Bricolage Forever!

Mordechai Ben-Ari
Department of Science Teaching
Weizmann Institute of Science
Rehovot 76100 Israel
ntbenari@wis.weizmann.ac.il

Abstract

According to constructivism, learning takes place by constructing cognitive structures. An experiment was carried out to uncover the mental models of experienced, but not professional, users of a word processor. The study shows that the mental models are superficial, and that bricolage (trial-and-error) is extensively used even though it is not viable for solving non-trivial problems. The conclusion is that conceptual models must be explicitly taught—contrary to the task-oriented approach of minimalism.

Introduction

In a recent paper (Ben-Ari, 1998), I discussed the application of constructivism to computer science education. One of my conclusions was that a model of the computer must be explicitly taught in introductory classes in computer science and programming; otherwise, the inevitable construction of a mental model by the student—as predicted by constructivism—will be haphazard and non-viable.

In the oral presentation of the paper at the Twenty-Ninth SIGCSE Technical Symposium on Computer Science Education, I gave an example of the presumed behavior of a user of Microsoft Word when presented with a non-trivial problem. My prediction was that the performance of an ordinary user would be erratic and inefficient. The reason is that the what-you-see--is-what-you-get (WYSIWYG) design hides the internal model of the implementation, and that vague visual clues and behaviorist training methods are not sufficient to enable the user to develop a *viable model* (Glasersfeld, 1995).

This paper reports on preliminary empirical research intended to investigate the truth of my prediction. Audio tapes were made of colleagues verbalizing the steps they took when modifying a Word document according to my instructions. Despite the intellectual capabilities and the experience of the subjects, the mental models they demonstrated were superficial, and did not approach the sophistication of the conceptual models used by the developers of the software.

Theoretical background

This work is set within the framework of three related theories and methodologies: constructivism, bricolage and minimalism.

Constructivism is a theory that claims that knowledge is not transmitted, but rather embodied in cognitive structures constructed by each individual. The theory has profound implications in philosophy (epistemology—what is knowledge?), psychology (cognition—how do you know something?) and education (pedagogy—how can you teach someone to know?). See Ben-Ari (1998) for a quick overview, and the articles in Steffe and Gale (1995) and Matthews (1998) for a comprehensive discussion. The question that interests me is this:

> If, as the theory claims, an idiosyncratic mental model is *necessarily* constructed, should teachers attempt to instill a canonical model, or should teachers let nature take its course, so to speak?

Bricolage is used to denote a concrete as opposed to an abstract learning style. Turkle and Papert (1990)[1] propose that bricolage be encouraged in computer science education as a way of making the subject accessible to a broader range of students. Given the highly structured nature of the computer domain, a tendency to bricolage would manifest itself as inefficient, trial-and-error actions based on a superficial model—here, the WYSIWYG model.

Minimalism (Carroll, 1990, 1998) is a methodology for designing manuals for software documentation and for using these manuals in training users of the software. A minimalist manual is short, stresses active learning and considers errors to be opportunities for learning rather than mistakes to be corrected. Minimalism has much in common with constructivism; it differs in its insistence that conceptual material *not* be included in training, or at least that it be deferred until the student is more experienced. These days, learning to use a word processor is, more likely than not, based on the behaviorist style favored by minimalism. When faced with an unfamiliar situation, the user will not attempt to employ or expand conceptual knowledge, but rather will attempt to find and recycle a task that was 'actively learned'.

Note that some supporters of minimalism have backed off the rejection of conceptual material:

> . . . a manual must: Help users grasp the big picture of the product, that is, help users develop a mental model that helps them predict what to do. (Redish, 1998, p. 240).

[1] The version of the article in Turkle and Papert (1991) may be more accessible.

Description of the Research

Subjects

Ten members of my department volunteered for the experiment which required only 15–20 minutes of their time. With one exception, the subjects were science teachers with 10–20 years of experience. They work in the department on course development, and are also working on graduate degrees in science teaching. They use Word extensively for writing educational material, proposals, theses, and so on.

Thus the subjects can be characterized as highly intelligent people who are sensitive to issues of cognition and pedagogy. They use the software day-to-day, but are not professional writers or typists. Almost all of them claimed that they were not fit subjects for an experiment because they are not experts in Word!

The one exception in terms of demographics was a summer student, a recent university graduate in engineering, who is developing a sophisticated educational software system for my group. Aside from him, I did not select subjects from my own group, as they had already heard my lecture and knew what I was looking for.

