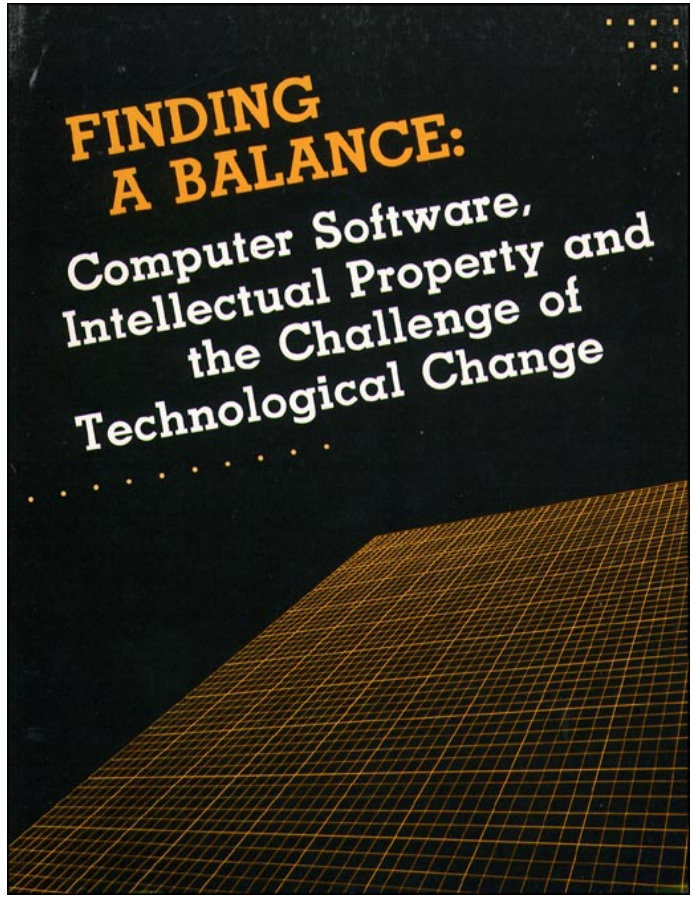
Finding a Balance: Computer Software, Intellectual Property and the Challenge of Technological Change

May 1992

OTA-TCT-527
NTIS order #PB92-169242



Recommended Citation:

U.S. Congress, Office of Technology Assessment, *Finding a Balance: Computer Software, Intellectual Property, and the Challenge of Technological Change, OTA-TCT-527* (Washington, DC: U.S. Government Printing Office, May 1992).

For sale by the U.S. Government Printing Office
Superintendent of Documents, Mail Stop SSOP, Washington, DC 20402-9328
ISBN 0-16-036188-5

Foreword

Our Nation's intellectual property system is intended to strike a balance between private incentives and protections of public interest. This report examines the rapid and complex technological changes and trends in computer software technologies and their possible effects on the Nation's intellectual property system. An effective policy must foster technological innovation to preserve economic competitiveness of the U.S. software industry in the face of changing technologies and markets.

The report identifies three policy issues: 1) the appropriate scope of copyright protection for computer software; 2) patent protection for software-related inventions and algorithms, and how the U.S. Patent and Trademark Office will handle these types of applications; and 3) complications facing libraries and commercial and private producers and users of digital information, including computer-based mixed media products.

This report was prepared in response to a request from the House Committee on the Judiciary. OTA drew upon work by staff and contractors and the comments of participants at six workshops. OTA received valuable assistance from members of the study's advisory panel, officials of the U.S. Copyright Office, U.S. Patent and Trademark Office, Congressional Research Service, Library of Congress, U.S. International Trade Commission, U.S. Department of Commerce, U.S. Department of Energy, U.S. Department of Defense, Office of the U.S. Trade Representative, and European Economic Community, as well as a broad range of individuals from law firms, trade associations, public interest groups, industry, and academia in the United States and abroad.

OTA appreciates the participation of the advisory panelists, workshop participants, Federal agency officials, and interested citizens, without whose help this report would not have been possible. The report itself, however, is the sole responsibility of OTA.



JOHN H. GIBBONS
Director

Finding a Balance: Computer Software, Intellectual Property, and the Challenge of Technological Change Advisory Panel

Michael S. Mahoney, *Chairman*
Professor of History, Princeton University

F. Christian Byrnes
Vice President, Product Planning
Landmark Systems Corp.

Dana E. Cartwright III
Director, Academic Computing Service
Syracuse University

Bill Curtis
Director, Software Process Program
Software Engineering Institute

Randall Davis
Associate Director, Artficial Intelligence Laboratory
Massachusetts Institute of Technology

Walter J. Doherty
Program Manager, Scientific Systems Support Laboratory
Computing Systems Department
IBM T.J. Watson Research Center

Carol Edwards
(formerly) Project Director
Southern Coalition for Educational Equity

Nick A. Farmer
Director of Information Systems
Chemical Abstracts Service

John Gage
Director of Science Office
Sun Microsystems

Morton David Goldberg
Partner
Schwab Goldberg Price & Dannay

Robbie Hardy
Senior Principal
American Management Systems, Inc.

Mitchell D. Kapor
chairman
ON Technology, Inc.

Ronald S. Laurie
Partner
Weil, Gotshal & Manges

Clifford A. Lynch
Director of Library Automation
University of California

Anita Mergener
Director, Product Design
Kelly Services

Jerome H. Reichman
Professor of Law
Vanderbilt University

Richard Rubinstein
Manager, Technology Assessment and Planning
Digital Equipment Corp.

Robert Schware
Senior Information Technology Specialist
The World Bank

Mary Shaw
Professor of Computer Science
Carnegie Mellon University

NOTE: OTA appreciates and is grateful for the valuable assistance and thoughtful critiques provided by the advisory panel members. The panel does not, however, necessarily approve, disapprove, or endorse this report. OTA assumes full responsibility for the report and the accuracy of its contents.

Finding a Balance: Computer Software, Intellectual Property, and the Challenge of Technological Change
OTA Project Staff

John Andelin, *Assistant Director, OTA*
Science, Information, and Natural Resources Division

James W. Curlin, *Telecommunication and Computing Technologies Program Manager*

Project Staff

Joan D. Winston, *Project Director*

Karen G. Bandy, *Senior Analyst*

Paula J. Bruening, *Analyst*

Alan Buzacott, *Analyst*

Ann M. Hironaka, *Research Assistant*[1]

Mark D. Johnson, *Legal Intern*[2]

Administrative Staff

Liz Emanuel, *Office Administrator*

Karolyn St. Clair, *PC Specialist*

JoAnne Young, *Secretary*

Other OTA Contributors

Linda Roberts, *Senior Associate (SET Program)*
Chair, Workshop on Public Interest Issues

Kevin O'Connor, *Senior Analyst (BA Program)*
Chair, Workshop on Patent, Copyright, and Trade Secret Law

Contractors

Frank W. Connolly, EDUCOM

Robert Echols, Consultant

Steven W. Gilbert, EDUCOM

Paul C.B. Liu, University of Washington

[1]Until August 1990.

[2]Catholic University of America (fall semester, 1990).

Contents

	Page
Chapter 1. Summary, Issues, and Options	3
Chapter 2. The Law	39
Chapter 3. The International Arena	93
Chapter 4. Software Technology and the Law	125
Chapter 5. Digital Information and Copyright	161
Chapter 6. Economics, Intellectual Property, and Software	183
Appendix A. Selected Computer Hardware and Software Initiatives Overseas	205
Appendix B. The European Economic Community Treaty: Structure and Function	213
Appendix C. Workshop Participants and Reviewers	216
Index	223

Chapter 1

Summary, Issues, and Options

Contents

Page

summary 3
Background and Study Approach 3
The Law 12
The International Arena 13
Software Technology 17
Digital Information and Copyright 19
Economic Perspectives 20
Issues and Options 22
Technological Challenges for Copyright Law 22
Technological and Institutional Challenges for the Patent System and PRO 23
International Dimensions 25
Digital Information and Copyright 26
Software Technology, Industry Structure, and the Future 26
Policy Choices and Options 28
Preparing for the Future 36

Boxes

Box *Page*

l-A.Recompilation 7
l-B. Patent Problem of Prior Art 8
l-C. Patent Examination Quality and Speed 9
l-D.The Software Debate: Some Stakeholder Groups and Their Concerns 10

Chapter 1
Summary, Issues, and Options

Summary

Background and Study Approach

When a new technology impinges upon the intellectual property system, questions arise concerning both the scope and appropriateness of its protection and the effectiveness of the system in promoting "the Progress of Science and useful Arts."[1] A series of OTA reports has explored the intellectual property challenges presented by new information technologies—particularly by the move to electronic representation of information and the proliferation of digital means of transmission, adaptation, and copying-and by biotechnology. Looking across these challenges, it would appear that, although the technological, economic, social, and industrial particulars differ, many of the tensions and issues that arise are general, rather than technology-specific.

In 1984, the House and Senate Committees on the Judiciary requested that OTA examine the impact of recent and future advances in communication and information technologies on the intellectual property system. In response, OTA prepared the 1986 report, *Intellectual Property Rights in an Age of Electronics and Information.*[2] That report found that technological changes in information technologies offer opportunities for social and private gain, but at the same time challenge existing business practices and legal doctrines. It examined the impact of new technologies on the effectiveness of intellectual property rights, including the right to control reproduction of copyrighted works, the right to control publication and performance of works, and the right to control the making of derivative works.[3] The report identified three types of information-based products—work *of art, works offact, and works of function-and* concluded that basic differences among these types of works create difficulties for the current intellectual property regimes.[4]

A 1989 OTA report, New *Developments in Biotechnology: Patenting Life-Special Report,*[5] examined challenges that biotechnology-specifically, the patenting of nonhuman living organisms—poses for the patent system (see box 2-A in ch. 2). Many of the questions and difficulties identified in that report (e.g., questions concerning the appropriateness and scope of patent protection, the newness of the technology, institutional difficulties in establishing a repository of prior art and in administering patent prosecution) have also been noted in this study.[6] Thus, it appears that-although the particulars differ—many current questions concerning the

---

1 U.S. Constitution, art. I, sec. 8.

2 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).

3 For example, technologies that lower the cost and time required to copy, transfer, or manipulate information and intellectual property can make works more accessible, make them more valuable to consumers, and make using them more convenient. But, these technologies can also make enforcing intellectual property rights more difficult, potentially reducing financial incentives to produce new works. For more discussion of technological change and the enforcement of intellectual property rights, including impacts on print, music, video, and other media, see *Intellectual Property Rights inan Age of Electronics and Information,* op. cit., footnote *2,* pp. 97-123.

4 *Intellectual Property Rights in an Age of Electronics and Information,* op. cit., footnote 2, pp. 64-68 et seq. Among other findings, the 1986 report identified particular difficulties with respect to achieving the public-policy objectives of our intellectual property system when applying it to computer programs (as works of function) and computer database systems (at the intersection of the factual and functional domains). (Ibid., pp. 67 and 78-88.) These included questions regarding the appropriate scope of copyright protection for programs (e.g., how ''expression'' is interpreted), patent protection for computer processes, and reverse engineering.

5 U.S. Congress, office of Technology Assessment, *New Developments in Biotechnology: Patenting Life—Special Report,* OTA-BA-370 (Washington DC: U.S. Government Printing Office, April 1989).

6 In many respects, OTA has observed that the difficulties associated with the accommodation of software-related inventions within the Patent regime are similar to those experienced in biotechnology. There are some historical differences, though. For biotechnology, the industry had been in general agreement concerning the desirability of patent protection and the major controversy within the industry was over the scope of such patents. (See footnote 5.) In software, OTA has observed that (in addition to concerns about scope) software developers' opinions are somewhat divided concerning the general desirability of patents for software-related inventions.

More recently, however, controversy over biotechnology patents has renewed, this time concerning the subject matter, as well as scope, of patents for gene sequences. This controversy (in industry, academia, and government) was prompted by the National Institutes of Health's announcement that it planned to file patent applications for thousands of complementary DNA sequences, even those whose function is unknown. See Leslie Roberts, ''Genome Patent Fight Erupts,'' *Science,* vol. 254, Oct. 11, 1991, pp. 184-186.

availability, scope, and administration of patents for computer processes and algorithms are not unique to software technology; rather., they are general questions that may arise when any new technology is introduced into the patent system.[7]

Another 1989 OTA report, *Copyright and Home Copying: Technology Challenges the Law*,[8] examined noncommercial, private use of copyrighted works and the implications of digital media and recording technologies, particularly for home audio recording. That report found that intellectual property laws serve to define the boundaries between *permissible* and *prohibited uses* of works; technology, driven by the social and economic objectives of its users, defines the frontiers of *possible* uses and feasible enforcement of boundaries.[9] OTA found that technological changes and trends[10] that substantially alter the nature and extent of possible uses, or the feasibility of enforcing prohibitions against certain uses, give rise to tensions between users and proprietors and may make modification or clarification of the law desirable.[11] In some cases, OTA found, new technologies (e.g., copy protection) may have the effect of a de facto change in the law.

In early 1989, the Subcommittee on Courts, Intellectual Property, and the Administration of Justice of the House Committee on the Judiciary (now the Subcommittee on Intellectual Property and Judicial Administration) requested OTA'S assistance in considering the issues related to the protection of computer software in a changing international marketplace. OTA prepared a background paper, *Computer Software and Intellectual Property*,[12] that examined current application of copyright, patent, and trade secret laws to computer software. It also provided an overview of the often conflicting views and concerns of various stakeholders among industry, academia, and the public at large.

Many of the tensions concerning intellectual property protection for a new technology are not necessarily technology specific.

Current Study Approach

In its prior work, OTA had identified several factors that contribute to the complexity of the software debate: 1) the nature of software technology itself, which makes it difficult to fit software into the current framework of copyright and patent law; 2) the rapid pace of technological change in computer hardware and software; 3) difficulties in reconciling cultural and definitional differences between the legal and technical communities; and 4) complications arising from the international scope of software markets and technologies. 13 In planning

[7] However, software is presently distinct in that multiple modes of legal protection can apply simultaneously to portions of a manifestation of the technology+. g., copyright for a program containing trade secrets, patentable software processes contained in a copyrighted program, etc.

Questions of patent scope and administration are not new—for a discussion of similar concerns in the 19th and early 20th centuries, see Fritz Machlup, *An Economic Review of the Patent System,* Study of the Subcommittee on Patents, Trademarks, and Copyrights of the Committee on the Judiciary, United States Senate, Study No. 15 Pursuant to S. Res. 236, 85th Congress, 2d Sess. (Washington, DC: U.S. Government Printing Office, 1958), ch. IV.

[8] U.S. Congress, Office of Technology Assessment, *Copyright and Home Copying: Technology* Challenges *the Law*, OTA-CIT-422, (Washington DC: U.S. Government Printing Office, October 1989).

[9] These legal bounds are based on the Constitutional intellectual property bargain, tempered by the feasibility and efficiency of enforcement. See *Copyright and Home Copying:Technology Challenges the Law*, op. cit., footnote 8, ch. 2 and ch. 3.

[10] The technological trends identified in the 1989 report were: 1) the movement to *digital representations* of music, video, and Other types of entertainment and information available to consumers; 2) the *erosion of niche boundaries* used to categorize copyrightable works according to their content (e.g., audio, video, computer software) or physical format (e.g., audiotape, videotape, computer disc); 3) the *emergence of new delivery infrastructures* to deliver information and entertainment (e.g., fiber optic cable, interactive cable services); and 4) the efforts of some copyright proprietors (e.g., in sound recordings and motion pictures) to develop and implement *technical means for copy protection.*

[11] Similar technological trends are relevant *to* the present study, especially the blurring of niche boundaries and the emergence of new infrastructures for delivering computation and for interacting with software and hardware. When stored or executed in a machine, software and data are in "digital" representations and thus can be interacted with, copied, or manipulated easily and efficiently. At the same time that new delivery infrastructures such as high-speed networks are being deployed, important new technologies like hypermedia, virtual reality, and scientific visualization are blurring content-based niche boundaries.

12 U.S. Congress, Office of Technology Assessment, *Computer Software and Intellectual Property*, OTA-BP-CIT-61 (*New* York, NY: Stockton Press, 1990). The background paper was released at the subcommittee's second day of oversight hearings (Mar. 7, 1990) on the topic of computers and intellectual property, at which OTA presented testimony. OTA had previously submitted a staff paper, ''Intellectual-Property Protection for Computer Software,'' to the subcommittee to assist in preparations for the first day of oversight hearings on that topic (Nov. 9, 1989).

[13] See Computer *Software and Intellectual Property*, op. cit., footnote 12, ch. 1.

and carrying out this study, OTA'S objectives were to:

- understand the characteristics of software as a technology, as well as identify the relevant technological changes and trends that will confront Congress;
- explore the relationships between the legal protection of software and incentives for innovation;
- understand the market, trade and policy implications of the emerging global software indus- V ;
- identify current intellectual-property challenges presented by software and computing technologies and anticipate future challenges from technological developments in computer software, operations, and architectures.

In the course of this study, OTA sought out the opinions, positions, attitudes and perceptions of the stakeholders in computer software protection, including individuals from academia and the research communities, the legal profession, the computer software and hardware industries, government agencies, and the public at large. This was accomplished through personal interviews and correspondence, and through public participation in the study's advisory panel and workshops. Each of the workshops focused on a specific set of issues or perspectives:

- Software Engineering Technology and Intellectual Property Issues (Sept. 24, 1990);
- Software-Developer Issues (Sept. 25, 1990);
- Public-Interest Issues (Dec. 7, 1990);
- Digital Libraries, Electronic Publishing, and Intellectual Property (Feb. 11, 1991);
- User Interface Technologies and Intellectual Property (Apr. 18, 1991);
- Patent, Copyright, and Trade Secret Protection for Computer Software (June 20, 1991).

In each of these workshops, as in the overall study, OTA sought to explore the dimensions of the software debate by examinin g software technology and its distinctive characteristics, asking questions such as:

- *What are the characteristics of the technology? How does it advance?*
- *What aspects of the technology are most important to society? To a proprietor?*
- *What might a proprietor want to secure rights for? Why? What are the private and public ramifications of granting or not granting these?*
- *How do these private and public objectives relate to current law? Do existing legal modes provide appropriate protection? Can they be implemented effectively and efficiently? If not, what might be done?*

This sequence of examination was not always possible. The published literature and the usual terms of the software debate tend to focus *on positive analyses 14 of Current law and* case law (e.g., whether copyright encompasses program command structures, whether certain computer algorithms are patentable subject matter, etc.) rather than on *normative analyses* of what is socially desirable and how that might be accomplished (e.g., what aspects of a program are valuable? how might a software developer obtain and preserve competitive advantages? to what extent should the law permit this?). This tendency is understandable and pragmatic. It reflects a natural reluctance to speculate (perhaps pointlessly) on hypothetical changes in the law or to propose changes too readily or too specifically and risk doing harm, rather than good.

The rapid pace of technological change in computer hardware and software complicates the "software debate."

Evolution of the Software Debate

Throughout the 40-some years of modern programming, computer software has not seemed to fit as easily as computer hardware within the traditional intellectual property framework. Most intellectual property protection for software has come through copyright and trade secret laws, and some through patent law, but software developers and users, the courts, and policymakers have engaged in a continual attempt to sort out *what should or shouldn' t be*

14$ 'positive analysis'' refers to an analysis of *what is*. ''Normative analysis'' is concerned with *what ought to be. In* this context, for example, focus on whether existing law can be interpreted as protecting program command structures would be part of a positive analysis. Focus on whether the program command structure *should* be protected to meet public-interest objectives would be part of a normative analysis.

protected (from a social perspective) and *what is or isn't protected* (according to current law).

By the mid-1970s, this ''software debate' helped motivate Congress to mandate the National Commission on New Technological Uses of Copyrighted Works (CONTU) to consider the question of how best to treat software. CONTU'S recommendation that copyright protection be explicitly extended to computer programs was reflected in the 1980 amendments to the Copyright Act.[15] But the debate was not put to rest, particularly with regard to the appropriate *scope* of copyright protection. In its 1978 report, CONTU had recognized certain difficulties in applying copyright to software, especially in distinguishing between the copyrightable "expression'' in a program and the processes or ideas the program implements, which are not copyrightable.[16] CONTU assumed that most copyright infringements in the then-immediate future would be ' 'simply copying, ' ' but recognized that technological advances would raise more difficult questions in determining the scope of copyright.[17]

One such question concerns "reverse engineering' ' of copyrighted programs, especially when it involves translation of object code into higher-level languages. This process is often referred to as "recompilation" (see box 1-A and ch. 4). Discussion of reverse engineering and recompilation brings together a number of copyright issues, including: whether it should be a copyright infringement to read/study a copyrighted digital work in order to extract noncopyrightable subject matter;[18] the extent to which fair use applies to unpublished works; whether the combination of copyright and trade secret laws should be used to achieve protection for noncopyrightable subject matter (ideas, processes, etc.) in copyrighted programs.

As software technologies and markets evolved and grew, so did the controversy concerning appropriate protection for computer programs, computer processes (implemented in software), and algorithms. Since 1981, there have been increasing numbers of patent applications and patents granted for *software-related inventions.*[19] *Over the* past decade, patents have been issued for software-related inventions such as linear-programming algorithms, spell-checking routines, logic-ordering operations for spreadsheet programs, brokerage cash-management systems, and bank college-savings systems. Patent litigation involving software-related inventions and controversies concerning patents for algorithms have become highly visible.[20] These causes of action and invention-specific controversies have focused attention on the appropriateness of patent protection for software-related inventions and algorithms, which present significant problems for patent-system administration. These problems include the incomplete stock of "prior art' available

15 Copyright Act of 1976, 17 U.S.C. 101 and 117. See also ch. 2.

16 *Final Report of the National Commission on New Technological Uses of Copyrighted Works (CONTU),* July 31, 1978, pp. 18-22. (Referred to by OTA as *CONTU Report.)*

17 CONTU concluded, however, that these questions should be answered on a case-by-case basis by the Federal courts (ibid., pp. 22-23).

18 "Ra~g~ ' or ''studying' a copyrighted work has never, in itself, been a copyright violation. It is only when analysis involves (perhaps requires) the making of a' 'copy' of the work—usually as an intermediate step in producing a competing work which may or may not be ' 'substantially similar' and therefore infringing-that the legitimacy of analysis to reverse engineer comes into question.

19 In this report, OTA sometimes uses phrases like 'patents for software-related inventions'' or ''software-related patents" to refer generally to patent protection for inventions implemented in software. (See discussion in ch. 4.)

The U.S. Patent and Trademark Office (pTO) considers terms like "software patents" to be misnomers because they may be interpreted to mean that a computer program per se (i.e., the code) is patentable, as opposed to the underlying computer process. The PTO position is that computer programs per se are not patentable, as opposed to patentable computer processes and algorithms that do not fall into the subject-matter exception for 'mathematical algorithms." (M. Keplinger, G. Goldberg, and L. Skillington, PTO, letter to Joan Winston, OTA, Dec. 18, 1989.)

20 An ''algorithm'' is a well-defined computational procedure for taking an input and producing an output. Algorithms are tools for solving computational problems-an algorithm describes a specific computational procedure for achieving a desired input/output relationship (see ch. 4).

In the United States, certain types of computer-implemented processes and algorithms can be patented. The U.S. Supreme Court has not ruled on whether computer programs per se are patentable subject matter, but has ruled that computer-implemented algorithms that are deemed "mathematical algorithms' per se are *not* statutory subject matter. Federal courts have thus held that a computer processor algorithm is statutory subject matter unless it falls within a judicially determined exception like the one for "mathematical algorithms.'

Currently, PTO patent examiners carry out a two-part test for mathematical-algorithm statutory subject matter; the test is intended to be consistent with legislative history and case law. For examination purposes, ''mathematical algorithms" are considered to refer to "methods of calculation mathematical formulas, and mathematical procedures generally,' and no distinction is made between manmade mathematical algorithms and mathematical algorithms representing discoveries of scientific principles and laws of nature, which have never been statutory subject matter. (U.S. Patent and Trademark Office, "Patentable Subject Matter: Mathematical Algorithms and Computer Programs, " 1106 O.G. 4, Sept. 5, 1989; also contained in Michael S. Keplinger and Ronald S. Laurie (eds.), *Patent Protection for Computer Software: The New Safeguard* (Englewood Cliffs, NJ: Prentice Hall Law and Business, 1989), pp. 9-42.)

Box l-A—Decompilation

There are three different types of programming languages: *machine language, assembly language, and high-level language. Machine* language programs can be executed directly by the computer, but are relatively difficult to write and understand. Assembly language programs and high-level language programs are easier to write and understand, but cannot be executed directly by the computer. For this reason, programs are usually first written in assembly language or a high-level language, and then translated into machine language so that they can be executed on the computer.

Programs are typically distributed in machine language form. Machine language programs do not have to be assembled or compiled by the user; they are ready to be loaded into the computer and executed. In addition, distribution in machine language form has the side effect that it is difficult for others to look at the program code and understand how the program works. This can help to keep secret those elements which give the program a competitive advantage.

The "recompilation'' issue has arisen because efforts to translate a machine language program back into a more understandable form, such as assembly language or high-level language, may be a copyright infringement because the translation process would involve the making of an unauthorized copy or derivative work. Legal scholars are divided on the question of whether this activity can be excused under the provisions of Section 117 or Section 107 of the copyright law.

Two terms are used to refer to the process of translating a machine language back into a more readable form. *Disassembly* is the process of translating a machine language program into an assembly language program; *recompilation* is the process of translating a machine language program into a high-level program. One issue in the policy debate has been the feasibility of recompilation. There are currently no commercially available decompiler. It appears that the term 'recompilation,' as it is used in the policy debate, encompasses disassembly and any other procedure by which a machine language program is translated into a more understandable form. There area number of disassembler programs available on the market. Translating a machine language program into assembly language is much easier than translating it back into a high-level language.

One view is that limits on recompilation are required in order to encourage the development of original programs. Those who take this position argue that recompilation significantly lowers the cost of implementing "clone" programs. They claim that the original program is decompiled, altered to disguise the copying, and marketed. The clone program can then be sold at a lower price, taking away market share from the original developer, and reducing incentives for the development of new programs.

Others argue that recompilation is a difficult and time-consuming process that does not significantly reduce the cost of developing clone programs. A large disassembled program takes a great deal of effort to understand. In addition, they emphasize that recompilation is required for a variety of other purposes, many of which have a less direct impact on the developer of the program being decompiled. (For more discussion, see ch. 4.)

SOURCE: OTA, 1992.

to patent examiners in evaluating patent applications for processes involving software and algorithms, and the long timelag between patent application and issuance, compared to fast-moving software life cycles (see below and boxes 1-B and l-C). Moreover, some members of the software and legal communities believe that software-related patents will tend to stifle, rather than encourage, technological progress.

Copyright and patent lawsuits have continued to test and explore the boundaries of the current laws.

An incomplete stock of prior art can present significant problems for patent-system administration.

Looking at the scope of current legal interpretations and at possible uncertainties in these laws, some have proposed that modifications to existing structures, or the development of *sui generis* protections[21] are preferable to forcing software to fit

[21] *Sui generis* is a Latin phrase describing a law that is "of its own kind or class.' The Semiconductor Chip Protection Act of 1984 (Public Law 98-620) is a *sui generis* law for chip mask works; it is not part of the patent or copyright laws.

Box l-B—Patent Problem of Prior Art

The quality and availability of the *published (as* opposed to *product)* prior art, or known technology, is often cited as affecting the quality of issued patents (see ch. 2). The U.S. Patent and Trademark Office relies upon its database of prior art to determine whether the invention defined in the patent application meets the patentability criteria set forth in statute. It is against this collection of prior art literature (including earlier patents) that the PTO compares the claimed invention and decides whether the claimed invention possesses the requisite novelty and nonobviousness.

Among the reasons cited for the perceived problem of prior art is the extensive use of trade secret protection for computer software. Unlike patent and copyright protection, trade secret does not require disclosure of information that is the subject of protection. Rather, trade secret requires that the holder of the trade secret make a deliberate effort to maintain the secret quality of the information. Such secret information cannot, by definition, function as part of the "known" technology available as a standard for patentability as required in the patent law. In the course of development of the computer sciences, some advances in the field were published in journals and industry communications, most especially within the academic community. However, many new innovations were not published because they were simply embodied in a product or not considered the type of "invention" which would be the subject of a publication. The prior art that is not the subject of a patent is not always considered to be as rich in the computer science field as in other disciplines. As a result, in large part, the resources available to the PTO for determining obviousness and novelty are more limited than in other fields; in large part the prior art database is limited to software that is already the subject of other patents for software-related inventions. It is therefore often difficult to determine what can be considered the current state of the technology and what can be considered, in the words of the patent statute, ''obvious to the ordinary person skilled in the art' for purposes of determining patentability.

The patent law does not. provide for a free system for third parties to add to the present general stock of prior art. However, it does allow parties to submit art that maybe pertinent to the patentability of particular *issued* patents. As a result, it has been suggested that a private database of prior art be developed by the industry itself, which would allow for free contribution of prior art in an effort to improve the quality of the prior art database available to the PTO and, consequently, of the patents issued.

SOURCE: OTA, 1992.

models more suitable to other types of works and discoveries. However, the majority of legal experts and firms in the industry takes the position that existing structures like copyright and/or patent are adequate to deal with software, that the case law as a whole is evolving appropriately, and that *sui generis* approaches risk obsolescence as the technology changes and lack an established treaty structure providing international protection (e.g., the Berne Convention provides reciprocal copyright protection in over 75 countries). Thus, their tendency is to try to find some way to accommodate specific aspects of software-like protection of user interfaces—within existing structures, particularly copyright. *In OTA'S view, despite the advantages, there are questions as to whether this process of accommodation can-or should---continue indefinitely. With respect to software, there may be a point where it becomes preferable to complement or substitute for the existing structures, rather than extend the scope of copyright to fit certain aspects of software—perhaps, cumulatively, at the expense of other types of works. In* continuing to assess the intellectual property bargain, Congress may conclude that the "balance" for software differs somewhat from that for other copyrighted works.[22]

The stakeholders in the software debate can be categorized in many ways—software creators, soft-

22 For example, see testimony on the varying concerns of software developers, journalists, and historians regarding fair use of unpublished works at hearings before the House Committee on the Judiciary on H.R. 2372 ("Copyright Amendments Act of 1991"), May 30, 1991 and June 6, 1991.

U.S. law provides that "original works of authorship" are copyrightable subject matter (17 U.S.C. 102(a)). Computer programs are considered to be in the category of "literary works," which are: "works, other than audiovisual works, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material object, such as books, periodicals, manuscripts, phonorecords, film, tapes, disks, or cards, in which they are embodied, " (17 U.S.C. 101).

Box l-C—Patent Examination Quality and Speed

The prolonged pendency period for patents between time of application and time of issuance has also been of concern in light of the fast-moving nature of the field of the technology.[1] At present, the U.S. Patent and Trademark Office target for pendency is 18 months. The lower range of estimates of the pendency period for software-related inventions is from 18 months to 2 years; some believe that it is more like 32 months from filing to issuance.[2] Stakeholders concerned with software development find this pendency period alarming, given the rapid pace at which the technology advances. They cite the possibility of "landmine patents,' patents which have been pending in the PTO to issue only after others have in the interim unknowingly developed infringing software products. The issuance of such a patent thereafter precludes the making, using, or selling of the software by anyone other than the patent holder.

Also of particular concern is the question of criteria for subject matter patentability under Section 101. Courts and the PTO have struggled with the question of patentability under Section 101 since the late 1960s, and the rapidly advancing nature of the technology forces that debate to continue (see ch. 4). Recently applicants have complained of a proliferation of Section 101 rejections from the PTO, causing some to conjecture that the PTO is implicitly asking the Board of Appeals and the Court of Appeals for the Federal Circuit to issue new rulings on the question of patentable subject matter.

1 See discussion in ch. 2.

2 Robert Greene Sterne and Edward J. Kessler, "Worldwide Patent Protection in the 1990s for Computer Related Technology,'' in Morgan Chu and Ronald S. Laurie (eds.), *Patent Protection for Computer Software* (Englewood Cliffs, NJ: Prentice Hall Law and Business, 1991), p. 359.

SOURCE: OTA, 1992.

ware users, large and small commercial software developers, computer hardware manufacturers, educators, students, academic and other software and computer science researchers, to name just a few (see box l-D). Sometimes issues in debate are characterized as conflicts between software producers and consumers, between large and small fins, between major firms and their smaller competitors, between commercial and academic/nonprofit software developers and researchers, or between industry and the general public. Although these characterizations can be helpful in understanding specific issues and positions, one must be cautious about overgeneralizing: for example, some aspects of the controversy over software-related patents are characterized along the lines "large firm versus small developer. ' While it is true that large firms, on average, are more likely to have greater financial and legal resources and more expertise dealing with the patent system, licensing, and litigation, some small firms and entrepreneurs are advocates of patents for software-related inventions and find them extremely advantageous, particularly in attracting investments and in dealing with large competitors.[23] *OTA has found that the most general line of demarcation across stakeholders separates those who perceive significant current financial advantages under the status quo and/or who are relatively confident that their legal and financial resources are adequate to deal successfully with any legal uncertainties or litigation, from those who do not perceive significant financial advantages under the status quo (compared to possible changes or modifications) and/or feel less well-equipped to deal with legal uncertainties or litigation.*

Some well-publicized recent copyright lawsuits have raised issues, which are also being debated outside the courtroom, regarding how far the scope of copyright extends beyond the literal written expression—the program code-to the program's "design,' to the logic underlying a program, and to the program's command structure and interfaces (see ch. 4). *At stake in these decisions is the extent to which copyright (in concert with trade secret law) should be interpreted to give protection to the*

23 See, e.g., Paul Heckel, "Epilogue: The Wright Brothers and Software Invention, " *The Elements of Friendly Software Design*, 2d ed. (Alameda, CA: SYBEX, Inc., 1991), pp. 223-294; and Elon Gasper et al., "Vital to Small Companies" (letter to the editor), *The New York Times*, June 8, 1989, editorial page.

Box I-D—The Software Debate: Some Stakeholder Groups and Their Concerns

Individual software ***creators and the software industry-Creators of commercial software are*** concerned about their financial viability; an important rationale for intellectual property protection for software is to give commercial software developers adequate market incentives to invest the time and resources needed to produce and disseminate innovative products. The software industry in general is concerned with revenue losses resulting from commercial piracy and counterfeiting and many developers are also concerned about unauthorized end-user copying (see ch. 3). But direct revenue losses are not the only concerns of commercial developers, who also want to gain and maintain a competitive advantage in the marketplace.

One powerful source of market advantage is lead time: the first company out with an innovative computer program benefits from its head start. Trends in software technology, like computer-aided software development, are eroding leadtime advantages. Another potential source of a market advantage is user and/or machine interfaces. In this area however, the industry's goals of expanding the market and a firm's goal of maintaining or increasing market share can beat odds (see chs. 4 and 6).

Software developers, and the industry as a whole, are concerned with access to state-of-the-art knowledge and diffusion of information about programs and programming, so that programmers can build on each others' work, rather than reinvent the wheel (or rewrite a matrix-multiplication subroutine) for each new program. For society as a whole, the pace of innovation maybe speeded up if competitors are able to build on others' advances, rather than allowing an innovator to block others (see chs. 4 and 6). A related concern is reverse engineering of software, particularly for the purpose of understanding the internal construction and functioning of a program (see ch. 4).

Software users—Millions of individuals and thousands of businesses rely on purchased software products for their day-to-day activities and livelihood. As with any product, they care about the price, quality, functionality, ease of use, and variety of software available. Most users, especially business users who rely on software tools for day-to-day operations, are also concerned with the availability of expert support for questions or problems.

The "software workforce" who use and/or create software as part of their jobs want to have transferable skills; thus they are concerned, sometimes only indirectly, with standards for programming languages and external consistency of user interfaces (see ch. 4). (For example, learning a new wordprocessing package is easier if it has commands in common with other packages one already knows.) However, users also want more powerful software

functionality of the program, as opposed to the program code.[24]

Software-related patent suits are also ongoing. This litigation and the recent publicity[25] given to some patents for algorithms[26] have stimulated debate over whether computer processes and algorithms should be patentable at all, or whether they are different enough from other areas of technology that special limitations should apply. Although questions pertaining to patent-system administration are extremely important, the long-term question of whether patent (or patentlike) protection for computer processes and/or algorithms is socially desirable is separate from the related question of how well current U.S. Patent and Trademark Office (PTO) procedures are working now,

In 1990, the Secretary of Commerce established an Advisory Commission on Patent Law Reform. The commission was to examine 13 sets of issues regarding the patent system, including protection of

24 As Paul Goldstein put it:

> "...a computer program is quintessentially a functional work. As a consequence, even the most closely circumscribed definition of a computer program's protectable subject matter will to some degree enable the copyright owner to monopolize the program's function-its 'procedure, process, system, method of operation, concept, principle, or discovery.' Consequently, the task in defining the scope of a computer program's protectable subject matter is not to distinguish between nonfunctional and functional elements, since function will pervade all elements. Rather, the task in any case is to separate those elements-protectable expression—whose monopolization will not overly inhibit competitors' use of the program's functions, from those elements—unprotectable ideas-whose monopolization will improperly inhibit competitors' use of the programs's function. (Paul Goldstein, *Copyright—Principles, Law and Practice* (Boston, MA: Little, Brown and Co., 1989), sec. 2.15.2, pp. 206-207.)

25 See, e.g., Edmund L. Andrews, ''Equations Patented: Some See a Danger, " *The New York Times,* Feb. 15, 1989, pp. D1, D6; Jack Shandle, "Who Will Weather the Gathering Storm in the Courts?' *Electronics,* August 1989, pp. 67-70; and ' 'Lodging Securities at the Patent Office, ' *The Economist,* Aug. 25, 1990.

26 See footnote 20, supra.

with improved functions; sometimes the desire for consistent (''standard' interfaces conflicts with ease of use and improved functionality. The software-training and temporary-help industries share these workforce interests; they also are developers of training software.

Thus, software users care about the health of and level of competition in the software industry, as well as having ''common ground' (compatibility) that allows them to use new products with their existing hardware and software (see ch. 4). Users care about having "reasonable" rights (e.g., being able to make a backup copy of an expensive piece of software); some need the ability to modify 'packaged' software in order to use it efficiently or meet other specialized needs. Most businesses and individuals who use software tools to create other products or services want a stable and predictable legal environment so they know what uses are permissible and which are not or must be licensed from developers.

Academic community—Academic and research communities traditionally value free access to and exchange of information (see ch. 5). Academic software/computer science researchers and developers who are motivated by incentives other than commercial potential (e.g., professional prestige, tenure, publication in scholarly journals) tend to view intellectual property protection somewhat differently than commercial developers. Like many small software vendors, many in the academic community are concerned that what they consider to be "overprotection' (e.g., copyright protection for "look and feel' and patenting of software processes and algorithms) might hamper research and long-term growth in their fields (see ch. 4).

In contrast to development of major commercial software packages, "small" software programs to help teach students are developed by faculty in a number of disciplines. The incentives to develop, distribute, and use such ''small' software programs, which are often distributed over academic computer networks, differ significantly from those for commercial software.

As financial pressures mount, universities and their faculties are becoming increasingly interested in commercializing technology and appropriating financial rewards from their intellectual property. At the same time, they are concerned about affordability of the software that students need, both inside and outside the classroom and laboratory.

SOURCE: OTA (adapted from U.S. Congress, Office of Technology Assessment, *Computer Software and Intellectual Property OTA-BP-CIT-61 (New York, NY:* Stockton Press, 1990)).

what PTO terms ''computer-related inventions, ' as well as procedural matters such as a first to file system, automatic publication of applications, and the term of patent protection.[27] *The latter* questions reflect a concern about the differences between the U.S. patent system and those in foreign countries (see ch. 3). The World Intellectual Property Organization (WIPO) has attempted to harmonize patent laws in member countries. U.S. agreement to WIPO'S draft treaty would entail adjustments in U.S. law including a change from a first to invent to a first to file system (awarding the patent to the applicant who has the earliest filing date) and a change in the term of patent protection from 17 years from grant to 20 years from date of filing.

The judicial system, along with PTO and the Copyright Office, participates in the process of defining the bounds of software protection. The courts help determine requirements for—and scope of—protection under the patent and copyright systerns by addressing issues in the course of litigation, whether between parties in the private sector or between the respective government agency and applicants (see chs. 2 and 4). *Given the rapid advancement of the computer sciences, the courts face enormous challenges in resolving the issues raised by the changing technology adequately and in a timely fashion, so as to properly serve the needs of both the industry and society at large.*

Courts must deal with complex and fast-moving technologies.

The problem confronted by the judiciary is twofold. In addressing computer and software issues, courts must deal with technology that is highly complex; the court must find a means to understand

27 See *Federal Register*, vol. *56, No.* 95, May 16, 1991, pp. 22702-22706

extremely technical concepts in order to decide legal issues fairly. In addition, the generally overworked judicial system, with its crowded docket, must render decisions on a technology that often advances faster than such decisions can be reached. These two difficulties raise questions about the courts' current capabilities to address the issues presented by the software industry.

The Law

The U.S. ' 'intellectual property system' is a mixture of Federal and State law. Laws concerning copyright, patent, trademark, and the protection of semiconductor chip mask works are under Federal jurisdiction. Laws concerning trade secrets and the misappropriation of confidential business information, and certain limited kinds of ''unfair competition, ' are under State jurisdiction. Trademarks may be federally registered and/or registered with an individual State; trademark rights may also accrue based on common-law usage. Computer software is distinguished from most other intellectual creations protected by intellectual property law in that it is eligible for protection by patent, copyright, and trade secret laws. Each kind of protection possesses certain strengths and weaknesses, and each protects certain aspects of software in specific ways.

The statutory subject matter of a utility patent is limited to a process, machine, article of manufacture, or composition of matter that is novel, nonobvious, and useful, or to new and useful improvements to these classes of patentable subject matter. In exchange for a sufficiently detailed disclosure of the invention by the inventor, the patent precludes others from making, using, selling or importing components of the patented invention. A patent protects against independent creation, so that to prove infringement the patentee need not show that an invention was ' 'copied' or acquired through some improper access or means. A U.S. utility patent allows for 17 years of protection for the invention (including application of the underlying idea), during which time the patented invention may be licensed, publicly disclosed, and distributed without altering its legal protection.

Design patent protection is available for surface ornamentation, configuration, or a combination of both. While the configuration of a useful object may constitute a patentable design, a design dictated by considerations of function is not a proper subject for a design patent. Patent protection for designs is granted for a period of 14 years.

A patent protects against independent creation. To prove infringement, the patent holder need not show that an invention was "copied."

Whether and to what extent software-related inventions are the subject of utility patent protection has been an issue for consideration by the courts and the U.S. Patent and Trademark Office since the early 1960s. The U.S. Supreme Court has examined the issue of patentability of software on a number of occasions, in the cases of *Gottschalk v. Benson, Parker v. Flook,* and *Diamond v. Diehr,* attempting to delineate the limits of patentable subject matter with respect to ''mathematical algorithms. '

At the same time, the PTO has grappled with several institutional problems, including issues such as: examiner training and turnover, length of pendency periods (from filing to issuance) for patent applications, a backlog of applications, and the quality and extent of the prior art database (see discussion in this chapter, pp. 6-8 and more in-depth discussion in ch. *2). In OTA'S view, these problems are serious in that they may affect the quality of the patents issued and create additional burdens for software developers and users (e.g., 4''landmine' patents--see box 1-C).*[28]

U.S. law provides that until the patent is issued, the information contained in the application for a patent remains secret, and therefore may be protected as a trade secret. Information beyond that required for inclusion in the patent to meet the ''enablement' and ''best mode' requirements can also be reserved for trade secret protection. Trade secret law protects confidential business information against unauthorized use or disclosure, and is based on statutory and common law and contractual provisions.

28 As a matter of policy, PTO does not comment on the examination process for issued patents. Because OTA could not be "walked though" application of PTO examination criteria or discuss interpretation of the criteria for specific patents, OTA was unable to make any independent finding on the quality of examination for particular software-related patents.

Like patents, trade secret law can protect the underlying idea of an invention, rather than any particular expression. Trade secret possesses the distinct advantage that, unlike patents, this form of protection does not require any disclosure of information; indeed, trade secret protection is critically dependent on the secret nature of the information, and on the steps taken by the trade secret holder to maintain secrecy. Unlike patent holders, possessors of trade secrets have no protection against independent creations and even subsequent patenting by others of the invention that is the subject of trade secret. (For more on trade secrets, see ch, 2.)

Copyright law, unlike patent and trade secret, protects the expression of an idea rather than the underlying idea itself. Copyright does not extend to any procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied. Rather, copyright is said to protect the expression in the program-which may include such program elements as source code, object code, screen displays, etc. (see chs. 2 and 4).

The evolution of case law in copyright has involved examination of several key issues. Among the most important to software are whether object code as well as source code[29] is protected *(Apple v. Franklin);* whether a program's structure, sequence and organization are protected (e.g., *Whelan v. Jaslow, Plains Cotton Cooperative Association v. Goodpasture Computer Serv., Inc., et al.),* and what such protection implies; and whether the ''look and feel' of the program and its interface is protected *(Lotus v. Paperback Software, Computer Associates v. Altai, Inc.).*

Current law provides for copyright protection for unpublished as well as published works. This is important for computer software, because it facilitates simultaneous use of copyright and trade secret protections. The published version of the copyrighted program can be distributed as ''object code, usually in a machine language that is difficult to read or study. The 'source code, ' usually written in a higher-level, easier-to-understand computer language, remains unpublished and is often held as a trade secret in order to protect the program's logic and know-how (see below and ch. 4, especially the section on recompilation). However, if the ''ideas' of the program can be ascertained by inspection of the object code, trade secret in such ideas is lost. Also, if and to the extent that recompilation is not a copyright or contractual violation, trade secret protection for the source code can be lost.

Copyright protects the expression of an idea rather than the underlying idea itself.

The Semiconductor Chip Protection Act of 1984 (SCPA) extends legal protection to a new form of subject matter-semiconductor chip mask works—in order to address the problem of chip piracy .30 The act provides for a 10-year term of protection, and registration under the SCPA is administered by the Copyright Office. Reverse engineering is a defense to a claim of infringement under the act, and provides an exemption from infringement liability in spite of proof of unauthorized copying and striking similarity, so long as the resulting chip product was the result of study and analysis and contained technological improvement. The SCPA provides for remedies similar to those associated with copyright protection, does not allow for criminal penalties, and maintains a higher limit on statutory damages than that provided for in the Copyright Act.

The International Arena

The software industry has become global in character, leading to increasing international efforts to protect intellectual property rights in software. The global nature of the industry and the law is important because of the effect of commercial activities in foreign countries on those in the United States, as well as the similar effect of U.S. activities on those in other countries. In the area of software, as in all industrial and service sectors, companies compete in international and domestic environments. As a result, U.S. legal concepts, definitions,

29 The computer term ''source code' is often used to refer to a computer program in the language that it was written, usually a high-level language but sometimes assembly language. The term "object code" refers to a program in the form of machine language. See footnotes 57, 58 and 59 (and accompanying discussion) below and also ch. 4.

30 17 U.S.C., ch. 9. See also Robert W. Kastenmeier and Michael J. Remington, ''The Semiconductor Chip Protection Act of 1984: A Swamp or Firm Ground?' *Minnesota Law Review,* vol. 70, No. 2, December 1985, pp. 417470.

and policy about software and intellectual property protection for software affect (and are affected by) those of other nations. This report discusses the global software industry and the issue of piracy, and examines various treaties, negotiation efforts and attempts to harmonize domestic and international laws to provide protection for intellectual property.[31]

The Global Software Industry

Although its share of the world software market has declined over the past decade or so, the United States is still the world's leading innovator and producer of computer software.[32] Accurate data on software industry revenues and market shares are difficult to obtain, in part because there are many types of "software industry" data being collected and reported by different organizations.[33] These include data about:

- *software and services*, including processing and professional services, as well as software products;
- *application and systems software*, whether packaged or custom-developed;
- *packaged software*, including applications and systems software;
- *custom software*, professionally developed or extensively tailored to meet a customer's specific needs;
- *personal computer (microcomputer) software*, usually sold as packaged software (although not all packaged software is for microcomputers); and
- *software from "independent" developers* who are not part of a hardware manufacturer.

This variety of data, collected by different organizations, makes comparison and synthesis extremely difficult.[34] Consistency across types and years is usually not possible when drawing from these published figures.[35]

By ah-nest any measure, though, the United States has a premier role as a producer and a consumer of software. According to one industry estimate, U.S. demand accounted for 52 percent of world *software* consumption in the late 1980s.[36] The U.S. Department of Commerce estimated that global revenues from sales of *software* were more than $65 billion in 1989 and that U.S. software suppliers accounted for more than 60 percent of global software sales .37 According to the Software Publishers Association (SPA), North American revenues from *packaged software for microcomputers* (personal computers) were $4.5 billion in 1990, up 22 percent from 1989.[38]

According to the Computer and Business Equipment Manufacturers Association (CBEMA), in the

---

31 Examination of North-South and East-West technology transfer is beyond the scope of this report. For a treatment of global economic competition, with an emphasis on high technology, see U.S. Congress, Office of Technology Assessment *Competing Economies: America, Europe, and the Pacific Rim, OTA-ITE-498* (Washington DC: U.S. Government Printing Office, October 1991).

Although this study does focus some attention on the relationship between intellectual property and standards (see chs. 4 and 6), a detailed examination of standards is outside the scope of this report. For a thorough treatment of international standards, see U.S. Congress, Office of Technology Assessment, *Global Standards: Building Blocks for the Future*, OTA-ITE-529 (Washington, DC: U.S. Government printing Office, April 1992).

32 Studies in the late 1980s estimated that U.S. producers held a 70 percent share of the global market for software. Estimates by the U.S. Commerce Department now place the U.S. market share at around 60 percent (see ch. 2). Part of the decline in the U.S. shares of software demand and supply has come about naturally as software use becomes more widespread abroad and other nations' software industries develop.

33 For example, the Software Publishers Association (SPA) collects data on packaged PC software; ADAPSO (The Computer Software and Services Industry Association) reports data on software and services, usually (but not always) from independent mainframe and minicomputer software houses; the Computer and Business Equipment Manufacturers Association (CBEMA) reports data on the information technology industry, including office equipment, telecommunications, electronic data processing equipment, and software and services (including software produced by hardware manufacturers). Moreover, "hardware" companies rdso are software producers-sometimes, like IBM, the largest in the world.

34 Whenever possible, OTA has specified the type and source of market data and estimates (e.g., "software," "independent software," "software and services' '); the reader should not expect figures for a given year to "add up" or figures from different sources to be readily comparable.

35 For instance, a firm whose products include PC application may have at least some of its revenues included in ' 'w-application software," or ''packaged software;'' it maybe included in 'software and services, ' and may or may not be an 'independent' software house. But a firm whose main products are PC networking software is likely not to be included in data on ''PC-application software."

36 ADAPSO estimate in Jeff Shear, "Competitive Software Industry Suits Up for Global Hardball," *Insight*, July 10, 1989, p. 38.

37 Commerce Department estimate cited in *Keeping the U.S. Computer Industry Competitive: Defining the Agenda*, Computer Science and Technology Board (Washington DC: National Academy of Sciences, 1990), pp. 30-31.

38 Ken Wasch, Nicole Field, and Sara Brown, SPA, personal communication, July 30, 1991.

market for *software and services,*[39] the U.S. industry had domestic revenues of about $93 billion in 1990, a 16 percent increase from 1989 revenues of about $80 billion. Of these domestic revenues, CBEMA estimates that software products accounted for about 45 percent of the total--$42.5 billion in 1990 and $35 billion in 1989.[40] (See ch. 3 for more industry revenue estimates.)

U.S. producers are increasingly challenged by competition from developing software industries abroad, particularly in Europe, where U.S. firms currently hold 70 percent of the PC-software market.[41] With the prospect of a unified market and common standards in Europe in 1993, U.S. firms are facing new competition from Japanese software producers who are establishing themselves in Europe through acquisitions, as well as invigorated competition from European vendors. The United States faces growing competition in Asia from Japanese producers, while software industries in Taiwan and Korea are developing rapidly. And in the United States, U.S. firms face new competition in the domestic market from foreign competitors like the Sony Corp. (see app. A for more on overseas markets and technology initiatives).

Although U.S. software developers face increasing competition from foreign competitors, the United States still has a premier role as a software producer.

Software Piracy

Illegal copying of software results in financial losses to U.S. software firms both directly, through loss of sales and/or royalties, and indirectly, through loss of investment opportunities.[42] Retail piracy—duplication of an entire program for sale by ''pirate' competitors-and counterfeiting are major concerns of most software companies.[43] *These concerns can be dealt with straighforwardly, at least in theory, by copyright law.*[44] In practice, *enforcement, especially overseas, is difficult. (See* below and ch. 3 for discussion of international treaties and agreements concerning intellectual property and software.)

Estimates of financial losses due to piracy vary. ADAPSO (The Computer Software and Services Industry Association) estimates that one of every two copies of personal-computer software used by corporations in the United States is an illegal copy.[45] In 1990, according to SPA estimates, developers of packaged PC software lost $2.2 billion to piracy within the United States,[46] up from an estimated $1 billion in 1986.[47] Industry estimates of losses from piracy abroad are larger: the Business Software Alliance (BSA) estimates that-looking at *all types of* software-software piracy worldwide causes the

39 *OTA note:* Revenue reported for 'software and services" includes revenues from processing and professional services, as well as from custom and packaged software products.

40 Oliver Smoot, CBEMA, personal communication, June 30, 1991. See also CBEMA, *The Computer, Business Equipment, Software and Services, and Telecommunications Industry, 1960-2000* (Washington, DC: CBEMA, Industry Marketing Statistics, 1990), p. 100. (Estimates from BDA Assoc. forecast.)

41 SPA estimate (see ch. 3).

42 For discussion of revenue losses due to piracy, see U.S. International Trade Commission, "Foreign Protection of Intellectual Property Rights and the Effect on U.S. Industry and Trade, " February 1988, ch. 4.

43 *OTA note:* This text uses the phrase ''re~l piracy'' t. mean unauthorized copying for the purposes of selling the illegal copies or close derivatives; "counterfeiting" to mean passing off illegal copies as the real thing; "end-user piracy" to mean copying by users but not to sell the copies.

44 Jerome Reichman notes that Anglo-American law tends to use copyright to redress ''pi~cy' (i.e., slavish imitation) because these countries lack a general-purpose unfair competition law on the European model. Reichman considers that more attention needs to be paid to repression of piracy through international norms of unfair competition law. (Personal communication Sept. 17, 1991.) See Jerome H. Reichman, *Proprietary Rights in Computer-Generated Productions,* paper presented at the WIPO Worldwide Symposium on the Intellectual Property Aspects of Artificial Intelligence, Stanford University, April 1991.

45 Ronald Palenski, ADAPSO, personal communication, July 10, 1991.

46 Ken Wasch, Nicole Field, and Sara Brown, SPA, personal communication, July 30, 1991.

SPA's estimate is based on "average" software prices and an "expected ratio" of software applications to new personal computers purchased in 1990. SPA obtained hardware sales numbers for DOS-based and Apple computers from Dataquest. SPA obtained expected ratios of software to hardware from Apple, Microsoft, and Lotus; these ratios were an expected 3 software applications per DOS machine and 5 per Apple machine. Actual ratios based on software sales were 1.78 for DOS machines and 2.55 for Apple machines. (Nicole Field, SPA, personal communication, Aug. 14, 1991.)

47 The SPA estimated that microcomputer-software producers lost about $1 billion in sales to ' 'piracy" (defined by SPA as including both copying for personat use and copying for commercial profit) in 1986. (SPA estimate cited in Anne W. Branscomb, ''Who Owns Creativity? Property Rights in the Information Age," *Technology Review,* vol. 91, No. 4, May/June 1988, pp. 39-45.)

U.S. industry to lose $10 to $12 billion annually, compared to an estimated $12 billion generated by foreign sales of U.S. software.[48] (For more on piracy and efforts to combat it in the United States and abroad, see ch. 3).

International Treaties and Agreements

The United States is a member of the Berne Convention, the Universal Copyright Convention, and the Paris Convention (patents); the United States is also a party to numerous other multilateral and bilateral agreements. (For a full discussion, see ch. 3.) This section briefly spotlights the provisions of the Berne Convention and U.S. participation in the Uruguay Round of the General Agreement on Tariffs and Trade (GATT) and other international agreements. It also notes the software directive recently adopted by the European Community.

The Berne Convention—-The United States is a signatory to the Berne Convention for the Protection of Literary and Artistic Works. In pursuing its goals of effective, uniform protection of authors' rights in literary and artistic works, Berne employs the principle of national treatment,[49] and limits national treatment through the principles of reciprocity, establishment of minimum rights and automatic protection, and providing for the making of reservations. The Berne Convention protects "literary and artistic works" and does not specifically protect computer programs and databases. However, as the United States protects computer programs as literary works in its copyright law, computer programs are granted protection under Berne in the United States.

When it agreed to Berne, the United States was required to change its copyright law to make it compatible with the treaty through the Berne Convention Implementation Act of 1988. These changes include: abolition of mandatory notice of copyright; maintenance of mandatory deposit requirements; establishment of a two-tier registration system that differentiates between works of U.S. origin and works of foreign origin; a limit on the use of compulsory licenses; a minimum term of protection (life of the author plus 50 years). These changes caused by Berne must be considered or recognized in evaluating options for protecting software in the international arena.

The General Agreement on Tariffs and Trade— Some parties to the Uruguay Round of negotiations of the General Agreement on Tariffs and Trade have attempted to include what they refer to as ''trade-related intellectual property rights' (TRIPs) as a subject of the negotiations. These countries specifically have proposed provisions for protection of intellectual property rights in computer software. The U.S. proposal establishes the Berne Convention as the basis for minimum rights to be granted to authors by contracting parties to the GATT', and then sets forth additional protections provided to computer software and databases in the TRIPs. Provisions in the U.S. proposal reflect the current status of U.S. law protecting computer software.

Other U.S. Participation in International Treaties-In addition to multilateral treaties such as the Berne Convention and the GATT, the United States is party to bilateral treaties with nations in which specific provisions for intellectual property protection for computer software are delineated. For the most part, the United States uses the provisions of the Berne Convention as the bases for these treaties. The United States is also a party to the Universal Copyright Convention, created by the United Nations Educational, Scientific, and Cultural Organization in 1952 to provide an alternative to the Berne Convention that would not require a forfeit of copyright notice requirements.

The United States is party to many multilateral and bilateral intellectual property agreements.

The European Economic Community's Directive on Legal Protection for Computer Software— Following its Green Paper on "Copyright and the Challenge of Technology--Copyright Issues Re-

48 Robert W. Holleyman and Lori Forte, BSA, personal communication, July 12, 1991. Estimate includes all types of software, not just PC software. Foreign sales of PC application software are substantially less: SPA estimates that sales of packaged PC application software amounted to $4.5 billion in 1990--up 22 percent from 1989—and that foreign sales amounted to about $2 billion. (Ken Wasch, Nicole Field, and Sara Brown, SPA, personal communication July 30, 1991.)

49 National treatment requires each member nation to provide the same protection to works of nationals of other member nations as it does to works of its own nationals.

The function, external design, and code of a computer program, as well as the design of its user interface, have been the subjects of intense policy debate.

quiring Immediate Action, ' and after extensive and heated debate, the EC released its Council Directive on the Legal Protection of Computer Programs. In the prologue, the directive asserts the variety and scope of protection given computer software among member states and noted the problems they present to the European common market.

The articles of the directive provide for protection of computer programs as literary works within the meaning of the Berne Convention and establish criteria for authorship and beneficiaries of protection. The directive sets forth specific restricted acts, providing that the author has the exclusive right to reproduce or authorize reproduction of a computer program, to alter, translate or adapt the program, and to distribute the program to the public. The directive provides exceptions to these restrictions, including copying needed to use the program according to its intended use. The directive addresses the issue of recompilation, by allowing reproduction and translation of the code without authorization of the owner under certain conditions and when the information garnered from recompilation is to be used to achieve interoperability. The directive provides a term of protection of life of the author plus 50 years after death. The extent to which the directive addresses the concerns of U.S. manufacturers and reflects US. law responds to the trends in globalization of the industry and the law.

Sofware Technology

In this report, OTA has focused on four elements of a computer programe program function, the external design, the user inteface design, and the program code. Each of these elements has been the subject of an intense policy debate concerning the appropriate level of protection, and the level of intellectual property protection available under current law.[50]

Program Function

Computer programs instruct the computer to perform a series of operations to transform input values to output values. Under current interpretations of patent law, patents may be granted for parts of the program function. The same program may contain many patentable inventions-maybe none at all-depending on whether parts of the program function are novel, nonobvious, and meet the statutory definitional requirements. In applying for a patent, the applicant need not specify each operation performed by the processor, but describes the steps at a higher level of abstraction-e. g., ''storing a set of picture element data in a memory device. ' '[51]

The case law and PTO guidelines indicate that patents may not be granted for a "mathematical algorithm' (see footnote 20, supra). The meaning of the term "mathematical algorithm" has been the subject of considerable discussion,[52] but it appears to refer to a program function that is a "mere calculation." According to PTO guidelines, claims that include calculations expressed in mathematical symbols include a mathematical algorithm.[53] On the other hand, the function is not considered ' 'mathematical' if it can be stated in terms of its operations on things in the ''real world, '"[54] e.g., processing architectural symbols[55] or translating languages.[56]

External Design

Another intellectual property question concerns the protection of the external design or "interface" of a program. The external design specifies the inputs and outputs, and the conventions for communicating with a program. For example, a user would

50 Patent, copyright, and trade secret law all have to be taken into account by software developers.

51 U.S. Pat. No. 4,197,590.

52 See Pamela Samuelson, "BensonRevisited," *Emory Law Journal*, vol. *39, No. 4,* fall 1990, pp. 1025-1 154; Donald S. Chisum, "The Patentability of Algorithms, ' *University of Pittsburgh Law Journal, vol. 47, No. 4,* summer 1986, pp. 959-1022; Allen Newell, ''The Models Are Broken, The Models Are Broken!'' *University of Pittsburgh Law Journal,* vol. 47, No. 4, summer 1986, pp. 1023-1035.

53 U.S. patent and Trademark Office, Computer *Programs and Mathematical Algorithms,* September 1989, p. 8.

54 *In re Bradley*, 600 F.2d 812 (C.C.P.A. 1979).

55 *In re Phillips*, 608 F.2d 879 (C. C.P.A. 1979).

56 *In re Toma*, 575 F.2d 872 (C. C.P.A. 1978).

The user interface specifies conventions for communication between the user and the program.

have to know the specific commands of a user interface, their meaning, and formats for entering data. Other examples of interfaces are communications protocols and operating system calls.

The interface is conceptually distinct from the program code that implements the interface: there are typically many different ways of writing a program to provide the same interface. There has been considerable discussion whether it should be permissible to write a program that has the same external design as a previously copyrighted program. Some believe that intellectual property protection of interfaces is needed, while others believe that it is sufficient that the program code implementing the interface not be copied.

User Interface Design

Courts have been asked to resolve cases that assert protection of communications protocols and operating systems calls, but the type of external design subjected to the most debate has been the user interface. The user interface specifies the conventions for communication between the user and the program. There are a number of different kinds of user interfaces. One is the command language dialogue, in which the user issues commands to the computer through typed commands. If the program is used infrequently, it may be difficult for the user to remember the commands and how they can be used together to perform more complex tasks. "Menu" systems avoid this problem by displaying the command options on a screen; the user can then issue a command by pressing a key indicated as corresponding to a particular menu option, or by moving a cursor on the screen until the appropriate selection is highlighted. Newer interfaces make use of graphics or icons.

Program Code

The program code is protected by copyright: unauthorized duplication of a program except as provided by law (e.g., 17 U.S.C. 107, 117) will nearly always be a copyright infringement. However, an important aspect of the software intellectual-property debate is the degree of similarity that two programs can have, without infringement. The issue is whether two programs should be permitted to have similar ''structure, ' even if not every instruction is identical-i. e., at what level of abstraction above the literal code should two programs be permitted to be the same.

There are three different types of programming languages: machine language, assembly language, and high-level language. Machine language programs can be executed directly by the computer, but are difficult to write and understand.[57] Assembly language programs[58] and high-level language programs[59] are easier to write and understand, but cannot be executed directly by the computer. For this reason, programs are usually first written in assembly language or a high-level language, and then translated into machine language so that they can be executed by the computer.

Programs are typically distributed in machine language form. The program on the diskette is ready

57 Machine language instructions are patterns of 1's and ()'s which represent digital electronic signals inside the computer. These signals can take on one of two different values; to make it easier to think about what is happening inside the computer, programmers represent one of the values with the symbol 1,' and the other with the symbol "O.' For example, one type of 'addition" instruction for the processor which is used inmost microcomputers may be represented as "OOOOO1OO.' Inside the computer, the pattern of electronic signals corresponding to this pattern of O's and 1's would cause the computer to add two numbers together.

58 Assembly language makes programming easier by associating a short mnemonic with each type of operation. For example, a programmer using assembly language would represent the: addition instruction discussed above with the word ''ADD. ' Another typical assembly language instruction is "MOV,' which is used to MOVe a piece of data from one place to another inside the computer. Because the computer does not understand the assembly language mnemonics, they have to be translated into machine language instructions using a special program called an assembler. The assembler reads each assembly language instruction and replaces it with the appropriate pattern of 1's and 0's. For example, the ADD instruction might be translated to ''00000100.''

59 High-level languages (e.g., FORTRAN) are even easier to use and understand than assembly language. The instructions are ' 'English-1 ike, ' as with assembly language, but differ from assembly language instructions in that they are more powerful. Each high-level language instruction does the same job as multiple assembly language instructions. Because the computer does not understand high-level language instructions, they arc translated into machine language instructions using a special program called a compiler. The compiler reads each high-level language instruction and replaces it with the appropriate sequence of machine language instructions. More sophisticated compilers then perform "optimization": they may delete or rearrange machine language instructions in an effort to make the program execute more efficiently.

to be loaded into the computer and executed. Rarely is the program distributed in the high-level language or assembly language in which it was written. Distribution in machine language makes it difficult for others to read the program code to understand how the program works. This helps to maintain secrecy about the elements that give the program competitive value. The ''recompilation" issue is concerned with the legality of efforts to translate a machine language program into a more understandable form such as assembly language or high-level language (see box l-A).

Digital Information and Copyright

''Digital information' refers to the data stored on computers and in other digital media (e.g., magnetic or optical discs). Computer programs are used to manage and retrieve digital information. Software is necessary for users to access and manipulate digital information stored inside a computer or on storage media. It is difficult, with some modem programming techniques, to distinguish between the computer program and the data the program manages. *Thus, decisions affecting intellectual property and software may also affect digital information and the industries that create and use it. (See* ch. 5.)

Computers are revolutionizing the publishing industry. "Electronic publishing" is now used in the publication of most traditional books, newspapers, and magazines, as well as for delivery of documents to users in digital form. Computer software offers an increasing range of tools for storing, accessing, and manipulating information, Computers make collaboration and multiple authorship easy. Information in digital form is easily copied, transmitted, and modified. *These characteristics make it a good publishing medium, but also raise many intellectual property questions concerning what constitutes a copyrightable work, criteria for evaluating originality and authorship, and new ways of 'using' works and compensating authors.*

Due to some uncertainties about the level of protection offered by copyright law to digital data, copyright holders and vendors make use of contracts to attempt to control the uses of digital information by users. Data is often not sold to the user, but is licensed. There is a wide variety of terms and

Computer software offers an increasing range of tools for storing, accessing, and manipulating information.

conditions included in these license contracts. Many institutional users of digital information (e.g., libraries and university data centers) complain about the difficulty of managing and complying with the variety of contract terms required by their large collections of data and software packages. There is also controversy about the enforceability of some of these contracts, particularly where vendors have sufficient bargaining power to force terms on the user. (Contracts are also discussed in ch. 2.) Despite provisions of copyright law and license contracts, unauthorized copying of digital information still occurs.

Digital information is not just words and numbers. Anything that can be seen or heard can be digitized, so databases can include music, motion pictures, or photographs of art works. Some databases consist primarily of images. Mixed media or multimedia works are those that package together information in the form of images, sound, and/or text. For example, a multimedia cultural history of the 1960s might include text from newspapers and pamphlets, photographs, recordings of news broadcasts, segments of movies, recordings of music, along with software to access the information, all packaged together in a set of magnetic and optical discs. *There is no specific copyright category for protecting mixed media works.* In* addition, it is not always completely clear what obligations one has to original copyright holders when creating a database of digitized versions of all or part of works that fall under other copyright categories.

If computers have changed the publishing industry, they have also affected libraries. Libraries began embracing computer technology in the 1960s, at frost for administrative tasks like acquisitions and circulation. The first on-line library catalogs began to be developed in the mid-1960s, and many are now available for use by library patrons, offering them much greater flexibility in searching for needed

60 U.S. law includes eight categories of copyrightable works of authorship: literary works; musical works; dramatic works; pantomimes and choreographic works; pictorial, graphic, and sculptural works; motion pictures and other audiovisual works; sound recordings; and architectural works (17 U.S.C. 102(a)).

works. Some libraries are providing patrons with access to databases of bibliographic citations and full-text journal articles provided by commercial firms, as well as other services like access to electronic mail. These services are most widely available through research libraries at universities, but are increasingly moving into public libraries, where they may be important services for segments of the population that have no other reasonable access to digital information.

Uncertainties have arisen about libraries and digital information. For example, provisions under the copyright law for libraries to lend materials or make preservation copies apply to both printed information and "computer programs' (instructions to the computer) but not necessarily to digital information or mixed media works. Some ways in which libraries might wish to enhance services to patrons, e.g., upgrading on-line catalogs to provide tables of contents and other information from the cataloged books, might be considered to infringe on the underlying works. It is not clear what responsibility libraries may have for patrons' violations of copyright or for contract conditions when patrons have direct access to digital information.

Economic Perspectives

U.S. patent and copyright laws define limited monopoly rights[61] granted to creators of certain classes of ''works and inventions. ' *In this country, these monopoly rights are not viewed as ''natural' or ' 'inherent' rights of creators; rather, they are granted by the government in order to promote the public interest and are designed within a framework involving an economic tradeoff between private incentives and social benefits.* Thus, in the United States, an "intellectual property bargain" underlies the Federal framework for intellectual property law.

The Intellectual Property Bargain in U.S. Law

The rationale for this economic tradeoff-the "bargain" —recognizes that for certain goods, market forces will not necessarily produce the most desirable outcomes from the perspective of society as a whole. These goods will tend to be produced in insufficient quantity or variety because producers are unable to fully realize the gains from investments in creating them.[62] In granting a limited monopoly via copyright or patent, government attempts to compensate for distortions arising from this market imperfection. [63]

The linkage between intellectual property rights and economic benefits to society as a whole has traditionally followed the logic that: 1) intellectual property rights increase innovators' ability to appropriate returns from their intellectual labors; 2) the resulting potential for increased private gains to innovators induces additional innovation; 3) *because of increased innovation,* additional benefits accrue to society as a whole.[64] The U.S. system of patents and copyrights is intended to strike a balance between holders of intellectual property rights and the public at large. This balance involves benefits and costs on both sides: legal protection for intellectual property imposes costs on a society, as well as benefits. These costs and benefits can be monetary (e.g., increased or decreased costs or royalties), or less tangible (e.g., social consequences of stimulated or stifled technological advances). *The specifics of how this balance is maintained--the exact form, scope, and duration of intellectual property rights—may evolve in response to changes in technology, markets, or social values.*

Intellectual Property and Software

Economists have been paying increasing attention to intellectual property and software, but as yet there are no firm conclusions as to what socially optimal protections may be. The lack of precise policy

61 *OTA note:* In this report, ' 'monopoly" is used in the economic sense and should *not* be taken as synonymous with illegal monopolization of a market or markets. For discussion see F.M. Scherer, *Industrial Market Structure and Economic Performance*, 2d ed. (Chicago, IL: Rand McNally College Publishing Co., 1980), pp. 527-594. As Scherer notes, "Congress [chose] the word 'monopolize' to describe what it condemned, and not some more conventional phrase such as 'obtain or possess monopoly power' '' (p. 527).

62 Some goods (like information) have the property of nonexclusivity: once the good has been produced and publicly distributed, it is impossible (or prohibitively costly) to exclude any individual from benefiting from it, whether or not he or she pays. Furthermore, consumers' individual self-interests provide strong incentives not to pay for the good, or to undervalue it, in hopes of getting access as "free riders, " See ch. 6.

63 ''The Congress shall have Power . . . To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries. (U.S. Constitution art. I, sec. 8, cl. 8.)

64 ''The economic philosophy behind the clause empowering the Congress to grant patents and copyrights is the conviction that encouragement of individual efforts by personal gain is the best way to advance public welfare through the talents of authors and inventors in Science and the useful Arts. " (*Mazer v. Stein*, 347 U.S. 201,219 (1954).)

prescriptions regarding linkages among intellectual property, technological innovation, and social benefits is not unique to software, although economic inquiry is made all the more difficult by the rapid changes in software technologies and markets. Literature on the economics of software is still evolving, along with the broader literature on intellectual property and innovation.[65] Chapter 6 of this report offers a "snapshot' of economic thinking, rather than economists' solution to the problem of how best to balance private incentives and social benefits in a rapidly moving area of technology.

In the United States, an "intellectual property bargain" underlies the Federal framework for intellectual property law.

The economics literature on intellectual property focuses mostly on patent and copyright. *In large part, this focus stems from the nature of patent and copyright." these exclusive rights have been designed within a framework involving an economic tradeoff between private incentives and social benefits .66 The* laws governing trade secrets do not incorporate this kind of explicit tradeoff.[67] Therefore, patent and copyright offer more established economic bases for theoretical and empirical analyses of markets for intellectual property. The bulk of economic analysis on linkages among technological progress, economic welfare, and intellectual property has dealt with the patent system, rather than copyright. Software is remarkable in being a technology for which copyright is so crucial.[68] However, many of the arguments concerning patents and duplication of innovations can be applied to software copyright, especially to issues like copyright protection of interfaces and the appropriate breadth of copyright protection.

As noted previously, this is an evolving literature. Sometimes, the analyses discussed in this report differ in conclusions or policy implications. In particular, some of the economic research done since CONTU suggests policy implications that differ from those of earlier work in the 1950s, 1960s, and early 1970s.[69] These differences result because the economic models incorporate different industry conditions, different types of innovation, and different timeframes. Much of the earlier economic work on intellectual property (mainly patents) focused on cost-saving *process* innovations, while later work looked at *product* innovations. Until after the mid-1970s, most analyses of (socially) optimal patent design focused on *patent term* and assumed static (one-shot) models of innovation. The more recent work focuses on *breadth of protection,* as well as term; dynamic models of innovation include the possibilities of multiple inventors,[70] cumulative innovation, and network externalities. The more recent work, using dynamic models for innovation,

65 *OTA note:* Unless otherwise specified, OTA uses "innovation" and "innovative activity'' in this chapter to refer to R&D and other creative processes producing scientific and technological advances, whether the form of these advances would legally be considered copyrightable, patentable, or neither. In reviewing the economics literature on this topic in ch. 6, OTA uses the authors' terminology.

66 Private incentives are expected to arise from the right holder's limited monopoly powers; social benefits are expected to include additional benefits to society from the induced disclosure and/or dissemination of innovations and technological advances.

For discussions of this balancing between private incentives and social benefits in the "intellectual property bargain," see *intellectual Property Rights in an Age of Electronics and Information,* op. cit., footnote *2. See also* Paul Goldstein, op. cit., footnote 24, sees. 1.1 and 1.2.

67 See Stanley M. Besen and Leo J. Raskind, "AnIntroduction to the Law and Economics of Intellectual Property, ' *Journal of Economic Perspectives, vol. 5, No.* 1, winter 1991, pp. 3-27, esp. p. 23,

The rightful possessor of a trade secret does not have an exclusive right to use the secret information, and the law only provides for legal remedies when the secret is lost through breach of contract or ''impmper" means of discovery (e.g., industrial espionage). A trade secret may be maintained indefinitely. See ch. 6 and also David Friedman et rd., "Some Economics of Trade Secret Law," *Journal of Economic Perspectives,* vol. 5, *No.* 1, winter 1991, pp. 61-72.

6s However, there is also a well-developed literature dealing with economic welfare, copyright, and consumer copying of journal articles, music, software, etc. (see the final section of ch. *6* on home copying).

69 As described by Sidney Winter, the pendulum of opinion on the ''optimal'' term of protection (e.g., whether increasing or decreasing the term Of patent protection would be more socially desirable) has swung back and forth over the years. For his discussion of changes in economic thinking about the term and strength of protection, see Sidney G. Winter, "Patents in Complex Contexts: Incentives and Effectiveness,' in Vivian Weil and John W. Snapper (eds.), *Owning Scientific and Technical information (New* Brunswick NJ: Rutgers University Press, 1989), pp. 41-43.

For another discussion of economics literature on innovation, see Robert P. Merges, "Commercial Success and Patent Standards: Economic Perspectives on Innovation," *California Law Review, vol. 76,* pp. 803-876, 1988.

70 In multiple-inventor "patent races," the rate of R&D spending affects the probability of invention.

suggests shorter, rather than longer, terms of protection.[71]

Software Industry and Technology Changes

Computing technologies and the software market have also evolved since the mid-1970s (see chs. 3 and 4). Because of timing, CONTU and its analyses could not foresee the time when powerful computers could be in every office and every home and individuals would be able to create and use sophisticated and valuable software outside large organizations, or the full impact of these changes a decade later.

CONTU saw the software market shares of hardware producers being "steadily eroded" by independent software developers and expected this trend to continue.[72] But, despite vigorous growth by independents, computer-hardware firms retain a major share of the software market. (In terms of revenues, IBM is the leading U.S. software producer overall and the largest producer of packaged software in the world—see ch. 3). In CONTU'S analysis,[73] the software industry was characterized by easy, rapid entry by small fins-a viable ''cottage' industry of small developer-vendors.[74] Although there are still thousands of small companies, the industry today is moving away from this picture. Significant changes, which may affect the way in which Congress sets the balance for software, include:[75]

- . the ''PC Revolution' and explosive growth in markets for personal computers and packaged software;
- widespread use of computers and software by nonprogrammers and the corresponding market importance of user interfaces;
- . increased barriers to entry by small firms and a trend toward centralized software-publishing houses that acquire rights to software and then distribute and market it, paying royalties to the program developers;
- maturity of the software industry and increasing firm size (through growth, acquisition, and consolidation); and
- increasing industry concentration, especially when considering submarkets like PC applications.

Issues and Options

In this report, OTA focuses on the various ways in which current U.S. copyright, patent, and trade secret laws apply to four key elements of computer program development—the *program function, the external design, the user interface design,* and the *program code. Our* study has examined the technology behind each of these aspects of the development process, the application of current intellectual property laws to each of the four elements, and the policy issues and arguments associated with them, in terms of current law and in terms of possible modifications to the existing intellectual property system.

Technological Challenges for Copyright Law

OTA finds that treating computer programs as literary works under copyright offers straightforward remedies for the literal copying of program code, although enforcement remains a problem, **especially overseas. (See** ch. **3.) OTA** also finds, however, that the functional aspects of computer programs pose difficult questions for application of the copyright law.[76] One important question is the extent to which copyright (particularly, in concert with trade secret law) should protect the functionality and design of a program. OTA finds that the traditionally "fuzzy" line between idea and expression in copyright law is confounded by the need to determine an appropriate scope of protection in order not to provide coverage for the program procedure, process, system, method of operation, or concept, con-

71 See Winter, op. cit., footnote 69

72 *CONTU Report*, op. cit., footnote 16, p. 24. (Quoting from the economic analysis prepared for CONTU by the Public Interest Economics Center, *An Analysis of Computer and Photocopying Copyright Issues From the Point of View of the General Public and the Ultimate Consumer* (Washington, DC: June 1977), p. IV-13.)

73 Ibid., p. 23.

74 Public Interest Economics Center, op. cit., footnote 72, p. IV-5.

75 For discussion, see box 6-A in ch. 6.

76 See also *Intellectual Property Rights in an Age of Electronics and Information,* op. cit., footnote 2; and Paul Goldstein, op. cit., footnote *24,* sec. 2.15.2, pp. 206-207. But see Morton David Goldberg and John F. Burleigh, "Copyright Protection for Computer Programs: Is The Sky Falling?' *AIPLA Quarterly Journal, vol. 17, No. 3, 1989;* and Antbony L. Clapes et al., "Silicon Epics and Binary Bards: Determining the Proper Scope of Copyright Protection for Computer Programs, ' *UCLA* Law *Review,* vol. 34, June-August 1987.

trary to the intent of the current law (17 U.S.C. 102(b)) .77 This key software-copyright question is manifested (among other places) in the debate over application of copyright to features of a program's external design and/or user interface, and in the debate over reverse engineering and decompilation (see above and ch. 4).

Technological and Institutional Challenges for the Patent System and PTO

Patent protection is used extensively to protect software-related inventions in the United States, Japan, and Europe. Many of the major commercial developers of software (including firms like IBM that tend to be thought ofas''hardware' companies) are extensively using patent protection for software-related inventions, As a result, patent protection is of importance to the U.S. software industry, both domestically and in the global market.

The protection of software-related inventions and algorithms by patent[78] is a fairly recent development and is controversial.[79] (See chs. 2 and 4 and boxes 1-B and l-C.) The case law and PTO guidelines indicate that patents may not be granted for a specific kind of program function called a ' 'mathematical algorithm' (see footnote 20, supra and discussion in ch. 4). The meaning of the term ''mathematical algorithm' has been the subject of considerable discussion and debate. Moreover, some academics and members of the software community perceive that the technology[80] and economics and industry structure[81] of software development make patents inappropriate for software-related inventions.[82] Many other members of the same communities, however, disagree with this perception and consider that patents for software-related inventions are appropriate for the industry and are in the public interest.[83] Furthermore, the latter argue that the ''disadvantages' perceived by critics are fully addressable by changes in the operation of the PTO.[84]

OTA finds that the PTO faces considerable challenges in examining applications for software-related inventions. At the same time, there ap-

---

77 "Somc concern has been expressed lest copyright in computer programs should extend protection to the methodology or processes adopted by the programmer, rather than merely to the 'writing' expressing his ideas. Section 102(b). . .is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law. ' (U. S.C.A. 17 sec. 102, "Nature of Copyright, " p. 17, from House Report No. 94-1476, Committee on the Judiciary.) See also footnote 24, supra.

Congress has already (17 U.S.C. 117) limited copyright holders' exclusive rights for computer programs.

78 See footnotes 19 and 20, supra.

79 ''Algorithms (also known technically as the partially recursive functions) form the essence of software. Increasingly we are seeing lawsuits or threats of lawsuits claiming patent infringement for the mere use of software on a typical computer, e.g., for such functionality as public key encryption compression and cursor blinking. If patent law establishes that such suits are justified, that will mean to me that algorithms do have patent protection. To underscore that algorithms do have the same unpatentability as scientific principles, I think patent law should be clarified to the effect that a patent is never infringed merely by the use of software on a computer. ' (Robert S. Boyer, Professor of Computer Sciences, University of Texas at Austin, letter to Harry F. Manbeck, Jr., Assistant Secretary and Commissioner of Patents and Trademarks, Sept. 22, 1991.)

so See, e.g., Daniel J. Bernstein, University of California, letter to E.R. Kazenske, PTO, July 10, 1991 (Response to Request for Comments for the Advisory Commission on Patent Law Reform). Using issued patents from the field of data compression as examples, Bernstein discusses his views that PTO examiners have failed to detect "mathematical algorithms" in claims, that PTO is not correctly evaluating criteria for nonobviousness and equivalence of patent claims for algorithms, and that "software patents" are beginning to damage the software industry. See also ch. 4.

81 For example, entry barriers are said to be lower for software than for commercial biotechnology, which has led to an industry with more small entrepreneurs for whom the administration of patents would be difficult, especially given that one program may contain (or infringe) many patented processes. See ch. 6.

82 For arguments against ''software patents, ' see Pamela Samuelson, op. cit., footnote 52; and Richard Stallman and Simson Garfinkel (The League for Programming Freedom), ''Against Software Patents, " Communications *of the ACM, vol. 35, No.* 1, January 1992, pp. 17-22,121.

Among the problems noted are an incomplete prior art and insufficient examiner training, leading to difficulties in examination and issuance of patents that are not novel and/or nonobvious. Another problem noted is that the pendency period is long compared to software-development cycles, so that ''landmine'' patents can issue. (See chs. 2 and 4.)

83 See, e.g., Paul Heckel, op. cit., footnote 23; John L. Pickett, President, CBEMA, letter to E.R. Kazenske, PTO, in response to Request for Comments for the Advisory Commission on Patent Law Reform, July 15, 1991; Esther Schachter, Chair, ADAPSO Intellectual Property Committee, letter to E.R. Kazenske, PTO, in response to Request for Comments for the Advisory Commission on Patent Law Reform, Sept. 4, 1991; and Robert G. Sterne, letter to Paula Bruening, OTA, Oct. 6, 1991. Steme argues that many inventive aspects of software-related inventions that are protectable by patent cannot be protected adequately and/or effectively and/or at all by copyright and trade secret (ibid., p. 2).

S4 John L. Pickett, op. Cit., footnote 83. CBEMA rogues that perceived disadvantages such as ' 'bad patents' anticipated by the prior art, examiners' difficulties in deciding questions of novelty and nonobviousness, and vulnerability to patents issuing after long delays *IOTA note:* i.e, ''landmine'' patents] are fully addressable by changes in operation of the PTO. (Ibid., p, 2.)

pears to be some variance-or, at least, uncertainty on the part of observers outside PTO—in how PTO examination guidelines for subject-matter determinations are being applied. A long series of (often inconsistent) court decisions has led to a situation in which some types of software-related inventions are patentable while others are not. Applying the PTO guidelines for distinguishing patentable and nonpatentable types of inventions is a complex part of the examination process and one that outsiders find difficult to understand and/or predict.[85] (As a matter of policy, PTO does not comment on the examination process for issued patents. Because OTA could not be "walked through' application of PTO examination criteria or discuss their interpretation for specific patents, OTA was unable to make any independent finding on the quality of examination for particular software-related patents.) In addition, the PTO has an incomplete database of "prior art" for software-related inventions.[86] This makes it even more difficult for examiners to judge whether an application describes a "novel" and "nonobvious" invention. Filling in the gaps in the database of prior art maybe difficult., because so much of what would constitute the ''prior art' has historically been in the form of products, not literature or issued patents.

To address the issue of examination quality, the U.S. Patent and Trademark Office is currently reviewing the nature of the qualifications required for examiners and has recently completed the first phase of its reclassification of the software arts. This reclassification process involves the creation of a new Class 395, "Information Processing System Organization.' '[87] To make searching easier, this new class will have a larger number of subclasses than the old Class 364, each encompassing a particular area of technology-e. g., database and file management systems or artificial intelligence. PTO intends that examiners will specialize in one of these subclasses. In addition, the Secretary of Commerce established an Advisory Commission on Patent Law Reform** to examine administrative and procedural challenges facing the patent system.

OTA finds that "filling in" the prior art database (patent and nonpatent) is extremely important, as one means of improving the quality of examination. OTA also finds that improving electronic search and retrieval capabilities for the PTO'S own database is critical, because it is used by the PTO'S own examiners during the application process and by the public. In September 1991, PTO reported that it is unable to provide statistics on the number of patents issued for software-related inventions (e.g., patents for computer processes and algorithms), which PTO refers to as ''computer-implemented process patents. '[89] Despite the intense controversy and policy focus on these areas of art since *Diamond v. Diehr, PTO* reported to OTA that it has no provisions for flagging, cross-referencing, or otherwise efficiently monitoring and reporting prosecution, issuance, and litigation for these types of patents, except through time-consuming manual search, review, and selec-

---

85 ''~~so finds the guidelines themselves to be a reasonable reading of case law but questions how the PTO is actually administering the stated Guidelines. Enough ADAPSO member companies have noted a substantial increase in subject-matter rejections on computer program-related claims as to constitute a new trend. . while these rejections cite Section 101, this new practice has no obvious basis in the statute itself and does not represent the kind of result that we believe the Guidelines would lead one to expect. " (Schacter, ADAPSO, op. cit., footnote 83, pp. 15-16.) Others have questioned how examiners interpreted the PTO guidelines in *issuing certain* patents. (See, e.g., Brian Kahin, ''The Impact of Software Patents, ' *EDUCOM Review*, winter 1989, pp. 26-31 .)

86 See Jeffrey M. Samuels, ~@ Commissioner of patents and Trademarks, testimony at Hearings on Computers and Intellectual Property, Mar. 7, 1990, U.S. House of Representatives, 101st Congress, 1st and 2d Sessions, Subcommittee on Courts, Intellectual Property, and the Administration of Justice, Committee on the Judiciary, Serial No. 119, pp. 325-354. According to Samuels' testimony, ''We are concerned that some computer processes that are sold or are in use are not fully described in the published literature or readily evident from use of the process. The sale or use of the process is evidence that the process is not new and should not be protected by a later filed application. Regrettably, there is not an efficiently searchable record of this type of prior art, not only for computer-related inventions but for all inventions. ' (Ibid., pp. 337-338.)

87 Gerald Goldberg, Director, Group 230, PTO, personal communication, Oct. 18, 1991.

88 For further discussion of the Advisory Commission on Patent Law Reform, see pp. 10-11 of this chapter, and ch. 2.

89 See Jeffrey M. Samuels, Acting Commissioner of Patents and Trademarks, letter to Congressman Robert W. Kastenmeier, Chairman, Subcommittee on Courts, Intellectual Property and the Administration of Justice, Nov. 1, 1989, answer to question 1: ''Computer processes are not classified within USPTO's patent classification system in any readily identifiable set of classes and subclasses. ' See also Harry F. Manbeck, Jr., Assistant Secretary and Commissioner of Patents and Trademarks, letter to Joan D. Winston, OTA, Sept. 11, 1991, p. 1 (". . .it is not possible to generate reliable data in response to questions directed to computer-implemented process patents through the manipulation of existing PTO databases' ').

tion from various large patent subclasses.[90] OTA found, however, that the private sector has had more success in developing such statistics and classification schemes. Electronic Data Systems Corporation (EDS) has compiled statistics on patents issued for software-related inventions during the years 1972 through 1989 by examining notices published in the Official Gazette of the Patent and Trademark Office. EDS obtained similar statistics for 1990 by reading official database tapes purchased from the PTO. With the 1990 data, an artificial intelligence technology (involving a natural language interface created by EDS) was used to characterize each patent and determine which involved software-related inventions. Using this method, EDS found that 576 patents were issued for software-related inventions during 1990. (According to EDS, this technology will be used to reexamine the statistics initially gathered for 1972 through 1989. For more on the EDS statistics, see table 2-1 and accompanying discussion in ch. 2.)

These shortcomings in the PTO database affect searches conducted by or for the public. Therefore, while "filling the gaps" in the prior art will be useful, these steps should be taken in conjunction with measures to improve electronic search and retrieval and provide statistical information for use within PTO and for reports to Congress. PTO is still in the process of deploying its automated patent system and has also begun to reclassify patents in the computer arts (see discussion inch. 2, pp. 54-56). As part of its oversight, Congress may find it useful to receive statistical profiles of patent activity in this and other important areas of technology .91 Therefore, Congress may wish to determine what improvements in statistical reporting by field of technology will be part of this automation and reclassification.

International Dimensions

Computer software markets are international, as are software research and development. Although software markets and industries abroad are growing, the United States currently remains a major force in this international market (see ch. 3). The issues and questions facing Congress are more complicated because of the "global" nature of software. The balance struck in the intellectual property bargain' cannot ignore increasing foreign competition in overseas markets-and within U.S. borders .92

The paradigm of software as or akin to a literary work under copyright is the keystone of existing international copyright agreements. However, foreign countries may treat software differently in some respects, even with a copyright framework. For example, the European Community's software directive includes specific provisions concerning certain aspects of reverse engineering, referred to in the directive as "recompilation" (see ch. 3).

The level of patent protection offered by foreign countries for computer software varies: while some may not protect software per se, they may grant protection for processes that include software, or if a program is claimed in conjunction with a method or computer. To the extent that the market for software is global, policy decisions about patent protection for software reflect these differences in

90 From letter t. Joan D. Winston, OTA, from Harry F. Manbeck, Jr., Assistant Secretary and Commissioner of Patents and Trademarks, Sept. 11, 199 1: ''In sum, PTO is not able, through its existing databases, to respond to OTA's request for data concerning 'computer process and algorithm' patents. To compile data with which to respond to OTA's range of questions involving computer-implemented process patents would require a manual search of many technology classes, a thorough review of the claimed invention, and the investment of hundreds of staff weeks. '

OTA had requested statistics from PTO concerning prosecution, issuance, and litigation of patents for software-related inventions. In its request, OTA had asked for statistics on what it referred to as "computer process and algorithm patents. " OTA staff asked to meet with PTO staff to discuss the request-particularly, to explore PTO'S suggestions for alternative formulations of OTA's questions in order to facilitate a meaningful response. (Personal communications with PTO staff March-June 1991 and letter from Joan D. Winston, OTA, to Lee Skillington, Office of Legislative and International Affairs, PTO, June 24, 1991,)

No such discussion or suggestions were provided and PTO responded that, ''As a general matter, the denomination 'computer process and algorithm' patent bears no direct correlation to PTO policy or practices. . patents issued for inventions involving computer-implemented processes are not classified in a single technology class or subclass. . and have not been otherwise 'flagged' to enable the retrieval of the requested data. . even if we were to limit the task to Subclass 364/200 and Subclass 364/900, the task would still be a significant one. , .[that] would probably exceed 30 staff weeks. " (Manbeck, op. cit., footnote 89, enclosure item 3.)

91 For example, OTA had asked PTO for data on patents and patent applications for computer processes and algorithms: the numbers Of such patents issued from 1974-present, cumulative numbers of patents in effect during this period, average tendencies, examiner rejections appealed to PTO and the courts, etc. PTO reported that it was unable to provide this information because it would have required ''hundreds of staff weeks' to prepare. (Manbeck, op. cit., footnote 89.) In 1989, the House Subcommittee on Courts, Intellectual Property, and the Administration of Justice asked for similar information as part of its oversight of computers and intellectual property; PTO reported that it could not provide it. (See footnote 89, supra.)

92 For example, the United States is a signatory to the Beme Convention and Universal Copyright Convention. Under these agreements, our domestic copyright law applies reciprocally to foreign copyright holders who are nationals of convention members (see ch. 3 and footnote 49, supra).

legal systems. The World Intellectual Property Organization's Committee of Experts on the Harmonization of Certain Provisions in Law for the Protection of Inventions is considering a draft treaty that would provide for modifications of general aspects of patent system, not specific to software, including a first-to-file (an application with the patent office) system, and establishment of a term of patent protection of 20 years from the date of filing a patent application.

Digital Information and Copyright

OTA finds that many of the issues of concern with copyright and computer software also apply to digital information. For example, copyright provides remedies for the literal copying of digital information, but as with software, enforcement may be a problem. There appear to be few technological or other remedies to prevent unauthorized copying, except within closed systems.

It would be helpful for both publishers and users of digital information if some aspects of fair use of digital information under the copyright law were clarified. Because of uncertainties about users' rights to "download" or make copies of information, providers of digital information rely on contracts to limit customers' uses of information, and do not sell information to customers, but merely authorize certain uses. On-line digital information may pass through several intermediaries between the publisher and the end user--distributor, database service, library-making contracts less effective for controlling end-user practices.[93]

Some aspects of fair use also remain unclear with regard to libraries. For example, while guidelines have been developed for libraries' making archival copies of books or of computer programs, no mention is made of rights to make such copies of databases or other information in digital form. There is also some question as to how far libraries can go in enhancing the content of their on-line information retrieval tools by including more information from the original works (e.g., the table of contents or the index). While such enhancements are now feasible and seem a logical step, some hold that such enhancements may infringe on the copyrights of the underlying works.[94]

Digital information includes multimedia or mixed media databases, which may include images, music, text or other types of works. The status of mixed media works under copyright is not clear. Mixed media is a fairly new concept; the acquisition of rights to convert copyrighted works to digital form for incorporation in mixed media databases is often difficult because conventions and standards for royalties do not yet exist, nor are there organizations of rights holders to collect the royalties.

Software Technology, Industry Structure, and the Future

Software and computer technologies are fast-moving and complex. The software industry and the discipline itself are maturing: the software industry structure has changed since CONTU and the costs of successful market entry are rising. Although there continue to be many viable small firms and entrepreneurs, the industry is moving away from the model of a cottage industry (see box 6-A inch. 6). With the vast increase in numbers and types of software users, user interfaces have become increasingly important, both to users (in terms of ease of learning and use, performance, productivity) and to developers as well (in terms of value in the marketplace and market share).

Despite the advantages of incremental accommodation within the current structures, especially in terms of established case law and reciprocal international protection, OTA finds that there may be a point where it is in the public interest to develop new law(s) either ***to complement*** the existing framework or ***to substitute for*** copyright and/or patent protections for software, rather than continue incremental accommodation. Congress may eventually find that the best means for achieving policy objectives with respect to software are different from those used for other

---

93 Brian Kahin, "Contract and Fair Use Issues in Downloading: Subcommittee Report, Committee 702 (Databases), " in *Section on Patent, Trademark and Copyright: Committee Reports* (Chicago, IL: American Bar Association 1989), pp. 405411.

94 Mary Jensen, Director, University of South Dakota Law School Library, personal communication, Feb. 8, 1991.

types of works, based on the intellectual property bargain.[95] In its deliberations, Congress could draw upon public input from many economic and social sectors[96] in assessing the net impact of new rights that might be created as alternatives to a strategy of accommodation within the current structure.

Some commentators favor *sui generis* approaches, either to complement or substitute for current software protections. Most proponents of a ' 'substitute' *sui generis* law seem to envision a modified copyright approach.[97] That is, a copyrightlike registration would continue, but the term of protection and the bundle of rights would be modified to conform to what is considered to be the needs of software. In the Semiconductor Chip Protection Act (SCPA), a modified copyright approach is used to protect chip mask works from copying. 98 Some early proposals for the protection of semiconductor chips had recommended amending the Copyright Act, but a *sui generis* approach was chosen to avoid distorting traditional copyright principles for other categories of works .99

Software features and advances that may be valuable and beneficial to society may not be traditionally patentable or copyrightable subject matter. Therefore, OTA finds that Congress may wish to consider periodically whether there is some public advantage in giving limited rights for incremental software advances that would not be patentable or for aspects of program functionality that fall outside copyrightable subject matter. In this case, the subject matter, scope, term, and exemptions from infringement of a "complementary' *sui generis* law could be carefully tailored to fit the characteristics of the technology and its uses.[100]

One intellectual property scholar has suggested that software is an example of a "legal hybrid" that falls between patent and copyright.[101] To encourage innovation, these hybrids are thought to require some kind of protection to ensure lead time. However, for these hybrids, it is thought that patents will usually protect only a small portion of the innovation, and the ''powerful reproduction rights and long term of protection [of copyright] implement cultural policies that are largely irrelevant to the needs of a competitive market. [102] According to this logic, in addition to fading to protect innovation

95 For example, Congress might wish to consider whether the trend toward more use of patent protection, absent alternatives to protect program functionality, affects the public-interest "balance" in terms of equity for small/large software firms and for those with many/few legal and financial resources: What will be the effect on end users and the public at large? Will ''stronger' protection for software (e.g., patents precluding commercial exploitation of independent program inventions, copyright protection for the design of user interfaces) spur innovation stifle it, or have no real effect overall? Will it disproportionately disadvantage individuals and small firms versus large (or rich) corporations?

96 See, e.g., the discussions of public input in crafting the SCPA in Robert W. Kastenmeier and Michael J. Remington, op. cit., footnote 30, esp. pp. 424-432 and 442-459; and Richard H. Stem, "Determining Liability for Infringement of Mask Work Rights Under the Semiconductor Chip Protection Act, " *Minnesota Law Review,* vol. *70, No. 5,* December 1985, pp. 271 et seq.

97 See Pamela Samuelson, op. cit., footnote 52, esp. pp. 1148-1153; and Richard H. Stern, "The Bundle of Rights Suited to New Technology,'' *University of Pittsburgh Law Review,* vol. 47, No. 4, p. 1229.

A modified copyright approach has previously been used for the protection of semiconductor chips (see below and ch. 2). There have been a number of proposals to protect industrial designs using a modified copyright approach, but these have not been enacted into law. For a discussion of industrial design protection, see ch. 2.

98 The SCPA Uses a modified copyright approach to protect the topography of integrated circuits against copying. There is 110 patentlike examination process; the ''mask work' is registered with the Copyright Office. However, the SCPA has a novelty standard somewhat higher than the mere ''originality' standard of copyright law: protection is not available for a mask work that ''consists of designs that arc staple, commonplace, or familiar in the semiconductor industry or variations of such designs, combined in a way that, considered as a whole, is not original''(17 U.S.C. 902(b)(2)). The bundle of rights is also somewhat different from that granted under copyright law, and copies of the ''mask work' made in the course of reverse engineering are not infringing (17 U.S.C. 906(a)). Finally, semiconductor chip protection differs from copyright in that the turn of protection is only 10 years.

99 See Kastenmeier and Remington, op. cit., footnote 30, pp. 424-430 and 442-444 and H.R. Report No. 781, 98th Congress, 2d Sess., 1984, pp. 5-11.

100 For example, use of copyright mandates a long term of protection for software, no provision for compulsory licensing, and limited exemptions from infringement (e.g., secs. 107 and 117). Patent has a shorter term, but many program features may not be patentable subject matter; although reverse engineering is allowed, independent invention is not a defense to claims of patent infringement.

The SCPA, which is not part of the copyright law but is "in harmony" with it, was tailored with a shorter term, technology-specific subject matter (original mask works), and explicit exemptions for reverse engineering that differ from copyright's fair-use exemptions. See Kastenmeier and Remington. op. cit., footnote 30, pp. 445-452,

101 J.H. Reichman, "Computer Programs as Applied Scientific Know-How: Implications of Copyright Protection for Commercialized University Research, " *Vanderbilt Law Review, vol. 42, No. 3,* April 1989, p. 655.

102 Jerome H. Reichman, ''Proprietary Rights in the New Landscape of Intellectual Property Law: An Anglo-American Perspective, ' study prepared for the International Literary and Artistic Association, Congress of the Aegean Sea II, June 19-26, 1991, p. 54.

320-226 0 - 92 - 3

properly, attempts to use existing laws for these hybrids risk distorting the existing laws.[103]

The CONTU report had noted that patent protection for software was limited, and that some additional form of protection would be required. However, the Commission did not recommend a *sui generis* approach, concluding that copyright was appropriate.[104] The argument that computer programs are ''useful articles' or otherwise fall outside the range of statutory subject matter was rejected by CONTU[105] and has been consistently rejected by the courts.[106] Proponents of the continued use of copyright law contend that copyright, as interpreted by the courts, is working well,[107] that a new system would create unacceptable uncertainty, and that existing international agreements provide a framework for the protection of computer programs in other countries (see ch. 2).

Policy Choices and Options

OTA has identified three principal policy areas that Congress may wish to address. These are:

1. difficulties that the functional aspects of computer programs present in determining the appropriate scope of copyright protection for programs;
2. difficulties in determining the scope of patent protection for software-related inventions and algorithms and the challenges facing the U.S. Patent and Trademark Office in these areas of art; and
3. complications facing libraries and developers and users of digital information, especially mixed media works encompassing several different categories of' copyrightable works.

As the preceding sections and body of this report detail, these principal areas encompass a variety of issues. For each, Congress haas fumdamental choices.

The first of these choices is to act or not to act. Not acting continues the status quo in terms of statute, and allows the continued evolution of the case law, but does not assuage uncertainty. On the other hand, taking action may reduce some uncertainties but add others, especially if additional bodies of case law and new international agreements had to be developed (e.g., for a *sui generis law).* This choice is not a static decision—Congress may wish to periodically reevaluate the choice to take action or continue the status quo. In so doing, Congress can draw upon input from the broad communities of stakeholders in these issues. These stakeholders include the computer and software industries; members of the academic, research, and library communities; private, corporate, and institutional software users; the software-using workforce; and the public at large (see above discussion on the evolution of the software debate and box l-D).

The second choice, if Congress determines that action will be in the public interest, is the timeframe for action. If sufficient information concerning an issue and how to resolve it unambiguously is available, Congress could act in the near term. In instances when institutional problems and alternative courses of action are well-defined, near-term actions may be both appropriate and necessary, especially when they require some lead time before yielding benefits (e.g., see discussion of Options 2.3 through 2.6).

Otherwise, Congress might wait until more information becomes available concerning the likely outcomes of an action (compared to the status quo) in order to avoid precipitous action and legislation that may not have the desired long-term effect or that may quickly be outdated by changes in technology. Delaying action will also allow Congress to have the benefit of additional information about the course of case law (on software interfaces, patent litigation and appeals, etc.) and on the impact on software developers and users of the current legal environment and uncertainties. In the nearer term, Congress could initiate strategic information- and input-gathering processes (e.g., Options 1.5 and 3.3, see also section below on “planning for the future”) and use these as a basis for evaluating

103 See discussion, ibid., p. 550

104 CONTU noted language in the House and Senate reports accompanying the 1976 Copyright Act indicating that the act did not need to be amended to include computer programs, and language indicating that computer programs were copyrightable as “literary works. ” (See *CONTU Report*, op. cit., footnote 16, p. 16.)

105 Hersey dissent, *CONTU Report*, Op. cit., footnote 16, p. 31.

106 See, e.g., *E.F. Johnson v. Uniden*, *623* F. Supp. 1485, 1498.

107 See, e.g., Morton David Goldberg and John F. Burleigh, op. cit., footnote 76, p. 294.

longer-term options.[108] Such a strategy fight be especially helpful in determining whether statutory changes are necessary and, if so, what definitional problems and uncertainties will be addressed. Possible disadvantages of waiting are that incremental accommodations through the case law may conflict over time, as the case law continues to evolve. As is the case with current legal uncertainties, the uncertainties that ensue will affect smaller/poorer firms and individuals (that do not have the resources to ''ride it out) more than large firms with deep pockets.

If Congress chooses to take action, it faces a third choice--how comprehensively to act. Congressional actions could take the form of: measures to address ongoing institutional problems (e.g., Option 2.4); measures to seek ''cooperative, ' rather than legislative, clarification of uncertainties (e.g., Option 1.5); or legislative measures to amend current copyright and patent statutes (e.g., Option 1.1), or create *sui generis* protection (e.g., Option 1,4). Depending on the specific action or actions taken, the overall effect might:

1. explicitly affirm the status quo and course of case law (e.g., in terms of the scope of copyright and patent);
2. make small adjustments at the margins of copyright and patent (e.g. through procedural changes);
3. clarify or modify the scope of patent and/or copyright (e.g., through definitional changes), but leave the basic paradigms the same;
4. introduce one or more *complementary, sui generis* regimes tailored specifically to certain aspects of programs and software functionality, recognizing particular patterns of innovation; or
5. develop a *sui generis* regime for software to *substitute for* copyright and/or patent protection, tailored to encompass program code as well as software design and functionality.

These alternatives appear to impact increasingly on the present intellectual property system as the list progresses. But it is not necessarily the case that the least disruptive choices (e.g., explicitly afirming the status quo) can, or should, be selected more quickly than the others. Any of these must result from careful deliberation and crafting, in order to specify clearly and unambiguously what is and is not covered, and what exceptions, if any, are to be made (e.g., along the lines of reverse engineering, fair use, etc.).[109] The following sections discuss the policy issue areas in the context of the above choices for congressional action.

Policy Area 1: Difficulties that the functional aspects of computer programs present in determining the appropriate scope of copyright protection

The functional aspects of computer programs pose difficult questions for application of the copyright law, most notably the appropriate scope of copyright. The traditionally ''fuzzy" line between idea and expression in copyright law is complicated by the need to determine an appropriate scope of protection in order not to cover for the program procedure, process, system, method of operation, or concept, contrary to the intent of the current law (17 U.S.C. l02(b)).

Despite the advantages of incremental accommodation within the copyright law, there may be a point where it becomes preferable to augment or complement the existing framework rather than extend the scope of copyright to fit software-perhaps, cumulatively, at the expense of other types of works. Sometimes, what is in question is the extent to which copyright (perhaps, in concert with trade secret law) is to be interpreted to protect the functional and design aspects of the program in addition to the code. This is the essence of the current debate over application of copyright to features of a program's external design and/or user interface, and in the debate over reverse engineering and recompilation. If or when Congress decides to take action, options include the following:

Definitional Issues and the Scope of Software Copyright

To clarify the scope of copyright protection for software beyond the code, Congress might want to explicitly include or exclude one or more aspects of software, such as computer languages, algorithms, design specifications, user and other interfaces. If it

108 Under this strategy, Congress might also conduct Series of hearings on the issues and assess the results of the executive branch review of the PTO.

109 For discussion of how this was accomplished for the SCPA, within a framework for evaluating proposed changes to the intellectual property system, see Kastenmeier and Remington, op. cit., footnote 30, esp. pp. 438451.

chose to explicitly include one or more of these, Congress could:

Option 1.1a: Expand upon the **Copyright Law's current language on** "subject matter of copyright" in Section 102 to specify that [computer languages/algorithms/design specifications/user and other interfaces/. ..] are copyrightable subject matter.

Alternatively, if it chose to make explicit exclusions, Congress could:

Option l.lb: Expand upon the Copyright Law's current language on "subject matter of copyright" in Section 102 to specify that [computer languages/algorithms/design specifications/user and other interfaces/. . .] are not copyrightable subject matter.

In order to do this, statutory definitions for software-specific terms like ''computer language,' ''algorithm, ' etc. would have to be developed for Section 101, along with means to keep them current or update them as technological changes require. This would not be easy, and would require input from the technical, as well as legal, communities to ensure that the resulting language is unambiguous.

If (under Option l.lb) Congress chooses to explicitly exempt any of these from inclusion within the scope of copyright, then it must determine whether they are to be left to the public domain, trade secret law, patents, or to new, *sui generis* laws (see Options 1.3 and 1.4 below).

Alternatives to Current Treatment of Programs as Literary Works

Option 1.2: Establish a separate category in the Copyright Act for "computer programs," instead of treating them as literary works.

In the international arena, however, this is counter to current U.S. standards of "adequate' protection for software as or akin to a literary work. Also, there would be a period of uncertainty as a new body of case law developed. An advantage of this approach would be that the courts would not have to apply the same principles to software and other literary works, whose economics, patterns of innovation/dissemination, and useful life spans are quite different. Using this approach, though, the term of protection would remain the same as for other copyrighted "literary and artistic' works.

Instead of establishing a separate category for software within Section 102(a)), Congress might limit the scope of "literary" copyright to the code, with the possibility of adopting a *complementary* regime for elements of software design and functionality.

Option 1.3: Leave "computer programs)' within the category of literary works but establish legislative bounds holding the extent of copyright protection as a literary work to the code (as text), not to the behavior of the program when it is executing or to "interfaces.'> Determine whether the latter are to be covered by a complementary, *sui generis* regime.

This option would continue to allow copyright protection for the program code both in the United States and in other countries, under the provisions of Berne. However, it would leave room for a different mode of protection for elements of design and functionality, such as the program's external design and the design of user interfaces. Alternatively, these could be left to the public domain except as protected by patent and/or trade secret law.

At this time, it is not clear to what extent the courts will find these elements to be protected by copyright law. Protecting them explicitly through a new, *sui generis* law would reduce uncertainty about their protection, and provide for features not permitted under copyright, such as a shorter term length (but long enough to allow some reasonable lead time), or compulsory licensing to facilitate standardization. A *sui generis* law might also have software-specific criteria for infringement or for exemptions and could impose a different threshold standard for innovation than copyright's ''originality' criterion.

A disadvantage of this approach, as for other options, is that it may be difficult to define what is and what is not covered under copyright and under the new, *sui generis* law. A new body of case law would have to develop, as would international agreements, particularly regarding the *sui generis* mode of protection.

Another option, which represents a more significant change from the existing modes of protection for software, is to *replace* copyright protection for

software with a new form of protection tailored especially for software.

Option 1.4: Withhold copyright protection from "computer programs" and substitute protection under a *sui generis* framework, including protection for the program code, as well as other elements of program functionality and design.

This approach would replace copyright protection for software with a *sui generis* regime tailored to the protection of computer programs. The new law could address both issues of scope and of reverse engineering.[110] It could either explicitly include or exclude ''interfaces and could determine under what circumstances, if ever, reverse engineering was permissible. A different length of protection could reflect possible differences in market life or purchase patterns between computer programs and traditionally copyrightable works. The availability of protection tailored especially for software might also decrease use of patent protection for some software-related inventions.

As with the *sui generis* choice in Option 1.3, it may be difficult to define what is and is not covered under the new law. Another disadvantage of a *sui generis* law designed to substitute for copyright is that there would be much uncertainty during a transitional period, as the case law develops. In addition, as with other sui generis options, there would be no established international treaty structure.

Reverse Engineering

The issues and uncertainties concerning reverse engineering might be handled by clarifying or modifying the scope and subject matter of copyright as described in Option 1.1 or Option 1.4 above. Another alternative would be to clarify the existing statute concerning ''fair use' (Section 107) and existing limitations on exclusive rights in computer programs (Section 117). This clarification could be accomplished through cooperation or through legislation. *A "cooperative approach could either stand alone or be a precursor to legislation. This type* of approach has been used in the past to reduce uncertainties about the acceptability of certain photocopying practices.[111]

Option 1.5: Direct the Copyright Office, with the assistance of software producers, software consumers, educators, and representatives of the public at large, to develop practical guidelines regarding "fair use" of programs and "essential steps in the utilization" of programs. These guidelines should address what reverse engineering practices, if any, are permissible.

If Congress decides to pursue a legislative approach to deal with the uncertainties surrounding reverse engineering, it might:

Option 1.6: Establish legislative guidelines regarding "fair use" of computer programs and Section 117, especially with respect to fair use for unpublished works (source code as trade secret, object code "published' and reverse engineering.

Among other things, these measures would establish whether incidental copies made during the course of reverse engineering are or are not copyright infringements and/or the extent to which factors such as the purpose of reverse engineering, whether or not a resulting program has taken protected expression from the first, etc. should be taken into account. Legislation might develop from study and input from the industry and the public, whether conducted under Option 1.5 orotherwise.[112]

Policy Area 2: Difficulties in determining the scope of patent protection for software-related inventions and algorithms and the considerable technological and institutional challenges the U.S. Patent and Trademark Office faces in examining applications in these areas of art

[110] For example, the new law could have a shorter span of protection than copyright, a higher standard of originality, compulsory licenses, and special provisions for reverse engineering.

[111] CONTU recommended that the Register of Copyrights and others periodically study and report on photocopying practices in and out of libraries. Based on lengthy consultations with concerned parties, CONTU developed guidelines for library photocopying that were intended to be fair and workable. (*CONTU Report,* op. cit., footnote 16, ch. 4, See also 17 U. S. C. A., sec. 108, pp. 136-137.)

See also footnote 119 and accompanying discussion.

[112] For example, Section 108 of the Copyright Act provides detailed guidance for library and archival photocopying; in developing the language for that section, the conferees agreed that CONTU's guidelines were a reasonable interpretation of sec. 108(g)(2). (17 U. S. C. A., p. 136.)

Whether and to what extent software-related inventions are the subject of utility patent protection has been an issue before the courts and the PTO since the early 1960s. The U.S. Supreme Court has examined the issue of patentability of software on a number of occasions *(Gottschalk v. Benson, Parker v. Flook,* and *Diamond v. Diehr),* attempting to delineate the limits of patentable subject matter. The PTO faces considerable challenges in examining applications involving computer processes. If or when Congress wishes to take action, options include the following:

Statutory Subject Matter

Under current interpretations of patent law, patents may be granted for certain parts of a program's function. The same program may embody many patentable inventions, or none at all, depending on which parts of the program function are novel, nonobvious, and meet the requirements for statutory subject matter.

The case law and PTO guidelines indicate that patents may not be granted for a program function called a‘ ‘mathematical algorithm’ (see footnote 20, supra). The definition of ‘mathematical algorithm’ has had considerable discussion; currently it seems to refer to a program function that is a “mere calculation. ” According to PTO guidelines, claims that include calculations expressed in mathematical symbols contain a mathematical algorithm.[113] On the other hand, the function is not considered ‘‘mathematical’ if it can be stated in terms of its operations on things in the ‘real world. [114] Over the past decade, patents have been issued for software-related inventions such as linear-programming algorithms, spell-checking routines, logic-ordering operations for spreadsheet programs, brokerage cash-management systems, and bank college-savings systems. To some industry observers, there appears to be varianc--or, at least, uncertainty on their part-in how PTO guidelines are being applied during examination.115

To reduce uncertainties and clarify legislative intent, Congress could explicitly address the question of patentability for software-related inventions and for certain algorithms. However, either of the options below would face even more difficult definitional problems than those of the copyright options. For example, a good deal of the software debate has focused on whether PTO should grant “software patents. ’ The term “software patent” does not correspond to any PTO category (see footnotes 19, 89 and 90, supra). As it is used in the debate, ‘software patent’ appears to refer to patents that can be infringed by a computer program executing on a general-purpose computer.[116] However, this class of inventions includes more than just ‘‘software patents. It also includes, for example, traditionally patentable processes which happen to employ a computer.117

The need to make the distinction between nonstatutory “mathematical algorithms’ and statutory inventions results from the courts' efforts to interpret, in the context of software-related inventions, the patent doctrine that “laws of nature” are not statutory subject matter. Any effort to redraw the line between statutory and nonstatutory software-related inventions is likely to encounter serious definitional problems.

In addition, the types of processes and apparatuses that typically are the subject of “software patents” can be claimed in a way that covers both hardware and software implementations. At present, the form of implementation (hardware or software) does not deterrnin e whether an invention is statutory subject

---

113 *U,S, patent* and Trademark Office, *Computer Programs and Mathematical Algorithms,* September 1989, p. *8.* However, the PTO guidelines state that a‘ ‘mathematical algorithm” may be present in prose form, without the presence of mathematical symbols: ‘‘It is not always possible to determine by inspection of the claim whether it indirectly recites a mathematical algorithm; in such instances the analysis ‘requires careful interpretation of the claim in the light of its supporting disclosure. ’ *Johnson,* 589 F.2d at 1079, 200 USPQ at 208. ” Ibid. Despite the presence of a mathematical algorithm, the claim may be statutory. See discussion of the ‘‘two-part test‘‘ in ch. 4 and in pp. 5-12 of U.S. Patent and Trademark Office, *Official Gazette,* Aug. 9, 1989.

114 *In re Bradley,* 600 F.2d 812 (C. C.P.A. 1979). Inventions that process architectural symbols (*In re Phillips,* 608 F.2d *879 (C.* C.P.A. 1979)) or translate languages (*In re Toma,* 575 F.2d 872 (C.C.P.A. 1978)) were not found to be “mathematical.”

115 See footnote 85, supra (ADAPSO, op. cit.).

116 Inventions of this type belong t. a genre of inventions PTO refers to as ‘‘computer-related inventions’ or ‘computer-implemented process patents” (see footnotes 86 and 90, supra). In the request for comments for the Advisory Commission on Patent Law Reform, PTO also used the term, “computer program-related inventions” *(Federal Register, vol. 56, No. 95,* May 16, 1991, p. 22702-22703).

117 An example would be the rubber-curing process found to be statutory by the Us. Supreme Court in *Diamond* v. *Diehr.* There does not appear to be much public concern about these types of inventions.

matter. The only issue is whether the inventor is attempting to claim a ''mathematical algorithm. " However, if software implementations were not statutory subject matter or could not infringe a hardware implementation, then some hardware-based inventions could have their value appropriated by software implementations.

Option 2.1: Refine the statutory definition of patentable subject matter to provide guidance for the courts and PTO. Legislation might address the extent to which processes implemented in software or "mathematical algorithms" are or are not statutory subject matter. Legislation might also address the issue of special exemptions, such as for research and education.

Option 2.2: Exclude software-related inventions and/or algorithms from the patent law and create a special, *sui generis* protection within a patent framework for some inventions. The latter might have a shorter term, lower criteria for inventiveness, and/or special exemptions from infringement.

Other measures could address some of the challenges facing PTO regarding the database of prior art and the timeliness and quality of examination:

Prior Art and Examination Quality and Timeliness

PTO has been grappling with institutional problems such as examiner training and turnover, length of pendency periods for patent applications from filing to issuance, a backlog of applications, and the quality and extent of the prior art database.[118] OTA views these problems as serious since they may affect the quality of the patents issued and create additional burdens for software developers and users (e.g., "landmine" patents—see box l-C). A major problem for patent-system administration with respect to software-related inventions and algorithms is the incomplete stock of ''prior art' available to patent examiners in evaluating patent applications for processes involving computers, especially those involving software and algorithms.

"Filling in" the prior art database (with both patent and nonpatent prior art) is important for improving the quality of examination. Improving electronic search and retrieval capabilities for PTO'S own database is also critical, because it is used by PTO examiners during the application process and is also used by the public. The public's access to an adequate prior-art database is crucial because it allows software developers to review the status of the art and to determine what has already been covered by patent. Given this information, developers can make more informed decisions about their design alternatives, their choice of patent protection versus trade secret, and what might be worthwhile areas for further research and development.

Measures to address the quality of the prior-art database and examination (e.g., Options 2.3 through 2.6) will benefit PTO and the public, no matter what other legislative options are chosen. Because they require some lead time, Congress might select a strategy of initiating one or more of these options now:

Option 2.3: Encourage establishment of a supplementary repository of nonpatent prior art, either public or private.

Patent examiners for the most part are limited to prior art that is already the subject of patent protection. Unlike other areas of technology, software prior art consists in large part of software products that are not a part of the PTO prior-art database. Such a supplemental prior-art database would expand the background against which examiners would compare patent applications, and would allow the software community to inform the PTO of art which is in the public domain but of which the PTO would not be aware because it is not patented. This would prevent the patenting of art which is arguably "old."

Whether publicly or privately developed and maintained, a supplemental repository of prior art would allow members of the software community to participate in upgrading the bank of prior art for software and thus assist in maintainingg the quality and legitimacy of specific, issued patents for software-related inventions. Public access to such a database might also be encouraged, allowing developers to track innovation and make decisions about future

118 The Department of Commerce has established an Advisory Commission on Patent Law Reform, due to report in 1992. The commission is examining 13 sets of issues regarding the patent system; these include protection of what PTO terms "computer-related inventions, " as well as procedural matters such as a first to file system, automatic publication of applications, and the term of patent protection.

research and development and about intellectual property protection for its products.

One nongovernmental response to these needs is the effort to establish a nonprofit Software Patent Institute to provide a supplementary repository of prior art for PTO and others and to offer educational and training opportunities for PTO staff.[119] Congress might wish to monitor the progress of this effort in order to determine what role, if any, government should play.

Option 2.4: Encourage PTO to include in its ongoing automation program means for improved data retrieval, such as flagging, cross-indexing, etc. Encourage the PTO to use this improved database and increased access to the prior art to monitor activity and trends in "hot" areas of art, as well as anticipate and plan for changes in staffing and expertise.

Examiners make comparisons between the invention described in the patent application and the prior art in order to make determinations about the novelty and nonobviousness of the invention described in the patent application. Including the means to cross-reference patents among different areas of art would give examiners easier and better access to pertinent prior art. Increased awareness of the nature of patents for software-related inventions issued by PTO and increased access to those patents by examiners would improve the quality of the examination with respect to nonobviousness and novelty.

In addition to these benefits, this system would enable the PTO to review the trends in prior art both generally and within specific areas of art to determine staffing needs (numbers and skills of examiners). Matching examiner staffing levels and expertise to the changing quantity and character of the influx of applications could improve the quality and efficiency of the examination process and, as a result, yield a higher quality of issued patents. Such a system would also improve PTO'S ability to respond to questions generated for congressional oversight.

Option 2.5: Encourage PTO'S ongoing efforts to improve funding, training, and support for PTO examiners, in order to assure high-quality examinations. Examination quality depends both on the clarity of the examination guidelines and on training and support for examiners.

The PTO has cited high turnover of examiners—particularly in the computer-related art units-as a concern. Well-trained examiners familiar with the prior art and the examination guidelines are important to the quality of issued patents. Steps are needed to induce or enable highly trained examiners in the computer arts to stay within the PTO, and avoid high turnover and the expense and delay of training new hires.

Option 2.6: Encourage PTO to continue to seek input from the software community in formulating examination guidelines, developing classification systems, anticipating technical change, improving the prior art database, and determining appropriate examiner qualifications. Expand efforts to communicate PTO practices and guidelines to the software community, especially in the period following new Court of Appeals for the Federal Circuit cases.

PTO/software community communication could be improved by establishing an external computer science and software engineering advisory committee for IWO, with balanced representation. PTO might also seek technical review from experts in government (e.g., at the National Institute of Standards and Technology) in reviewing changes to examination guidelines.

A procedure for challenging software-related patents (presenting additional prior art to the PTO) on an expedited basis could aid in the development of the prior-art database. Challenges could take place prior to issuance of a patent, or shortly thereafter, and could help ensure that patents not be issued for developments that are in fact well-known or well-established ''inventions. '

Option 2.7: Because of gaps in the prior art, either: expedite challenges to newly issued patents in the software area or establish procedures for preissuance challenges.

Publication of applications after 18 months, whether or not a patent issues, would make the subject matter available to the public. If the subject of the application is deemed patentable by the PTO,

[119] See ch. 2 discussion of efforts to establish a Software Patent Institute.

the published information becomes proprietary. If not, and the application is not withdrawn, the information becomes part of the prior art. The major advantage of this approach is the avoidance of "landmine" patents that issue after years of pendency. This system of publication would allow developers to avoid investing in research and development on technology that is already the subject of an application and, potentially, covered by a patent.

Option 2.8: Because of application backlog and pendency problems, and the possibly shorter market life of software, publish applications after 18 months, whether or not issued, or provide PTO with resources to shorten pendency for software-related patent applications to 18 months.

Another consequence of this system could be a reduction in the number of applications for patents, as some developers chose to avoid the risk of losing trade secret protection for their inventions. If the pendency for patent applications could be shortened, such an approach would likely be unnecessary, as the problem of "landmine" patents would be reduced by timely issuance of patents.

Policy Area 3: Complications facing libraries and other developers and users of digital information, **especially mixed media** works from several different categories of copyrightable works

Decisions affecting intellectual property and software can also affect digital information and the industries, individuals, and institutions that create and use it. Government may have a role in clarifying "fair use' with regard to digital information. Guidelines might be developed to clarify the rights of libraries to make archival copies of digital information or to provide copying and other services to patrons (computer networks allow patrons to be people miles away, not just those within library walls). The rights of libraries to lend, archive or share traditional materials have been well established in the copyright law, but in the case of digital information are often defined by contracts with information providers.

Clarification might also be provided on the extent to which computer-based catalogs can be enhanced by incorporating material from underlying works. While such clarification might be made through legislation, several sets of nonstatutory fair-use guidelines exist,[120] and it may be useful to update them or to develop additional ones through consultation with users and other interested parties. Publishers rely on contracts specifying what users may do with data to deal with the uncertainties about what users' rights are to ''download, ' ''use, ' or 'copy' data under fair-use principles of the copyright law. Clarification of users' rights under copyright could simplify or reduce the need for such contracts. Guidance would, however, have to be carefully crafted for a wide variety of users, products, and technologies. If or when Congress decides to take action, options include the following:

Use of Works in Electronic Form

Option 3.1: Clarify "fair use" guidelines with regard to lending, resource sharing, interlibrary loan, archival and preservation copying, copying for patron use, for works in electronic form.

Option 3.2: Establish legislative guidance regarding "fair use" of works in electronic form and what constitutes "copying" and "reading" or ''using. "

These clarifications would reassert Congress's intention, as expressed in copyright law, to establish limitations on the rights of copyright holders and to permit certain uses of information for research and educational purposes. *Alternatively, a ' 'cooperative' alternative that might or might not lead to subsequent legislation would be to:*

Option 3.3: Direct the Copyright Office, with the assistance of producers and users of electronic information, to develop and disseminate practical guidelines regarding "fair use" of works in electronic form and what constitutes "copying" and "reading" or "using."

Whether established through legislation or through nonstatutory, cooperative guidelines, these clarifications would require careful crafting, with

120 Guidelines for fair use related to educational and nonprofit Org animations were incorporated into the House Committee Report prior to the enactment of the 1976 Copyright Act. Another set concerning off-air taping of broadcast television was approved by the House Subcommittee on Courts, Civil Liberties and the Administration of Justice (Congressional Reeord Sec. E?4751, Oct. 17, 198 1).

input from all interested parties—users, as well as producers-in order to cover the wide variety of users, products, and technologies existing now and in the future.

Multimedia Works

Multimedia works raise questions in two areas. First, what type of protection should they be afforded-as single works or as collections of different works in different categories. The second deals with the incorporation of copyrighted works in a mixed media work. Guidelines may be needed to determine what rights should be obtained, for example, in determining whether a multimedia presentation on a personal computer constitutes a public performance or merely an adaptation of the music or drama incorporated within.

option 3.4: Clarify the status of mixed media works with regard to their copyright protection.

Permissions and Royalty Collollection

The difficulty of obtaining permission for including images, text, or other copyrighted works incorporated into multimedia databases could be eased by the creation of a royalty collecting agency or clearinghouse (or perhaps several agencies for different types of works). The ability to deal with one source, or a small number of sources, for permission to use these works would aid development of multimedia projects, as would creation of standards and conventions about the royalties to be paid.

Option 3.5: Create, or encourage private efforts to form, clearing and royalty collection agencies for groups of copyright owners.

The responsibility for creating such agencies probably rests with the rights holders, rather than with government. There may, however, be a role for government in easing antitrust or other regulations to encourage the creation of such organizations.

Preparing for the Future

Each of the principal policy areas and intellectual property issues discussed in this report is complicated by the complexity of software and computer technologies and by the rapid pace of change in these technologies. Congress and the courts could begin to benefit now from institutional means to understand and remain current about the emerging issues surrounding them.

Establishing a "Congressional Commission on Computing Technologies" would help. The commission (composed of technology experts from academia and the private and public sectors) could be charged with monitoring the fields of computer and software technologies and reporting periodically (perhaps each Congress) on the status of, new directions in, and problems facing these fields. The commission reports could be helpful to Congress in anticipating future areas of policy concern and topics for further study.

Courts are asked to make difficult decisions about technology that is new, changing, and complex. *Congress might consider measures that the courts would find helpful as they deal with complex cases involving computer and software technologies.* Such measures might include special software and computer technology courses tailored for the judiciary, use of special masters, and other means for educating the courts and keeping them abreast of developments in hardware/software technologies. The largely nontechnical judicial staff-judges and clerks—could, in this way, be tutored about current computer/software technologies, thus contributing to a high standard of judicial decisionmaking. Special masters, focused in the area of computer litigation, could provide understanding and expertise on a continuing basis, and make technical determinations when needed.

For the longer term, Congress might also explore the merits of establishing a ''faster track" for intellectual property litigation concerning software and software-related inventions. Software products have a relatively brief market life. Courts, already burdened by caseloads and crowded dockets must render decisions for a fast changing and complex industry. An expedited time period for software-related litigation would assist the courts in reaching timely decisions.[121]

121 Congress has in the past made special provisions to shorten the time period for litigation for other patented products; the Waxman-Hatch Act provides for a 30-month period for litigation of certain actions for patent infringement involving pharmaceuticals and certain drug and veterinary biological products which must undergo an often lengthy approval process with the Food and Drug Administration. (See 35 U.S.C. 271 and 21 U.S.C. 355(j) (4)(B)(iii).)

Chapter 2

The Law

Contents

	Page
Patents	39
Introduction to the Patent System	39
Evolution of Case Law	45
PTO Policy and Procedural Response to Court Cases	52
Copyright	56
Goals of U.S. Copyright	56
Development of U.S. Copyright	58
The Scope and Nature of Copyright Protection	60
The 1976 Copyright Act and Fair Use	61
The Copyright Office	65
CONTU Recommendations Regarding Registrations and the Rule of Doubt	66
CONTU and the 1980 Amendment	67
Evolution of Case Law	68
Databases	73
Hybrid Design Protection	77
Trade Secret Law	78
Introduction	78
Characteristics of a "Trade Secret"	79
The Need to Maintain the Trade Secret	82
Software and Trade Secret Law	82
Relationships Among Patent/Trade Secret/Copyright Laws	86

Boxes

Box	Page
2-A. Biotechnology Challenges to the Patent System	40
2-B. Court of Customs and Patent Appeals Cases After Prater	48
2-C. Court of Customs and Patent Appeals Cases Following Benson	50
2-D. The Cases of Pardo, Taner, Abele, and Meyer	53
2-E. Analogous Patent Law in Foreign Countries	57
2-F. Early Protection Against Copying	58
2-G. Copying in Renaissance Art	62
2-H. Analogous Copyright Law in Foreign Countries	80
2-I. Contract Law Protection of Electronic and Computer Technology	83
2-J. The Shrink-W:rap License Controversy	85
2-K. Analogous Trade Secret Law in Foreign Countries	89

Tables

Table	Page
2-1. Number of Patents Issued for Software-Related Inventions, 1970-91	55
2-2. Copyright Registrations, Generally: 1974 to Present	65
2-3. Copyright Registration, Computer-Related Works: FY 1986 to Present	65

Chapter 2
The Law

Patents

Patent protection is used extensively today to protect software-related inventions both in the United States and abroad. According to practitioners in the field of computer law, most of the major developers of software are presently using patent protection for software-related inventions.[1] The following is a discussion of the patent system, and a survey of the issues arising from the protection of software-related inventions by patent.[2]

Introduction to the Patent System[3]

The first U.S. patent law, enacted in 1790, embodied Thomas Jefferson's philosophy that ''ingenuity should receive a liberal encouragement." This law provided protection for "any new and useful art, machine, manufacture, or composition of matter, or any new and useful improvement [thereof]. Subsequent patent statutes were enacted in 1793, 1836, 1870, and 1874, employing the same broad language as the 1790 act. The Patent Act of 1952 replaced 'art' with 'process' as patentable subject matter.[4] The Committee Reports accompanying the 1952 act demonstrated that Congress intended patentable subject matter to include "anything under the sun that is made by man. However, the U.S. Supreme Court has held that laws of nature, physical phenomena, scientific principles, and abstract ideas are not patentable.

Patents grant to inventors a limited property right to exclude others from practicing (making, using, or selling) the claimed invention for 17 years. In this way, patents are intended to encourage inventiveness. In the United States, patent law is exclusively within the jurisdiction of the Federal Courts.[5] Strict examination of an invention is required,[6] making the patent the most difficult form of intellectual property protection to obtain. Once obtained, a patent is maintained by periodic payment of maintenance fees during the life of the patent.[7]

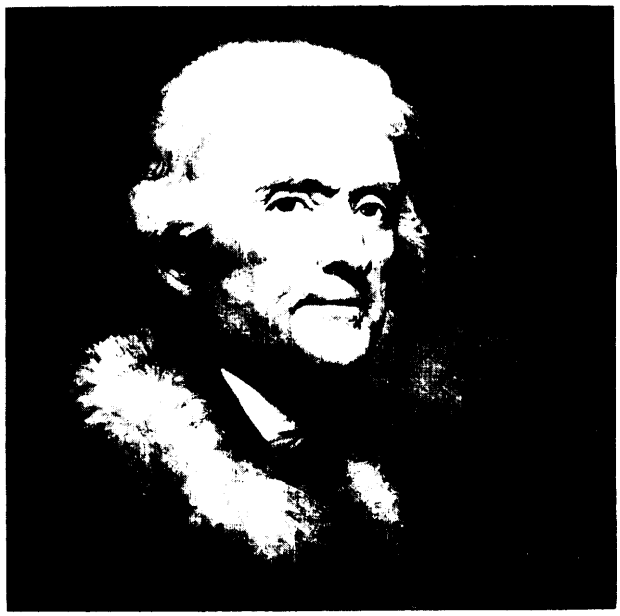


Photo credt: U.S. Library of Congress

The first U.S. patent statute was enacted by Congress in 1790. The patent law embodied Jefferson's philosophy that "ingenuity should receive a liberal encouragement."

---

1 Robert Greene Sterne, Sterne, Kessler, Goldstein and Fox, personal communication Oct. 8, 1991.

2 In this report, OTA sometimes uses phrases like 'patents for software-related inventions, ' ' 'software-related patents, ' or 'patenting algorithms'' to refer generally to patent protection for computer-implemented processes and algorithms. The U.S. Patent and Trademark Office considers terms like "software patents' to be a misnomer because they maybe interpreted to mean that a computer program pa se (i.e., the sequence of coded instructions itself) is patentable, as opposed to the underlying computer process it carries out. (M. Keplinger, G. Goldberg, and L. Skillington, PTO, personal communication, Dec. 18, 1989.)

3 This material is adapted from U.S. Congress, Office of Technology Assessment, New Developments *in* Biotechnology: *Patenting Life,* OTA-BA-370 (Washington, DC: U.S. Government Printing Office, April 1989), pp. 37-43. For further discussion of biotechnology challenges to the patent system, see box 2-A.

4 35 U.S.C. 101.

5 35 U.S.C. 1 et seq.; 28 U.S.C. 1338. Questions related to patent licenses arc governed by State contract law, but maybe decided in Federal or State court, depending on the circumstances.

6 Some observers bel ieve that the level of examination in the computer arts cannot be characterized as strict, particularly in the area of software-related inventions. Richard Stallman, Free Software Foundation personal communication, September 1991.

7 35 U.S.C. 41(b).

Box 2-A—Biotechnology Challenges to the Patent System

Concerns about the integration of new technologies into the patent system are not confined to the area of computer software. In 1989 OTA published a report New *Developments in Biotechnology: Patenting Life, in* which it examined many of the intellectual property issues confronted by this industry. Biotechnology, broadly defined, includes any technique that uses living organisms, or parts of organisms, to make or modify products, to improve plants or animals, or to develop microrganisms for specific uses. Interest in commercial uses of living organisms has increased greatly over the past 15 years or so, spurred by new capabilities to select and manipulate genetic material. Proprietors' interest in obtaining biotechnology patents has increased accordingly. Like the computer software industry, biotechnology has confronted questions of subject-matter jurisdiction, the quality and efficiency of the examination process, disclosure and deposit requirements, and the expense and uncertainty of litigation of patents. In addition, Congress has addressed specific areas where the nature of the field of biotechnology was believed to warrant special treatment for the patents granted.

Subject-Matter Issues

Although there has been steady increase in the number of biotechnology patents issued since the early 1980s (i.e., since *Diamond v. Chakrabarty),* controversy has surrounded the issue of the appropriateness of intellectual property protection for biotechnology. The *Chakrabarty* decision concluded that Congress had intended the patent law to be given wide scope and that genetically engineered microorganisms, defined as useful products of human ingenuity as distinct from manifestations of nature, were patentable subject matter.

The controversy surrounding the issue of patents for biotechnology arises in large part from the very creation and patenting of 'inventions' that are themselves alive. The debate over whether to permit the patenting of living organisms frequently goes beyond the relatively simple question of the appropriateness of patents per se in this field of technology, focusing on the consequences of commercial use of patented organisms or the underlying merits of biotechnology itself. Discussion regarding patenting of genetically engineered organisms can turn to scientific questions, philosophical and ethical issues, environmental concerns, concerns for the welfare of genetically engineered animals, and economic considerations. One difficulty inherent in examining the desirability of these types of patents lies in the need to separate arguments that are new and directly related to patents in these areas of technology from broader arguments that would pertain independent of patent considerations. The 1989 OTA report on *Patenting Life examin*ed subject-matter issues, as well as issues related to patent prosecution and the operation of the Patent and Trademark Office.

Examination Speed and Quality

In the 1989 report, OTA found that the PTO was unable to process the ever increasing number of biotechnology applications in a timely manner. The PTO cited turnover among patent examiners, lured to the private sector by substantially higher salaries, as a significant reason for these difficulties. The number of applications severely challenged the process and examination capabilities of the PTO, despite the reorganization of biotechnology activities into a separate examining group in 1988, so that as of July 1988 there was a backlog of nearly 6,000 applications. The PTO issued just under 1,500 biotechnology patents in 1987. OTA found that approximately 15 months, on average, lapsed before examination of a biotechnology application was begun and another 27 months passed before examination was completed (either by issuance of a patent or abandonment of the application).

Disclosure and Deposit Considerations

OTA found that biotechnology presents a differentiating administrative issue in that it is the only art where words alone may be incapable of describing an invention sufficiently to enable one skilled in the art to make and use it in a reproducible manner. Currently, patent applications for inventions involving biological materials that are not generally available or reproducible without undue experimentation by a person skilled in the art are often supported by a deposit in a recognized patent depository. Although not automatically required, a deposit is employed in many cases to meet the requirement that a patent provide enablement or the 'best mode' of practicing an invention.

Patent Litigation

OTA found that biotechnology patent litigation required courts to assess whether patent holders have met the requisite requirements of novelty, usefulness, and nonobviousness, as well as issues relating to the scope of claims, infringement, and patent enforcement. OTA also noted the cost resulting from the existing uncertainty over the scope of protection, citin,g, for example, the additional litigation necessary to define the parameters of patent

protection. OTA further found that such costs undoubtedly influenced the R&D and intellectual property strategies of many companies: 85 percent of firms surveyed by OTA reported that, even though they might consider patent protection more desirable, they expected to pursue trade secret protection for biotechnology lines in addition to patent protection.

Experimental Use **Exception**

In biotechnology, the most relevant exemption from patent infringement is the experimental **use exception,** a court-created doctrine that holds that an experiment with a patented invention for the sole purpose of gratifying true scientific inquiry or philosophical curiosity does not attack the right of the patentee, and thus does not constitute infringement.

In 1984 the Court of Appeals for the Federal Circuit considered a case involving the testing of a pharmaceutical pursuant to Food and Drug Administration approval for a generic drug equivalent. This testing was conducted close to the end of the patent term for the original drug. The court found that such testing did not fall within the narrow confines of ''experimental use' and thus was an infringement. In the wake of this case, Congress amended the patent code (Public Law 98-417) to allow a statutory exemption with respect to testing human drug products in order to meet FDA approval requirements.

PVPA **Research Exemption**

A complementary form of intellectual property is provided by the Plant Variety Protection Act of 1970 (PVPA), intended to encourage the development of new, sexually reproduced plants (new varieties and hybrids). Upon application to, and examination by, the U.S. Department of Agriculture, a plant variety certificate maybe issued on any novel variety of sexually reproduced plant, other than fungi, bacteria, or first-generation hybrids. Amendments in 1980 added protection for 6 vegetable crops and extended the period of protection to 18 years so that PVPA would be consistent with international plant-breeding provisions.

The PVPA has two important exclusions to a certificate holder's protections: a farmer's exemption and a research exemption. The farmer's exemption allows individuals whose primary occupation is growing crops for sale (rather than plant breeding) to save protected seed for use on their farm or for sale to other such individuals. **The research exemption precludes a breeder from excluding others from using the protected variety to develop new varieties.** By contrast, utility patents for plants (which have been granted since 1985) do not have statutory exemptions from infringement: the holder of a plant utility patent can exclude others from using the patented variety to develop new varieties.

SOURCE: U.S. Congress, Office of Technology Assessment New *Developments in Biotechnology: Patenting Life, OTA-BA-370* (Washington, DC: U.S. Government Printing Office, April 1989), pp. 37-43.

An invention must fulfill five major requirements to be deemed patentable.[8] The invention must: 1) be patentable subject matter; 2) be useful; 3) be novel; 4) not have been obvious at the time it was made; 5) be disclosed in sufficient detail to "enable persons of ordinary skill in the art' to practice the invention without ''undue experimentation. The patent must also disclose the "best mode" of practicing the invention known to the applicant as of the filing date. These issues of *statutory subject matter, utility, novelty, nonobviousness,* and *enablement/best mode* are addressed by 35 U.S.C. 101, 102, 103, and 112 respectively. If it is determined that these requirements have been fulfilled, and if certain other formal requirements are met, a patent will be granted. The grant of a patent by the U.S. Government to an applicant gives a property right from the U.S. Government to the applicant or the applicant's assignee.

Subject Matter

A patent may issue to:

> [w]hoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof. ..9

[8] *Application of Bergy*, 596 F.2d 952 (C.C.P.A. 1979). *It* should also be noted that it *is* the burden of the Patent and Trademark Office to show nonpatentability, rather than the burden of the applicant to show patentability. If, however, the PTO makes out a *prima facie* case of nonpatentability, the burden of prwf then does shift to the applicant.

[9] 35 U.S.C. 101,

Patents of this kind are known as *utility patents,* and are divided for examination purposes into three classes by the U.S. Patent and Trademark Office (PTO): chemical, electrical, and mechanical. Patents for software-related inventions are classified within the electrical class, relating to data processing methods and apparatus.

Under section 101, the invention must be

- characterizable as a process, machine, manufacture, or composition of matter,
- a new invention, or a new, useful improvement of an existing invention; and
- useful.

Congress and the courts have interpreted the classifications of process, machine, manufacture, or composition of matter broadly. Although the subject matter of things that may be patentable is broad, it is not unlimited. Laws of nature, physical phenomena, scientific principles, and abstract ideas cannot be patented .10 The Supreme Court, in the case of *Parker v. Flook,* stated that the rule that discovery of a law of nature cannot be patented is based not on the belief that natural phenomena are not processes, but on the concept that natural phenomena are not the kind of 'discoveries" that patent law was designed to protect. The court stated that mere recognition of existing phenomena or relationships carries with it no rights to exclude others from their use or exploitation. [11]The issue of what constitutes patentable subject matter has been especially problematic in the area of software, as will be discussed below .12

In addition to the types of utility patents permitted under section 101, U.S. law provides for two other types of patents:

- *Design patents*[13]*—A* design patent may issue to the inventor of any new, original and ornamental design for an article of manufacture. Unlike other types of patents (which have a term of 17 years from grant), design patents have a term of 14 years from grant.
- *Plant patents* [14]—A plant patent may issue to the inventor of any distinct and new variety of plant which is capable of asexual reproduction.

Utility

In general, utility is easily shown by the patent applicant. Utility is demonstrated by experimental data, commercial use, or through the drawings or description of the patent application.

Novelty

The requirement for patentability that an invention be novel is set forth in section 101 and is addressed in section 102 of the Patent Act. Under the provisions of section 102, an invention should not have previously existed through the work of others. The specific provisions of section 102 require that in order to qualify as prior art there be some public aspect to the previously existing work of others; inventions concealed through trade secret protection do not preclude patent protection on grounds of lack of novelty or nonobviousness.[15]

Under section 102, a patent can be denied under several conditions, including:

- the invention was known or used by others in the United States or patented or described in a printed publication in the United States or a foreign country before the invention date claimed by the applicant for patent;
- the invention was patented or described in a printed publication in the United States or a foreign country, or sold or used in the United States more than 1 year prior to the date of the application for a patent in the United States;
- the invention was abandoned by the applicant;
- the invention was made in the United States by another person who has not abandoned, sup-

10 *Diamond* v. *Chakrabarty,* 447 U.S. *303 (1980); Funk Brothers Seed Co.* v. *Kalo Inoculant Co.,* 333 U.S. 127 (1948); *Gottschalk v. Benson,* 409 U.S. 63 (1972).

11 *Parker* v. *Flook, 437* U.S. 584 (1978).

12 Biotechnology faces similar concerns about integration into the patent system as does the area of software. For further discussion of the problem of subject matter, as well as a brief survey of problems of the PTO in maintaining the quality and efficiency of the patent examination process for biotechnology patents, determining requirements for disclosure and deposit considerations, issues confronted in litigation, and special exceptions to the patent law as applied to biotechnology, see box 2-A.

13 35 U.S.C. *171-173.*

14 35 U.S.C. 161-164.

15 Acts by an inventor to withhold an invention from public knowledge by either: 1) deliberately hiding the invention, or 2) failing to apply for a patent within a reasonable time after invention, *may* extinguish a first inventor's priority of invention such that a patent is granted to a later rival inventor. (35 U.S.C. 102Q).)

pressed, or concealed it (in such cases determining the priority of invention becomes important); and
- the applicant is not the inventor.

Nonobviousness

To be patentable, an invention must be found to be not obvious under the terms of 35 U.S.C. 103, which states that a patent may not be obtained:

> ... if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which the subject matter pertains. [16]

Obviousness addresses the degree of difference between the invention sought to be patented and the inventions or technology that are known or available (the "prior art") to the hypothetical person skilled in the relevant field of technology. Evidence of prior art (e.g., existing patents, publications) is evaluated not only for what it expressly teaches, but also for what it would reasonably suggest to one of ordinary skill in the relevant field of technology. Since an invention may be new (novel) but still be obvious, a determination as to whether or not the proposed invention is obvious needs to be made. The Supreme Court set forth the test for obviousness in 1966:

- determine the scope and content of the prior art;
- ascertain the differences between the prior art and the claims at issue; and
- resolve the level of ordinary skill in the pertinent art.[17]

In addition, the Court stated that secondary considerations of nonobviousness such as commercial success and long felt but unsolved industry need may be relevant to particular situations.[18]

How a Patent Is Obtained

An application for a patent must generally be made by the inventor. It must be in writing, contain a specification, and, where necessary, a drawing. The application must include claims and an oath or declaration that the inventor believes himself or herself to be the original and first inventor of that for which the protection is sought.



Photo credit: U.S. Library of Congress

The "model room" of the old Patent Office, where inventors deposited models of their inventions.

The specification is the written description of the invention, describing the manner and process of making and using it ''in such full, clear, concise, and exact terms" as to enable any person skilled in the art to which it pertains to make and use the same, and setting forth the "best mode contemplated by the inventor' (at the time of the application) of carrying out the invention (35 U.S.C. 1 12). The specification includes a set of one or more claims, each of which particularly points out and distinctly claims the subject matter which the applicant regards as his or her invention. The claims represent the metes and bounds of the property to be protected. As in real property, the claims stake out the patent holder's territory, and any encroachment on that territory constitutes infringement.[19]

The purpose of the "best mode' requirement of section 112 is to prevent inventors from applying for patents while concealing from the public the preferred embodiment or implementation of the inven-

Is 35 U.S.C. 103.

17 *Graham* v. *John Deere Co.,* 383 U.S. 1 (1966).

18 Ibid.

19 B.A. Amernick, *Patent Law for the Nonlawyer: A Guide for the Engineer, Technologist, and Manager* (New York, NY: Van Nostrand Reinhold Co., 1986).

tion. The meaning of this requirement in the context of computer software has been explored by the courts. The claims in *In re Sherwood*[20] were rejected by the examiner for failure to disclose any computer hardware, flow charts, algorithms, or programs with which best mode would operate. Yet the court overturned that determination, asserting that the question is not *how an* applicant discloses the best mode, but *whether* he has done so. The Court of Customs and Patent Appeals (C. C.P.A.) concurred, stating:

> [T]here is no objective standard by which to judge the adequacy of a best mode disclosure. Instead, only evidence of concealment (accidental or intentional) is to be considered. That evidence, in order to result in affirmance of a best mode rejection, must tend to show that the quality of an applicant's best mode is so poor as to effectively result in concealment.[21]

Notwithstanding the applicant's failure to disclose the listing of the known program, the disclosure was sufficient to satisfy the best mode requirement.

In *White Consolidated Industries, Inc. v. Vega Servo-Control, Inc.*,[22] a program that was an essential element to a claim for machine tool had been identified in the patent specification as an example. The program was not disclosed in the specification, but rather was maintained as a proprietary trade secret. The court held that, absent disclosure of the program, the specification required a great deal of experimentation by a skilled programmer to develop a workable program to make the invention operational. The proprietary nature of the program was irrelevant, given that it was the only way of disclosing the best mode of practicing the invention. The U.S. Court of Appeals for the Federal Circuit invalidated the patent on the ground that the disclosure failed to satisfy the enabling requirement under 35 U.S.C. 112.

The patent application may be made by the individual inventor, by two or more inventors jointly, by legal representatives of the inventor or inventors, or by a person to whom the inventor has assigned a proprietary interest in the invention.2g The actual filing date of the application is important because the presumption is that patent applications filed and documents published after the priority date do not constitute prior art for purposes of the earlier filed patent application.

Upon filing, the application is assigned to a primary examiner at the PT0, who conducts an examination of the PTO prior-art database as part of determining whether a patent should issue.[24] After the examiner initially reviews the application and the search results, there is an exchange of written correspondence between the patent applicant and the examiner. During this exchange, the examiner will often cite prior-art references in addition to those found and cited in the patent application that limit or preclude patentability of the claimed invention. The examiner will provide these to the applicant, who may then respond with amendments to the claims, information, or arguments to distinguish the claimed invention from the prior art. The back-and-forth procedure in which the applicant attempts to demonstrate the patentability of the claimed invention is called ''prosecuting' a patent application.[25]

If, after the initial examination process has been completed, the examiner determines that any claim of the patent application is unpatentable, the claim is rejected and the applicant is notified of the rejection with an explanation. If the applicant makes a request within 6 months,[26] he or she has a right to automatic reconsideration of the rejection of the claims, after which the examiner may make the rejection 'final. '[27] An applicant whose claims have been finally rejected may appeal the decision of the primary examiner to the Board of Patent Appeals and

---

20 613 F.2d at 816 (C. C.P.A. 1979)

21 *In re Sherwood*, 613 F.2d 809 (C. C.P.A. 1980) at 816.

22 *White Consolidated Industries, Inc.*, v. *Vega Servo-Control, Inc.*, 214 U. S.P.Q. 796 (S.D. Mich. 1982), *aff'd* 713 F.2d 788 @~. Cir. 1983).

23 35 U.S.C. 116-118.

2435 U.S.C.131.

25 J. Drabiak, ''Paten~, Copyrights, and Trademarks: A Primer on Protecting Intellectual Property Work product, ' *Southern Illinois University Law Journal*, vol. 11, No. 1, fall 1986, pp. 1-28.

26 The PTO can and often does set a shorter response period.

27 35 U. SC. 132-133.

Interferences. Each appeal is heard by at least three members of the Board.[28]

If an applicant is dissatisfied with the decision in an appeal to the Board, he or she may file an appeal with the Federal Circuit or file a civil action against the Commissioner in the U.S. District Court for the District of Columbia.[29] Appeals of *inter-partes* interference actions in the PTO to establish the priority of an invention proceed in a similar manner.[30] For the applicant who chooses to appeal to 'he District Court, a new hearing is conducted.[3]1 One advantage of such anew hearing is that the applicant may be able to introduce additional evidence into the prosecution record.

The Term of Patent Protection

In the United States, patents are granted for a term of 17 years from issuance, so long as maintenance fees, which are required only for utility patents, are paid.[32] Design patents are granted for a term of 14 years, For certain utility patents, the term may be extended for an additional 5 years.[33] **For** example, where a patent claims a product which must undergo regulatory review by the Food and Drug Administration prior to approval for commercial marketing or use, the patent may be eligible for such an extension if certain conditions are satisfied. Products eligible for such an extension are limited to human drug products, medical devices, and food or color additives.

Evolution of Case Law

Pre-Benson Patents for Computer Processes and Algorithms

During the early 1960s, the PTO faced a backlog of patent applications and a 4-year pendency for an application prior to issuance as a patent. The PTO and the patent system experienced further difficulties because of a limited budget and processing methods that could not adequately handle this volume of applications. As a result, the President's Commission on the Patent System was established in 1965 to address these problems and suggest revisions to the Patent Act.[34] Reflecting the policy concerns of the PTO, the Commission recommended against patent protection for computer programs. The Commission report stated:

> The Patent Office now cannot examine applications for programs because of the lack of a classification technique and the requisite search files. Even if these were available, reliable searches would not be feasible or economic because of the tremendous volume of prior art being generated. Without this search, the patenting of programs would be tantamount to registration and the presumption of validity would be all but non-existent.

As early as 1964, the Patent Office denied the patentability of programs, characterizing them as ''creations in the area of thought. ' In 1966, the PTO attempted to formulate standards for patentability of software. In its first guidelines for Examination of Programs, the Patent and Trademark Office made recommendations regarding process claims based solely on computer execution of mathematical formulae and process claims called. These recommendations were withdrawn after hearings in which hardware manufacturers opposed the guidelines on the ground that software users must have access to software to promote the development of the technology. Software developers disagreed, maintaining that software was a technology entitled to protection under the patent laws.[35] These recommendations by the PTO came under attack during Senate hearings on the Patent Reform Act of 1967 for denying software manufacturers rights equal to those of hardware manufacturers.[36] During later House hearings, former PTO Commissioner Edward J. Brenner indicated that the PTO lacked sufficient pertinent prior art since there was a lack of prior art patents and most of the literature was not in the Office possession. The Commissioner also noted the diffi-

~ 35 U.S.C. 7, 134.

29 35 U.S.C. 141, 145.

.3035 U.S.C.141,146

31 *Hoover* CO. v. *Coe* 325 U.S. 79 (1945).

32 35 U.S.C. 154.

33 35 U.S.C. 156.

34 Nelson Moskowitz, "The Metamorphosis of Software-Related Invention Patentability," *Computer Law Journal*, vol. III, 1982, p. 281.

35 Ibid., p. 283, citing Report of the Hearings on the Patent Office's Guidelines to Examination of Programs (1966).

36 Ibid., citing Hearings on S. 1042 Before the Subcommittee on Patents, Trademarks and Copyrights of the Senate Committee on the Judiciary, 90th Cong., 1st Sess. (1967).

culty of finding qualified examiners for computer software technology, the lack of experienced personnel, and the inadequate examination procedures in the area of software.[37]

In 1968 the PTO, disregarding the arguments and line drawing of the previous guidelines published in 1966, issued another set of guidelines in which it stated that computer programs, however claimed, were not patentable unless combined with an apparatus which caused the physical transformation of matter. The PTO cited examples of transformation such as knitting a pattern or shaping metal. The legal theory on which the PTO based its guidelines was the doctrine of "mental steps' '—i.e., if the process could be carried out purely in one's mind, the invention was not patentable. As discussed below, the C. C.P.A. rejected these guidelines in *In re Prater,* and the PTO rescinded them.[38]

In 1968, the C. C.P.A. issued a series of decisions testing the limits of the ''mental steps' doctrine and the PTO guidelines. Because a programmed computer performed calculations that, in theory could have been carried out in the human mind, it was thought that the mental steps doctrine precluded the patentability of algorithms.

The C. C. P.A. 'S first decision in this series of cases was *In re Prater.*[40] *The court* reversed the Patent Office Board of Appeals which had affirmed the rejection of appellant's claims to a spectrographic analysis in which linear equations were solved on a programmed general purpose digital computer. In its initial decision the court expressly held that the mental steps doctrine of *In re Abrams*[41] *and In re Yuan*[42] did not preclude the patenting of software. The court held that,

> [P]atent protection for a process disclosed as being a sequence or combination of steps, capable of performance without human intervention and directed to an industrial technology-a "useful art" within the intendment of the Constitution—is not precluded by the mere fact that the process could alternatively be carried out by mental steps.[43]

The court found that in a case where the electronic or mechanical means for performing the process are disclosed in the application and where it is not reasonable to carry out the invention mentally, the invention is patentable subject matter under section 101. Because the applicants had disclosed that a computer, an electronic device, would be used for performing the process, and the process could not reasonably be per-formed in the mind, the court found that the invention was statutory.

On rehearing, the court sidestepped the 35 U.S.C. 101 statutory subject matter issue, and found that the mental steps doctrine did not apply because the applicants disclosed an apparatus for implementing the process without human intervention. The court, in a footnote, asserted that a general purpose digital computer programmed with a claimed process becomes a special purpose digital computer and could qualify as a patentable invention, assuming the requirements of novelty, utility, and nonobviousness are met. For this reason, the court allowed the apparatus claim. However, the court found that the applicants' language failed to limit the process claim to its machine implementation and affirmed the patent examiner's rejection of the specific process claim.[44]

In response to the Prater decision, Con-missioner William E. Schuyler withdrew the PTO'S 1968 guidelines, stating:

> We now will consider patent applications for computer programs on the basis of the merits for the specific inventions sought to be protected rather than refuse consideration for reasons such as those

37 Ibid., citing Hearings on H.R. 5924, H.R. 13951, and Related Bills Before Subcommittee No. 3 of the House Committee on the Judiciary, 90th Cong., 1st Sess., Serial No. 11, pt. 1 at 37 (1967).

38 Moskowitz, op. cit., footnote 34, at P. 284.

39 Ibid., p. 286. Later court decisions distinguished between ''mathematical' and other algorithms. ''Mathematical' algorithms, as defined by the case law, are not statutory. However, some observers believe that patents for mathematical algorithms are, in fact, being granted. They cite U.S. Patent No. 5,031,134, which falls into, they claim, a classical area of mathematics, that of numerical integration. Richard Stallman, The League for Programming Freedom, personat communication, September 1991.

40/n *re Prater*, 415 F.2d 1378 (C.C.P.A. 1968), *aff. on rehearing*, 415 F.2d 1393 (C.C.P.A. 1969).

41 *In re Abrams*, 188 F.2d 165 (C. C.P.A. 1951).

42 *In re Yuan*, 188 F.2d 377 (C. C. P.A. 1951).

43 Ibid., at 1389.

44 *In re Prater*, 415 F.2d 1393 (C.C.P.A. 1969).

discarded by the court in the *Prater and Wei* case.[45]

The C. C. P.A. 'S conclusion in *Prater was* reaffirmed by the C. C.P.A. in *In re Bernhart.*[46] *The* permissible scope of subject matter was extended by the C. C.P.A. to include computer program processes in *In re Musgrave*[47] *in* which the court rejected the application of the mental steps doctrine and set forth a constitutional ''technological arts' standard for determining patentability. The concurring opinion in *Musgrave,* concluded that ''there is very little left of the 'mental steps' doctrine. ' This view was borne out in subsequent C. C.P.A. decisions. (See box 2-B.)

Gottschalk v. Benson

The Supreme Court finally considered the issue of the patentability of computer software in the case of *Gottschalk v. Benson.*[48] The invention at issue in that case was considered frost by the C. C.P.A. in *In re Benson .49* That case was an appeal from the decision of the Patent Office Board of Appeals which affirmed the rejection of two of the claims on the grounds that the claims were not directed to statutory subject matter. One of these rejected claims, claim 8, was for a method to be practiced in part using a particular apparatus called a "reentrant shift register. ' The other, claim 13, contained no reference to any apparatus, and in fact referred to the thing operated on not as ''signals' but a ' 'representation. The method claimed in claim 13 was one for converting 'binary coded decimal number representations" into ''binary number representations. ' The court thus addressed the issue of whether method claims reciting methods of converting binary-coded decimal numbers to binary numbers were statutory under section 101.[50]

The C. C.P.A. found that Claim 8, reasonably interpreted in light of the specification (as directed in *Mahoney*[51]), was for statutory subject matter because the shift register referred to in the claim was part of an electronic computer. The C. C.P.A. found digital computers used to manipulate signals representing binary numbers comparable to conventional cash registers and other machines which manipulate numbers. Since these machines had never been denied patentability, the court held that patentability could not then be denied digital computers.

Analyzing claim 13, which did not recite a shift register, the C. C.P.A. rejected the PTO'S argument that the claim was not restricted to any apparatus and could be carried out directly by mental steps. Rather, the C. C.P.A. found that the claim must be interpreted in light of the specification and concluded that, with regard to claim 13, the "process had no practical use other than the more efficient operation and utilization of a *machine known* as a digital computer. " The court held claim 13 to be within the statutory subject matter of section 101, and concluded in its opinion by stating that computers are ''in the technological field and are in the useful arts, regardless of the uses to which their users may put them.' '[52]

The Supreme Court considered the Benson application and reversed the C. C.P.A. in a brief opinion .53 While the C. C.P.A. considered claims 8 and 13 separately, the Supreme Court's analysis made no distinction between the two. The Court stated that both claims 8 and 13 could be "performed without a computer, ' and that the "claims were not limited. . . to any particular apparatus. The court viewed the claims as broad, and not tied to any particular application. The Court stated:

> Here, the 'process' claim is so abstract and sweeping as to cover both known and unknown uses of the BCD to pure-binary conversion. The end use may (1) vary from the operation of a train to verification of drivers' licenses to researching the law books for precedents and (2) be performed through any existing machinery or future-devised machinery or without any apparatus.[54]

45 Moskowitz, op.cit., footnote 34, p. 287 at footnote 26.

46 *In re Bernhart,* 417 F.2d 1395 (C. C.P.A. 19@).

47 *In re Musgrave,* 431 F.2d 882 (C.C.P.A. 1970). For further discussion of *Bernhart,* and *Musgrave,* see box 2-A.

48 *Gottschalk* v. *Benson, 409* U.S. 63 (1972).

49 *In re Benson,* 441 F.2d 682 (C. C.P.A. 1971).

50 Ibid.

51 *In re Mahoney,* 421 F.2d 742 (C. C.P.A. 1970).

52 *In re Benson,* 441 F,2d 682 at 688.

53 *Gottschalk v. Benson, 409* U.S. at 64 (1972).

54 Ibid., at 68.

Box 2-B-Court of Customs and Patent Appeals Cases After Prater

In re Bernhart[1]

The C.C.P.A. reaffirmed the ***Prater*** holding in ***In re Bernhart. In*** that case, a general purpose digital computer was used to solve a set of transformation **equations** for a claimed apparatus and method which plotted three-dimensional figures onto any desired planar surface. The solutions derived by the general purpose digital computer were used to drive the plotter in drawing the two-dimensional representation from any desired angle and distance. Thus, as in ***Prater, the court*** **confronted the** question of whether a new machine is formed when a computer operates under the control of a program. In response to the examiner's argument that a programmed computer was structurally equivalent to the same computer without that program, and that the addition of new signals to the computer did not make it a new machine, the C. C.P.A. confirmed the patentability of the apparatus claims, and amplified its ***Prater*** footnote, stating:

> . . . if a machine is programmed in anew and unobvious way, it is physically different from the machine without the program; its memory elements are differently arranged. The fact that these physical changes are invisible to the eye should not tempt us to conclude that the machine has not been changed. If a new machine has not been invented, certainly a 'new and useful improvement' of the unprogrammed machine has been, and Congress has said in 35 U.S.C. section 101 that such improvements are statutory subject matter for a patent.[2]

In re Musgrave[3]

The invention of *In re Musgrave* **consisted** of a process for establishing weathering corrections for seismic exploration. The claims include steps such as "generating signals" and "applying corrections. " The Patent and Trademark Office rejected the claims, asserting that none of the claims defined a "process" within the meaning of section 101. The PTO Board of Appeals considered the claims by separating the ''mental steps" from the ''physical steps" and found that the point of novelty was found in the mental steps. Thus, the Board held the process did not fall within the category of patentable inventions.

The C.C.P.A. rejected the view that a mental versus physical distinction could be read into section 101. According to the court, dissecting the claims to arrive at some "point of novelty" was irrelevant to an analysis of statutory subject matter of the process under section 101. The court explicitly stated that:

> As maybe seen from the statutory language, it contains nothing whatever which would either include or exclude claims containing 'mental steps' and whatever law may be on the subject cannot be attributed to Congress.

1 417 F.2d 1395 (C.C.P.A. 1969)

2 Ibid., at p. 1400.

3 431 F.2d 882 (C.C.P.A.1970)

The Court characterized the Benson method as a "mathematical formula' and went on to rule that processes are not patentable if they only claim or "preempt" a mathematical formula.

> It is conceded that one may not patent an idea. But in practical effect that would be the result if the formula for converting binary code to pure binary were patented in this case. The mathematical formula involved here has no substantial practical application except in connection with a digital computer, which means that if the judgment below is affirmed, the patent would wholly pre-empt the mathematical formula and in practical effect would be a patent on the algorithm itself.[55]

Nevertheless, the court stated specifically that it did not hold that there could be no patent for any program servicing a computer.[56] In ensuing cases before the C. C.P.A., the court attempted to interpret the *Benson* decision and apply it to a variety of factual situations.

55 Ibid., at 71.

56 Ibid., at 72.

It was irrelevant to the court in its section 101 analysis that some or all of the steps read on (but not solely on) a mental process. As a practical matter, the court stated that in their broadest context, process claims 'read on' physical and mental steps:

> All that is necessary, in our view, to make a sequence of operational steps a statutory 'process' within 35 U.S.C. section 101 is that it be in the technological arts so as to be in consonance with the Constitutional purpose to promote the progress of 'useful arts.' [Const. art. 1, sec. 8.]

The court by this holding thus expanded the bases for a process as defined by section 101 to include any sequence of steps that can be performed by a machine as well as by a thinking person and that promotes the progress of the useful arts.

In re Mahoney[4]

The invention of *In re Mahoney* involved a data communication system, in which circuits and methods synchronized a receiver of digital information. The application for the invention disclosed a *method* of synchronizing a receiver with a bit stream containing digital information. The claims had been directed to steps in this process involving "comparing," "registering," and "counting." These functions were performed on "bits" and 'bit streams." The PTO rejected these claims under 35 U.S.C. 112, stating that the claims failed to particularly point out and distinctly claim the invention since they read simultaneously on statutory subject matter (machine implementation) as well as on nonstatutory subject matter. The C. C.P.A. disagreed with this application of section 112, and pointed out that both section 112 and 101 rejections would be overcome if the claims in fact covered only machine implementation. The court found that references in the claims to 'bits' and 'bit streams' were sufficient to preclude reading the claims on a mentally performable process. It was clear from this case that method claims must include machine limitations to meet the requirements of section 112 and 101.

In re Waldbaum[5]**/In re MacIlroy**[6]

In re Waldbaum involved a process consisting of a method for controlling the operation of a data process, i.e., a computer program, to determine the number of "1s" in a data word. The Patent Office Board of Appeals affirmed a rejection of the claims on the basis that, *inter alia, the* application did not claim statutory subject matter. The board adopted the examiner's grounds for a "mental steps" rejection, stating that since the apparatus limitations in the claims were only functional, the claims embrace "that which could be only an act of the mind rather than calling for an act on a physical thing. . ." The C. C.P.A. rejected the mental steps grounds for rejection and stated that "whether the appellant's process is a 'statutory' invention depends on whether it is within the 'technological arts'. . ." The C. C.P.A. reversed the decision of the board.

4 *In re Mahoney*, 421 F.2d *742* (C.C.P.A. 1970).

5 *In re Waldbaum*, 559 F.2d 611 (C.C.P.A. 1977).

6 *In re McIlroy*, 442 F.2d 1397 (C.C.P.A. 1971).

SOURCE: OTA, 1992.

Post-Benson to Diehr

After *Benson, the* C. C.P.A. applied a "point of novelty test to determine patentability of the claimed invention. (See box 2-C.) The claims in *In re Christensen*[57] were directed to a process for determining the porosity of a subsurface formation. The claims recited a series of known steps for obtaining certain information and a new equation for computing the porosity from this gathered information. While the C. C.P.A. was clearly of the opinion that the issue decided by the U.S. Supreme Court in *Benson* was narrowly drawn, it determined that in the *Christensen* claim reciting a new invention the "point of novelty" was a mathematical equation. Such an equation, the C. C.P.A. held, is not patentable. The steps involving establishing values for the variable steps which were not novel were not sufficient to render the invention patentable.[58]

57 *In re* Christensen, 478 F.2d 1392 (C. C.P.A. 1973).

58 Ibid., at 1394.

Box 2-C--Court of Customs and Patent Appeals Cases Following Benson

In re Noll[1]*: The* C. C.P.A. in this case found apparatus claims for displaying images patentable. The input of the display device was provided in part by a programmed data processor.

In re *Chatfield*[2] In this case the majority **opinion comments that the** C. C.P.A. 'S "point of novelty test" ***in*** *Christensen* does not mean that a claimed method maybe dissected to determine whether the novel portion lies outside the statutory classes of invention. The court repudiated any such analysis in *Christensen, and* reaffirmed its opinion in *Bernhart*[3] that classification of the point of novelty as nonstatutory is inappropriate and irrelevant for Section 101 considerations.

In re Deutsch[4]*: The court* found the claimed method for control and optimization of multiunit plants at different geographic locations patentable, since the specific computing methods were independent of the claimed invention.

In re Waldbaum[5] Characterizing the applicant's claims to a data processor controlled telephone switching system as "so abstract and sweeping" as to cover both known and unknown uses of the method claimed, the C. C.P.A. found the claims nonstatutory in view of ***Benson,*** even though some of the claims were drawn to a data processing apparatus.

In re Richman[6]*: The court* held the claimed method of calculating using a mathematical formula unpatentable.

In re ***deCastelet***[7]***: The court*** held a machine method for generating a curve from data supplied to a computer to be nonstatutory because it was derived from the solution of a set of mathematical equations (a set of points along a curve) rather than a step in achieving a larger result.

In re Freeman[8]*: This case* focused on apparatus claims in means-plus-function form for a system of typesetting information using a computer-based control system in conjunction with a phototypesetter of known design. In its analysis, the C. C.P.A. set forth a two-part test which, with certain modification, has become the standard test for determiningg subject-matter patentability of software.

In re Torna[9]***: The*** **C. C.P.A. in** *Torna* considered a method of operating a digital computer to translate from a source natural language (e.g., Russian) to a target natural language (e.g., English). The C. C.P.A. applied the Freeman rationale, determiningg whether the claims recite a *Benson type algorithm.* Finding that the program did not involve a solution for a mathematical problem, the method of enabling the computer was found to be within the technological arts and thus statutory.

---

1 *In re Noll*, 545 F.2d 141 (C.C.P.A. 1976).
2 *In re Chatfield*, 545 F.2d 152 (C.C.P.A. 1976).
3 *In re Bernhart*, 4]7 F.2d 1395 (C.C.P.A. 1969).
4 *In re Deutsch*, 553 F.2d 689 (C.C.P.A. 1977).
5 *In re Waldbaum*, 417 F.2d (C.C.P.A. 1969).
6 *In re Richman*, 563 F.2d 1026 (C.C.P.A. 1977).
7 *In re deCastelet*, *562* F.2d 1236 (C.C.P.A. 1977).
8 *In re Freeman*, 573 F.2d 1237 (C.C.P.A. 1978).
9 *In re Toma*, 575 F.2d 872 (C.C.P.A. 1978).

SOURCE: OTA, 1992.

In a subsequent case, *In re Johnston*,[59] *the* application characterized the claims as apparatus claims for a computer program which conti-oiled an automatic computerized check sorting device. The C. C.P.A. held the invention patentable, construing *Benson* to apply only to process claims. Although the Supreme Court reversed the decision, finding the invention obvious, the Court did not consider the matter of subject matter patentability under section 101.

Parker v. Flook

The Supreme Court again addressed the question of software patentability in *Parker v. Flook*.[60] *The*

---

59 *In re* Johnston, 502 F.2d 765 (C.C.P.A. 1974), *rev'd sub nom., Dunn v. Johnston,* 425 U.S 219 (1976).
60 *Parker* v. *Flook*, 437 U.S. 584 (1978).

Court reversed the C. C.P.A., holding that a method for updating "alarm limits" during a catalytic conversion process, in which the only novel feature was the use of a particular mathematical formula, was not patentable under section 101. The Court stated that the only difference between the claimed invention and conventional methods of changing alarm limits was that the applicant provided what the Court referred to as a *"mathematical algorithm* or *formula"*[61] [emphasis added] for computing alarm limits.[62] The application did not include disclosure about monitoring the process or the means of setting off an alarm or adjusting an alarm system.

The Supreme Court characterized its holding in *Benson* as precluding the patenting of the discovery of a novel mathematical formula, and addressed the issue of ' 'whether the identification of a limited category of useful, though conventional, post-solution applications of such formula makes respondent's method eligible for patent protection. ' The Court stated that it did not, holding that 'a claim for an improved method of calculation, even when tied to a specific end use, is unpatentable subject matter under Section 101 if the end use involves trivial ''post solution activity. '

The Supreme Court left open the possibility of patenting computer programs, and recognized that ''a process is not unpatentable simply because it contains a law of nature or mathematical algorithm. ' A patent must not preempt a law of nature or a mathematical algorithm.[63]

In subsequent cases, the lower courts bore in mind the Flook principles and continued to struggle with the limits of the patentability of software. In the cases of *In re Johnson*,[64] *in re Walter*,[65] and *In re Bradley*[66] *the* C. C.P.A. attempted to apply the test set out in *Freeman in* light of Flook to determine whether a software-related invention constitutes statutory subject matter.[67]

Diamond v. Diehr

In 1981 the Supreme Court decided the case of *Diamond v. Diehr.*[68]

The application in *Diehr* recited a process of curing synthetic rubber. The process included use of a known mathematical formula to determine the time for curing, and a programmed digital computer to determine the proper length of time for curing the product and automatically opening the mold. The claims were rejected by the Patent Office as non-statutory subject matter. The Court of Customs and Patent Appeals reversed. The Supreme Court decided in a 5 to 4 decision that the process was patentable under section 101. The claims were not disqualified from patentability because of the use of a mathematical equation and programmed digital computer.

The Supreme Court characterized a mathematical formula or an algorithm to be "like a law of nature, which cannot be the subject of a patent. ' The Court asserted that a claim drawn to otherwise statutory subject matter is not rendered nonstatutory simply because it involves a mathematical formula, computer program, or digital computer. The Court stated that the claims must be considered in their entirety, and held that the incorporation of a computer in the process claimed in the application does not render the process as a whole to be unpatentable subject matter.

---

61 The court in footnote 1 of the case stated, ''We use the word 'algorithm' in this case, as we did in *Gottschalk* v. *Benson,* 409 U.S. *63, 65, 93 S. Ct. 253, 254, 34,* L.Ed. 2d *273,* to mean '[a] procedure for solving a given type of mathematical problem. . . .' "

62 *Parker* v. *Flook,* Sup. Ct. 2522 (1978), at 2523.

63 The use of the term ''mathematical algorithm' here refers to the case law definition of the word, rather than the meaning given it by the computer industry.

64 *In re Johnson,* 589 F.2d 1070 (C. C.P.A. 1978).

65 *In re Walter,* 618 F.2d 758 (CCPA 1980).

66 *In re Bradley,* 600 F.2d 807 (C.C.P.A. *1979), aff'd by equally divided court sub nom. Diamond v. Bradley,* 450 U.S. 381 (1981).

67 For further discussion of these and other post-*Flook* cases, see Cary Sherman, Hamish Sandison, and Mark Guren, *Computer Software Protection Law,* (Washington DC: Bureau of National Affairs, 1991.)

68 *Diamond" Diehr,* 450 U.S. 175 (1981). Some commentators believe that *Diehr* has been misinterpreted by the Patent and Trademark Office and the lower courts. Professor Pamela Samuelson argues that as a result of *Diehr, the* Patent Office issues patents for a wide variety of nonindustrial computer program-related inventions and seems to be issuing patents for Benson-type algorithms. While some attorneys argue that this change is consistent with *Diehr,* she finds a substantial bias in patent law for *Benson's* ruling that computer program algorithms are unpatentable. See, Pamela Samuelson, ''Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions, ' *Emory Law Journal,* vol. *39, No.* 4, fall 1990.

The Court in Diehr clearly stated that purely mathematical procedures or algorithms alone are not proper subject matter of patent protection:

> [W]hen a claim recites a mathematical formula (or scientific principle or phenomenon of nature), an inquiry must be made into whether the claim is seeking patent protection for that formula in the abstract. A mathematical formula as such is not accorded the protection of our patent laws, and this principle cannot be circumvented by attempting to limit the use of the formula to a particular technological environment. Similarly, insignificant postsolution activity will not transform an unpatentable principle into a patentable process. To hold otherwise would allow a competent draftsman to evade the recognized limitations on the type of subject matter eligible for patent protection.

In cases such as that being considered by the Court, however, where the applicant wishes to protect an overall industrial process rather than simply a mathematical formula, the circumstances change.

> [W]hen a claim containing a mathematical formula implements or applies that formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect (e.g., transforming or reducing an article to a different state or thing), then the claim satisfies the requirements of section 101.

In spite of this holding by the Supreme Court, it is still believed that the *Diehr* decision does not conclusively determine whether a computer program, standing alone, may be patented.[69] However, *Diehr* indicates that at least some subject matter involving software may be patented and that algorithms may be included as part of patentable subject matter .70

PTO Policy and Procedural Response to Court Cases

Following the C. C. P.A. 'S decisions in the companion cases of *Pardo, Taner, Abele,* and *Meyer in* mid-1982 (see box 2-D), the court did not further consider the subject of algorithms and statutory subject matter. The Court of Appeals for the Federal Circuit, established on October 1, 1982, did not hand down decisions on these issues until November 3, 1989 with the opinion in *In re Grams.*[71] Just prior to this decision in September 1989, the PTO, in part in response to commentary in the media and academic journals, published a set of guidelines on the patentability of mathematical algorithms. In these guidelines, the PTO interpreted the law to encourage greater consistency in examination procedures among the different groups which handled patent applications that include mathematical algorithms.

The PTO relied upon the two-part test of *In re Freeman*[72] as modified by *In re Walter*[73] and *In re Abele*[74] as the proper legal analysis of mathematical algorithm-statutory subject matter cases. Part 1 of the test is to determine whether a mathematical algorithm is recited, directly or indirectly, in the claims. The guidelines warn that, since mathematical algorithms have been determined not to fall within the section 101 statutory class of a ' 'process, ' applicants may attempt to circumvent the nonstatutory subject matter rejection by drafting mathematical algorithms as ''apparatus' claims.[75] The guidelines require that the claims first be inspected to determine whether the claim recites a mathematical algorithm.[76] Noting that such a determination is not always possible by inspection of the claim, the guidelines indicate 'careful interpretation of each claim in light of its supporting disclosure. '[77]

Part 2 of the test is to determine whether the mathematical algorithm is ''applied in any manner to physical elements or process steps. ' The guide-

---

69 Sherman et al., *Computer Software Protection Law*, op. cit., footnote 67. Sherman enumerates an extensive list of scholarly legal articles discussing the meaning of the *Diehr* decision.

70 Sherman et al., op. cit. footnote 67, at page 403-419.

71 *In re Grams*, 888 F.2d 835 (Fed. Cir. 1989).

72 *In re Freeman*, 573 F.2d 1237 (C. C.P.A. 1978).

73 *In re Walter*, 618 F.2d 758 (C.C.P.A. 1980)

74 *In re Abele*, 684 F.2d 902 (C.C.P.A. 1982)

75 Lee E. Barrett, Associate Solicitor, *Official Gazette of the Patent Office*, Sept. 5, 1989.

76 Ibid.

77 *In re Johnson*, 589 F.2d at 1079.

Box 2-D—The Cases of Pardo, Taner, Abele, and Meyer

In re Pardo:[1] *The case* of ***In re*** *Pardo* involved an invention which controlled the internal operations of a computer through the compiler program, and consisted of a method for converting the computer from a sequential processor to a processor that is not dependent on the order in which it receives program steps. The patent application characterized the invention as an algorithm of a compiler program. The Court of Customs and Patent Appeals viewed the invention not as an algorithm, but as the rules by which a data processor operates, thereby governing the manner in which programs are executed. It rejected the argument that the claims were really mathematical calculations.

In re Taner:[2] Following the handing down of the Diehr decision, the Patent and Trademark Office rejected a patent application for a method of seismic exploration using simulated seismic waves which were generated by summing the signals of conventional waves. A mathematical algorithm carried out on a digital computer was a part of the claimed invention. According to the court, the simulated signal constituted a physical conversion of the summed actual signals, and was therefore a patentable process.

In re Abele[3]*: In re Abele* involved a patent application for an improvement for CAT scanning. The C. C.P.A. concluded that some portions of the invention were patentable subject matter in spite of the use of a computer to perform some of the calculations involved.

In re Meyer[4]*: The* C. C.P.A. in *In re Meyer* held a process and apparatus claims covering the use of a computer program and algorithm for storing and comparing medical test results. The applicant conceded and the court found that the claimed invention computerized the thought process of a physician. The court concluded that the claims were properly rejected by the Patent Office because they related to a:

> mathematical algorithm representing a mental process that has *not* been applied to physical elements or process steps and is not, therefore, limited to any othewise statutory process, machine, manufacture, or composition of matter.

[1] *In re Pardo*, 684 F.2d 912 (C.C.P.A. 1982).
[2] *In re Taner*, 681 F.2d 787 (C.C.P.A. 1982).
[3] *In re Abele*, 684 F.2d 902 (C.C.P.A. 1982).
[4] *In re Meyer*, 688 F.2d 789 (C.C.P.A. 1982).

SOURCE: OTA, 1992.

lines suggest that the *Abele* rule be followed, wherein the claim is viewed without the mathematical algorithm to determine whether what remains is "otherwise statutory. " If it is, the claim does not become nonstatutory simply because it includes a mathematical algorithm. Citing *Flook*, and recognizing that the line between a patentable ''process' and an unpatentable ''principle' is not always clear, the PTO suggests some "useful guidelines" synthesized from the court decisions, that should aid in drawing the line between a patentable process and an unpatentable ''principle. " The PTO issued guidelines resulting from leading court decisions:

1. If the only limitation other than the mathematical algorithm is insignificant or nonessential "post solution activity, " the claimed subject matter is nonstatutory .[78]
2. A mathematical algorithm is not rendered statutory by 'attempting to limit the use of the formula to a particular technological environment,' so that ''field of use" or "end use" limitation in the claim preamble do not suffice to constitute statutory subject matter.[79]
3. If the only claim limitations in addition to the mathematical algorithm are data gathering steps which 'merely determined values for the variables used in the mathematical formulae used in making the calculations, such steps are insufficient to change a nonstatutory method of calculation into a statutory process,[80]
4. The PTO suggests that it is useful to analyze whether there is transformation of something physical into another form. A distinction is made between transformation of matter in one

[78] *Parker* v. *Flook*, 437 U.S. at 590.
[79] *Diamond* v. *Diehr*, 450 U.S. at 191.
[80] *In re Richman*, 563 F.2d at 1030.

physical state to another physical state, which is a statutory process in electrical arts, and mathematical manipulation of "data' which is not a statutory process.

5. Structural limitations in method claims have a limited effect on patentability. While such limitations are not improper, they are not generally to be given patentable weight unless they affect or form an essential part of the process.[81]

The analysis of the guidelines were used before the Federal Circuit in the Solicitor's brief for the Commissioner in *In re Grams,*[82] *the* first reported decision by the Federal Circuit in this area. The subsequent case of *In re Iwahashi*[83] indicates how mathematical algorithms in apparatus claims maybe handled by the PTO.

In re Grams involved method claims that combined a mathematical algorithm with data gathering steps. The invention at issue was a process for diagnosing a patient using an algorithm that manipulated data obtained from results of tests performed on the patient. The claim involved performing clinical laboratory tests on an individual to obtain data indicative of the individual's medical condition. The invention further involved processing steps for manipulating the collected data. As a result of this process, a set of parameters corresponding to a combination of constituents responsible for the medical condition could be identified. Citing *Benson, the* Federal Circuit stated that these remaining steps were "a procedure for solving a given kind of mathematical problem. The court cited the rule of *Abe/e,* which stated that:

> [i]f a claim would be otherwise statutory, albeit inoperative or less useful without the algorithm, the claim likewise presents statutory subject matter when the algorithm is included.

The Federal Circuit would not interpret this language as declaring any claim that was statutory without the algorithm as patentable. It also pointed to the "provided that' qualification in *Abele as* denying effect to field of use limitations or nonessential postsolution activity. The court decided that the claimed process was nonstatutory subject matter because the step of performing lab tests "merely provides the data for the algorithm," and that whether or not the claims required that the method be performed by a programmed computer was irrelevant to the determination of whether the claim defined a section 101 process.[84]

In re Iwahashi[85] involved an electronic 'unit' for providing coefficients to be used in pattern recognition based on input values. The method in the prior art was based on a formula involving a multiplication step. The improvement in *Iwahashi* was based on the fact that a close approximation of the correct coefficient values could be obtained without multiplication by using instead a stated formula. This eliminated circuitry necessary to perform the multiplication function. All of the elements in the claim were in means-plus-function form, except for a read only memory containing certain necessary values. The Federal Circuit found that while the claim recited a mathematical algorithm, the preemption aspect of the two part test was not met because the mathematical algorithm was physically implemented to define structural relationships between physical elements of the claim. Even though the read-only memory unit in the computer stored a mathematical formula for calculating the square of numbers, the claim defined more than a mathematical algorithm. The Federal Circuit reversed a finding of nonstatutory subject matter.

As discussed previously in this chapter, the PTO in the 1960s was seriously concerned about a variety of administrative problems confronted by the office, citing long pendency periods for applications prior to issuance of a patent, a backlog of applications, and problems with classification and maintenance of prior-art search files which ultimately would affect the quality of patents issued. These complaints and similar ones are not new, nor have the patent community and industry ceased to voice them. They

---

81 *In re de Castelet,* 562 F.2d at 1244.

82 *In re Grams,* 888 F.2d 835 (Fed. Cir. 1989).

83 *In re Iwahashi,* 888 F.2d 1370 (Fed. Cir. 1989).

84 One commentator has offered as a possible reconciliation of the *Abele* and *Grams* decisions that the pre-algorithm step of *Abele,* when combined with the post-algorithm step, defined a physical process. In *Grams the algorithm* merely replaced the subjective mental processes of a physician. See Ronald S. Laurie, "Patentability of Computer Programs in the USA," *The Law of Information Technology in Europe 1992: A Comparison With the USA,* Computer Law Series.

85 *In re Iwahashi,* 888 F.2d 1370 (Fed. Cir. 1989)

are also not exclusive to the U.S. system. In as early as the 1850s royal commissions in Britain investigated the operation of the patent system. Testimony damaging to the reputation of the patent system led to passage of a patent reform bill.[86]

Presently, the PTO is again under scrutiny due to concerns about the patent system in general, as well as its implementation with respect to software, In addition to the problems cited in the 1960s which apparently persist, recent concerns also include the competence of examiners to make fair and proper decisions about applications for software-related inventions, citing in particular the controversy over the educational requirements for examiners, difficulty in training and retaining adequate technical staff, and the resulting insufficient number of examiners to address the huge and growing volume of incoming applications for software-related inventions. There are, in addition, ongoing concerns about the quality of the prior-art database, and problems of classification. [87] (For statistics regarding issuance of patents for software-related inventions, see table 2-l).

In response to these concerns within the software community and beyond, the Secretary of Commerce established in August 1990 the Advisory Commission on Patent Law Reform.[88] The Commission's objective is to advise the Secretary on the state of the patent system in the United States and the need for any reform, as well as to advise the Secretary on the need for any changes in U.S. law relating to enforcement and licensing of U.S. patents. The Commission is to consist of 8 to 15 members, serving for 2 year terms, appointed by the Secretary of Commerce to represent corporations involved in filing patent applications, members of the bar, and the general public.[89] The Secretary may appoint official observers from various government agencies, such as the U.S. Trade Representative, the Office of Science and Technology Policy, the National Science Foundation, and the Small Business Administration.[90]

Table 2-l—Number of Patents Issued for Software-Related Inventions, 1970-91

Year	Number
1970	27
1971	51
1972	70
1973	35
1974	11
1975	14
1976	5
1977	7
1978	30
1979	23
1980	25
1981	21
1982	52
1983	64
1984	136
1985	153
1986	187
1987	227
1988	131
1989	193
1990	599
1991	602

SOURCE: Electronic Data **Systems Corporation (EDS). The** data **in this** table given for 1972 through 1989 **reflect statistics compiled by EDS by examining notices published in the** *Official Gazette.* **Information given for 1990 and 1991 was collected using artificial intelligence technology involving a natural language interface created by EDS. Official database tapes purchased from the PTO were read using this technology to characterize each patent and determine which involved software-related inventions. When there was some difficulty ascertaining the nature of patent, EDS personnel referred to the** *Official Gazette.*

On May 16, 1991, the Patent and Trademark Office published in the Federal Register a ''Request for Comments for the Advisory Commission on Patent Law Reform. ''[91] In that request, the PTO solicited comments about whether current U.S. patent laws provide adequate and appropriate pro-

86 U.S. Congress, Senate Subcommittee on Patents, Trademarks and Copyrights of the Committee on the Judiciary, *An Economic Review of the Patent System,* **Committee** Print, 1958, p. 4.

87 Office of Technology Assessment Workshop on June 20,1991; Eliot Marshal, "The Patent Game: Raising the Ante, " *Science, vol.* 253, July 5, 1991.

88 U.S. Department of Commerce, Charter of the Advisory Commission on Patent Law Reform.

89 As of this writing, advisory commision members are as follows: *industry representatives*: John E. Pepper, President, Procter and Gamble; Keith R. McKennon, Executive Vice President, The Dow Chemical Co,; Vincent J. Rauner, Senior Vice President, Motorola, Inc.; Howard G. Figueroa, Vice president, IBM Corp.; Ardon B. Judd, Vice President, Dresser Industries, Inc.; Mike Pantuliano (representing Edmond Pratt), Pfizer, Inc.; *academic representatives:* Roland Schmitt, President, Rensselaer Polytechnic Institute; Edward L. MacCordy, President, Association of University Technology Managers; *patent bar representatives:* Douglas W. Wyatt, Wyatt, Gerber, Burke & Badie; Donald R. Dunner, Finnegan, Henderson, Farabow, Garrett & Dunner; *antitrust bar representative:* Robert P. Taylor, Pillsbury, Madison & Sutro; *small business representative:* Juan A. Benitez, President, Life-Stream Diagnostics; *public representatives:* William Keefauver, Edward H. Baxter.

90 Ibid.

91 56 *Federal Register* 22702 (May 16,1991).

tection of novel computer-related inventions, and the relationship of patent protection for computer Program-related inventions to other intellectual property protection. In addition to these questions specific to computer software, the request sought public comment on broader issues critical to the patent system generally, e.g. Federal protection for trade secrets, cost and complexity of patent enforcement, frost to file system, automatic publication of applications, and the term of patent protection.[92]

In response to this invitation for public comment, the PTO received over 545 letters from individuals, small and large businesses, trade associations, academics, patent attorneys, and bar associations. While the responses addressed the range of questions raised by the PTO request, the majority focused on the issues surrounding computer-related inventions.

Many of the issues raised by the Patent Reform Commission are addressed by the efforts of World Intellectual Property Organization (WIPO) to harmonize patent laws in member countries. WIPO'S Committee of Experts on the Harmonization of Certain Provisions in Laws for the Protection of Inventions is considering a draft treaty which contemplates instituting requirements that would require changes in U.S. law. These would include a change from a first to invent to a first to file system, awarding the patent to the applicant who has the earliest filing date. Such a change would signifi cantly reduce the U.S. interference practice. The WIPO treaty also contemplates a term of patent of 20 years from the date of filing a patent application, as opposed to the 17-year **term** from date of grant provided by the U.S. law.[93]

One attempt by the private sector to address the problem of the prior art is the formation of a Software Patent Institute (SPI) as a mechanism to provide access to prior art in the software area and to further the PTO'S knowledge of software.[94] The proposed near-term objectives of SPI are: 1) to establish a database of prior art in the area of software and to encourage contributions to the database; 2) to facilitate searches and other appropriate access to the prior-art database by interested parties, including the PTO, potential applicants, potential challenges to issued patents, and others with similar needs; 3) to facilitate the advancement of the PTO's knowledge of the nature of software and software engineering by organizing educational and training opportunities for PTO personnel; and 4) to disseminate information about services of the SPI to the broad software community .95 (See box 2-E for discussion of Analagous Patent Law in Foreign Countries.)

Copyright

Copyright law in the U.S. protects the right of an author to control the reproduction, adaptation, public distribution, public display, and public performance of original works of authorship of every kind, ranging from books to sound recordings.[96] While copyright comprises only one aspect of intellectual property protection for computer software, its role in that protection is a major one.

Goals of U.S. Copyright[97]

A fundamental goal of U.S. copyright law is to promote the public interest and knowledge-the "Progress of Science and useful Arts. ' '[98] Although

---

92 Ibid.

93 For further discussion of the WIPO draft treaty, see Edward G. Fiorito, "WIPO Experts Make Progress on Patent Harmonization Draft," *BNA's Patent Trademark & Copyright Journal, vol. 41, No. 1013,* Jan. 10, 1991, pp. 231-241.

94 Critics of such an idea note that an improved database of prior art is a common response to criticism of software patents. These critics claim, rather, that such a database would have little practical effect, in part because of the frequency with which techniques were developed in the past and never published, and in part because such a proposal assumes that it is sound policy to decide that the techniques patented today should be unavailable to the programmer provided they were in fact not published at an earlier date. Richard Stallman, The League for Programming Freedom, personal communication September 1991.

95 Bernard Galler, Professor of Electrical Engineering and Computer Science, University of Michigan, personal communication, September 1991.

96 Sherman et al., *Computer Software Protection Law,* op. cit., footnote 67.

97 This material is adapted from U.S. Congress, OTA, *Copyright & Home Copying: Technology Challenges the Law,* OTA-CIT-422 (Washington, DC: U.S. Government Printing Office, October 1989).

98 The Constitution provides that:

> Congress shall have Power. . . to Promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors, . . the exclusive Right to their respective Writings and Discoveries.

This provision arguably distinguishes authors from inventors and writings from discoveries, suggesting a fundamental distinction between copyrightable works of authorship and patentable works of utility. For further discussion of this question of scope, see *Computer Software* Protection *Law,* op.cit. at footnote 1, Section 204,6(a).

Box 2-E—Analogous Patent Law in Foreign Countries

The Pacific Rim

Patent laws of Japan,[1] Taiwan, Korea and Thailand are all silent on the issue of patentability of computer programs. In theory, patent law does not provide protection for a computer program itself. However, Japan and Taiwan have granted patents for certain computer programs, especially if the computer program is described in conjunction with a method or computer in which the program is used in the specification of an application.[2]

Western Europe

The European Communities have agreed in their Software Directive that the prescribed protection of computer programs under copyright law does not prejudice the application of other forms of protection where appropriate. Computer software may be protected under patent law in addition to copyright in European Community member nations.[3]

France—French patent law provides that computer programs are not patentable. This rule is narrowly interpreted so that computer program standing alone is not patentable, while a process including software may be patentable. The patent protects the process, but not the software; the software can be used independently or in another process. Hardware may be patented.[4]

Switzerland-Swiss law does not provide patent protection for computer software.[5]

Latin America

Argentina-In Argentina software was not known or considered when the patent Law was enacted so that it is not specifically mentioned in the law. The Patent Office may allow patent protection when it is part of the essence of an invention.[6]

Brazil—While under Brazilian law hardware is subject to patent protection, software is not considered patentable and is expressly excluded from patentable subject mater by the Industrial Property Code.[7]

---

1 Several reviewers have voiced particular concern regarding what they believe to be aspects of the Japanese patent system that are unfair to United States applicants. An investigation of such charges is outside the scope of this report. However, Senator Lloyd Bentsen and Senator John D. Rockefeller IV plan to ask for an investigation into charges that Japanese companies use their patent program to acquire foreign technologies. This request was made in a letter to General Accounting Office Comptroller General Charles Bowsher. "Washington Technology," October 10, 1991, p. 4.

2 Paul C. B. Liu, "Computer Software and Intellectual Property Law in the Pacific Rim Countries," Contractor Report for the Office of Technology Assessment, March 1991, p. 37-38.

3 Council Directive of May 14, 1991 on the legal protection of computer programs (91/250/EEC) preamble, para. 30; published in the Official Journal of the European Communities, No. L 122/42, May 17, 1991.

4 Cary Sherman, et al., *Computer Software Protection Law* (Washington DC: Bureau of National Affairs, Inc. 1990) p. FR-26.

5 Ibid. at p. CH-3.

6 Ibid. at p, AR-41-42.

7 Ibid. at p. BR-20.

copyright is a property interest, its primary purpose was not conceived of as the collection of royalties or the protection of property; rather, copyright was developed primarily for the promotion of intellectual pursuits and public knowledge. As the Supreme Court has stated:

> The economic philosophy behind the clause empowering the Congress to grant patents and copyrights is the conviction that encouragement of individual efforts by personal gain is the best way to advance public welfare through the talents of authors and inventors in Science and the useful Arts.[99]

Therefore, the congressionally mandated grant of a limited monopoly for authors is based upon a dualism: the belief that the public should benefit from the creativity of authors and the belief that a copyright monopoly is necessary to stimulate the

99 *Mazer* v. *Stein*, 347 U.S. 201, 219 (1954).

Box 2-F—Early Protection Against Copying[1]

Professional copyists have long reproduced famous or popular works to sell to those who could not afford originals. Forgers have copied works of well-known artists, or more commonly have produced works that imitate their styles. Paintings and drawings have been reproduced as prints, sculptures as plaster or bronze casts. With the rise of prints in the early Renaissance came the first measures to protect *against* copying.

The increased attention given to copies by art theorists in the 17th and 18th centuries reflects the rise of the professional copyist, catering to the expanded art market that resulted from the general public's rising interest in art. Many young artists of this period, particularly in tourist centers like Rome, worked in the mass production of copies of famous or popular paintings. Given the prevalence of copying, artists often sought ways to protect the authenticity of their works. One such attempt was the *liter veritatis*, ''*the* book of truth," of Claude Lorrain, an often imitated artist. The art historian Baldinucci recounts:

> Not only was his composition cribbed by some envious persons desirous of unfair earnings, but, through imitation of his manner, copies were sold in Rome as originals by his brush; but this the master was being discredited, the patron for whom the pictures were painted badly served, and the buyers defrauded since they were given copies instead of the originals. . . . Poor Claude did not know from whom to guard himself among the numerous persons who came to his studio nor what decision to take. Everyday similar pictures were brought to him so that he might recognize whether they were by his hand or not. Thereupon he decided to keep a book and began copying the composition of all the pictures which left his studio.[2]

The most famous instance of a complaint by one artist against another for copying his work is recounted by Vasari of Durer and the engraver Marcantonio Raimondi. Marcantonio, visiting Venice, found a set of Durer's engravings for sale there and was so impressed that he spent almost all of the money he had brought with him to purchase them:

> Marc 'Antonio, having considered what honor and profit might be acquired by one who should apply himself to [the art of engraving] in Italy, formed the determination to give his attention to it with all possible assiduity and diligence,

[1] This material is excerpted from: "Copying in the Visual Arts in Europe Since the Renaissance: Forms, Functions, Response and Restrictions," contractor paper, Robert Echols, July 1991.

[2] Rudolf and Margot Wittkower, *Born Under Saturn: The* Character *and Conduct of Artists* (London: 1965).

greatest creativity of authors.[100] A direct corollary to this concept is that the grant of a monopoly would not be justifiable if the public did not benefit from the copyright system.[101] Thus, policy arguments that equate copyright with royalty income, or theories that assert that copyright is necessary in order to secure royalty income, run counter to this theory and appear to be inconsistent with the intent of the Framers.

Development of U.S. Copyright[102]

Much of the structure and basis for American law is derived from its British legal antecedents. After the introduction of the printing press in England in the late 1400s, the Crown's first response was to control what writings were printed or copied. The earliest British copyright laws were enacted in the 1500s to promote censorship by the government in cooperation with a monopolistic group of printers known as the Stationers' Guild.[103] This system collapsed when the company failed to exercise discretion as censor, but used its monopoly power to set high prices. Parliament's response in 1695 was to allow the Stationers' copyrights to expire, but this resulted in a period of anarchical publication. In 1709 Parliament responded to the situation by

100 The Supreme Court commented on this dualism in *Sony Corporation of America* v. *Universal City Studios*, *464* U.S. 417, 429 (1984), wherein it discussed the:

> difficult balance between the interest of authors and inventors in the control and exploitation of their writings and discoveries on the one hand, and society's competing interest in the free flow of ideas, information and commerce on the other hand.

101 Melville Nimmer observed that the Framers of the Constitution regarded the system of private property as existing per se fOr the public interest. Therefore, in recognizing a property status in copyright, the Framers extended a recognition of this public interest into a new realm (Melville Nimmer, *Nimmer on Copyright (New* York, NY: Bender, 1991), vol. 1, sec. 1-32.1).

102 This material is adapted from *Copyright & Home Copying: Technology Challenges the Law*, OTA-CIT-422, op. cit., footnote 97.

103 See 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. 34-36.

> He thus began to copy those engravings by Albrecht Durer, studying the manner of each stroke and every other detail of the prints that he had bought, which were held in such estimation on account of their novelty and their beauty, that everyone sought to have some. Having then counterfeited on copper, with engraving as strong as that of the woodcuts that Albrecht had executed, the whole of the said Life and Passion of Christ in thirty-six parts, he added to these the signature that Albrecht used for all his works, which was "A.D.," an they proved to be so similar in manner, that, no one knowing that they had been executed by Marc' Antonio, they were ascribed to Albrecht, and were bought and sold as by his hand. News of this was sent in writing to Albrecht, who was in Flanders, together with one of the counterfeit Passions executed by Marc' Antonio; at which he flew into such a rage that he left Flanders and went to Venice, where he appeared before the Signoria and laid a complaint against Marc'Antonio. But he could obtain no other satisfaction but this, that Marc' Antonoio should no longer use the name or the above-mentioned signature of Albrecht on his works.[3]

Marcantonio continued to produce copies of Durer's engravings with his own monogram in place of Durer's.

Perhaps as a result of this expedience, when Durer published his wood cuts of the Life of the Virgin in 1511, he prefaced the edition with the warning:

> Woe unto him who ventures to assail us and lay hands on the toil and invention of another!

The next year the Town Council of Nuremberg issued a notice stating that:

> . . . a foreigner is taking the liberty of selling pictures bearing Durer's mark, but these are counterfeits; he shall be compelled to remove all the marks, or everything shall be confiscated.[4]

It was in fact in Venice that the first protections akin to modern copyright were issued. The concept of an exclusive right to publish a text or image for a fixed period of time, known as a privilege, originated during the rapid rise of publishing in the 15th century in centers like Venice and Nuremberg. The Venetian Senate issued the first recorded privilege in 1469 and soon began to grant privileges for prints as well.[5]

3 Giorgio Vasari, T'& *Lives of the Artists,* trans. George Bull (Penguin Books: 1965) (vol.1) and 1987 (vol.2).

4 Michel Melot et al., *Prints* (Geneva: 1981), p. 47.

5 Susan tire, *The Image Multiplied Five Centures of Printed Reproductions of Paintings and Drawings* (London: Abaris Books, *1987).*

enacting legislation known as the Statute of Anne. This statute granted a copyright to authors, as opposed to printers, for a period of 14 years. The copyright was renewable for an additional 14 years if the author was still alive. After the expiration of the copyright, the writing became part of the public domain, available for use by anyone. This first modem copyright law became the model for subsequent copyright laws in English-speaking countries.[104]

After severing political ties with Great Britain, the former American colonies sought means to secure copyright laws. In 1783, the Continental Congress passed a resolution encouraging the various States to enact copyright legislation. All of the States except Delaware enacted some form of copyright statute, although the various State laws differed greatly.[105] Because of the differences in the State copyright laws and the ensuing difficulties, the Framers of the Constitution, notably James Madison, asserted that the copyright power should be conferred upon the legislative branch.[106] This concept was ultimately adopted, and Congress was granted the right to regulate copyright (art. I, sec. 8, cl, 8).[107]

The First Congress in 1790 enacted the frost Federal copyright act.[108] This legislation provided for the protection of authors' rights. Commentators have written that the central concept of this statute is that copyright is a grant made by a government and a statutory privilege, not a right.[109] The statute

104 Ibid.

105 Lyman Ray Patterson, *Copyright in Historical Perspective* (Nashville, TN: Vanderbilt University Press, 1968), p. 183.

106 Ibid., pp. 192-193.

107 congress constitutional grant of copyright regulation is more restricted than its English antecedent.

108 Ch.15, Sec. 1, 1 Stat. 12. See: OTA-CIT-302, op. cit., footnote 103, p. 64.

109 Patterson, op. cit., pp. 198-199.

was substantially revised in 1831110 to add copyright coverage to musical compositions and to extend the term and scope of copyright. A second general revision of copyright law in 1870[111] designated the Library of Congress as the location for administration of the copyright law, including the deposit and registration requirements. This legislation extended copyright protection to artistic works. The third general revision of American copyright law in 1909[112] permitted copyright registration of certain types of unpublished works. The 1909 legislation also changed the duration of copyright and extended copyright renewal from 14 to 28 years. A 1971 amendment extended copyright protection to certain sound recordings.[113] The fourth and most recent overhaul of American copyright law occurred in 1976, after years of study and legislativeactivity.114 The 1976 legislation modified the term of copyright and, more significantly, codified the common law fair-use concept as a limitation on the exclusive rights of the copyright holder. In 1980, following recommendations made by the National Commission on New Technological Uses of Copyrighted Works (CONTU), legislation explicitly extended copyright to computer programs.115

The *Scope and Nature of Copyright Protection*

The copyright statute[116] interprets the Constitutional term ''writings' broadly, defining it as:

> works of authorship. . . fixed. in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced or otherwise communicated, either directly or with the aid of a machine or device.

Copyright protection is expressly provided for eight categories of works: literary; musical; dramatic; pantomimes and choreographic; pictorial, graphic, and sculptural; motion picture and other audiovisual works; sound recording; and architectural; however, the legislative history indicates that these categories are not meant to be exhaustive. Computer programs are copyrightable as "literary works' as defined in 17 U.S.C. 101.117 The term ''computer program'' is also defined in section 101 as ''a set of statements or instructions used directly or indirectly in a computer in order to bring about a certain result.

Copyright protection subsists from the time a work of authorship is created in a fixed form. The copyright in the work becomes the property of the author immediately upon creation. Only the author, or one deriving rights through the author, can rightfully claim copyright.

In the case of works made for hire, the employer rather than the employee is presumptively considered the author. A work made for hire is defined as:

1. a work prepared by an employee within the scope of his or her employment; or
2. a work specially ordered or commissioned for use in a variety of circumstances enumerated by the statute.[118]

Copyright does not protect ideas, but rather the expression of ideas. Copyright protection does not extend to any:

> . . . procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated or embodied.[119]

Copyright protects the writings of an author against unauthorized copying, distribution, etc., and protects the form of expression rather than the subject matter of the writing. Unlike patents, it does

---

[1] *104* Stat. 436.

111 Act of July 8, 1870, C. 230, 16 Stat. 198.

112 Act of March 4, 1909, C. 320, 35 Stat.1075.

113 Public Law 92-140, Oct.15,1971, 85 Stat. 391.

114 Public Law 94-553, Oct.19, 1976, 90 Stat. 2541, codified at 17 U.S.C. 101, et seq. (1982).

115 In the 1980 amendments, a definition of ''Computm program'' was added to section 101 of the Copyright Act of 1976 and a new section 117 w&s added, limiting computer-program copyright holders' exclusive rights. The owner of a computer program may make another copy or adaptation of that program if the copy is needed for a specific step in using the computer program or if the copy is for archival purposes.

116 17 U.S.C. 102(a)

11717 U.S.C.101 Provides in pertinent part: ''Literary works'' are works, other than audiovisual works, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects, such as books, periodicals, manuscripts, phonorecords, film, tapes, disks or cards, in which they are embodied.

118 17 U.S.C. 101.

119 17 U.S.C. 102(b),

not protect against independent creation. Copyright grants the owner the exclusive right to do and to authorize others to do the following: [120]

- reproduce copies of the copyrighted work;
- prepare derivative works based on the copyrighted work;
- distribute copies of the copyrighted work to the public by sale or other transfer of ownership, or by rental, lease, or lending;
- perform the copyrighted work publicly; and
- display the copyrighted work publicly.[121]

The statute does, however, specify certain limitations to the copyright owner's exclusive rights that are noninfringing uses of the copyrighted works. These limitations include the "fair use" of the work[122] (17 U.S.C. 107 (1988)), certain kinds of reproduction by libraries and archives (17 U.S.C. 108 (1988)), certain educational performances and displays (17 U.S.C. 110 (1988)), and certain other uses (17 U.S.C. 117 (1980)).

It is an infringement of the copyright for anyone to engage in any of the above activities without the authorization of the copyright owner. The copyright statute provides that the copyright owner may institute an action for infringement against the alleged infringer (17 U.S.C. 501(b) (1988)). A court may issue an injunction against the copyright infringer to prevent further infringement of the copyright (17 U.S.C. 502 (1988)). An infringer of a copyright may be subject to the payment of actual damages and profits to the copyright owner (17 U.S.C. 504(b) (1988)); or in certain circumstances the copyright owner may elect specified statutory damages within specified ranges in lieu of actual damages and profits (17 U.S.C. 504(c) (1988)). In addition, in certain cases the court may permit the recovery of legal fees and related expenses involved in bringing the action (17 U.S.C. 505 (1988)). Criminal sanctions may also be imposed for copyright infringements in certain cases (17 U.S.C. 506 (1988)).

The 1976 Copyright Act and Fair Use[123]

The tension between the stimulation of intellectual pursuits and the property interests of the copyright owner has been a central issue in the development, implementation, and interpretation of American copyright laws. Moreover, the concept of copyright presents a seeming paradox or contradiction when considered within the context of the first amendment freedom-of-speech guarantees: while the first amendment guarantees freedom of expression, it can be argued that copyright seems to restrict the use or dissemination of information. It can be argued, however, that copyright, to the degree that it stimulates expression and encourages writing and other efforts, furthers first amendment expression values by encouraging the quantity of speech' that is created.[124] In attempting to resolve these conflicting interests, the courts have adopted a test that weighs the interests of freedom of expression and the property interests of the copyright holder to arrive at an acceptable balance.[125] An extensive body of case law has been developed that weighs and counterbalances first amendment concerns and the rights of the copyright holder.[126]

Hence, the American copyright system is based on dual interests: intellectual promotion and property rights. Combined with these factors is the first amendment freedom-of-expression concern. Courts have balanced and assessed these seemingly con-

120 Not all works, however, enjoy all rights. For example, sound recordings have no public performance right. 17 U.S.C. 106(4).

121 17 U.S.C. 106.

122 Before codification of the "fair-use" exception in the 1976 copyright act, the fair-use concept was upheld in a common law copyright action in *Hemingway v. Random House, Inc., 53* Misc.2d 462,270 N.Y.S. 2d 51 (Sup. Ct. 1967), *aff'd on other grounds* 23 NY.2d 341,296 N. Y.S.2d 771 (1968). The common law concept of 'fair use' was developed over many years by the courts of the United States. See, for instance, *Folsom v. Marsh, 9* F.Cas. 342 (N. 4901) (C.C.D. Mass. 1841); and *Mathews Conveyor Co. v. Palmer-Bee Co., 135* F.2d 73 (6th Cir. 1943).

123 This material is adapted from *Copyright & Home Copying: Technology Challenges the Law,* op. cit., footnote 97.

124 It is also argued that freedom of speech guarantees the speaker the right to speak his own expression, and that it does not give him the right to speak (or copy) someone else's expression. Nor does it prevent a speaker from using the ideas or information in someone else's ideas, facts, or information. Copyright requires the speaker to arrive at his own expression for the ideas he wishes to express. The resulting conflict or balance between these interests is part of copyright itself—limited protection with the limitations specifically designed to encourage publication and access to information. The remaining conflict, it is argued, may be resolved by fair use. Mary Jensen, University of South Dakota School of Law, personal communication, Sept. 29, 1991.

125 Nimmer, op. cit., foomote 101, vol.1, *sec.* 1.10.

126 See *Harper & Row Publishers, Inc. v. Nation Enterprises,* 471 U.S. 539 (1985).

Box 2-G Copying in Renaissance Art[1]

Copying has appeared in many guises in the history of the visual arts. Artists have produced multiple originals, sometimes identical, sometimes variants of the same composition, with varying degrees of assistance from others in their workshops. Students have copied masterworks as part of their training. Mature artists, even the greatest, have copied the works of other artists as a way of enriching their own styles. Compositions or individual motifs invented by one artist have reappeared in the works of others, transformed to a greater or lesser extent, and questions about the roles of imitation and innovation in art have been among the most frequently debated in the many treatises about art theory written since the Renaissance.*

The present day tendency to prize originality of form and content, while denigrating art that relies on earlier art as derivative and retrogressive is inconsistent with the practice of art in the past.[3] One scholar writes:

> Art into art, the acquisition of style by limitation, is the way almost all artists learned their skills until very recently. Certainly they worked from life, taking inspiration whenever they found it but the fundamental techniques of design, of drawing and modeling, were gained from other art. The study of plaster casts, the visit to the church or, in later centuries, to the museum to copy the works of others, the examination of illustrated woods, these were part of the experience of every Western artist. Copying was the right and logical thing to do. It was the way one learned, and it kept the artist in touch with the wellsprings of the past.[4]

The Paduan painter Cennino Cennini advises the young artist to begin by:

> constantly copying the best things which you can find done by the hand of the great masters. . . . You will eventually acquire a style individual to yourself, and it cannot help being good. ...5

A century later, Leonardo da Vinci prescribed a course of study in which the young artist turned to nature only after copying first from drawings, then from paintings, then from sculpted reliefs.[6]

Renaissance art theorists prescribed copying as a way of reaching beyond the imperfections of the real world to achieve the ideal. Leonbattista Alberti, the most important theorist of the 15th century, first applied the Aristotelian idea of imitation (mimesis) to painting, defining the goal of art to be the imitation of nature ***as it*** *ought to be.* Models for this perfected nature could be found in the art of antiquity and of modem masters who had themselves learned from antiquity. Over the course of the 16th century art became increasingly self-referential,

---

1 This material is excerpted tire: "Copying in the Visual Arts in Europe Since the Renaissance: Forms, Functions, Response and Restrictions," contractor paper, Robert EchoIs, July 1991.

2 Ibid., p. 1.

3 Ibid., p. 2.

4 Bruce Cole, The *Renaissance Artist at Work (New York, NY: Harper&* Row, 1983) p. 31.

5 Cennino Cennini, *The Craftsman's Handbook: "I1 libro dell'arte (trans.* Daniel V. Thompson, Jr., New York, NY: Dover, 1954), p. 15.

6@& Haverkamp-Begemann and Carolyn Logan, ''Creative Copies: Interpretive Drawings from Michelangelo to Picasso,' exhibition catalogue, The Drawing Center, New York, 1988, p. 17.

flicting elements, and Congress has considered them in enacting copyright legislation.

Much of the historical balancing has occurred in the context of the fair-use doctrine. The doctrine of fair use as codified in the "1976 Copyright Act has antecedents in English law of the 18th and 19th centuries and in 19th-century U.S. case law.[127] Various approaches have been adopted to interpret the fair-use doctrine. It has been said that *the* doctrine of 'fair use' allows the courts to bypass an inflexible application of copyright law, when under certain circumstances it would impede the creative activity that the copyright law was supposed to stimulate.[128] Indeed, some commentators have viewed the flexibility of the doctrine as the ''safety valve" of copyright law, especially in times of rapid

---

127 For thorough treatment of the evolution of fair use and an analysis of case law and the fair usc factors, see William Patry, *The Fair Use Privilege in Copyright Law* (Washington, DC: The Bureau of National Affairs, 1985).

128 See *Harper* & Row, *Publishers, Inc.* v. *Nation* Enterprises, 471 U.S. 539 (1985); *Iowa State University Research Foundation, Inc. v. American Broadcasting Co., 621* F.2d 57 (2d Cir. 1980).

especially in Rome and Florence; the imitation of styles and the quotation of forms from the work of past masters was the basis of the *maniera* "*the* correct way of painting. ' Later in the century the reform movement led by the Carracci reacted against overreliance on past masters, but advocated not a rejection of the past by greater "invention" in imitating its models. In the 17th century the influential critic Bellori returned to the Aristotelian notion of imitation:

> *The* idea, originating in nature, supersedes its origin and become the origin of art.

The idealized nature found in the great works of art, ancient and modem, was thus the appropriate source of inspiration for the artist.[7]

Such theories of art were codified and put into practice in the art academies that gradually became the principal institutions controlling the production of art: Lorenzo de' Medici's informal school for artists in Florence, where Michelangelo studied the antique sculptures in the Medici collection; the Academy of the Carracci in Bologna; the Accademia di San Luca in Rome; the French Academy, founded in 1648; numerous academies in Germany; and eventually the Royal Academy in England. In Vienna, the Empress Maria Theresa opened the Imperial Gallery to students, providing them with a room in which to make their copies. Although copying as a mode of art instruction has gone out of present day fashion, visitors to museums can observe that the practice still has its adherents.[8]

The "translation of images" from artist to artist extends beyond copying to what is generally referred to as "quotation" or "borrowing' '-the use of compositions or motifs taken from other works of art. A typical monograph or catalogue on a Renaissance subject lists many instances of this practice. The National Gallery of Art's 1979 catalogue entry on Tintoretto's Conversion of Saint Paul identifies motifs in the painting taken from Leonardo, Pordenone, Titian, Schiavone, and Raphael, and notes that the influence of Tintoretto's painting may be seen in a work by Rubens.[9] One critic states that:

> During the Renaissance and long after it, imagery was still largely shared, . . . The sense, to which we attach so much importance, of the personality of the artist with its incumbent personal baggage of imagery and manner was in embryo and the vocabulary of images was datively small. 10

The English artist Fuseli said that he found William Blake "damn good to steal from.' '[11] Manet took the basic funeral motifs of two of his most avant-garde paintings, *Olympia* and Le *dejeuner sur l'herbe,* from Titian's *Venus of Urbina* and a print by Marcantonio Raimondi after Raphael, respectively. Such practices have continued into the 20th century in various forms of paraphrase, parody, quotation, collage, and most recently, "appropriation."

[7] *The summary* of artistic theories of imitation in this paragraph is based on overviews provided in Haverkamp-Begemann ibid., pp. 16-20; and Rosario Assunto, "Mimesis," *Encyclopedia of World Art,* vol. 10 (New York, NY: McGraw-Hill, 1%5), pp. 93-117.

[8] *Children of Mercury: The Education of Artists in the Sixteenth and Seventeenth Centuries* (Providence: Brown University, 1984).

[9] Fern R. Shapley, *Catalogue of the Italian Paintings (Washington DC: The* National Gallery of Art, 1979), PP. 468-469.

[10] Ayrton in K.E. Maison, Theme *and Variations: Five* Centuries *of Master Copies and Interpretations, 2d* ed. (London: H.N. Abrams, *1966),* p. *16.*

[11] Ibid., p. 21.

technological change. Others have considered the uncertainties of the fair-use doctrine the source of unresolved ambiguities.

In codifying the fair-use exception in the Copyright Act of 1976, Congress did not formulate a specific test for determining whether a particular use was to be construed as a fair use. Rather, Congress created statutory recognition of a list of factors that courts should consider in making their fair-use determinations. The four factors set out in the statute are:

1. the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
2. the nature of the copyrighted work;
3. the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
4. the effect of the use on the potential market and value of the copyrighted work (17 U.S.C. 107 (1988)).

Congress realized that these factors were 'in no case definitive or determinative" but rather "provided

some gauge [sic] for balancing equities. ''129 It appears that Congress developed a flexible set of criteria for analyzing the circumstances surrounding each fair-use case, and that each case would be judicially analyzed on an ad hoc basis. Therefore, courts seem to have considerable latitude in applying and evaluating fair-use factors.[130] Courts have given different weight and interpretation to the fair-use factors in different judicial determinations. The following illustrations demonstrate how some courts have interpreted certain fair-use factors.

In evaluating the first factor, the purpose and character of the use, courts have not always held that use ''of a commercial nature' precludes a fair-use finding,[131] nor does a "nonprofit educational" purpose mandate a finding of fair use.[132] A defense of fair use on the basis of the first criterion will more often be recognized, however, when a defendant uses the work for educational, scientific, or historical purposes.[133]

Consideration of the second factor, the nature of the copyrighted work, must be based on the facts and circumstances of each particular case. For instance, courts have interpreted the scope of the fair-use doctrine narrowly for unpublished works held confidential by their authors.[134]

In examining the third factor, the amount and substantiality of the portion of the work used, courts have looked at both the quantitative aspect-how much of the work is used [135]—and the qualitative factor—whether the "heart" or essence of the work is used.[136] The fair-use doctrine is usually not considered to be applicable when the copying is nearly a complete copy of the copyrighted work, or almost verbatim.[137] Prior to the Court of Claims' decision in *Williams & Wilkins"ns Co. v. United States,*[138] courts as a rule did not allow fair use for copying of entire works or substantial portions of a work. However, the issue of copying entire works was the topic of significant debate prior to passage of the 1976 act. The result of this debate, which allows for this kind of copying under limited circumstances, is found in section 108, which sets out guidelines for classroom copying, and in interpretation of fair use in the legislative reports.[139]

In assessing the fourth factor, courts have examined the defendant's alleged conduct to see whether it poses a substantially adverse effect on the potential market for, or value of, the plaintiff's present work.[140] These considerations are used with great care by the courts in applying the fair-use doctrine on a case-by-case basis.

Congress looked to the issue of copyright fair use at some length in 1991, examining whether the fair use doctrine and the First Amendment permit biographers to make unauthorized use of their subject's unpublished letters and manuscripts. The courts have decided this issue on the basis of the specific facts of each case, but emphasizing the unpublished nature of the work in denying fair use.

In 1991 the Senate passed S. 1035 to clarify that the unpublished nature of a copyrighted work does

---

129 H.R. Rep. *No.* 1476, 94th Cong., 2d Sess. 65 (1976).

130 For a historical analysis of the fair-use factors, see William Patry, op. cit., footnote 127, ch. 17.

131 *Harper & Row Publishers, Inc.* ~. Nation *Enterprises,* 471 U.S. 539,593 (1985) (Brennan, J., dissenting); *Consumers Union of U.S., Inc. V. General Signal Corp., 724* F.2d 1044 (2d Cir. 1983).

132 *Marcus* v. *Crowley,* 695 F.2d 11.71 (9th Cir. 1983).

133 See *Italian Book* COW. v. *Amen'can Broadcasting Cos., 458* F. Supp. *65* (S.D. N.Y. 1978).

134 A recent case articulating the fair-use doctrine involved the personal correspondence of author J.D. Salinger. The court determined that the author had a copyright interest in his correspondence. *Salinger v. Random* House, *Inc.,* 811 F.2d 90 (2d Cir. 1987), *cert. denied, 108* Sup. Ct. 213 (1987).

135 Consumers *Union of* U. S., *Inc. v. General Signal Corp., 724* F.2d 1044 (2d Cir. 1983).

136 *Maxtone-Graham v. Burtchaell, 803* F.2d 1263 (2d Cir. 1986).

137 Walt Disney Productions v. *Air Pirates,* 581 F,2d 751 (9th Cir. 1978), *cert. denied* 439 U.S. 1132 (1978). But see *Universal City Studios, Inc.* v. *Sony Corp. of America, 480* F. Supp. 429 (D.C. Cal. 1979, *rev'd,* 659 F, 2d %3 (9th Cir. 1981), *rev'd, 464* U.S. 417 (1984). It might well be noted, however, that with respect to the questions of 'amount and substantiality of the portion used" in the area of computer programs, this aspect of the fair use limitation is argued by some to be of little use, because, as they assert, copies of only part of a program are rarely useful. Mary Jensen, University of South Dakota, School of Law, personal communication, September *29, 1991.*

138 Williams & Wilkins Co. v. *United States,* 172 U.S.P.Q. *670* (Ct. Cl. 1972), 487 F.2d 1345, 180 *U.* S.P.Q. *49* (Ct. *Cl.* 1973), *aff'd by an equally divided court, 420* U.S. *376,* 184 U. S.P.Q. 705 (1975).

139 William Patry, op. cit., footnote 127, pp. 449-450.

140 This factor was of considerable importance in *Universal City Studios, Inc.* v. *Sony Corp. of America,* 480 F. Supp. 429 (D.C. Cal. 1979), *rev'd,* 659 F.2d 963 (9th Cir. 1981), *rev'd, 464* U.S. 417 (1984). See also *Consumers Union of U.S., Inc. v. General Signal Corp., 724* F.2d 1044 (2d Cir. 1983).

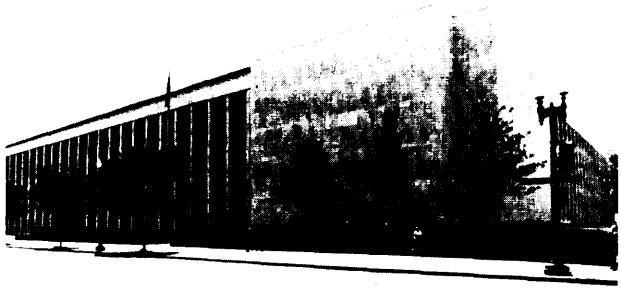


Photo credit: U.S. Library of Congress

The Copyright Office is housed in the James Madison Building of the Library of Congress.

not per se preclude applicability of the fair use defense to infringement. A similar measure was deleted from H.R, 2372 when a district court ruled in favor of a biographer in *Wright v. Warner Books.*[141]

The Copyright Office

The Copyright Office is charged with administration of the copyright law, and is an influential source of legal authority in the field of copyrights. Courts give weight to the Copyright Office's interpretation of the copyright statute.

The 1976 act authorizes the Copyright Office to issue regulations concerning the requirements for registration of claims for copyright and concerning fixation and placement of copyright notice.[142] These regulations contain authoritative interpretation of the copyright statute, which has been cited with approval by the courts.

The Copyright Office may issue certificates of registration for deposited material only if it determines that it constitutes ''copyrightable subject matter. [143] Thus, the Copyright Office's decision to register a work is evidence that a work is copyrightable. Likewise, a denial of registration indicates that the work is not eligible for protection.[144] Weight is given by the courts to the Copyright Office's determination about registrability of a particular category of works. The Copyright Office maintains statistics regarding the rate of issuance of copyrights generally and for computer-related works. (See tables 2-2 and 2-3.)

Table 2-2—Copyright Registrations, Generally: 1974 to Present

Year	Number
1974	372,832
1975	401,274
1976	410,969
1976[a]	108,762
1977	452,702
1978	331,942
1979	429,004
1980	464,743
1981	471,178
1982	468,149
1983	488,256
1984	502,628
1985	539,165
1986	560,212
1987	581,276
1988	565,801
1989	611,328
1990	617,241
1991 (January to June only)	332,582

aTransitional qUarter—Registrations made July 1, 1976, through September 30, 1976, reported separately owing to the statutory change making the fiscal years run from October 1 through September 30 instead of July 1 through June 30.

SOURCE: U.S. Copyright Office.

Table 2-3-Copyright Registration, Computer-Related Works: FY 1986 to Present

Year	Textual	Unpublished textual works
1986	5,565	4,744
1987	5,137	4,433
1988	5,124	6,046
1989	4,397	5,412
1990	5,350	5,214
1991 (1st quarter only)	2,441	4,519

NOTE: "Textual works" is a Copyright Office Administrative classification which, in the case of computer-related works, includes such items as user manuals, floppy disks, and program code, i.e., all aspects of computer-related works which are written.

SOURCE: U.S. Copyright Office.

To facilitate efficient issuance of copyright and to be of assistance to the public, the Copyright Office issues forms and instructions for copyright registration, as well a series of circulars discussing copyright issues. Of particular interest is *Circular R61,*

141 *Wright* v. *Warner Books,* 74$ F. Supp. 105 (DC SNY 1990). The Second Circuit affirmed.

142 The Berne Convention Implementation Act (now part of Title 17) amends the U.S. copyright notice requirements. See discussion of the Beme Convention, ch. 3.

143 Registration is not a *Perquisite* for copyright protection, but may be required for court enforcement of the copyright.

144 Such denial is not a *conclusive* finding of lack of eligibility for protection.

which deals with copyright registration of computer programs, discussing regulations, policy, and practice related to software. In addition, the Copyright Office publishes a more general document, the *Compendium of Copyright Office Practices,* which sets forth guidance of the Office in making registrations and recording documents. While these forms and guidelines reflect practices, policies, and legal interpretations, the positions contained in the documents are not necessarily adopted by the courts.[145]

CONTU Recommendations Regarding Registrations and the Rule of Doubt

CONTU'S Final Report included recommendations for regulations concerning the registration and deposit of databases and other works fixed in computer readable media, These state that the registration and deposit regulations should permit and encourage registration and periodic updating of identifying material rather than the actual databases. With respect to computer programs, the CONTU report suggests that regulations relating to deposit and registration requirements should "promote public access to computer programs while being flexible enough to accommodate future changes in computer technology. Recognizing that repeated deposit of each version of a program would be burdensome to the program proprietor and the Copyright Office, the Commission set forth a variety of options, including a system of temporary deposit, and permanent deposit of complete copies of the original version of the program with subsequent filing of descriptions rather than complete copies of amended versions.

Rule of Doubt

The copyright law in general requires that an application for registration be accompanied by the deposit specified in the statute and the regulations issued under the statute.[146] The deposit must include "one complete copy" of an unpublished work or "two complete copies" of a published work. The Copyright Office is authorized to issue regulations permitting the deposit of "identifying material" instead of the actual copies.

As discussed, CONTU contemplated that the regulations would permit the deposit of identifying material other than "actual databases, ' and asserted that computer databases were appropriate for exemption from the deposit requirement. Further, the legislative history of the optional deposit provision states that the regulations could provide for the deposit of printouts of computer programs under certain circumstances.

The Copyright Office will issue a certificate of registration, even when an applicant for registration of a computer program containing trade secrets is not willing to submit source code and submits object code instead. When it issues such a registration, it does so under a "rule of doubt" procedure.[147] The "rule of doubt" is more accurately described as the rule of "the benefit of the doubt" (in favor of the copyright applicant).[148] If the application is otherwise proper, the Copyright Office will issue such a registration, which makes it clear that no determination has been made concerning the existence of copyrightable authorship. The Copyright Office issues its registration on this limited basis because of its belief that the object code is ''basically unintelligible" to its examiners, so that they cannot make a definitive determination of its copyrightability .149 In order to receive such a rule of doubt registration, the applicant must submit a letter stating that the program does contain original authorship.

While there is no clear case law delineating how the rule of doubt registration affects the status of the registered work in litigation, it is likely that such a registration would not be accorded the same weight as a conventional registration. The Copyright Office has recognized that in making this kind of registration, the burden is placed on the courts to make a determination about the existence of copyrightable authorship. This additional burden is especially important in the case of requests to the court for preliminary relief in the form of temporary restraining orders and preliminary injunctions.

---

145 *Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc.,* 797 F.2d 1222, 1242, n.38 (3rd Cir. 1986), cert. *denied, 479* U.S. *1031* (1987).

146 However, under the Berne Convention and the Berne Implementation Act, registration and deposit themselves are not required.

147 37 C.F.R. 202.20(c) (2)(vii)(B).

148 Ronald Laurie, Weil, Gotshal & Manges, personal communication, September 1991.

149 Cary Sherman et al., op. cit. footnote 90, at Section 208.2(g)(5).

CONTU and the 1980 Amendment

Congress established the National Commission on New Technological Uses of Copyrighted Works to make recommendations for computer copyright legislation. The Final Report, published in 1978, is generally regarded as quasi-legislative history. While the relevance of the report is somewhat discounted or even ignored by some courts,[150] it is used by many courts as an aid in interpreting the 1980 amendment.

In carrying out its congressional charter, CONTU made specific recommendations for legislation dealing with computer software or programs, databases, and works created by the use of computers.

With respect to computer software or programs, CONTU recommended that the new copyright law should be amended:

1. to make it explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright;
2. to apply to all computer uses of copyrighted works by the deletion of section 117; and
3. to ensure that rightful possessors of copies of computer programs may use or adapt these copies for their use.

CONTU also recommended that the 1976 act be amended to apply to all computer uses of copyrighted databases and other copyrighted works fixed in computer media, and that works created by the use of computers should be afforded copyright protection if they are original works of authorship within the 1976 act.

These recommendations allayed doubts concerning the copyrightability of programs under the 1976 act,[151] as the 1980 act expressly added a definition of ''computer program' in section 101.[152] Further, the old version of section 117[153] was repealed and replaced by what is now section 117 of the current statute, which provides a defense to a claim of software copyright infringement if the defendant's activity falls within its scope. Section 117 insulates from infiingement liability certain steps that CONTU considered essential in the utilization of a computer program in conjunction with a computer.[154]

Section 117 of the Copyright Act now provides as follows:

> Notwithstanding the provisions of section 106, it is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:
>
> 1. that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or
> 2. that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.
>
> Any exact copies prepared in accordance with the provisions of this section may be leased, sold or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

Only the ''owner of a copy of a computer program' may take advantage of the privileges set out in section 117. An owner is one who purchases a disk, diskette, or other medium in which a program is stored. A person who does not own a copy of the plaintiff's computer program may not take advantage of a section 117 defense. The licensee who acquires possession but not ownership of a disk copy

---

150 See *Whelan Associates, Inc.* v. *Jaslow Dental Laboratory, Inc.*, *797* F.2d 1222, 1241, which commented that the report was nOt binding and did not consider it authoritative legislative history. See also *Lotus* v. *Paperback Software, 740* F. Supp. 37 (D. Mass. 1990).

151 The House Report of the 1976 act indicates that computer programs are to be considered ''fite~ works'' (H.R. Rep. No. 1476, 94th Cong., 2d Sess. 54, reprinted in 1976 U.S. Code Cong. & Ad. News 5659,5667 ("literary works'. . . includes]. . . computer programs.'')

152 Section 101 defines a computer program as a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result. Some commentators maintain that this amendment merely suggested that writings otherwise copyrightable could be in the form of a computer program. Stephen Y. Chow, personal communication, Cesari and McKenna, Sept. 27, 1991.

153 The original text of Section 117 stated:

> Notwithstanding the provisions of sections 106 through 116 and 118, this title does not afford to the owner of copyright in a work any greater or lesser rights with respect to the use of the work in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information, or in conjunction with any similar device, machine, or process, than those afforded to works under the law, whether title 17 or the common law or statutes of a State, in effect on December 31, 1977, as held applicable and construed by a court in an action brought under this title.

154 See CONTU Report at 13.

of a licensed program is not entitled to exercise these privileges. There must be a transfer of title, as provided for under applicable State law, for the privilege to obtain.

This requirement of ownership in section 117 is a change in the statutory language from the CONTU recommendation. CONTU would have allowed a rightful possessor of a copy of a program to perform or authorize the acts permitted by the section. There appears to be no legislative history on the reasons for this change; however, those who were involved in the congressional hearing on the 1980 amendments say that the change reflected concerns of the Justice Department relating to antitrust considerations.[155]

Evolution of Case Law

Complete coverage of software copyright case law is beyond the scope of this report. However, protection of software via copyright has involved several key issues. Three of the most important are: 1) whether object as well as source code is protected; 2) whether a program's structure, sequence, and organization is protected (and what such protection implies); and 3) whether the user interface is protected. A summary of the evolution of cases addressing these three key issues follows.

Apple v. Franklin

The case of *Apple v. Franklin*[156] specifically addressed the question of whether a copyright can exist in a computer program expressed in object code as well as source code. The court described source code as usually written in a higher-level programming language and object cede as the version of the program in which the source code language is converted into (binary or hexadecimal) machine language. The court determined that both the source code and the object code are copyrightable.

In its decision, the court traced the legislative history which, it stated, suggests that computer programs are considered copyrightable as literary works under section 102(a) of the Copyright Act.[157] CONTU later recommended that the copyright law be amended "to make it explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright. '[158] In accord with the CONTU report recommendations, the 1980 amendments to the Copyright Act included a definition of a computer program:

> A "computer program" is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result. (17 U.S.C. 101.)

The court further noted that language of 17 U.S.C. 117 carves out an exception to normal proscriptions against copying of computer programs, thus indicating that programs are, in fact, copyrightable and are otherwise afforded copyright protection. Indeed, the Third Circuit Court of Appeals had, in the prior case of *Williams Electronics, Inc. v. Artic International Inc.,*[159] concluded that "the copyrightability of computer programs is firmly established after the 1980 amendment to the Copyright Act. "

In arriving at its finding that object code as well as source code are copyrightable, the court in *Apple v. Franklin* also stated that, under the statute, copyright extends to works in any tangible means of expression "from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device.[160] As stated above, the definition of a ''computer program" adopted by Congress in the 1980 amendment is a "set of statements or instructions to be used *directly* or *indirectly in* a computer in order to bring about a certain result. '[161] Since source code instructions must be translated into object code before they can be utilized by the computer, only instructions expressed in object code can be used "directly' by the computer, and as such, object code falls under the definition in the statute. Further, the court emphasized that a computer program in object code could be classified as a literary work, since the category "literary work," one of the seven copyrightable

155 Personal communication, Ronald Laurie, Weil, Gotshal & Manges, September 1991.

156 *Apple* v. *Franklin* 714 F.2d *1240* (3rd Cir. 1983).

157 See H.R. Rep. No. 1476, 94th Cong., 2d Sess. 54, reprinted in 1976 U.S. Code Cong. & Ad. News 5659,5667 ('literary works'. . .include[s]. . computer programs").

158 National Commission on New Ethnological Uses of Copyrighted Works, *Final Report.*

159 *Williams Electronics, Inc. v. Artic International Inc., 685* F.2d 870 (3rd CU. 1983).

160 17 U.S.C. Section 102(a).

161 17 U.S.C. Section 101.

categories in section 101, includes expression not only in words but also ''numbers, or other . . . numerical symbols or indicia.' Thus, the court held that a computer program, whether in object code or source code or whether an operating system or application program, is a ''literary work' and is protected from unauthorized copying. The court cited *Midway Mfg. Co. v. Strohon*[162] and *GCA Corp. v. Chance*[163] *as in accord* with its holding.

Apple v. Franklin addressed the issue of copyrightability of operating and application systems. While Franklin conceded that application programs are an appropriate subject of copyright, it contended that operating systems are not the proper subject of copyright regardless of the language or medium in which they are freed, and that operating system programs are per se excluded from copyright protection under the express copyright terms of section 102(b) of the Copyright Act and under the precedent and underlying principles of *Baker v. Selden.*[164] According to the court, ''an application program usually performs a specific task for the computer user" (e.g., word processing, checkbook balance function) while "operating system programs generally manage the internal function of a computer or facilitate the use of an application program" (e.g., translates an application program from source code to object code.) Franklin based its argument on the grounds that an operating system program is either a ''process,' ' 'system,' or ''method of operation' and hence uncopyrightable, since section 102(b) specifically precludes copyright protection for these.

In *Apple v. Franklin, the court* found that operating system programs are copyrightable. The court pointed to prior courts which rejected the distinction between application programs and operating system programs. The court also cited the CONTU majority and the Congress, neither of which distinguished between operating system and application programs. The court reasoned that, since both operating system programs and application programs instruct the computer to do something, it should make no difference under section 102(b) whether these instructions tell the computer to prepare an income tax return or translate high level language from source to binary object code. The court stated that, "Since it is only the instructions which are protected, a process is no more involved because the instructions in an operating system program may be used to activate the operation of the computer than it would be if instructions were written in ordinary English in a manual which described the necessary steps to activate an intricate complicated machine.' [165] The court found no reason to afford any less copyright protection to the instructions in an operating system program than to the instructions in an application program.

Structure, Sequence and Organization[166]

The concept of 'structure, sequence, and organization' is found outside the area of computer software in elements such as the plot, subplot, sequence of scenes, setting characterization and patterns of dialogue in works of fiction or drama; or in the detailed outline and organization and selection, coordination and arrangement of information in textbooks or other nonfiction works. In computer software, structure, sequence, and organization include the arrangement of computer program modules in relation to each other, as opposed to the literal text of the program.[167] The cases addressing the issue of the protectability of the structure, sequence, and organization of a program have found that courts must look beyond the literal text of the defendant's program to determine whether there is substantial similarity to the plaintiff's program.

Whelan Assocs., Inc. v. Jaslow Dental Laboratory, Inc.[168] *presented the issue* of whether there can be "substantial similarity' of computer programs when the similarity exists in the structure, sequence, and organization of the program and there is no line-for-line copying. The case involved a program designed by the plaintiff to run a dental laboratory

162 *Midway Mag. co. v. Strohon* 564 F. Supp. at 75@751.

163 *GCA Corp. v. Chance* 217 U. S.P.Q. at 719-20.

164 *Baker v. Selden*, 101 U.S. 99, 25 L.Ed. 841 (1879).

165 *Apple* v. Franklin, *714* F.2d at 1251.

166 For a more extensive discussion of issues of structure, sequence, and org anization, see Morton David Goldberg, 'Copyright Protection for Computer Programs: Is the Sky Falling?" *American Intellectual Property Assn. Quarterly Journal*, vol. 17, pp. 294-322 (1989).

167 Ibid.

]68 *Whelan Assocs., Inc.* v. *Jaslow Dental Laboratory, Inc.*, 609 F. Supp. 1307 (E.D. Pa. 1985), *aff'd*, 797 F.2d 1222 (3d Cir. 1986), *cert. denied, 479 U.S. 1031 (1987).*

business, written for the IBM Series 1 computer in Event Driven Language (EDL). The defendant's program was written in BASIC for the IBM PC computer. The evidence demonstrated the defendant's access to the plaintiff's source code.

The court found that computer programs were protected under copyright against "comprehensive nonliteral similarity, " and held that "copyright protection of computer programs may extend beyond a program's literal code to its structure, sequence and organization. In the particular case of *Whelan,* copyright did protect the structure, sequence, and organization. The court defined the protectable expression in the structure, sequence, and organization to include everything about the program's construction and design except its basic purpose or function (i.e., ' 'efficient operation of a dental laboratory '').[169] Thus, it was possible to infringe the copyright of" a computer program without verbatim copying of the computer code. In a significant footnote, the court stated that it did not intend to imply by this characterization of copyrightable expression that the idea or purpose behind every utilitarian or functional work will be exactly what it accomplishes, so that structure and organization would therefore always be part of the expression of those works. It drew the distinction between the situation in *Whelan,* and instances where the idea or purpose behind a utilitarian work is to accomplish a certain function in a *certain way,* such that the structure or function of a program is essential to that task.[170]

Other cases reflect the court's reasoning in *Whelan.*[171] *In SAS Institute, Inc. v. S&H Computer Systems, Inc.*[172] the court found that S&H infringed the copyright held in a program called SAS 79.5, which was written to run on IBM and IBM-compatible computers by converting it to run on Digital computers. The court cited instances of ''literal, near literal and organizational copying,' of structural detail and nearly exact duplication of the SAS structure and organization. The court also discussed the idea of merger of idea and expression, stating:

> . . . throughout the preparation of a complicated computer program such as SAS, the author is faced with a virtually endless series of decisions as to how to carry out the assigned task. ., At every level, the process is characterized by choice, often made arbitrarily, and only occasionally dictated by necessity. Even in the case of simple statistical calculations, there is room for variation, such as the order in which arithmetic operations are performed. . . As the sophistication of the calculation increases, so does the opportunity for variation of expression.

Finding that the processes of SAS could be expressed in a variety of ways, the Court stated that:

> . . . to the extent that similarities between the SAS and the S&H product have existed, they represent unnecessary, intentional duplication of expression.

Q-Co. *Industries, Inc. v. Hoffman*[173] reflected the idea/expression merger concept. The defendants' program was written to run on an IBM PC in Pascal and IBM Assembler language. The court found that the defendants' program did not infringe plaintiff's program, written in Basic and Atari to run on an Atari 800-XL," in spite of similarities in the structure, sequence, and organization of the program, in addition to similarities in the program text between plaintiff and defendants' programs. In making this finding, the court stated that the similarities between the two programs were similarities in ideas rather than in expression. The use of functionally similar modules in the same sequence in the two programs was an inherent part of any program of the type developed by the plaintiff.

169 'T& aspect of the *Whelan* decision has been the subject of heavy criticism.

170 *Whelan Assocs.* v. *Jaslow Dental Laboratory, 807* F.2d 1256, 1260, footnote 34.

171 A number of cases uphold structure, sequence and organization protection as a legal principle and find substantial similarity on at least some of the facts in each case. See *Johnson Controls, Inc. v. Phoenix Control Systems, Inc., 886* F.2d 1173 (9th Cir. 1989); *Broderbund Software, Inc. v. Unison World, Inc., 648* F. Supp. 1127 (N.D.Ca. 1986); *Dynamic Solutions, Inc. v. Planning & Control, Inc., [1987]* Copyright L. Dec. (CCH) Para. 26,062 (S.D.N.Y. 1987); *Pearl System, Inc. v. Competition Electronics, Inc. 8* U. S.P.Q. 2d 1520 (S.D.Fla. 1988); *Soft Computer Consultants, Inc. v. Lalehzarzadeh, [1989]* Copyright L. Dec. (CCH Par. *26,403* (E.D.N.Y. 1988); *Manufacturers Technologies, Inc. v. CAMS, Inc., 706* F. Supp. 984 (D. Corn. 1989); *Lotus Development Corp.* v. *Paperback Software Int'l. 740* F. Supp. 37 (D. Mass. 1990); Customs Service Decision 90-40 (Jan. 10, 1990) File: HQ 732291 CPR-3 CO: R:C:V 732291 SO. 24 Cust. B & Dec. No. 14, p. 28, [1990] Guide to Computer Law (CCH) Par 60,212 (Apr. 4, 1990).

172 *SAS Institute,* Inc. v. *S&H Computer* Systems, *Inc., 605* F. Supp. 816 (M.D. Tenn. 1985).

173 Q-Co. *Industries, Inc. v. Hoffman, 625* F. Supp.608 (S. D.N.Y. 1985).

In Healthcare Affiliated Services, Inc. v. Lippany[174] *the court* held that the result of very general creative decisions were not protectable structure, sequence, and organization. Basing its findings upon plaintiff comparison of the first 50 lines of the two programs, the court stated:

> The evidence merely documents that certain choices were made among factors at a gross level, e.g., the scope of the system, the number of variables to be used or the portions of the work force to be included in calculations of labor hours. The result of these choices, however, do [not] constitute the programs' structure, sequence and organization within the meaning of *Whelan.*

The Fifth Circuit Court of Appeals did not follow *Whelan in Plains Cotton Cooperative Ass'n v. Goodpasture Computer Serv., Inc.*[175] Relying on expert testimony, the court found no copying when an allegedly infringing program, designed to run on a personal computer rather than a mainframe computer, was found very similar to the plaintiff's program on the functional specification. Even though the court found the two programs very similar with respect to programrning and documentation levels, and found that portions of the design appeared to be direct copies, the court looked to other evidence and found no copying. The court did not adopt the *Whelan* holding that the structure, sequence, and organization of a computer program is copyrightable. The court held that similarities in the two programs--each of which was designed to perform the same particular task within the agricultural cotton market-were dictated by the "externalities of the market. The record indicated that the market significantly affected the determination of the sequence and organization of cotton marketing software, since both programs attempted to provide the same information to the user. The court did not hold that such patterns could not constitute an idea in the context of computers. Thus, the decision in *Plains Cotton* narrowed *Whelan* such that the defendant can show that similarities in structure and organization may be dictated by market factors—externalities-so that the same information must be presented to the user.

NEC Corp. v. Intel Corp.[176] involved two parties whose microprocessors both utilized the Intel 8086/88 instruction set. NEC'S V-series microprocessors contained similarities to the hardware of the 8086/88 microprocessor, but also had additional hardware. Intel claimed that NEC'S microcode violated its 8086/88 microcode copyrights, but not that the hardware similarities or use of the microinstruction set violated its copyright. The court found no infringement, basing its holding on the following findings:

1. no substantial similarity of the works ' 'considered as a whole;
2. insufficient evidence that NEC copied important parts of Intel's microcode;
3. programming "constraints" accounting for similarities between the two microcodes; and
4. the limited number of ways in which to express the ideas underlying some of Intel's more basic microroutines.

The findings of the court were particularly well supported through the evidence of ''Clean Room' microcode presented to the court. NEC had contended that many of Intel's micro sequences were not copyrightable because they were made up of only a few obvious steps and thus lacked the originality necessary for copyright protection. NEC focused on cases cited by Melville Nimmer, in which copyright protection was denied to fragmentary words or phrases, noncreative variations of musical compositions, and forms of expression dictated solely by functional considerations, The court looked to Clean Room microcode, developed by a third party, as compelling evidence that the similarities between the NEC microcode and the Intel microcode resulted from constraints. It found that the Clean Room microcode was governed by the same constraints of hardware, architecture, and specifications as applied to the NEC microcode, and that copying was not involved. The developer of the 8086 microcode for Intel acknowledged that the microarchitecture of the

174 *Healthcare Affiliated Services, Inc.* v. *Lippany*, 701 F. Supp. 1142 (W.D. Pa. 1988). Additional cases have upheld structure, sequence, and organization as a principle but found the evidence or pleading insufficient. See Q-Co *Industries, Inc. v. Hoffman, 625* F. Supp. 608 (S. D.N.Y. 1985); *Digital Communications Associates, Inc. v. Softklone Distributing Corp., 659* F. Supp. 449 (N.D. Ga. 1987); *Telemarketing Resources v. Symantec Corp., 12* U. S.P.Q. 2d 1991 (N.D. Cal. 1989); *Bull HN Information Systems, Inc. v. American Express Bank Limited, [1990]* Copyright L. Dec. (CCH) Par. 26,555 (S. D.N.Y. 1990).

175 *Plains Cotton Cooperative Ass'n. v. Goodpasture Computer Serv., Inc.*, 807 F.2d 1256 (5th Cir.) *cert. denied*, 484 U.S. *821* (1987).

176 *NEC Corp. v. Intel Corp ,* 645 F. Supp. 590 (N. D.Cal. 1986), *vacated, 835* F.2d 1546 (9th Cir. 1988), 10 U, S.P.Q. 2d 1177 (N.D. Cal. 1989).

8086 microprocessor affected the manner in which he created his microcode, and that he would expect that another independently created microcode for the 8086 would have some similarities to his. The court found that the similarities between the Clean Room microcode and the Intel microcode must be attributed in large part to these constraints.

With respect to the issues of copying and the limited number of ways in which to express ideas underlying basic microroutines, the court cited testimony that independently created microcode for the 8086 would have fewer similarities in the longer sequences than in the shorter sequences, because more opportunities exist for longer sequences to be expressed differently. The court found that this was borne out: the longer sequences in NEC'S code and in Intel's microcode were not nearly so much alike as the shorter sequences.

The court in *Computer Associates International, Inc. v. Altai, Inc.*[177] rejects the Whelan test of ''structure, sequence and organization' to determine similarities in computer programs. Instead, the court applied the "levels of abstractions test" articulated by Learned Hand in *Nichols v. Universal Pictures,*[178] which, they stated, was the law of the Second Circuit Court of Appeals. The "levels of abstractions test' of *Nichol's* reads:

> Upon any work. . . a great number of patterns of increasing generality will fit equally well, as more and more of the incident is left out. The last may perhaps be no more than the most general statement of what the [work] is about and at times might consist only of its title; but there is a point in this series of abstractions where they are no longer protected, since otherwise the [author] could prevent the use of his "ideas" to which, apart from their expression, his property is never extended.[179]

Applying this test, the court found no infringement of computer Associates' copyright,

User Interface (the Screen Display Cases)

Courts have also addressed copyright issues in disputes relating to computer program screen displays, distinguishing copyrightable expression from unprotected elements in the text, menu hierarchies, command structures, key sequences, and other aspects of a program's ''interface' with the user.

The court in *Broderbund Software, Inc. v. Unison World, Inc.*[180] held that the structure, sequence and organization of screen displays in defendant's 'Print Master" infringed the copyright on the audiovisual displays of the plaintiff's program, "The Print Shop. " Citing *Whelan, the* court upheld protection for the ''overall structure of a program, *including its audiovisual displays* [emphasis added]. ' According to the court, the *idea* of *creating printed materials* (which may vary infinitely in their combination of text and graphics) is the concept behind "The Print Shop" and "Print Master. " The created printed materials may vary indefinitely in their combination of text and graphics, and thus the idea is separable from the expression in the screens.

Broderbund differs from the earlier case of *Synercom Technology, Inc. v. University Computing Co.,*[181] *in* which the court considered the issue of whether the sequence and ordering of plaintiff's input formats used in a structural analysis program was protected expression or an unprotected idea. Synercom supplied its customers with instructions describing the order in which data should be entered in the analysis program. University Computing, providing its users with similar printed input instructions, filed suit. The court held that the sequence and ordering of data was inseparable from the idea underlying the formats. These were not, therefore, copyrightable.

In the case of *Digital Communications Associates* v. *Softklone Distributing Corporation*[182] *the court* **was** confronted with the question of what elements in a single menu screen constituted an idea and what elements constituted expression. The court concluded that the copied elements of the defendant's program that were nonessential to program operation constituted expression and therefore were infringements. It rejected Softklone's arguments that:

---

177 *Computer Associates Internatio*, *Inc.* v. *Altai, Inc.* No. CV 89.0811, U.S. District Corn, E.D. New York, Aug. 9, 1991. This decision has been appealed; arguments were to be heard January 9, 1992.

178 *Nichols* v. *Universal Pictures* 45 F.2d 119, 121 (2d Cir. 1930), *cert. denied,* 282 U.S. 902 (193 1).

179 *Nichols* v. *Universal Pictures,* 45 F.2d at 121.

180 *Broderbund Software, Inc. v. Unison World, Inc., 648* F. Supp. 1127 (N.D. Cal. 1986).

181 *Synercom* Technology, *Inc.* v. *University Computing CO.,* 462 F. Supp. 1003 (N.D. Tex. 1978).

182 Digital *Communications Associates* v. *Softklone Distributing* Corporation, 659 F. Supp. 449 (N.D. Ga. *1987).*

1) the idea and expression of the Crosstalk screen merged because the screen was a‘ ‘necessary expression of its idea, ’ and 2) the status screen was nothing more than an unprotectable ‘‘blank form’ designed to record the user’s choices of parameter values.

This issue was most recently addressed in *Lotus Development Corporation v. Paperback Software International.*[183] *The Lotus* decision extended the copyrightability of the nonliteral elements of computer programs to menu command structures. The structure, sequence, and organization of the menu command system were all found copyrightable—including the overall structure, the choice of letters, words, or symbolic tokens’ used to represent each command, the structure and order of the command terms in each menu line, the presentation of the command terms on the screen, and the long prompts.

Lotus brought suit in 1987 against Paperback Software International for copyright infringement of Lotus 1-2-3 by Paperback’s VP-Planner, which was advertised by Paperback to be a ‘‘workalike’ of Lotus 1-2-3. The programs were similar in appearance, and knowledge of Lotus 1-2-3 could be transferred to VP Planner without retraining. Although Paperback had not copied the literal elements of Lotus 1-2-3 (the source code or object code), the court found Paperback had copied the copyrightable nonliteral elements of the program.

The Lotus court established a three-part test for determination of the copyrightability of a particular nonliteral element. Applying this test, the court held that the idea of an electronic spreadsheet was not copyrightable. The rotated “L’ at the top of the screen used by Lotus to represent the headings and columns normally found on a paper spreadsheet the court found, was a format used by most other electronic spreadsheet computer programs. For these reasons, the court held that the rotated ‘‘L’ was not copyrightable. The court also held that the use of the slash key to evade the menu, the ‘‘enter’ key to invoke a command, and the arithmetic symbol keys were not copyrightable, because of the limited number of keys remaining on the computer keybourd which had not already been assigned some specific purpose (such as an alphabetical or numerical value.

The court in Lotus also concluded that the menu command structure is not essential to the idea of an electronic spreadsheet and that, as a result, Lotus 1-2-3’s menu command structure was copyrightable expression and infringed by VP-Planner. The court emphasized that each nonliteral element of the user interface may or may not be protectable and that the computer program must be viewed as a whole. ‘‘The fact that some of these specific command terms are not quite obvious or merge with the idea of such a particular command term does not preclude copyrightablity for the command structure taken as a whole. ’ To determine if illegal copying had occurred, the court found that it need only identify copyrightable elements and decide if those elements considered as a whole had been copied.

In the recent case of *Engineering Dynamics, Inc. v. Structural Software, Inc. and S. Rao Guntur,*[184] *the* court, in spite of plaintiff’s urgings, did not follow the reasoning of *Lotus,* looking instead to the Fifth Circuit for guidance. Citing *Plains Cotton, the court* held that formats are not copyrightable. Engineering Dynamics claimed defendants infringed several of its manuals in the development and marketing of defendant’s product StruCAD. It also claimed the defendants infringed its copyright in the ‘‘user interface, ’ comprised mainly of input and output reports, The court found that the scope of infringed materials included the text, pictures, diagrams, illustrative examples and flow charts depicted in the manuals, but not the input and output formats since the law of the Fifth Circuit provides that a user interface in the form of input and output reports is not copyrightable.

Databases

Databases are protected under copyright law as compilations. Under the copyright law, a compilation is:

> A work formed by the collection and assembling of preexisting materials or of data that are selected, coordinated, or arranged in such a way that the resulting work as a whole constitutes an original work of authorship (17 U.S.C. Section 101).

Copyright protection in a compilation does not provide protection for every element of the compilation. Section 103(b) of the Copyright Act provides that:

> The copyright in the compilation. . . extends only to the material contributed by the author of such work,

183 *Lotus Development Corporation v. Paperback Software International,* 740 F. Supp. 37 (D. Mass. 1990).

184 *Engineering Dynamics, Inc. v. Structural Software, Inc. and S. Rao Guntur, Civ.* Act. No. 89-1655.

> as distinguished from the preexisting material employed in the work, and does not imply any exclusive right in the preexisting material. The copyright in such work is independent of, and does not affect or enlarge the scope, duration, ownership or subsistence of, any copyright protection in the preexisting material.

Circuit courts of appeal have been inconsistent in their treatment of compilations. The Ninth Circuit Court of Appeals has consistently held that the discovery of a fact, regardless of the necessary input of labor and expense, is not the work of an author, so that verbatim repetition of certain words in order to use the nonprotectible facts is also noninfringing. To hold otherwise, according to the Ninth Circuit, would extend copyright protection to facts.[185] It is well established that copyright law never protects the facts and ideas contained in published works.[186] Indeed, the Ninth Circuit in *Cooling Systems & Flexibles Inc. v. Stuart Radiator, Inc.*[187] *stressed the* narrow range of protectable expression in factual works, acknowledging that to whatever extent the arrangement and expression of facts is original, an author is protected against its copying. 188 similarly, the Second Circuit requires that selection, coordination or arrangement is necessary to forma copyrightable compilation.[189]

The Eighth Circuit case of *West Publishing Co. v. Mead Data Central*[190] expands the scope of what is protectable as a compilation. West publishes texts of cases decided in State and Federal Courts and has developed a citation system in which cases can be found by reference to the volume number of the West volume and the page number on which that case appears. Mead Data publishes Lexis, a computer database of cases published by West as well as other Federal and State Court decisions. Mead Data uses the West citation system to locate cases, placing in its databases the first page on which a case appears and also the ‘‘jump pages” for each case. West claimed that the page numbering system of its reporters was copyrightable and the court agreed, holding that the compiling and arranging of the cases meets the originality requirement of the copyright law. Mead Data’s infringement consisted of taking the arrangement of the cases, not the numbers themselves. However, by using the citation system Mead had infringed West’s copyright in the arrangement and selection of cases.

The Seventh Circuit, on the other hand, took into consideration the author’s industry, or ‘sweat of the brow” in producing a compilation. In *Schroeder v. William Morrow & Company,*[191] *an* action for infringement of a copyright on a gardening directory, the court stated that copyright protects not individual names and addresses but compilation, *the product of the compiler’s industry* [emphasis added]. In making its finding of infringement, the court stated that:

> [i]t is clear ., . that the bulk of compilations in plaintiff’s directory were made with substantial independent effort and not by merely copying from other sources. The use of another copyrighted directory to obtain sources of information or for verification and checking, to the extent it occurred, was not wrongful and did not put plaintiff’s compilation beyond the protection of the statute.[192]

The Supreme Court finally addressed this issue in *Feist Publications Inc. v. Rural Telephone Service Co., Inc.*[193] *in* which it rejected the “sweat of the brow’ basis for copyrightability in fact-based works such as compilations. The court concluded instead that the Copyright Act of 1976 indicated that originality is the proper test in such cases. The *Feist* case involved the suit by Rural Telephone against Feist for copyright infringement, on grounds that Feist had illegally copied Rural Telephone’s phone listings.

---

185 *Worth v. Selchow & Righter Co , 827* F.2d 569 (9th Cir.1987).

186 *Harper & Row,* Publisher, *Inc.* v. *National Enterprises,* 471 U. S.539,105 Sup. Ct. 2218,85 L. Ed. 588 (1985); *Mazer v. Stein, 347* U.S. 201, 217, 74 Sup. Ct. 460,470, 98 L. Ed. 630, *reh'g denied, 347* U.S. *949, 74* Sup. Ct. 637, 98 L. Ed. 1096 (1954).

187 *Cooling Systems & Flexibles* Ire-. v. *Stuart Radiator, Inc., 777* F.2d 485 (9th Cir.1985).

188 Ibid., p.492; See also *Landsberg* v. *Scrabble Crossword Game Players Inc.,* 736 F.2d 485 (9th Cir. 1984) *cert. denied 469* U.S. 1037 (1984).

189 *Eckes* v. *Card Price's Update,* 736 F.2d 859 (2d Cir.1984), *Financial Information, Inc.* v. Moody's *Investor Service, Inc.,* 751 F.2d 501 (2d CU. 1984); 808 F. 2d *204* (2d Cir. 1986).

190 *West Publishing* CO. v. *Mead Data Central, 799* F. 2d 1219 (8th Cir.1987).

191 *Schroeder* v. *William* Morrow & *Company, 566* F.2d 3 (7th Cir.1977).

192 See also *Gelles-Widmer Co.* v. *Milton Bradley Co., 313* F.2d 143 (7th Cir. 1963).

193 *Feist Publications Inc.* v. *Rural Telephone Service Co., Inc.,* — U.S.—, 111 Sup. Ct. 1282 (1991).

The Court noted that the case involved two propositions of law which are traditionally in tension: first, that facts are not copyrightable and second, that compilations of facts generally are. The court concluded that while Feist clearly appropriated a significant amount of factual information from Rural Telephone's directory, Rural Telephone's selection, coordination and arrangement of its white pages did not satisfy requirements for copyright protection. The Court, therefore, held that Feist's taking of the listing could not constitute an infringement

Two new cases flow from Feist and appear to establish an emerging line of authority regarding the treatment of spreadsheets. The Second Circuit, in *Kregos v. Associated Press,*[194] found baseball pitching forms to be sufficiently original in the selection for copyright protection of nine categories out of the universe of pitching statistics. In *Key Publications Inc. v. Chinatown Today Publishing Enterprises Inc.,*[195] the Second Circuit upheld the copyright in the yellow pages of a Chinese-American community directory. The copyright was based upon the compiler's original selection and arrangement of business listings. At the same time, the court found the copyrighted directory not infringed by a competing directory that used a different arrangement of categories and principles of selection for included listings. This finding of non-infringement suggests that thin protection exists in a compilation.[196]

Other Concerns About Copying

Software developers, especially packaged-software developers, have also been concerned about two issues related to unauthorized copying: software rental and States' sovereign immunity from money damages for copyright infringement. These concerns have received congressional attention resulting in new legislation in the 101st Congress: Title VIII of Public Law 101-650 makes it an infringement of copyright to rent computer software without the copyright holder's permission; Public Law 101-553 allows Federal courts to hold the States and their agencies and employees liable for copyright infringement. Before the latter was enacted, Federal courts had refused to hold the States or their agencies (e.g., State universities) liable for money damages for copyright infringement, on the grounds that the copyright law does not clearly show the intent of Congress to abrogate the States' sovereign immunity under the 1 lth Amendment.[197]

The rental legislation was motivated by software industry concerns that most software rentals would be motivated by the desire to copy, rather than to "try before buying," and that software rental to potential copiers would displace sales. Similar concerns had previously resulted in the record-rental provisions of the current copyright law.

The *Semiconductor Chip Act-The* Semiconductor Chip Protection Act of 1984 was enacted to extend legal protection to a new form of statutory subject matter, semiconductor chip products and mask works,[198] According to the legislative history, the Semiconductor Chip Protection Act is intended to combat the problem of chip piracy,[199] as Congress perceived that the existing law failed to address that problem. In effecting this purpose, Congress attempted to incorporate the goals of the U.S. Constitution regarding copyrights and patents: to reward authors and inventors for their labors, to provide them with an incentive for future creativity, so as to ultimately benefit the public.

The Chip Act is a special or *sui generis* law, creating a statutory scheme to provide proprietary protection for chip products separate from and

194 *Kregos* v. *Associated Press,* 937 F.2d 700, 19 U. S.P.Q. 2d 1161 (2d Cir. 1991).

195 *Key Publications Inc.* v. *Chinatown Today Publishing Enterprises Inc.,* 20 USPQ1122 (2d Cir. 1991).

196 A number of other rulings have flowed from *Feist. Bellsouth Advertising & Pub. Corp. v. Donnelly Info. Pub., 933* F.2d *952* (11th Cir. 1991) held that copying the categories of a yellow page directory infringed that directory even though the copying was for unrelated use. *Victor Lalli Enterpr.* v. *Big Red Apple, Inc., 936* F.2d 671 (2d Cir. 1991) held that an particular format for reporting racing-related data was not copyrightable because it was a format used by many others and was dictated entirely by the intended use as a means to gamble on the numbers game.

197 See *BNA Patent, Trademark, and Copyright Journal,* vol. 41, January 1991, pp. 301-302.

198 Semiconductor chips are integrated circuits containing transistors, resistors, capacitors and their interconnection, fabricated into a very smallsingle piece of semiconductor material. A mask work is a set of images fixed or encoded at a later stage of manufacturing, that produces the circuitry of the final chip product. Stanley M. Besen and Leo J. Raskind, 'An Introduction to the J-aw and Economics of Intellectual Property, *The Journal of Economic Perspectives,* vol. 5, No. 1, pp. 3-27, at 19.

199 The legislative history indicates that incentives for piracy are great. There is a great disparity between the cost of developing a chip and the cost of copying it; the legislative reports indicate that initial development can cost as much as $100 million, while copying costs as little as $50,000.

320-228 0 - 92 - 6

independent of the Copyright Act.[200] Protection for domestic products attaches upon fixation and commercial exploitation.[201] ***Registration with the copy***right Office is a condition of mask work protection. Protection is forfeited if the mask work is not registered within 2 years after the date of first commercial exploitation. The Copyright Office makes provisions for registration. The act provides for a 10 year term. Owners of a protected mask work are granted the right to bar reproduction of the mask work by any means and the right to import or distribute a semiconductor chip product in which the mask work is embodied. The Act establishes reverse engineering as a defense to a claim of infringement. The reverse engineering provisions provide an exemption from infringement liability in spite of proof of unauthorized copying and striking similarity, as long as the resulting chip product was the result of study and analysis and contained technological improvement. The act also provides remedies similar to those associated with copyright protection. However, criminal penalties are not available, and the limit on statutory damages is higher than that provided for by the Copyright Act.[202]

Design Patent Protection-Design patents provide protection for designs for an article of manufacture that are new, original, and ornamental. The design may be surface ornamentation, configuration or a combination of both. Courts have defined a patentable ornamental design as one that must ''appeal to the eye as a thing of beauty. ''[203] As with other inventions granted patent protection, the subject of design patent protection must undergo an examination process in the Patent and Trademark Office and meet the standards of novelty and nonobviousness. The configuration of a useful object may constitute a patentable design, so that the elements of a design may be functional. However, a design dictated by considerations of function is not a proper subject for a design patent. A design is not patentable if the only points of novelty or nonobviousness over prior designs are dictated by fictional improvement or alteration.[204] once a patent is granted for a design, the term of protection is 14 years. For infringement of a design patent to exist, the accused article must be so similar to the protected one ''as to deceive an observer, inducing him to purchase one supposing it to be the other. . .''[205]

Industrial Design Bills in the United States— Industrial design protection is crafted to protect designs inadequately protected under patent, trade dress and copyright law. The history of proposals in the Congress of this method of protection is long.[206]

Several proposals to protect industrial designs were presented to the 100th Congress.[207] These proposals use a similar modified copyright approach. All three would have amended Title 17 to protect designs that are "original." The bills provided for copyrightlike registration process, rather than a patentlike examination process. Commonplace designs, those "dictated solely by utilitarian function' were excluded from protection. All provided for a term of protection of 10 years. Design rights, under the statute, would not affect any rights under patent, trademark, or copyright law. The bills required that notice of protection appear on the article. Copying an article without knowing that it was a protected design would not constitute an infringement.[208]

---

200 See Robert w. Kastenmeier and Michael J. Remington, "The Semiconductor Chip Protection Act of 1984: A Swamp Or Firm Ground?' *Minnesota Law Review*, vol. 70, No. 2, December 1985, pp. 417-470. According to Kastenmeier and Remington, while working in harmony with the copyright law, The Semiconductor Chip Protection Act avoids tailoring copyright principles to accommodate the singular character of the use of chip designs in the manufacturing process so as to distort the way in which copyright was applied to other categories of copyrightable works. At the base of their theory is the proposition that dissimilar things should not be treated in a similar fashion. See especially pages 443-444.

201 Foreign products are granted protection by the president upon a finding that a foreign nation extends to U.S. nationals the same protection as the United States accords to the foreign nationals.

202 Ibid.

203 *Wabern* P_ac_kag_in_g *Indus., Inc.* v. Cut Rate Plastic Hangers, *Inc.*, 652 F.2d 987,2 10 U. S.P.Q. 777 (2d Cir. 1981) *Bliss* v. *Gotham Indus., Inc.,* 316 F.2d 848, 137 U. S.P.Q. 189 (9th Cir. 1%3).

204 Chisum, *Patents,* Section 1.04[2][d].

205 *Gorham Mfg.* co. v. *White, 81* U.S. (14 Wall.) 511, 20 L. Ed. 731 (1872).

206 See, Jacques M. Dulin, ''Design Protection: Walking the Pirate Plank?'' *Bulletin, Copyright Society of the U.S.A.*, vol.12, No. 6, August 1%5; Note, ''Protection of the Design of Useful Articles: Current Inadequacies and Proposed Solutions," *Hofstra Law Review, vol.* 11, spring 1983, p. 1043 at p. 1065.

207 See H.R. 902, H.R. 3017, H.R. 3499.

208 "Court Rescinds Ruling That dBase Copyright Is Invalid," *BNA Patent, Trademark & Copyright Journal, vol. 4,* p. 543.

Opponents of the industrial design bills have argued that there is already sufficient incentive for production of articles of industrial design.[209] Other critics of the bills maintain that industrial design bill might, by virtue of its characterization as ‘ ‘industrial, ’ cover functional designs, such as automobile windshields, replacement parts, and product packaging, thus favoring original equipment manufacturers and brand name marketers over the makers of less-expensive after-market auto parts and store brand consumer products. Publishers are concerned about liability for publishing books that contain type face designs protected under the legislation.[210]

H.R. 1790, the Design Innovation and Technology Act of 1991, was introduced in the 102d Congress. This bill would amend the copyright law to provide for the protection of industrial designs of useful articles, including typefonts.[211] The bill sets the term of protection at 10 years and provides for requirements for marking, application, and fees. The bill specifies criteria for determination of infringement of a protected design and grants the owner of a protected design the exclusive right to make, import, or distribute for sale or use in trade any useful article embodying the design.

H.R. 1790 addresses concerns raised during hearings on design legislation held in 1990. As a result, it requires that protected designs meet a standard of ‘originality’ if they are to be protected, such that the design must be the result of a designer’s creative endeavor that provides a ‘‘distinguishable variation over prior work pertaining to similar articles. ” This variation must be more than trivial and must not have been copied from another source. The bill also expands an exemption for certain replacement parts for automotive and other products. The bill protects distributors and retailers who innocently trade in infringing products. Publishers are not subject to infringement actions under the legislation for reproducing, modifying or distributing printed materials even if these contain an infringing typeface. The aggrieved party must seek a remedy from the actual infringer. Finally, the legislation requires that the registrant for design protection forego simultaneous protection under the patent and copyright laws.

Design protection granting the designer or other owner of the design exclusive rights in the use of his creation has been enacted in foreign countries including Canada, France, the Federal Republic of Germany, India, Italy, Japan, the Netherlands, and the United Kingdom.[212] Such legislation in the United Kingdom and Canada has been recently enacted. Other laws, such as those of Germany and Italy, have been amended in recent years, The definition of industrial design may vary from country to country. However, it appears that generally design protection involves elements such as configuration, shape, pattern, and combinations of lines and colors which provide a product with a new or aesthetically improved appearance. Novelty and the industrial application of the design are generally required to obtain protection. In the countries listed above, the term of protection ranges from 8 to 15 years. The laws in these countries are enforced and provide for civil remedies in cases of infringement of exclusive rights. In some cases, the law provides for imposition of sanctions for criminal offenses.[213]

Hybrid Design Protection

One intellectual property scholar, Professor Jerome Reichman, has suggested that software is, like industrial design, an example of a ‘ ‘legal hybrid’ falling between the patent and copyright systems.[214] Other examples are biotechnology and medical processes. These hybrids are characterized by the fact that considerable investment is required to

209 For a discussion of some of the economic considerations associated with the protection of industrial design, see Robert C. Denicola, ‘‘Applied Art and Industrial Design: A Suggested Approach to Copyright in Useful Articles, ’ *Minnesota Law Review*, vol. 67, pp. 707, 721-727.

210 Ibid.

211 The bill specifically excepts designs that are: 1) not original; 2) staple or commonplace; 3) different from commonplace or staple designs in insignificant ways; 4) determined solely by a utilitarian function; 5) embodied in a useful article that was made public by the designer or owner in the United States or in a foreign country more than one year before the date of application for registration; 6) composed of three dimensional features of shape and surface in wearing apparel; 7) a semiconductor chip product already protected under another provision; 8) embodying a process or idea or system; or 9) for motor vehicle glass.

212 See Giovanni Salvo, ‘‘Industrial Design Protection,’ document of the Law Library of Congress, European Law Division, March 1990, LL90-23, pp. 1-2.

213 Ibid.

214 J. H. Reichman, “Computer Programs as Applied Scientific Know-How: Implications of Copyright Protection for Commercialized University Research,” *Vanderbilt Law Review*, vol. 42, No. 3, April 1989, p. 655,

achieve incremental innovation, and the "know-how" is vulnerable to rapid duplication by competitors who bear no part of the development expenditure. However, these products have fallen outside the copyright regime, and patent protection would not be available because the innovation is incremental, not ''non-obvious. '

Reichman believes that a *sui generis* know-how law built on modified copyright principles could provide adequate protection to this kind of legal hybrid without embracing the full copyright paradigm. He believes that this approach would eventually unify the treatment of innovations such as computer software and industrial design.

Reichman has written that:

> [t]he fundamental problem remains that of rewarding or simply recompensing large expenditure for incremental innovations that fall chronically short of the current legal threshold for patentable inventions.[215]

These technologies are not adequately protected because they deviate from the assumptions underlying the classical forms of intellectual property.[216] The solution, in Reichman's view, is a new intellectual property paradigm that provides this incremental innovation with artificial lead time in which investors can recoup their investment and turn a profit. *17

Other commentators have also argued that the patent and copyright laws are not appropriate for computer software, and that a *sui generis* law based on a modified copyright approach would be better.[218] However, while Professor Reichman argues that software belongs to a larger class of "legal hybrids' requiring a new intellectual property regime, these commentators favor the creation of a law specifically directed at computer software. One of the arguments which has been advanced against a *sui generis* regime for software is that it risks being obsoleted by changing technology.

Other arguments have been advanced for continuing to work within the existing patent and copyright regimes. First, it is argued that the present regimes are working well, and their economic effects are appropriate.[219] The CONTU report concluded that copyright law was an appropriate mechanism for protecting computer programs and, they claim, the case law has been evolving properly.[220] Further arguments against *sui generis* protection are that a new regime would create uncertainty, and that international copyright agreements provide a framework for the protection of computer programs in other countries.[221] (See box 2-H for discussion of Analagous Copyright Law in Foreign Countries.)

Trade Secret Law

Introduction

Trade secret law protects certain types of confidential technical or business information against unauthorized use or disclosure. Some believe that the object of the trade secret law is to protect *confidential relationships*,[222] and promote ethical standards of competitive behavior while others subscribe to the theory that its purpose is to protect the secret *information itself.* As with copyright and patent, the proprietary interest in the information

---

215 Ibid., at p. 653.

216 Ibid., at p. 661.

217 J. H. Reichman, ' 'proprietary Rights in the New Landscape of Intellectual Property Law: An Anglo-American Perspective, ' study prepared for the International Literary and Artistic Association (ALAI), Congress of the Aegean Sea II, Athens, June 19-26, 1991.

218 See Pamela Samuelson, ''Benson Revisited: TheCase Against Patent Protection for Algorithms and OtherComputer Program-Related Inventions, ' *Emory Law Journal*, vol. 39, *No. 4*, p. 1025, p. 1150; Richard H. Stern, "The Bundle of Rights Suited to New Technology,'' *University of Pittsburgh Law Review,* vol. 47, No. 4, p. 1229. Professor Paul Goldstein argues that copyright law runs the risk of providing too much protection to functional aspects of works belonging in the domain of patents. He further perceives problems with patent protection for software (prior art problems, problems of patenting obvious subject matter, etc.) so that, he asserts, subject matter is being protected which is not appropriately covered by patent or copyright law. He believes that, if improperly applied, the law will result in consumers paying higher prices for software than warranted, among other dislocations. See generally, Paul Goldstein, "Infringement of Copyright in Computer Programs," *University of Pittsburgh Law Review,* vol. 47, No. 4, Summer 1986.

219 Ronald T. Reiling, Chairman, proprietary Rights Committee, Computer and Business Equipment Manufacturers Association Testimony at Hearings before the House Subcommittee on Courts, Intellectual Property and the Administration of Justice, Nov. 8, 1989, Serial No. 119, p. 167.

220 Morton David Goldberg, op. cit. at footnote 166.

221 Ronald T. Reiling, op. cit. at footnote 218.

222 See *Rockwell Graphic Systems, Inc. v. DEV Industries, Inc.*, *925* F.2d 174 *(7th* Cir. 1991).

may differ from a proprietary interest in the physical object embodying the trade secret.[223]

Trade secret law is generally based on common law and contractual provisions. State law addresses all trade secret claims, even when a trade secret claim is tried in Federal court. As a result, the fine points of trade secret law may vary from state to state. However, a Uniform Trade Secrets Act (UTSA) has been adopted in about half the States.[224] In States where it has been adopted, the UTSA operates as a statute and is part of the State civil code. In spite of this state to state variation of trade secret law, one accepted definition of a "trade secret" is that of the *Restatement of Torts:*

> A trade secret may consist of any formula, pattern, device or compilation of information which is used in one's business and which gives him an opportunity to obtain an advantage over competitors who do not know or use it. It maybe a formula for a chemical compound, a process for manufacturing, treating or preserving materials, a pattern for a machine or other device, or a list of customers.[225]

The UTSA sets out a simpler definition of trade secret than that contained in the *Restatement*. According to the UTSA, a trade secret may be any kind of information, including but not limited to, "a formula, pattern, compilation, program, device, method, technique or process. , .' A trade secret must derive independent economic value, actual or potential, from the fact that it is *not generally known to the public* or competitors. The UTSA does not require absolute secrecy; the information may derive actual or potential economic value from relative secrecy. However, the information must be ''the subject of efforts that are reasonable under the circumstances to maintain its secrecy. ' Thus, even if information is not leaked, protection may be lost if it is not treated as secret.

The law of trade secret generally focuses on inequitable use of the information, whether by abuse or breach of confidence or trust, impropriety in obtaining or using the information, or breach of contract or other express obligation not to disclose the information. The indicia for establishing a claim of misappropriation of a trade secret or confidential information set forth by the *Restatement of Torts are* that:

1. There must be a *protectable interest i.e.* a trade secret;
2. The plaintiff must have *a proprietary interest* in the trade secret or information;
3. The trade secret must be *disclosed* to the defendant *in confidence* or it must be *wrongfully acquired* by the defendant through improper means;
4. There must be a *duty not to use or disclose* the information; and
5. There must be a *likely or past disclosure or use* of the information, if in a different form, which is unfair or inequitable to the plaintiff.

Under the UTSA, two major types of trade secret misappropriation are prohibited:

1. Simple acquisition of trade secrets, regardless of whether the information is used, by a person who knows or has reason to know that the trade secret was acquired by improper means.
2. Misappropriation based on use or disclosure:
   - use of improper means to obtain a trade secret;
   - use or disclosure of a trade secret knowing or having reason to know that the secret was improperly obtained;
   - use of information acquired by accident or mistake after learning, and before materially changing position, that the information is a trade secret.

Characteristics of a "Trade Secret"

In order to qualify as a trade secret, information must possess certain characteristics. First, the information that is the subject of trade secret protection must be of some minimal *competitive value or advantage* to the owner or his business. Trade secrets can include technical information, customer lists, suppliers, or accumulated business wisdom. The information must also be the result of some minimal *investment or expense,* and must not be generally known to the public. Courts also consider the amount of effort that is invested in creating a

---

223 For example, even if people own certain cards on which information is printed, they do not necessarily have a right to use the confidential information on those cards. *American Republic Ins. Co, v. Union Fidelity Life Ins. Co., 470* F.2d 820, 825 n. 3 (9th Cir. 1972).

224 Ronald Abramson, ''Trade Secret Protection for Computer Software-Proeedures for Protection: Recent Decision On Its Scope, ' *Computer Software 1990: Protection and Marketing (New* York, NY: Practicing Law Institute, July 1990) p. 479.

225 *Restatement of Torts*, section 757, comment b, at 5 (1939).

Box 2-H—Analogous Copyright Law in Foreign Countries

The Pacific Rim

Japan—To ensure inclusion of computer programs as protectable subject matter of copyright, Japanese revised copyright law defines computer programs as "a set of instructions for a computer which are combined in order to function [sic] the computer so that one result can be obtained."[1] This definition establishes four requirements for a copyrightable computer program. First, the set of instruction must function to operate a computer. Second, the minimum requirement for a copyrightable program is a set of instructions which can effect one result.[2] Third, a program consists of a set of instructions to be input into a computer.[3] Fourth, a program must be an expression; to be the subject matter of copyright law, a program must be the expression of an idea.

Under Japanese law, both source code and object code are copyrightable. Translation from source code to object code constitutes a reproduction of the source code.[4] A report prepared by the Japan Copyright Committee in January 1984 defines a set of instructions stored in read only memory to operate a preselected operation as a microprogram, making them subject to Japanese copyright. This definition is not widely accepted, leaving some doubt as to the copyrightability of microprograms.

Japanese Copyright Law further provides that the author shall have the exclusive right to reproduce his work. The law provides that the author shall have the exclusive right to translate, arrange, transform, dramatize, cinematize, or otherwise adapt his work. In principal, a person who possesses a copy of a program is prohibited from making another copy or adapting the original copy without the copyright owner's consent. However, like U.S. law, Japanese Copyright Law limits the scope of the author's exclusive right of reproduction regarding a program work, by allowing copies or adaptation to the extent deemed necessary for the purpose of using the work in a computer to be made by the owner of a program for his own use.[5]

The period of protection for computer software in Japan is life of the creator plus 50 years.[6] For unpublished software, the copyright endures 50 years after the creation of the work.[7]

Korea—In Korea computer programs are defined as a set of instructions and commands, expressed in a specific manner, to be used directly or indirectly in a computer to bring about a certain result. The scope of protection afforded software by Korean law is similar to that granted to software in Japan.

The term of protection for software in Korea is 50 years from the time at which the program is created.

Taiwan—Taiwan law defines a computer program as a set of instructions to be used in a computer in order to bring about a certain result. The Ministry of the Interior published a document in 1986 clarifying the coverage of software by the copyright law. Prior to release of that document, software registrations were accepted but no explicit provision provided for software.[8]

Taiwanese copyright law provides protection for the life of the author plus 30 years. If the work is created by an employee, 30 years of protection are provided.[9]

---

1 Paul C.B. Liu, "Computer Software and Intellectual Property Law in the Pacific Rim Countries," contractor report prepared for the Office of Technology Assessment, p. 20, citing copyright law, art. 2, para. 1, item 10-2 (Law No. 48, 1970), translated by EHS (Eibun-Horei Sha Inc.), Law Bulletin Series, Japan Vol. 3. The definition has also been translated as: "an expression of combined instructions given to a computer so as to make it function and obtain a certain result," Atsuo Torii, Legal Protection of Computer Software in Japan, *AIPPI Journal*, December 1985, p. 150.

2 "One result" is the final result of one data processing function. For example, if a whole program includes a set of subroutines, each subroutine is protectable as a copyrightable program.

3 Thus, material written in language other than machine-readable language, such as a flow chart, is not a computer program under Japanese copyright law.

4 Liu, op. cit. footnote 1, citing 14 (3) Mutaizaisanken Kankei Minji Gyosei Saiban Reishu 7% (Tokyo District Ct., Dec. 6, 1982).

5 Japanese Copyright Law, art. 47(b).

6 Ibid., art. *53,* para. 1

7 Ibid., *art. 53,* para. 3.

8 R. Michael Gadbaw, et al., *Intellectual Property Rights: Global Consensus, Global Conflict?* (Boulder, *CC):* Westview press, 1988) p.374.

9 Ibid.

Western Europe

The European Communities have adopted a directive on the legal protection for computer software, which must be implemented by each of the EC member states. This directive requires that software be protected by copyright as a literary work within the meaning of the Berne Convention.[10]

Each of the member states must adopt legislation necessary to comply with the directive by January 1, 1993. However the following are examples of copyright protection for software as it now exists in the EC member states of France and in Switzerland, which is not a signatory to the EC treaty.

France—The French legal system, based on statutory law rather than case law, did not address the question of copyright in its Civil Code Law of March 11, 1957. The law of July 3, 1985 modifies the copyright law to take into account modem technologies such as computers. The July 3, 1985 law expressly provides that software is copyrightable, but leaves to the courts the task of defining software (referred to in France as ''logiciel' '). The law of July 3, 1985 differs in several respects from the normal rules of copyright. A decree of December 22, 1981 proposes a definition of software as:

> the combination of programs, processes and rules, and possibly the documentation, relevant to the functioning of a system for the treatment of information.

This text is not strictly regarded by the courts.

Copyright protects not the idea of a creation but the form or expression of the creation. French law has no requirements for registration, deposit or notice. While there is no value requirement for protection, the creation must be original; i.e., it must reflect the personality of the creator.

The owner of a copyright has rights of reproduction and adaptation. While the buyer of software is entitled to make one backup copy of software, any other unauthorized copy or unauthorized utilization is an infringement.

The term of protection under copyright in France is 25 years from the date of creation of the software.

Switzerland--Opinion as to whether computer programs are copyrightable in Switzerland is mixed. However, generally, in order to be protected under Swiss law, a work: 1) must be a creation, 2) must be in the literary or artistic field, and 3) must have an original character, Swiss law lists explicitly the exclusive and absolute rights of copyright holders. Among these are the right to reproduce a protected work, and the right to sell, offer for sale, or put into circulation copies of the work Protection extends for the life of the author plus 50 years. Switzerland has no formality requirements.

Latin America

Brazil-Copyright law in Brazil pertains to all "creations of the mind. . . regardless of their form of expression. [11] Software programs are not included in an enumerated list of creations subject to protection, and are not registered by the National Copyright Council of the Ministry of Culture.

A work is protected by copyright in Brazil for the life of the author plus 60 years after the author's death. Registration is not required for copyright protection.

Argentina--Argentine law protects all traditional forms of creative expression. Source code programs may have copyright protection. While the Argentine National Copyright Registry allows for registration of source code programs and object code programs, the courts have not made a specific ruling on this practice. Draft laws grant protection to both object code programs and source code programs, as well as to the operating system software and application software.[12]

Mexico-Mexican law now includes computer programs as a category of protected works under the copyright law. Mexican law includes no private use or "fair use" type of limitation.

---

10 See Couch Directive of May 14, 1991 on the legal protection of computer programs (91/250/EEC), preamble, para. 30; published in the Official Journal of the European Communities, No. L 122/42, May 17, 1991. For further discussion on the European Economic Community's treatment of computer software in the software directive, see ch. 3.

11 Gadbaw, et al. op. cit., footnote 8, p. 172.

12 Gadbaw, ibid, p. 133; Cary Sherman, et. al., *Computer Software Protection Law* (Washington DC: Bureau of National Affairs, Inc., 1990), p. AR 8-9.

SOURCE: OTA, 1992.

program when determining whether a trade secret exists.[226] Finally, the owner of a trade secret must affirmatively *maintain the* secret.

The Need to Maintain the Trade Secret

A critical requirement for establishing the existence of a protectable trade secret is that the proprietor has exercised a substantial effort to retain secrecy that was reasonable under the given circumstances. While the trade secret owner must take affirmtive, concerted and continuing action to protect his trade secret, the requisite level of secrecy is characterized by relative secrecy:

> ''The owner of the secret need only take reasonable precautions to ensure that it would be difficult for others to discover the secret without using improper means. '[227]

There are several widely recognized indicia as to whether the information is entitled to protection as a trade secret:

1. the extent to which the information is known outside the company;
2. the extent to which it is known by employees and others involved in the business;
3. the extent to which the owner has gone to assure its secrecy;
4. the value of the information to the owner and his competitors;
5. the cost of developing the information; and
6. the ease with which the information could be properly duplicated or acquired by others.[228]

General disclosure of protected information will entail permanent loss of its character as a secret. However, the ''necessary element of secrecy is not lost if the holder of the trade secret reveals the trade secret to another in confidence, and under an implied obligation not to use or disclose it,' or under a similar express obligation.[229] Thus, licensing of software or its disclosure to an employee will not void the secrecy of the information embodied in it, provided that the recipient is subject to an implied or express obligation to maintain confidentiality. While one court has held that secrecy remains when software is distributed in object code only,[230] the question remains how wide a distribution vitiates requisite secrecy.

Employment contracts through which employees are placed under an obligation to maintain secrecy are governed by State law and are an important tool in implementing trade secret protection. It is suggested that the prevalence of such contracts is one argument against a Federal trade secret law, since to attempt to regulate such contracts on the federal level would impinge upon the power of the States to govern employer/employee relations.

Software and Trade Secret Law

Trade secret law is one of the most widely used forms of legal protection for intellectual property interests in computer software. Numerous courts of a variety of U.S. jurisdictions have ruled that trade secret properly protects computer software.[231]

When software is distributed to relatively few customers, licenses establishing the confidential relationship and obligations necessary for trade secret can be obtained through signed written agreements.[232] *Developers* of computer software have attempted to address the more difficult problem of maintaining trade secrecy in mass marketed

---

226 *Cybertek Computer Products, Inc.* v. *Whitfield*, 203 U. S.P.Q. 1020, 1023 (Cal. Super. Ct. 1977) *Computer Print Systems, Inc. v. Lewis, 422* A.2d 148, 153-54 & n. 2 (Pa. Super. 1980),

227 Henry *Hope* x-Ray *Prods., Inc.* v. *Marron Carrell, Inc.*, 674 F.2d 1336, 1340 (9th Cir. 1982), *Jostens, Inc., v. National Computer Sys., Inc.* 318 N.W. 2d 691,700 (Minn. 1982). Similarly, the UTSA provides that the information as discussed earlier in this section, must be "the subject of efforts that are reasonable under the circumstances to maintain its secrecy." Information must be consistently treated as a secret, as otherwise it may lose its trade secret status even if it does not leak out.

228 Restatement *of Torts*, section 757, *Cement* b, at 6 (1939), reprinted in app. D.; *SI Handling Systems* v. *Heisley*, 753 F.2d 1244 (3rd Cir. 1985).

229 *Kewanee Oil Co.* v. *Bicron Corp.*, *416* U.S. 470, 475 (1974).

230 see, e.g., *Q-Co Industries Inc.* V. *Hoffman*, 625 F. Supp. 698, 617-18 (S.D.N.Y. 1985).

231 *Telex Corp.* v. *Inter'l* Business Machines *Corp.*, 510 F.2d 894 (10th Cir.), *cert. dismissed*, 423 U.S. 802 (1975); *University Computing CO.* v. *Lykes-Youngstown Corp., 504* F.2d 518 (5th Cir. 1974); Q-Co. *Industries, Inc. v. Hoffman, 625* F. Supp. 608 (S. D.N.Y. 1985); *Cybertek Computer Products, Inc. v. Whitfield, 203* U. S.P.Q. 1020 (Cat. Super. Ct. 1977); *Corn-Share, Inc. v. Computer Complex, Inc., 338* F. Supp. 1229 (E.D. Mich. 1971), *aff'd.*, 458 F.2d 1341 (6th Cir. 1972).

232 Considerable scholarship exists discussing the use of contract above and beyond the protection provided by the traditional intellectual property laws. Such modes of providing protection for software and software related inventions are of importance, given the European Community's Directive on legal protection for computer software, which in article 9 specifically provides that contractual provisions contrary to article 6 or to the exceptions provided for in article 5(2) and (3) are nullified by article 9. For further discussion of contracts in this area, see box 2-I.

Box 2-I-Contract Law Protection of Electronic and Computer Technology

Apart from the provisions of the intellectual property law, contractual agreements are used to provide the terms for the distribution of computer goods and services. These agreements have allowed the vendor or licenser to define its relationship with the user or licensee concerning the goods and services. Traditionally, a considerable amount of service and maintenance was required in connection with computer goods. Contracts covered this aspect of the relationship and defined certain obligations which existed on the part of the vendor/licensee and the user/licensee, including warranties, limitation on liability, risk of loss and damages.

As computer technology evolved, both in terms of its cost-effectiveness and the extent of its distribution, the nature and breadth of contractual agreements also developed. In the area of mass-marketed software for microcomputers, the industry adapted the contractual relationship and developed the so-called shrink-wrap license agreement. The shrink-wrap license does not necessarily require that the user/licensee formally execute the agreement and return it to the vendor/licensee. Instead the contract may become binding upon use of the licensed program by the user/licensee.[1]

According to some sources, the distinction between hardware, software, and data is beginning to blur significantly at the same time that more and more resources are being invested in their development. Typically, the relationships between the buyers and sellers regarding their rights in these are set forth between parties in written agreements.[2]

The Interface Between State Contract Law and Federal Patent Law

While Federal patent law preempts State-based protection that provides patentlike protection,[3] Federal patent law does not preempt State-based protection of trade secrets[4] and does not prohibit States from enforcing valid contracts that provide protection for unpatented products.[5] In the case of *Aronson v. Quick Point Pencil Co. the* Supreme Court stated:

> Commercial agreements traditionally are the domain of state law. State law is not displaced merely because the contract relates to intellectual property which may or may not be patentable, the states are free to regulate the use of such intellectual property in any manner not inconsistent with federal law. *[citations omitted In this as in* other fields, the question of whether federal law preempts state law ''involves a consideration of whether that law 'stands as an obstacle to the accomplishment and execution of the full purposes and objectives of Congress. ' *[citations omitted]. "* If it does not, state law governs.[6]

The Court of Appeals for the Federal Circuit has adopted the principles expressed by the Supreme Court in *Aronson in* a number of decisions.

In *Universal Gym Equipment v. ERWA Exercise Equipment,[7] the* Federal Circuit held that parties to a contract may limit their rights to take action they previously had been free to take. For example, a licensee or licenser may contract to prohibit the licensee from reverse engineering or manufacturing any features of a licenser's product-even after the agreement is terminated. Before contracting, the licensee may have reverse engineered and manufactured the licenser's unpatented products without violating the licensor's rights.[8]

In *PowerLift, Inc. v. Watherford Nipple-Up Systems, Inc.,[9] the Federal Circuit, in* deciding a *preemption issue,* stated:

---

1 The shrink-wrap license is discussed in further detail in this chapter, and in box 2-J.

2 Robert Greene Sterne, Sterne, Kessler, Goldstein and Fox, personal communication, Nov. 20, 1991.

3 Indeed, not all forms of State-based protection are preempted by Federal patent law. For example, *Bonito Boats* did not prohibit States from protecting trade *dress . Bonito Boars, Inc. v. Thunder Craft, Inc., 489* U.S. 141 (1989); 376 U.S. 225 (1964); and *Compco Corp. v. Day-Brite Lighting, Inc., 376* U.S. *234 (1964).*

4 *Keewanee Oil Co. v. Bicron Corp., 416* U.S. 470 (1974).

5 *Aronson v. Quick Point Pencil CO.,* 440 U.S. 257 (1979).

6 *Aronson v. Quick Point Pencil Co., 440 Aronson v. Quick Point Pencil CO.,* U.S. 257 (1979) at 301-302.

7 *Universal Gym Equipment* v. *ERWA Exercise Equipment, 827* F.2d 1542 (Fed. Cir. 1987).

8 *Universal* Gym *Equipment v. ERWA Exercise Equipment, 827* F.2d 1542 at 1550. See also *Richardson* v. *Suzuki Motor Co.,* 868 F.2d 1226, 1241-42 (Fed. Cir. 1989), which held that where parties contract to limit the use by the recipient of features, designs, technical information or know-how disclosed under the contract, such a contractual ararngement is not incompatible with the patent law.

9 *Power Lift, Inc. v. Watherford Nipple-Up System, Inc., 871* F.2d 1082 (Fed. Cir. 1989)

Continued on next page

Box 2-I--Contract Law Protection of Electronic and Computer Technology--Continued

> *[w]e* follow the Supreme Court guidance in *Aronson* which binds all circuits addressing federal preemption of state contract law.[10]

The Interface Between State Contract Law and Federal Copyright Law

The Copyright Act preempts private contracted arrangements between parties that affect their rights in copyrighted works, only in a few limited circumstances specified in the statute. The legislative history of section 301 of the act states:

> nothing in the bill derogates from the rights of parties to contract with each other and to sue for breaches of contract. ..11

In the instances in which Congress intended in the Copyright Act to create exceptions to individuals' freedom to contract, it did so unequivocally. Section 203 grants authors the right to terminate transfers or licenses to their works after 35 years, and section 203(a)(5) explicitly limits an author's ability to contract that termination right away, providing that:

> [termination of the grant maybe effected notwithstanding any agreement to the contrary. . .

The legislative history is clear on the effect of the provision:

> [A]lthough affirmative action is needed to effect a termination, the right to take this action cannot be waived in advance or contracted away.[12]

Contracts that impose restrictions beyond the immediate scope of the rights granted under section 106 of the Copyright Act have been upheld. For example, in *SAS Institute, Inc. v. S & H Computer Systems, lnc.*[13] *the court* found the defendant liable for violating specific terms of a license agreement, among them a prohibition against using the computer program on an unauthorized central processing unit. In addition, agreements that relate to subject matter that is not protected under any form of intellectual property protection are not uncommon.

Providers of copyrightable and uncopyrightable databases rely on contracts to protect against unauthorized reproduction and use of their data.[14] Particularly since the Supreme Court's decision in *Feist Publications, Inc. v. Rural Telephone Service Co.,*[15] contractual protection is considered by some to be essential to the development and marketing of databases.[16]

---

10 *Power Lift, Inc. v. Watherford Nipple-Up* Systems, *Inc.*, 871 F.2d 1082, at 1085, footnote 5.

11 H.R. Rept. No. 94-1476 at 132 (1976).

12 H.R. Rept. No. 94-1476 at 125 (1976).

13 *SAS Institute, Inc.* v.s & *H Computer Systems, Inc., 605* F.Supp. *816 (M.D.* Tenn. 1985)

14 In the case of *Telerate Sys* rem, *Inc.* v. *Caro* (689 F. Supp. 221 (S.D.N.Y. 1988)) on a motion for preliminary injunction the court found that Telerate, a financial database provider, was likely to succeed on its claim for tortious interference with contract. The claim was brought against the developer of a software package that allowed speedup and manipulation of Telerate data but which required that foreign equipment be interfaced with Telerate equipment in violation of Telerate's contract with its subscribers.

15 *Feist* Publications, *Inc.* v. *Rural Telephone Service CO.* 111 S.Ct. 1282 (1991).

16 The right t. make agreements respecting a copyrighted work may be limited by a judicial doctrine of copyright misuse. The CO@ in *Lasercomb America, Inc. v. Reynolds, 911* F.2d 970 (4th Cir. 1990) refused to enforce a copyright where plaintiff employed a license agreement with some of its licensees other than the defendant which prohibited the other licensees from independently creating a competing program. However, *in Atari Games Corp. v. Nintendo of America, Inc. (Nos. C-884805* -FMS, C-89-0027-FMS, slip opinion (N.D. Cal. Mar. 5, 1991)) the court rejected a misuse defense and distinguished *Lasercomb v. Reynolds.* For further discussion of the defense of copyright misuse, see 3 *Nimmer on Copyright* section 13.09, II; P. Goldstein, *Copyright: Principles, Law & Practice* section 9.6 (1989 & 1991 Supp.).

SOURCE: OTA, 1992.

software, extensive distribution of which might otherwise destroy requisite secrecy, by use of what is known as a ''shrink wrap"' license. (See box 2-J.) Theoretically, such a license is used in conjunction with the practice of publishing program code in object code form. Object code is understandable to people only after extensive effort, and would ordinarily require intermediate steps to recover a higher level language representation of the program (see ch. 4). Distributing the code in such a form is intended to maintain the secret nature of the information. In addition, it invokes the provisions of the copyright law, since recovering a high-level language version may involve the making of a copy or derivative work

Box 2-J—The Shrink-Wrap License Controversy

Considerable controversy surrounds the practice of using shrink-wrap licenses to maintain trade secret status for mass-distributed software. Questions arise both from the perspective of the trade secret and the copyright law.

The decision in *Bonito Boats* would indicate that for products sold on the mass market there is a Federal policy that favors allowing the buyers to examine a distributed product and use any unpatented and noncopyrightable aspects of that product. But it is argued that copyright protection limits this buyer right by preventing reverse engineering.

From the perspective of the trade secret law (which requires reasonable efforts to maintain secrecy for trade secret protection to exist), some observers believe that the degree of difference in the level of secrecy maintained in limited distribution products and in mass-marketed products is so great that a shrink-wrap license alone does not qualify as the requisite effort to establish trade secret status. These observers also question whether any form contract can establish the kind of contract or confidential relationship that would enable the publisher to sue the end user who simply reverse engineers the work.[1]

From the perspective of the copyright law, some hold that copyright does not prevent reverse engineering of software. They believe that fair use may well permit the copying of an entire work where the sole purpose of the copying is that it is a necessary part of making the work perceptible to humans so that they can perceive and use the underlying ideas which are not protected by copyright. Further, there is concern that a shrink-wrap license allows a licenser to extend his or her right beyond the underlying property right on which the license is based; i.e. the license is based on copyright and yet attempts to prevent the licensee from copying the noncopyrightable *expression in the* work.

In keeping with the requirement that effort be made to maintain secrecy for trade secret protection to exist, these observers believe that a trade secret is only enforceable to the extent that mass marketing can be determined to be a reasonable action for someone attempting to keep the idea secret, Some would argue that any trade secret status would be lost when a program was put on the market, because the secret could be obtained through recompilation. The copyright law's prohibition on copying, they assert, would not prevent a court from finding that information related to computer program, e.g., how a particular step in the program is configured or the sequence of steps used to obtain a particular result, was readily accessible because the program in question was mass marketed without copy protection. A shrink-wrap license, they believe, might be enforceable as a contract, but likely would not be found adequate to show the proprietor of the secret information in a mass-marketed computer program had taken reasonable steps under the circumstances to maintain the secrecy of the information.[2]

1 Raymond Nimmer, Professor of Law, University of Houston Law Center, personal communication, Aug. 9, 1991.

2 Mary Jensen, University of South Dakota School of Law, personal communication, Sept. 29, 1991.

SOURCE: OTA. 1992.

of the object code program. The making of such a copy or derivative work is believed by some to be a violation of the copyright law, and is not given explicit treatment under Section 117 of the Copyright Act, or the doctrine of fair use.[233]

The shrink-wrap license further signals secrecy, and is established by marketing software in a sealed package with a notice and a license agreement that is visible on the exterior of the package. The agreement generally provides that the user, by opening the package, is deemed to have accepted the license terms and conditions. The terms of such a license generally prohibit recompilation, disassembly or copying of a program for any reason except for use and backup purposes, so that copying of the object code in the process of obtaining a higher level language version of the program arguably becomes, in addition to a violation of the copyright law, a breach of the shrink-wrap agreement, assuming that the agreement is enforceable. Some shrink-wrap agreements contain an express prohibition on ''reverse engineering" or decompilation/disassembly. Occasionally a card is provided which must be signed and returned in order to receive information

233 See, e.g., *SAS Institute, Inc.* v. S&H *Computer Systems, Inc.*, 605 F. Supp. 816, 828.831 (M.D. Tenn. 1985); *Hubco Data Prods. Corp. v. Management Assistance Inc. 219* U, S.P.Q. (BNA) 450, 455-56 (D. Idaho 1983). But compare *Vault Corp. v. Quaid Software Ltd , 847* F.2d 255 (5th Cir. 1988). For further discussion of the technical aspects of decompilation, See ch. 4.

on product updates or extended warranty, thus securing the user's consent to the terms of the license. The more traditional shrink wrap procedure allows the user to either agree to the contract terms or to return the product.

The enforceability of shrink wrap licenses remains in question, and has not been tested specifically by any courts. In the case of *Vault Corp. v. Quaid Software Ltd.*,[234] the district court ruled on the enforceability of the shrink wrap license, stating that the contract was an unenforceable contract of adhesion under state law, so that it could only be enforced through special statute. The Fifth Circuit held that certain provisions of the shrink-wrap license at issue were unenforceable because they conflicted with the Copyright Act by attempting to avoid the provisions of the first sale doctrine and to extend the term of copyright protection indefinitely. The court did not address the issue of the enforceability of the shrink-wrap license itself. Other cases raising this issue have been filed and settled, so that they do not provide guidance on the issue.[235]

Other arguments cited as grounds for a finding that such licenses are not enforceable are that shrink-wrap licenses do not provide for proper offer and acceptance, the agreement is unconscionable, the agreement violates State consumer protection legislation, or the agreement is a sham intended to circumvent the provisions of the Copyright Act of 1976, particularly the restrictions of the first sale doctrine.[236]

Some believe that while the issue has not been addressed from the standpoint of trade secret law, it is unlikely that a court would find that a shrink-wrap license constituted reasonable efforts on the part of the trade secret proprietor, under the circumstances of mass marketing, to keep the information secret.[237] Still others assert that section 117's adaptation right permits the copying necessary to disassemble and recover higher level code than object code for purposes of fixing a bug or adapting a program to the user's specific needs or even to determine the ideas embodied in the program and not protected by copyright.[238]

Relationships Among Patent/Trade Secret/ Copyright Laws

In the Kewanee Oil case, the Supreme Court expressly ruled that State law may protect trade secrets and that trade secret law is compatible with patent law. However, the Supreme Court also ruled recently that State law may not prohibit copying of utilitarian and design ideas that the patent laws have otherwise left unprotected.[239] State law further may not provide protection akin to the patent laws and may not set down a rule in favor of legal protection where Congress has ''struck a balance' in favor of nonprotection.[240]

When a patent is granted, the patent itself, which must disclose the ''best mode' for practicing the invention, becomes a public document and the file wrapper, consisting of supporting materials on file (specifically, the prosecution history of the application), becomes available for public inspection. As a result, the trade secret status of the matter disclosed in the patent or related PTO file is destroyed. These submitted materials remain secret unless and until the patent is granted, so if the patent is not granted or is withdrawn by the applicant prior to issuance, the secret is maintained. This is the case only in the United States; in foreign countries applications are published after 18 months, whether or not the patent ultimately issues. Some believe that with harmonization under WIPO, the confidential system in the United States is likely to change.[241]

---

234 *Vault Corp.* v. *Quaid Software Ltd., 847* F.2d *255* (5th Cir. 1988).

235 MicroPro Int'l Corp. v. *United Computer Corp.*, Civ. Action No. C-893-3019 W.D.S. (N.D. Cal. filed June n', 1983; *Lotus Dev. Corp. V. Rixon (D. Mass.* filed Jan. 31, 1984).

236 In this context, see *StepSaver Data Sys. v. Wyse Tchnology*, 939 F.2d 91 (3d Cir. 1991), which holds that a' 'package license' did not become part of the contract between two merchants. In effect, there was an oral contract for sale followed by the packaged form. Applying standard Uniform Commercial Code analyses, the court stated that this does not alter the primary contract. The new terms in the form do not become part of the agreement unless the vendor makes clear through its actions that it will not proceed with the transaction unless the buyer consents to the new terms.

237 G. Lee Skillington, Alice Zalik, U.S. Patent and Trademark Office, Office of Legislation and International Affairs, personal communication, Sept. 27, 1991.

238 Mary Jensen, University of South Dakota School of Law, personal communication, Sept. 29, 1991.

239 *Bonito* Boats, *Inc.* v. Thunder *Craft Boats, Inc., 109* SUp. Ct. 971 (1989).

240 *Bonito Boats,* supra; *Sears* Roebuck& *Co.* v. *Stiffel Co.,* 376 U.S. 225 (1964); *Compco Corp. v. Day-Brite Lighting, Inc.,* 376 U.S. 234 (1964).

241 J. Jan@ American Bar Association, personat Communication Sept. 22, 1991.

Because of the ''secret' nature of the protected subject matter, inventions maintained as trade secret are not made of public record and necessarily do not function as prior art for purposes of the patent system (although earlier filed, secret, patent applications are prior art).[242] As a result, inventions maintained as a trade secret cannot, for the most part, preclude issuance of a patent on the basis of lack of novelty or obviousness unless the invention is described in an earlier filed patent application. Thus, while a patent requires complete disclosure, it allows the rightholder to exclude everyone else from practicing that invention, even if independently invented. Conversely, while trade secret law allows, and indeed requires, that an idea be held in secrecy, the fact that the trade secret cannot be used as prior art against a third party patent application precludes protection from independent invention by another, who may then obtain patent protection. Some, however, maintain that, because the requisite level of secrecy required for trade secret protection is not total, i.e., the information which constitutes a trade secret must only nor be generally available or readily ascertainable, situations exist in which a trade secret may be available as prior art.[243] In any case, once a U.S. patent is granted, it can be used as prior art from the filing date of the corresponding patent application notwithstanding the secret status of the application during prosecution.

In contrast to the mutually exclusive effect of the patent and trade secret law, and despite the arguments of some commentators that the benefits of copyright protection should not be granted except in exchange for full disclosure of a work,[244] copyright and trade secret protection are simultaneously available for computer software, The legislative history of 17 U.S.C. 301 (the preemption section) under the Copyright Act of 1976 and the software amendments of 1980 make this clear. When it enacted the Copyright Act of 1976, Congress stated that the evolving common law rights of, among others, trade secrets,

> would remain unaffected as long as the causes of action contain elements. . . that are different in kind from copyright infringement. Nothing in the bill derogates from the rights of parties to contract with each other and to sue for breaches of contract. . . however, to the extent that the unfair competition concept known as 'interference with contract relations' is merely the equivalent of copyright protection, it would be preempted.

Congress reaffirmed its position on the limited scope of preemption of ''remedies for protection of computer software under State law" when it passed the software amendments of 1980. Trade secret law provides protection for the underlying ideas, concepts, processes and algorithms (as well as the form in which expressed), while copyright law protects only the *form* of expression. Thus, some observers believe that the simultaneous protection of software by copyright and trade secret arguably affords coverage for both idea and the expression and avoids the idea-expression dichotomy of copyright law.[245] By contrast, others assert that this is seldom the case, and that few ideas, concepts, processes, and algorithms underlying mass-marketed computer programs would not be generally available or readily

242 One commentator observed that in many cases a software "invention' is not published for reasons other than maintenance of a trade secret. He cited among the common reasons why programmers do not publish the techniques they develop the following: techniques maybe circulated informally among programmers; techniques may be ''so obvious that scholarly credit is impossible;" the developer may lack the motivation and writing skill to write a paper suitable for magazine publication, and may not see the purpose in any other kind of publication; the developer may not realize that he has developed anything worth publishing; the developer may keep the source code of a program secret to conceal the overall design rather than any particular technique. Richard Stallman, The League for Programrn ing Freedom, personal communication, September 1991.

Another spoke of programming techniques not described in patents or the general technical literature that are known generally to programmers as the ''folk art. ' Letter of John L. Pickitt, President, Computer and Business Equipment Manufacturers Association, to E.R. Kazenske, Executive Assistant to the Commissioner, U.S. Patent and Trademark Office, July 15, 1991.

243 For example, the information might appear in a printed publication that is obscure because of age or because it has not been translated into a language understood by many Americans. If the publication was in the PTO's Scientific Library, however, it could be cited as prior art to show 1ack of novelty or to show obviousness of an invention claimed in a patent application. These same commentators assert that when the validity of a patent is challenged in patent litigation, trade secrets might also be considered prior art. G. Lee Skillington, Alice Zalik, U.S. Patent and Trademark Office, Office of Legislation and International Affairs, personal communication, Sept. 27, 1991,

244 See, e.g., Pamela Samuelson, ''CONTU Revistied: The Case Against Copyright Protection of Computer programs in Machine-Readable Form," 1984 *Duke Law Journal,* 663, 705-27. By contrast, the actions of Congress and the courts would indicate that that these bodies have rejected this point of view. The grant of copyright to unpublished works in the 1976 act and provisions for registering unpublished works under the 1909 acts are evidence that Congress did not intend to require full disclosure for copyright protection. Some observers comment that the limitation in the copyright law and the idea/expression dichotomy as opposed to the fuller protection granted by patent are the reasons why Congress might not have seen fit to require full discosure for more limited protection.

245 Office of Technology Assessment Workshop on June 20, 1991.

ascertainable to those in the industry. Further, as discussed above, this dual protection potentially addresses the problem of loss of trade secret protection through mass distribution of software. If the extensive marketing of" the software arguably destroys the secrecy requisite for trade secret protection, some believe that copyright law may be invoked to protect the unpublished source code of the program, with its relatively easily understandable ideas by prohibiting illegal copying of the published object code which must be copied or transformed to allow for easy understanding of the underlying idea.[246] Indeed, copyright law also provides a safety net in the event that trade secret protection is lost by accidental or public disclosure.

Foreign countries grant trade secret protection under a variety of provisions and to differing extents. While some countries provide specifically for trade secret protection, others allow for similar protection under unfair competition law and the law of contracts (see box 2-K).

246 Office of Technology Assessment Workshop on June 20, 1991. However, one observer believes that the exclusive reproduction right can never be used to preclude indirect access to nonprotectable matter excluded by section 102(b) merely because use of the unprotected matter entails an unauthorized reproduction, He believes that this was the holding of ***Baker** v. Se/den,* which, he asserts, already guaranteed "reverse engineering' if the utilitarian feature of a functional work can only be used by making a copy. Jerome Reichman, Vanderbilt University, personal communication September 1991.

Box 2-K—Analogous Trade Secret Law in Foreign Countries

The Pacific Rim

Japan—Japan is the only Pacific Rim nation whose law provides for trade secret protection. The Japanese law defines a trade secret as technological or business information useful for business activities, controlled as a secret, which is not publicly known art. Under the law, if a computer program properly qualifies as a trade secret, the owner of a computer program who is damaged or is likely to suffer damage by unauthorized use or disclosure of his program may require the offending party to stop the unauthorized use or disclosure of the program. The owner of a trade secret may request that the media on which the program is stored be destroyed (However, since there are no ''protective orders'' in court proceedings, the secret may be lost as a result of bringing the litigation.) Unfair activity includes acquisition of a trade secret by stealing, deception, or threats, or acquisition from a third party while aware that the trade secret was originally acquired by an unfair activity.[1]

Korea-The Republic of Korea has committed itself to the future adoption of a law of trade secrets. Presently there is some recognition that trade secrets should be protected from unlawful encroachment or misappropriation under tort theory. There is also some recognition of a law of trade secrets in the criminal code.[2]

Taiwan—There is no specific law protecting trade secrets in Taiwan.[3] Taiwan's criminal code enables an employer to obtain criminal sanctions against an employee in violation of a secrecy agreement made with the employer. However, there is no current provision of the law allowing one company to take action against another for misappropriation of a trade secret.[4] A district court in Taiwan determined that know-how is not an intellectual property right nor a property right recognized in the civil code, and that there is no legal obligation to keep such knowledge secret.[5]

Thai/and-Thailand has no specific law covering, nor a clear definition of, trade secrets. Protection of this kind comes from the Penal Code or Consumer Protection Law. The Penal Code covers only industrial secrets, scientific discoveries, and scientific inventions which may include industrial know-how. The Consumer Protection Act prohibits unauthorized disclosure of business secrets.[6]

Western Europe

The European Communities have agreed in their long debated Software Directive that the prescribed protection of computer programs under copyright does not prejudice the application of other forms of protection where appropriate. Thus, computer software is properly protected by trade secret in addition to copyright in European Communities member nations.[7]

---

1 Paul C.B. Liu, ''Computer Software and Intellectual Property Law in the Pacific Rim Countries, '' contractor report prepared for the Office of Technology Assessment, March 1991, pp. 37-38.

2 Ibid., pp. 38-39.

3 R. Michael Gadbaw, et al., *Intellectual Property Rights: Global Consensus, Global Conflict?* (Boulder, CO: Westview Press, 1988), p. 374.

4 Ibid., pp. 374-375.

5 Liu, op. cit., foomote 1, p. 39 (footnote 16).

6 Ibid.

7 Council Directive of May 14, 1991 on the legal protection of computer programs (91/250/EEC), preamble, para. 30; published in the Official Journal of the European Communities, No. L 122/42, May 17, 1991.

Continued on next page

Box 2-K—Analogous Trade Secret Law in Foreign Countries-Continued

France—French law presents two independent aspects of trade secret protection. In the first instance, divulgence of a trade secret is a criminal offense under the Penal Code. The circumstances giving rise to such a criminal offense are narrowly drawn, making prosecution under this provision difficult. Unauthorized disclosure of a trade secret may also be actionable under the law of tort or contract. For the most part, however, in the absence of contractual agreement, divulgence of a trade secret will not give rise to liability. Under French law an invention not protected by copyright or patent is considered an idea only, which is believed to be in the public domain. An action brought in tort must rely on theories of unfair competition, which requires proof of misconduct beyond theft of ideas.[8]

Switzerland-Trade secret protection in Switzerland is derived from three major sources. The Swiss Penal Code provides that a person who discloses an industrial or commercial secret which he is under a duty to keep secret, and the person who benefits from the disclosure, will be punished by fine or imprisonment. The Swiss Code of Obligations provides that employees and agents are bound to secrecy with respect to confidential business information obtained in the course of the contractual relationship. The 1986 Federal Act makes it an offense for a competitor to entice employees or agents to discover a secret, and another offense to use or disclose trade secrets which have been improperly revealed. In addition to these statutory provisions, a body of case law exists in Switzerland in which a trade secret holder is protected on the basis of contract law and by the fact that the other party is bound by a nondisclosure agreement.[9]

Latin America

Brazil—Brazil has no specific law of trade secret protection. Case law and Brazilian scholarship have concluded that three identifiable elements of trade secret law exist. First, a trade secret must give its owner a competitive advantage and must have commercial value. Second, a trade secret involves an element of innovation. Third, parties must sign contracts in which the confidentiality requirement is set forth.[10] Brazilian courts have dismissed the majority of trade secret cases brought under this law, usually because the original trade secret holder was found not to have taken proper measures to protect his trade secret. Thus, scholars have maintained that article 178 of the Code of Industrial Property provides protection for trade secrets. Under this provision a company may sue an employee for disclosure of trade secrets and may take action against a third party for acquiring secrets by unfair means.[11]

Argentina-There is no Argentine law directed specifically toward protection of trade secrets. Disputes about unauthorized divulgence of confidential information are addressed by enforcement of secrecy agreements between employers and employees.[12]

Mexico-Mexican law protects generally industrial secrets, or industrial application information kept confidentially by an individual or corporation when sufficient measures or systems have been adopted to preserve the secrecy and restricted access. No specific provisions are made for trade secrets in computer software.

---

8 Cary Sherman, et al., *Computer Software Protection Law* (Washington DC: Bureau of National Affairs, Inc., 1990), p. ~-16.

9 Ibid., p. SW-16

10 Ibid., p. BR-17

11 Gadbaw, et al., op. cit., footnote 3, p. 141 (footnote 17).

12 Ibid., p. 141.

SOURCE: OTA, 1992.

Chapter 3

The International Arena

Contents

Page

The Global Software Industry 93
What Do We Mean by the “Software Industry’’?. 93
Global Markets, Global Technology 94
The Issue of Piracy 97
Estimates of Financial Losses From Piracy 98
Examples of Retail Piracy and Counterfeiting 99
End-User Piracy in Businesses 100
Examples of Piracy Overseas 101
Piracy and Trade Issues 103
The Berne Convention 104
The General Agreement on Tariffs and Trade 107
The U.S. Trade Representative 109
Trade-Related, spects of Intellectual Property Rights and the U.S. Proposal to the GATT 110
Draft Final Act Embodying the Results of the Uruguay Round and Trade-Related Intellectual Property 114
Participation by the United States in Other International Treaties 114
The European Economic Community’s Directive on Legal protection for Computer Software 115
The Council Directive 115

Boxes

Box *Page*

3-A. A Snapshot of the Domestic Software Market in the Late 1980s 95
3-B. The Omnibus Trade Act and ’’Special 301’’ 104
3-C. History of the Uruguay Round.... 108
3-D. Role of the Congress inTrade Negotiations and the Fast-Track Implementation Procedure 110
3-E. The Green Paper on Copyright and the Challenge of Technology: Copyright Issues Requiring Immediate Action 116

Tables

Table *Page*

3-1.Top 10 Software Companies in North American Market 94
3-2.Top 10 Personal-Computer Software Companies in 1990 94
3-3. Top 10 Packaged Software Vendors in Europe (1989) 97
3-4. Comparison of Software Sales and BSA EstimatesofPC-Software Piracy in Selected European Countries 102

Chapter 3
The International Arena

The international nature of the software industry and market is mirrored in the global significance of national intellectual property laws and international treaties and agreements. This ''globalization' [1] of the law reflects the reality that the laws of a country are affected by, and in turn, affect, the laws of other countries, Influenced by politics, trade agreements, and the reality that similarly trained professionals, the same companies, and the same technology issues exist throughout the world, there is an increasing tendency for countries to make at least somewhat similar policy choices. This chapter examines the nature of the global software industry and the issue of piracy, multilateral and bilateral negotiations and treaties entered into to provide protection for intellectual property rights, ongoing efforts at harmonization[2] of' international intellectual property law. and the United States' participation in those negotiations and efforts. [3]

The Global Software Industry

only 15 years ago, computers and software were not mass-marketed. retail items. The mainframe and minicomputers of the day were few in number, compared to the number of microcomputers (personal computers) in use today, Those machines were operated by expert staff using expensive, often custom-developed (almost certainly customized) software; some relatively sophisticated users (e.g., in universities, large corporations and research organizations) developed and maintained their own programs. An independent software vendor community had begun to develop in the mid- 1950s;[4] nevertheless, much of the application software for specific tasks like inventory control, payroll, or ' 'number crunching' was provided by hardware manufacturers, or custom-developed under contract. Where software packages were available, they almost always required custom-tailoring to meet users' needs and operating requirements. Although there were some independent software vendors in the systems-software marketplace,[5] almost all operating-system software to run the computer and control its input, output, and logic functions was provided by computer-hardware manufacturers. [6] In the late 1960s, this changed as the ''independent" software industry began to flourish. By 1990, there were thousands of independent software developers of various sizes in the United States. comprising a $35-billion industry.[7]

What ***Do We Mean by the*** *"Software Industry*

Accurate data on software industry revenues and market shares are difficult to compile. Indeed, there are many types of ''software industry data being

---

[1] For further discussion of the concept of global ization, see Raymond T Nimmer, ' 'Globalization of Law: Commercial and Intellectual Property Markets '' Paper delivered at the Law and Society Conference, Amsterdam, June 1991, to be published as ''Global17~tion of Law: The Lessons of Software and Intellectual Property Law, " *Law in Context,* vol. 10, No. 2, 1992. Nimmer characterizes ''globali~~tion' as ' 'a world process m which legal concepts, approaches to defining and solving legal issues, and the development of legal policy in one country are resolved with explicit attention to the laws of other countries and m which import and export rules arc seen as important factors in commercial competition and in the regulation of that competition and of commercial transactions by law '' He further notes the striking impact of globalization in the past decade and points out the time and energy devoted to influencing the law outside of one's own country and in responding to such efforts.

[2] Ibid '' Harmonizat ion' is defined by R, Nimmer as a systematic effort to bring about some uniformity of the law. The underlying premise is that there are advantages in various areas of law that can be attained by establishing a basic symmetry in some area of national laws and an acceptance of legal principles from one country to another

[3] **Such** efforts at glob. lization raises issues of tensions between developed and developing countries in the context of the General Agreement on Tariffs and Trade negotiations on Trade Related Intellectual property and North-South tensions. Extensive discussion of these questions lies beyond the scope of this study However, for analysis of these issues, see Jerome H. Reichman, ''Intellectual Property in International Trade: Opportunites and Risks of a GATT Connection, ' *Vanderbilt Journal of Transnational Law,* vol. 22, 1989, p **747, at** pp. 751-769.

[4] Ronald Palenski, ADAPSO (The Computer Software and S ervices Industry Associat Ion), personal communication, July 10, 1991. ''Independent'" developers arc not part of a hardware manufacturer

[5] Ibid

[6] While this ' s still prevalent there i s a trend away from computer manufacturers prov id ing operating-systems so ftware in the personal computer market (e g , MS/DOS, DR/DOS) and in workstations and mainframes (e g , Unix) Ibid

[7] Input data provided by ADAPSO (performance of ''softw.arc' sector). total for ''In formation technology' products and serv ices for 1990 is $100 billion

Table 3-l—Top 10 Software Companies in North American Market

Company	Estimated revenues ($ millions) 1989	1990
IBM	$8,424	$9,952
Microsoft	821	1,323
Computer Associates	1,290	1,311
Digital	825	810
Oracle	554	702
Lotus	516	635
Unisys	875	600
D&B Software	450	539
WordPerfect	281	452
Novell	288	388

SOURCE: Revenues shown for the *Datamation 100* North American market, *Datamation, vol. 37, No.* 12, June 15, 1991, p.22.

Table 3-2—Top 10 Personal-Computer Software Companies in 1990

Company	Revenues ($ millions)
Microsoft	$953
Lotus	556
WordPerfect	281
Ashton-Tate	265
Autodesk	177
Adobe	121
Logitech	112
Software Publishing	110
Borland International	104
Aldus	88

NOTE: Network software companies not included.

SOURCE: Data compiled by *Soft-Letter* (Watertown, MA: 1991),

collected and reported by different organizations.[8] These include data about:

- *software and services,* including processing and professional services, as well as software products;
- *application and systems software,* including applications software and systems software, whether packaged or custom-developed;
- *packaged software,* including applications and systems software;
- *custom software,* professionally developed or extensively tailored to meet a customer's specific needs;
- *personal computer (PC) software,* usually sold as packaged software (although not all packaged software is for personal computers); and
- *software from ''independent' developers,* who are not part of a hardware manufacturer.

This variety of data, collected by different organizations, makes comparison and synthesis difficult.[9] *Consistency across types of data and years is usually not possible when drawing from these published figures.* Wherever possible, OTA will specify the type and source of market data (e.g., ''software, ''independent software, ''software and services and estimates. Therefore, estimates in this report for a given year may not ' 'add up' and different data sources may not be comparable, With market figures drawn from various sources, box 3-A provides a snapshot of the U.S. software market in the late 1980s. In 1987, the largest U.S. software vendor was IBM, followed by Digital Equipment Corp. (DEC), Unisys, Computer Associates International (CAI), and Lotus Development Corp. For comparison, table 3-1 shows 1989 and 1990 revenues for the largest software companies in the North American market and table 3-2 shows 1990 revenues for the largest personal computer software companies,

Global Markets, Global Technology

By almost any measure, the United States has a premier role, both as producer and consumer of software:

- According to one industry estimate, U.S. demand accounted for 52 percent of world *software* consumption in the late 1980s.[10]
- According to estimates by the U.S. Department of Commerce, global revenues from sales of *software* were more than $65 billion in 1989 and U.S. software suppliers accounted for more than 60 percent of global software sales,[11]

[8] For example, the Software Publishers Association collects data on packaged PC software; ADAPSO reports data on software and services, usually (but not always) from independent mainframe and minicomputer software houses; CBEMA reports data on the information technology industry, including office equipment, telecommunications, electronic data processing equipment, and software and services (including software produced by hardware manufacturers). Moreover, "hardware" companies also are software producers-sometimes, like IBM, the largest in the world,

[9] For instance, a firm whose products include PC applications may have at least some of its revenues included in ' 'PC-application software, ' or "packaged software' it may be included in ''software and services, and may or may not be an ' 'independent software house, But a firm whose main products are PC networking software is likely not to be included in data on ' 'PC-application software. "

[10] ADAPSO estimate in Jeff Shear, ' 'Competitive Software Industry Suits Up for Global Hardball," *Insight*, July 10, 1989, p, 38.

[11] Commerce Department estimate cited in Keeping *the U.S Computer Industry Competitive Defining the Agenda*, Computer Science and Technology Board (Washington, DC: National Academy of Sciences, 1990), pp. 3031.

Box 3-A—A Snapshot of the Domestic Software Market in the Late 1980s

According to the Computer and Business Equipment Manufacturers' Association (CBEMA), the U.S. software and services industry earned some $54 billion in domestic revenues in 1987 and about $68 billion in 1988. [1] Of these figures, revenues from software products (as opposed to processing and professional services) amounted to about 34 percent of the total in 1987 and 40 percent in 1988.[2]

According to the Association of Data Processing Service Organizations (ADAPSO), application and systems software from *independent software* houses comprised a $20.6 billion U.S. market in 1987 and a $25.1 billion market in 1988, split roughly 50-50 between application and systems revenues.[3] If the value of software developed ''in-house' by businesses and other organizations is taken into account (measured by salaries and other costs), some estimate that the total domestic U.S. software 'market' may be as much as $150 to $200 billion larger.[4]

According to the market-research firm, Input, application and systems software revenues in the United States totaled $20.6 billion in 1987. The largest U.S. software vendor *overall* **that year was** IBM, with 15 percent of the U.S. market and $3.1 billion in revenues; next were DEC ($935 million), Unisys ($585 million), Computer Associates International ($415 million), and Lotus ($396 million).[5] The leading *applications software* vendors in 1987 were IBM ($775 million in applications software), followed by Lotus ($396 million), DEC ($195 million in applications software), Dun & Bradstreet Corp. ($170 million), and Management Sciences America ($169 million).[6] The leading systems software vendors in 1987 were IBM ($2.3 billion in systems software), DEC ($740 million in systems software), Unisys ($420 million), Computer Associates International ($250 million), and Hewlett-Packard ($190 million).[7]

The largest *independent* U.S. software vendor in 1987 was Computer Associates International ($415 million); next were Lotus ($396 million), Microsoft ($240 million), Ashton-Tate ($170 million), and Management Sciences America, Inc. ($169 million).[8]

---

[1] CBEMA, *The Computer, Business Equipment, Software and Services, and Telecommunication Industry, 1960-2000* (Washington, DC: CBEMA, Industry Marketing Statistics, 1990), table 4-7, p. 100.

2 Ibid.

3 Market research data from Input reported by ADAPSO, 1989. [*OTA note:* "Independent" software houses are those that are not part of a hardware manufacturer. For example, as of 1990 Microsoft was the largest U.S. ''independent' software developer, but IBM was the largest software vendor in the world.]

4 Michael L. Dertouzos et al., *Made* in *America: Regaining the Productive Edge* (Cambridge, MA: MIT Press, 1989), p. 264 (cited in: *Keeping the U.S. Computer Industry Competitive: Defining the Agenda*, Computer Science and Technology Board (Washington, DC: National Academy of Sciences, 1991), p. 30.)

5 Input, "U.S. Software Products Market, 1988-93," Mountain View, CA, December 1988.

6 Ibid.

7 Ibid.

8 Ibid.

. According to the International Trade Commission, by 1988, U.S. *independent software* developers' revenues exceeded $25 billion, up from $20 billion in 1987;[12] about 40 percent of these revenues were from foreign sales.[13]

. According to the Business Software Alliance (BSA), in 1989 the *computer programming and software industry* comprised 1.18 percent of *GNP (gross* national product), generated $61.6 billion, and generated $12.1 billion in foreign sales by U.S. firms.[14]

. Sentry Market Research has estimated that 1990 worldwide sales of U.S. *packaged software* approach $25 billion.[15]

---

12 ADAPSO figures on industry performance, 1989. These data for 'noncaptive" firms excludes the value of software produced in-house by hardware manufacturers; revenues are split about evenly between application and operating-system software.

For comparison, CBEMA estimates of North American software and services revenues were about $63 billion in 1988 and $56 billion in 1987. (' 'Information Technology Industry Global Market Analysis," CBEMA, 1989, table 4-22.)

13 U.S. International Trade Commission, ''The Effects of Greater Economic Integration Within the European Community on the United States,' July 1989, pp. 4-39. [*OTA note:* "Independent" software houses are those that arc not part of a hardware manufacturer.]

14 BSA press release, Oct. 31, 1990.

15 *Software Magazine Executive Letter,* vol. 7, No. 2, March/April 1990, p. 2.

. According to the Software Publishers' Association (SPA), North American revenues from *packaged software for microcomputers* (personal computers) were $4.5 billion in 1990, up 22 percent from 1989.[16]

Taking a different tack and looking at the market for *software and services:*[17]

- The Computer and Business Equipment Manufacturers Association (CBEMA) estimates that North American demand has accounted for a substantial and steady share of world consumption—about 50 percent of software and services between 1973 and 1988. During this period, the world market for software and services as estimated by CBEMA grew from $4.7 billion in 1973 to $63.1 billion in 1988 (a compound growth rate of almost 19 percent) .18
- According to International Data Corp. (IDC) estimates, the worldwide market for software and services was about $110 billion at the end of the 1980s, with 57 percent ($63 billion) held by U.S. companies. The next largest share, according to IDC, was held by Japan ($14 billion, 13 percent), followed by France ($9 billion, 8 percent), Germany ($8 billion, 7 percent) and Britain ($7 billion, 6 percent) .19
- According to CBEMA, the U.S. software and services industry had domestic revenues of some $93 billion in 1990, about a 16 percent increase from 1989 revenues of about $80 billion. Of these domestic revenues, CBEMA estimates that software products accounted for about 45 percent of the total--$42.5 billion in 1990 and $35 billion in 1989.[20]

Although its share of the world software market has declined over the past decade or so, the United States is still the world's leading innovator and producer of computer software.[21] U.S. producers are increasingly challenged by competition from developing software industries abroad, particularly in Europe. Europe has been a very important market for U.S. firms, which dominated their European rivals. In the late 1980s, U.S. software producers held almost half of the European software market, with IBM being the largest single software vendor in the European market.[22] (See table 3-3.) In 1990, according to SPA, U.S. companies had more than 70 percent of the European PC-software market.[23]

Growth in the U.S. software market had slowed from the 50 percent per year (or better) rates of the early 1980s to about 15 percent per year by 1989, according to the SPA. But markets in Europe and Japan are booming. Industrywide, international sales account for some 34 percent of software publishers' revenues, according to the SPA. But many companies (like Microsoft) report international sales closer to half their overall revenues. U.S. software is so pervasive, in part, because of the head start the U.S. industry enjoyed and the large size of the domestic U.S. market.[24] Our large domestic market has given the U.S. industry significant advantages: a nation's domestic software market is an important base for developing the expertise and experience that are necessary to compete successfully (through exports)

16 Ken Wasch, Nicole Field, and Sara Brown, SPA, personal communication, July 30, 1991.

17 *OTA note:* Revenue reported for 'software and services'' includes revenues from processing and professional services, as well as from custom and packaged software products.

18 ''Information Technology Industry Global Market Analysis, ' CBEMA, 1989, table 4-22.

19 IDC data reported in Richard Brandt et al., "Can the U.S. Stay Ahead in Software?" Business *Week, Mar.* 11, 1991, pp. 98-99.

20 Oliver Smoot, CBEMA, personal communication, June 30, 1991. See also CBEMA, The *Computer,* Business *Equipment, Software and Services, and Telecommunications Industry, 1960-2000* (Washington, DC: CBEMA, Industry Marketing Statistics, 1990), p. 100. (Estimates from BDA Assoc. forecast.)

21 By 1993, the United States is still expected to hold about half of the world software market. (Robert Schware, ''Sofwme Industry Entry Strategies for Developing Countries,' *World Development Journal,* vol. 20, No. 2, February 1992, p. 3.) Studies in the late 1980s reported that U.S. producers held a 70 percent share of the global market for software, with European producers holding a 10 percent share and Japanese producers holding a 15 percent share. (Commission of the European Communities, "Green Paper on Copyright and the Challenge of Technology—Copyright Issues Requiring Immediate Action,'' June 1988, pp. 1'71-172.) Part of the decline in the U.S. share of the software market has come about naturally as software use becomes more widespread abroad and other nation-s' software industries develop.

22 According to some market estimates, in the mid-1980s IBM accounted for 60 percent of world volume in software sales and 70 percent of world profit in software. (Market estimates cited by Gene Bylinski, "The High Tech Race: Who's Ahead," *Fortune,* vol. 114, Oct. *13, 1986,* p. 28.)

23 Ken Wasch, Nicole Field, and Sara Brown, SPA, personal Communication July 30, 1991.

24 Rachel Parker, "Software Spoken Here, " *InfoWorld,* June *25, 1990,* pp. 47-49.

Table 3-3—Top 10 Packaged Software Vendors in Europe (1989)

Company	1989 sales ($ millions)
IBM	$2,120
Siemens AG	398
Nixdorf AG	374
ICL	318
Bull HN	314
DEC	279
Olivetti	236
Unisys	183
Microsoft	152
CAI	152

SOURCE: Market research data from Ovum cited in: Ralph Bancroft, "Europe Strugglingin Software;' Computerworld, July 23, 1990, p,97.

in the international marketplace.[25] (For more on global economic competition, with an emphasis on high technology, see the fall 1991 OTA report *Competing Economies: America, Europe, and the Pacific Rim.*[26]*)*

With the prospect of a unified market and standards in Europe in 1993, U.S. firms are facing new competition from Japanese software producers who are establishing themselves in Europe through acquisitions, as well as invigorated competition from European vendors. The United States faces growing competition in Asia from Japanese producers; at the same time, software industries in other Asian nations are developing rapidly. And in the United States, U.S. firms face new competition in the domestic market from foreign competitors like the Sony Corp. The selected examples of computer hardware and software initiatives in Europe, Japan, Taiwan, and Singapore found in appendix A are intended to give a flavor of the varying stages of maturity and areas of emphasis in some of the overseas industries that are competing with the United States in the global marketplace.

The Issue of Piracy

Creators of commercial software are concerned about their profitability; an important rationale for creation and enforcement of intellectual property rights is to give commercial software developers adequate market incentives to invest the time and resources needed to produce and disseminate innovative products. Illegal copying of software results in financial losses to U.S. software firms both directly, through loss of sales and/or royalties, and indirectly, through loss of investment opportunities.[27]

Retail piracy-duplication of an entire program for sale by ' 'pirate' competitors—and counterfeiting are major concerns of most software companies.[28] These concerns *can be dealt with fairly straightforwardly, at least in theory, by copywright law.*[29] *In practice, enforcement--especially overerseas-is difficult.* Unauthorized end-user copying may be of more concern to some segments of the software industry than to others. For example, noncommercial, private copying by one's current or prospective customers (e.g., making an unauthorized copy of a spreadsheet program for a friend or family member) is a priority concern for developers of *packaged* software, especially personal computer software.[30] Unauthorized end-user copying by businesses and other organizations (e.g., making multiple copies of packaged software instead of obtaining additional legitimate copies or arranging for a site license) is a major concern currently receiving vigorous attention

25 See Schware (1992), op. cit., footnote 21. Schware's analysis concludes that countries without a fairly robust software industry will find it increasingly difficult to ''catch up' and that the learning curves for domestic and export market activities are quite different, with the domestic market providing an important foundation for subsequent export activities. Schware examines software-industry strategies used in India and Brazil and concludes that both industries are trying to ''walk on one leg—the domestic leg in the case of Brazil and the export leg in the case of India' (p. 1).

26 U.S. Congress, Office of Technology Assessment, *Competing Economies: America, Europe, and the Pacific Rim,* OTA-ITE-498 (Washington, DC: U.S. Government Printing Office, 1991).

27 For discussion of revenue 1osscs due to piracy, see U.S. International Trade Commission, ''Foreign Protection of Intellectual Property Rights and the Effect on U.S. Industry and Trade, " February 1988, ch. 4.

28 *IOTA note.* This text uses the phrase ''retail piracy' (suggested by BSA) to mean unauthorized copying for the purposes of selling the illegal copies or close derivatives; ' 'counterfeiting' to mean passing off illegal copies as the real thing; ''end-user piracy' to mean copying by users but not to sell the copies.]

29 Jerome Reichman notes that Anglo-American law tends to use copyright to redress ' 'piracy" (i.e., slavish imitation) because these countries lack a general-purpose unfair competition law based on the European model. Reichman considers that more attention needs to be paid to repression of piracy through international norms of unfair competition law. (Personal communication, Sept. 17, 1991.) See Jerome H. Reichman, *Proprietary'} Rights in Computer-Generated Productions,* paper presented at the WIPO Worldwide Symposium on the Intellectual Property Aspects of Artificial Intelligence, Stanford University, April 1991.

30 This type of unauthorized copying is difficult t. detect and enforce against—copying software at home is relatively easy and inexpensive. Estimates of losses vary and reports of losses may be somewhat overstated because it is not clear that each unauthorized copy displaces a sale.

from software publishers.[31] By contrast, developers are unlikely to worry about end users making copies of 'hard-wired' microprocessor instruction sets at home or at the office, at least with currently available technology.

Estimates of Financial Losses From Piracy

Estimates of financial losses due to piracy vary. ADAPSO (The Computer Software and Services Industry Association) has, estimated that one of every two copies of personal computer software used by corporations in the United States is an illegal copy. In introducing legislation (S.893) to institute strong penalties for violation of software copyright, Senator Orrin Hatch noted that estimates of 1989 losses to the software industry from illegal copying in the United States amounted to $1.6 billion.[32] In 1990, the Software Publishers Association estimated that developers of packaged PC software lost $2.2 billion to piracy within the United States,[33] up from an estimate of $1 billion in 1986.[34]

Industry estimates of losses from piracy show marked increases. In 1988, the International Intellectual Property Alliance (IIPA) estimated that the U.S. software industry lost $547 million to piracy in 12 "problem" countries; by 1990, the IIPA estimated that software piracy in 22 "problem" countries caused software-industry losses of over $2.7 billion.[35] Other estimates of the extent of piracy worldwide are much higher: the Business Software Alliance estimates that-looking at all *types of* software--software piracy worldwide causes the U.S. industry to lose $10 to $12 billion annually, compared to the $12 billion generated by foreign sales of U.S. software.[36]

Redress of piracy abroad is often difficult and is intertwined with issues of technology transfer and assistance to developing countries. *A complete treatment of issues involved in North-South or East-West technology transfer and/or international assistance is beyond the scope of this report.* The following points are intended to suggest some of the complexities in international agreements on intellectual property standards and enforcement. Most of the industrialized, developed countries have strong intellectual property protections, whereas many of the lesser developed countries, where software development itself is much younger,[37] either do not have strong intellectual property laws or do not enforce them.[38] In terms of North-South trade and technology transfer issues, the views of the self-interest of the more industrially advanced nations often conflict with those of the lesser-developed nations. Industrialized countries want to protect industries that are

---

31 The Business Software Association (BSA) notes that site licenses are not the packaged-software industry norm. (Robert W. Holleyman and Lori Forte, BSA, personal communication, July 12, 1991.)

32 S. 893 would amend Title 18 U. S. C., 2319, to include reproduction or distribution of 10 or more infringing copies of One Or more copyright programs. See *Congressional Record*, Apr. 23, 1991, pp. S4862-4863 for Sen. Hatch's statement and the text of the bill, sponsored by Sen. Hatch and Sen. DeConcini.

33 Ken Wasch, Nicole Field, and Sara Brown, SPA, personal communication July 30, 1991.

SPA's estimate is based on "average" software prices and an "expected ratio" of software applications to new personat computers purchased in 1990. SPA obtained hardware sales numbers for DOS-based and Apple computers from Dataquest. SPA obtained expected ratios of software to hardware from Apple, Microsoft, and Lotus; the expected ratios were 3 software applications per DOS machine and 5 per Apple machine, Actual ratios, based on software sales, were 1.78 for DOS machines and 2.55 for Apple machines. (Nicole Field, SPA, personal communication, Aug. 14, 1991.)

34 The SPA estimated that microcomputer-software producers lost about $1 billion in sales to "piracy" (defined by SPA as including both copying for personal use and copying for commercial profit) in 1986. (SPA estimate cited in Anne W. Branscomb, "Who Owns Creativity? Property Rights in the Information Age,' *Technology Review, vol. 91, No. 4,* May/June 1988, pp. 39-45.)

35 The IIPA's 1988 estimate of losses to software piracy considered these 'problem" countries: China, Saudi Arabia, South Korea, India, Philippines, Taiwan, Brazil, Egypt, Thailand, Nigeria, and Malaysia. IIPA estimates cited in: "Curbing International Piracy of Intellectual Property," prepared by Gary M. Hoffman, Report of the International Piracy project, The Annenberg Washington Program, 1989.

The 1990 IIPA estimate considered 22 "problem'* countries; estimate provided by Robert W. Holleyman and Lori Forte of the BSA (personal communication, July 12, 1991).

36 Robert W. Holleyman and Lori Forte, BSA, personal communication, July 12, 1991. Estimate includes all types of software, not just PC software. Foreign sales of PC application software are substantially less: SPA estimates that sates of packaged PC application software amounted to $4.5 billion in 1990--up 22 percent from 1989—and that foreign sales amounted to about $2 billion. (Ken Wasch, Nicole Field, and Sarah Brown, SPA, personal communication% July 30, 1991.)

37 In the United States during the 1970s, Congress and the courts focused on the application and scope of copyright for software; issues concerning patent protection for software-related inventions and algorithms resurfaced in the 1980s. Subsequently, Western Europe, Japan, and Taiwan have developed at least some intellectual property provisions for software. In the Third World, where software development itself is much younger, development of intellectual property measures for software maybe slower than in nations whose domestic software industries are more advanced.

38 See Raymond T. Nimmer and Patricia Krauthaus, "Classification of Computer Software for Legal Protection: International Perspectives, " *International Lawyer, vol. 21, s*ummer 1987, pp. 733-754.

strong sectors in their economies and want to promote free trade to benefit from these investments. Lesser-developed countries want low-cost access to technology in order to promote and modernize business; many (e.g., Brazil) also want to encourage fledgling domestic industries.[39]

The industrialized countries want international agreement with high minimum standards and long periods of protection for intellectual property (for example, 20-year terms for patents, 50-year terms for copyright, 10-year terms for semiconductor chip layouts).[40] Many of these nations also are interested in pursuing harmonization of laws among countries. Industrialized countries argue that, in the long term, ''strong' intellectual property regimes will encourage both domestic innovation and foreign investment by reducing fears of piracy. According to this view, without an acceptable intellectual property regime, technology transfer to lesser-developed countries will suffer because foreign firms will be unwilling to sell goods there or invest in production facilities, absent intellectual property protections. Moreover, proponents consider that adequate protection will also help foster the emergence and development of a domestic software industry[41] and facilitate access to world-class technology .42

In some newly industrializing countries these long-term arguments may be well received but in other countries, where domestic high-technology development is far from a reality, there maybe more urgency for nearer-term considerations. These nations argue that tighter protection for intellectual property will harm development by reducing technology transfer and diffusion in the near term, will strengthen multinational corporations at the expense of domestic industries, and will raise prices of goods (e.g., for patented pharmaceuticals, copyrighted software, etc.) for consumers who are already poor by Western standards.[43] Therefore, near-term U.S. threats of trade retaliation against piracy have been more persuasive than long-term arguments about foreign investment and technology transfer in encouraging countries in Southeast Asia and Latin America to strengthen the terms and enforcement of their intellectual property systems.[44]

Examples of Retail Piracy and Counterfeiting

In 1984, Apple Computer, Inc. filed civil suit against Franklin Computer Corp. for copying Apple's operating system and other software; Franklin subsequently paid Apple $2.5 million to settle the case. Since then, Apple has filed criminal complaints against manufacturers producing ''clones' of its popular Macintosh line of personal computers. In 1990, Apple filed criminal complaints against two Taiwanese manufacturers, Flive Computer Corp. and Akkord Technology, Inc., for producing and selling copies of the Macintosh Plus.[45]

Also in 1990, Novell began concerted efforts to find and file suit against dealers selling or distributing illegal copies of Novell's Netware network software. In announcing these efforts, an attorney representing Novell stated that many instances of Netware piracy involve dealers and resellers who give away illegal copies of Netware to make a hardware sale or install illegally copied Netware under value-added applications.[46] Novell estimated

39 For discussion of Brazil's software strategies, see Schware (1992), Op. Cit., footnote 21.

40 Minimum copyright standards are keyed to the Beme Convention's minimum standards.

41 Robert Schware notes that, in India, software piracy has forced some companies out of the domestic packaged software market and is likely to force others out soon. In Brazil, lack of protection for software prior to the 1987 Software Law (No. 7646) was a serious concern for U.S. companies and the U.S. Department of Commerce. (Schware (1992), op. cit., footnote 21.)

42 These arguments are not unique t. intellectual property for software. For discussion, see Robert M. Sherwood, *Intellectual Property and Economic Development* (Boulder, CO: Westview Press, Inc., 1990).

43 For further discussion of the conflicting self. interests of industrialized and less. developed nations see, for example, ' 'Thought Control: GATT' and Intellectual Property, " *The Economist,* July 7, 1990, p. 68 and Robert Schaffer, "Trading Away the Planet,'' *Greenpeace,* September/October 1990, PP. 13-16. Sherwood (1990), op. cit., footnote 42, offers a different perspective, looking at cases from Brazil and Mexico.
Schware (1992), op. cit., footnote 21, examines Brazil and India and discusses how both domestic and export-oriented strategies are necessary for a country to ''catch up' in software.

44 Robert Sherwood notes that,' 'Mexico's recent enactment of a comprehensive patent and trademark law reflects the long-term argument mOre than the near-term threat of retaliation." (Personal communcation, Aug. 13, 1991.)

45 James Daly, ''Apple Zaps Clone Makers in Taiwan, ' *Computerworld,* vol. 24, No. 14, Apr. 2, 1990, p. 96.

46 Roxanna Li Nakamura and Margie Wylie, "Novell Goes After Dealers Who Sell Fake Netware," *InfoWorld,* vol. **12, No. 28,** July 9, 1990, p. 5 (quoting Stephen Tropp of Shea & Gould).

that each illegal copy of Netware represented from $1,000 to $8,000 in lost sales.[47]

According to Microsoft Corp., software counterfeiting-where pirated programs are sold as legitimate copies—is on the rise in the United States. Microsoft has gathered evidence that its popular microcomputer operating system, MS/DOS, has been hard hit by counterfeiters.[48] (By contrast, copyright infringement of Microsoft's other software products usually takes the form of ''end-user piracy' by individuals or businesses.) A Microsoft market-sampling effort uncovered evidence for lawsuits alleging that counterfeit MS/DOS sales displaced more than $1.5 million in legitimate sales.[49] In June 1991, a Federal jury awarded Microsoft and the Everex Corp. (a personal computer manufacturer and systems integrator) $1.4 million in damages in a suit against eight defendants accused of counterfeiting Microsoft software.[50]

End-User Piracy in Businesses

In 1988, a group of six major U.S. software publishers formed the Business Software Alliance (BSA) to pursue corporate customers abroad who distribute unauthorized copies of programs to employees (rather than obtaining legitimate copies or abiding by license agreements) and to educate users in the commercial and educat ional markets about the copyright laws. BSA'S worldwide activities focus on: 1) litigation on behalf of its members against infringers of software copyrights; 2) public awareness (encouraging organizations to ensure legitimate and ethical software use); and 3) government relations (working with the U.S. and foreign governments to strengthen intellectual property legislation and enforcement) .51 The BSA, currently comprised of eight corporate members, has become an ' 'international policing arm' for the software industry and works closely with SPA.[52]

Even prior to 1988, software industry groups such as SPA and ADAPSO have been addressing the issues of retail piracy *(OTA term: copying to sell the copies)* and end-user piracy *(OTA term: to avoid buying more copies but not to sell copies)* by businesses and other organizations. SPA continues to fight domestic piracy by filing lawsuits and conducting audits of corporations, computer dealers, bulletin boards, and individuals who allegedly have illegal copies of software. Over the last 3 years, SPA has filed over 100 lawsuits for unauthorized copying; in mid-1991 SPA filed suits at a rate of two per week .53

The SPA estimates that unauthorized corporate copying of business PC software in the United States costs software publishers $2.2 billion a year in lost sales[54] and that, for every legal software package in use in the United States, an unauthorized copy is also in use.[55] According to Ken Wasch, executive director of SPA, "It is most unfortunate that the software industry, which is a leading international competitor, is faced with enormous losses every year from individuals unwilling to purchase software legitimately. '[56] Although each unauthorized copy does not necessarily constitute a lost sale for the industry, some industry spokespersons judge that the overall piracy rate is high enough to damage the software industry by limiting funds available for research and development and by driving up retail prices.[57]

---

47 Ibid.

413 ''Counterfeiting'' refers to illegal copies passed off as "the real thing.'

49 Roxanna Li Nakamura, ''Software Publishers Crack Down on Piracy," *InfoWorld,* June 25, 1990, p. 39 (quoting Debra Vogt of Microsoft; Vogt headed the market-sampling project).

50 "Roundup," *The Washington Post,* June 13, 1991, p. B12.

51 ''BSA Profile," Business Software Alliance, July 1991.

52 Andrew Jenks, ''As Software Piracy Spirals, Industry Cops Get Tougher, " *Washington Technology,* June 13, 1991, p. 16 (interview with Robert W. Holleyman, BSA managing director).
BSA has affiliates ("eyes") in about 30 countries. (Robert W. Holleyman and Lori Forte, BSA, personal communication, July 12, 1991.)

53 Ken Wasch, Nicole Field, and Sarah Brown, SPA, personal communication, July 30, 1991.

54 Ibid.

55 According t. SPA, others have estimated that the ratio is as high as five unauthorized copies for every legal one. (Janet Mason, ' 'Crackdown on Software Pirates," *Computer-world,* vol. 24, No. 6, Feb. 5, 1990, pp. 110-115, quoting Peter Beruk of SPA, pp. 110-1 11,)

56 Ken Wasch, Nicole Field, and Sarah Brown, SPA, personal communication, July 30, 1991.

57 Janet Mason, ''Warning: Here Come the Software Police, " *Across the Board,* October 1990, p. 42, quoting Mary Jane Saunders, then-general counsel of SPA.

In 1989, SPA helped five software publishers file the first multivendor suit against a corporation for copying their programs. The suit, against a New York-based publishing company, reportedly reached a six-figure, out-of-court settlement. SPA reportedly also settled four other business piracy cases out of court, with the proviso that the corporations' names would not be released.[58] As part of their education and enforcement efforts, SPA and BSA maintain toll-free piracy hot lines for reporting of cases of suspected piracy and assist firms in conducting voluntary software audits and formulating organizational software-ethics policies.[59] SPA provides a free auditing kit, which comes with diskettes and a license for "SPAudit," SPA's software-auditing program, a list of suggested procedures for a corporate self-audit, sample corporate memoranda on illegal software use, and educational brochures about copying and the law.[60]

In January 1991, SPA announced a $75,000 settlement against a sports management and marketing group based in Northridge, Illinois. In addition to the settlement, the organization was required to destroy its unauthorized copies of software.[61] The SPA, accompanied by a U.S. marshal, had staged a raid in November 1990 and found 80 unauthorized copies of WordPerfect and Lotus 1-2-3. SPA publicized the raid and settlement to remind the public that software piracy is illegal: one SPA advertisement pictured handcuffed wrists with the caption, "Copy software illegally and you could get this hardware absolutely free."[62] In February 1991, SPA announced a $300,000 settlement with a large construction engineering firm, its largest settlement at that time. The firm agreed to destroy all unauthorized copies of software published by Lotus, WordPerfect, and Software Publishing, institute formal internal control procedures in all its offices, and allow SPA to perform annual audits over the next 2 years. In May 1991, SPA announced a $350,000 settlement with a Seattle-based environmental and engineering consulting firm. The firm agreed to destroy the illegal software and institute formal control procedures.[64]

Some software publishers have offered "amnesty" programs allowing unauthorized users to register their copies of software and become eligible for support and future upgrades. In November 1989, one publisher of software utilities announced that it had signed up some 5,000 previously unauthorized users under the amnesty program; users of unauthorized copies paid $20 and received a registered copy of the latest version of the program and a user manual. The firm also signed up an additional $1OO,OOO in corporate site licenses.[65]

Examples of Piracy Overseas

The BSA anti-piracy program operates outside the United States and Canada. For example, in early 1990, BSA identified a major New Zealand bank, an oil company, and an entertainment group as being among firms allegedly pirating software in New Zealand, and announced plans to prosecute one of these organizations. As a result of BSA activities in New Zealand, many firms reportedly began requesting software audits and reevaluating their software acquisition policies.[66]

BSA estimates the level of unauthorized PC-software copying in foreign countries using ratios of the total numbers of legitimate application software packages and hardware units shipped and comparing them to the U.S. ratio. In the United States, approximately 1.66 legitimate software packages were shipped for every hardware unit shipped in 1989; by contrast, there is only one software package

58 Janet Mason, "Crackdown on Software Pirates, " *Computer-world*, vol. 24, No. 6, Feb. 5, 1990, p. 111.

59 In early 1990, SPA reported that more than 20 people a day were calling SPA's hot line to report piracy in their companies. (Ibid., quoting Mary Jane Saunders of SPA, p. 113.)

60 Michael Fitzgerald, "SPA Offers Free (Audit) Software, " *Computerworld*, Dec. 10, *1990*, p. 41. The free software kit is available by writing or calling SPA's offices in Washington DC.

61 Michael Fitzgerald, "SPA To Crank Up Efforts in Copy Crusade," *Computer-world*, Jan. ***28,*** 1991, pp. 1,92.

62 ''Keel~uling Software Pirates, ' *Business Week*, Feb. 18, 1991, p. 122H.

63 SPA press release, Feb. 25, 1991.

64 SPA press release, May 7, 1991.

65 Rachel Parker, ''Xtree Says Amnesty Program Is an 'Overwhelming Success, ' " *InfoWorld*, vol. 11, No. 46, Nov. 13, 1989, p. 87.

66 The New Zealand distributor for Lotus software, a member of the BSA, had estimated that, based on the number of requests for product updates and service it received, it had probably supplied only 20 percent of all the Lotus software in use in New Zealand. (Randall Jackson, ''Software Group Charges Major Firms With Piracy, " *Computerworld*, vol. 24, No. 13, Mar. 26, 1990.)

shipped for every three computers shipped in Italy or Spain and less than one for every computer shipped in France. Comparing each country's ratio with the U.S. ratio and using an "average unit value" for software, BSA estimated that lost revenues from PC-software piracy in 1989 amounted to $628 million in France and $439 million a year in the United Kingdom-roughly equivalent to the amount of software sales revenues in each of these two countries. This method yielded estimated losses in Italy and Spain of $768 million and $792 million, respectively; the largest estimated losses ($1.44 billion) were in what was then West Germany.[67] (See table 3-4.)

In December 1990, three members of BSA (Microsoft Corp., Ashton-Tate, and Lotus Development Corp.) filed suits for piracy against Rhone-Poulenc Films of France and Marconi Instruments, a division of General Electric Co. PLC of the United Kingdom. These suits were filed following court-ordered searches following tip-offs to the BSA.[68] The three software houses also announced settlements of similar copyright suits against units of three French companies: Banque Parabas S.A., Telediffusion de France, and France Distribution Systems.[69] Another suit was filed against Italy's Montedison S.p.A. after a 1988 investigation found that 50 personal computers were running Lotus 1-2-3 with only 1 copy purchased and 20 were running dBase software with 1 copy purchased. Montedison contended that Italian law permits copying for personal use and that this provision applied to corporations.[70]

Using the "ratio" method and assuming that on average one would expect to find at least two application programs on each personal computer,

Table 3-4-Comparison of Software Sales and BSA Estimates of PC-Software Piracy in Selected European Countries

Country	Software sales-1989 estimate ($ millions)	Value of pirated software--BSA estimate ($ millions)
France	$605	$ 628
Italy	190	768
Sweden	188	151
United Kingdom . .	795	439
W. Germany	581	1,440

SOURCE: Dataquest, Inc. and Business Software Alliance, table shown in William M. Bulkeley, "Software Makers Are Pursuing 'Pirates' Around the Globe With Fleets of Lawyers," The *Wall Street Journal,* Dec. 13, 1990, p. B1.

BSA has prepared estimates of PC-software piracy in selected Asian countries.[71] For example, BSA estimates that 75 percent of the software in use in South Korea in 1990 was pirated. Individual companies' estimates of the extent of piracy in South Korea are higher: Lotus estimates that 90 percent of the Lotus 1-2-3 software used in South Korea is pirated; Ashton-Tate estimates that 85 percent of its database software in use in South Korea is pirated; and Microsoft estimates that about 65 percent of its MS/DOS software in use in South Korea is pirated.[72]

In countries encompassing the former Soviet Union, unauthorized copying of software has been rampant. The market research firm IDC estimated that by 1990, U.S. software producers had lost revenues on the order of $1 billion due to illegal copying.[73] In June 1990, the United States and the then Soviet Union signed an agreement[74] that included reaffirmation of both nations' commitments to adhere to the Berne Convention, to provide copyright protection for software, and to provide

67 "BSA Software Piracy Fact Sheet: European Countries, 1989," and Robert Holleyman and Lori Forte, BSA, personal communication, July 12, 1991. BSA estimated losses in 12 European countries in excess of $5 billion using the "ratio" method.

68 "SofWWe Firms Pursue Piracy," *New* Technology *Week,* Jan. 2, 1991, p. 7; and Holleyman and Forte, op. cit., footnote 67. By the close of 1991, the Marconi Instruments case had settled. (Lori Forte, BSA, personal communication, Feb. 14, 1992.)

69 William M. Bulkeley, "Software Makers Are Pursuing 'Pirates' Around the Globe With Fleets of Lawyers," *The Wall Street Journal,* Dec. 13, 1990, pp. B1, B6.

70 Ibid., p. B6. By the close of 1991, the Montedison case had been settled. (Lori Forte, BSA, personal comunication, Feb. 14, 1992.)

71 In its report "BSA Software Piracy Fact Sheet: Asian Countries, 1990," "BSA estimates that onty 3 percent of the software in Thailand is legitimate. By contrast, BSA's method produces a "135 percent" legitimacy figure for Singapore, so the method is somewhat inexact.

72 Damon Darlin, "U.S. Group Targets South Korea Firms for Unauthorized Copying of Software," *The Wall Street Journal, May 23, 1991,* p. B5.

73 "Soviet Software Pirates Are Plaguing Microsoft," *Business Week,* Mar. 5, 1990, p. 84A. According to press accounts, the most widely used program in the Soviet Government was Alpha-DOS, a copy of Microsoft's MS-DOS; the MS-DOS code was copied even down to Microsoft's copyright notice.

74 The agreement also included provisions for other copyrighted works and for patents. (Discussed in Jack E. Brown, abstract of presentation for intellectual property panel of *The Moscow Conference on Law and Bilateral Economic Relation,* Sept. 19, 1990, published in *Computer Industry Litigation Reporter,* Oct. *22, 1990,* pp. 12,074-12,083.)

comprehensive protection for trade secrets. A new law was in process, with a set of principles for software copyrights, but many considered them deficient in that authors were not given enforceable protections against piracy.[75] Copyright enforcement is still problematic in this area of the world-even before the breakup of the Soviet Union the more traditional types of works were widely copied, often for commercial use. For example, unauthorized copying of motion pictures on videocassettes—often, to be shown to paying audiences—has been so widespread that in June 1991, the major U.S. film studios decided as a matter of principle to stop licensing films for showings in what was then the Soviet Union.[76]

Piracy and Trade Issues

On February 15, 1991, the International Intellectual Property Alliance submitted a response to the United States Trade Representative's (USTR) January 11, 1991 *Federal Register* notice requesting comments under the Special 301 provisions of the 1988 Trade Act. (For a discussion of "Special 301," see box 3-B.) The IIPA supplemented its filing on April 18, 1991. In its filing, the IIPA requested that 22 countries (the People's Republic of China, India, Thailand, Indonesia, Mexico, Brazil, Greece, Philippines, Poland, Turkey, United Arab Emirates, Cyprus, Egypt, El Salvador, Germany, Italy, South Korea, Pakistan, Saudi Arabia, Taiwan, the then U. S. S. R., and Yugoslavia) be identified for their failure to protect U.S. intellectual property or for denying market access. The IIPA estimated that 1990 trade losses to the software industry due to software piracy in these countries exceeded $2.7 billion.[77]

In April 1991, the U.S. Economic Policy CounciI recommended to President George Bush that the United States formally cite the People's Republic of China (PRC), India, and Thailand under the Special 301 measure for tolerating violations of U.S. copyrights and patents in a number of industries, including pharmaceuticals, agricultural chemicals, sound recordings, motion pictures, book publishing, and software. The PRC was singled out for software-copyright violations. In its 1991 annual review of foreign trade barriers, the USTR found that PRC lacked a copyright law and that its proposed new law did not meet international standards.[78] The new regulations for software copyright, made public by the PRC'S Ministry of Machine Building and Electronics on June 13, 1991, took effect on October 1, 1991. However, according to the Office of the USTR, the new regulations have too many loopholes to be adequate from the perspective of foreign software developers. In particular, according to the Office of the USTR, the new regulations appear not to provide copyright protection for software developed prior to October 1, 1991.79

In late 1990, the United States opened a formal investigation of Thailand's enforcement of the Thai copyright law. The Special 301 investigation was prompted by a petition from the IIPA, Recording Industry Association of America, and Motion Picture Association of America, alleging massive piracy. The groups estimated that losses to U.S. industry from piracy of U.S. videos, audio cassettes, books, and computer software in Thailand was between $70 and $100 million in 1990.[80] The BSA has charged that 97 percent of the software in use in Thailand is pirated.[81]

---

75 Article 4 of the *All Union Fundamentals of Civil Legislation* deals with software copyright; the *Fundamentals* provide the framework for all commercial law in the U.S.S.R. ("Soviets Recognize Copyrights," *Computerworld* (News Shorts), July 1, 1991, p. 80.)
Reviewer comments indicated that these provisions are considered inadequate and are being protested by the U.S. copyright industries and the USTR (Oliver Smoot, CBEMA, personal communication, June 28, 1991; Ronald Palenski, ADAPSO, *personal* communication July 10, 1991; Robert W. Holleyman and Lori Forte, BSA, personal communication, July 12, 1991).

76 Keith Bradsher, ' 'Hollywood Bars Films To Protest Soviet Piracy," *The New York Times,* June 12, 1991, pp. C13, C16.

77 Information on IIPA filing and estimate of piracy provided by Robert W. Holleyman and Lori Forte, BSA, personal communication, July 12, 1991.

78 Keith Bradsher, ''Panel Asks Bush To Cite 3 Nations,' *The New York Times,* Apr. 26, 1991, pp. Dl, D6.

79 James McGregor, ' 'China's New Software Protection Rules Are Called Inadequate by U.S. official, ' *The Wall Street Journal,* June 17, 1991, p. A7 (quoting Joseph Massey, assistant U.S. trade representative for China).

80 ''U.S. Launches Investigation of Thailand's Weak Enforcement of Copyright Legislation,'' *BNA International Trade Reporter* (News Highlights), vol. 8, Jan. *2, 1991,* p. *4.*

81 Andrew Jenks, ''As Software Piracy Spirals, Industry Cops Get Tougher, ' *Washington Technology,* June 13, 1991, p. 16; and BSA, ''BSA Software Piracy Fact Sheet: Software Piracy in Selected Asian Countries in 1990, "

Box 3-B—The Omnibus Trade Act and ''Special" 301

Under section 301 of the Trade Act of 1974, the United States Trade Representative (USTR) is authorized to identify, investigate, and retaliate against foreign countries engaged in unfair trade practices.[1] The USTR may initiate a section 301 investigation if a foreign country's act, policy, or practice is unreasonable or discriminatory and burdens or restricts U.S. commerce.[2] The statute enumerates trade practices that are unreasonable, citing as an example those which deny 'fair and equitable provision of adequate and effective protection of intellectual property rights. ''3 Any interested party may file a petition with the USTR requesting that action be taken under Section 301.

Subject to the direction of the President, the USTR is authorized to take action after the investigation. The USTR may: 1) suspend, withdraw, or prevent the application of, or refrain from proclaiming benefits of, trade agreement concessions; 2) impose duties or other import restrictions on the products of the foreign country under investigation; and/or 3) restrict, in the manner and to the extent appropriate, access to U.S. markets for services by denying or limiting licenses or other authority to provide services. The USTR must publish in the *Federal Register* its decision whether to investigate under section 301 and any contemplated action at the conclusion of an investigation. The USTR must hold public hearings on issues raised by petitions resulting in investigations and must consult with appropriate congressional committees on the decision to investigate or sanction a foreign country under section 301.

The Omnibus Trade and Competitiveness Act of 1988 (Public law 100-418) treats a wide range of subjects including granting negotiating authority to the President for the current world trade talks, giving a legal mandate for coordinating economic policies and exchange-rate strategies. It also grants the USTR increased surveillance of intellectual property protection in foreign countries. This measure, commonly referred to as ''Special 301, directs the USTR to identify:

1) those countries that deny adequate and effective protection of intellectual property rights[4] or deny fair and equitable market access to United States persons that rely upon intellectual property protection,[5] and

2) those foreign countries identified under paragraph 1 that are determined by the Trade Representative to be priority foreign countries.[6]

---

1 19 U.S.C. 2411-2419.

2 19 U.S.C. 241 l(b).

3 19 U.S.C. 2411 (d)(3) (B) (i)(II).

4 19 U.S.C. 2241 (a)(l).

5 19 U.S.C. 2241(a)(l).

6 19 U.S.C. 2242(a)(2) through (b)(l).

The Berne Convention

The Berne Convention for the Protection of Literary and Artistic Works is a multilateral, international copyright treaty. The purpose of the Berne Convention is to bring nations together in an effort:

> . . . to protect, in as effective and uniform manner as possible, the rights of authors in their literary and artistic works.

The Convention attempts to achieve this objective through the principal of national treatment, which underlies the workings of the Convention.[82] Under the principle of national treatment, each member nation must give the same treatment to the nationals of the other member nations as it gives to its own nationals.[83]

Several secondary exceptions play a role in the Berne system.[84] These exceptions include recip-

---

82 Paul Goldstein, Stella W. and Ira S. Lillick Professor of Law, Stanford Law School, personal communication, Sept. 20, 1991.

83 Mark L. Damschroder, "Intellectual Property Rights and the GAIT: United States Goals in the Uruguay Round, " *Vanderbilt Journal of Transnational Law, vol. 21, No. 2, 1988,* p. 379.

84 Paul Goldstein, Stella W. and Ira S. Lillick Professor of Law, Stanford Law School, personal communication, Sept. 20, 1991.

A priority country is defined in the amendments as one maintaining a large number and wide pervasiveness of practices which constitute significant barriers to U.S. exports and to foreign investment by U.S. persons.

The provisions of Special 301 requires the USTR to identify only those foreign countries with the most egregious practices of denying adequate and effective intellectual property rights or fair market access to U.S. persons relying upon intellectual property protection. The USTR must identify those foreign countries not entering into good faith negotiations or not progressing in ongoing negotiations whose practices have the greatest impact on relevant U.S. products. If the USTR determines that the investigation would harm U.S. economic interests, no investigation is required. The USTR must take into account information submitted by interested parties, so that a U.S. patentee may bring a complaint against a foreign country before the USTR and possibly involve the U.S. Government into negotiations for better protection.

The USTR must enter into consultations with the foreign country to negotiate a resolution to the trade dispute once an investigation under either section 301 or Special 301 is initiated. If the investigation involves a trade agreement, the USTR, under certain circumstances, must request formal dispute resolution under the agreement.

On the basis of these activities, the USTR must decide whether a U.S. right under any trade agreement is being denied or any act, policy or practice comes under section 301. If such an action is appropriate, the USTR must determine what action it should take. This determination must be made within 12 months after the date on which the investigation is initiated, within 18 months if the investigation involves a trade agreement's dispute resolution process. USTR has 30 days to implement its chosen course of action. The actions must be monitored and may be modified. USTR may terminate actions after 4 years if members of industry do not ask that they be continued. If such continuation is requested, USTR must study its potential effect.

A section 301 investigation into Korea's laws protecting intellectual property was instituted in 1985. Korea's law did not provide copyright protection for the works of U.S. authors and provided only limited patent protection for U.S. inventions. After discussions with the USTR, the Korean Government agreed to submit legislation to its National Assembly providing for comprehensive copyright protection for written works, sound recordings, and computer software. The Korean Government agreed to seek stronger patent laws and to join the Universal Copyright Convention.[7] India, the People's Republic of China, and Thailand were name "priority foreign countries" by the USTR in May 1991. An investigation was not conducted into the intellectual property laws and practices of Thailand, as that country was already the subject of two separate investigations based on complaints filed by the International Intellectual Property Alliance and the Pharmaceutical Manufacturer's Association.[8]

7 Albert C. Smith and John Sullivan, "The Impact of U.S. Patents and Customs on Importation.., " *The Computer Lawyer*, vol. 8, *No.* 10, October 1991.

8 Ibid.

SOURCE: OTA 1992, and cited footnotes.

rocity, establishment of minimum rights in the substantive clauses of the Convention, the principle of automatic protection, and the provision for making reservations.

Reciprocity alters the working of national treatment somewhat. The principle of reciprocity provides that a nation may limit the protection granted to a foreign national to that level bestowed upon its own citizens in the foreign nation of the person seeking its domestic protection. Thus, a member nation may treat foreigners as the foreigners' own governments would have treated them under similar circumstances. Reciprocity often results from political pressure from domestic interest groups who are receiving substandard protection in the foreign nation.[85]

The establishment of minimum rights in the substantive clauses of the Berne Convention and the principle of automatic protection work in tandem, giving authors and artists substantive protections

85 Damschroder, op. cit., footnote 83.

without a requirement of compliance with formalities.[86]

The provision for the making of reservations in article 27 of the Berne Convention allows member nations to make reservations to the introduction of new rights where such reservations are required by the domestic laws. The reservations may later be withdrawn when domestic law is brought into line with Berne.[87]

Article 2 of the Berne Convention defines what is included within the subject matter of protection under the Convention. Certain categories of works, among them books, pamphlets, addresses, choreographic works and sculpture: are explicitly included subject matter. Berne specifies that the scope of protection of other works should be defined by domestic law of each individual member country of the Berne Union. The Berne Convention does not address computer programs and databases, largely because the Convention was most recently revised in 1971, when computer technology was not so prevalent. It has been suggested that because computers are not mentioned, much confusion exists as to the interpretation of the current text with respect to these works of new technology. However, under Berne, there appears to be no written obligation to protect computer programs.[88]

The United States acceded to the Convention on March 1, 1989. At that time, the United States was already a member of the Universal Copyright Convention (UCC). Both groups are administered by United Nations agencies: Berne by the World Intellectual Property Organization (WIPO), and the UCC by the United Nations Educational, Scientific and Cultural organization (UNESCO) (to which it had already been a party for many years). To implement its accession, it was necessary for the United States to pass domestic legislation that created the most significant changes to the U.S. copyright law since 1976.[89] The Berne Convention was originally adopted to apply to works traditionally subject to copyright. Since then, new advances in technology required that copyright protect works in addition to art and literature. New technologies such as software and databases, international in nature because of the ease with which they can be copied and disseminated across national boundaries, have made international copyright protection and the changes rendered by U.S. adherence in Berne at least as important as domestic copyright protection.[90]

As required by article 36 of Berne, the United States made changes in its copyright law to make its law compatible with the treaty by passing the Berne Convention Implementation Act of 1988. Some of these changes bear on Congress' options to treat software differently from other kinds of copyrighted works. Among these changes are the following:

1. *Abolition of mandatory notice of copyright*—Notice of copyright, traditionally indicated in the United States by the symbol ©, the year of publication, and the author's name, is no longer required for works first published on or after March 1, 1989. Failure to place a notice of copyright on copies or phonorecords of such works can no longer result in loss of copyright. This abolition of the notice requirement is not retroactive, and voluntary use of the notice is still encouraged. If notice does appear, it limits a defense of innocent infringement.[91]

86 Ibid. Some concern has been expressed that adoption by the United States of the Berne Convention, which has eliminated the need even to place copyright notices on published material, has raised the issue of whether and to what extent a public disclosure system should be required for software copyright protection. Some assert that under existing copyright law and practice it is impossible for a legitimate software developer to conduct a due diligence process, as the scope of existing protection is not available in any public form since a copyright claimant can delay filing any public record until after an alleged infringement has occurred. This contrasts with current patent law and practice, whereby an applicant for a patent is required to specifically claim the elements to be protected. Those claims, if allowed, become of public record. Subsequent inventors can use those records as part of their due diligence to determine whether a proposed development will infringe existing rights. These observers raise the question whether, and to what extent, a similar practice should be required of software copyright claimants. Thomas E. Kirkland, Vice President and General Counsel, Microelectronics and Computer Technology Corporation, personal communication, Sept. 24, 1991.

87 Ibid.

88 Carol A. Motkya, "Impact of U.S. Adherence to the Berne Convention," *Rutgers Computer & Technology Law Journal,* vol. 16, 1990, pp. 195, 213-215.

89 The Berne Implementation Act expressly states that the Berne Convention is not self-executing in the United States and that it is not an independent source of right in the United States, though it is in other countries, Thus, copyrighted works receive protection under domestic U.S. copyright law as amended by the Berne Act, rather than by direct enforcement of the provisions of the Berne Convention itself,

90 Motyka, op. cit., footnote 88 at p.195.

91 U.S. Copyright office, The *United States Joins the Berne Union*, Circular 93a (Washington, DC: U.S. Government Printing office, 1989), p.4.

2. *Mandatory* deposit-Copyright owners must deposit in the Copyright Office two complete copies or phonorecords of the work subject to copyright that are publicly distributed in the United States, whether or not the work exhibits a notice.[92]
3. *Registration-Berne* brought about a two-tier registration system that differentiates between works of U.S. origin and works of foreign origin with regard to registration.[93] Under 17 U.S.C. 41 l(a), authors or works whose origin is not the United States are exempt from the requirement to register in order to bring an infringement action. However, works of U.S. origin must be submitted to the Copyright Office for registration before suit can be brought. This domestic requirement for registration can be problematic for databases, which are copyrightable under limited circumstances. Registration requirements for active databases are complex because such databases are being revised continually.
4. *Compulsory* licenses—Article 1l(l)(i-ii) of Berne grants authors of literary works the exclusive right of authorizing public recitation of their works and communication to the public of the recitation of their works. This provision had an immediate effect upon U.S. law in 17 U.S.C. 116, wherein the right to publicly perform music by means of a jukebox was the subject of a compulsory license. The Berne Implementation Act amended the law to provide for negotiated licenses between jukebox operators and copyright owners, so that such negotiated licenses take precedence over compulsory licenses.[94] From a broader perspective, this provision precludes the use of compulsory license for any literary works. Thus, compulsory licenses cannot be applied to computer programs, as these are considered under U.S. law to be literary works.
5. *Duration of protection-The* Berne Convention establishes the minimum terms of protection that must be provided by member countries. The general term of protection is life of the author plus 50 years following the author's death.[95] Special minimum terms are indicated for certain categories of works. As a result, a term of protection for computer software less than the life of the author plus 50 years is precluded by U.S. adherence to Berne.

The General Agreement on Tariffs and Trade

The General Agreement on Tariffs and Trade (GATT) is a multilateral trade agreement, entered into force in 1948, intended to promote freer trade among member countries. The GAIT is the main instrument regulating trade among market economy nations of the world. The obligations contained in the the original text of the GATT' have been augmented and changed periodically at the prompting of its signatories, most recently during the Tokyo Round of negotiations from 1973 to 1979. Procedures under the GATT "provide for extensive exchanges of information, regular review of key subject areas and *ad hoc* consultations on particular concerns. " The GATT's dispute settlement procedure is the last resort for governments involved in a trade dispute. This procedure is termed the "panel procedure, ' and consists of third-party adjudication of claims .96

A new round of negotiations under the GATT was begun in 1986, and was originally scheduled to end in December of 1990 (see box 3-C). Because the agenda for the talks was set at Punta del Este, Uruguay, this round of negotiations is referred to as the Uruguay Round. During the round, negotiators undertook to improve and strengthen the existing GATT structure and to extend the rules of the GATT to aspects of international trade that remain largely outside the discipline of the GATT The United States was successful in making intellectual property rights, as well as other matters, a part of the negotiations.[97] Currently the GATT contains no specific express provisions for protection of intellectual property rights. While an important U.S. goal

92 Ibid.

93 Ibid.

94 *The United States Joins the Berne Union,* op. cit., footnote 9, pp. 4-5.

95 The Berne Convention for the Protection of Literary and Artistic Works, Paris Act of July 24, 1971 as amended on Oct. 2, 1979, article 7(1), World Intellectual Property Organization (Geneva 1987).

96 Damschroder (1988), op. cit., p. 384.

97 Ibid., p. 372.

320-226 0 - 92 - 8

Box 3-C—History of the Uruguay Round

The Uruguay Round of' the General Agreement on Tariffs and Trade (GATT) negotiations is the eighth round of multilateral trade negotiations and may be the most comprehensive and significant in the GATT's history.[1] In addition to improvement of existing GATT articles and inclusion of trade-related aspects of intellectual property rights, the United States succeeded in including the following subjects in the Uruguay Round negotiations: 1) greater liberalization of the agriculture policies of member nations; 2) trade-related investment measures; 3) trade in services; and 4) modification and strengthening of the GATT's dispute settlement mechanisms.[2]

In December 1988, a mid-term review began in Montreal to assess progress made during the first half of the Uruguay Round and to arrive at framework agreements on work over the remaining 2 years. Disagreement on agriculture delayed the conclusion of the mid-term review until April 1989, when negotiators met in Geneva to approve the set of mid-term agreements.[3]

Negotiators set July 1990 as the deadline for final draft framework agreements. These agreements were expected to show the broad shape of the final package. Instead, in July negotiators remained far apart on most major issues. The absence of consensus correctly indicated that the final months of negotiations would be intense and difficult.[4] Since the breakdown of the talks in early December 1990, the European Community put forward proposals for reforming the common agricultural policy. As discussed previously, talks held in Geneva January 31 through February 2, 1991 were aimed at producing a platform for restarting the round, according to GATT Director General Arthur Dunkel.[5]

The Congress oversees the negotiations and the Bush administration is required to consult with Congress during negotiations and prior to entering into an agreement. The President will submit the eventual agreement package and implementing legislation to the Congress when it is 'agreed upon by negotiators. Under the fast-track approval procedure, Congress must take an up-or-down vote on the legislation within 60 days after the President submits the legislation (90 days in the case of an implementing revenue bill).[6]

---

1 Mark L. Damschroder, "Intellect Property rights and the GATT: United States Goals in the Uruguay Round," *Vanderbilt Journal of Translational Law*, vol. 21, No. 2, 1988, p. 390.

2 "The General Agreement on Tariffs and Trade," memorandum of Ernst& Young, Washington, DC Office, July 1990.

3 Lenore Sek, Trade *Negotiations: The Uruguay Round*, Congressional Research Service, Issue Brief No. IB86147, p. 3.

4 Ibid., at p. 4.

5 "Free Trade Talks Imperiled by Fight on Farm Subsidies," *The New York Times, Nov.* 13, 1990.

6 Ilona *B.* Nickels, Trade Agreement Legislation on a "Fast Track," CRS *Review*, May-June, 1990, pp. 11-12; see also Sek, op. cit., footnote 251 at p. 11.

SOURCE: OTA, 1992.

during the Uruguay Round is to develop better international standards for protection of intellectual property rights and to establish dispute settlement and enforcement procedures in the GATT, there is a significant portion of the membership of the GATT', particularly the Third World, that opposes the inclusion of intellectual property rights into the instrument. Certain developing countries, led by India and Brazil, question whether the GATT is an appropriate forum and prefers that such discussion take place in the World Intellectual Property Organization, a treaty administered by the United Nations that lacks any dispute settlement provision. Some believe that the effectiveness of the intellectual property provisions in the GATT will depend in large part upon the enforcement provisions of the GATT treaty .98

The talks collapsed on December 7, 1990, when the United States, along with most of the other participating countries, and the European Community failed to agree at a ministerial meeting in Brussels on ways to revamp agricultural trade. After the breakdown in negotiations, the Uruguay Round was originally extended into the first few months of 1991. GAIT Director General Arthur Dunkel met separately with representatives of the EC, the United States, Japan, and the Cairns Group of agricultural

98 Damschroder, op. cit., footnote 83, at 390.

producing countries January 31 through February 1, 1991 in an effort to find some common ground for resuming the formal Uruguay Round in February.

In light of the lack of results of these consultations, most delegates expected that the Uruguay Round would continue through the summer, if the U.S. administration could obtain from Congress an extension of its "fast-track" negotiating authority (see box 3-D). Indeed, the Bureau of National Affairs reported that Dunkel obtained the agreement of most major players in the Uruguay Round of trade negotiations to have the talks extended at least through the end of 1991. U.S. Trade Representative Carla Hills had stated that she was hopeful that Congess would approve an extension of the administration's fast-track authority beyond June 1, while some lawmakers, including Senator Max Baucus, chairman of the Senate Finance Subcommittee on International Trade, indicated that the subcommittee would oppose extending the authority for the Uruguay Round without some evidence of significant progress toward successful completion of the round. President Bush formally requested an extension of the fast-track implementation legislation on March 1, 1991. Following the lead of the House of Representatives, the Senate voted to extend the fast-track negotiating authority for 2 years.[99]

The U.S. Trade Representative

The Office of the U.S. Trade Representative, an agency of the Executive Office of the President, is the body involved in the GATT negotiations on behalf of the United States. The process by which the USTR arrives at treaty proposals such as those for the GATT is not a highly visible one.[100] The statutory basis for the process is set out in 19 U.S.C. 2155, which provides that the president must seek information and advice from representative elements of the private sector and the nonfederal government with respect to negotiating objectives and bargaining positions before entering into a trade agreement. To effect such communication between the executive branch and the private sector, the statute provides for the establishment of an Advisory Committee for Trade Policy and Negotiations to provide overall policy advice on these matters. This broadly based committee is made up of representatives of nonfederal governments, labor, industry, agriculture, small business, service industries, retailers, and consumer groups. The committee is to be representative of the key sectors and groups of the economy, especially those affected by trade. Members are recommended by the USTR and appointed by the President.

The statute also provides for the establishment of individual general policy advisory committees for these same interest groups to provide general policy advice. These committees are organized by the USTR and the Secretaries of various executive departments, and are to meet at the request of the USTR and the Secretaries of the executive departments to provide policy advice, technical advice, and information.

Negotiators at USTR meet with parties concerned with the negotiations, including the White House and pertinent government agencies (in this case the Copyright Office, the Department of Commerce, the Department of State, or the Patent and Trademark Office, *inter alia),* as well as with industry representatives, on both a formal and informal basis.[101] In some cases a specific call is made for public comment, in other instances the USTR seeks out key players it believes to have a stake in the negotiations.[102] Congress' role, as discussed above, is to monitor USTR'S activities, to act as a source of advice and consultation, and to hold public hearings on issues critical to the negotiations.[103] Some sources assert that the treaty implementing legislation is essentially written by the Congress and the

99 "Brussels Meeting To Conclude Uruguay Round in Jeopardy, Director General Dunkel Warns," *International Trade Reporter, vol. 8, No, 22,* May 29, 1991, p. 802.

100 Personal communications with officials of the Office of the U.S. Trade Representative, the Department of Commerce, and the U.S. International Trade Commission September 1990 to November 1990.

101 Eliza Patterson, Deputy Director, External Affairs, International Trade Commission, personal Communication, Aug. 13, 1990.

102 Emery Simon, Office of U.S. Trade Representative, personal communication, September 1990.

103 Patterson, op. cit., footnote 101.

Box 3-D—Role of the Congress in Trade Negotiations and the Fast-Track Implementation Procedure

The Administration and Congress have interactive roles in the negotiation of trade agreements. Under his constitutional powers to negotiate international agreements and to conduct international relations, the President has certain power to negotiate on trade matters. Congress has constitutional authority to regulate foreign commerce. The President may negotiate trade agreements, but if those agreements require a change in the law, the Congress must approve the statutory changes.

Congress has traditionally delegated some of its authority to the President for past rounds of multilateral trade negotiations. The Trade Act of 1974 (Public Law 93-618) authorized the President to agree to certain matters during the Tokyo Round. The U.S. Government implemented the Tokyo Round agreements by enacting the Trade Agreements Act of 1979 (Public Law 96-39).

The President began negotiations in the Uruguay Round without congressionally delegated authority. At the time, however, the Congress was considering legislation to extend negotiating authority to the President and subsequently approved authority under Public Law 100-418, the Omnibus Trade and Competitiveness Act of 1988 (1988 Trade Act).

The 1988 Trade Act designated negotiating objectives, which gave congressional direction to the U.S. negotiator. It listed specific objectives on 16 subjects, including intellectual property. The 1988 Trade Act also extended the President's authority to enter and proclaim certain tariff and nontariff agreements, imposed limits on tariff reductions, and provided for fast-track approval procedures for certain agreements, as well as congressional withdrawal of fast-track consideration ("reverse fast-track").

The fast-track approval procedure ensures that as long as the Administration has consulted adequately with Congress prior to and during negotiations and has negotiated an agreement consistent with congressional directives, then Congress quickly and without amendment will consider and vote on the agreement and implementing legislation as submitted by the Administration. This fast-track procedure is an exercise of the constitutional rulemaking power of the House and the Senate. Either House of the Congress may change these procedures in the same way that it may change any of its rules. The 1988 Trade Act provides that Congress may also terminate the fast-track procedure. Such a change or termination of the fast-track procedure would signal a lack of congressional support for the agreement.

executive branch, including the USTR, together.[104] The perception exists, however, that the USTR has not involved the Congress as fully as would be considered appropriate.[105] Indeed, the fast-track procedure for passage of implementing legislation relies upon congressional awareness and participation throughout the negotiation and drafting process.[106]

Trade-Related Aspects of Intellectual Property Rights and the U.S. Proposal to the GATT

That portion of the GATT negotiations concerned with intellectual property is referred to as TRIPs, Trade Related Aspects of Intellectual Property Rights. Part Two of the U.S. Trade Representative's Draft Agreement on TRIPs, entitled "Standards in the Field of Intellectual Property,' provides specifically for protection of computer software as a literary work under its copyright provisions. The Berne Convention forms the basis for protection set forth in the GATT proposal.[107] Article 1 of these standards establishes that the Berne Convention provides the minimum economic rights to be granted to authors by contracting parties to the GATT'. Articles 2 through 7 set forth the additional protections provided specifically for computer software and databases in the TRIPs. The more controversial aspects of the U.S. draft agreement are discussed below.

104 Ginger Lew, Ernst & Young, Washington, DC, personal communication, Aug. 9, 1990.

105 Testimony of the U.S. Trade Ambassador Carla Hills before the U.S. House of Representatives Subcommittee on Intellectual property and Judicial Administration, hearing on Intellectual Proprety and International Issues, May 15-16, 1991.

106 Patterson, op. cit., footnote 1101.

107 Lewis Flacks, Policy Planning Advisor to the Register of Copyrights, personal communication, Aug. 7, 1990.

As part of the required consultation with Congress, the President must meet and consult with the appropriate committees during negotiations and prior to entering into an agreement. At least 90 days before entering into an agreement, the President must notify the House and the Senate of the intention to enter into the agreement and must publish notice of such intention in the *Federal Register.*

After entering into an agreement, the President must submit a document to the House and the Senate containing a copy of the final legal text of the agreement. The document also must contain: 1) a draft of the implementing bill, 2) a statement of any administrative action proposed to implement the agreement, and 3) supporting information as described by law.

Under the fast-track approval procedure, Congress must take an up-or-down vote on the implementing legislation within 60 days of session after the President submits the legislation (within 90 days in the case of an implementing revenue bill):

- The implementing bill submitted by the President must be introduced in each House of the Congress on the same day that a trade agreement is submitted to the House and the Senate. The bill is referred jointly to the appropriate committees.
- The committees have 45 days to report the implementing bill. At the end of the 45 days, if the committees have not reported the bill, they are discharged from further consideration. The bill is placed on the appropriate calendar.
- Within 15 days of session after the committees report the bill or are discharged from further consideration of the bill, a floor vote must be taken in each House of Congress. No *amendments may be made and debate is limited.*
- *If* it is an implementing revenue bill, the bill must originate in the House of Representatives. After the bill is received in the Senate from the House, it is referred to the appropriate Senate committees. The Senate committees have 15 days in which to report the bill, otherwise the committees are discharged from further consideration. A vote in the Senate shall be taken within 15 days after the committees report the bill or after the committees are discharged from further consideration of the bill.

SOURCE: Lenore Sek, Trade *Negotiations: The Uruguay Round,* Congressional Research Service, Issue Brief No. IB86147, p. 3.
Ilona B. Nickels, Trade Agreement Legislation on a "Fast Track" *CRSReview,* May-June, 1990, pp. 11-12.

Article 2

Article 2 attempts to define the scope of protection for computer-related works. Section (l)(a) provides for protection of computer programs, which, under the provisions, include "application programs and operating systems. " Section (l)(a) further provides that computer programs may be expressed in any language, including source and object code, and that these are to be protected as literary works. Section (l)(a) also provides for protection of works created by or with the use of computers.

Protection of computer programs in source and object code are the subject of article 2, section (l)(a). This provision essentially incorporates the provision of section 101 of the U.S. Copyright Act, Title 17, U.S.C. 101 regarding the treatment of computer programs and databases as literary works [108] and codifies the outcome of litigation in the American courts dealing with these issues. [109] It provides that both the source and object code of computer programs, as well as operating and application systems, are the subject of protection.

It has also been suggested that the language of article 2, section (l)(a) that states that ''all types of computer programs. . . expressed in any language. . ." should be noted as being potentially construable to include algorithms and source code languages. This

108 Title 17, sec. 101 provides in pertinent part that "literary works'' are:

> . . . works, other than audiovisual works, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of material objects, such as tapes, disks, or cards, in which they are embodied.

Melville Nimmer states that this statutory definition is broad enough to include computer databases and programs. This issue is further discussed in the analysis of the Apple decision, ch. 2.

109 For further discussion of U.S. case law pertinent to the protection of computer programs, see the discussion of *Apple v. Franklin, as* well as other relevant cases, ch. 2.

is especially pertinent in light of the recently adopted European Community software directive,[110] which may not specifically preclude protection of algorithms and computer languages.[111] Works created by or with the use of computers are granted protection under article 2, section (l)(a) of the U.S. TRIPs proposal.[112]

The compendium of the U.S. Copyright Office explains that the term ''authorship' in the copyright law:

> implies that, for a work to be copyrighted, it must own its origin to a human being.

To determine whether works created by computers are works of authorship within the meaning of the 1976 Copyright Act, a distinction must be made between computer-aided works and computer-generated works. Computer-aided works, works generated with the aid of a computer, are entitled to the same copyright treatment as other works created with the more traditional implements of human authorship. However, if and when artificial intelligence makes it possible for new works to be generated by a computer alone and not merely with its assistance, new questions will be presented. To date, there are no cases on computer-aided or computer-generated works. When the National Commission on New Uses of Technological Works (CONTU)[113] considered the question, it believed that there was "no reasonable basis for considering that a computer in any way contributes authorship to a work produced through its use. ' CONTU concluded that:

> . . . no special problem exists with respect to the ''creation of new works by the application or intervention of such automatic systems or machine reproduction'; that existing statute and case law adequately cover any questions involved; and that no action by Congress is necessary at this time.[114]

This finding by CONTU effectively deferred consideration of the problem of computer-generated works, and focused exclusively on the question of computer-aided works. CONTU examined a number of issues with respect to the issue of computer-aided works.[115]

First, it asked whether a computer-aided work is an ''original work of authorship' within the meaning of the Constitution and the current statute. CONTU expressed the view that the intervention of a computer should not affect the copyrightability of any work, noting that the quantum of originality needed to support a claim of authorship in a work is small. ' Although computers may be used to:

> . . . produce writings that lack the degree of originality held necessary to copyright, [still] the criteria that determine if a work is sufficiently original to qualify for copyright are already well established, and the intervention of the computer should not affect them,

CONTU compared a computer to a camera in its ability to extend human power rather than substitute for it, A computer-aided work is no less a work of human authorship than a work created by the aid of a camera, a typewriter, or any other ' 'inert instru-

---

110 The European Economic Community Treaty and the European Community Council directive on the legal protection of computer programs are discussed later in this chapter.

111 Jerome Reichmann, Vanderbilt University, personal communication, Aug. 9, 1990. The Council Directive on the legal protection of computer programs (91.250 .EEC) provides in article 1, section 2:

> Protection in accordance with this Directive shall apply to the expression in any form of a computer program. *Ideas and principles which underlie any aspect of a program including its interfaces, shall not be protected by copyright under the Directive.* (Emphasis added.)

This section differs from a prior iteration, which provided:

> protection in accordance with this Directive shall apply to the expression in any form of a computer program but shall not extend to the ideas, principles, logic, algorithms or programming languages, underlying the program. Where the specification of interfaces constitutes ideas and principles which underlie the program, those ideas and principles are not copyrightable subject matter.

Other commentators are uneasy with such a conclusion that the EC directive does not, as a result of this language, include an express exclusion of ''algorithms'' from copyright. It should be noted in light of the language of the directive that Japan, as well as several other countries, have adopted express exclusions for algorithms from copyright protection, Raymond Nimmer, Professor of Law, University of Houston Law Center, personal communication, Sept. 23, 1991.

112 For further discussion of treatment of works created by or with the use of computers, see Pamela Samuelson, ''Allocating Ownership Rights In Computer-Generated Works, " *University of Pittsburgh Law* Review, vol. 47, p. 1185.

113 For further discussion of CONTU and its activities, see ch. 2.

114 *Final Report of the National Commission On New Technological Uses of Copyrighted Works*, July 1978 (Washington, DC: The Library of Congress, 1979), p. 46.

115 Ibid.

ment' which is ''capable of functioning only when activated either directly or indirectly by a human. [116]

Second, CONTU addressed the question of who is the author of a computer-aided work. CONTU determined that "the obvious answer is that the author is one who employs the computer. ' Where a number of people have been involved in using the computer to prepare a complex program or database the author may be a common employer of the programmer under the work-made-for-hire doctrine. On the other hand, a team of independent programmers might be joint authors, and they can define their relative rights by agreement. [117]

Some commentators believe that this provision reflects the status of the law in the United States, and that foreign jurisdictions generally agree that such protection is appropriate. Others believe that the area has not yet been examined by the courts, and is therefore not yet ripe for negotiation in the GATT.[118]

Databases are protected under section (1)(b) of article 2:

> . . . if they constitute intellectual creation by reason of the selection, coordination, or arrangement of their contents.

While this issue has also been addressed in the U.S. courts, little consensus on the issue had been reached until a recent decision by the U.S. Supreme Court.[119]

Section 2 (a) of' article 2 of the TRIPs proposal delineates economic rights provided to contracting parties that are over and above those minimum rights provided for in Berne. The rights provided in this section closely mirror those rights set forth in section 337 of the Trade Act of 1974 and section 602 of the Copyright Act. These include the right to import into the territory of the contracting party lawfully made copies of the copyrighted work, and the right to prevent the importation of unauthorized copies,

Article 2, section 2(b) of TRIPs deals with the issue of rental rights in computer programs, an issue recently considered by the U.S. Congress, which passed legislation in November 1990 (Public Law 101-650, section 801-805). Section 2(b) states that the first sale of the original or a copy of a computer program shall not exhaust the rental or importation right in the computer program. The provision defines "rental right' as the right to authorize or prohibit the disposal of the possession of the original or copies for commercial advantage. This provision reflects the substance of legislation recently passed.[120]

This TRIPS provision, like the U.S. legislation, limits the first sale doctrine, embodied in 17 U.S.C. 109. The first sale doctrine permits the owner of a lawfully made copy of a copyrighted work to sell or otherwise dispose of the possession of that copy without the permission of the copyright owner. The provision and the legislation are a response to the computer software industry's concern about the rental of its works and the ease with which they can be copied.

Among the justifications for this limitation of the first sale doctrine is the argument that computer programs, unlike movie videos, cannot realistically be rented for an evening's entertainment and then returned. It is argued that the various commands and features require study and understanding of users manuals, and many programs have no real value until the user enters their own database. [121] The industry asserts that unchecked rental of software and its unauthorized copying feeds on itself, since copying drives the price of software up and makes the incentive to pirate greater, It has been asserted, however, that rental restrictions could interfere with the practice of ''trial rentals," which would limit consumer information and potentially reduce sales. 122

The issue of software rental has garnered significant attention internationally. The directive of the Council of the European Economic Community on the legal protection of computer programs includes

116 Ibid.

117 Ibid.

118 Jerome H. Reichman, Professor of Law, Vanderbilt University, personal communication, Aug. 9,1990.

119 For further discussion of protection of databases under U.S. law, see ch. 2.

120 For additional discussion of the issue of rental rights and the enacted legislation, see ch. 2.

121 Testimony of Ralph Oman Register of Copyrights Hearing of the House Subcommittee on Courts, Intellectual Property, and the Administration of Justice, July 30, 1990.

122 Ibid.

a provision to allow authors to retain rental rights in their software after its first sale.[123]

Draft Final Act Embodying the Results of the Uruguay Round and Trade-Related Intellectual Property

On December 20, 1991, GATT Director General Arthur Dunkel tabled the "Draft Final Act Embodying the Results of the Uruguay Round of Multilateral Trade Negotiations. ' This draft was issued with the understanding that it offered a concrete and comprehensive representation of the final global package of the results of the Uruguay Round, that no single element of the draft could be considered as agreed upon until the total package is agreed, and that final agreement on the draft act would depend upon achievement of meaningful results for all parties in ongoing negotiations pertaining to access to markets and in liberalization commitments in the area of services. The draft discussed copyright and related rights, including provisions for computer programs and compilation of data.

According to article 9 of the draft, parties to the agreement are required to comply with articles 1 to 21 and the Appendix of the Berne Convention, with the exception of article *6bis* of the Convention which deals with moral rights of authors. Further, the draft states that copyright protection shall extend to expression and not to ideas, procedures, methods of operation or mathematical concepts.

Article 10 provides that computer programs, whether in source or object code, shall be protected as literary works under the Berne Convention. Compilations of data or other material, whether in machine readable or other form, which are intellectual creations because of the selection or arrangement of their contents are entitled to protection, without prejudice to any copyright subsisting in the data or material itself.

Article 11 provides that with respect to computer programs, authors shall be provided the right to authorize or prohibit the commercial rental to the public of originals or copies of their copyrighted works. This obligation does not apply to rentals where the program itself is not the primary purpose of the rental.

Article 12 provides for a term of protection for computer programs of 50 years from the end of the calendar year of authorized publication, or absent such authorized publication within fifty years from the making of the work, fifty years from the end of the calendar year of the making.

Article 13 of the draft agreement provides that parties to the agreement ''shall confine limitations or exceptions to exclusive rights to certain special cases which do not conflict with a normal exploitation of the work and do not unreasonably prejudice the legitimate interest of the right holder."

Earlier in the text, the Draft also sets forth "Measures in Favor of Least-Developed Countries,' which would affect the application of the computer software provisions to least-developed countries that are signatories to the GATT. Under these provisions, parties to the agreement recognize the plight of the least developed countries and their special needs with respect to effective participation in the world trading system, especially in the area of market access. It states that least-developed countries, recognized as such by the United Nations, will be required to apply the terms of the agreement only to the extent consistent with their individual development and capabilities.

This portion of the draft also states that parties to the agreement agree that expeditious implementation of measures taken in favor of least-developed countries shall be ensured through regular reviews, and that least-developed countries are to be accorded increased technical assistance in the development, strengthening and diversification of their production and export bases to enable them to maximize the benefits from liberalized access to markets. They further agree to keep the problems of these countries under review and adopt positive measures which facilitate the expansion of trading opportunities in favor of these countries.

Participation by the United States in Other International Treaties

In addition to its participation in such multilateral treaties as the Berne Convention, and the General Agreement on Tariffs and Trade,[124] the United States is a party as well to many bilateral treaties

[123] Council Directive of May 14, 1991 on the legal protection of computer programs (91/250/EC), art. 4(c).

[124] The GATT and U.S. proposals for trade-related intellectual property rights are discussed earlier in this chapter.

with individual nations in which provisions for intellectual property protection for computer software are specifically laid out. In large part, the bases for these bilateral treaties are the provisions of the Berne Convention. Most recently, the United States has established such bilateral treaties with the then Soviet Union, Romania, Czechoslovakia and other emerging Eastern European countries.[125]

Also, as discussed above, the United States is a party to the Universal Copyright Convention (UCC), which was created in 1952 by UNESCO to provide an alternative multilateral agreement to Berne which would not require the United States and other Western countries to forfeit copyright notice requirements. While the UCC prohibits member states from requiring formalities as a prerequisite for copyright protection, the UCC differs from Berne by dispensing with those formalities only upon use of a prescribed copyright notice. The UCC grants priority to the Berne Convention, making it the premier multilateral copyright treaty with the highest standards of protection, followed by the UCC.[126]

The European Economic Community's Directive on Legal Protection for Computer Software

The Council Directive

On May 14, 1991, following its 1988 Green Paper (see box 3-E), the European Community adopted its extensively debated Council Directive on the legal protection of computer programs.[127]

Preamble

The preamble of the Council Directive asserts the varying nature and scope of protection afforded to computer software among member states and the negative consequences of these differences on the functioning of the European Common Market.[128] This section emphasizes the investment of human, technical and financial resources in development of computer programs, the increasingly important role played by computer programs in a broad range of industries and the resulting fundamental importance of computer software to the European Community's industrial development.[129] It also asserts that differences in protection which have negative effects on the operation of the Common Market must be eliminated, and sets forth copyright law as the European Community's legal framework for the protection of computer programs.[130] The European Community's commitment to the promotion of international standardization is emphasized.[131] The prologue initially makes reference to the issues of idea/expression dichotomy, reverse engineering, limited rights of the owner of software to copy, and the copyrightability of logic, algorithms and programming languages (see discussion above). *32 The prologue establishes the term of protection for computer programs as the life of the author and 50 years from the authors' death.[133]

Article l—Object of Protection

Under article 1, computer programs are protected as literary works within the meaning of the Berne Convention for the Protection of Literary and Artistic Works.[134] Protection applies to the *expression* of a computer program. Underlying ideas and principles, including those that underlie its interfaces, are not protected by copyright under the directive.[135] According to the directive, a computer program is eligible for protection if it is original in the sense that it is the author's own intellectual creation. No other criteria are to be applied.[136]

125 Eric Schwartz, Policy Planning Advisor to the Register of Copyrights, personal communication Apr. 30, 1991.

126 Melville B. Nimmer, *Nimmer on Copyright (New* York, NY: Matthew Bender, 1988), sec. 17.01 [B].

127 For further discussion of the European Economic Community Treaty and the procedure by which the EC arrives at legislation such as the directive discussed in this chapter, see app. A.

128 Council Directive of May 14, 1991 on the legal protection of computer programs (91/250/EC).

129 Ibid,

130 Ibid.

131 Ibid,

132 Ibid.

133 Ibid.

134 Ibid., art. 1, sec. 1.

135 Ibid., sec. 2.

136 Ibid,, sec. 3.

Box 3-E—The Green Paper on Copyright and the Challenge of Technology: Copyright Issues Requiring Immediate Action

On June 7, 1988, the Commission of the European Communities issued a "Green Paper on Copyright and the Challenge of Technology--copyright Issues Requiring Immediate Action. "[1] This paper analyzes various issues concerning the copyright law, suggests legislative and technical solutions, and invites comments on the subjects discussed. Chapter 1 of the Green Paper considers the emergence of important copyright issues at the European Community level, the Community's general concerns about the state of copyright protection, and the Community's powers under the European Economic Community Treaty in relation to copyright goods and services.[2]

The Green Paper then addresses six focus areas:

1. piracy,
2. audio-visual home copying,
3. distribution and rental rights,
4. the legal protection of computer programs,
5. the legal problems and protection of databases and their operation, and
6. the role of the European Community in multilateral and bilateral external relations.[3]

This box highlights those sections relevant to the computer software industry.

Chapter **1: Copyright and the European Community**

Chapter 1 considers the emergence of important copyright issues at the Community level. The Commission expresses its concern that intellectual property has so far been dealt with by national law and has been neglected at the EEC level.[4] in the opinion of the Commission, the Community must provide for proper functioning of the Common Market to provide creators and suppliers of copyrighted goods and services with a single internal market.[5] The paper highlights the need to reconcile protection of the economic interests of the author and other creators, the promotion of ready access to information, and the pursuit of cultural goals.[6] Copyright law and policy are believed by the Commission to be means to pursue and accomplish these goals.[7] The Green Paper also outlines the growing importance of copyright to industry and commerce, as well as the importance of the market for goods and sew ices protected by copyright to the health of the European Community economy.[8]

The paper focuses on four fundamental concerns regarding copyright protection. First, the Commission states that it is important that the Community ensure the proper functioning of the Common Market, so that creators and providers of copyright goods and services are able to treat the Community as a single internal market. This would require elimination of obstacles and legal differences that disrupt the functioning of the market by obstructing trade and distorting competition. Second, the Community should, according to the Commission, develop policies to improve the competitiveness of its economy in relation to its trading partners. In addition to product-oriented measures, the paper suggests that the Community take legislative measures regarding intellectual property to ensure that European creators and firms can rely on legal protection for their products that is at least as favorable to their development as that granted by their principal competitors in their home markets. Third, steps must be taken to ensure that intellectual property resulting from creative effort and substantial investment with the Community is not misappropriated by non-EEC countries. The Commission believes that action should be taken by the Community to ensure a fair return from the exploitation of intellectual property by nonmember states. Finally, the interests of third parties and the public must be considered.[9]

---

1 Commission of the European Communities, ''Green Paper on Copyright and the Challenge of Technology—Copyright Issues Requiring Immediate Action,'' Communication from the Commission Brussels, 7 June 1988, COM (88) 172 final.

2 Ibid., pp. 10-18.

3 Ibid., chs. 2-7.

4 Ibid., p. 10

5 Ibid., p. 12.

6 Ibid., p. 10.

7 Ibid.

8 Ibid., pp. 11-12.

9 Ibid., pp. 12-14.

Chapter 5: Computer Programs

In Chapter 5 the Commission outlines the importance of the computer software industry to the Community's economy and industrial and technological future, and examines the present status of the computer industry in the Community.[10] Chapter 5 also addresses many problems encountered under the existing law applied to computer programs and urges that action be taken to provide for more consistent and effective protection.[11] In its conclusion, the Commission states its intention to submit a proposal and directive addressing the following issues:

1. whether copyright protection should apply to computer programs fixed in any form;
2. whether programs should be protected where they are original in the sense that they are the result of their creator's own intellectual efforts and are not commonplace in the software industry;
3. whether access protocols, interfaces, and methods essential for their development should be excluded from protection;
4. how broadly the use right should be formulated;
5. whether the adaptation of a program by a legitimate user exclusively for the users own purposes and within the basic scope of a license should be permitted;
6. whether reproduction, without authorization, of programs should be permitted for private purposes;
7. what the term of protection should be;
8. how authorship should be defined, including authorship of computer-generated programs;
9. whether protection should be available for creators who are nationals of States adhering to the Berne Convention or the Universal Copyright Convention or enterprises of such countries, or whether protection should be extended to all persons regardless of origin or domicile; and
10. upon which party the burden of proof should lie in infringement cases.[12]

Chapter 6: Databases

The Green Paper defines databases as ''collections of information stored and accessed by electronic means.[13] The paper points out that under certain conditions, 'compilations' are, at least in part, protected under the copyright laws, but electronic databases raise a number of technical and legal problems.[14]

The paper discusses two alternative solutions. First, it suggests legal action to protect the compilation of works within a database where those works are themselves the object of copyright protection.[15] The second alternative would be protection of databases composed of material which is not itself protected by copyright. The Commission suggests that the second option would only be exercised if it were felt that the considerable investment which a compilation of a database presents could best be served by copyright protection rather than by other means.[16]

Thus, the Commission considered the following issues:

1. whether the mode of compilation within a database of work should be protected by copyright; and
2. whether the right to protect the mode of compilation, in addition to possible contractual arrangements to that effect, should be extended to databases containing material not itself protected by copyright and whether this protection should be copyright or a right in general.[17]

Chapter 7: External Relations

One goal of the Commission is to improve the existing protection of intellectual property rights recognized by existing national legislation through the application of some of the general principles of the GATT (General Agreement on Tariffs and Trade) .18

10 Ibid., pp. 171-175.
11 Ibid., pp. 175-180.
12 Ibid., pp. 200-201
13 Ibid., p. 205.
14 Ibid., pp. 207-211.
15 Ibid., p. 211.
16 Ibid., p. 211.
17 Ibid., p. 216.
18 Ibid., p. 218.

Continued on next page

Box 3-E—The Green Paper on Copyright and the Challenge of Technology: Copyright Issues Requiring Immediate Action-Continued

The Commission suggests that all GATT member countries adhere to international conventions on intellectual property, e.g., the Paris Convention for the Protection of Industrial Property and the Berne Convention for the Protection of Literary and Artistic Works.[19] Further, the Commission has proposed, ***inter alia,*** that computer programmers should have exclusive rights to the use of their programs.[20] It has also proposed that semiconductor manufacturers should be given exclusive rights to the topography of the semiconductor.

The Commission also states that the application of "national treatment" and "most favoured nations treatment" would ensure that discrimination between national and foreign right holders and among foreign right holders themselves is avoided, both with regard to the substantive standards applied and the enforcement procedures and remedies.[21]

The Commission invited comments on:

1. the priorities to be given to the different aspects of reinforcement of intellectual property protection in the international context;
2. the development by (MIT of new disciplines as regards the effective enforcement of intellectual property laws, in particular, copyright and/or adoption of improved substantive standards; and
3. the more systematic use of bilateral relations, to ensure better protection in nonmember states of the intellectual and industrial property of Community right holders, particularly in the copyright field.[22]

19 Ibid., pp. 221-224.

20 Ibid., p. 223

21 Ibid.

22 Ibid., p. 236.

SOURCE: OTA, 1992.

Article 2—Authorship of Programs

Article 2 establishes the criteria for authorship under the Directive.[137] This article specifies the criteria to determine authorship in works created by individuals or groups and in collective works.[138] With respect to works-made-for-hire, the directive states that where a computer program is created in the course of employment, the employer is entitled to exercise all economic rights in the program, unless otherwise provided by contract.[139]

Article 3—Beneficiaries of Protection

Article 3 provides protection to all natural or legal persons eligible under national copyright legislation as applied to literary works.[140]

Article 4-Restricted Acts

Subject to the provisions of article 5 and 6, the author has the exclusive right to do or to authorize the permanent or temporary reproduction of a computer program by any means, in any form, in part or whole.[141] Insofar as loading, displaying, running, transmission or storage of the computer program requires a permanent or temporary reproduction of the program, such activities are subject to authorization by the right holder.[142] The author also has exclusive rights to translation, adaptation, arrangement, and any other alteration of a computer program and the reproduction of the results of these without prejudice to the rights of the person who alters the program.[143] The distribution of a computer

137 Ibid., art. 2. However, article 2 does not impose criteria with respect to authorship of works created by legal persons or as collective works. These questions remain regulated by the member states.

138 Ibid., art. 2, sec. 1. However, article 2 does not impose criteria with respect to authorship of works created by legal persons Or as collective works. These questions remain regulated by the member states.

139 Ibid., sec. 3.

140 Ibid., art. 3.

141 Ibid., art. 4.

142 Ibid.

143 Ibid.

program to the public, whether a copy or the original, is to be subject to right holder authorization.[144] This distribution right is exhausted, under the provisions of article 4, following the first sale of the program in the EC by the right holder or with his or her consent, with the exception of the subsequent rental of the software. 145

Article 5—Exceptions to the Restricted Acts

Article 5 provides that, in the absence of specific contractual provisions, the restricted acts of article 4 (a) and (b) will not require the authorization by the right holder where they are necessary for the use of the program by the person who lawfully acquired it in accordance with its intended purpose, including correction of errors.[146] Article 5 also provides that the right to make a backup copy by a person having a right to use a program cannot be contracted away.[147] Further, the person having a right to use a copy of a program shall be entitled, without authorization of the right holder, to observe, study, or test the functioning of the program in order to determine the ideas and principles which underlie the program, even if this is accomplished while loading, displaying, running, transmitting, or storing the program as provided for in article 5[148] or by the terms of his license. Read in conduction with the preamble, article 5 provides that if there is no license agreement, or if the license agreement is silent on the point, error correction is permitted. The license may also deal with error correction to regulate it, i.e., make an offer to provide correction service but not prohibit it entirely.

Article 6--Decompilation

Article 6 deals with the issue of recompilation.[149] Article 6 allows for reproduction of the code and translation of its form without the authorization of the owner (notwithstanding contractual provision to the contrary) when these activities are indispensable to achieve the interoperability of an independently created computer program, provided that certain conditions are met.[150] These conditions are that: 1) these acts are performed by the licensee or by another person who has the right to use a copy of a program, or on their behalf by a person authorized to decompile the program; 2) the information necessary to achieve interoperability has not been made readily available to these persons; 151 and 3) these activities are confined to the parts of the original

---

144 Ibid.

145 Ibid.

146 Ibid., sec.1. Some commentators believe that this provision reflects that European law may be more consumer-oriented in character than U. S. law, which might require an option of either provision of error-correction or an implied license to modify the copyrighted code. They state that in the United States there is a court-created doctrine of 'implied license" in patent law not necessarily wholly displaced by ''fair use' in copyright law. Domestic courts tend to decide controversies over mass-distributed software resident on floppy disks under the Uniform Commercial Code (UCC) article 2, which have some lesser consumer protection provisions. Efforts arc now underway under the direction of the National Conference of Commissioners on Uniform State Laws to consider a uniform software licensing act and in committees within the Massachusetts Bar Association and the Licensing Executives Society to develop a UCC article 2B addressed to licensing of intellectual property rights generally. These are not expected to add significantly to consumer protection. Stephen Y. Chow, Cesari and McKenna, personal communication, Sept. 27, 1991.

147 Ibid., sec. 2.

148 Ibid., see. 3.

149 Ibid., art. 6. The term ''dccompilation'' is a matter of some debate within the EC. Some observers view decompilation as one aspect of the larger process of reverse engineering, and assert that the directive broadly allows 'recompilation. ' Representatives of Bull S. A., and the European Committee for Interoperable Systems, personal communication, June-July 1991.

Others believe that the word 'recompilation'' is inappropriately used, that since the term 'recompilation'' is not defined in the Directive no meaning can be attributed to it other than that it covers only those acts covered specifically in article 6 of the directive, irrespective of whether the word is used with other broader meaning in other contexts. These observers argue that the term ''recompilation' might well be eliminated, as member states are required only to transpose the substance of the directive to create the same legal effect in national legislation as that intended in the directive in order to fulfill the implementation requirement. Indeed, member nations are not required to adopt any particular terminology and, in particular, not obliged to take any particular topic heading. Representatives of IBM Europe, personal communication, June-July 1991. For further discussion of the question of decompilation, see ch. 4.

150 Ibid., sec. 1.

151 The language ''made readily available'' is interpreted differently by different stakeholders. Some parties would assert that the language was initially included to preclude publication of the code in an obscure language and location and to then maintain that the code was therefore ''available' or "public," However, debate continues whether code that is made available for a negotiated price is considered "readily available. " Further, observers question whether code is ' 'readily available' when parties wishing to decompile must first consult with software developers to obtain code, disclosing the purpose and nature of their request. Still others assert that it was made clear by the Commission to the Council in December 1990 that the right holder and the would-be decompiler could enter in to a dialog on the possible supply of information with or without payment. Representatives of IBM Europe, Bull S. A., the European Committee for Interoperable Systems, personal communications, June-July 1991.

program which are necessary to achieve interoperability.[152]

Information gained through reproduction of the code under article 6 may not be used for goals other than to achieve interoperability of the independently created program.[153] It may not be given to others, except when necessary for interoperability of the independently created program.[154] It may not be used for the creation or marketing of a program which infringes the copyright of the original progarm.[155]

Article 6 also recognizes that the article, in accordance with the provisions of the Berne Convention, may not be interpreted so as to allow its application to be used in a manner which unreasonably prejudices the right holder's legitimate interests and conflicts with a normal exploitation of the computer program.[156]

Article 7--Special Measures of Protection

Under article 7, member states are required to provide, without prejudice to articles 4, 5, and 6, appropriate remedies against persons committing any of the following acts:

1. placing in circulation a copy of a computer program, knowing, or having reason to believe, that it is an infringing copy;
2. possessing a copy of a computer program for commercial purposes, knowing, or having reason to believe, that it is an infringing copy;
3. putting into circulation or possessing for commercial purposes any means, the sole intended purpose of which is to facilitate the unauthorized removal or circumvention of any technical device which may have been applied to protect a computer program.[157]

An infringing copy of a computer program is liable to be seized pursuant to the individual member state's legislation.[158] Member states may provide for seizure of any means described above,[159]

Article &Term of Protection

Article 8 establishes the term of protection as the life of the author plus 50 years after the author's death. Where the computer is an anonymous or pseudonymous work, the term of protection shall be 50 years from the time that the computer program is first lawfully made available to the public. The term of protection is deemed to begin on the first of January of the year following these events.[160] Member states, which presently have a term of protection longer than this may maintain that term

---

152 Ibid, sec. 1, subsec. (a).(c). This provision of the directive was the focus of particular controversy. The previous draft to the directive, ''Amended proposal for a Council directive on the legal protection of computer programs" COM(90) 509 final—SYN 183; (90/C 320/1 1), (art. S(a), sec. 2), read:

> [T]hese acts are confined to the parts of the original program which are necessary to achieve interoperability *within it*. (Emphasis added.)

Thus, the final version of the Council Directive deletes the final two words of the provision ''with it. "

The Commission further commented on this aspect of the directive in the Commissioner's Communication to the European Parliament, 2,1, Sec. 4.g.2). In that document, the Commission stated:

> A particularly important question was to determine the extent to which the decompiling of a program without the copyright holder's authorization would be possible. The solution adopted in the common position was that decompiling was permitted in so far as it proved necessary for the interoperability of a computer program created independently. Decompiliation is permitted to the extent necessary to ensure the interoperability of an independently created computer program. Such a program may connect to the program subject to recompilation. Alternatively it may compete with the decompiled program and in such cases will not normally connect to it. Article 6 does not however permit interoperability of the independently created program. It cannot therefore be used to create a program reproducing parts of a decompiled program having no relevance to the interoperability of the independently created program.

While there is some agreement that a programmer can isolate the critical sections of code needed to accomplish interoperability by viewing externals, running the program on a display screen and looking at the code, the amount of program necessary is subject to debate. Courts will likely be required to judge the appropriateness of the use of sections and amounts of the code.

153 Ibid., subsec. (a).

154 Ibid., subsec. (b).

155 Ibid., subsec. (c).

156 Ibid., sec. 3.

157 Ibid., art. 7, sec. 1 (a)-(c).

158 Ibid., sec. 2.

159 Ibid., sec. 3.

160 Ibid.. art. 7.

until the term of protection for copyrighted works generally is harmonized by EC law.[161]

Article 9—Continued Application of Other Legal Provisions

Article 9 ensures that the provisions of the software directive do not prejudice any other legal provisions,[162] and provides that the provisions of the directive are applicable to programs created prior to January 1, 1993 without prejudice to any acts concluded and rights acquired before that date.[163] Contractual provisions contrary to article 6 or to the exceptions provided for in article 5(2) and (3) are nullified by this provision. [164]

Article 10—Final Provisions

Under article 10, member states are required to bring into force the laws, regulations, or administrative provisions needed to transpose the directive by January 1, 1993.[165] Member states are to inform the Commission of the provision of national law which they adopt pursuant to the directive. [166]

Article 11

Article 11 addresses the directive to the member states. [167]

161 Ibid., art. 8, sec. 2.

162 Ibid., ch. II, art. 8, sec. 1.

163 Ibid., sec. 2.

164 Ibid,, sec. 1. For discussion of U.S. law dealing with use of contractual agreements to protect intellectual property, see ch. 2.

165 Ibid., art. 9, sec. 1.

166 Ibid., art. 9, sec. 2.

167 Ibid., art. lo.

Chapter 4

Software Technology and the Law

Contents

	Page
Introduction	125
Technology	125
Introduction	125
Program Function	126
External Design	126
User Interface	127
Program Code	130
Software and the Application of Intellectual Property Laws	130
Program Function	132
External Design	138
User Interface	142
Program Code	144
Recompilation	146
Introduction	146
Recompilation and Disassembly	147
Uses of Recompilation	148
Other Methods of Reverse Engineering	150
Legal Arguments for Policy Positions	150
Patent Law	150
Copyright Law	151
Software Development	153

Boxes

Box	*Page*
4-A. Authorship	131
4-B. Cryptography	137
4-C. Neural Networks	152
4-D. Software Reuse	154
4-E. Special Concerns of the Federal Government	156
4-F. The Discipline of Computer Science	158

Figures

Figure	*Page*
4-1. Comparison of ''Substituting" and ''Attaching'' Programs	127
4-2. High-Level Language, Machine Language, and Disassembled Versions of a Program	149

Chapter 4
Software Technology and the Law

Introduction

There are intellectual property issues associated with four elements of a program: the program function, the external design, the user interface design, and the program code. The first section of this chapter describes the technology behind each of these elements. The second section outlines the application of existing intellectual property laws to each element, and discusses the policy issues associated with the current level of protection. There have been various policy positions advanced for maintaining or changing the scope of protection, and most of these policy positions have been supported by legal arguments; the final section of the chapter briefly summarizes these legal arguments.

Technology

Introduction

One way to think about computers and programs[1] is to look at the hardware. The core parts of the computer are the processor and memory. Both the processor and memory usually consist of one or more *integrated circuits,* which are semiconductor chips that manipulate digital electronic information. The processor and memory work together to perform logical and arithmetic operations on data; the program is stored in memory and specifies the order in which the operations are to be performed.

A program consists of a list of instructions. Each type of processor has an instruction set—a set of operations that it is capable of performing. Most of these operations are simple; for example, a typical instruction set would include an instruction for operations such as moving data from memory to the processor, logical operations such as checking if two pieces of data have the same value, and arithmetic operations such as adding two numbers. A program is executed when the instructions are transferred to the processor and the processor performs the specified operations.

Inside the computer, ‘‘instructions’ and ‘‘data’ arc both patterns of electronic signals. These signals can take one of two values; to make it easier to comprehend what is happening inside the computer, programmers represent one of the values with the symbol” 1,’ and the other with the symbol ‘‘ O.” For example, the addition instruction for the processor that is used in most microcomputers may be represented as “00000100.” In the same way, in most computers the letter “A’ is represented by the pattern of signals corresponding to “01000001. ” The complex functions that programmed computers perform for users often seem far removed from the patterns of electronic signals and very simple operations that characterize the computer at the hardware level. Computers perform complex tasks by performing a large number of simple operations in sequence—typically millions of operations per second.

The processor and memory are usually part of a larger system. Data to be used in a computation may be read from a disk or tape drive. Data can also be exchanged or shared with other computers using a network. The data is exchanged using *communications protocols,* which specify the format and sequence in which data is sent. It is also possible for part of a computation to be carried out on a distant computer. Sometimes specialized computers are used for different parts of a task; for example, a supercomputer may carry out the computationally intensive portions of a task, while a workstation is used for displaying the results of the computation.

There are also a variety of input and output devices for communicating with the user. The display is the output device most commonly used for providing users with information about the results of a computation. More advanced displays, faster processors, and cheaper memory enable program developers to go beyond the simple display of text to include graphics. Color monitors are also increasingly common. In some applications, small displays inside helmets or goggles are used to give users the

[1] For an introduction to computers and computer science, see Alan W. Biermann, *Great Ideas in Computer Science* (Cambridge, MA: MIT Press, 1990).

illusion of a three-dimensional image.[2] Sound, which can include warning tones, music, and synthesized speech, may also be used to provide information to users.

Input devices include the keyboard, for entering text, and the *mouse,* a pointer used for drawing figures or selecting commands on a screen. The use of a special pen can also allow the entry of written information and commands.[3] Research is under way on speech recognition technologies that allow commands to be spoken instead of typed or selected using a mouse. Other experimental input devices detect eye movements[4] or gestures made with a special glove.

Sometimes the memory and the processor are not part of a conventional computer, but are *embedded in* industrial machinery and other devices. The processor receives information from a variety of sources, processes the data, and uses the results of the computation to direct the operation of a machine. Examples of embedded systems are the microprocessor-based controllers used in appliances, automobiles, and industrial processes.

Program Function

Programmed computers perform a series of calculations to transform input values to output values. A well-defined computational procedure for taking an input and producing an output is called an *algorithm.*[5] Algorithms are tools for solving computational problems. The statement of a problem specifies in general terms the desired relationship between the input and the output, while the algorithm describes a specific computational procedure for achieving the input/output relationship.[6]

The transformation of input data to output data can also be performed in hardware. Integrated circuits can perform the same simple logical and arithmetic operations that programmed computers perform. Connecting together these electronic circuits has the same effect as programming a computer. Just as the calculation that the computer will perform can be understood by looking at the program, the calculation that a circuit will perform can be understood by looking at the circuit diagram.

The choice of whether to perform the calculation by programming a computer or building a circuit is an engineering decision. Often, a calculation can be performed more quickly by hardware, which may be an important consideration in some applications such as signal processing. On the other hand, programming a computer is potentially less costly and more flexible. The function of a programmed computer can be changed by writing a new program; with hardware, however, a new circuit must be built.

Often many different problems can be modeled in a similar way, and solved using the same class of algorithms. For example, many applications that operate on speech signals and video images use similar signal processing algorithms. Searching and sorting algorithms are also among the basic tools that are commonly used in software development. Problems such as finding the fastest route between two cities or determining when to perform tasks in a manufacturing process may be modeled in a way that makes them solvable by using graph algorithms.

External Design

Programs have an external design or interface—the conventions for communication between the program and the user or other programs. The external design is conceptually separate from the program code that implements the interface (the internal design). It specifies the interactions between the program and the user or other programs, but not how the program does the required computations. There are typically many different ways of writing a program to implement the same interface.

---

[2] Scott S. Fisher, "Virtual Interface Environments," in Brenda Laurel (cd.), *The Art of Human-Computer Interface Design* (Reading, MA: Addison-Wesley, 1990), p. 423.

[3] Robert M. Carr, "The Point of the Pen," Byte *Magazine,* vol. 16, No. 2, February 1991, p. 211.

[4] Robert J.K. Jacob, "What You Look Is What You Get," *Proceedings of CHI (Conference on Human Factors in Computing Systems), 1990 (New* York, NY: Association for Computing Machinery, 1990), pp. 11-18.

[5] The definition given here reflects the use of the term 'algorithm' in computer science. In applying patent law to inventions involving programmed computers, the courts are required to determine whether the claimed invention is a "mathematical algorithm." The term "mathematical algorithm' was used by the Court of Customs and Patent Appeals to characterize a method of converting binary coded decimal to binary numbers that the U.S. Supreme Court held to be nonstatutory in its 1972 decision *Gottschalk* v. *Benson (409* U.S. 64). The meaning of "mathematical algorithm," and the relationship between 'algorithm' as the term is used in computer science, and "mathematical algorithm," as the term is used in the case law, has been the subject of considerable discussion (see pp. 133-134).

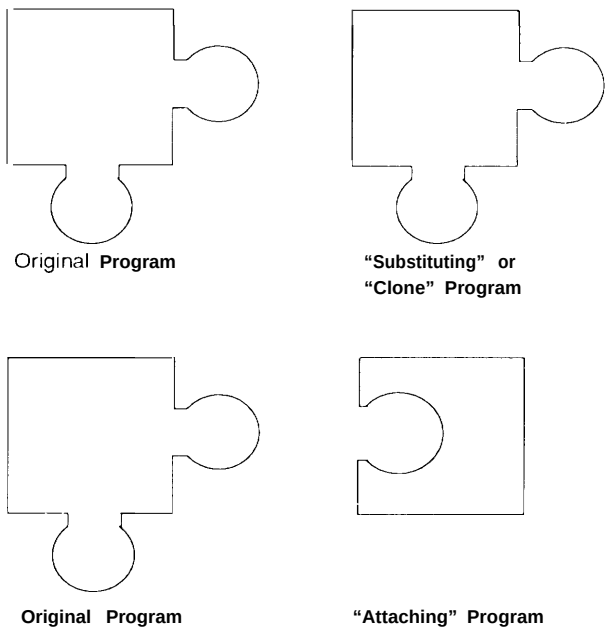
[6] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest, *Introduction to Algorithms* (Cambridge, MA: MIT Press, 1990).

The external design will sometimes reflect constraints such as the speed of the processor, the amount of memory available, and the time needed to complete the product. In addition, the process of developing software is iterative—the external design is refined as testing reveals more about user needs or constraints on the implementation.[7]

One example of an external design is the user interface, the conventions for communication between the user and the program. There are also interfaces between programs, such as the ''operating system calls' applications programs use to access functions provided by the systems software of a computer. Communications protocols and the specifications of procedures are other examples of interfaces.

The discussion of appropriate intellectual property protection for interfaces often involves the use of terms such as ''open systems, ' ''interoperability, ' or ' 'compatibility. These terms sometimes have ambiguous meanings.[8] They may be used to describe a situation in which a program from one vendor is able to exchange information with a program from a different vendor. However, these terms are also sometimes used to describe a situation in which multiple vendors offer a product with the same external design. Each of these meanings of ' 'compatibility' implies a different economic effect. For this reason, participants in the software debate sometimes distinguish between ' 'attaching" programs, which are able to exchange information with a program written by a different vendor, and ''substituting programs, which have the same external design (see figure 4-1). ' 'Substituting' programs are sometimes referred to as "workalike" programs or ' 'clones.

Figure 4-l—Comparison of "Substituting" and "Attaching" Programs



SOURCE: OTA.

User Interface

The user interface specifies the interactions between the program and the user. There are a number of different kinds of user interfaces. A programming language is in a sense a user interface—programming using conventional programming g languages is one way to direct a computer to perform a task. For most people this style of interaction is too difficult and inconvenient. By using new technologies, however, different ways of using computers have been developed, Sophisticated, but easy-to-use, user interfaces have created new markets where there are end users who are not programmers.[9] Command languages, menu-based dialogs, graphical user interfaces, and newer interaction techniques have expanded the design choices available to user interface designers.

One interaction style[10] is the *command language* dialog, in which the user issues instructions to a computer through a typed command.[11] The Unix and DOS operating systems usually have this type of user interface. For example, the command for

[7] Daniel S. Bricklin, President, Software Garden, Inc., testimony at hearings before the House Subcommittee on Courts, Intellectual Property, and the Administration of Justice, Nov. 8, 1989, Serial No. 119, p. 221,

[8] ''What Does 'Open Systems' Really Mean?'' *Computerworld*, vol. 25, No. 19, May 13, 1991, p. 21.

[9] Jonathan Grudin, ' 'The Computer Reaches Out, ' *Proceedings of CHI (Conference on Human Factors in Computing Systems), 1990* (New York, NY: Association for Computing Machinery, 1990), pp. 261-267,

[10] For a discussion of different interaction styles, see Ronald M. