graduate School of Education and Psychology, Pepperdine University 21825 Barbara Street Torrance, CA 90503 USA

Abstract

This paper presents four case studies representing the use of LEGO robotics materials and MicroWorlds EX Robotics programming by learners from five years old to mid-career postgraduate educators representing a variety of communities and prior school success. The robotics examples presented are more whimsical, playful and gender neutral than the traditional "battlebots" and vehicles dominating much robotics instruction.

Nearly two decades of using robotics in a constructionist context as inspired by Seymour Papert led the author to propose a new pedagogical theory, "A good prompt is worth 1,000 words." When the four critical factors in this approach are in place, learners are able to develop projects more sophisticated than those resulting from traditional curriculum and instruction.

During the conference presentation, video of the case studies and similar student projects will be presented in order to illuminate the powerful ideas contained within this paper. This work provides provocative ramifications for the successful teaching of robotics and implications for all teaching.

Keywords

Robotics, constructionism, Seymour Papert, school reform, Reggio Emilia, project-based learning

Introduction

LEGO robotics has enjoyed a reasonable level of popularity in classrooms over the past twenty years. The flexibility, durability, ease-of-use, familiarity and brand recognition of LEGO construction materials have made enormous contributions to the viability of robotics in education. Without LEGO's contributions to the educational marketplace robotics would remain the domain of science fiction or post-graduate education.

Robotics involves aspects of mechanical and electrical engineering as well as computer science. A working robot required both construction and programming. Therefore, a variety of student expertise, interests and learning styles are supported. Teachers otherwise reluctant to learn or teach computer science are often attracted to the tactile (hands-on) nature of LEGO robotics. Such hands-on activities often facilitate minds-on programming, complete with its requisite problem solving, debugging and mathematical thinking.

Principles of robotics engineering are neither the primary or secondary objectives of using robotics materials in K-12 education. Robotics principles may be learned at ages younger than previously anticipated, but such understanding is incidental to the use of robotics as material for constructing knowledge. (Papert and Franz, 1987) The sorts of learning made possible by robotics activities are often greater than the sum of its parts. A variety of school subjects are integrated while serendipitous connections to powerful ideas and forms of creative expressions are commonplace in the pedagogical approach described in this paper. The case studies presented represent a departure from traditional teaching practice.

Five Ways to Use Robotics in Education

There are at least five general approaches to the use of robotics in education. Each approach has its own objectives and requires different levels of teacher intervention.

Robotics as a discipline – Robotics is taught as its own discipline. Popular school age robotics competitions, such as the FIRST LEGO League, are examples of this approach.

Teaching specific S.T.E.M. concepts – Robotics may be used to teach physical science concepts such as: simple machines, force, torque, power, friction, mechanical advantage; computer science concepts of programming, debugging and feedback; mathematical concepts of fractions, variable, arithmetic operations, etc.…

Thematic units – Students build and program robots to model machines and systems such as airports, factories, amusement parks or a city. The hope is that traditional school subjects and concepts are experienced or embedded in these themes.

Curricular themes – Robotics is used as a medium for solving specific problem connected to a formal curriculum topic. An example might be, “Identify a problem in Sub-Saharan Africa and build a robot to solve that problem.” The realism of the solution may be subordinate to thinking about the nature of the problem.

Freestyle – Robotics materials and computer programming are used as construction material as part of a student's intellectual laboratory and vehicle for self-expression. The learner may use the materials to make anything they wish. Powerful ideas are experienced within the context of the activity.

Constructionist Approach

A detailed discussion of Papert's theory of constructionism is beyond the scope of this paper. However, the following case studies are based on the principles that knowledge is constructed and the best way to ensure learning is through the deliberate construction of something shareable outside of one's head. (Ackerman, 2001; Papert, 1993, 1991; Papert et al., 1991; Stager, 2002; Stager, 2007; Turkle & Papert, 1991)

The role of the teacher is to create the productive context for learning, including material organization, scaffolding, consulting, collaboration and anticipating forthcoming needs of each student. Teaching is subordinate to learning and the teacher is available to seize teachable moments and collaborate with students, rather than direct activity.

Four Case Studies of Robotics Projects

Ballerina

A five year-old kindergartener in an underperforming school expressed interest in being a ballerina. While this is hardly a unique aspiration of young girls, I suggested that she might be able to build a LEGO ballerina. "Anna" quickly set off to build a spinning mechanism as the core of her ballerina. She then took great care in creating a dress out of a paper napkin decorated with colored markers and made hair for the ballerina out of pipe cleaners. Such creative expression is consistent with Papert's "computer as material" (Papert & Franz, 1987) metaphor, Reggio Emilia's use of mixed media as a vehicle for personal expression (Topal, et al., 1999) and the Piagetian notion of "objects to think with." (Papert, 1980a) It also contributed to an awfully cute robot.

Since a ballerina needs choreography, I showed Anna how a touch sensor could be used as a switch telling a Logo program to do something new. She decided that two touch sensors could be used and programmed to make her ballerina spin left or spin right. While one switch might have been sufficient for changing direction, holding a button in each hand felt more consistent with commanding the ballerina robot as if it were a marionette.

The entire project took two or three morning sessions to complete.

Anna was justifiably pleased with her creation and it became a favorite project of her older classmates. In a video clip captured during the project, the school principal asks Anna "And you did this with your computer?" at which point, the little girl confidently makes a modification to her Logo program with a nonchalance unusual for five year-olds speaking with adults. Also on video you can see Anna whistling and spinning her head synchronously with the ballerina while working its controls. This is a demonstration of the syntonic body geometry on which Logo's turtle graphics system is built; a learner makes sense of the world and powerful geometric ideas by relating such concepts to their physical motion.

Teddy Bear

A group of third grade students from an affluent private girls school worked with me for four consecutive mornings (approximately three hours per session). A consensus was reached for teams of students to work on inventions one might find at a state fair. One group of girls decided to bring a teddy bear to life by making it dance.

Once I made an incision in the bear's torso, the girls set about building a skeletal system capable of making the bear dance. Three to five students approached the task with great enthusiasm and focus. An immediate challenge was translating the rotational motion of a motor into the up and down movement required for dancing limbs. The constraints of the stuffed bear's limbs helped constrain the arms and legs and approximate dancing gestures.

As in other robotics projects, bugs required problem solving and alternative strategies while each successful breakthrough led students to set more complex challenges or grander theories to test. The dancing teddy bear was no exception.

Having achieved mechanical animation, the girls asked if the bear could also be "taught" to sing. As quickly as I was able to tell them that the LEGO programmable brick was capable of playing a simple single-note melody, one student rushed off to the music classroom to borrow a piece of sheet music.

Armed with the sheet music, a new challenge emerged. The expertise of team members capable of reading music was called upon. Then the musical notes and rests had to be converted to numerical values representing frequency and duration and programmed via MicroWorlds. The singing was added to the dancing procedure and a new Logo superprocedure was downloaded to the RCX brick functioning as the bear's attached brain. Once the singing was satisfactory, their main program would need to be modified to sequence singing and dancing.

When asked to sing, the music played so quickly that the folk melody was barely perceptible. Faced with this bug, the girls needed a solution. One student noticed that the melody sounded correct, but too quick. Since there is no knob for adjusting musical speed on the computer or RCX, a programming solution would be necessary. The girls collaboratively arrived at a solution. They needed to multiply each of the duration values by a constant, hence slowing down the melody. (They may have experimented with increasing both variables, pitch and duration, before realizing that only one needed to change.)

Isolating the correct variable and multiplying it by a constant is a nice piece of mathematics for eight year olds, especially when you consider that it is during that grade level that most children are tortured by the rote memorization and recall of multiplication tables. These students demonstrated a working understanding of multiplication, variable, music notation, computer programming and both the physical science concepts and affective skills gained during robotics projects.

Phonograph

The next project description is of a robot built and programmed by incarcerated fifteen year-old who like most of his peers, was diagnosed with a variety of learning disabilities. He also had a poor record of school success in addition to truancy.

This teenager was inspired to recapitulate the invention process of Edison by creating a working phonograph without access to Edison's work, life experience or laboratory. The student's primary motivation was not to construct a phonograph, but to build "something hard, something nobody has ever done before." This is a remarkable stance for any young person, even more impressive when you consider this student's prior poor academic experiences. Constructing a sophisticated robot was a way to assert his competence as a learner in a school setting just different enough to inspire such innovation. It is also critical to understand that the phonograph was the first robotics project ever engaged in by this at-risk learner.

Young Edison's "school" was the alternative high tech constructionist learning environment Seymour Papert and I created inside of a state prison for teens in Maine, USA. The Constructionist Learning Laboratory (CLL) provided a computer per child, a rich variety of material with which to construct and sufficient time to work on substantive projects. Having been liberated from curriculum and assessment requirements by the Governor and Secretary of Education, the CLL was able to put the needs, interests and talents of severely at-risk students ahead of a traditional, albeit arbitrary, scope and sequence.

The robotic phonograph is an important example of a constructionist approach to robotics and the use of LEGO materials. The student used non-LEGO elements, constructed an invention from a simple prompt and developed the vocabulary for talking about his work. He *became* Edison and invented the phonograph for himself. In the process he came to understand gearing, computer control, transforming vibrations into sound and the satisfaction that accompanies a sound engineering effort. The narration demonstrates his understanding of gear ratios, the use of a microscope and even an appreciation of margin of error in his description of the device The student learned about gearing, sound amplification, magnification and a host of other big ideas valued by educators concerned with traditional notions of curriculum value such concepts. (Stager, 2007a)

Project goals were intrinsic to the learner, There was no hidden curriculum or expectation that by building a phonograph each student will demonstrate an understanding of X or Y curricular objectives. No attempt was made to institutionalize this student's experience by compelling his classmates or future classes of students to build a robot phonograph. Young Edison's teachers had confidence that the use of such materials in the type of constructionist learning environment created would lead to the development of powerful ideas – even if some of those ideas were impossible to predict or the finished product imperfect. Best of all, the invention was original, conceived, constructed and programmed by a student.

Adult professional development

Robotics not only captures the imagination of children, but also provides a terrific context for educators to explore the power of learning technology in a playful, tactile and non-threatening fashion. For more than a decade I have employed the same pedagogical strategy for "teaching" robotics to teachers. The approach employed is similar to the way in which I introduce robotics to children. Adults are asked to form project teams of two to four. However, since professional development time is in shorter supply than classroom time, brainstorming project ideas is a luxury one can rarely afford. Frankly, adults in a workshop or graduate course are less likely than children to share their imagination or whimsy prior to experience with new technology.

Therefore, I begin my adult workshops by asking each team to pull an open-ended project idea[i] "out of a hat" and immediately get started trying to solve the challenge stated on the sheet they chose. Each project sheet contains a one line prompt and an extension problem for more ambitious teams. Other craft materials and props useful in the challenges are available in the classroom.

Prompts might include:

- Build a robot card dealer that deals a hand of playing cards
- Invent a machine to walk a dog

Create a robot capable of playing a song on a xylophone or percussion instrument

Design and program a working chairlift or gondola

Construct a machine capable of blowing soap bubbles

Extension activities might be to have the chairlift drop a paratrooper on command or attach the bubble machine to a moving vehicle. The whimsical nature of these challenges makes the activities more gender neutral and respective of a plurality of personalities and learning styles. (Bers, 2007; Rusk & Resnick, et al. 2008; Resnick, 2006; Resnick & Ocko, 1991)

Adults are encouraged to stop working and visit with other teams in order to learn from their colleagues and share expertise. Children do not need such reminders since the routinely explore the work of their peers, a phenomena called "collaborations through the air" by Yasmin Kafai. (Kafai & Harel, 1991; Kafai 1995; Kafai & Resnick 1996) Throughout the project development activity, participants are asked to remove their teacher hats and think about thinking – their thinking and that of their colleagues. Possible lessons and implications for teaching practice are discussed after participants experience the successful, if novel, learning adventure.

Repeatedly teachers in conference workshops, Pepperdine University graduate program orientations (Cannings & Stager, 2003; Stager, 2005) and most recently a project to teach Brooklyn, New York middle school science teachers to integrate robotics into their curriculum marvel at their successful work in an unfamiliar domain free of didacticism.

Commonalities

In most cases, I introduce robotics to the class by showing them the motors and sensors that are part of the LEGO RCX materials. I explain how to turn on the RCX brick, the sensor inputs and the motor outputs. The class is shown the infrared tower used to transmit programs from the computer to the programmable brick and that is about it. That five-minute presentation is about the extent of the formal robotics instruction. Two sets of pictorial engineering reference materials, one created by Fred Martin (Martin, 1995) and the other by MIT Media Lab students (The Art of LEGO), show students how various structures are built with LEGO and are made available in the classroom. A two-page MicroWorlds EX Robotics programming reference is also provided for students or workshop participants.[ii]

In all four cases presented, students of all ages, socio-economic status and academic achievement were able to invent, construct and program extraordinarily complex robots ***the first time*** they used the materials. How can this be true? A learner might be unable to create projects of similar sophistication after having completed a formal robotics curriculum lasting a year or more?

A teacher impressed by the ingenuity of a successful student project may be inclined to institutionalize a particular activity. It would be a mistake to require every student to build his or her own phonograph or add “dancing teddy bear” to the curriculum. Such pedagogical practices are found in science curricula that require every student to repeat identical experiments for decades and in the robotics teachers who come to tell me how their class “just built the traffic light.” In 1987, LEGO published several step-by-step tutorials designed to help people learn to use LEGO TC Logo. Nearly two decades later, what were once provided as mere examples have been chiselled in stone as sacred curriculum.

Learning cultures built upon the principles of this paper’s pedagogical approach require educators secure in their knowledge that with each project triumph or “bug,” the community gets smarter as the collective student expertise increases.

In each case there was no direct instruction, no model plans, no step-by-step instructions, no online tutorials, no formal assessment, no extrinsic motivation and no online access. After close to twenty years of teaching children and adults about and with robotics, learner after learner has been able to create impressive machines without traditional teaching. Such counterintuitive results required the construction of an explanatory theory.

Emergence of A New Pedagogical Theory

My experience suggests that the successful project development described in this paper’s four case studies is based on four critical factors:

A good prompt

a personally meaningful and motivating question, challenge or prompt

Appropriate materials

availability of an assortment and ample quantity of construction materials allowing a learner or team of learners to build something they’re proud of and leave it assembled long enough for others to admire to learn from it.

Sufficient time

quality work takes time and students deserve an opportunity to experience a level of project “completeness” and the satisfaction that comes from accomplishing one’s goals

Supportive culture

a non-coercive, collaborative, non-competitive environment facilitates risk taking, inspires reflection, stimulates inquiry and rewards creativity

When these four factors are present, students are capable of exceeding their own expectations and learning a great deal along the way.

This pedagogical approach is not restricted to robotics or computer science. However, the number of disciplines, modes of interaction and individual learning styles expressed within such projects makes robotics particularly compelling. Teachers observing students working in the contexts described or while assuming the role of students themselves begin to see robotics as a learning lens for reflecting on their own practice.

A Good Prompt is Worth 1,000 Words

“A good prompt is worth 1,000 words,” is the way I describe the open-ended learner-centered approach to teaching with robotics. While each of the four critical factors appear simple and self-evident, traditional schooling too often creates significant obstacles to creating the productive context for learning about robotics and more importantly, learning *with* robotics. This is not an excuse for not teaching robotics or for limiting the intellectual potential of students.

More importantly, these four critical factors may have implications for all project-based learning and challenges the hierarchical approach to curriculum typically employed by schools. In other words, when these four factors are present, students may learn more than is traditionally expected of them.

Students served by a constructionist approach grow in ways beyond the typical goals of a robotics project. An at-risk student raises their personal educational standards when in a rich environment that places their talents, needs and expertise ahead of standard curricular requirements. Anna’s ease and confidence in speaking with an authority figure, like the principal, may be the result of the collegial stance I maintain while teaching children with robotics materials. Robotics provided a gateway to literacy for another at-risk teenager who wrote documented his invention process in a class newspaper and shared the results with a corporate CEO via the first letter he ever wrote,

Students not only encounter powerful ideas, but they are empowered as well. To critics suggesting that childhood innocence is robbed or creativity sapped by children using computers, the robot ballerina and dancing teddy bear are active antidotes to concerns over passive screen watching. (Cordes and Miller 1999) Children who may have urged parents to buy them an expensive mechanical toy now are empowered by the means of production to invent their own fanciful high-tech plaything. Math, science and engineering are brought to life with an artist’s aesthetic.

Broader Implications For Schooling

The examples presented serve as a challenge for educators to create the productive contexts necessary for learning across various knowledge domains and disciplines. A constructionist approach to robotics expands student potential over more traditional instruction and justifies the investment of time, resources and energy required to introduce a new medium for knowledge construction.

The lessons from the specific vignettes presented are less about how a young girl transfers knowledge constructed through the act of building a robot teddy bear to another school subject than what educators can learn about learning from careful observation of a student’s experience. Three lessons for educators of all subjects and ages may be distilled.

Memories are Made of Project-Based Learning

The pedagogical approach called project-based learning (PBL) affords students an opportunity to solve authentic personally relevant problems in a context in which they make connections between different disciplines and employ a variety of skills. Effective PBL experiences may precipitate a need for structural changes in scheduling, curriculum and assessment while teachers recognize that the richest learning emerges from an environment adhering to the principle of “less us, more them.” [iii]

When Ideas Go to School, They Lose Their Power

The four critical factors described earlier may appear simple, but are deceptively difficult for schools to realize. This failure may result from exterior pressures on “schooling” or a lack of understanding of learning in general or specifically, the subtleties of project-based learning. Many teachers yearning for the benefits of project-based learning perceive an inability to change expectations, their role or the context of the classroom in the ways required by the four factors. These “yearners” (Papert 1993) compromise by skimping on time dedicated to a project, limiting collaboration or narrowing the objective of the project. For too many schools, project-based learning means any non-lecture-based activity. Little more than an illusion of freedom is created when students are allowed to explore for a short time before being expected to produce the “right answer.”

Much ado about challenges

One example of new pedagogical wine in old bottles is “Challenge Based Learning.” (CBL) Apple Computer is the primary advocate of what they consider a new pedagogical approach, even if CBL bares an uncanny similarity to time-honored notions of project-based learning. (Apple 2008) CBL errs in what Papert might consider instructionist ways with its overreliance on teacher-specified topics, rubric-constrained parameters and a narrow range of experiences, including virtually no computation. The CBL emphasis is on information retrieval, synthesis and presentation like most school activities.

Paradoxically, it is the very enormity of the CBL prompts that constrains the potential for authentic learning. Asking students to solve problems that may not be solvable by the collective efforts of the world’s smartest people for generations, is not an authentic context for learning. As a result, the range of experiments, paths of inquiry, tinkering and even complexity are sacrificed. When faced with impossible challenges, limited resources, insufficient time and looming assessment, students respond in a rational fashion. They produce digital movies, posters, presentations and other forms of “reports” containing the politically correct and overly simplistic conclusions their teachers expect. Sentimentality is substituted for intellectual depth.

Children are certainly competent and capable of making genuine intellectual and creative contributions to the world. They can and do solve important problems. Some play make beautiful music or exhibit athletic prowess. It is however foolish to expect children to solve the world’s problems, like famine, global warming or improving “the air that you breathe.” Even if this critique of CBL is a bit harsh, computer use in the majority of its challenges is quite inconsequential, particularly when compared with the sorts of activities advocated by members of the constructionist community.

Curriculum

Curriculum should neither be coercive not exploitative. When asked about the changes in schooling required to support project-based learning, Papert replied, “Well, first thing you have to do is to give up the idea of curriculum. Curriculum meaning you have to learn this on a given day. Replace it by a system where you learn this where you need it. So that means we're going to put kids in a position where they're going to use the knowledge that they're getting. So what I try to do is to develop kinds of activities that are rich in scientific, mathematical, and other contents like managerial skills and project skills, and which mesh with interests that particular kids might have… I imagine the learning environment of the future as we've given up the idea of there being curriculum that says you have to learn this at the seventh of May in your eighth year…. Kids will work in communities of common interest on rich projects that will connect with powerful ideas." (Papert GLEF 2001)

Assessment

Another enemy of constructionism is the perceived need for assessment and the public's increased demand for measurement. Newer pedagogical strategies, such as rubrics, appeal to some educators as a form of alternative assessment since numerical grades are replaced by seemingly more subjective criteria. However, "The Trouble with Rubrics," (Kohn 2006) demonstrates how rubrics preserve the coercive elements of traditional grading schemes, reduce student motivation and curtail possibilities for serendipitous learning.

Any attempt to test, measure or quantify what a student knows is intrusive and disrupts the learning process. There is no substitute for teachers really knowing their students and assuming a responsibility for investigating their thinking, as exemplified by the preschool teachers of Reggio Emilia. Such teacher researchers make each child's thinking visible so it may benefit the learner, their peers, parents seeking demonstrations of progress and teachers responsible for creating the next intervention required to propel student thinking and learning.

Much of the success experienced by children in Papert's "prison project" (Stager 2007a; Papert 2000) was possible because of our exemption from all state assessment and curriculum requirements. "I think that this project allowed some of them [students] to get a new sense of themselves as learners -- that learning is something valuable, that setting yourself a goal and working to achieve it is something which some of them have never seen before in their lives. They've never known anybody who works over time for the achievement of some goal. So you can change their view of life." (Papert in GLEF 2001)

Educational innovation remains elusive without a willingness to challenge, revise or even abandon curriculum and assessment. The quest for better forms of assessment and curriculum may distract educators from creating productive contexts for learning in which students may achieve more than they or their teachers are capable of anticipating.

"For those of us who want to change education the hard work is in our own minds, bringing ourselves to enter intellectual domains we never thought existed. The deepest problem for us is not technology, nor teaching, nor school bureaucracies. All these are important but what it is all really about is mobilizing powerful ideas." (Papert 1998)

Kid Power

Schools remain plagued by Dewey's century-old criticisms of school's overreliance on curriculum and scarcity of authentic experiences. (Papert 1998a) Papert suggests that the computer and kid power, the fundamental building blocks of constructionism, allow children to drive learning rather than be the passive recipient of teaching; shift agency to the learner; and introduce powerful ideas to even young children without losing their power. (Papert 1998a)

"We have to look at different kids differently. The most common element with all kids is that they start off as enthusiastic learners, but by the time they have been in school for a few years they have stopped being enthusiastic about learning. The learning instinct is strangled. That makes their lives poorer. It makes society poorer. It makes the economy rigid and inflexible. It makes for a more rigid society all around." (Papert in Bennahum 1996)

The children responsible for the learner-centered robotics projects described earlier were engaged in sophisticated knowledge construction unconstrained by the curriculum or adult preconceptions of final products or measurable achievement along the way. Other domains may be explored with different tools and material in a similar spirit if educators assume a constructionist stance. The computer remains the protean material for making, learning and doing.

Children have a remarkable capacity for intensity. It is incumbent upon educators to leverage this gift in order to support children in exceeding their potential. Too often, our focus on achievement, assessment and curriculum coverage creates an artificial ceiling. Having the courage to behave as if learning is natural and children are competent liberates students and teachers to achieve in powerful ways and at a level previously unimaginable.

References

Ackerman, E. (2001). *Piaget's constructivism, papert's constructionism: What's the difference?* Paper presented at the 2001 Summer Institute, Mexico City.

Apple (2008). "About Challenge Based Learning." Retrieved June 1, 2010, from http://ali.apple.com/cbl/.

Bennahum, D. S. (1996). "School's Out: Interview of Seymour Papert." Meme 2.13. Retrieved June 1, 2010, from http://memex.org/meme2-13.html.

Caldwell-Edmonds, C. The concept of emergent design [web site] *The concept of emergent design* [web site] Retrieved June 20, 2003, from http://www.techlearninglink.org/Emergent%20Design%20and%20Decentralized%20Models/Emergent%20Design.htm.

Cannings, T., & Stager, G. (2003). Online constructionism and the future of teacher education. Paper presented at the ICT and the Teacher of the Future - Selected Papers from the International Federation for Information Processing Working Groups 3.1 and 3.3 Working Conference, Melbourne, Australia.

Cavallo, D. (2000a). Emergent design and learning environments: Building on indigenous knowledge. *IBM Systems Journal, 39*(3&4), 768-781.

Cavallo, D., Papert, S., & Stager, G. (2004, June 22-24, 2004). *Climbing to understanding: Lessons from an experimental learning environment for adjudicated youth.* Paper presented at the Sixth International Conference of the Learning Sciences, Santa Monica, California.

Cordes, C. and Miller, E. (1999) *Fool's Gold: A Critical Look at Computers in Childhood*, Alliance For Childhood, http://www.allianceforchildhood.net/.

Dunn, S., & Larson, R. (1990). *Design technology : Children's engineering*. New York: Falmer Press.

Edwards, C., Gandini, L., & Forman, G. (Eds.). (1998). *The hundred languages of children: The reggio emilia approach - advanced reflections* (Second ed.). Westport, Connecticut: Ablex Publishing.

Forman, G., & Fyfe, B. (1998). Negotiated learning through design, documentation and discourse. In C. Edwards, L. Gandini & G. Forman (Eds.), *The hundred languages of children: The reggio emilia approach - advanced reflections* (Second ed.). Westport, Connecticut: Ablex Publishing.

Gandini, L. (1998). Projected curriculum constructed through documentation - progettazione: An interview with lella gandini. In C. Edwards, L. Gandini & G. Forman (Eds.), *The hundred languages of children: The reggio emilia approach - advanced reflections* (Second ed., pp. 120-121). Westport, Connecticut: Ablex Publishing.

Gardner, H. (1998). Foreward: Complementary perspectives on reggio emilia. In C. Edwards, L. Gandini & G. Forman (Eds.), *The hundred languages of children: The reggio emilia approach - advanced reflections* (Second ed., pp. 7). Westport, Connecticut: Ablex Publishing.

GLEF (2001). Seymour Papert on Project-Based Learning. Expert Interviews. Edutopia, George Lucas Foundation.

Harel, I. (1991). *Children designers: Interdisciplinary constructions for learning and knowing mathematics in a computer-rich school*. Norwood, N.J.: Ablex Pub. Corp.

Harel, I. (1993). Who is a constructionist? Keynote address at the international logo conference. Kew, Australia: unpublished.

Kafai, Y., & Harel, I. (1991). Learning through design and teaching. In S. Papert, I. Harel & MIT Epistemology and Learning Group. (Eds.), *Constructionism: Research reports and essays, 1985-1990*. Norwood, N.J.: Ablex Pub. Corp.

Kafai, Y. B. (1995). *Minds in play: Computer game design as a context for children's learning.* Hillsdale, N.J.: L. Erlbaum Associates.

Kohn, A. (2006). "The Trouble with Rubrics." English Journal **95**(4): 12-15.

Miller, G., Church, R., & Trexler, M. (2000). Teaching diverse learners using robotics. In A. Druin, Hendler, James (Ed.), *Robots for kids: Exploring new technologies for learning*. San Francisco: Morgan Kaufmann.

Papert, s. (1980a). Computer-based microworlds as incubators for powerful ideas. In R. Taylor (Ed.), *The computer in the school: Tutor, tool, tutee* (pp. 204-210). NY: Teacher's College Press.

Papert, S. (1980b). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Papert, S. (1984). Computer as mudpie. In D. Peterson (Ed.), *Intelligent schoolhouse: Readings on computers and learning*. Reston, VA: Reston Publishing Company.

Papert, S. (1993). *The children's machine: Rethinking school in the age of the computer*. NY: Basic Books.

Papert, S. (1998). "Technology in Schools: To support the system or render it obsolete." Milken Exchange on Education Technology. Retrieved June 1, 2010, from http://www.mff.org/edtech/article.taf?_function=detail&Content_uid1=106.

Papert, S. (1998, June 2). *Child power: Keys to the new learning of the digital century [lecture transcript].* Paper presented at the Eleventh Colin Cherry Memorial Lecture on Communication, Imperial College, London, UK.

Papert, S. (2000). What's the big idea? Toward a pedagogical theory of idea power. *IBM Systems Journal, 39*(3&4), 720-729.

Papert, S. (2005). Beyond wires and boxes: Technology as a tool for educational transformation, *Presentation at COSN's K-12 School Networking Conference*. Washington D.C.: Consortium on School Networking.