Methodology

I developed a set of five quizzes. The intention was that each quiz could be easily solved if the subject had the requisite conceptual knowledge required, but that trial and error would be very inefficient in solving the problem. The quizzes check the following concepts:

- *Attributes* like fonts can be associated with *empty space*.
- In the Hebrew version of Word, *cursor movement is logical*: when moving the cursor 'forward' within right-to-left Hebrew text, it will jump to the left to the beginning of enclosed left-to-right English text. It is possible to position the cursor such that the character displayed when you press a key depends on the direction of arrival of the cursor!

- An ordinary carriage return inserts a *paragraph mark*. Within numbered and bulleted lists, a paragraph mark will start a new item. To create a new line within an item, you have to use a *soft carriage return*.
- Within a table, there is a distinction between *selecting a cell* and *selecting the text* within the cell.
- A text box can be *anchored* to the surrounding text.

A 'successful' solution to a quiz would be a viable diagnosis of the problem by the subject. The ability of the subject to actually solve the problem is secondary, as that depends on the subject's training and experience.

The subjects were presented with a Word document containing the five quizzes and a printed sheet with instructions. They were encouraged to verbalize their actions while I taped them. As they had used similar techniques in their own research, they understood and accepted the procedure. Nevertheless, some were defensive as I 'showed-up' their lack of knowledge of Word. Upon completion of the exercise, I explained my intent.

Results

Considering the high quality of the subjects, the most surprising result was the low level of use of this very sophisticated software tool. For example, no subject mentioned concepts such as *style sheet* which are an essential part of the internal model of the software. On the first quiz concerning the font of empty space, several subjects talked about the default font, even though there is no such thing—only the font of the default style.

One of the subjects performed the experiment at what I regard as a significantly higher cognitive level than the others. Her performance will be analyzed in a separate section below.

The young student turned on the option Show All that displays some formatting signs instead of pure WYSIWYG. Despite having relatively little experience using Word, he was readily able to solve the problems (in my sense of the word as discussed above). Fonts of empty space are displayed, as are the soft carriage returns and the text box anchor. In spite of the contribution of explicit formatting cues, another subject who did know of the option talked about it as being 'annoying'!

A surprising result was the degree of anthropomorphic volition attributed to the software. While this behavior is well-known from the study of novices (Pea, 1986), it was unexpected from experienced science teachers. Here are some quotes:[2]

- "You see that's what I mean, it behaves strangely."
- "He did it to me again."
- "He knows, but I don't."
- "That's not nice of him."

A quiz in depth

I will now discuss one quiz in depth, as it seemed to bring out the most consistent results. The subjects were given the following text fragment:

1. The quick brown fox.
   The quick brown fox is sympathetic.
2. The lazy dog.
3. The clever cat.
   The clever cat is too smart for this story.

and the following instruction:

> Each numbered item should have a note after it. The first and third items already have a note. Insert a note for the second item with the text: The lazy dog sleeps all day. Then change the numbers to bullets.

[2] The first quote is taken from a native English speaker. The others are translated from Hebrew which does not have a neuter pronoun, so I may be over-interpreting the anthropomorphic meaning of 'he'. Nevertheless, I felt from that the tone of the utterances that the subjects behaved *as if* Word were an intelligent being.

Almost invariably, the subjects brought the cursor to the end of the second item, pressed carriage return and then typed the requested note. The note appeared as an extra numbered item. They then proceeded to visually remove the number and indent. Of course, changing to bullets meant that the whole process had to be repeated.

Only one subject (in addition to 'S' who is discussed below) was aware of the *concept* paragraph mark and its significance for numbered items. The subject did not remember the exact key sequence for a soft carriage return, but the trial-and-error search was quickly successful. If the subjects had worked with the Show All option, the indication that denotes the 'soft' carriage return could have raised a question in their minds. Similarly, if they had searched Help for 'bullet', they would immediately have found a help topic entitled:

> Every line is bulleted or numbered instead of just the first line.

which clearly explains the problem and gives the solution. Yet not one subject attempted to use Help; in fact, one subject claimed that she never used Help because it was too hard to use. Considering the comprehensiveness of Help in Word, this in itself says quite a bit about the psychology of the users of computer technology. Alternatively, the Help facility may be so difficult to use that—paradoxically—it can only be used if you know the approximate answer.

An exceptional subject

One subject—whom I shall name 'S'—performed significantly better than the others, even though her personal characteristics and experience in Word were in no way different from the others. It is instructive to analyze her performance in some detail.

