The Spreadsheet


The Software Publishers Association (SPA) has approximately 1,200 corporate members and is broadly representative of the software industry. The SPA annually gives Codie Awards to outstanding people and products. The winner of the 1998 Codie Award for Lifetime Achievement was Dan Bricklin. He invented the first spreadsheet—named VisiCalc (for “visible calculator”)—in 1979.

**A Little History**

Back in 1979, the microcomputer industry was in its infancy. Many competing companies were producing 8-bit microcomputers that ran at a speed of approximately 1 megahertz. These machines used the BASIC programming language, and a steadily growing number of games and application software was available. From the viewpoint of mainframe, minicomputer, and time-shared computer manufacturers, microcomputers were just toys. They lacked both the power and software to serve the needs of business customers.

The spreadsheet changed this. A spreadsheet is “merely” a computerization of the two-dimensional accounting sheet used for bookkeeping. The software is designed to allow easy entry of numbers and formulas into the spreadsheet’s cells. The software automatically carries out the computations indicated by the formulas.

A spreadsheet is a simple but powerful idea. It allowed businesspeople to develop financial models of a proposed business activity quickly and easily. A person can ask “What if?” questions with a spreadsheet, exploring various alternatives. Modern spreadsheet software also has the ability to graph results by using a wide variety of built-in graphing routines. This adds a visual dimension to the original spreadsheets’ numerical feedback mechanism.

Looking back, it’s easy to see why the spreadsheet was so successful. The end user was empowered. An ordinary business person could learn to develop spreadsheet. It was no longer necessary to deal with the programmers who worked in the company’s management information system. It was no longer necessary to wait weeks or months to have software developed that could explore new ideas.

The spreadsheet also represents another really important idea. The business people who learned to use spreadsheet software already had a great deal of knowledge about business, business problems, and how to represent business problems with numbers and formulas. The spreadsheet brought this previous knowledge and these skills together in a way that it could be used to solve real, current problems.

**Educational Implications**

During the past 20 years, the spreadsheet has significantly changed the bookkeeping, accounting, and business curricula in high schools and colleges. We commonly expect that even beginning bookkeepers can develop and use powerful spreadsheet software.
Spreadsheet software is now built into integrated packages such as AppleWorks and into suites such as Microsoft Office and thus is available to tens of millions of computer users.

An introduction to spreadsheets is now considered an important component of the information technology standards being developed by various school districts, states, and the International Society for Technology in Education. Spreadsheet use can be taught to relatively young students, but it turns out to be far more than a simple task.

Here are the two important education-related ideas discussed earlier: First, many businesspeople learned to use spreadsheets quickly so they could build on what they knew about their businesses, bookkeeping, problem solving, and business math. They already knew the types of common business formulas—such as compound interest and amortization—that require extensive calculations. Second, businesspeople now use spreadsheets to develop models of proposed business situations and to explore answers to “What if?” questions. They have the business knowledge to interpret and use the results produced by spreadsheet computations.

My point is that businesspeople learned spreadsheets in the context of their current knowledge, and they were highly motivated because they had existing business problems they wanted to address and solve. Constructivism (or building on previous knowledge) and intrinsic motivation were the keys to their success. Contrast this condition with the typical middle or high school student who is exposed to spreadsheets for the first time. Most likely, the student lacks the background in modeling, calculation-intensive formulas, and problem solving—that is, the basic prerequisites to understanding spreadsheet use. The problem this student is learning to solve with a spreadsheet may not be personally or intrinsically motivating and may not require the use of such powerful software.

This presents a major challenge to teachers. The spreadsheet provides a superb environment for studying the representation or modeling of a wide range of problems, use of formulas for mathematical calculations, and problem solving. Problems can come from business, as well as from science, mathematics, the social sciences, engineering, architecture, and other academic disciplines. Developing a spreadsheet involves many of the ideas of computer programming—such as testing, debugging, and procedural thinking. Use of the graphing routines in a spreadsheet requires knowing effective ways to represent numerical data.

All of these ideas can be learned by teachers, although most of them need significant professional development to learn these ideas and how to integrate them effectively into curriculum, instruction, and assessment.

This provides yet another example of the crying need for further professional development and more curriculum materials for teachers. The spreadsheet is a powerful idea that contributed substantially to the microcomputer industry’s development. In the hands of a well-qualified teacher, the spreadsheet can contribute substantially to the education of students of all ages.

Over the years, Learning & Leading With Technology has carried many articles that detail how to use spreadsheets in the curriculum. We look forward to seeing examples from our readers on their own specific and integrated use of spreadsheets in their curricula.

Retrospective Comments 9/2/2008

This editorial captures the essence of the difficulty of getting routine higher-order (beyond amplification) uses of computers in schools. The higher-order (beyond amplification) uses tend to be quite domain-specific and require good knowledge of problem solving within the domain.
As a simple example, consider getting better at writing. Initially the conjecture was that if we provided students with a computer and word processing software, they would get significantly better at writing. That turned out to be an overly simple conjecture. Writing is a very complex problem-solving task. Some parts of the task can be isolated and instruction can be focused on them. Thus, writing instruction can focus on developing good, legible handwriting or printing. It can focus on spelling and punctuation. However, good writing requires careful and deep thinking in developing ideas to be communicated, and good understanding of the audience. It requires developing good skills in providing feedback to oneself and in doing the “revise, revise, revise” aspects of process writing.

Both desktop publication and writing interactive hypermedia bring new challenges to the writer. These are higher-order challenges, requiring considerable learning in order to move beyond simple amplification applications.