Kits for Learning and a Kit for Kitmaking

Carol Strohecker
Adrienne H. Slaughter

A shorter version of this paper appears in *Extended Abstracts, CHI'00, Human Factors in Computing Systems,* Assoc. for Computing Machinery, 149-150.

Originally appeared as Technical Report 2000-02, Mitsubishi Electric Research Laboratories.

Abstract

We bring together concerns in software design and learning theory through creation of a Java framework for development of software construction kits. The kits are highly visual and highly interactive, and are premised on the notion of "microworlds" as environments for learning and learning research [6]. Usage of four existing kits is informing development of the framework, which in turn we are applying to development of a new kit. The kits support construction of two-dimensional, graphical structures that behave in characteristic ways when activated. We employ design heuristics of "object permanence," "transparency," and use of multiple simultaneous views to illustrate shifts of scale, perspective, time, and representation. Broader use of the general "Kit4Kits" will help us address viability of our "elements and operations" design approach.

Kits for Learning and a Kit for Kitmaking

Carol Strohecker
Adrienne Slaughter
Mitsubishi Electric Research Laboratory (MERL)
201 Broadway
Cambridge, MA 02139 USA
+1 617 621 7517, +1 617 621 7594
stro@merl.com, slaughter@merl.com

ABSTRACT

We bring together concerns in software design and learning theory through creation of a Java framework for development of software construction kits. The kits are highly visual and highly interactive, and are premised on the notion of "microworlds" as environments for learning and learning research [9]. Usage of four existing kits is informing development of the framework, which in turn we are applying to development of a new kit. The kits support construction of two-dimensional, graphical structures that behave in characteristic ways when activated. We employ design heuristics of "object permanence," "transparency," and use of multiple simultaneous views to illustrate shifts of scale, perspective, time, and representation. Broader use of the general "Kit4Kits" will help us address viability of our "elements and operations" design approach.

Keywords

Java framework, design patterns, microworlds, learning

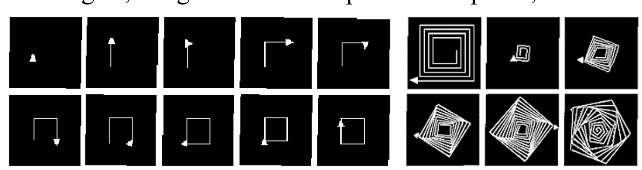
INTRODUCTION

Recently, CHI contributors have become increasingly interested in human learning processes. In many discussions the term "usability" has given way to "learnability," reflecting the growing complexity of interactive computational environments and the struggles to render them discernable, tailorable, managable. Learning research shares motivations and methods with usability research but is more far-reaching: to understand how a person learns we have to consider deep conceptual structures and their growth over time. The inquiry is fundamentally longitudinal. It benefits from the premise that people do not simply absorb or acquire knowledge, but actively create it for themselves, and that they do so especially well when engaged in the actual creation of something in the world – something that can be regarded, shared, changed, and liked [3, 5, 6, 9]. The benefit of this view is that it encourages fabrication of environments to support people's constructions, thus enabling study of conceptual structures.

In order to be useful in learning research, a constructive environment has to be specifically designed. A phase of learning research involves design and development of tools for the inquiry, just as a chemist might participate in development of new microscope. Papert describes how carefully designed, interactive "microworlds" can focus users' thinking on particulars of a conceptual domain while providing for self-motivated, free-form manipulation and transformation of the representations –and the ideas [9]. We think of this sort of interactivity as "intuition building," a kind of exercising that promotes growth of understandings.

FROM LEARNING THEORY TO DESIGN PREMISE

In Papert's example of Turtle Geometry, a graphical object is characterized by just two properties: position and heading. The object leaves a trace as it moves. By applying operations like forward movement and turns to create pictures, users (typically children) deal with and can come to understand the concept of vector. This building-block idea can, in turn, help in developing further understandings – of angles, the geometries of squares and spirals, etc.[1]

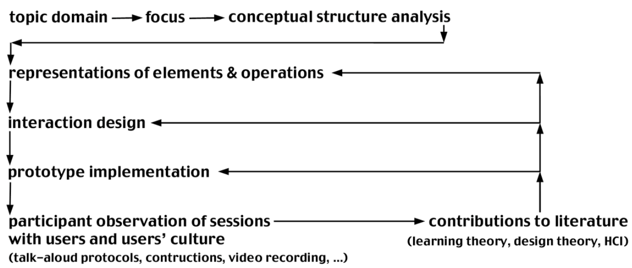


Thus an understanding of a shape like the square includes both its elemental properties and the operations that transformed them. The shape, and the knowledge of the shape, consist of relationships of one line segment to another, which resulted from application of basic geometric transformations. Actions become operations. Over time, through repeated constructions of squares, the person no longer thinks about how the shape was formed. Understanding the shape becomes a matter of intuition, basic knowledge. But if that knowledge were to be

[1] These illustrations are adapted from [9, 72-73].

