Software Trends


We all understand the rapid pace of change in the capabilities of computer hardware. We can trace the historical development of computer hardware by looking at mainframe computers, minicomputers, and microcomputers. The trend has been toward putting more powerful computers in the hands of the end user. From the time of the first commercially produced computers in the early 1950s, the cost effectiveness of computers has improved by a factor of about a million.

The pace of change of computer software has been slower, but steady progress has occurred. Quite a bit of the software change has been dependent on the steadily increasing power of computer systems. Software trends can be summarized as follows.

1. Technology-centered software. In the early days of computers, [compute] power was at a premium. Programmers devised clever tricks to make use of the very limited computer memory and the relatively slow speed of the machines. A person needed to have programming skills and good insight into computer hardware to make effective use of the machine.

2. User-centered software. As computer speed and memory size increased dramatically, much of the increased computer power was put into making computer applications more “user friendly.” The focus became one of developing computer applications that were powerful tools for workers. A steadily increasing percentage of computer users had little knowledge of computer programming or computer hardware.

3. Learner-centered software. This is the newest “wave” of change in software. As computer power continues to increase rapidly, we are beginning to see more emphasis on the computer user as a learner. Software will be designed to help the learner use the software. This change in emphasis will help learners of all ages.

Learner-Centered Software

You are undoubtedly familiar with various pieces of computer-assisted instruction (CAI) software. General categories include drill and practice, tutorial, simulations, and microworlds. Some researchers in artificial intelligence work on developing intelligent computer-assisted instruction (ICAI). ICAI software contains considerable knowledge of what the student is trying to learn, what an “expert” knows, and what can help a student learn. Moreover, it builds a model of what the student knows and his or her progress in learning. Thus, it adapts to individual learning needs.

You are probably also familiar with computer software that contains built-in “help” files. You can think of this as a type of “just in time” assistance or instruction in solving problems encountered when using the software.
The ICAI and help file ideas can be combined in any computer application such as a spreadsheet, database, or graphics package. A learner-centered version of such software could incorporate ICAI in its help system. As the user begins learning to use the software, the computer application would have knowledge of what an expert user knows and can do. It would have knowledge of a variety of pedagogy strategies that help move a novice user toward becoming a competent user and then an expert user. The software might begin by interacting with the user, gaining information about the user’s computer background, knowledge of the application area, and goals in learning to use the software. This initial learner profile would serve as a starting point as the computer application builds a profile of the novice user.

The development of learner-centered software is currently considered “cutting edge” research and development. Ten articles in the April 1996 issue of the Communications of the ACM examine various research projects that are developing and field testing such software. In most cases, the focus is on developing stand-alone ICAI, rather than taking the next step of integrating such ICAI into standard application tools. Some of the applications being explored include:

- software that helps students develop broadcast news reports. Roger Shank and Alex Kass’ system is a form of ICAI with extensive computer-based video-production capabilities. This environment can be used to explore a wide range of topics—essentially those that would ordinarily be explored in a news broadcast.
- an intelligent multimedia tutoring system developed by Beverly Woolf, who has researched its applications to both medical and engineering education. The former application helps students learn to make use of cardiac care instrumentation in a hospital emergency room. The latter helps students learn to use computer-aided design (CAD) software.
- a “collaboratory notebook” developed by Daniel Edelson, Roy Pea, and Louis Gomez. They are exploring its use in helping groups of students work together as they learn inquiry-based science.
- lifelong learning. Hal Eden, Mike Eisenberg, Gerald Fischer, and Alexander Repenning’s software attempts to strike a balance between self-directed, discovery-based constructivism and guided instruction. One of their major focuses is to develop tools and environments in which students can create computer simulations.

A unifying theme in all of these examples is a combination of constructivism and problem solving in an advanced information-technology environment. Students, individually and in groups, use information technology tools as they address problems that are both meaningful to their current developmental levels and authentic.

**Final Remarks**

The development of learner-centered software is in its infancy. You can see its beginnings in modern software tools, such as word processors and spreadsheets. You can also see it in good computer-assisted learning materials. Look for this as you evaluate software for your personal use and for use by your students.