Changing Technical and Market Environments

Changes in computer hardware and software technologies and markets have shaped concerns about protection for computer software and ideas about what kinds of protection are needed.

Computer hardware technologies have changed dramatically over the past decade. With these changes have come important changes in how software is developed, sold, and used. Consequently, some software developers have modified their ideas about what aspects of software need the most protection. For example, as writing and checking lines of program instructions ("code") becomes more automated through computer-aided software development, some software producers propose to protect the *logic and idea* of a program, not just the effort required to write code and check ("debug") it. Others are concerned that computer-aided software development will make it easier to "disguise" copying.

Technological change also challenges traditional copyright concepts. For example, with developments in artificial intelligence and in interactive software and database systems, it will likely become increasingly difficult to draw the line between derivative works[1] and new creations, and to determine what constitutes "authorship."[2]

Twenty-five years ago, computers and software were not mass-marketed, retail items. The mainframes and minicomputers of the day were relatively few in number, compared to the number of microcomputers (PCs) in use today. These machines were run by expert staff using expensive, often custom-developed, software.[3] In the late 1960s, the "independent" software industry began to flourish. By 1988, U.S. independent software developers' revenues exceeded $25 billion, up from $20 billion in 1987.[4] About 40 percent of these revenues were from foreign sales.[5] Domestic revenues from all software and related services totaled over $50 billion in 1987, and were expected to increase to about $60 billion for 1988.[6] The United States currently commands a 70 percent share of the world software market.[7]

The fortunes of computer-software and computer-hardware developers are closely intertwined. A computer may gain popularity if plentiful and/or novel software is available. Conversely, lack of suitable application software (programs designed to perform specialized tasks for users) can be a barrier to the market success of a new computer or can limit the effective use of a computer.[8] Scarcity of application software can impede the use of a whole class of computers: software to make most effective use of massively parallel processors and other supercom-

---

[1] A "derivative Work" is a work based on one or more preexisting works (e.g., a translation, abridgement, or other form of transformation or adaptation). (See Title 17, U.S.C. 101.) Section 117 allows the rightful owner of a piece of software to make a copy or adaptation if the new copy or adaptation is for archival purposes (a "backup" copy) or is an essential step in utilizing the program in the computer.

[2] For some interactive software, it is increasingly difficult to determine where the programmer's expression ends and the user's contribution begins-the computer mediates and intermingles the creative efforts of both. Interactive computer-based works may generate new questions about ownership and originality. See discussion and example of a hypothetical interactive music-composition program, "Minstrel," in U.S. Congress, Office of Technology Assessment, *Intellectual Property Rights in an Age of Electronics and Information*, OTA-CIT- 302 (Melbourne, FL: Kreiger Publishing co., April 1986), pp. 70-73.

[3] Although some relatively sophisticated users (e.g., in universities or research organizations) did develop and maintain their own programs, most application software for specific tasks like inventory control or number crunching was either provided by hardware manufacturers or custom-developed under contract. Almost all operating-system software to run the computer and control its input, output, and logic functions was provided by computer-hardware manufacturers.

[4] Association of Data Processing Service Organizations (ADAPSO) figures on industry performance, 1989. These data for "non-captive" firms excludes the value of software produced in-house by hardware manufacturers; revenues are split about evenly between application and operating-system software.

[5] U.S. International Trade Commission, "The Effects of Greater Economic Integration Within the European Community on the United States," July 1989, ch. 4, p. 39.

[6] Computer and Business Equipment Manufacturers Association, *The Computer, Business Equipment, Software and Services, and Telecommunications industry, 1960-1996* (Washington, DC: CBEMA, Industry Marketing Statistics, 1987), table 4-3, p. 99.

[7] Commission of the European Communities, "Green Paper on Copyright and the Challenge of Technology-Copyright Issues Requiring Immediate Action," June 1988, pp. 171-172.