On the first quiz, when asked to explain why the paragraph came out in large boldface font, she said:
"It is apparently in the title; it retains the font of the title; apparently only here [at the beginning of the paragraph] was the new font entered."
Clearly, 'S' is associating a font with empty space and is using the (viable) model that the document contains data that denotes where a font change begins and ends. Furthermore, she made several attempts to copy 'only the font', another indication that she sees the font as a concept whose concrete representation can be manipulated and not just as a WYSIWYG attribute.

The performance of 'S' on the numbered list was extremely interesting. She knew that entering a carriage return for a new line would cause a new item to be created:
"What happened is exactly what I thought would happen."
'S' knew that there had to be a way of doing what I requested, but she did not know how to do it. When I asked how she knew it, she replied that she had had experience with documents that did it. After she remarked:
"I am sure that there is a way to do it,"
I asked a leading question:
"How would you find out if there was a way to do it or not?"
Her initial answer was bricolage: "I'd look at all sorts of options,"
but then she finally raised the possibility of using Help. Eventually, she found the answer.

Despite her relatively high conceptual knowledge, 'S' repeatedly used expressions like 'playing around'—indicating a preference for bricolage as a style of work, and even borderline anthropomorphisms like:
"The tool is smart so that you do things that are common sense."

Discussion and conclusion

The superficiality of a WYSIWYG word processor hides a deep internal structure of the document and the software. It encourages task-oriented behavioristic training in the minimalist tradition, which in turn seems to awaken an irresistible urge to non-viable, or at least inefficient, bricolage. Explicit phrases like 'trial-and-error' and 'let us see what happens' were ex-

tensively used by the subjects, indicating conscious use of bricolage.

The subjects had not taken a course in the use of the software package; rather they learned it informally from colleagues or family members. Their performance tends to indict discovery and hands-on activities as ineffective learning methods, unless they are guided by a teacher with expertise in pedagogy and not just the subject matter. To quote Mitchel Resnick:

> Their hands are on, but their heads are out. (Resnick, 1997, p.28)

The fact that this behavior was consistent across a group of highly intelligent subjects is disturbing. In fact, two subjects went so far as to say that if the task had really been important, they would have dumbed down their techniques. One said that she would have worked like a donkey,[3] and the other said that if it was a question of life or death she would have used techniques that fixed the visual appearance regardless of what it did to the structure of the document.

The task-oriented minimalist approach is initially successful, but ultimately does not equip the students to solve non-trivial problems that may be encountered. It seems to destroy whatever incentive that may have existed to improve one's ability to learn. Viable concepts are not easily constructed and should be expressly taught. Even the use of the Help facility has to be taught, and the design of Help has to have a conceptual framework, rather than an all-encompassing list of terms and phrases.

To what extent are these results relevant for programming? I regard using a modern word processor as a programming activity: the user must perform a sequence of actions, highly constrained in syntax and semantics, to achieve a computational goal. Thus I believe that a similar experiment on programming tasks would yield similar results.

[3] The connotation in Hebrew is working without thinking.

I would be very interested if any of the workshop participants would like to collaborate in expanding this preliminary work into a long-term research program.

Acknowledgement

I would like to thank Yifat Ben-David Kolikant for her comments on the paper.

References

Ben-Ari, M. (1998). Constructivism in computer science education. *SIGCSE Bulletin*, *30*(1), 257–261.

Carroll, J. M. (1990). *The Nurnberg Funnel: Designing minimalist instruction for practical computer skill.* Cambridge, MA: MIT Press.

Carroll, J. M. (Ed.). (1998). *Minimalism beyond the Nurnberg Funnel.* Cambridge, MA: MIT Press.

Glasersfeld, E. von. (1995). A constructivist approach to teaching. In L. P. Steffe & J. Gale (Eds.), *Constructivism in education* (pp. 3–15). Hillsdale, NJ: Lawrence Erlbaum Associates.

Matthews, M. R. (Ed.). (1998). *Constructivism in science education.* Dordrecht: Kluwer Academic Publishers.

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

Redish, J. (1998). Minimalism in technical communication: Some issues to consider. In J. M. Carroll (Ed.), *Minimalism beyond the Nurnberg Funnel* (pp. 219–245). Cambridge, MA: MIT Press.

Resnick, M. (1997). *Turtles, termites, and traffic jams: Explorations in massively parallel microworlds.* Cambridge, MA: MIT Press.

Steffe, L. P., & Gale, J. (Eds.). (1995). *Constructivism in education.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Turkle, S., & Papert, S. (1990). Epistemological pluralism: Styles and cultures within the computer culture. *Signs: Journal of Women in Culture and Society*, *16*(1), 128–148.

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.