Papert, S. (Ed.). (1991). *Situating constructionism*. Norwood, NJ: Ablex Publishing Corporation.

Papert, S., & Franz, G. (1987). Computer as material: Messing about with time. *Teachers College Record, 89*(3).

Papert, S., Harel, I. (1991). *Constructionism: Research reports and essays, 1985-1990*. Norwood, N.J.: Ablex Pub. Corp.

Rathjen, D. (1988). *Lego crazy action contraptions: A lego inventions book* New York: Klutz Press.

Resnick, M. (1993). Behavior construction sets. *Communications of the ACM, 36*(7).

Resnick, M. (2006). Computer as Paintbrush: Technology, Play, and the Creative Society. In Singer, D., Golikoff, R., and Hirsh--Pasek, K. (eds.), *Play = Learning: How play motivates and enhances children's cognitive and social--emotional growth*. Oxford University Press.

Resnick, M., Bruckman, A., & Martin, F. (2000). Constructional design: Creating new construction kits for kids. In A. Druin, Hendler, James (Ed.), *Robots for kids : Exploring new technologies for learning*. San Francisco: Morgan Kaufmann.

Resnick, M., & Ocko, S. (1991). Lego/logo: Learning through and about design. In S. Papert, I. Harel & MIT Media Lab Epistemology and Learning Group. (Eds.), *Constructionism: Research reports and essays, 1985-1990*. Norwood, N.J.: Ablex Pub. Corp.

Resnick, M., Ocko, Stephen. (1991). Xylophones, hamsters and fireworks: The role of diversity in constructionist activities. In S. Papert, I. Harel & MIT Media Lab Epistemology and Learning Group. (Eds.), *Constructionism: Research reports and essays, 1985-1990*. Norwood, N.J.: Ablex Pub. Corp.

Resnick , R. and Silverman, B. (2005) Some reflections on designing construction kits for kids, *Proceeding of the 2005 conference on Interaction design and children*, p.117-122, June 08-10, 2005, Boulder, Colorado.

Rusk, N., Resnick, M., Berg, R., and Pezalla-Granlund, M. (2008). New Pathways into Robotics: Strategies for Broadening Participation. Journal of Science Education and Technology, vol. 17, no. 1, pp. 59-69.

Stager, G. (2002, June 18, 2002). *Papertian constructionism and at-risk learners.* Paper presented at the 2002 National Educational Computing Conference, San Antonio, Texas.

Stager, G. (2005). Towards a Pedagogy of Online Constructionist Learning. Paper presented at the *2005 World Conference on Computers in Education*. Stellenbosch, South Africa.

Stager, G. S. (2007). *An investigation of constructionism at the maine youth center.* [Doctoral Dissertation] The University of Melbourne, Melbourne.

Stager, G.S. (2007) *Towards the construction of a language for describing the learning potential of computing activities.* Paper presented at the EuroLogo 2007 Conference, Bratislava, Slovak
Taylor, R. (1980). *The computer in the school: Tutor, tool, tutee*. NY: Teacher's College Press.

Topal, C.W., Kong, E. and Gandini, L. (1999*) Beautiful Stuff!: Learning with Found Materials*. New York: Sterling Publishing Company.

Turkle, S., & Papert, S. (1991). Epistemological pluralism and the revaluation of the concrete. In I. Harel & S. Papert (Eds.), *Constructionism* (pp. 161-191). Norwood, NJ: Ablex Publishing Corporation.

---

[i] A collection of the open-ended robotics challenges may be found at http://www.stager.org/lego

[ii] The two LEGO engineering documents and MicroWorlds EX reference sheet are available for download at http://www.stager.org/lego

[iii] Articles about effective project-based learning may be found at http://stager.org/articles/goodpbl.pdf

Modelling spatial aspects of forest-savanna dynamics - An educational web-based tool

Richard Taylor, *richard.taylor@sei.se* and **Fábio Ferrentini Sampaio**, *ffs@nce.ufrj.br*
Stockholm Environment Institute (Oxford Office) and Federal University of Rio de Janeiro

Description

Spatial interaction rules that govern ecosystem function are relatively easy to hypothesise, and yet the consequences for global behaviour are not always so obvious. A small number of rules, by invoking a variety of slow and fast feedback mechanisms, will produce highly complex outcomes. This is a recurrent theme that is addressed in the study of 'Complex Adaptive Systems' (CAS).

Students that interact with Cellular Automata (CA) based models can learn about an ecosystem's behaviour and how it is affected by various management actions. Learning by 'composing' alternative behaviours and testing their consequences may generate a more sustained learning experience for the student that will allow them to develop new intuitions. Our model of forest-savanna dynamics, and the Modelling4All platform on which it is hosted, aims to provide a test of how much students can learn from this approach.

The model is a reimplementation of the one presented in Hochberg et al. (1994) showing the dynamics of forest and savanna areas in the presence of seasonal fire events. Different processes and parameterisations are analysed in terms of their influence upon the spread of trees in the CA environment. Though it was originally designed for research, we have reimplemented this model in NetLogo on the Modelling4All platform as a pedagogical tool. We plan to extend the model to consider the role of human actors, and to encourage its users to consider issues of sustainable resource management.

The teaching materials progress the student through a set of learning exercises with the model. At each stage a number of questions are asked: If we increment the number of total mature trees in the beginning does it make any difference (% increase/decrease) in the total number of mature trees after a certain number of years ? Why ?; Does the occurrence of fire in a certain year make any difference in the total number of mature trees after a certain number of years ?; In the case you have fire in a certain year, how many years will your model need to recover?

The Modelling4All platform on which the model is hosted, is itself based on NetLogo coding. The modelling environment runs in any web browser, offers an intuitive interface of buttons and check boxes for exploring various options, and most importantly for student collaboration allows sharing of models (via http links). The materials will be presented and tested through use with two classes in Brazil. Evaluation and feedback from the classes will be collected.

References

Hochberg, M. E.; Menaut, J. E.; Gignoux, J. The Influences of tree biology and fire in the spatial structure of the West African savannah. **Journal of Ecology** V. 82, pp 217-226 , 1994.

Keywords

environmental education, modelling in education, agent-based modelling**.**

Professional Networking by Disadvantaged Youth in Recife Brazil using Information and Communication Technology

Ann Berger Valente[1], *ann@dglnet.com.br*
Learning Technologies Network (LTNet – Brasil)

Abstract

The study describes how Information and Communication Technology are enabling disadvantaged young women from Recife, Brazil to establish and expand their professional networks. Funded by the Nike Foundation, *Programa Para o Futuro* – Young Women in Action (PPF-YWA) is a program designed by the Academy for Educational Development (AED) and the Academia para o Desenvolvimento da Educação – ADE Brasil, and implemented in partnership with Learning Technologies Network (LTNet-Brasil), Equipe Técnica de Assessoria, Pesquisa e Ação Social (ETAPAS) and the Faculdade Maurício de Nassau.[2] In PPF-YWA, a population of at-risk of young women between the ages of 15 and 24, who have limited access to mechanisms to break out of their insular and impoverished communities, participate in a comprehensive economic empowerment learning program which uses productivity tools to support a myriad of project based learning experiences. One of these activities includes building eMentoring relationships between pairs of youth and volunteer professionals using email and instant messaging as the primary means of communication.[3] The role of productivity tools in this setting is discussed from both the point of view of the eMentors as well as eMentees. Through the project's structured monitoring and evaluation effort, eMentors report that mentoring using ICT solves a series of logistical problems, allows them to help the young women with writing and communication skills, and provides them with a level of intimacy that feels comfortable to the majority of eMentors. Productivity tools provide a means for the young women, who began the program with precarious language and IT literacy, to build knowledge about the subtle cultural norms of professional communication that is essential for success in the workplace. Finally, culminating results on expanding professional networks show, for example, an increase in the percent of young women who report having a contact who knows them well professionally and who could submit a personal reference from a professional.

Figure 1: Comparison of young women's professional contacts before and after the program

Keywords

ICT, Mentoring, eMentoring, professional networks, disadvantaged youth, economic empowerment

---

[1] Special thanks to Mariana Françozo, research assistant at LTNet-Brasil, for her systematic and perceptive analysis of the Monitoring and Evaluation data.
[2] AED is an international non-profit headquartered in Washington, D.C., ADE Brasil is a Brazilian NGO located in Recife, LTNet-Brasil is an NGO located in Brasilia and ETAPAS is an NGO located in Recife.
[3] eMentoring was pioneered in Brazil by AED as part of the first *Programa Para o Futuro* pilot in 2003.

Introduction

Programa Para o Futuro – Young Women in Action, an economic empowerment project for disadvantaged youth in Recife, Brazil, uses ICT as a powerful means of enabling youth to expand their horizons and make inroads into socio-economically dominant cultures. Youth use these technologies to build new networks, to gain basic employability and marketing skills, to research and reflect upon questions of “Who am I”, “Where do I want to go?” “How can I get there?” In *Programa Para o Futuro* – Young Women in Action (PPF-YWA), ICT provides a backbone for a myriad of project based learning experiences. Technology is ubiquitous in these experiences as youth, their professional mentors, and their project facilitators plan, register, interact, create, and produce content and communicate. “When technology enables, empowers, and accelerates a profession's core transactions, the distinctions between computers and professional practice evaporate.” (Weston & Bain, 2010. p. 10). In PPF-YWA, the young women learn to use productivity tools – information sources, time management, organization, networking, and text, image and data processing. Instead of classes in how to use productivity tools, the young women simply use these tools as they learn the importance of organization, time commitments, follow through, personal and collective responsibility, searching for opportunities, speaking up and speaking out. The productivity tools become their means to participate in the dominant professional social structure.

Technology itself cannot shorten the distance between social classes in Brazil. Digital exclusion cannot be reduced to a question of access. Social inequality “…is not expressed merely in terms of access to material goods – radio, telephone, television, Internet -, but also in terms of the user’s intellectual and professional capacity to extract the maximum potential offered by each instrument of communication and information.” (Sorj, 2003, p.59). Similarly, Warschauer (2007) explains that even when comparing schools in New York City, the lack of basic traditional literacy severely compromises what youth can do with the digital communication. “Indeed, the divide allegedly attributed to unequal information literacy or multimedia literacy most frequently has its roots in differential access to basic reading and writing competency and cultural capital.” (ibid, p. 44).

The young women who comprise PPF-YWA live in the Northeast region of Brazil where the literacy rate is one of the lowest in the country. According to the Indicator of Functional Illiteracy developed by the IBOPE and the Instituto Paulo Montenegro, 47% of the population 15 years and older is functionally illiterate.[4] (Instituto Paulo Montenegro, 2007) These young women come from impoverished learning environments and are isolated from much of mainstream society. “Young people from the poorest families consistently have… formal sector employment rates that are one-eighth of the national average” (World Bank, 2007). Youth from these marginalized communities also lack the social and technical skills to transition into the workplace. Young women are at an even greater disadvantage due to discriminatory social, cultural, economic and political norms, where obstacles to productive activity only contribute to the cycle of poverty. Unemployed, disadvantaged young women need professional role models who can provide new points of reference, guidance, and ‘cultural capital’.

Programa Para o Futuro – Young Women in Action is a program designed by Dr. Eric Rusten and Alexandra Fallon of the Academy for Educational Development (AED) and Tania Ogasawara of the Academia para o Desenvolvimento da Educação – ADE Brasil, and implemented in partnership with Vera Suguri of Learning Technologies Network (LTNet-Brasil), Equipe Técnica de Assessoria, Pesquisa e Ação Social (ETAPAS) and the Faculdade Maurício de Nassau. PPF-YWA is funded by the Nike Foundation as part of its worldwide effort to promote the economic empowerment of adolescent girls through its campaign for the *Girl Effect*[5]

[4] Based on a classification of four levels of literacy – 16% illiterate, 31% rudimentary, 34% basic and 19% complete.
[5] http://www.girleffect.org/

where investing in girls, "…will unleash a powerful ripple effect" for them, their families and the community (Nike Foundation, 2008). PPF-YWA is a three year project designed to promote the economic empowerment of over 800 young women between the ages of 15 and 24 residing in the urban metropolis of Recife Brazil. The Program has objectives on four major fronts: Employability knowledge and skills, Gender awareness, life skills and voice, Girls friendly environment, and Professional networks. It is a complex, ambitions program which seeks to promote the social, educational, and economic development of disadvantaged young women through learning projects ranging from personal marketing, public speaking, mentoring partnerships with active professionals, and gender and reproductive health awareness, to name a few. It also includes a rigorous monitoring and evaluation (M&E) component for in-depth, objective documentation, analysis and reflection on all aspects of the program.

This paper focuses on the role of technology in the mentoring partnerships that the young women have with working professionals. This core component of PPF-YWA uses digital communication tools to convert conventional mentoring into "eMentoring", and to bring the young women into direct contact with professional role models [6]. This is critical because it provides the young women with a personal contact and it provides a meaningful purpose for using the productivity tools. Initial findings from the first nine month cycle of the program, including a four month eMentoring activity, involving 100 young women, and their respective eMentors, collected as part of the project's M&E component, are presented.

eMentoring and the eMentors' perspective

According to the group, MENTOR, an advocate of mentoring partnerships, "Mentoring is a structured and trusting relationship that brings young people together with caring individuals who offer guidance, support and encouragement aimed at developing the competence and character of the mentee." (Mentor, 2010). Within the context of PPF-YWA, mentors provide young women with the constructive example of a professional engaged in the job market and a window into possible careers. Given the marginalized nature of most disadvantaged communities in Brazil, these professionals represent a sphere of contacts normally unavailable to disadvantaged youth. eMentoring is a modality of mentoring where interaction takes place almost exclusively through synchronous and asynchronous communication using digital technologies, with face-to-face contacts being limited to once or twice during the mentoring experience. In PPF-YWA, professionals from fields including information technology, finance, administration, communications, education, human resources, social development, and law volunteered their time to participate as eMentors. The eMentors and their eMentees engage in regularly scheduled, one-on-one, synchronous interactions using instant messaging. The scheduled conversations are facilitated by, though not restricted to, weekly agendas of topics pertinent to the current moment in the program, for example, the job interview process, group dynamics, writing a resume, and career choices. For each of these topics the eMentor may provide specific information and guidance. However, eMentoring is also a space to personalize experiences, discuss issues of confidence, image and self perception. In addition, there is an ongoing asynchronous exchange of emails and a sharing of information and products by both eMentors and eMentees, for example, CVs, articles, video clips, and poetry according to the particular needs and interests of the mentoring pair. A byproduct of this interaction between engaged individuals from distinct social milieus, is that the young women build knowledge about and skills

[6] Dr. Rusten of AED designed the pioneering eMentoring program for the pilot project of *Programa Para o Futuro* in 2003 and Cida Neuenschwander implemented the pilot and developed many of the detailed elements of what comprises an eMentoring activity. Expanding on this success, ADE Brasil's eMentoring team, Cida Neuenschwander and Marisa Selva have cultivated an extensive network of volunteer eMentors. This invaluable resource depends upon these two full time staff members' efforts to engage in an extensive recruitment and selections process. They also provide initial and on-going training to eMentors, while coaching and guiding both eMentor and eMentees to build their capacity to engage in active, supportive and productive interactions.

with using ICT. In addition, because of the intense use of written communication, through instant messaging and emails, they are obliged to express their ideas in writing, either by formulating and responding to questions, making comments, expressing who they are. The written word gains new significance as the young women seek to build a professional relationship.

A series of formal Focus Group Discussions (FGDs) were conducted with a subgroup of randomly selected eMentors at the end of the 4 month activity as part of the project's M&E component. Focus Groups were divided solely on the parameter of experience, some FGDs were composed of eMentors who had participated in previous ADE projects and others were composed of first time eMentors. Gender and professional background were not considered in the group formation and each FGD contained a mix of men and women, and representatives from a variety of professions including IT, finance, education, social service, and law. FGD analysis protocol followed the methodology defined in the AED manual for systematic analysis of qualitative data (de Negri & Thomas, 2003). A broad range of topics were raised in the FGD protocol ranging from the initial training of the eMentors to the eMentors' expectations for the young women's futures. Responses related to the pros and cons of the use of electronic media in the eMentoring activity are presented here.

The "e" in "eMentoring" was almost unanimously considered a positive aspect of the mentoring experience. Electronic media bring a series of unique components into play. Foremost is the ability to participate in the program from the work setting. Everyone said that it would be difficult to be a mentor if they had to travel to a particular location given their tight schedules and lack of free time. *"Everyone here already wanted to do some kind of social benefit work but didn't have the time, so eMentoring is a way to make this a reality."* (new eMentor). Electronic media allows more people to be mentors.

While the IT professionals have greater facility with the technology, ICT is sufficiently disseminated among professionals in the metropolitan Recife area that none of the eMentors had trouble using instant messaging or email. Many things get solved using electronic media within the work place. It's not an artificial situation. On the other hand, for some eMentors, the computer is impersonal. *"Electronic communication is kind of cold, you don't see how the person is feeling, her expression."* (returning eMentor).

When discussing electronic communication, eMentors raised the issue of the young women's language and how the low level of language skills leaves much to be desired for the work place. Because most of the communication between the pairs used the written word, it was possible for the eMentor to evaluate the young woman's writing and to help her with her Portuguese.

Another attribute raised in each of the focus groups was that eMentors felt more comfortable, and less inhibited, in expressing themselves and raising certain issues. *"If I were face-to-face with her I would not feel comfortable saying certain things."* (returning eMentor). Likewise, eMentors felt that using electronic communication enabled the eMentees to open up and talk more readily about their day-to-day lives and about the problems they face.

Part of this spontaneity may come from a feeling that communication with the young women is safer through electronic tools. One returning male eMentor explained that electronic communication protects the person from getting too involved. *"…it protects a little; it creates a thin film that makes it so that you don't get pulled into the relationship."* (returning eMentor). A first time female eMentor also explained how electronic communication creates a necessary distance for her: *"The personal issues affect us, we don't have the training [to deal with them well]. If it were face-to-face I would have been destroyed, and she would have seen that."*

The following table summarizes these findings on the pros and cons of using electronic communication tools in the mentoring relationship. The prevalence of statements supportive of each reason is shown according to gender and profession, the greater the number of "X"s the more frequently eMentors expressed the particular idea. "XXX" means that almost all the members of the category expressed the particular idea. Fewer “X”s means that the idea was mentioned by fewer people, which is not to say that the others disagreed, but simply that they did not comment. Zero “X”s means that no one from that category mentioned the corresponding idea.[7]

Prevalence of statements for and against the use of ICT in mentoring communication according to gender and profession				
Statement: Pro	Gender		Profession	
	Male	Female	IT, Finance	Education Social Dev. Law
Facilitates in terms of time and space: people who are busy or who can't spend time moving around the city can still be eMentors.	XXX	XX	XX	XXX
Many things get solved using electronic communication within the work place.	-	X	-	X
The eMentors' familiarity with instant messaging facilitates the activity (it requires abilities that are already known and commonly used.)	X	-	X	-
Through electronic communication it is possible to evaluate the young woman's writing and to help her with her Portuguese.	X	XX	X	XXX
Using electronic communication mentors and mentees are more comfortable expressing themselves on various issues.	XXX	XXX	XXX	XXX
Communication through electronic communication tools is safer; it creates a necessary distance so that the person doesn't get too involved in the relationship.	XX	XXX	XX	XXX
Statement: Con				
Electronic media is “cold”. You can't see people's feelings.	X	X	X	X

It is clear from this table that the eMentors were overwhelmingly positive about the use of electronic communication tools in the mentoring activity and that all eMentors felt that these tools allowed them to express themselves more freely. Not surprisingly, educators more readily mentioned the opportunity to correct Portuguese language than technical professionals. Statements on the “necessary distance” created by electronic media were voiced by all categories, but even more frequently by women and by professionals from the areas of education, social development and the law. While the majority of eMentors preferred this distance, for others it was a barrier that interfered with establishing the level of intimacy they desired. It is important to note that this group of volunteers is a purposeful sample in the sense that these individuals actively chose to participate in PPF-YWA as Mentors knowing that they would be using ICT.

[7] Because of unequal numbers of men and women, frequencies were calculated based on the percentage of responses in each category.

Changes in Young Women's Professional Networking Capacity using ICT

When the young women entered PPF-YWA they demonstrated the level of IT skills that young people acquire "spontaneously" with friends and in cyber cafes when they have limited access and nothing very productive to do. They view each others' social networking pages and look up pop stars on the Internet. Some didn't know how to use the mouse or scroll the screen. Their extensive use of unconventional acronyms and abbreviations such as the Portuguese equivalent of using the letter "u" in place of the word "you", masked a plethora of Portuguese language errors. Although 71% reported having email accounts when they entered the program, it became apparent that they lacked many basic ICT skills, for instance, how to attach a file, unattach a file, copy people on emails, make IM accounts, save chats, save their work so as to facilitate retrieval, etc. The list was quite extensive.

Nevertheless, the skills to use productivity tools were readily acquired as the young women tried to learn the more challenging and subtle cultural norms of professional communication through the use of these electronic tools. For example, they had to learn to distinguish when it is appropriate or inappropriate to use cutesy emoticons. They had to learn to use inoffensive language. They had to learn that during an important synchronous communication with a busy professional contact, not to have multiple IM windows popping up with little jingles. They had to learn to respond to an email from their eMentor within a reasonable time frame, and if they didn't understand what was being requested, they had to learn to ask their eMentor to explain instead of remaining silent.[8] They had to learn to be forthcoming, active participant in the conversation and not "monosyllabic" passive participants. Developing good communication skills was very difficult for many of the young women because of their relatively low level of Portuguese literacy. It was a challenge to organize their ideas and formulate them in writing. Miscommunications with their eMentor serve to raise awareness as to the relevance of conventional spelling and sentence construction. Even though the young women were exercising their writing skills in purposeful communication, many eMentors exclaimed that their eMentees' language skills will need greater attention. Nevertheless, the young women were making important inroads into professional communication through the acquisition of ICT literacy.

Results from a formal assessment of the young women's reflections on what they learned through eMentoring are not yet available. However, initial data shows some important changes. This data was collected using an extensive questionnaire designed to gather information on Knowledge, Attitudes and Perceptions (KAP). The questionnaire was applied online prior to Program start up and again nine months later at the end of the first round. Here are responses to some of the questions related to eMentoring and Professional Networking.

Essential to career development is having contacts in important places who know you well and in a professional context. Self responses show a slight increase in the percent of young women who responded affirmatively to the question "Is there a person with knowledge of the existing job market who knows you well professionally?" Since most of the young women had just finished the Program and had not yet found quality employment, few had made contacts who know them well professionally, though there was a slight increase from 43% to 57%. Greater increase in this indicator is expected at one year follow-up.

[8] This is an especially important set of skills since being able to ask adults and "important people" questions is new to youth and can be seen as exposing ignorance. However, having this ability is critical to success in the workplace.

Is there a person with knowledge of the existing job market who knows you well professionally?		
Response	**% at Baseline**	**% at Endline**
Yes	43.0	57.4
No	54.8	41.2
N/A	2.2	1.5

Another essential component of employability is the personal reference. Through eMentoring and PPF-YWA there was a significant increase in the percent of girls who responded that they could submit a reference from a professional contact who could attest for their capabilities – an increase from 29% to 58%.

Would you be able to submit a personal reference from a professional who can attest for your capabilities?		
Response	**% at Baseline**	**% at Endline**
Yes	29.0	58.8
No	60.2	35.3
N/A	10.8	5.9

The baseline / endline comparison of the percent of girls who have these two kinds of professional contacts is represented below:

Figure 1. Comparison of young women's professional contacts at baseline and endline

Young women were also asked the following question. "Think about a situation in which you made an important decision for your life and the people you turned to for advice. How do you know these people? In each category write the number of people who have helped you make an important life decision." Presented below is the percent of girls who report having one or more people who have helped them to make an important life decision, distributed by type of contact. The results show that at baseline young women reported that family (1st), friends (2nd) then religious organizations (3rd) are the most common sources of advice. Professional contacts are one of the least common sources of advice. At endline, while the previous contacts remain stable, professional contacts increased dramatically, as well as teachers, and both of these surpassed religious organizations. The increase in teachers may in part be due to a stronger personal involvement in their school setting but also may include the relationships the young women established with the PPF-YWA's learning facilitators who the young women may refer to as "teachers".

Percent of girls who report having 1 or more people who have helped them to make important life decisions		
Category	**% at Baseline**	**% at Endline**
Family members	90.2	89.7
Friends	69.9	75.0
Teacher	32.0	55.9
Professional contacts	11.7	47.1
Church member or religious organization	36.9	36.8
Neighbors	29.1	32.4
Members of community centers	9.7	7.4
Other	3.9	0

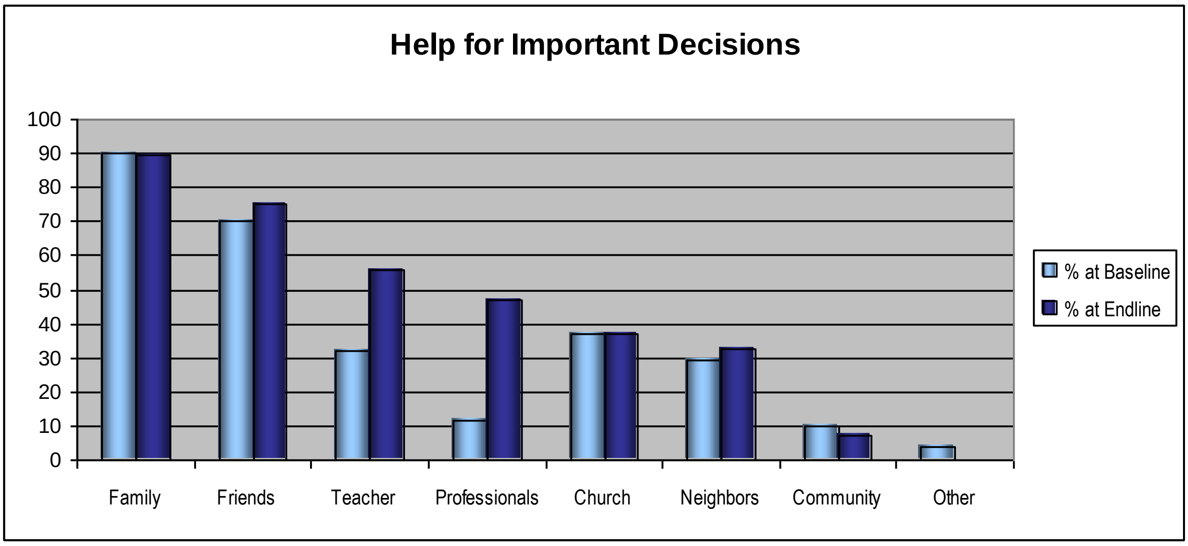


Figure 2: People who have helped to make important life decisions

Taken together we see clear evidence of expanding professional networks. It is through these precious new relationships that young woman can begin to break the downward spiral of isolation and lack of opportunities so common to youth in disadvantages communities.

Concluding Reflections

Results from the first of four rounds of the PPF-YWA program, show that as the young women learned to express themselves using ICT within the eMentoring relations they were able to expand their professional networks, an essential element of economic empowerment. The process of learning to use productivity tools makes explicit innumerous subtle conventions in professional communication, allowing the young women to construct inroads into the culture of the workplace, increasing their 'cultural capital'. Through interactions with their eMentors the young women were able to start to build new kinds of relationships previously unavailable to them. The professionals who are eMentors in PPF-YWA use ICT in their normal routine of communication and solving problems, so mentoring using electronic tools came as second nature. For some it even provided a safe filter through which the mentoring relationship could be built. However, like the young women, ICT allowed the eMentors to come in contact with a reality that was previously unavailable to them. Hopefully this exchange between new contacts will continue to bring significant learnings on both sides.

References

de Negri, B. and Thomas, E. (2003). *Making Sense of Focus Group Findings: A Systematic Participatory Analysis Approach.* Washington, DC: Academy for Educational Development.

Instituto Paulo Montenegro (2007) *Inaf 2007 mostra a evolução da educação no Brasil* Retrieved on February 1, 2010 from http://www.ipm.org.br/ipmb_pagina.php?mpg=4.02.02.00.00&ver=por

Mentor (2010) Mentor / National Mentoring Partnership. Retrieved on February 1, 2010 from http://www.mentoring.org/mentors/about_mentoring/

Nike Foundation (2008) *Investing in the Girl Effect: the most powerful force for change.* Retrieved on February 1, 2010 from http://www.nikefoundation.org

Rusten, E, Ogasawara, T., and Brady, K. (2005) *Programa Para o Futuro: Enabling Disadvantaged Youth to Build New Futures*. Washington, DC: Academy for Educational Development.

Sorj, B. (2003) *brasil@povo.com: A luta contra a desigualdade na Sociedade da Informação*. Jorge Zahar Editor, UNESCO, Rio de Janeiro.

Warschauer, M. (2007) *The Paradoxical Future of Digital Learning*. Learning Inq, 1:41-49. Retrieved on Febuary 1 2010 from http://www.gse.uci.edu/person/warschauer_m/docs/paradox.pdf

Weston, M.E. and Bain, A. (2010) *The End of Techno-Critique: The Naked Truth about 1:1 Laptop Initiatives and Educational Change*. Journal of Technology, Learning and Assessment, 9(6). Retrieved on February 1, 2010 from http://www.jtla.org.

World Bank (2007) *Brazil: Youth at Risk in Brazil, vol. 1; Policy Briefing.* World Bank, Washington, DC. Retrieved on February 1, 2010 from http://lnweb90.worldbank.org/ext/epic.nsf/ImportDocs/C9395B68583A026E4825733100109802?opendocument&query=PH

ICT Based Learning Community: empowering socio-economically disadvantaged people

José Armando Valente[i], *jvalente@unicamp.br*
Department of Multimeios and Nied, Unicamp & Ced, PucSP

Abstract

The objective of this article is to present the concept of an ICT based learning community, to describe two computational tools, audio and video, that were integrated into the ICT, and to show how these communities can be used with subjects who are considered digitally and socio-economically excluded, such as the elderly, adolescents, community health agents, and to implement the concept of a learning city. One of these examples is described in more detail.

This work is part of a research project that is taking place at the Lipacs (Laboratório Interdisciplinar de Pesquisa-Ação para Comunidades Saudáveis – Interdisciplinary Action-Research Laboratory for Healthy Communities), sponsored by Fapesp, set up at the Media Department, Art Institute, at the State University of Campinas (UNICAMP), Brazil.

The creation of an ICT based learning community is supported by several concepts such Wenger's concept of communities of practice, Freire´s educational principles and Papert's constructionism ideas. When people construct a product using ICT, and understand how they have done it, they can experience the feeling of empowerment - the sensation of being able to produce something that has been considered impossible.

The target population working with Lipacs is composed of people with a low level of schooling and difficulties with reading and writing. In order to help these people to be engaged in activities developed in an ICT based learning community we implemented audio and video facilities in the learning management system (LMS) used by the project (Tidia-Ae). Thus instead of communicating through written text, community members can use audio and video to express their ideas. The ICT based learning community was implemented as part of 4 projects:

- Literacies with the elderly – aiming to investigate how elderly people who are considered excluded from the digital world use ICT for the construction of new literacies;
- Appropriation of hypermedia by community health agents – how hypermedia resources are used by community health agents, doctors, community leaders, young people and teachers to interact, to access information, to communicate and to continue their education;
- Implementing a learning city – creating situations that can foster personal, social, professional, cultural and economic development for citizens, so it can become a learning city;
- Preparing adolescents for the world of work – the goal is to build a sense of empowerment based on mathetic and aesthetic experiences provided by the development of multimedia projects. This particular example is described in depth, showing how these adolescents are using the ICT to promote their digital inclusion and their inclusion into the world of work.

In all these examples it was possible to observe that different community members are incorporating ICT into their practice and doing things that are meaningful to them, as well as being empowered by this experience.

Keywords

ICT, information and communication technology, learning community, digital exclusion, technology in education, learning process, empowerment

Introduction

The objective of this article is to present a theoretical framework for a series of projects related to the creation of an information and communication technologies (ICT) based learning community; to briefly present the development of two computational tools that are helping the creation and support of ICT based learning communities; and to show how these communities can be used with subjects who are considered digitally excluded and socio-economically disadvantaged, such as the elderly, adolescents, community health agents, and to be used to implement the concept of a learning city. The example with adolescents is described in depth.

This work is part of a research project that is taking place at the Lipacs (Laboratório Interdisciplinar de Pesquisa-Ação para Comunidades Saudáveis – Interdisciplinary Action-Research Laboratory for Healthy Communities), sponsored by Fapesp[ii], set up at the Media Department, Art Institute, at the State University of Campinas (UNICAMP), Brazil. The main goal of this research is to develop computational tools that facilitate the creation of ICT learning communities so people who are considered digitally excluded and functionally illiterate can have access to these technologies, use them to solve problems, to learn and to communicate. Four graduate students are developing their studies in this area respectively working with elderly people, community health agents, adolescents at risk, and one student is working with the concept of learning cities, setting up learning activities involving residents from a small town[iii].

The creation of an ICT based learning community is supported by several concepts such Wenger's concept of communities of practice (Wenger, 1998), Freire´s educational principles and Papert's constructionism ideas. After presenting the theoretical framework that supports the creation of an ICT based learning community, two computational tools will be described, that were developed to help functionally illiterate people to use ICT facilities, followed by four examples to illustrate how these communities can be implemented involving different populations, in different settings and with different educational objectives. These examples show how particular populations can construct knowledge by participating in an ICT based learning community and how this knowledge can empower each community participant. The goal of the work is not only to create conditions for people from these different populations to have access to ICT but, in the process of using these technologies, to experience the feeling of having powerful ideas, as proposed by Papert (1980), to be conscious of the knowledge constructed, and to be able to use this knowledge to act and help to transform the environment where they live, as proposed by Freire (1975).

In these ICT based learning communities the knowledge construction process is based upon Papert's constructionist ideas, since the subjects are using ICT to produce concrete object (Papert, 1986; 1992). The availability of computationally-rich construction materials can afford these people the opportunity to experience the empowerment associated with the feeling of learning how to use ICT and having wonderful ideas (Stager, 2003; Valente, 1999). Considering that the subjects participating in these studies have very little experience with ICT and limited ability to read and write, the empowerment feeling is even deeper when they are able to produce something that was considered impossible. They are proud of their work, show the product to everybody and are very happy to exchange ideas about what they have done. Thus the ICT have a very important role in the creation of these learning communities.

The foundations of an ICT based learning community

The concept of an ICT based learning community was developed using different concepts that were proposed independently. The first contribution was the concept of community of practice. This term was first used in 1991 by Jean Lave and Etienne Wenger in their work related to situated learning when they first proposed the idea that learning could be a process of participation in communities of practice. They noticed that learning did not have to be strictly related to school, but could be social, and developed from our experience of participating in daily life (Lave & Wenger, 1991). The emphasis was in the social practice, the "doing in a historical and social context that gives structure and meaning to what we do" (Wenger, 1997).

As mentioned by Wenger, communities of practice can take different forms and they can vary along a number of dimensions, although they must preserve a few essential characteristics such as: the domain, a community of practice is focused on a domain of shared interest; the community, which is formed by the fact that people are pursuing their interest in their domain, creating the opportunity to engage in joint activities and discussions, to help each other, and to share information; and the practice, since members of a community of practice are practitioners. They develop a shared repertoire of resources such as experiences, stories, tools, and ways of addressing recurring problems— in short, a shared practice. (Wenger, 2001)

The concept of community of practice has varied over the years and is now used to define "groups of people who share a concern, a set of problems, or a passion about a topic, and who deepen their knowledge and expertise in this area by interacting on an ongoing basis" (Wenger, McDermott & Snyder, 2002, p. 4). However, as pointed out by other authors, a community of practice is much more complex than a network of informal relationships or communities of interest where members interact and exchange information. In a community of practice participants work collaboratively to enhance their practice and to do so they exchange information, reflect on what is proposed by peers, build knowledge, apply this knowledge to improve their skills and consequently improve the activities they carry out as part of their practices (Saint-Onge & Wallace, 2003).

Another important concept was "virtual community" proposed by Howard Rheingold. He defined virtual communities as "social aggregations that emerge from the Net when enough people carry on those public discussions long enough, with sufficient human feeling, to form webs of personal relationships in cyberspace (Rheingold, 1993). These are people who are actives, who share values, interests and goals, and who assume an attitude of mutual support, through interactions in the cyberspace. Based on this idea, other authors understood the importance of virtual communities, such as Levy (1999), who proposed the creation of virtual communities as one of the three principles that guided the initial growth of cyberspace.

The dissemination of ICT was instrumental in setting up virtual communities related to the learning process. ICT offer communication facilities, allowing the participants to interact with community experts and even among the participants, creating the means for these learners to share ideas, reflect on different points of view and form communities that are working to support the process of learning. This was a very common idea among researchers working with distance education mediated by ICT who have created the concept of virtual learning communities. The fact that students are working with ICT, they should not be passive, absorbing the information received from teachers, but teachers and students should form a learning network, as proposed by Harasim and colleagues (Harasim, et al, 1995). Palloff and Pratt (1999) played an important role in developing the concept of virtual learning communities by proposing that online courses should be viewed as learning communities in cyberspace.

In our work we noticed that the constitution of learning communities does not necessarily happen only in the virtual space. Learning communities can also be established when people are interacting, helping each other and practicing whether they are working in a face-to-face situation or interacting via ICT. What is important is the fact that these people are using ICT to develop products of their interest and are interacting among themselves. Thus we have adopted the concept of an **ICT based learning community**.

Another important ingredient in the ICT based learning community is the fact that people are using ICT to develop a product of their interest. This is taken from Freire's ideas that the more the learning process is related to the interest and the situation in which the learner lives, the better her/his chance of understanding the content and, thus, of becoming involved in the educational activities. Also based upon another of Freire's educational principles, members of these communities assume a very altruistic attitude, using knowledge they have constructed to help other members of the community as well as to transform the reality they belong to (Freire, 1975).

However, as mentioned by Wenger, the fact that there is a group of people interested in learning and they are interacting through the use of ICT in a particular learning context, does not necessarily mean the constitution of an ICT based learning community. Another important aspect to be considered is the type of learning that occurs in these communities. For example, how much they contribute to the exchange of information and how they provide conditions for the construction of knowledge.

Memorization of information and knowledge construction are part of the learning process. An education based entirely on memorization is not consistent with the proposal for a community of practice, as explained above. In order to change practices, besides information, it is necessary to have competence, defined as concepts, skills and attitudes (OECD, 2005), which are impossible to be memorized.

As observed by Piaget (1976) children are able to construct knowledge about certain concepts through spontaneous interaction with objects and people. However, the development of more abstract concepts, for example, sophisticated logical-mathematical concepts, depends on the help of more experienced people, educators as proposed by Piaget (1988). A similar distinction was made by Vygotsky. He distinguished between spontaneous and scientific concepts, the former being developed from the individual's experience with the world. The scientific concepts are developed from the spontaneous, but depend on the social interaction, especially the school (Vygotsky, 1986).

Therefore, it is illusory to think that in an ICT based learning community the process of knowledge construction happens spontaneously. This construction depends on the interaction of learners and the guidance of experts, community mediators, who know how to keep the community in action. This means helping to define themes that are compatible with the learners' interests and the expert's pedagogical intention, to adjust the difficulty of the discussion, and the problem being solved, to a level that is consistent with the zone of proximal development (ZPD) of each learner or of the community collectively. The experience with communities of practice shows that the active construction and the success of a community depend on one person or core group who takes responsibility in order for the community to develop (Wenger, 1998). This person or this group assumes the role of promoting, enabling and helping to create conditions for the knowledge construction process.

Developing ICT facilities to be used in learning communities

The target population working with Lipacs is formed by socio-economically disadvantaged people, in general people with a low level of schooling and having difficulty reading and writing. In order to help these people to take advantage of a system to continue their learning, we implemented audio and video facilities in the learning management system (LMS) used by the project (Tidia-Ae). The use of voice and video facilitate the interaction of community members. Also the integration of these new ways of communicating into their practices, and the fact that these people are in constant contact with written words, is opening up new possibilities for them to learn to read and write and improve their living conditions.

Video and audio communication systems were integrated to the LMS so users could "talk" to the system and "listen" instead of writing and reading. Also the fact that these communication facilities are integrated to the LMS the users do not need to leave the LMS and use software external to the system for real time conferencing, such as Skype, MSN, and GoogleTalk or other software for creating multimedia files such as Audacity, Windows Movie Maker. The integration of these software to the LMS avoids people going out of the learning environment to develop learning activities that would not be registered in the system. Also it avoids the trouble of installing these software in everyone's computers and getting people to use them. Figure 1 shows the integration of the voice system to the LMS.

Figure 1 – audio communication system integrated to the LMS used in the project

With these audio and video facilities instead of sending a written e-mail, for example, the user can record the message directly in the browser and send the audio file. The person receiving this message can open it and listen to the message. It is not necessary for the user to have any other software for manipulating audio files. The same functions were implemented in the forum and chat tools. A similar solution was adopted for the video facility.

These facilities are allowing several learners who were totally excluded from the digital world to be able to get familiar with these ICT and to expand their communication capabilities. This is also true for users who have no problem reading and writing but are acquiring new ways of expressing through different media such as audio, video, and image.

The role of the ICT in the construction of knowledge

ICT are used by the community participants either as tools to develop their products in a face-to-face situation or to communicate with other community members or with people outside the community.

When ICT are used to solve problems, the learner has to apply the knowledge s/he has in order to instruct the ICT about how to solve a particular task, making explicit what s/he knows. In this sense, ICT activities become a window into the learner's thinking and knowledge. Also the interactions user-ICT can be seen in terms of a sequence of actions – description-execution-reflection-debugging-description. With the help of other members in the community or an expert, in general, the community mediator, the products can be improved, as well as the learners' ideas and practice. Thus, each of these actions creates opportunities to increment knowledge, contributing to its growth in a crescent spiral – the learning spiral – that takes place as the user interacts with the computer to solve a particular project (Valente, 2003).

The idea of the learning spiral can also be established in an ICT based learning community when the participants, learners and experts (or the community mediator), communicate online – I have called this approach "virtually being together" since the community participants can be together, side by side, although via Internet. It is highly interactive and the interactions are established in order for the expert or community mediator to help the learner to solve particular problems s/he encounters in her/his practice. These interactions allow the mediator to continuously monitor and advise the learner so as to understand her/his interest and level of knowledge about a particular topic. In this way the mediator is able to propose challenges and to help the leaner to attribute meaning to what s/he is doing. In this situation the learner can process the information, apply it, transform it, seek new information and thus build new knowledge.

Initially all interaction is happening between the mediator and the learners. As the learners start to develop their product and interact with other community members it is possible to identify learners who know particular subjects that even the mediator does not dominate. These learners can help other colleagues and in these circumstances the ICT based learning community is formed, each working with their potentials and cooperating with each other.

In this kind of ICT based learning community each learner is engaged in a project or in a problem to be solved. In this situation s/he is producing results and reflecting upon them. If a difficulty emerges, the learner sends information (problem description, video, pictures) to the mediator. The mediator reflects upon the information received and sends back questions, articles, and examples of activities or specific support material so the learner can use this information to debug her/his project. New results can be obtained, new difficulties can emerge and the sequence of actions repeats, as shown in figure 2.

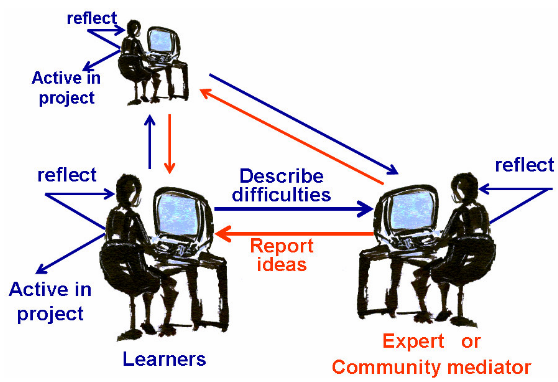


Figure 2: Interactive cycle learner-expert in the "virtually being together" approach

As illustrated in figure 2, the interactions are not taking place only between one learner and the mediator but all participants can interact among themselves and with the mediator. This helps to constitute a learner's network, the ICT based learning community. Everybody can see and comment everybody's work, exchange experiences, presenting different points of view for deeper levels of reflection. This learning community encourages collaborative work since learners can identify and jointly develop projects that have common backgrounds. At the same time it creates the conditions for learners to construct knowledge, fulfilling each participant's needs and interest. Each participant can use her/his working reality as a context to know more and to improve her/his practices. Thus, the learning spiral is taking place at different levels and with respect to different types of knowledge being constructed (Prado & Valente, 2002).

ICT based learning communities in action

This section briefly presents four examples to illustrate how the concept of an ICT based community can be used with elderly people, community health agents, adolescents and for the implementation of a learning city

Literacies with the elderly

This is a long term study, involving a group of 16 adults aged between 60 to 78 years, 14 women and 02 men, with educational background equivalent to 3rd to 6th grade level and digitally excluded. The first important aspect of this work is the fact that the elderly is considered the age group with the greatest number of digitally excluded in Brazil (IBGE, 2008).

The second important aspect is related to the use of ICT for the construction of new literacies. The elderly belong to the “pencil and paper” culture. The question is “how do these people use ICT to expand their ways of expressing themselves by using different communication channels such as oral, written, kinaesthetic, visual and digital, according to the design of media literacies proposed by Buckingham (2008a; 2008b)?”

The results have shown that these elderly were able to develop the ability to search for information, enhance communication, increase their social network, participate in leisure activities, help other colleagues to learn, promote products made by themselves, generate content, use ICT to help domestic economy, preserve the memory, contribute to increased self-esteem and insert themselves into the digital culture with a more critical and creative attitude.

Appropriation of hypermedia resources by community health agents

This work takes place in public spaces that provide access to ICT in the municipality of Pedreira-SP which receive people with different backgrounds and levels of schooling. These spaces are used to train community health agents as well as to work with the population to develop activities using different media in order to promote digital inclusion. The products, related to the participants’ interest and working context, such as health, for example, are presented, discussed and reworked by the group and then distributed and shared with the community involved. From these activities it is possible to notice how the different tools, audio and video, are allowing various forms of expression and communication. ICT are helping "to give voice" to each participant and in this context allowing to emerge the richness of their experiences so they can be reported, discussed and made available on the network.

Implementing a learning city

This project is providing personal, social, professional, cultural and economic development for the citizens of Sud Mennucci, first digital city in Brazil, with about 8 (eight) thousand inhabitants and with Human Development Index (HDI) equal to 0.779. The goal is for this city to become a learning city (Yarnit, 2004). The idea is to stimulate learning in various fields and promote participation and collaboration among people, and through these activities to identify skilful and talented individuals. Also a survey was conducted with the local community to identify people who want to teach and who want to learn. Based on the results of this survey, several events were organized which engaged people in learning activities on various subjects such as sports, music, cuisine, crafts, computers, reading, environment and others. These activities occurred on weekends in public spaces and they counted on their own community of teachers and learners.

The interesting aspect in this project is how communities have been established. First, the organization of these events involved the City Hall and its Departments, local business, local companies, external companies that provide services to the city and the population. Several planning meetings were carried out with various committees through video-conferences, practice that did not exist before, creating a social network so that there was constant communication and exchange of information among these committees’ participants.

Second, during the events, the participation and collaboration among people of different ages, skill levels and backgrounds provided an environment for identification of talents, common interests and allowed the organization of common interest groups that are looking for establishing new business. These groups migrated into ICT based learning communities to carry out the activities that started during the face-to-face events. Some results can be observed in the city website (Sud, 2010)

Preparing adolescents for the world of work

How to prepare digitally excluded adolescents for the world of work? One way is to develop operational competences, mechanical and repetitive, that allow the adolescents to merely get placed in a job. The other way is to value people in their differences and give them the opportunity to become autonomous and creative and, thereafter, become socially included with responsibility and political and social fairness. Valuing the youth as a whole person must include actions that facilitate the realization of their creative potential and, therefore, in developing a sense of empowerment.

This project is carry out with adolescents from 14 to 18 years old, digitally and socially excluded, as part of the Project Acreditar, developed in the city of Atibaia, State of Sao Paulo. The goal is to help these adolescents to build a sense of empowerment based on mathetic experiences (Papert, 1980; 1992) and on aesthetic experiences (Pareyson, 1993), and to promote the inclusion of adolescents into the world of work as they develop multimedia projects on issues related to work.

In this project is important to notice the role of the community mediator, emphasizing certain values which traditionally are not part of job training process. Second, to create conditions for the adolescents to reflect on what they do as part of daily activities to recognize their immense creative and operational capacities For example, groups of adolescents went around the region where they live and document living conditions as well as type of houses and what were the features that called their attention. Two pictures are shown in figure 3a and 3b. These pictures are discussed in the blog of this group (Feras, 2010) pointing out how one construction is creative and the other is poor. Similar activities were developed by other groups, looking for creative graffiti, documenting flooding in the region, and types of playgrounds.

Figure 3a – house that was considered very poor

Figure 3b – house that was considered very creative, constructed with material encountered in the street

These adolescents developed activities in an ICT based learning community using different types of digital media such as mobile phones, low cost digital cameras, and mp3. Each group created a blog to disseminate and receive comments about the work done. Also their production was stored in their respective portfolio in a LMS in order to help the processes of reflection and grasping of awareness about the action to undertake training, and simultaneously develop competences required for multiple literacies in accordance with the guidelines of OECD (2005).

Conclusion

Several authors working with the digital inclusion agree that it is not enough to just provide socio-economically disadvantaged people with access to technology. Besides this access it is necessary to create opportunities for them to incorporate ICT into the activities they develop (Sorj, 2003; Silveira, 2003). The approach we have take in our research is to adequate this technology with appropriate facilities, such as audio and video, and create an ICT based learning community so its members can be active practitioners, producing content for them as well helping their colleagues and use their knowledge to transform their reality.

The concept of ICT based learning community is founded on ideas borrowed from Wenger's work related to the community of practice, on Papert's constructionism ideas and on some of Freire's educational principles. Our goal is to provide community members a chance to learn about ICT, how to use ICT in activities related to their work or personal interest, to understand about the content of what they are doing, and to experience the feeling of empowerment. The examples discussed show that different community members are incorporating ICT into their practice, and doing things that are meaningful to them.

In fact, the ICT are creating circumstances for people to express themselves as a whole, expanding their capability to use different media that goes beyond written words, and through these media to be able to overcome certain difficulties such reading and writing. The resources to explore and to form networks of people interacting face-to-face or via internet have facilitated the exploration of these human dimensions, forcing us to continuously rethink our role as learner, the role of technologies in this process, and our conceptions about learning, especially when done with the help of the ICT.

References

Buckingham, D. (2008a). Defining digital literacy. What do young people need to know about digital media? In: C. Lankshear & M. Knobel (orgs.). *Digital literacy: concepts, policies and practices*. New York, USA: Peter Lang.

Buckingham, D. (2008b). *Media education: literacy, learning and contemporary culture*. Cambridge, UK: Polity.

Harasim, L. *et al*. (1995). *Learning networks: a field guide to teaching and learning online*. Cambridge: MIT Press.

IBGE. (2008). Acesso à internet e posse de telefone móvel celular para uso pessoal. Brasilia, DF: IBGE. Available at: www.ibge.gov.br/home/default.php.

Feras (2010). A vida como ela é. Available at: http://ferasprojetoacreditar.blogspot.com.

Freire, P. (1975). *Pedagogy of the Oppressed*. New York: The Seabury Press.

Lave, J. & Wenger E. (1991). *Situated Learning: Legitimate Peripheral Participation*. Cambridge: Cambridge University Press.

Levy, P. (1999). *Cibercultura*. Rio de Janeiro: Editora 34.

OECD (2005). T*he definition and selection of Key Competences*. Executive summary. Available at: www.oecd.org/dataoecd/47/61/35070367.pdf.

Palloff, R. M. & Pratt, K. (1999). *Building learning communities in cyberspace: effective strategies for the online classroom*. San Francisco, EUA: Josey-Bass.

Papert, S. (1980). *Mindstorms: Children, Computers and Powerful Ideas*. New York: Basic Books.

Papert, S. (1986). *Constructionism: A new opportunity for elementary science education.* A proposal to the National Science Foundation, Massachusetts Institute of Technology, Media Laboratory, Epistemology and Learning Group, Cambridge, Massachusetts.

Papert, S. (1992). *The Children's Machine: rethinking school in the age of the computer*. New York: Basic Books.

Pareyson, L. (1993). *Estética: teoria da formatividade*. Petrópolis, Rio de Janeiro: Vozes.

Piaget, J. (1976). *A Equilibração das Estruturas Cognitivas*. Rio de Janeiro: Zahar Editores.

Piaget, J. (1998). *Sobre Pedagogia*. São Paulo: Casa do Psicólogo.

Prado, M.E.B.B. & Valente, J.A. (2002). A Educação a Distância Possibilitando a Formação do Professor com Base no Ciclo da Prática Pedagógica. In: M. C. Moraes (Org) *Educação a Distância: fundamentos e práticas*. Campinas, SP: NIED-UNICAMP, p. 27-50. Available at: www.nied.unicamp.br/oea.

Rheingold, H. (1993). *The Virtual Community: Homesteading at the Electronic Frontier*. Available at: www.rheingold.com/vc/book/.

Saint-Onge, H & Wallace, D. (2003). *Leveraging Communities Of Practice For Strategic Advantage*. Oxford: Butterworth-Heinemann.

Silveira, S. A. (2003). Inclusão digital, software livre e globalização contra-hegemônica. In: S.A. Silveira & J. Cassino (Orgs). *Software livre e inclusão digital.* São Paulo: Conrad.

Sorj, B. (2003). *Brasil@povo.com: A luta contra a desigualdade na Sociedade da Informação*. Rio de Janeiro: Jorge Zahar Ed; Brasília, DF: Unesco.

Stager, G. (2003). The Feeling of Wonderful Ideas – Implications for the Future of Teaching and Learning**.** IFIP Working Groups 3.1 and 3.3 Working Conference: ICT and the Teacher of the Future. The University of Melbourne, Austrália. Available at: //crpit.com/confpapers/CRPITV23Stager.pdf

Sud (2010). Projeto Cidade Aprendente. Available at: www.sudmennucci.sp.gov.br/index.php/cidadeaprendente.html

Valente, J.A. (1999). Analysis of the Different Types of Software Used in Education. In J. A. Valente (Org) *Computers in the Knowledge Society*. Campinas, SP: UNICAMP/NIED. Available at: www.nied.unicamp.br/oea/pub/livro1/livro_ingl.zip, chapter 4.

Valente, J.A. (2003). How Logo Has Contributed to the Understanding of the Role of Informatics in Education and its Relation to the Learning Process *Informatics in Education*, Institute of Mathematics and Informatics, Vilnus, Vol 2, Number 1, p. 123-138.

Vygotsky, L.S. (1986). *Thought and Language*. Cambridge: The MIT Press.

Wenger, E. (1997). Communities of practice: Learning, meaning and identity. Available from: http://www.co-i-l.com/coil/knowledge-garden/cop/lmi.shtml.

Wenger, E. (1998). Communities of practice: Learning as a Social System. Available from: www.ewenger.com/pub/pub_systems_thinker_wrd.doc.

Wenger, E. (2001). Supporting communities of practice: a survey of community-oriented technologies. Available at http://www.ewenger.com/tech/

Wenger, E., Mcdermott, R. & Snyder, W. (2002). *Cultivating Communities of* Practice**:** *A Guide to Managing Knowledge*. Boston: Harvard Business School Press.

Yarnit, M. (2004). *Building local initiatives for learning, skills and employment*. National Institute of Adult Continuing Education. England and Wale. Available at: www.martinyarnitassociates.co.uk/downloads/Testbed%20Learning%20V5%20Final.pdf.

---

[i] This work is sponsored by Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq, grant 306416/2007, to which I am very thankful.

[ii] Laboratório Interdisciplinar de Pesquisa-Ação para Comunidades Saudáveis is part of the TidiaAe project, sponsored by Fundação de Apoio à Pesquisa do Estado de São Paulo (Fapesp), grant 2005/60628-7.

[iii] I am particularly grateful to Carla Lopes Rodriguez, Ivan Ferrer Maia, Flávia Amaral Rezende and Raquel Zarattini Chebabi for carrying out this work.

Application of a Creative Approach by Building Spatial Mechanical Models in a Microworld of Dynamic Geometry

Jiří Vaníček, *vanicek@pf.jcu.cz*
Dept of Informatics, Faculty of Education, University of South Bohemia in České Budějovice

Abstract

Constructionism principles of education employing computers are often associated with programming. However, there are many environments based on the similar approach applicable in various school subjects. In the field of mathematics, there is a set of dynamic geometry software available. It can be conceived as a microworld which could join school mathematics and computer graphics together. When supported by creative and attractive activities it could be used for learning by doing and at the same time for applying mathematics knowledge. One of the activities is represented by creating movable models using geometric construction tools.

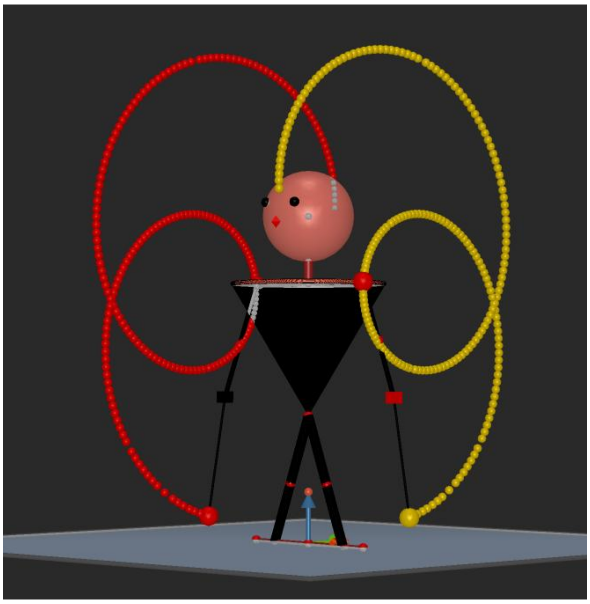


Figure 1. POI dancer and traces of twisting balls on twines, a dynamic geometry figure.

The article deals with a project of modelling simple mechanisms in spatial dynamic geometry software Cabri 3D within a preparatory course for in-service mathematics teachers. The research was to investigate whether the learners were able to leave traditional mathematics teaching approach in order to learn geometry through creating animated models. Three ways of creating computer animation are shown onwards, as well as the principle of a motion in dynamic figure using knowledge of geometry acquired previously is explained. Four difficulty levels of used motion mechanism in a created figure are described. And other experiences referring to the advantages and the difficulties of this activity are commented.

Keywords

Dynamic geometry, mechanisms, modelling, Cabri 3D, pre-service teacher preparation

Dynamic geometry microworld

Dynamic geometry environment represented by Cabri, Geogebra, Cinderella, Sketchpad and other software products, has been recently getting into Czech schools. First experience of using this software as a learning facilitator shows that teachers use computers mostly as a tool for teaching common geometry topics by a standard “paper and pencil“ method. A method which uses this software only as a tool for precise and fast drawing is most common among czech teachers (Vaníček, 2009, p. 78). At the same time, these tools promise much more. They fulfil parameters of a microworld in which students can understand given rules quickly, they can discover, create, project and at the same time they learn mathematics.

Constructionist approach is mainly connected with Logo and other programming environments built on a similar principle. A naturally arising question is whether the educational principles and their results as well as contribution to education proved and realized in Logo might be transferable to any other environment intended for different issues. If the principles of constructionism are similar, the crucial point in constructionist learning depends on the method applied more than the tool used. This article deals with creating computer graphics in another way than usual.

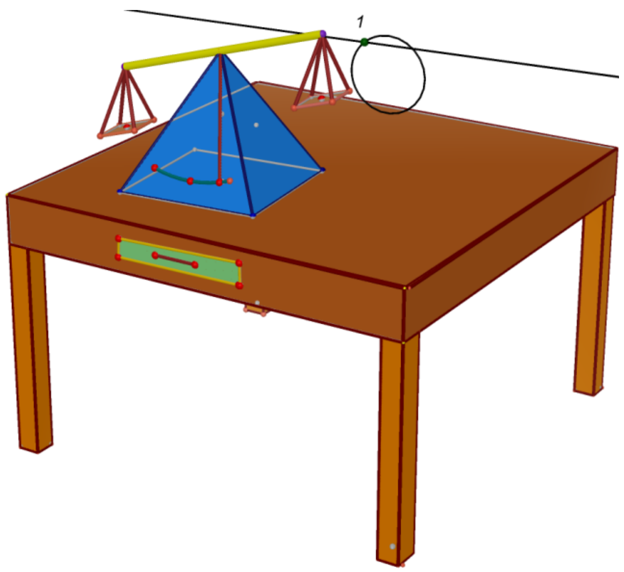


Figure 2. A scale on the table. The internal mechanism of pendulous scale-beams is shown in black (point 1 is moving continuously along the circle and end point of the lever follows the horizontal line).

Dynamic geometry environment software enables learners to work as in a microworld in which they can create graphics by applying geometric knowledge. Their skills that have been learned in classical geometry and improved by constructing figures on the computer can be applied when creating movable models of real objects. For example, it can be a simple engine or any equipment used for moving people or vehicles. Working on open-end tasks can attract students who may feel like programmers while performing. There are two main advantages of this method, a quick progress in learning based on the previous knowledge in geometry and the enriched environment enabling geometric, constructing and project skills development.

Animated graphics creation

One of the benefits of programming in Logo is the graphical output. In addition, turtle drawing comprises of turtle motion itself; some of Logo environments, e. g. Imagine, enable to comprehend turtle as a moving object (Blaho, Kalaš, 2003). According to Freudenthal (1983, p. 342) motion is, first of all, something that occurs to an object in a space and in a time. It is natural that we can see a motion from different points of view enabling us to think of different ways how to describe moving objects. On the contrary, classical paper-pencil geometry approach understands geometrical figures as sets more than objects and motion in geometry is sometimes seen statically as a parametric geometry.

We can split animated graphics creation on the computer among three basic approaches which nature is different (see Figure 3): direct access, programming and (geometrical) constructing. Although there are more simple approaches for dynamic modelling, the advantage of constructing consists in learning mathematics by doing and giving an interesting direction of applying school knowledge in meaningful and driving activities.

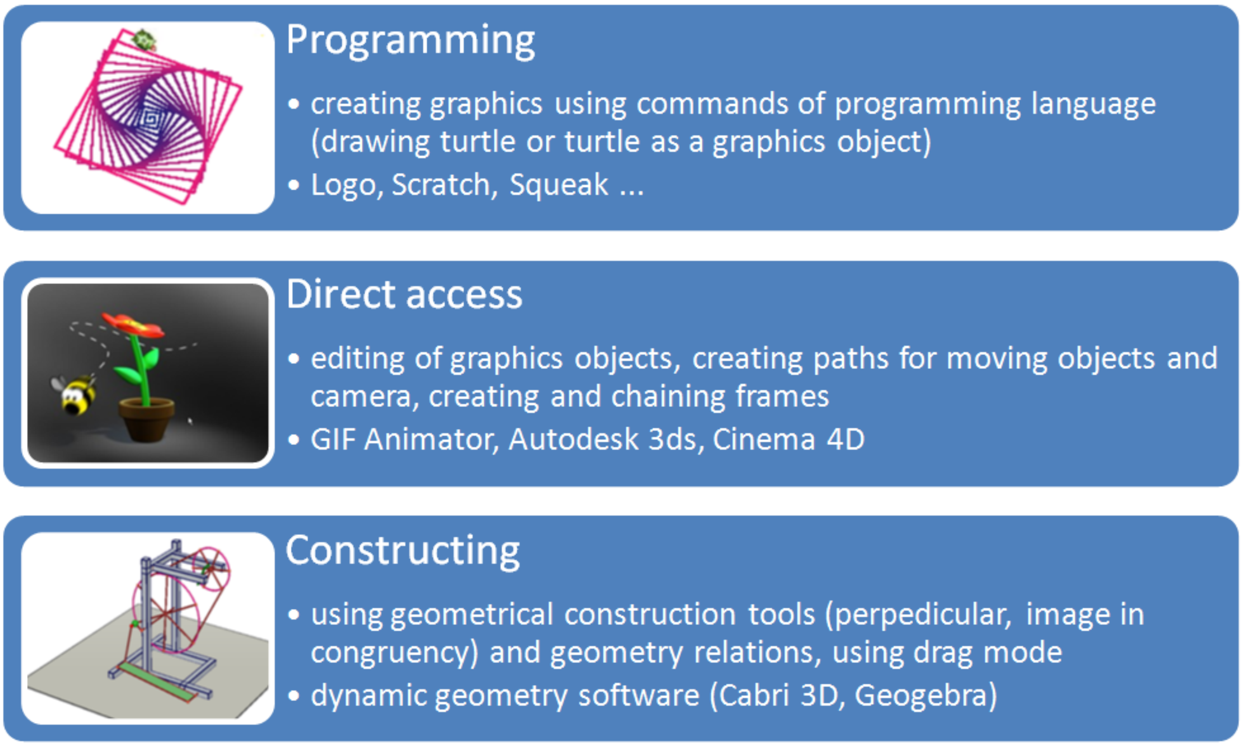


Figure 3. Three types of graphical animation creation on a computer[1]

Motion in dynamic geometry figures is realized mostly by so-called “drag modus”. The user can use a mouse to drag suitable point lying e. g. on a line or circle. When dragging this point along the object lying on the other objects in the figure. They change their positions and shapes according to the relationships between objects. E.g. when a line is going through a moving point it has to go through it even when it is manipulated. If this line changes its slope, another line perpendicular to it has to change its slope, too. A task for the author of the dynamic model is to arrange objects and geometrical relationships among them so that changes in positions and shapes look as a real movement. The principle is the same when using plane or spatial geometry software.

Some dynamic geometry software provides automatic animation of chosen points by a special animation tool. There are some other tools using dynamics as Trace and Locus which “stamp”

[1] Second picture downloaded from: Gudang tutorial et 3Dstuff, http://gudang3d.wordpress.com, 10.1.2010

chosen moving object to the plane and create a new object from a trace. This is the way of creating complicated curves or surfaces in a geometric figure (see Figure 1).

The fundamentals of work in dynamic geometry are very different from creating graphics in Logo. Final models are not robots as no programming structures (alternative boxes, cycles) are used. Algorithms of geometric construction are completely different from algorithms of turtle behaviour. Nevertheless, we are convinced that this environment allows the same learning style in which, “part of an activity the learner experiences as constructing a meaningful product”, in Papert’s words (Award#8751190).

Mechanical models principles

One of the main features of the dynamic geometry environment is the ability to create movable and interactive pictures using geometric objects that can be dragged with a mouse or moved automatically along a given path (Vrba, 2000). Models constructed in dynamic geometry software are simplified models simulating the main function of a movable object. Functional testing and simulation of a dynamic model is carried out either manually in the drag mode or by the automatic animation (which has the advantage of keeping the observer undistracted). (Schumann, 2004, p. 17).

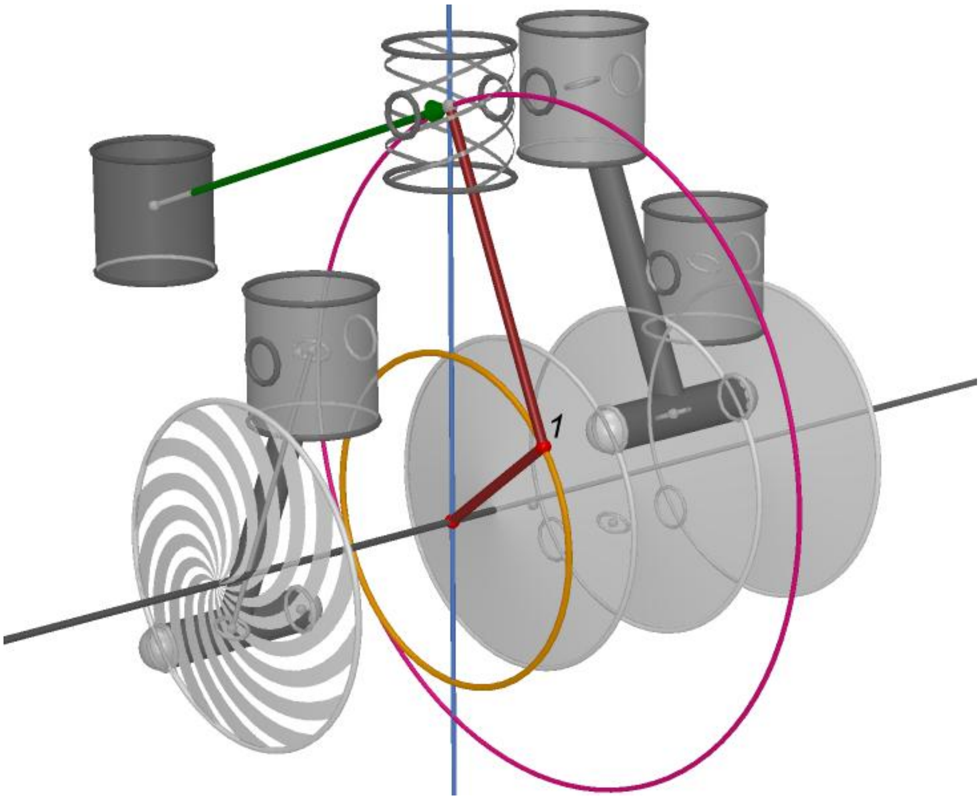


Figure 4. Internal geometry mechanism of an engine model constructed in Cabri 3D, a student's work. Point 1 is moving along a yellow circle, the piston centre is an intersection point of the vertical blue line and the pink circle which ensures constant length of piston rod. The piston (with small hatches on its surface) is an image of a previously created dark cylinder on the left in translation by green vector.

Users can see an internal geometry mechanism which can move the whole figure by dragging one point. Then the users can create graphic figures without knowing any special commands or language unlike other computer applications, such as graphics and movie editors (GIF Animator, Cinema 4D) or programming environments (programming languages). Dynamic geometry allows children to apply skills and knowledge previously learned and related to geometry and elementary mechanics in constructing movable models of real things or situations (e.g. simple engines, motion of figures, animals or vehicles). This activity enables us to compare the computable representation of geometry properties to the real mechanism motion by means of dynamic drawings (González-López, 2001).

What do we understand by simple mechanisms? External appearance or theme does not matter, but internal geometry mechanism is important as it controls movement of the others objects while dragging some points. This mechanism is based on clear geometry relations among objects (perpendicularity, parallelity, laying on, projection) and on the dependence among dynamic figure elements. Actually, this mechanism is often hidden in the figure.

One of the examples describing the internal mechanism is a parallel wire on the front and the rear wheel of a bicycle. If the wheels are of the same size they wheel the same velocity and always need to keep parallelism. The rear wheel could be made as an image of the front wheel in translation because parallelism retains in translation. Another example (Figure 5) describes two cogwheels turning in opposite directions. They might be constructed using reflection.

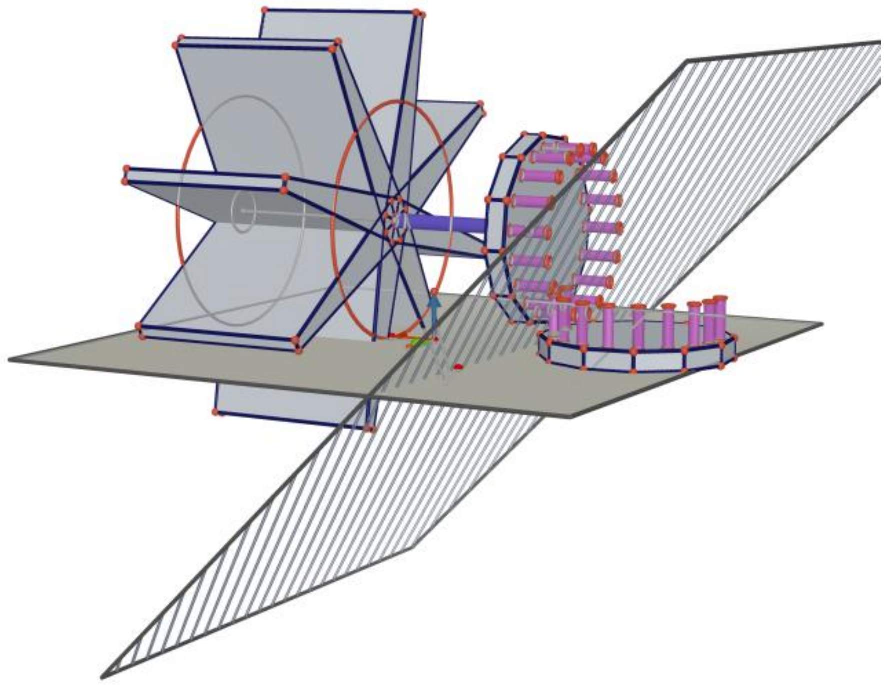


Figure 5 – A model of a water mill made in Cabri 3D. Plane symmetry as a construction step: horizontal transmission wheel is an image of the vertical one with respect of a shaded slant plane.

Spatial geometry modelling

Modelling using dynamic geometry software as a method of teaching geometry is described in various papers dating back to the 90s'. Such open-ended environments as Cabri offer a large range of geometric tools for creating dynamic geometry figures that can be fully animated (Laborde, 1999). Laborde (1996), Gonzáles-López (2001), Schumann (2004) and others used Cabri for reconstruction a set of simple dynamic models of moving elements and machines described by Bolt in his known book “Mathematics meets Technology” (1991).

Spatial modelling has been limited by non-existence of an appropriate environment till lately. Schumann (2003, p. 14) alludes to three restrictions of modelling in dynamic geometry software, the two of which have been topical up to now: first, confinement to geometrically-describable objects and machineries and secondly, confinement to means and methods of dynamic geometry. The third restriction to objects and machineries which can be interpreted as "plane" has expired due to quick progress in development of dynamic environment for 3D geometry. Of course, most of construction steps used for creating 3D models are still static or "2-dimensional"; however, some of the new tools available are to use e.g.: plane symmetry as seen in the Figure 5.

Levels of difficulty of the motion

Dynamic constructions of mechanical models can be sorted by dependency of moving objects to four difficulty levels (according to Vaníček, 2009, p. 139):

Level No. 1 – **independent motion**
Objects move constantly on basic lines (straight lines, circles), some parts of the figure move independently of one another (using several independent moving points).

Level No. 2 – **dependent but not exact**
There is a central object in the construction. Usually it is a point which controls movement of the other objects that are dependent on it and they move according to the central object. The internal mechanism of the figure does not reflect the reality truly. Sometimes an incorrect mechanism is used, which only looks like a correct one (Figure 6, 7).

Level No. 3 – **geometrically exact**
It is the same principle as in the level described above but the internal mechanism of the figure reflects the reality correctly (Figure 4).

Level No. 4 – **complicated motion applying intersection**
Some movable objects are constructed as dependent on some intersection points. If such intersection point exists then the dependent object exists. This technique enables the author to shift visibility of two similar objects upon an actual situation, so that it looks as if there is only one object with a complicated motion in the figure (Figure 8).

Figure 6. A model of a dancing doll in different stages of the motion and from different perspectives. A student's work.

A figure of a dancing doll

Let us describe the difficulty level of the dynamic figure representing a dancing doll (see Figure 6). It is a student's project. The doll periodically bends its knuckle and moves both of its forearms, the animation looks perfect. The whole mechanism is created in the way that a movement of one point (No. 1 in the Figure 7) can set the other movable objects move.

The analysis of the hidden mechanism allows us to find out whether every part is constructed correctly. On the phased figure on the left, you need to know, the big red point 1 is constantly moving along the short cyan arch which is a part of the magenta circle. Point 1 is translated by a solid vector to point 2 and point 3 (an elbow) is created as a vertex of a solid triangle with a fixed point 4. The green triangle remains fixed when in motion.

In the two figures on the right, there is a visible mistake. Points 3 and 2 (knee and foot) are moving along the congruent arches, which is incorrect. Both thighs and calves change their length during the motion of point 1. If point 3 is laid on the circle with the middle point 5, the thigh length would be constant. The difficulty of the construction of the doll is only level 2 because it is not exact geometric construction and no advanced technique is used.

It is necessary future teachers could discover, analyze and discuss these mistakes so that they could deeply understand pieces of knowledge applied and later tutor their pupils in such learning activities.

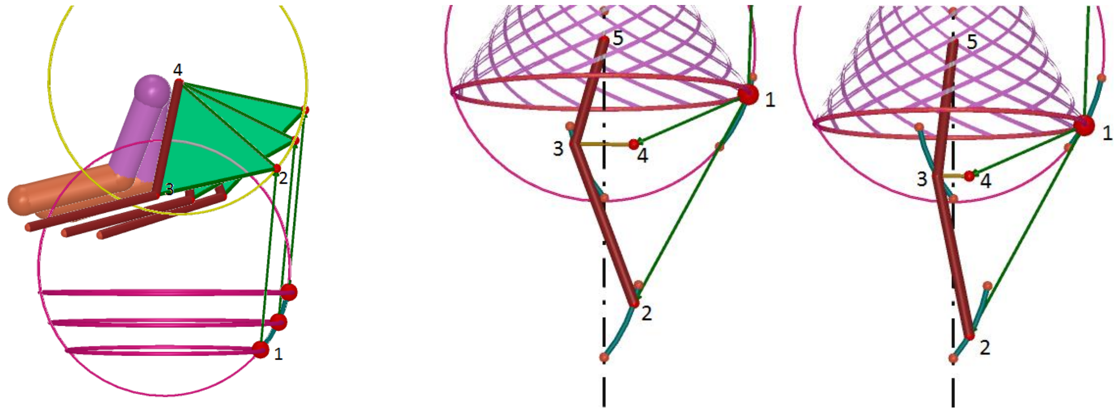


Figure 7. The internal mechanism of the higher (left) and the lower (right) part of the dancing doll. It is evident that the length of both " leg bones" is not constant.

Project of future teachers preparation

When starting to deal with dynamic geometry environment, students meet various difficulties. On the contrary to programming in Logo, which both turtle graphics and work style are completely new to them, the environment of dynamic geometry looks only as a similar to a paper-pencil learning environment. Apparently familiar environment however it has its own specifics fixed in geometry itself. E.g. a point is not mentioned as a place in space but an object. Consequences of this fact are: there could be more than one point positioned on one place, a point lying on a line is not in the same relationship as one to a line going through a point. Moreover, spatial construction uses different construction steps which students do not master as they can when drawing on a sheet of paper.

Other difficulties could be expected from the rank of typical tasks. Student is used to get a problem solving tasks, with an aim exactly described and search for an appropriate algorithm. Creating models seems to be a bit strange to the students, somehow out of topic.

The main reason for realizing such project is to prepare teachers for tutoring such learning activities. The project of creating spatial dynamic models of mechanisms within the preparatory course for future teachers of mathematics was realized at the Faculty of Education, University of South Bohemia this year. We started this project after several years of providing similar project in plane geometry. 17 students experienced in using Cabri 3D took part in this project, most of them had no programming skills. The first part took 7 hours, when students were supposed to create dynamic models in "plane" Cabri first and then they were asked to create a moving figure in Cabri 3D. These "plane" exercises were e.g. creating a rotating prism and a pyramid, a moving engine, a cyclist on a bicycle, a dancing house.

Students seemed to enjoy creating static models, the problem arised when they had to transfer the motion of a point to another part of the figure using construction tools. Students often chose a construction which only evoked correct behaviour of the moving objects, in result some "solid" objects got shorten or disappeared. From a programmer's point of view, it was not such a big problem when students discovered a geometry procedure which described the reality somehow.

Some of the students' models were very elaborate, they contained a lot of objects. For example, the model of a dancing doll contained 43 visible and 118 invisible objects, model of railway gates (Figure 8) contained 99 visible and 289 invisible objects. But we claim that quantity of used objects caused less trouble to students than mastering the essentials of internal geometrical mechanism. Most of objecs are simple to create, they are static or made in the same way.

We acquired some experience allowing us to conclude it is possible to claim that spatial modelling is much more difficult for students than plane modelling for two reasons:

- Handling the Cabri 3D application is difficult as mere "plane manipulation" with a mouse controls spatial manipulation with both objects and camera. There is no appropriate positioning tool which could facilitate the user to orient in space behind the screen where the construction proceeds;
- Students do not have any experience with 3D construction tools as until recently they have constructed only projections of spatial models into a plane; spatial constructing requires animating spatial imagery which is difficult.

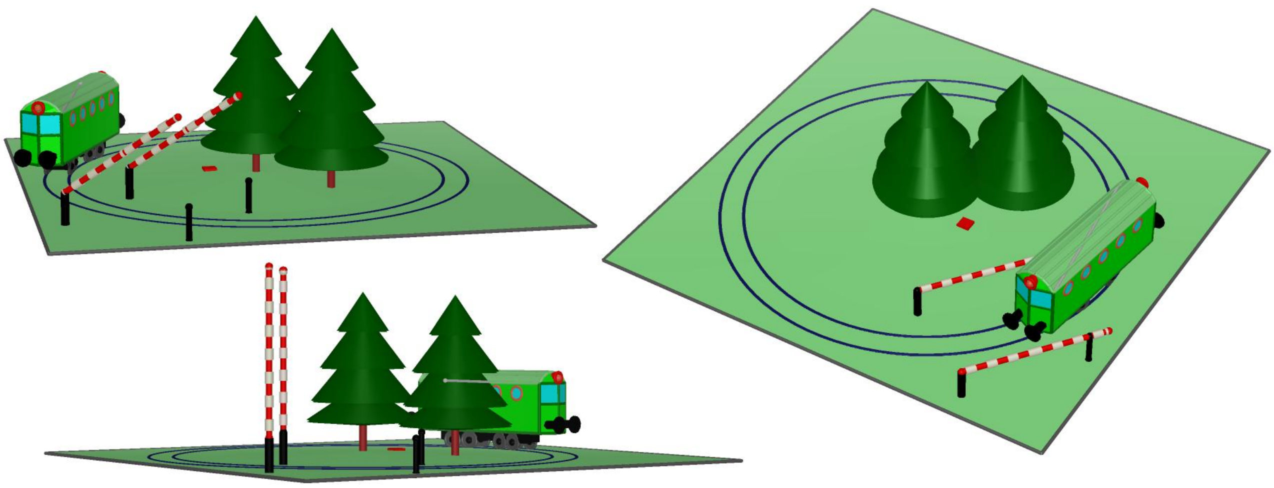


Figure 8. A train and railway gates, a student's work. The gates suddenly start to move when the train is coming closer. The problem of different actions of the gates dependent on the train position (going up and down, staying up or down) is solved by creating three different objects representing the gates which visibility depends on the existence of some hidden intersection points. The model represents difficulty level 4. In these three views, you can see the same train, the same railway and the same trees but three different pairs of gates.

Conclusion

We can conclude that it is worth to use spatial dynamic geometry environment to learn how to create animated models and at the same time direct students to deeper understanding of geometrical concepts which are included in dynamics. Final figures were a good source of discussing the character of geometrical rules and concepts. Such activities suggest how to explain what a creative experiment is in a comprehensive way, which is suitable for trainee teachers who are not able to program.

According to our experience, only less than a half of participating students can enthuse over cretive character of these activities, they have to be extra motivated to work creatively. After finishing quite well and correctly behaving figures, some students expressed that they did not enjoy doing the activity. The fact that a part of students do not like creating at all was a bit surprising finding.

Selected difficulty level of movement in mechanical model depends on the author's mathematical skills. It is interesting that this level does not correspond to the author's creativity. Very creative figures may often consist of very simple movements. A big part of authors reveal their creative potential more by choosing the topic and drawing plentiful static parts of the figure. Not only pupils but also pre-service teachers cannot imagine first what technique might be used to manage nicely and correctly behaving animated figures. If we omit a nice design of the finished figure it might be the novelty of possibilities offered by spatial modelling that made this activity so attractive to the students.

Acknowledgement: The research was supported by the GACR 406/08/0710.

References

Award#8751190. *Constructionism: A New Opportunity for Elementary Science Education.* Retrieved 2010-02-12 from http://nsf.gov/awardsearch/showAward.do?AwardNumber=8751190

Blaho, A., Kalaš, I. (2003*) Imagine Logo, primary workbook*. Cambridge: Logotron

Bolt, B. (1991). *Mathematics meets Technology*. Cambridge University Press.

Freudenthal, H. (1983). *Didactical Phenomenology of Mathematical Structures*. Dordrecht: Reidel Pub.

González-López, M. J. (2001) *Using dynamic geometry software to simulate physical motion.* International Journal of Computers for Mathematical Learning 6: 127–142. Kluwer Academic Publishers.

Laborde, C. (1996) *Geometry as a modelling tool for simple mechanisms*. In Geometry for the world. Texas Instruments Incorporated.

Laborde, C. (1999) *Core Geometrical Knowledge for using the Modelling Power of Geometry with Cabri-Geometry*. Teaching Mathematics and its Applications 1999 18(4). pp. 166 – 171

Schumann, H. (2003). *Modelling with Dynamic Geometry Systems – suggestions for projects in applied geometry*. Powerpoint presentation, Dag- og Aftenseminarium Aarhus, 18.09.2003. Retrieved 2010-02-11 from http://www.mathe-schumann.de/modeling2003/index.htm

Schumann, H. (2004) *Reconstructive Modelling inside Dynamic Geometry Systems*. EduMath, a publication of Hong-Kong Association for Mathematics Education, 19 (12/2004), pp. 3 – 21.

Vaníček, J. (2009) *Počítačové kognitivní technologie ve výuce geometrie*. Praha: Karlova univerzita, 2009. ISBN 978-80-7290-394-8.

Vrba, A. (2000) *Oživlá geometrie*. Matematika, fyzika, informatika 10 (2,3), pp. 105 – 116 & 167 – 174. ISSN 1210-1761

Ideas-to-think-with: Useful pieces of knowledge about natural selection

Aditi Wagh & Uri Wilensky, *aditiwagh@u.northwestern.edu*
School of Education & Social Policy, Northwestern University

Short presentation

The purpose of this study was to examine productive pieces of knowledge middle school children might have about natural selection. In this study, sixth and eighth grader students were interviewed about a scenario of natural selection. The interview was designed to elicit students' ideas and thinking about how natural selection takes place. The data was analyzed from a knowledge-in-pieces framework to identify productive pieces of knowledge that learners used in reasoning about natural selection in a population of butterflies. Preliminary results indicate that students are able to draw on some pieces of knowledge that are productive in helping them explain natural selection. The results have implications for the design of learning environments to help students learn about microevolution.

Keywords

Evolution, knowledge-in-pieces

Constructing Complex Things without Getting Confused - Programming Techniques and Reduction of Cognitive Load

Michael Weigend, *michael.weigend@uni-muenster.de*
Institut für Didaktik der Mathematik und der Informatik, University of Münster, Germany

Abstract

A central issue in practical informatics is coping with complexity. This paper discusses three programming techniques, which are strongly connected to the idea of structuring complex problems and which can be used to reduce working memory load: smart naming, overloading operators and implementing transparency. These techniques can be integrated in constructive classroom projects. The goal is not just to practice specific programming skills but to develop computational thinking (especially the ability to structure) which is useful in many areas.

According to Baddeley's theory of working memory, the part of the cognitive system managing information processing during problem solving has a very limited storage capacity. Each part of a computer program can be seen as an external aid used to reduce cognitive load. Programmers use the program text they have already written, to step on in the development of software. People also use formal program text to document algorithmic ideas and explain them to other persons.

There are several methods to increase the readability of program texts. Since working memory can only store textual information, which can be articulated in two seconds (Baddeley 1998), short names for objects are essential. Polymorphism in object-oriented programming can be used to adopt a familiar concept of activity to a new domain. For example, since childhood we have developed a rather abstract concept of addition, which we can use to create all kinds of algorithms. By overloading the plus operator (+) it is possible to extend the concept of addition to new areas. Adopting existing concepts makes it unnecessary to rehearse and internalize new concepts. Paradoxically we use object oriented programming for overloading operators in order to avoid object oriented *thinking* on a higher level of problem solving.

A third and rather technical method of reducing cognitive load is making necessary but complex activities completely transparent. For example in Python you can define so called *properties* in order to hide the implementation of safe access to object attributes.

All examples in this paper are based upon the programming language Python, which might be called Logo-like for several reasons, mainly because it is easy to learn (low threshold) and supports developing cognitive skills by active programming.

Keywords

programming; naming; polymorphism; Python; working memory

Introduction

Students learn programming at school not (necessarily) because they want to or are supposed to become professional software developers. Many pedagogues have the feeling that programming leads to fundamental competences that are useful in many areas of the lives of future members of knowledge societies and that everybody should learn (e.g. Guzdial, Wing, Schwill).

One of these computational competencies is structuring a complex problem into smaller coherent parts. It is a facet of “computational thinking” (Wing) and a “master idea” of informatics (Schwill). According to Seymour Papert, constructing things is an opportunity to discover “powerful ideas” (Papert, 2000). Just telling such ideas is not enough. The power of powerful ideas has to be *experienced* within a context of activity to be truly understood. Papert uses the example of “probability” (one might also say nondeterminism) to illustrate this (Papert 2000). Imagine a student developing a light-seeking Lego robot. A Logo program controlling its movement compares the outputs of two light sensors. When the light source is more to the left, the vehicle turns to the left, and otherwise to the right. A problem occurs, when there are flat obstacles on the ground, which do not block the line of sight to the light source. The vehicle might stop with turning wheels and never reach the light. Nondeterminism helps to solve this problem. The control program could be extended by a command like this:

```
With probability p turnleft x units
```

Basically this means, that sometimes surprisingly the vehicle does something strange: It just goes to the left. In case it is blocked by an obstacle in that very moment, this arbitrary movement helps to get free again and finally find the way to the light. In case it is not blocked this movement leads to a more erratic path, but (if p and x are small enough) it does not hinder the vehicle from going to the light. In Papert’s words the idea of “probability” is *powerful in its use*, because students can experience that it is useful to solve real problems. Additionally this idea is *powerful in its connections*, which means that can be used in several different domains. For example “probability” is essential to explain evolution. And finally it is *powerful in its roots*, since the comprehension of this idea is based upon intuitive knowledge, kids already know, making them feeling powerful.

In this contribution I am going to discuss ideas connected to a central aspect of problem solving, namely the reduction of working memory load by applying certain techniques like naming, polymorphism and transparency.

Working memory and computer programming

We need structuring because the capacity of our working memory is very limited. During problem solving our mind processes chunks of information very quickly and partly in parallel. In moments of “hard thinking” we focus attention to just a few things. The part of the cognitive system managing this, is called working memory (Baddeley 1998, 2003, Dehn 2008). According to Baddeley’s model, working memory consists of several subcomponents for storing verbal and visual information: the Phonological Loop and the Visuospatial Sketchpad. The Phonological Loop is analogue to an audiotape with very limited length. It is able to store verbal content (like variable names and function names), which can be articulated by the subject in at most two seconds. That implies for example that word length affects memory span. You can handle more variable names when they are short. The Visuospatial Sketchpad is supposed to have similar limitations but these have been less well assessed yet.

One of the consequences of the small storage capacity of working memory is that we have to develop big knowledge structures “piece by piece” using external aides like pictures, diagrams and text documents. The process of computer programming can be considered as building up a system of external aides for further development. Basically you start with a few lines of code. When you are sure they work correctly, you go on and add more lines of code using the existing text as external aid.

Thus, readability of the text is essential for development progress and not just for maintaining existing programs. The quality of the language style (including class structure and naming) is checked all the time during development and usually needs improvement from time to time. When you get stuck it might be caused by an inefficient external aid for your working memory. This point is stressed especially in Extreme Programming (Beck 1999). Refactoring the whole program in order improve its technical quality (including readability) is a feature of this methodology. While efficiency aspects can be handled in an isolated "tuning phase" at the end of a project, readability of already existing code is crucial *in each phase* during the development. A bad structure or missing naming conventions is a burden which might slow down the development speed or – especially in case of novice programmers – even block the process totally.

Smart naming

"Code is read much more often than it is written" (Guido van Rossum)

Style guides give much advice how to name objects. 30% of the Python style guide is about naming conventions (van Rossum 2009). This includes rules like "use capitalized identifiers as class names and non-capitalized identifiers for attributes, methods and class instances" or "a name should express the role or meaning of the named object within the algorithmic context".

Some rules like the first example represent some traditions within the programmers' community, but nothing more. Still, these conventions increase readability. The second example goes deeper. A meaningful name like *count* or *sum* represents a chunk of algorithmic information and thus helps to understand a program. When you read the word sum you immediately think of addition and the concept that the sum of some numbers is stored in this variable. It is of advantage that you already know the concept of a sum. There exist some very short algorithms, which are still difficult to understand, because the identifiers do not represent well known concepts.

Explicit naming implies structuring. You do not use an existing name and refer to an object (one step) but you introduce a name first and use the new name then (two steps). This might lead to an immense reduction of cognitive load. Example: The following lines of Python code calculate the size of a window, which is part of a house.

One step (no naming):

```
size = (house.floors[floor_nr].windows[window_nr].height *
        house.floors[floor_nr].windows[window_nr].width)
```

Three steps (introducing names):

```
window = house.floors[floor_nr].windows[window_nr]
height = window.height
width = window.width
size = width * height
```

By introducing the names `window`, `width` and `height`, the reader is supported to divide the process of verifying the logical correctness in three independent parts. Thus, introducing names is a kind of semantic preprocessing. The program text is written taking into account that the working memory capacity of a future reader is limited. Moreover, the *programmer* can use short names like *height* and *width* instead of complex references to develop an algorithm. During the problem solving process it is now possible to keep *additional* relevant chunks of information in working memory and process them in order to find a solution. Structuring by naming is also common in everyday language: “I have got a Renault Megane Scenic, built in the year 2001. For *this car* I need a new headlight and a screen wiper blade.” “Tim is my brother’s youngest daughter’s dog. Yesterday I saw *Tim* alone on the road to the forest.”

High school students learn how to use names for modeling and problem solving in science: “Let c be the concentration of hydrochloric acid in a specimen…” These examples suggest that smart naming is not just a technical skill of specialists but part of general competences like problem solving and communication.

In summer 2009 I conducted a study on the question to what extent students without former computer science education are capable to apply naming techniques, when they have to write algorithms (Weigend 2010). One type of workshop was about referring to objects within a complex three-dimensional structure. In phase 1 the students had to follow given instructions identifying ten broken parts of a fictive power plant on Mars. These instructions contained different types of naming and referring, for example

(1) The pyramid in the right corner of the platform is called *corner pyramid*. Write number 1 on the *corner pyramid*.

(6) When you move from the *corner pyramid* along the edge of the platform to the left, you reach a cube. Write 6 on this cube.

The name corner pyramid (introduced in instruction 1) was later used in instruction 6 making a path-like reference simpler.

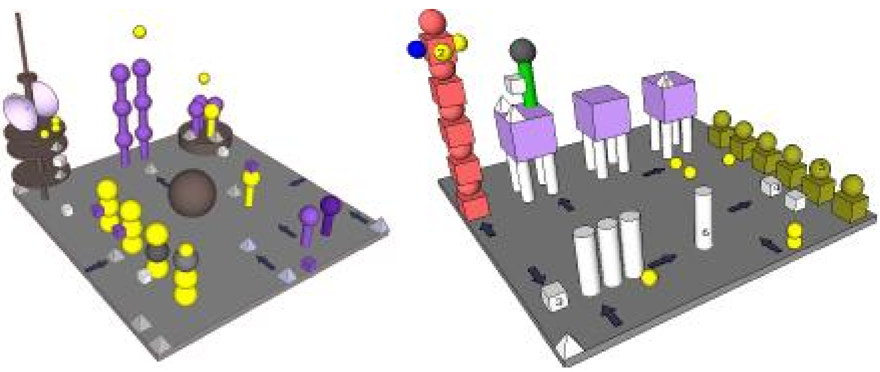


Figure 1. 3D-structures used in “Mission to Mars 3” (Weigend 2010)

Most of the 49 students (grade 9, average age 15.5 years) had no problems to solve this task. In average 95% of the instructions were interpreted correctly. In phase 2 the students had to write similar instructions by themselves. They got a picture showing a different structure, representing a fictive factory on Mars. Six parts were labeled with numbers. The task was to write references to them. Later, the students gave their instructions to classmates who had to identify the parts on an unlabelled picture of the factory. So we had a social situation in which it was important to create readable algorithmic text. The question was: Which of the naming and referring techniques from phase 1 did the students adopt in phase 2?

Only 23 out of 49 students used some kind of naming concept for individual entities. And not more than just three of them explicitly introduced a name (like *corner pyramid*), which they used in other instructions later. Avoiding explicit naming could also be observed in other workshops, where participants had to read and write algorithms for drawing two-dimensional ground plots (Weigend 2010). This suggests that smart naming – which I consider being a facet of computational thinking – is not just learned en route during normal socialization but needs to be taught explicitly.

What can teachers do to encourage smart naming in a programming project? Since naming is essential in every program, the content or topic of a programming task does not matter. Methodology is more important. Extreme Programming (Beck 1999) seems to be an appropriate approach for classroom projects, since it provokes communication and quick development and leaves space for experimenting and learning (Weigend 2005). Let me just mention four features: (1) Pair programming. Two developers share one computer and work together. Thus, they talk about the evolving program text all the time. Names are not arbitrarily chosen but discussed. (2) Development in short iterations. Students select "stories" (short descriptions of functionality) and implement them. At the end of an iteration (which might take just two lessons) there is a runnable program which can be presented and discussed. (3) Refactoring. From time to time the whole program is restructured. This includes changing names, in order to get a better language quality. In contrast to traditional software engineering this is not seen as an annoying disruption which should be avoided, but as a necessity. (4) Collective ownership of code. Rephrasing program text is made easy because each member of the team is allowed to change each line of code. The team (consisting of several pairs) shares the responsibility for the whole project.

Overloading operators – using familiar concepts in new domains

Structured computer programming implies dividing a complex problem into smaller parts by defining functions and/or classes. According to working memory theory the parts are worthless for further program development unless the developer is really familiar with them. A function must be an intuitive coherent piece of knowledge (chunk). If it is not, the developer has to rehearse its usage until she or he has completely internalized its effect. Otherwise the developer is in a split attention situation (Ayres & Sweller 2005). She or he has to look up the function definition (which is written at a different place within the program) and memorize it explicitly while trying to formulate an appropriate function call.

Andrea diSessa states that there are relatively few abstract intuitive concepts people use for problem solving within unfamiliar domains (diSessa 2001). He calls them phenomenological primitives (p-prims). He observed that even well trained scientists use intuitive concepts (like resistance or friction), when they encounter a problem which is new to them. Adopting an abstract concept to a specific situation seems to be easier than developing a completely new concept. In the next section I am going to discuss the concept of addition as an example.

The concept of addition

People use the concept of adding since kindergarden. In mathematics, adding is an arithmetical operation on numbers. In primary school adding natural numbers is often introduced adopting the metaphor "arithmetic is collecting things" (Lakoff & Nunez 1997). A number is represented by a collection of things of one kind, e.g. beads. The operation 2 + 3 can be visualized like this: put two beads on the table, add another three beads and then count the resulting number of beads on the table. In this domain subtraction means taking away some beads from the table. Another metaphor common in elementary math teaching is "arithmetic is moving along a line". Visualizing the set of real numbers by a horizontal line, the mathematic term 2 + 3 -1 can be represented by this sequence: Move to the right for two steps. Move to the right for 3 steps. Move to the left for one step.

The activities I have just described can be seen as metaphors for an arithmetic operation. But vice versa the arithmetic operation addition can also serve as an abstract model of real life activities like moving, collecting or concatenating.

In informatics this is called polymorphism or – more specific – overloading of operators. Joseph Bergin states that “polymorphism and not the class concept is the big idea of object-oriented programming” (Becker et al. 2001, p. 410). The funny thing is that polymorphism allows the programmer *not* to think object oriented. You have a rather abstract operation in your mind – in the world of OOP one might say a type of message – and use it for algorithm development. The operation exists *independent from objects*. The details of the execution are of secondary importance, they are implemented within the class definition and you need not to think about them while developing an algorithm. Instances of different classes may react in a slightly different way to the same message, but the meaning of the message is – on an abstract level – the same.

Overloading operators in Python

In Python operators and built-in functions can be overloaded by defining “magical” methods with certain names starting and ending with two underscores. The following Python statement defines a simple class modeling length (example taken from Weigend 2010a).

```
class Length(object):
    meter = {'mm': 0.001, 'cm':0.01, 'm':1, 'km':1000,
             'in':0.0254, 'ft':0.3048,
             'yd':0.9143, 'mil':1609}                 #1

    def __init__(self, value, unit):
        self.value = float(value)
        self.unit = unit

    def getMeter(self):
        return self.value * self.meter[self.unit]

    def __add__(self, other):                         #2
        s = self.getMeter() + other.getMeter()
        return Length(s/self.meter[self.unit], self.unit)
    ...

    def __repr__(self):
        return str(self.value)+' ' + self.unit        #3
```

An instance represents a length or spatial distance through a float number and a unit like *cm* or *ft*. The class attribute meter (#1) is a dictionary mapping units to the number of meters they represent. For example 1 millimeter is equal to 0.001 meters. Overloading the plus-operator is implemented by defining method `__add__()`. Each time the Python interpreter evaluates an expression like a + b, it sends a message like `a.__add__(b)` to object a. In the same way the developer can define further operations, including subtraction, multiplication, division or Boolean functions like <, >, == and so on. The method `__repr__()` returns a printable representation of the object.

Like Logo, Python can be used in an interactive mode. After executing this script the class can be used and tested. Behind the prompt >>> you type a single statement. It is executed immediately after you have hit the ENTER-key. Any output is displayed in the subsequent line. This is an example dialog:

```
>>> foot = Length (1, "ft")
>>> earth_diameter = Length(12713.507, "km")
>>> earth_diameter + foot
12713.5073048 km
```

```
>>> Length(2, "cm") + Length(2, "in")
7.08 cm
```

The concept of adding in the abstract meaning of “putting together” is also used in the context of blending substances. We *add* sugar to lemon juice to make it taste sweeter and we use the plus-operator in chemical reaction equations. Instances of the following Python class represent potions, consisting of different ingredients. The content of a potion is stored in a dictionary mapping the name of an ingredient (like lemon juice) to the amount to which it is contained (number of grams).

```
class Potion(object):

    def __init__(self, ingredient=None, grams=None):
        if ingredient:  self.content = {ingredient:grams}
        else: self.content = {}                            # empty
dictionary

    def __add__(self, other):
        result = Potion()
        result.content = self.content.copy()
        for i in other.content:
            if i in self.content:
                result.content[i] += other.content[i]      # update
ingredient
            else: result.content[i] = other.content[i]     # new
ingredient
        return result

    def __repr__(self):
        c = self.content
        s = "Substance consisting of \n"
        for i in c:
            s += i + "("+str(c[i])+ " g)\n"
        return s
```

In the following dialogue with the interactive Python shell, I make lemonade in three steps and then display the result. Note that the meaning of the +-operator in this context is quite easy to comprehend without any knowledge about the class definition. Actually, this formal program text could be used to explain another human how to make lemonade.

```
>>> sparkling_water = Potion("water", 998) + Potion("carbon dioxide",
2)
>>> syrup = Potion("lemon juice", 50) + Potion("sugar", 100) +
    Potion("water", 100)
>>> lemonade = sparkling_water + syrup
>>> lemonade
Substance consisting of
water(1098 g)
carbon dioxide(2 g)
lemon juice(50 g)
sugar(100 g)
```

Models of blends are useful in many areas. In Witten, the city where I live, there is a steel company producing high quality steel from scrap. Basically, the electric arc furnace is charged with different kinds of scrap in such a clever way that the resulting blend has the exact chemical composition of the target product. A computer program calculates the required amounts of various scrap (and pure alloy). Such program should contain a class modeling the chemical composition of scrap charges similar to the class `Potion` above.

Dealing with fuzziness

Intuitive concepts like “addition” are fuzzy. This is an implication of being abstract. Abstraction (Latin abstrahere = to take away or to remove) means that you focus on the important and ignore the unimportant. Abstraction is essential when you process a concept in your working memory while creating an algorithmic idea on a high level. But when it comes to implementation as a runnable computer program you have to explicate the ignored details in further steps of development. In this perspective, overloading an operator means removing the fuzziness of a familiar (intuitive) operation you are applying to a new domain.

Figure 2 shows two screenshots from an application developed with Python, which evaluates a voting by raising voting cards. First the user loads a picture file (in figure 2 you see a simplified voting situation: yellow cards on a gray carpet). Then she or he clicks on a voting card on the picture to tell the program the color of the card. This color is shown on a label at the bottom of the application window. Due to varying illumination the color of the voting cards on the picture is slightly different. In the next step the user has to define a tolerance for the color recognition using a slide. The selected color is split in a darker and a lighter version (see right screen shot). After pressing the button with the raised hand, the program counts the voting cards and displays the result.

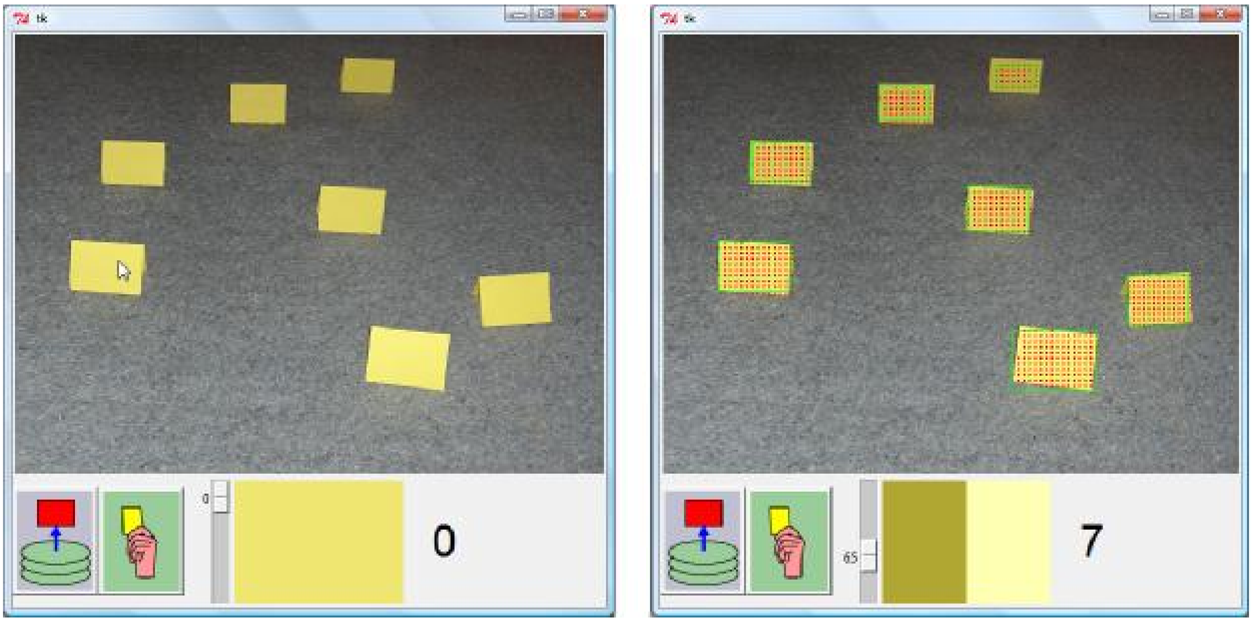


Figure 2. Screenshots from a graphical Python program, which counts voting cards.

The whole Python program consists of 200 lines of code and is too long to be discussed completely here. I focus on a few lines using overloaded operators. The algorithm of counting is mainly implemented in this iteration:

```
c = Cards()                                                #1
for (x, y) in self.grid:                                   #2
       if self.__get_color(x, y) == self.color:            #3
           self.pic.put("{red red} {red red}", to=(x,y))   #4
           c.extend(Dot(x, y))                             #5
```

#1: This represents a list of voting cards.

#2: The list `self.grid` contains pairs of numbers `(x, y)` indicating pixels of the photo which have a certain distance to each other. (This is to get a better performance. We need not to check every pixel).

#4: Draw a red dot on the picture. So the user can see that the program has found a pixel probably belonging to a voting card.

#5: Update the list of voting cards c. This goes like this: If the pixels is at the border of an already found voting card, this card is extended now including the area of the new pixel. Otherwise a new voting card is created covering just the area of the new pixel.

I did not write a comment to line #3. However, do you understand its meaning? Probably you do. The meaning is: " If the color of current pixel is the color of voting cards, …"

Suppose the voting cards are yellow. In this case we can say: "If the current pixel is yellow, it probably belongs to a voting card, and we have some business to do." This is a comprehensible idea. It is simple. Still, it represents the very core of a rather complex algorithm. The idea is comprehensible because it is a combination of just a few chunks of information, which we can process in our working memory. But the prize for simplicity is fuzziness. For example, what does it mean to claim "the pixel is yellow"? In line #3 two objects representing colors are compared by means of the equal-operator ==. In this context a fuzzy concept of "being equal" is enough to understand the if-statement. We trust that the method implementing the comparison works in an appropriate way. Obviously, this is not the usual arithmetical concept of equality. The program contains a class representing color-objects. And here the equal-operator == is overloaded by defining a method with the name __eq__:

```
def __eq__(self, other):
       t = self.tolerance                # use a shorter name
       return (abs(self.r - other.r) <= t) and \
              (abs(self.g - other.g) <= t) and \
              (abs(self.b - other.b) <= t)
```

This method returns the Boolean value `true`, if and only if the differences of the rgb-values of the compared color-objects are within a tolerance interval.

Transparency and information hiding

In OOP an object is sometimes described as a fortress keeping most of its internal structure hidden from the public. The attributes are private, nobody in the environment is allowed to read or change them by immediate access via assignments like `object.attribute = newValue`. But the environment may use special methods – called getters and setters – to read or change some attributes. The setters may contain safeguards protecting the internal data from invalid changes.

Consider a class Label, which models labels for articles offered in a shop. A Label-object has two attributes: the price of the article and a text with at most 12 characters. These two attributes are declared as private. But for each of them a setter and a getter method is defined, which allow controlled access. Fig. 3 shows on the left hand side the corresponding UML class diagram.

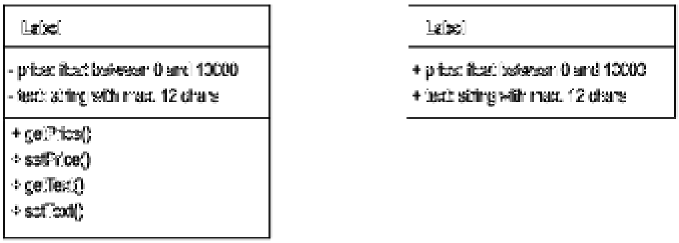


Figure 3. UML class diagrams

The point is that both attributes represent information which is meant for the public and not just for internal calculations. So these attributes are intended to be public. The class diagram with public attributes and without the setters and getters would be much more simple and comprehensible (fig. 3 right diagram). Python offers a mechanism to define attributes, which “look public” from the outside. Still, the access to them is controlled by special private methods. This technique is called properties. The following script illustrates how to use it.

```
class Label:
    def __init__(self):
        self.__price = 0.0                  #1
        self.__text = "no name"

    def getPrice(self):
        return self.__price

    def setPrice (self, x):
         assert 0 <= float(x) < 10000       #2
         self.__price = float(x)            #3

    def getText(self):
        return self.__text

    def setText(self, t):
        try:
            self.__text = str(t)[:12]       #4
        except:
            self.__text = "no name"

    text = property(getText, setText)       #5
    price = property(getPrice, setPrice)
```

The method `getPrize()` returns the present value of the private attribute `__price` (see line #1). Attributes starting with two underscores are private and cannot be accessed directly.

Method `setPrize()` changes the value of the private attribute `__price`. The assert statement (line #2) checks whether or not the argument of the method call is a number between 0 and 10000. If this condition is not true, an exception is raised. That means the program run is aborted and the Python interpreter outputs a message about that. Python does not require typing. Statement #3 contains a casting and makes sure that the private attribute __price gets a floating point number.

In contrast to the first setter, `setText()` does not raise exceptions. It contains a `try` … `except`-statement to prevent abortions caused by inappropriate arguments. Statement #4 guarantees that the attribute text is set to a string with at most 12 characters.

In the last two lines of the script, two so called *properties* are defined. The (public) names `price` and `text` can be used in assignments like regular attribute names. In the following statements I check the effect of the safeguards.

```
>>> t = Label()        # create an instance of the class Label
>>> t.price = 10       # assign the integer object 10
>>> print(t.price)     # the attribute value is a floating number
10.0
>>> t.price = -1       # invalid for Label-objects
Traceback (most recent call last):
  File "<pyshell#8>", line 1, in <module>
    t.price = -1
  File "…/label.py", line 10, in setPrice
    assert 0 <= float(x) < 10000                                  #2
```

Using properties instead of public `set()` and `get()` methods does not mean overall reduction of complexity but just redistribution of complexity. The access to attributes from the environment of the class becomes simpler. But the class definition becomes more complex. The total complexity cannot be reduced.

The motive for controlling access to attributes is to provide logical safety. A complex system must contain mechanisms that support error detection. For example, some erroneous program component might try to assign a negative price-value to a Label-instance. But this security aspect is completely irrelevant for a programmer, who is developing a routine that just *uses* Label-objects. During the algorithm development it would be an unnecessary burden on working memory. One facet of computational thinking is to be aware of this and to avoid such burdens.

Discovering the poetry of computer programs

Professional computer programmers have to deal with huge class libraries. They use complex development environments with IntelliSense-like functionality, which help to find an appropriate method for a given purpose. One might claim that students should rather learn how to use repositories and integrated development environments (IDEs) like Eclipse, in order to get a more realistic idea of modern programming. But in this case the programming language loses its function as means of expression. A well written program is a document made for humans. Like a diagram or a natural language text, it can be used for creating and communicating intellectual content. For example textbooks on bioinformatics contain programs explaining natural information processing. On the other hand, a complex program, consisting of highly specialized parts, created with the support of semiautomatic code generators is not a well readable document any more. At this point classroom programming – focusing on computational thinking – differs fundamentally from professional software development. Classroom activities at schools and partly even at universities are not a vocational training for future software engineers. In these educational contexts a programming language primarily serves as a means to express ideas rather than as engineering technology to build efficient and secure software.

Let me conclude with two suggestions for teachers, who are organizing constructionist computer science education and intend to create an environment that inspires students to discover the expressive power of programming language constructs.

(1) Encourage students to communicate their algorithmic ideas and programs to other people. The poetry of programming text can only be discovered when students read it and talk about it all the time. A social environment based on the ideas of Extreme Programming might be a good approach. There should be space for experimenting and refactoring.

(2) Create interesting software project ideas based on a model of some aspect of reality (like length, color, blends etc.). Usually such model is a class, of which a rudimentary version might

be prepared by the teacher. The language constructs, which the teacher wants to be learned, are not interesting per se to the students, when they start a project. The genuine motives to construct a product are rooted somewhere in the everyday life of the students. They just want to create something cool. The academic learning – including discovering "big ideas" – takes place en route.

References

Ayres, P. and Sweller, J. (2005) *The Split-Attention Principle in Multimedia Learning*. The Cambridge Handbook of Multimedia Learning, Richard E. Mayer, Ed. Cambridge University Press, pp. 135–146.

Baddeley, A. (1998) *Recent Developments in Working Memory*. Current Opinion in Neurobiology, Vol. 8/2, pp. 234—238.

Baddeley, A. (2003) *Working Memory Looking Back and Looking Forward.* Nature Reviews Neuroscience, Vol. 4, pp. 829—839.

Beck, K. (1999) *Extreme Programming Explained.* Addison Wesley.

Becker, B. W.; Rasala, R.; Bergin, J; Shannon, Ch.; Wallingford, E. (2001) *Polymorphic Panalists*. Proceedings of the 32nd SIGCSE technical symposium on Computer Science Education. Charlotte, pp. 410−411.

Dehn, M. J. (2008) *Working Memory and Academic Learning*. John Wiley & Sons, Hoboken, New Jersey.

Guzdial, M. (2008) Paving the Way for Computational Thinking. Communications of the ACM Vol. 52 No. 8, pp.25—27.

Lakoff, G., & Núnez, R. E. (1997). *The Metaphorical Structure of Mathematics: Sketching Out Cognitive Foundations for a Mind-Based Mathematics.* In L. D. English (Ed.) Mathematical Reasoning. Analogies, Metaphors, and Images, Lawrence Erlbaum Associates, pp. 21–92.

Papert, S. (2000) *What's the big idea? Toward a pedagogy of idea power.* IBM SYSTEMS JOURNAL, Vol. 39, pp. 720−729.

Schwill, A. (1993): *Fundamentale Ideen der* Informatik *[Fundamental ideas of Computer Science].* Zentralblatt der Mathematik 20, pp. 20–31.

diSessa, A. A. (2001) *Changing Minds. Computers, Learning, and Literacy*. MIT Press, Cambridge, Massachusetts.

van Rossum, G. (2009) *Style Guide for Python Code* (PEP 8, version 68852), (last modified 2009), http://www.python.org/dev/peps/pep-0008/

Weigend, M. (2005) *Extreme Programming im Klassenraum [Extreme Programming in the classroom].* INFOS 2005, München, Germany, http://www.fernuni-hagen.de/schulinformatik/xp_weigend.pdf.

Weigend, M. (2010) *Mission to Mars. A Study on Naming and Referring*. In ISSEP 2010 Proceedings, J. Hromkovic, R. Královiè, and J. Vahrenhold (eds), LNCS 5941, Springer, Heidelberg, pp. 185−196.

Weigend, M. (2010a) *Objektorientierte Programmierung mit Python 3. [Object Oriented Programming with Python 3]*. 4th edition. Bonn, mitp.

Wing, J. M. (2006) *Computational Thinking* Communications of the ACM, Vol. 49, No. 3, pp. 33-35.

Restructurations: Reformulating Knowledge Disciplines through New Representational Forms

Uri Wilensky, *uri@northwestern.edu*
Learning Sciences, Computer Science and Complex Systems, Northwestern University

Abstract

The goals of instruction are usually taken to be fixed, at least in their broad outline. For example, in elementary school mathematics, students progress from counting, to addition, multiplication, and fractions. Given this state of affairs, the business of educational research has been to determine how the fixed instructional aims can best be reached. Education researchers have traditionally asked questions such as: What are the typical difficulties that students experience? Which means of instruction – method A or method B – is better for achieving our instructional aims?
In contrast, we will describe a line of work in which we have shifted the focus from the *means* to the *object* of learning. We are concerned with how the structure and properties of knowledge affect its learnability and the power that it affords to individuals and groups. We briefly review three agent-based restructurations of traditional science content and discuss the consequences for scientific power and learnability.

Keywords

Epistemology, representation, computer-based modeling, agent-based modeling

Introduction

The advent of powerful computation has brought about dramatic change in many areas of life including dramatic changes in the practice and content of science. But, to a great extent, these dramatic changes have not resulted in significant change in the world of education. The authors of this paper have worked for many years in a more or less loosely coupled interaction on projects directed at bringing the benefits of these changes to students. This paper develops a conceptualization of this enterprise in historical and epistemological terms that go beyond computers and suggest broad new directions for the sciences of learning.

As a first step to presenting this conceptualization, we look back historically at changes in science that had significant benefits for both scientists and learners. The example that we have found most useful in presenting our idea is the shift from Roman to Hindu-Arabic numerals in arithmetic. This was not done with an "educational intent." But it had profound consequences for education. The new direction suggested in the paper is to study more systematically changes of this kind, to examine the practices of science in search of cases that could but have not had similar educational consequences and to consider the possibility of deliberately making such changes in thinking about scientific (and, indeed, other) topics with an educational intent. The conceptualization of our own projects developed in the paper presents it as exemplifying this direction of work.

We begin by looking more closely at the Roman to Hindu-Arabic transition through the lens of a thought experiment:

A thought experiment

Imagine a country, FOO, where people represented numbers as the Romans did, using symbols such as MCMXLVIII. Learning Science researchers in this imaginary country were very concerned with the difficulty of learning to handle numbers, and they worked hard to make these skills accessible to more of their citizens. They engaged in a number of different approaches. Some researchers collected the misconceptions and typical mistakes made by children. For example, they might have discovered that some children believed that since CX is ten more than one hundred, then CIX must be ten more than CI. Others constructed and studied computer programs that allowed students to practice numerical operations. Still others constructed specially developed manipulatives --wooden blocks marked with the symbols C, X, V, and I— to help students learn. Yet another group tried to elucidate the problem by framing it in evolutionary terms, speculating that perhaps humans were just not wired to do multiplication and division. It is not hard to imagine, in our thought experiment, that many of these approaches brought about substantial improvement in learning. But let us now imagine that, at some point, Hindu-Arabic numerals were invented by the educators of this country. This invention then opened up a new way to handle and think about numbers. Resulting gains in educability towards a functional numeracy would likely far outstrip any of the benefits that would have accrued from any of the improved techniques for teaching with the Roman numeral system. Before: the learning gap in arithmetic was immense: only a small number of trained people could do multiplication. After: multiplication became part of what we can expect everyone to learn.

This parable is not intended to show that the other approaches were wrong. They added knowledge that would likely have useful applications even after the shift in representations. But the point is that the most dramatic improvements did not come from what we usually think of as the main part of the science of learning.

In point of fact, Hindu-Arabic numerals were not invented with an educational intent. But they could have been, and that allows us to show the need for a new branch of the learning sciences with the mission of understanding, facilitating and even designing shifts similar to the shift from Roman to Hindu-Arabic numerals.

A first step is to name the sort of innovation associated with the shift from Roman to Hindu-Arabic representations of number. This sort of transformation has no name in the standard educational discourse. It is not sufficient, for example, to say that we have a new “curriculum,” or a new “instructional approach.” Even in this simple case, the algorithms that are taught, students’ mental representation, their sense of systematicity in the field, psychologically important landmark values, and even social embedding (“who can do what,” e.g., scribes for the emperor vs. modern carpenters or business people) changes. In our terminology, we will say that we have a new *structuration* of a discipline. The main thrust of this paper is to flesh out this term through concrete examples. But, for now, we introduce a preliminary formal definition: By structuration we mean: the encoding of the knowledge in a domain as a function of the representational infrastructure used to express the knowledge. A change from one structuration of a domain to another resulting from such a change in representational infrastructure we call a *restructuration*.

Our Roman-to-Arabic example is just one of many examples we could have chosen. diSessa in his book *Changing Minds* (2000) describes the historical restructuration of simple kinematics from a text-based to an algebraic representation. He illustrates the restructuration though a story of the 16th century scientist Galileo. He describes Galileo struggling to handle a problem involving the relationship between distance, time and velocity without being able to appeal to algebraic notations such as d=vt. The central new idea in his book is exemplified by this representation of algebra as an *epistemological* entity capable of transforming what was a complex and difficult idea for as powerful an intellect as Galileo’s into a form that is within the intellectual grasp of every competent high school student. The vista opened to the imagination is dramatic: if the problems with which we struggle today could be so transformed, think of the new domains we could enter and conquer! Or as educators we might take the prospect in a different slant: if algebra could make accessible to students what was hard for Galileo, our holy grail should be whatever can similarly transform what is hard for them today. This is the quest on which we and diSessa are embarked. A subtext that is left implicit in his description, perhaps out of politeness to his colleagues, is the relative puniness of what mainstream educators would have brought to bear on helping students understand Galileo's thinking: design a curriculum, introduce manipulables, create a learning community, embed it in a computer game. Surely bringing all the machinery of How *People Learn*[1] to bear on teaching Galileo's students would have resulted in improved understanding of kinematics, but inventing algebra did better by a long, long shot.

You can imagine the reply of our FOOian learning scientists if somebody proposed to develop Hindu-Arabic numerals to solve the social problem of the low proportion of the population that could pass the multiplication tests. The funders of the Fooian Learning Science Foundation and the chairpeople of the Learning Science departments said: "That's not learning science". Some might say: "It's not learning science, it's mathematics". Others would say: "It is not science research, it is design." From our perspective, they are caught in a dilemma: either they expand their conception of the learning sciences discipline to include such restructurations or they exclude a dramatic improvement in learning from the province of learning sciences and give it up to, for example, the mathematics department. The mathematics department of FOO might recognize this restructuration as research, but their criteria for whether this is important research is whether it addresses the set of problems currently regarded as important by professional mathematicians and not whether it addresses problems in education. This dilemma is not just confined to our imaginary country of FOO. In the contemporary world the science of structurations has no natural home. {It requires deep disciplinary knowledge, creativity in the design of representations and sensitivity to the epistemological and learning issues.} The structure of academic departments, funding organizations, etc. does not have a place for such work. We would like to see the fact that a few researchers including ourselves have found places in the University and funding system as a manifestation of the trend we noted at the beginning of the paper and believe that it would be accelerated by the development of the new branch of Learning Sciences.

It is the argument of this paper that computation-based restructurations are poised to make a significant impact on knowledge domains. The Learning Sciences is thus presented with an opportunity to study the process of restructuration and to direct it for the benefit of learners.

Ways to evaluate restructurations

From our current perspective, it is obvious that the Hindu-Arabic restructuration leads to better results in being able to handle numerical relationships than the Roman structuration. However, to the people of Foo and to Foo's evaluators and test-makers, it was not obvious. In our thought experiment, students who learned the new Hindu-Arabic system would likely not be able to pass the standardized tests developed using the old Roman system. Suppose they were asked: Which is the largest, CIX, XCI, or CXI? A student of the Hindu-Arabic system might not even understand this question, yet still be much better prepared to deal with real-world arithmetical problems. The same set of difficulties may be anticipated today. To overcome this difficulty the Learning Sciences must strive to create evaluation measures that go beyond the specifics of a representation which is a means to an end and instead devise measures for the ends themselves.

[1] The most widely used text on improving education by changing *how* children learn in a child-centered way without changing *what* they learn.

In order to study and evaluate restructurations, we have found it useful to focus on five core properties of structurations:

a) *Power properties.* By definition a restructuration of a domain must be able to do what could be done before and as in the cases we describe here preferably more as well. A new structuration can, in some respects, have the effect of broadening a discipline, in some cases by bringing what were regarded as essentially different phenomena into a common framework, and in other cases, by encompassing phenomena that could not be treated at all. An example of the former kind will be seen in our use of a formalism for multi-agent models to treat phenomena in physics and in biology. An example of the other kind is chaotic dynamics, which is famously intractable with algebra and calculus, yet is readily amenable to simulation and study with computers.

b). *Cognitive properties.* We are interested in restructurations that are more easily learned while preserving or augmenting the power of the old. Among factors that make for learnability some are traditionally classified as cognitive. For example, the sheer "complexity" of the tasks to be performed surely affects ease of learning. A subtler dimension is the fit of knowledge to be learned with pre-existing knowledge (whether it is learned or innate) as for example in diSessa's theory of p-prims (1983) or in Chomsky's theory of linguistics (1957).

c) *Affective properties.* A restructuration can make the knowledge more or less engaging, holding or simply likeable. Computational media offer especially rich opportunities to make use of this fact to increase engagement of the learner (Turkle, 1984).

d) *Social properties.* Richard Dawkins (1976) has used the concept of "meme" in analogy with gene to describe how ideas can spread in an evolutionary manner through a society, social niche or culture. Restructurations generate memes that can have varying evolutionary fitness in the social landscape. Ecology is an example of a meme that has spread quite rapidly from science into the general culture. The presence of ecology as a common meme enables some of the restructurations we present below to more easily spread thru the culture. This is an example of interactions between properties of structurations. The presence of the ecology meme is affected by and affects the affective response of individuals and creates knowledge elements which are good cognitive fits with systems-based restructurations.

e) *Diversity properties.* One way in which structurations of a discipline can differ is in their match with a diversity of learning styles and ways of thinking. How does the learnability of the new structuration differ for learners with different backgrounds and learning styles? How does a learner's or a teacher's culture, ethnicity, gender, cognitive or emotional style affect their interaction with the properties of a structuration? Howard Gardner (1993) has shown how learning is served by an appropriate match between the learner's kind of "intelligence" and the material being learned. However the situation in schools where matching means choosing among given domains strongly limit the power of this idea. The prospect of restructuring domains offers dramatically greater scope: instead of characterizing people in terms of their match with subject domains (mathematical, musical. Literary, etc) we look for restructuration of domains to match people's styles.

We now turn to giving concrete example of restructurations. We have chosen as domains core representative topics in the areas of mathematics, science and engineering. All of the examples we will give make use of computational objects as their representational infrastructure. In these examples, the computational object replaces a more traditional mathematical representation such as geometric, algebraic or calculus-based. As we will see in the examples, the computational object (aka agent) has power properties that make it attractive to scientists and has cognitive, affective, and social and diversity properties that make what we currently think of as advanced topics learnable by a much wider and younger population. Just as the advent of Hindu-Arabic numerals enabled a democratization of numerical facility so we suggest will computational agents enable a democratization of STEM knowledge, particularly so for understanding the evolution of systems over time.

Our examples are framed by a large story with two plot lines. One is about the development of science through three major restructuring phases; the other is about how these restructurations enter (or fail to enter) the learning lives of children.

We introduce the scientific restructurings by looking at a number of ways to think about a circle.

For Euclid a circle is defined by the fact that all its points are at the same distance from a certain point called its center. An aspect of this that is to be kept in mind is that the decision about whether a point is on the circle requires access to a point that is not on the circle: an ant crawling on a path could not use this definition to decide whether the path is circular.

The second view of the circle is its definition by an equation such as $x^2 + y^2 = K$. This was made possible by a major restructuration of geometry, due to Descartes, by representing geometric entities in algebraic form.

The third view will be presented anachronistically in terms of a computational object known as "the turtle" or "the Logo turtle." Think of this as an entity that has two essential "state properties." In Euclid's geometry the fundamental element is a point defined by the fact that it has position *and no other properties*. As a mathematical entity it has no color, size or shape although the geometer may represent it as a small, black dot. The turtle is much like the point except that its has position and ONE other property, called its heading. Again on a computer screen it is represented as something with shape, color and size but these are not properties of the pure mathematical turtle. A turtle in motion has two velocities: its linear velocity is the rate of change of its position; its angular velocity is the rate of change of its heading.

With these preliminaries we can state our third view of the circle. If a turtle moves with both velocities constant it will draw a circle! What is remarkable about this is that the turtle draws the circle without reference to any external entity such as Euclid's "center" or Descartes "coordinate axes." Another way of saying this is that with this definition an ant walking on the circle can know that it is a circle. Yet another is that someone with a tiny field of view can tell whether a figure is a circle by looking at all parts of it confined at each instant to the tiny field of view – and this is true no matter how tiny the field. We recall that using Euclid's definition the observer's field of view would have to be big enough to include the center as well as points of the circumference.

We used a computational object to define a way of thinking about the circle for the reason that underlies the second plot line of our story: the turtle enables us to explain the concept in a simpler and more concrete way than the one used by the pioneer of this way of thinking two centuries before the computer was invented. The pioneer was Isaac Newton and the concept is the core of what is now known as "calculus" although most contemporary students who are required to undergo school courses with this name would probably not recognize any connection. Newton's great achievement was to deduce a global property (such as being a circle or an ellipse) from local properties (such as having constant curvature or the force of gravity at each point.) This achievement gave rise rapidly to a restructuring of large areas of science. But – and one might say that this is the main theme of this paper – this restructuration could be appreciated and used only by people who had already acquired a rather complex body of prerequisite skills and knowledge, until the computer enables us to restructurate the restructuration and so make it accessible to many more people including, particularly, children considered too young to "learn calculus" in its pre-computational form.

This last assertion will be elaborated shortly, but first we introduce our fourth way of thinking about a circle. Place a large number of turtles at the same place. Give each one a random heading. Make them all move forward (i.e. in the direction of their headings) by the same amount. They will form a circle. In some ways this goes back to Euclid's definition: but it has a new slant: the circle *emerged* from the behaviors of a large number of agents. What will emerge in this case is "obvious." But we shall see how the interaction of large numbers of agents each following a very simple rule can give rise to complex, scientifically interesting and by no means obvious emergent effects.

The view of the circle as an emergent property of a large number of agents captures in a very simple form the second plot line of our story of scientific restructurings. Newton's breakthrough led to the understanding ("Newtonian mechanics") of the behavior of individual physical objects such as the earth or moon in orbit. It did not take long before scientists tried to apply mechanics to large populations of entities, especially the molecules of a gas. But it was not until the nineteenth century that the necessary mathematical methods were developed to create the systematic theory now known as statistical mechanics which led to a deep restructuring of the understanding first of gas laws and later of liquids and solids as the aggregate behavior of large numbers of molecules. Here again representing these situations as collections of computational objects (agents ...we must decide on language) allows us to make accessible to young students a level of understanding that in the past has been very difficult even for much older students.

We now turn from this outline to the actual chapters of our story. The first example is about individual (or small groups) of entities matching the original Newton restructuration, the next is about gas laws and the last shows the beginnings of an extension of the ideas to understanding solids.

Turtle Geometry and Beyond

We show in this section how the introduction of the turtle opens the possibility of far-reaching restructuring of early mathematics education. But first we want to counter in advance justified skepticism about the idea that we have an overly grandiose obsession with our turtle as a "silver bullet" or "panacea." Our response has two opposite parts. On one side we point out that what is at play here is not an idea that we invented: what we are doing is showing that the computer can make available to children the essence of one of the most important ideas of all time made by one of the scientific geniuses of all time. This goal is surely worthy of a few lifetimes of obsession. We are not being overly grandiose in expecting this idea that has deeply transformed science could have deep consequences for learning as well. On the other side although we believe that many very different innovations will come as more people join the search for restructuration, we also believe that the best way to serve this goal is not to spread ourselves thin by trying to go in too many different directions but rather to bring out the depth and variety of the one we have opened.

Example 1: The Tick model - Newtonian Physics and Beyond

In chapter two of *Changing Minds (2000),* diSessa describes the "tick model" of motion. It is a computational model of motion whereby an imaginary clock repeatedly ticks at a fixed interval and an object moves in the interval between ticks. The tick model is fundamental to a computational restructuration of kinematics. diSessa argues that the tick model is marvellously adapted to representing kinematics content. It is both very expressive as well as precise. It reveals the essential components of motion: it's repetitiveness and it's differential components accumulating over time. Indeed the tick model has become essential not only for describing motion but also for describing any system that changes over time.

Our first example uses a single turtle agent or a small number of agents to restructurate traditional kinematics. In the next two examples, we will employ large numbers of agents and the methodology of agent-based modelling. We will claim that computational agents form a new computational infrastructure that is already restructurating scientific disciplines and that careful thought is needed in order to both understand the impact of these restructurations on learning of the knowledge domains and to design restructurated curriculum that takes advantage of the properties of the restructuration. We take a brief digression to introduce agent-based modelling.

Agent-based Modeling

One powerful methodology that has emerged from complex systems theory is agent-based modeling. In contrast to more traditional mathematical modeling which is typically done with equations, agent-based modeling makes use of simple computational rules as the fundamental modeling elements. The equational modeling game is to observe a phenomenon and try to fashion an equation that fits the observed data. A classic example is the Lotka-Volterra equations used to model the change in predator and prey population levels over time. In equational modeling, the core elements of the model are variables that refer to population-level descriptors. In the Lotka-Volterra equations, the core elements are L, the population level of the lynx predators, H, the population level of the hare prey, and K, the interaction constant that describes the average predation. To understand the state of the system at a future time T, you solve the equations for that time. In contrast, in the agent-based modeling game, the core elements are computational objects or “agents” that represent individual lynxes or hares. Each of these agents has state variables that describe its particular state, such as age, energy level, hunger, etc. The behavior of the agents is determined by the computational rules that tell each agent what to do at each “tick” of a clock. The rules are framed from the agent’s point of view. For example, if the agent is a lynx, the rules might say: move a step in the direction you are headed, reduce your energy variable by a fixed amount, look for prey in the vicinity, if found where you are, try to eat it, if not turn to face closest prey you can see… To determine the state of the system at future time T, you run the system for T clock ticks. As rules typically have stochastic components, one would typically run the system many times to capture the space of possible trajectories for the system.

Increasingly scientists are making use of agent-based models as both explanatory and predictive tools. Across a wide variety of domains in the natural and social sciences, scientists are framing their theories in terms of agent-based models. In the natural sciences, agent-based models have several advantages over equational approaches. Chief among these are a) the epistemological match – rules for individual predators or molecules are closer to our intuitive notions of these “objects” as distinct individuals rather than as aggregate populations. b) the greater adjustability – equational representations tend to be brittle, that is, for some small change in environmental conditions, the algebraic forms themselves do not typically change only a little. An entire new formalism may be required to capture the new situation. In our lynx-hare example, if we discover that when hares become too populated, they start to attack each other, the changed needed to the LK equations is not straightforward. In contrast, in the agent-based approach, it is a trivial matter to give the hares an extra rule to that effect. c) Visualization – related to the epistemological match is the greater realism afforded by visualization of individual lynx and hare and their dynamic behaviors rather than just dynamic graphs of their populations.

All three of these advantages are magnified in the educational context. Students can reason about and visualize individual animals in an ecology far better than they can population levels. They can draw on their own body and sensory experience to assess and/or design sensible rules for the behavior of individuals. They can therefore make much greater sense and meaning from the agent-based representations. Furthermore, the extensibility/adjustability of the models enables students to engage in real inquiry by asking what-if questions of the models and adjusting rules in order to get answers to their questions. While this is common practice for scientists, it is not so for students. The alternate representation, in effect, enabling them to think more like scientists (Wilensky & Reisman, 2006).

In the educational context, there is one more advantage that is greatest of all: the greater ease of mastering the representations themselves. Learning to master the LK equations requires the prior mastering of an extensive algebraic and calculus-based infrastructure that is out of reach for large numbers of our students. These students are therefore shut out of the scientific exploration of most worldly phenomena that change over time. And even those students who do eventually master algebra and calculus do so late in their student "careers". Agent-based representations, in contrast, require significantly less effort to master. Research we have conducted shows that typical middle school students can profitably employ these representations with only a small amount of prior instruction. Widespread adoption of agent-based representations can therefore lead to tremendous democratization of scientific knowledge.

Our next two examples employ agent-based modelling to restructurate relatively advanced science content making it more learnable as well as accessible to young learners

Example 2: GasLab - Statistical Mechanics and beyond

The GasLab package is a suite of NetLogo models of kinetic molecular theory. In the basic model, gas molecules are represented by turtle agents that bounce off each other and off their enclosing container like billiard balls with elastic collisions. Using GasLab, many groups of students have conducted experiments with the Gas-in-a-Box models. They also revised and extended the model, creating the nucleus of the set of models, which comprise GasLab. The set of extensions of the original Gas-in-a-Box model is impressive in its scope and depth of conceptual analysis. Among the many extensions students tried were: heating and cooling the gas, introducing gravity into the model (and a very tall box) and observing atmospheric pressure and density, modeling the diffusion of two gases, allowing the top to be porous and seeing evaporation, relaxing elasticity constraints and looking for phase transitions, introducing vibrations into the container and measuring sound density waves, and allowing heat to escape from the box into the surrounding container. They also reinvented various well-known thought experiments of statistical mechanics related to Maxwell's demon and second law considerations[2]. Over the course of several weeks, these high school students "covered" much of the territory of collegiate statistical mechanics and thermal physics and their understanding of it was deeply grounded in both a) their intuitive understandings gained from their concrete experience with the models and b) the relations amongst the fundamental concepts.

[2] As one example of these reinvented thought experiments, they constructed a model of a divided box with a small opening in the divider in which a propeller is embedded. They measured the work done on the propeller by the particles hitting it and the propeller's consequent motion. A version of their model is downloadable from http://ccl.northwestern.edu/netlogo/models/GasLabSecondLaw.

GasLab provides learners with a set of tools for exploring the behavior of an ensemble of micro- elements. Through running, extending and creating GasLab models, learners were able to develop strong intuitions about the behavior of the gas at the macro level (as an ensemble gas entity) and its connections to the micro level (the individual gas molecule). In a typical physics classroom, learners usually address these levels at different times. When attending to the micro level, the focus is, typically, on the exact calculation of the trajectories of two colliding particles. When attending to the macro level, the focus is on "summary statistics" such as pressure, temperature, and energy. Yet, it is in the connection between these two primary levels of description that the explanatory power resides.

Two major factors enable students using GasLab to make the connection between these levels -- the replacement of symbolic computation with simulated experimentation and the replacement of "black-box" summary statistics with learner-constructed summary statistics. The traditional secondary physics curriculum segregates the micro- and macro- levels of description because the mathematics required to meaningfully connect them is thought to be out of reach of high school students. In the GasLab modeling toolkit, the formal mathematical techniques can be replaced with concrete experimentation with simulated objects. This experimentation enables learners to get immediate feedback on their theories and conjectures. In traditional curriculum learners are typically handed concepts such as pressure as "received" physics knowledge. The concept (and its associated defining formula) is, thus, for the learner, a "device" built by an expert, which the learner cannot inspect nor question. Learners do not come to see that this concept represents a summary statistic – a way of averaging or aggregating the behavior of many individual particles. Most fundamentally, the learner has no access to the design space of possibilities from which this particular summary statistic was selected. In the GasLab context, learners must construct their own summary statistics. As a result, the traditional pressure measure is seen to be one way of summarizing the effect of the gas molecules on the box, one way to build a gauge. The activity of designing a pressure measure is an activity of doing physics, not absorbing an expert's "dead" physics.

Example 3: MaterialSim - Materials and Beyond

Materials science and engineering has grown considerably from its roots in experimental metallurgy. It is now a fundamental part of engineering education. T*raditional* methods for investigating properties of materials reflect the tools that were available in the nineteen-fifties: mathematical abstractions, geometrical modeling, approximations, and empirical data. These tools have inherent limitations both in their "power properties" for scientists but even more so in their "learnability properties" for students.

In the past two decades, massive computing power has made a new and promising restructuration possible: computer simulation of individual molecules of the materials. Practicing material scientists have rapidly adopted this new approach. However, it has not as of yet migrated to the teaching of materials science, which still relies on the traditional methods.

As a specific example, let us consider the phenomenon of "grain growth". Most materials are composed of microscopic "crystals". A crystal is just an orderly arrangement of atoms, a regular tri-dimensional grid in which each site is occupied by an atom. In Materials Science, scientists use the term "grain" to refer to such an arrangement. The notion of grain is fundamental to Materials Science and Materials Engineering.

Among other properties, grain size determines how much a material will deform before breaking apart, which is one of the most important issues in engineering design. For example, a car built with steel with a wrong grain size could significantly increase the risk of serious injury for the passengers. But grain size can change, too – high temperature is the main driving force. This phenomenon, known as *grain growth*, is exhaustively studied in Materials Science: small grains disappear while bigger ones grow (the overall volume is maintained). Airplanes turbines, for instance, can reach very high temperatures in flight – an incorrectly designed material could undergo "grain growth" and simply break apart. The following photographs (magnified 850x) show typical results.

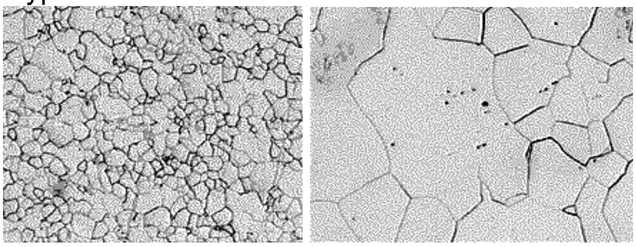


Figure 1: Metallic sample before and after grain growth (Blikstein & Tschiptschin, 1999)

Burke (Burke, 1949) was one of the first to introduce a law to calculate grain growth and proposed that the growth rate would be inversely proportional to the average curvature radius. Burke's law states that large grains (lower curvature radius) grow, while small grains (high curvature) shrink. The mathematical formulation of Burke's law also reveals that, as grains grow, the growth rate decreases. A system composed of numerous small grains (see Figure 1, left) would have a very fast growth rate, while a system with just a few grains (see Figure 1, right) would change very slowly. In the beginning of the century, metallurgists believed grains to have a "maximum size" for a given temperature – but that was only due to the lack of tools to detect the very slow growth rate at the end of the process. However, even Burke's description had its limitations. In order to make the math feasible, for example, Burke was led to consider grains as spheres with just one parameter to describe their size (the radius). For most practical engineering purposes, this approximation yields acceptable results – however, its practical efficacy does not necessarily mean that this approach is the best way to understand the phenomenon, nor to build on it to understand other phenomena in materials science.

In the 1980s Anderson, Srolovitz et al. (Anderson, Srolovitz, Grest, & Sahni, 1984a, 1984b) proposed the now widely used theory for computer modeling of grain growth using an agent-based approach. This kind of simulation not only made predictions faster and more accurate, but also allowed for a completely new range of applications. Researchers were no longer constrained by approximations or general equations, but could make use of more precise mechanisms and realistic geometries.

Anderson et al. state that the classic rule-of-thumb for grain growth ("large grains grow, small grains shrink") is not always valid, and that randomness plays an important role. Given the microscopic dimensions and small time scale of the phenomenon, practically the only way to visualize this new finding is through computer simulation. As a result of these "power properties", this approach became widely adopted for the use of professionals. But, since at first glance, it would seem that since the situations for which it is a superior approach are not the simple cases, but the advanced ones used by professionals, that there was no reason to change instruction for novices in the field.

However, an agent-based approach to grain growth has *learnability* properties that make it particularly suited for novice learners. The agent-based simulation of grain growth offers a different perspective. Its principle is the thermodynamics of atomic interactions – one of the *extensible*, *transferable*, *anchor* models. Consider the learning environment, MaterialSim (Blikstein & Wilensky, 2004a), which employs the agent-based approach to teach Materials Science. MaterialSim is a set of exploratory models built within the NetLogo (Wilensky, 1999) environment. There are models for investigating crystallization, solidification, casting, grain growth and annealing.

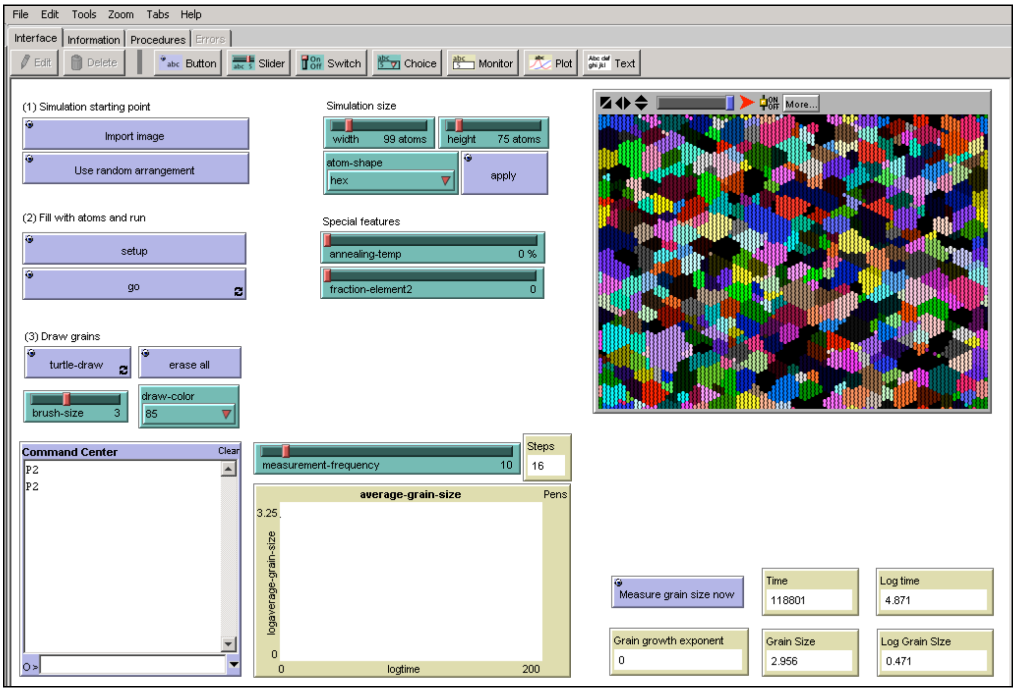


Figure 2. MaterialSim's grain growth model (Blikstein & Wilensky, 2004b)

MaterialSim represents a material as a hexagonal 2D matrix, in which each site corresponds to an atom and contains a numerical value representing its crystallographic orientation. Contiguous regions (containing the same orientation) represent the grains. The grain boundaries are fictitious surfaces that separate volumes with different orientations. MaterialSim's grain growth algorithm is described below:

- Each element (or agent) of the matrix has its free energy (G_i) calculated based on its present crystallographic orientation (Q_i, represented by an integer) and its neighborhood (the more neighbors of differing orientation, the higher its free energy). Figure 3 (left side) shows the central agent with four different neighbors; hence the value of its *initial free energy* (G_f) is 4.

- One new random crystallographic orientation is chosen for that agent (Q_f), among the orientations of its neighbors. In this case, as observable in Figure 3, the current value of the central agent is "2", and the new transition value is "1".

- The agent's free energy is calculated again (G_f), with the new proposed crystallographic orientation (Q_f=1). **Error! Reference source not found.** Figure 3 (right side) shows that there are only two different neighbors in the new situation, thus the *final free energy* (G_f) decreases to **2**.

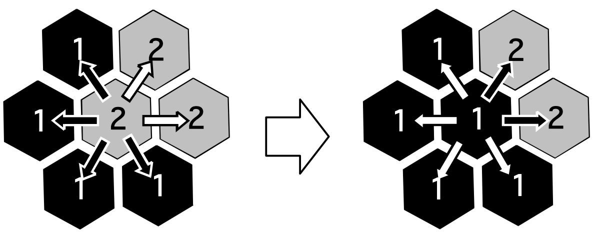


Figure 3: Initial and final free-energy calculations. Black and white arrows denote different or equal neighbors.

- The two states are compared. The value that minimizes the free energy is chosen. In this case, $G_i=4$ and $G_f=2$, so the latter value is lower and constitutes a state of greater stability. Thus, the proposed change in orientation is accepted.

From this basic model, one can understand what is going on in the material at the micro-level. Instead of having to use rules of thumb to predict what will happen to the grains, we can use this formal model to visualize and reason about the evolution of the material. For most initial conditions, we will indeed see a rule of thumb such as “large grains swallow others” obtain. But, we will also see how this process develops, how it emerges from the micro-level “decisions” of the molecules. And we will see that under some conditions the traditional rule of thumb will be violated.

But the greater value of the agent-based approach lies in more than understanding this particular phenomenon. Once the basic model is set up, it is easy to explore a large set of configurations and to understand possible trajectories of the system. And because the representation system is composed of simple modifiable micro-rules rather than aggregate level equations, it is easy to modify them to explore a host of other phenomena. In our implementations of MaterialSim, we have seen students adapt the basic model to explore a diversity of materials science phenomena such as recrystallization, diffusion, interfacial energy, nucleation, solidification, and phase transformations.

The agent-based restructuration of materials science enables students to reason about materials from the atom on up. Whereas traditionally, they employ heuristics and formulae given to them by authority, they are now able to author their own heuristics and formulae, derived from their modeling experience. Just as the restructuration of numerals enabled ordinary folks to do multiplication and division for themselves, the agent-based restructuration of materials science enables learners to set up experiments and author new models for themselves.

Conclusion

We have briefly laid out the theory of restructurations and called for careful consideration of computer-based, and in particular, agent-based restructurations of science content and instruction. We have presented three examples of such restructurations. We note that such restructurating is not confined to mathematics and natural science. Indeed, we suspect that agent-based restructurations of social science may give even grater leverage. We note that our own fields of education and learning sciences can be restructurated using agent-based approaches. To understand educational reform and phenomena such as curricular adoption, homework collaborations and the effects of educational policies, agent-based modeling can be a powerful tool. It enables us to study these phenomena as emergent from the interactions of the individuals rather than through properties of the aggregate populations.

Acknowledgements

We wish to thank our colleagues with whom we have had many productive and exhilarating discussions on the topic of this paper. In particular we'd like to thank Andy diSessa, Richard Noss and Bruce Sherin who co-developed with us a proposal for an NSF Science of Learning Center in which these ideas were discussed and refined (Wilensky et al, 2006). We have benefited also from numerous conversations with Alan Kay and Sherry Turkle on the properties of restructurations.

We also wish to thank students and former students who have carried out further restructurations: Dor Abrahamson, Paulo Blikstein, Mike Eisenberg, Sharona Levy, Mitchel Resnick, Pratim Sengupta, Michael Novak, Spiro Maroulis. and Michelle Wilkerson.

References

Abraham, R. & Marsden, J. (1978). *Foundations of mechanics*. Reading, MA: Benjamin/Cummings.

Abrahamson, D. & Wilensky, U. (2004). ProbLab: A computer-supported unit in probability and statistics. In M. J. Hoines & A. B. Fuglestad (Eds.), *Proceedings of the 10th International Congress on Mathematical Education*, Bergen, Norway, July 14-18, 2004.

Allen-Conn, B.J. & Rose, K. (2003). *Powerful ideas in the classroom*. Glendale, CA: Viewpoints Research Institute.

Axelrod, R. M. (1997). *The complexity of cooperation: Agent-based models of competition and collaboration*. Princeton, NJ: Princeton University Press.

Blikstein, P., & Wilensky, U. (2009). An atom is known by the company it keeps: A constructionist learning environment for materials science using multi-agent simulation. *International Journal of Computers for Mathematical Learning, 14*(1), 81-119.

Blikstein, P. & Wilensky, U. (2005). Less is More: Agent-Based Simulation as a Powerful Learning Tool in Materials Science. *Proceedings of the IV International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005), Utrecht, Holland.*

diSessa, A.A. (2000). *Changing minds: Computers, learning, and literacy*. Cambridge: MIT Press.

Epstein, J., & Axtell, R. (1996). *Growing artificial societies: Social science from the bottom up*. Washington, DC: Brookings Institution Press.

Kay, A. & Goldberg, A. (1977). Personal Dynamic Media. *IEEE Computer,* 3, 31-42.

Levy, S. T. & Wilensky, U. (2004). Connected Chemistry - A study of secondary students using agent-based models to learn chemistry. In U. Wilensky (Chair) and S. Papert (Discussant), *Networking and complexifying the science classroom: Students simulating and making sense of complex systems using the HubNet networked architecture*, The annual meeting of the American Educational Research Association, San Diego, CA, April 12-16, 2004.

Maroulis, S. & Wilensky, U. (2005). Leave No Turtle Behind: An Agent-Based Simulation of School Choice Dynamics. *The annual meeting of the American Educational Research Association*, Montreal, Canada, April 11 - 15, 2005.

McCloskey, M. (1983). Naive theories of motion, in D. Gentner & A. Stevens (Eds.), *Mental Models* (pp. 299-324). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Papert, S. (1980). *Mindstorms*. NY: Basic Books.

Papert, S. (1996). An Exploration in the Space of Mathematics Educations. *International Journal of Computers for Mathematical Learning.* **1**(1): 138-142.

Resnick, M. (1994). *Turtles, termites and traffic jams: Explorations in massively parallel microworlds*. Cambridge, MA: MIT Press.

Wilensky, U. (1999a). GasLab—An extensible modeling toolkit for exploring micro- and macro-views of gases. In N. Roberts, W. Feurzeig, & B. Hunter (Eds.), *Computer Modeling and Simulation in Science Education* (pp. 151–178) . Berlin: Springer-Verlag.

Wilensky, U. (1999b). *NetLogo*. [Computer software]. Evanston, IL: Northwestern University, Center for Connected Learning and Computer-Based Modeling. http://ccl.northwestern.edu/netlogo.

Wilensky, U. (2001). *Modeling nature's emergent patterns with multi-agent languages*. Paper presented at Eurologo 2001, Linz, Austria.

Wilensky, U., Hazzard, E., & Froemke, R. (1999). GasLab—An extensible modeling toolkit for exploring statistical mechanics. Paper presented at the The Seventh European Logo Conference, Sofia, Bulgaria.

Wilensky, U., & Papert, S. (2006). Restructurations: Reformulations of knowledge disciplines through new representational forms (Manuscript in preparation).

Wilensky, U., Papert, S., Sherin, B., diSessa, A. A., Kay, A., & Turkle, S. (2006). *Center for Learning and Computation-Based Knowledge (CLiCK). Proposal to the National Science Foundation - Science of Learning Center*.

Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep or a firefly: Learning biology through constructing and testing computational theories. *Cognition & Instruction*. *24*(2), 171-209.

Wilensky, U. & Resnick, M. (1999). Thinking in levels: A dynamic systems perspective to making sense of the world. *Journal of Science Education and Technology*, *8*(1), 3–19.

Wolfram, S. (2002). *A new kind of science.* Champaign, IL: Wolfram Media.

Restructuring Change, Interpreting Changes: The *DeltaTick* Modeling and Analysis Toolkit

Michelle Hoda Wilkerson-Jerde, *m-wilkerson@northwestern.edu*
Learning Sciences Program and Center for Connected Learning, Northwestern University

Uri Wilensky, *uri@northwestern.edu*
Learning Sciences, Comp. Sci., and Center for Connected Learning, Northwestern University

Abstract

Understanding how and why systems change over time is a powerful way to make sense of our world. By modeling those systems, learners have the opportunity to consider how their own actions influence that world, and to make predictions and recommendations for the future. But often, the notion of change is as complex as it is powerful – populations, global temperatures, and economic trends all represent *multiple* events and actors, but are measured in terms of only a few quantities. In this paper, we discuss the motivation and design of *DeltaTick*, an extension to the NetLogo (Wilensky, 1999) agent-based modeling environment that allows learners to easily construct and analyze models of complex quantitative change. To do so, they define models using *agent behavior*-based units, rather than the *rate*-based units typical of equations or systems dynamics models. They can then explore, compare, and annotate model results to investigate how their behavioural models relate to typical equation-based representations.

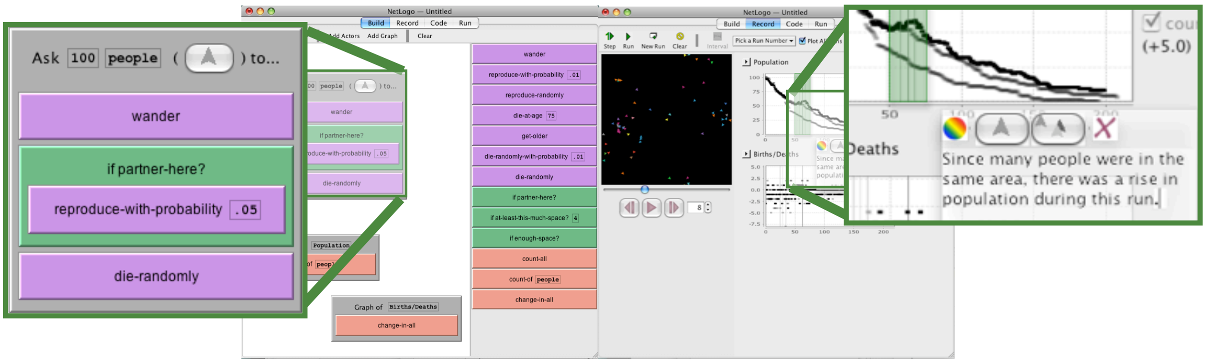


Figure 1. Constructing (left) and analyzing (right) models using DeltaTick

Our design is rooted in constructionism (Papert, 1980), and integrates work on complex systems education (Wilensky & Resnick, 1999), low-threshold agent-based modeling (Kahn, 2007; Repenning & Ambach, 1997), representational infrastructure shift (diSessa, 2001; Kaput et al, 2002), and intuitive calculus (Kaput, 1994; Nemirovsky et al, 1993; Stroup, 2002). It leverages what Wilensky and Papert (2006; In Prep) refer to as *restructuration*: re-encoding of disciplinary content with a new representational technology to emphasize different properties of that content. We argue that by constructing and interacting with agent-based models, learners can recognize the *relevance* of ideas of change and variation to learners' own experiences as actors and observers in their world, and the *learnability* of some core concepts of change and variation.

Keywords

mathematics; mathematical modeling; restructuration; agent-based modeling; scientific literacy

Restructuring Change

We want to give learners a way to critically think about how they influence and are influenced by large-scale, systemic changes in their world. *DeltaTick* is a simple, extensible construction and analysis toolkit to support this goal by leveraging recent findings regarding how people think and learn about quantitative change. In this paper, we describe the motivation for the *DeltaTick* environment, discuss some of its design features, and briefly review interviews with learners who used a preliminary version to construct and explore alternative models of population growth. We argue that by providing learners with tools and activities that allow them to express notions of rate and accumulation as the outcome of specific individual behaviors as they occur over time, learners can model and explore quantitative change in a way that (a) emphasizes its relevance to their own lives by leveraging their own experiences of actions and change in the world, and that also (b) provides a novel access point to many of the ideas of mathematical change as they exist in more typical calculus and differential equations-based representations (namely, notions of derivative, integral, and the reversibility of the two).

The theoretical and design contributions of this work are threefold. First, we are exploring a design space that provides learners a low-threshold entry point to building flexible, personally relevant scientific and mathematical models while still having the opportunity (and, as we argue in this paper, encouraging) more sophisticated model refinement. Second, we are leveraging and contributing to existing work on learners' experiences and understanding of quantitative change, but in the specific context of change in complex systems, where multiple interactions and events are embedded in only one or a few measured trends. Finally, we are exploring the role of tools for the analysis of student artifacts in constructionist environments – such that those tools and learners' own artifacts serve as a bridge to typical representations of disciplinary content. In terms of practical contributions, we are working toward providing learners with a viable, intellectually honest alternative to symbolic calculus for modeling mathematical change, while at the same time providing a potential bridge to more typical calculus-based concepts. Our goal is to present the mathematics of change as a relevant, accessible, and empowering tool that can help learners understand and predict their world.

Motivation

Examining rates of change over time and their accumulations have become some of the most ubiquitous practices in not only the natural and social sciences (AAAS, 1991), but also for navigating modern society (Roschelle et al, 2000; OECD, 2006). Often, however, the quantitative trends used to explore economic, environmental, or social phenomena reflect large-scale, systemic processes that involve and affect a number of actors and events. In this sense, it is not just understanding quantitative change, but also understanding how that change reflects the events and interactions of a given system that helps us to make sense of the world and our role as citizens within it.

Shifting representational infrastructures - and specifically, computational tools that represent and simulate processes over time - reflect a powerful way of exploring, thinking about, and simulating change over time - and potentially, for allowing more people to do so (Papert, 1980; diSessa, 2001; Kaput et al, 2002). Agent-based modeling (ABM; Langton, 1997; Wilensky & Resnick, 1999) is one example of a computational representation appropriate for modeling complex systems such as those described above. This technology models a phenomenon by encoding the behaviors and interactions of individual agents or elements of a system (for instance, the rules that govern motion and collision of particles in a gas), and then simulating that system by having a collection of those agents execute those behaviors over time (for instance, to illustrate how that gas exerts pressure on a container; Wilensky, 2003). It has fundamentally changed *how* scientific content is represented and explored, as well as *who* can author and interact with that content (Blikstein & Wilensky, 2009; Levy et al, 2004; Sengupta & Wilensky, 2008).

But while building and interacting with agent-based models can help learners develop a more deep and generative understanding of traditionally advanced content, less is known about how they link this understanding with more conventional representations of those concepts - namely, algebraic and calculus-based equations. This project explores how ABM can serve as an access point to the mathematical aspects of complex phenomena and the ways they connect with the mechanisms and patterns those mathematics represent. To do so, we leverage Wilensky and Papert's (2006; In Preparation) notion of a *restructuration*: a re-encoding of existing disciplinary knowledge using a new representational technology that emphasizes different aspects and properties of that knowledge. In other words, we are exploring how agent-based modelling can be used to provide learners with a new language to "speak" and practice quantitative modeling.

In the following sections of this paper, we describe in more detail the notion of *restructuration*, and make the case for how agent-based modeling can provide learners with more access points to not only specific scientific content, but also to the mathematical representations typically used to present that content. Next, we describe a set of computational tools to provide learners the opportunity to build and explore agent-based models with explicit focus on how those models represent mathematical change over time; along with a short description of the sort of activities that would give learners the opportunities to use these tools meaningfully and constructively. Finally, we discuss some of the specific design features of this environment in the context of preliminary interviews with learners using earlier versions of these tools. We argue that these findings suggest that constructing and analyzing agent-based models with specific attention to ideas of change and variation in systems helps learners to understand the *relevance* of ideas of change and variation in their own lives, as well as makes many difficult concepts in change and variation (such as the reversibility of rate and accumulation) more *learnable* for learners. We conclude with a brief discussion of future work and implications.

The Computational Restructuration of Mathematical Modeling

We base our motivation for the design of DeltaTick within the framework of *restructuration theory* (Wilensky & Papert, 2006; In Prep.) – that different technologies can encode the same disciplinary content, but in ways that emphasize very different aspects and properties of that content. While some *structurations* or encodings of knowledge – for example, using mathematical or agent-based representational systems – are more or less appropriate for some goals or make certain content more accessible and usable, we contend that each can also complement and inform and understanding of the other. Below, we use a figure to illustrate how agent-based modeling (and more generally, computational behavior-based simulation) can be viewed as a restructuration of the ideas of change and variation (boxes 1 and 2), and how it can inform more typical rate-based representations by providing an opportunity to *coordinate* the results of each through the plots or numerical results they generate (boxes 2 and 3). Although the figure is informed by work on mathematical modeling (Niss, et al, 2006) in the sense that a "real situation" is distilled and then formalized into some symbolic notation, we heavily adapt it here to reflect that those situations can be differently conceptualized, that different conceptualizations can be more or less commensurate with a given symbolic encoding, and that symbolic encodings can be mathematical or computational. We are also careful to note that while we are highlighting the connections that are emphasized through the activity of modeling, this is not a clean process - connections can be made between or within any world, depending on the similarities recognized and actions taken by the modeler (Pozzi et al, 1996; Noss et al, 1997).

The box to the left represents a "real world" representation or experience of some dynamic system - for instance, trends in unemployment as a student may experience them as he reads a newspaper article, or searches for a summer job. Those experiences and understandings that are viewed as the key elements, events, trends, or patterns for a particular phenomenon of interest (in this case, unemployment) can be considered together as a “situation model”. In the case of unemployment, an individual may think of his own and his friends' experiences in the workplace; but he may also consider a recent history of rising unemployment, or of national or international trends in consumer spending. These different ways of conceptualizing the causes and effects of a changing system are more or less appropriately represented by different symbol systems - agent-based modeling, for instance, is more appropriate for encoding individual experiences and interactions; while a differential equation or system dynamics model is more appropriate for considering larger-scale patterns and historic trends. Each restructuration, however, can generate results that can be compared and coordinated with one another – so that encodings in one structuration (the specific circumstances that lead to an individual getting or losing a job) can be compared to those in another (increases and decreases in employment levels), and the relationships between them interrogated (as more people are hired, they are able to spend money, which in turn allows more companies to hire more employees).

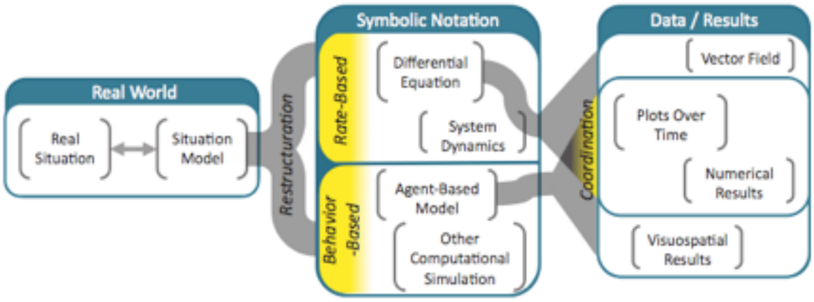


Figure 2. Restructurations are alternate encodings of the same content, with each encoding emphasizing different aspects of that content.

As a result of this shift, agent-based modeling encodes and reflects quantitative change in a way that includes a clear link to specific real-world behaviors that change can represent, as well as emphasizes notions of randomness, sensitivity to local conditions, non-uniform distributions, and other powerful ideas characteristic of *systems* that are not dealt with in traditional calculus. It also provides learners an easy to way to manipulate that encoding in ways they find interesting. On the other hand, this encoding de-emphasizes many of the powerful aspects of typical calculus-based methods, such as the ability to optimize, quickly apply solutions to new and different contexts or scenarios, or quickly compute specific solutions. We argue, and provide evidence below, that it is the transition between different structurations – including the practice of building models in each in order to explore and resolve conflicts between them – which is where a lot of learning can happen around the mathematics of change. In this sense, the plots and numerical results produced by each serve as a *bridging tool* (Abrahamson & Wilensky, 2007) for access to and from typically advanced mathematical and computational concepts.

The DeltaTick Modeling Toolkit

The main goal of our project is to provide learners an easy way to *construct* models of changing real-world systems, and then to *analyze* those models with specific attention to one or a few quantities that are typically used to represent that change.

Constructing Models

To build a model, learners begin by defining one or more types of *actors*, a collection of homogeneous entities that all behave similarly. A window on the construction screen represents each actor type. Learners can then add to those actor windows one or more pre-specified *behaviors* that each actor of that type will execute during each unit of simulated time or "tick". Behaviors can also be placed inside of *conditions*, which limit the conditions under which each agent performes that behavior happen. Finally, learners can add one or more *graphs*, also represented by a window, to the screen and add one or more quantities of interest that they wish the graph to feature. Finally, users have the option to move to an "advanced" version of the model by clicking on the "Code" tab that allows them to view and modify the text-based NetLogo code that underlies the visual representation – which allows them much more flexibility and generativity than the visual language alone. In Figure 3 below, a user has constructed a model that will start with 100 "people" agents. These agents will each wander around the world, and during any time unit that they encounter another agent nearby (*if partner-here?*), they will have a 5% chance to reproduce. They also each have a 1% chance of dying during each time unit. The model includes two graphs: one of the total population of agents (*Population*), and one of the number of agents added (born) or subtracted (died) from the system (*Births/Deaths*).

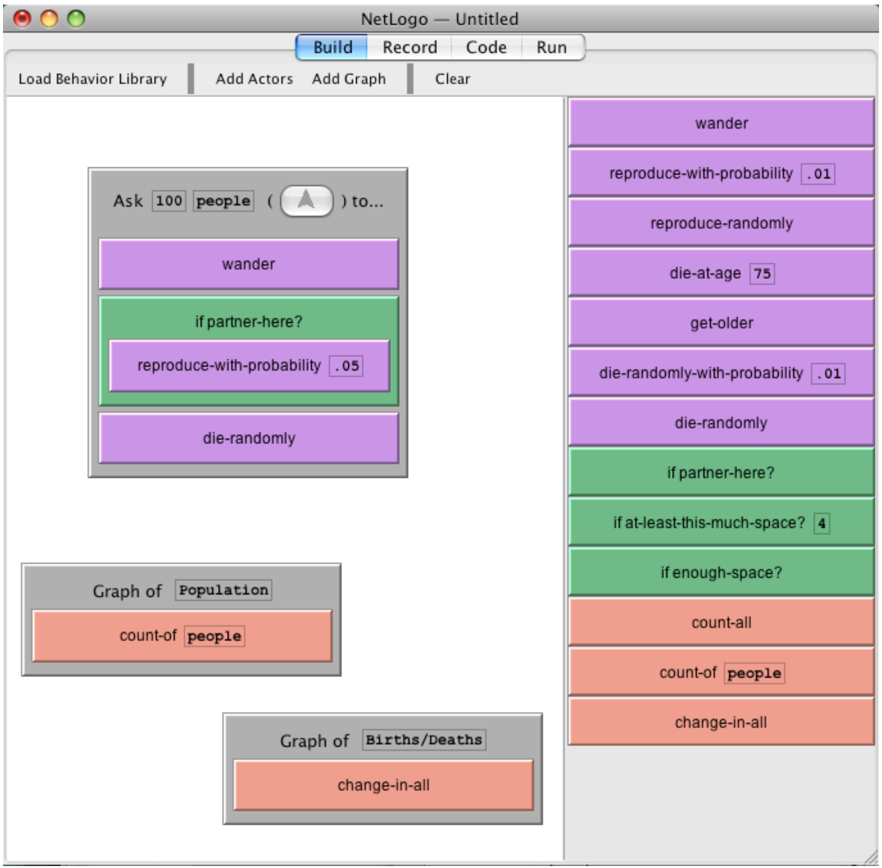


Figure 3. The DeltaTick construction interface. Blocks are designed such that different combinations correspond to markedly different resulting mathematical patterns.

The behavior blocks that appear for learners in this environment are loaded into the environment as sets of "behavior libraries". Above, a "Population Growth Library" is featured. This library includes individual-level instructions that can affect population growth patters: such as *reproduce-with-probability*, *die-at-age*, or *wander*. It also includes some conditions under which these behaviors might occur for an individual and that might also affect the population growth trajectory: such as *if partner-here?*, or *if enough-space?* (which both affect population growth trends in different ways depending on the density of agents in the world). These libraries are written as xml files that can be imported into the DeltaTick environment, so they are authorable and modifiable. This environment is inspired by Kahn's microbehaviors (2007) and Repenning's Behavior Composer (2000), in that behaviors are encapsulated and portable across agents, and that new behaviors can be written and added to the library. However, the behaviors in a given library are be designed for specific disciplinary explorations and to relate to specific potential mathematical patterns, and in that sense are more specific than the behaviors featured in behavior composer, and larger-grained than microbehaviors. This is intended to preserve a more direct relationship between the addition or removal of each behavior block and changes in the resulting mathematical patterns generated by the model, as well as to provide a considerably "low threshold" access point to model construction.

Analyzing Models

After building a model, learners have the opportunity to analyze it using the *HotLink Replay* tool, which includes a visualization of the model and the resulting graphs. These two representations are dynamically linked, so that learners can click on an area of a graph and see its corresponding point in time in the simulation, or play the simulation over time as a cursor indicates the corresponding area on the plot. Learners can also highlight any intervals on a plot, and annotate that interval. In addition to plots and visualizations, the environment also calculates a user-defined piecewise linear approximation of change on any interval of a featured graph. Figure 4 below features three consecutive runs of the model illustrated in Figure 3. The user can switch back and forth between visualizations of each different run of the model; the plot for the specific run that they are visualizing is black and the rest are grey. Below, an interesting feature of the current graph – a point during which the population rose despite a general downward trend – is highlighted in green, and a short annotation is attached to the highlight. The user has also clicked on this interesting point on the graph, so that the visualization itself displays what was going on at that time during the model's execution. Since the model was constructed such that people reproduce only if they find a partner nearby, this plot shows an increase in population while there are clusters of many individual agents together, emphasizing the relationship between specific model rules and the trends that can result from those rules.

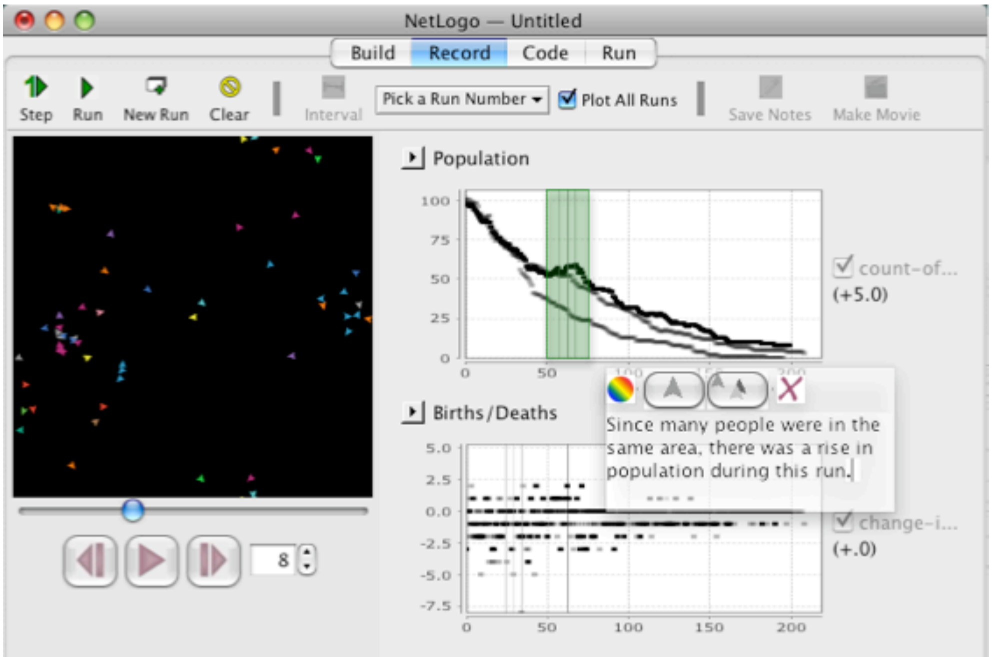


Figure 4. The HotLink Replay interface. Learners can replay, annotate, and compare different model runs.

HotLink Replay is inspired by environments that have enabled learners to develop more robust understandings of rate and accumulation by providing them a means to *control* a phenomenon that produces change (Kaput, 1994; Wilhelm & Confrey, 2003); interact with *plots* of change and rate of change over time (Confrey et al 1997); and make linkages between *intervals* and *shapes* of plots and the events they represent (Yerushalmy, 1997).

DeltaTick Activities

It is not within a tool, but in a student's use of, interaction with, and discourse around tools and activities where learning happens. As such, we argue the our design provides learners with the opportunity to engage in the activities and ways of thinking that can help them understand change in their world, view it as relevant to their own lives, and learn the powerful underlying concepts that are so ubiquitous in the natural and social sciences, as well as everyday life.

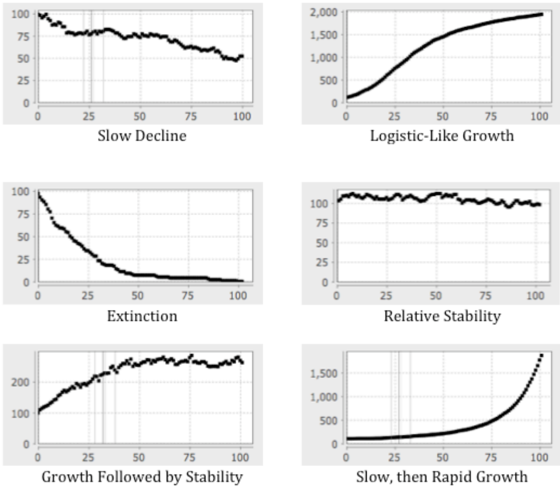


Figure 5. Graph-Matching activities encourage students to consider how different behaviors interact to produce a trend.

Activities we have found to be particularly productive for learners as they interact with this environment are to describe how patterns that do not emerge as a result of conventional mathematical notation can exist in an agent-based model and what those patterns mean (for instance, why the number of births in a population can fluctuate upward and downward even while the overall pattern of growth appears exponential); construct a specific agent-based model that they believe will create graphs that match graphs that we or their peers provide (akin to "my graph rules" activities; Wilensky & Abrahamson, 2006); research topics of interest within the domain of population trends using public scientific data on the behaviors and trends that characterize populations and use that data to create an agent-based model; find mathematical functions that approximate the quantitative trends produced by their own models; and hypothesize how different behaviors in their model correspond (or do not correspond) to different elements of the mathematical models conventionally used to represent population growth. As our collection of behavior libraries grow, different activities may emerge for different domains.

Student Interviews

To explore whether an approach such as the one above was feasible, as well as to explore our hypothesis that agent-based modelling can provide learners a new and productive means to engage with and think about mathematical content, an early, text-based version of the construction tools described above were introduced to 10 U. S. high school learners (ages 15-17) during semi-structured clinical interviews in Summer 2009. In this section we will briefly describe some examples of how learners (a) recognized agent-based modeling a way to explore specific, personally-relevant questions about population growth, and (b) connected mathematical notions of rate, accumulation, and the relationship between the two to model behavior.

Relevance: Meaningful Modeling and Extension

One of the questions we were most interested in was whether building models using a set of prespecified behavior-based units allowed learners to recognize the flexibility of this modeling language and the applicability of notions of change and variation to real-world systems. During our interviews, we encouraged learners to modify a simple, exponential model of population growth we initially provided them (in which agents each simply had a 1% chance of reproducing for each iteration of time) in any way they chose. When given this opportunity, 7 out of 10 learners added behavior parameters or behavior sequences in a way that they explicitly related to real-world behaviors, or included real-world constraints: for instance, learners explicitly mentioned issues of life expectancy (*Interview 7*) and the heterogeneity of life expectancy (*Interview 8*), family planning (*Interview 11*), and the role of age in partner selection (*Interview 3*), or the fact that many factors interact to produce patterns in a "nation or city" (*Interviews 8 and 9*). Many of these factors are ones that are difficult to include in conventional mathematical models.

In addition to constructing models that they find relevant and applicable for thinking about the world, we are also interested in making it easier for learners to think about extensions to their model, rather than only using what is available in the pre-specified library. In our interviews, 4 out of 10 learners explicitly suggested new behaviors they wished to add the their models that were not available, and 3 of those learners actually wrote NetLogo code with the to create new behaviors they could add to their models (no learners had prior experience with NetLogo).

Learnability: Interpreting and Connecting to Conventional Representations

Another question we were interested in was whether learners were able to interact with notions of rate of change, accumulation, and the relationship between the two in a way that lets them "unpack" these notions as they relate to the mechanisms of change in systems. In this section, we recount two such cases.

Rate as Representing Complex, Multi-behavioral Events. We found that often in our interviews, making sense of aggregate rates of change in terms of individual behaviors was an interpretive challenge for learners (much how it is difficult for learners to interpret behaviors at different levels in a systems; Wilensky & Resnick, 1999). In several cases, learners started off speaking of rate and derivative as an inert mathematical notion disconnected from the very phenomena it is intended to model. This was the case for Hannah (*Interview 6*), who before building her own models was asked what might make the graph rise, then fall. She suggested that "a genocide or natural disaster happened", but went on to explicitly note that events that reduce the population "doesn't really affect the rate, it just, it's just something like an outside thing that affects the population".

Hannah's confusion regarding what behaviors or aspects of the model the *rate* of population change actually represented was echoed by many other learners – 6 of those interviewed were not able to make a link between the number of people born in the model and the rate of change of the population for a given unit of time without guidance. After having modified the model to observe how different behaviors all contributed to the same quantity and the way it changed over time, however, Hannah not only felt comfortable talking about rate in the context of more than one behavior, but also in terms of how differences in how a single agent behaves (extending agents' life span) can interact with other behaviors (more time to reproduce) and contribute to overall changes in population over time: "Well I knew that if I increased the death age then it wouldn't decrease as much and um, I don't think I needed to increase the uh, probability of the to reproduce… because they would have more time to live and reproduce they would still be people (mmkay) around."

By observing and controlling a *mathematical* idea in terms of *behavior*, we suggest that Hannah was able to integrate an understanding of how multiple behaviors interact with one another with the notion of rate, which measures the results of those behaviors. In this case, she thought about multiple different ways to increase population: first, by giving each agent more time to reproduce by allowing them to live longer, and second, by increasing their probability of reproducing at each tick.

Defining Mathematical Terms with Behavioral Relationships. Finally, we argue that representing change in systems in terms of agent behaviors can provide learners with insight into how the mathematical notions of rate of change and accumulation relate, and what they represent in a modeling context. In one interview, we ask Brooke (*Interview 3*) how we could find the population's rate of change for a given year. Although the model featured a plot of the number of individuals born at each tick in the model, as well as a plot of the total population per tick from which this information could be extracted, she only suggests that "you could use derivatives" – presumably referring to the mathematical procedure for determining a rate of change given an algebraic expression of the change itself – as a means to determine change for a tick in the model. When probed for whether she could think of any other way "with all the information you're given here", she responded "Um, I dunno, I'd have to think about that. Kind of like derivatives all stuck in my mind."

Later, after explicitly being asked how the two plots featured in the model are related, Brooke recognizes birth in the population as defining the rate of change in this simple model: "…our original population is taking this (points to lower graph) added to uh people that there were, that there were beforehand, (mhm) before they were (mhm) the people were born". Later, when asked whether she could relate these graphs to the idea of a derivative, Brooke notes: "…derivatives is basically taking like an exact (mhm) point divided by another exact point finding the exact um, like change, but this gives us the exact change over the exact time. It gives us the exact number of people born at a certain time which is what derivatives is, is solving for."

Discussion and Future Work

Change – especially complex, systemic change – is an increasingly important part of our world. In this paper, we argue that agent-based modelling can provide learners with a new way to "speak change", as well as a bridge to the conventional mathematical models used to represent such change. The results we have reported are preliminary, and we are currently conducting a new series of studies that we hope will provide more insight into the relationship between agent-based and mathematical modeling. However, we are excited by the potential that such environments hold for exposing learners to the complexity, power, and relevance of the mathematics of change for understanding our world.

References

Abrahamson, D., & Wilensky, U. (2007). Learning axes and bridging tools in a technology-based design for statistics. *Int. Journal of Computers for Mathematical Learning, 12*(1), 23-55.

Blikstein, P., & Wilensky, U. (2009). An atom is known by the company it keeps: A constructionist learning environment for materials science using agent-based modeling. *International Journal of Computers for Mathematical Learning, 14*(2), 81-119.

diSessa, A. (2001). *Changing minds: Computers, learning, and literacy*. Cambridge, MA: The MIT Press.

Kahn, K. (2007). *Building computer models from small pieces.* Paper presented at the Proceedings of the 2007 Summer Computer Simulation Conference, San Diego, CA.

Kaput, J. (1994). Democratizing access to calculus: New routes to old roots. In A. Schoenfeld (Ed.), *Mathematical thinking and problem solving* (pp. 77-156). Hillsdale: Erlbaum.

Kaput, J., Noss, R., & Hoyles, C. (2002). Developing new notations for a learnable mathematics in the computational era. In L. D. English (Ed.), *Handbook of international research in mathematics education*. Mahwah, NJ: Lawrence Erlabum Associates.

Langton, C. G. (1997). *Artificial life: An overview*. Cambridge, MA: The MIT Press.

Levy, S., & Wilensky, U. (2009). Learners' learning with the connected chemistry (CC1) curriculum: Navigating the complexities of the particulate world. *Journal of Science Education and Technology, 18*(3), 243-254.

Nemirovsky, R., Tierney, C., & Ogonowski, M. (1993). *Children, additive change, and calculus*. Cambridge, MA: TERC-WP-2-93.

Niss, M., Blum, W., & Galbraith, P. L. (2006). Introduction. In W. Blum, P. L. Galbraith, H.-W. Henn & M. Niss (Eds.), *Modelling and applications in mathematics education: the 14th ICMI study* (pp. 1-32). New York, NY: Springer.

Noss, R., Healy, L., & Hoyles, C. (1997). The construction of mathematical meanings: Connecting the visual with the symbolic. *Educational Studies in Mathematics, 33*, 203-233.

Pozzi, S., Noss, R., & Hoyles, C. (1998). Tools in practice, mathematics in use. *Educational Studies in Mathematics, 36*(2), 105-122.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Repenning, A., & Ambach, J. (1997). *The agentsheets behavior exchange: Supporting social behavior processing.* Presented at the Conference on Human Factors in Computing Systems.

Roschelle, J., Kaput, J., & Stroup, W. (2000). Simcalc: Accelerating learners' engagement with the mathematics of change. In M. Jacobson & R. B. Kozma (Eds.), *Research, design, and implementing advanced technology learning environments*. Hillsdale, NJ: Erlbaum.

Rutherford, F. J., & Ahlgren, A. (1991). *Science for all Americans*. New York: Oxford University Press New York.

Sengupta, P., & Wilensky, U. (2008). On the learnability of electricity as a complex system. In G. Kanselaar, J. van Merriëboer, P. Kirschner & T. de Jong (Eds.), *Proceedings of the Int. Conference for the Learning Sciences,* (Vol. 3, pp. 258-264). Utrecht, The Netherlands: ISLS.

Steed, M. (1992). STELLA, a simulation construction kit: Cognitive processes and educational implications. *Journal of Computers and Mathematics and Science Teaching, 11*, 39-52.

Stroup, W. M. (2002). Understanding qualitative calculus: A structural synthesis of learning research. *International Journal of Computers for Mathematical Learning, 7*, 167-215.

Wilensky, U. (1999). NetLogo: http://ccl.northwestern.edu/netlogo.

Wilensky, U. (2003). Statistical mechanics for secondary school: The gaslab modeling toolkit. *International Journal of Computers for Mathematical Learning, 8*(1), 41.

Wilensky, U. (2006). Complex systems and restructuration of scientific disciplines: Implications for learning, analysis of social systems, and educational policy. In J. Kolodner (Chair) and C. Bereiter & J. Bransford. (Discussants), *Complex Systems, Learning, and Education: Conceptual Principles, Methodologies, and Implications for Educational Research*. Presented at AERA 2006, San Francisco, CA.

Wilensky, U., & Abrahamson, D. (2006). *Is disease like a lottery? Classroom networked technology that enables student reasoning about complexity.* Paper presented at the Annual Meeting of the American Educational Research Association.

Wilensky, U., & Papert, S. (In Preparation). Restructurations: Reformulations of knowledge disciplines through new representational forms.

Wilensky, U., & Rand, W. (in press). *An introduction to agent-based modeling: Modeling natural, social and engineered complex systems with NetLogo.* Cambridge, MA: MIT Press.

Wilensky, U., & Resnick, M. (1999). Thinking in Levels: A Dynamic Systems Perspective to Making Sense of the World. *Journal of Science Education and Technology, 8*(1).

Yerushalmy, M. (1997). Mathematizing verbal descriptions of situations: A language to support modeling. *Cognition and Instruction, 15*(2), 207-264.

Agent-Based Modeling with NetLogo: Exploring, Designing, and Building

Uri Wilensky, Michelle Wilkerson-Jerde & CCL *{uri; m-wilkerson}@northwestern.edu*
Learning Sciences, Comp. Sci., and Center for Connected Learning, Northwestern University

Introductory description and overall goals

NetLogo is a modeling environment that enables students and teachers to create agent-based models of *complex systems* – that is, systems that involve many interacting elements. NetLogo follows the Logo design philosophy of "low threshold and high ceiling," meaning that it provides a gentle learning curve for beginners by enabling students, teachers and curriculum developers to create their own “emergent” models of systems in terms of individual, embodied behaviors. There are also several model-based curricula that employ NetLogo at middle school, high school, and undergraduate levels. In this workshop, we will introduce participants to the NetLogo modeling environment, and how to use it to explore, design, and construct agent-based models.

Figure 1. NetLogo logo

Method

We will utilize a hands-on approach to learning ABM. We will start out with a discussion of ABM concepts, techniques and examples. We will also introduce participants to NetLogo’s extended capabilities – including support for robotics and data collection, participatory simulation, and systematic model analysis. Workshop participants will be led together through the development of a basic model in the NetLogo environment. Finally, participants will have a chance to brainstorm and receive help and feedback in designing their own agent-based model.

Expected outcomes

The workshop is intended to serve the needs of both educators and researchers. <u>No previous experience with NetLogo or programming is required for this workshop.</u> The large variety of phenomena that can be explored with NetLogo will make this workshop a truly interdisciplinary experience. Participants will complete the workshop having built a basic NetLogo model and with the tools and access to information and resources for designing and building their own models.

Keywords

modeling, NetLogo, simulation, complex systems, agent-based modeling, programming

Introduction To Flunstellas: Using StarlogoTNG to represent Flunstellas Psychological Systems

Neil Winterburn, *neil.winterburn@flunstellas.org*
Co-Director Re-Dock, Liverpool, England.

Introductory description and overall goals

The workshop will introduce participants to the 'Flunstellas Narrative Framework', developed to actively engage as wide a range of people as possible, in the creation of personal representations of Mind.

Method

This workshop uses a combination of methods to enable participants to represent mind(s) as personal & dynamic, complex systems, populated by thoughts, memories and ideas.

Building Hands on Information Architecture.

Using simple materials to explore the concepts behind new technologies, away from their shiny digital newness, see figure 1.

Figure 1. Flunstella 'Mental Object', (Knex, Plastercine, Blu-Tac & Card)

Figure 2. Screen Shot Close Up of a Simple Model of a Flunstella.

Using StarLogo TNG to Populate and Explore Models of Mind, see figure 2.

Expected outcomes

Participants will collaborate to develop a Visual Grammar to represent their mental systems, Create tangible representations of different Flunstella, 'Mental Objects', and use these physical objects as stimulus to create Flunstellas Systems, using StarLogo TNG.

Keywords

Flunstellas, Starlogo TNG, Re-Dock, Mind, Systems.

Programming a robotic system to deal with water problems

Marios Xenos, *mariosxenos@ppp.uoa.gr*
Educational Technology Lab, School of Philosophy, University of Athens

Short presentation

The following proposal exploits the graphical programming environment that comes with Lego Mindstorms NXT kit, in order to introduce students with essential programming structures. Specifically, in order to motivate students to build a program, it presents the authentic problem of water management and asks them to formulate their suggestions - solutions through programming the behaviour of a well prepared system.

A water tank with sensors and valves comprises a watering system and needs an appropriate "behaviour" (figure 1). Students are facing the authentic problem of the water management: "How must the system be programmed in order not to waste water in a farm?"

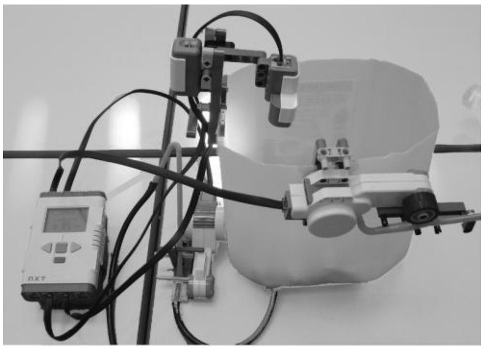


Figure 1. The watering system

The above main question breaks down into programming problems that have been addressed by the students-programmers:

- How can the sensor values be inspected continuously?
- In which way the sensor values have to affect the motor/valve movement?
- Is there a necessity to have somehow a "save" functionality for some values?

To solve the above problems, students reconstruct a predefined program by adjusting correctly the available "blocks" that appear on the programming environment, in order to give the system the appropriate behaviour.

Behind the well-designed blocks, programming structures and concepts are hidden. Teacher does not present any of these structures or teaches programming concepts. He acts as a "facilitator" and just helps the learners to find a solution to the watering problem. However, some important and interesting issues that refer to the use of variables, repetition structure and selection structure come to surface.

Keywords

Educational Robotics, problem-solving, programming

School Effects Reinterpreted from the Bottom up

Christine K. Yang, *cky@u.northwestern.edu*
Learning Sciences, School of Education and Social Policy, Northwestern University

Poster Abstract

This is a work-in-progress project that involves the design of an agent based model that enables educational researchers to understand school effects as an emergent process: a complex systems view on how hypothetical changes in education policy can bring about various outcomes. I define school effects as associations between school-level variables and student achievement outcomes. This is measured as the percent variation that lies between schools rather than within schools in student achievement outcomes. School-level variables consist of school attributes, which are stable traits, and treatment variables, which can be changed and hence be used as policy levers.

This project contains two parts. The first part involves a statistical analysis on a national survey data from the National Educational Longitudinal Study in the U.S., where student and school information were gathered from 8th, 10th and 12th grades. Using HLM, I examine how much of the variation in student academic achievement is explained by factors specific to school attributes, such as structure and composition, as well as school treatment variables, such as pupil-teacher ratio. I also estimate how much between-school differences account for the variability in student achievement, measured as achievement status and achievement gains. The preliminary results indicate that school level attributes and treatment variables are differentially associated with students' academic achievement gains versus achievement status. In particular, schools' treatment variables seem to matter less than schools' attributes, especially with regards to student achievement gains. This is a little disheartening, as one would hope that changes in such policy levers will have a positive impact on students' average performance over time, regardless of the location, structure, or socio-economic status of the schools.

In response to such findings, and to theorize and better explicate the mechanisms behind school effects, I have been developing a simple agent-based model using NetLogo as the second part of my project. This model uses parameter estimates of treatment and attribute variables from the HLM model. The model also allows for the possibility to include school choice to better emulate reality, particularly the current policy interest in open district enrolments. This computational model aims to illustrate how changes in treatment variables and students' school choice preferences, while accounting for school level attributes, can lead to emergent phenomena that are reflective of or different from the statistical results. Its purpose is be used mostly as a research tool for educational researchers, but also possibly a learning environment that can guide the decision making processes of policy makers, as they attempt to understand and make suggestions for improving the educational system.

Keywords

School effects, agent based modelling, education policy