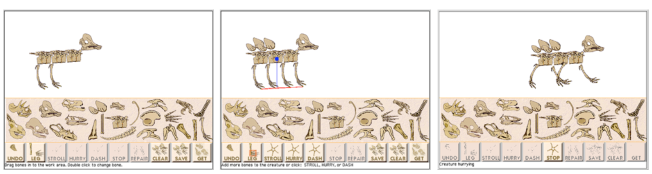
unpacked, it would consist fundamentally of the constituent elements and operations [3, xxix-xxxiv].

We apply this theory as a design principle for our microworlds, or "kits." Our goal is to provide elements and operations with which people can build new structures – and with them, certain structured understandings. Thus the research process involves conceptual analysis, design, tool building, and extensive work with users/learners. Like any good design process, it is cyclic:



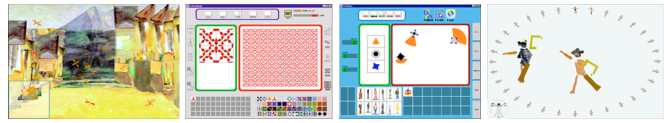
KITS FOR LEARNING

Like Turtle Geometry, our kits are simple in principle but rich in possibilities for playful, thoughtful, creative activity. The Bones kit, for example, includes elements that have two simple properties: mass and position. When combined to locate a composited creature's overall center of mass, these properties determine whether the creature will maintain its structural integrity and balance. The software includes gait patterns for creatures with any number of legs and three speeds of movement.



A microworld is like a microscope zooming in on one aspect of a complex subject. For example, the Bones topic domain is most broadly, science, but more particularly, physics, and even more particularly, motion study. Still more particularly, the kit focuses on balance, and most particularly on the role of center of mass in balancing. Microworlds pare away all but the most salient features of a subject, retaining its integrity while enabling learners to experiment with basic ideas.

Other kits in our growing genre focus on aspects of topology, geometry, symmetry, sensori-motor functions, time/space relationships, and system dynamics. Kit users assemble characters and objects from smaller parts, a process that involves designing behaviors as well as structures. Dinosaur skeletons balance as they walk and run; maps reveal street-level views, geometric tiles form symmetric patterns, animistic creatures spawn, maintain, and disrupt social distances; and dancers' breathing patterns determine cyclic timing for a shared dance.[2]



These kits are in various stages of prototyping and evaluation.[3] During the past several months we have attempted to generalize from what we have learned in their design, development, and use. Our endeavor has involved several simultaneous activities:

- observation of what works best in the existing kits (Bones, WayMaker, PatternMagix, and AnimMagix),
- formulation of these observations as high-level design principles,
- application of these principles to a new kit (Zyklodeon), and
- harvesting and development of Java code to implement the design principles generally, for the Kit4Kits, and to apply them particularly, for the new kit.

A KIT FOR KITMAKING

Our Kit4Kits is a Java framework including a package of modifiable code and guidelines for using it [2]. Like most such frameworks, the Kit4Kits includes code for specifying, tracking, altering, and reporting on system states; for creating structure, function, and appearances of objects; for generating screen layout items and widgets; and for capturing and dealing with actions in the event-driven system.

HCI patterns have been proposed as a means of managing increasing complexity in interaction design [1, 14]. Pattern-related Java frameworks are also emerging as means of developing educational software, like Brown University's interactive illustrations [11]. What distinguishes our effort is the conceptual underpinning that guides design of the microworlds for both learning and learning research.

We have combined premises from this tradition with usage data from our existing prototypes to formulate these key design heuristics:

- "object permanence" – This is a play on the Piagetian term for a process by which young humans come to understand that things remain in the world even when

---

[2] Publication of this work is too extensive for complete citation here; check [13] and http://www.merl.com/threads/learn/index.html for details.

[3] Bones v.1 design, development, and use, 1994; v.2 design and development, 1997; v.3 projected 2000. WayMaker v.1 design and development, 1996; v.1 use and v.2 design and development, 1998; v.3 projected 2000. PatternMagix v.1 design, development, and use, 1997; further use projected 2000. AnimMagix v.1 design, development, and use, 1998; further use projected 2000. Zyklodeon v.1 design and development, 1999; use projected 2000. Kit4Kits design and development, 1999; use projected 2000.

they are not being noticed or used [3]. In our software worlds, screen areas, buttons, and other devices don't just disappear when not needed. We maintain some miniature representation or other recall mechanism when screen real estate becomes a problem.

- "transparency" – [c.f. 10] We display visualizations of algorithms, calculations, and processes whenever possible [15, 17]; we represent constituent properties of objects, often in ways that facilitate users' modifications of them; and we provide various forms of visual and aural feedback so that results of actions are apparent.

- multiple simultaneous views – Comparisons help people to perceive the shifts of scale, perspective, time, and representation that can be fundamental to understanding dynamic phenomena [16, 17]. Through the use of graphic treatments such as side-by-side views, miniature displays, and the like, we address the principle that you don't really understand something unless you understand it in more than one way [8]. The kits may also be more accessible to a range of users with diverse thinking styles [18].

We have in mind a chain of users:

- stagers – designers and implementers who use the Kit4Kits to assemble microworld-style, domain-specific, software construction kits
- players – people who use the microworld-style kits to create objects that transform and behave in domain-specific ways (end-users from the perspective of the Kit4Kits)
- friends – The Players may find their own users as people trade and play with the objects they produce. The contexts in which such trading and playing occurs are beyond the scope of this paper but very much present in our thinking as we develop the Kit4Kits [e.g. 12].

The kit we are currently developing (Zyklodeon) realizes the design heuristics most comprehensively. We are nearing completion of the v.1 implementations of Zyklodeon and the general Kit4Kits. We will soon enter a phase in which we will offer the general kit to developers experienced in Java programming who want to make game-like learning environments of the kind we envision.

FURTHER INQUIRY

As we refine the Kit4Kits and work with players of kits whose development it has facilitated, we hope to address questions such as: What are players thinking about as they use a design realization that has been guided by notions of conceptual elements and operations, and principles of object permanence, transparency, and multiple simultaneous views? How does the players' thinking change over time? What other kinds of activities do they engage that seem to be informed by the same kinds of thinking? Are they doing a kind of visual programming as they use a kit? Does the "elements and operations" formulation work as a design approach, or do we get caught in the problem of inverting an analytic process, which critics of architectural design patterns have complained about [4, 7]?

ACKNOWLEDGMENTS

Several people have contributed to design, development, and use of the kits that have informed our framework: AARCO medical illustrators, William Abernathy, Edith Ackermann, Aseem Agarwala, Noah Appleton, Maribeth Back, Barbara Barros, Dan Gilman, Mike Horvath, John Shiple, students at Harvard University's Graduate School of Design, colleagues at MERL, and anonymous friends. We thank Glenn Blauvelt, Mike Eisenberg, Barry Perlman, Vennila Ramalingam, Doug Smith, and Tom Wrensch for discussions of museum- and community-based learning environments and uses of kits like ours in such environments. Discussions with Stephanie Houde, Sarah Kuhn, Warren Sack, Tim Shea, and with Fred Martin, Seymour Papert, Brian Silverman, and other members of the Epistemology & Learning Group at the MIT Media Lab, have inspired and/or improved various kits as well as the more general formulations. MERL supports the work.

REFERENCES

1. Erikson, T., and Thomas, J. Putting it all together: Pattern languages for interaction design. *Proceedings of CHI'97,* 226. See also the summary report of the workshop at http://www.pliant.org/personal/Tom_Erickson/Patterns. WrkShpRep.html.
2. Gamma, E., Helm, R., Johnson, R., and Vlissides, J. *Design Patterns: Elements of Reusable Object-Oriented Software.* Addison-Wesley, Reading, MA, 1977.
3. Gruber, H. E., & Vonèche, J. J. (eds.). *The Essential Piaget.* Basic Books, New York, 1977.
4. Gabriel, R. P. The failure of pattern languages. *Journal of Object-Oriented Programming* (1994).
5. Harel, I., & Papert, S. (eds.). *Constructionism.* Ablex, Norwood, NJ, 1991.
6. Kafai , Y., and Resnick, M. (eds.) *Constructionism in Practice: Designing, Thinking, and Learning in a Digital World.* Lawrence Erlbaum, Mahwah, NJ, 1996.
7. Lynch, K. Reconsidering the image of the city, in Rodwin, L. and Hollister, R, M. (eds.). *Cities of the Mind: Images and Themes of the City in the Social Sciences.* Plenum Press, New York, 1984, 151-161.
8. Minsky, M. *The Society of Mind.* Simon and Schuster, New York, 1986.
9. Papert, S. *Mindstorms: Children, Computers, and Powerful Ideas*.Basic Books, New York, 1980.
10. Resnick, M., Berg, R., and Eisenberg, M. Beyond black boxes: Bringing transparency and aesthetics back to scientific investigation. *Journal of the Learning Sciences* (1999).
11. Simpson, R. M., Spalter, A. M., van Dam, A. Exploratories: An educational strategy for the 21st century. Brown University online ID number: schoolhouse_1449, 1999.
12. Strohecker, C. A model for museum outreach based on shared interactive spaces. *Multimedia Computing and Museums: Selected Papers from the Third International Conference on Hypermedia and Interactivity in Museums,* (Archives & Museum Informatics, Pittsburgh, 1995), 57-66.

13. Strohecker, C., and Barros, B. 1997. A prototype design tool for participants in graphical multiuser environments. *CHI'97 Extended Abstracts,* 246-247.

14. Tidwell, J. Common ground: a pattern language for human-computer interface design. http://www.mit.edu/~jtidwell/ui_patterns_essay.html, 1999.

15. Tufte, E. R. 1983. *The Visual Display of Quantitative Information.* Graphics Press, Cheshire, CT, 1983.

16. Tufte, E. R. 1990. *Envisioning Information.* Graphics Press, Cheshire, CT, 1990.

17. Tufte, E. R. *Visual Explanations: Images and Quantities, Evidence and Narrative.* Graphics Press, Cheshire, CT, 1997.

18. Turkle, S., and S. Papert. Epistemological pluralism: Styles and voices within the computer culture. *Signs* 16:1 (1990).