[8] Because software has become so critical t. so many industrial sectors, while productivity growth for software technology has been relatively slow, there is some concern that software could become a bottleneck-or the "Achilles heel of the information age." (Ian M. Ross, President, AT&T Bell Laboratories, keynote address, 1988 Bicentennial Engineering Conference, Sydney, Australia, Feb. 23, 1988.)

puters is currently scarce.[9] On another front, application-software developers can find the existing "installed base" of older computers and earlier programs (e.g., spreadsheet, database, or word-processing programs) a barrier to adoption of new programs designed for more advanced machines. They may also need to upgrade their products periodically; these new versions must be compatible with new hardware and also with older versions of the product.[10]

System software (programs, including operating systems, that make the computer usable and control its performance) can be an important factor in hardware firms' competitive strategies. For example, product competition in PC markets is based in part on differences in system *features (e.g.,* processing speed, ways of shipping data for processing in different parts of the computer, graphics capabilities) and *user-interface features (e.g.,* pictoral "icon," manual "mouse," or keystroke "macro" commands for functions such as moving the cursor or saving a file). These advantages are acquired from shrewdly mixing hardware and software designs.

When Congress created the National Commission on New Technological Uses of Copyrighted Works (CONTU) in 1974, the "PC revolution" had not yet begun to bring desktop computing power to the millions of individuals that now use it. By the time CONTU issued its final report in 1978, the PC revolution was under way, creating anew generation of computer users who were not primarily programmers or computer experts. The rapid proliferation of PCs in homes, offices, and schools created a very large retail market for application software-for word processing, spreadsheets, even games-as well as a lucrative market for PC operating-system software. In 1988, domestic revenues for PC application software reached almost $3 billion.[11] The widespread use of PCs also facilitated the growing use of online databases.[12]

Rapid growth and technological innovation made markets for PCs and PC software quite volatile, compared to the mainframe and minicomputer markets a decade earlier. Some new hardware and software firms would introduce ,new products, enjoy brief success, then go out of business within the space of a few years. Other firms built on early successes and went on to become industry leaders. A few years after introducing a successful product, however, they might find a substantial fraction of their potential market taken by competitors offering similar-sometimes improved—products, often at a lower price. The volatility of PC markets has focused new attention on questions about how best to provide intellectual-property protection for software, as well as hardware. At the same time, the history of the computer hardware and software industries illustrate the complex relationship between intellectual-property protection and stimulation of creativity.

---

[9] A recent press briefing by the Institute of Electrical and Electronic Engineers reported that while U.S. supercomputer manufacturers are focusing on new hardware developments to stay ahead of Japanese competitors, they are giving little attention to software to exploit the hardware's speed and power. As a result, a supercomputer's speed in solving problems may be only 1 to 2 percent of its advertised peak speed. ("Software SolutionJ'Science, vol. 246, No. 4930, Nov. 3, 1989, pp. 574-575.) See also: "The Computer Spectrum," *Computer*, vol. 22, No. 11, Nov. 11, 1989, pp. 61-62.

[10] Successive generations of upgrades tend to be increasingly complex. For example, one software developer's first database-management package had several thousand lines of code and took a single developer less than a year to create. The most recent version, designed to accept data files created under earlier versions of the package, has hundreds of thousands of lines of code and has taken a team of developers several years to create. (Ruthann Quindlen, "installed Base Becoming Obstacle toSoftware Companies' Success,'' *Infoworld*, vol. 11, No. 36, Sept. 4,1989, p. 82.)

[11] Ann Stephens, Software Publishers Association, personal communication, Oct. 2, 1989.

[12] An "electronic database" is a collection of information stored and accessed by electronic means. (Commission of the European Communities, "Green PaperonCopyright and the Challenge of Technology-Copyright Issues RequiringImmediateA ction," June 1988, p. 205.) Databases can be copyrighted as *works of compilation; the* copyright extends to the material contributed by the author of the compilation, or to the author's creative efforts in selecting, ordering, and arranging preexisting material, not to the preexisting material per se or ideas included in the compilation. Domestic revenues for on-line business databases alone amounted to $6.5 billion in 1988 (Information Industry Association data, 1989).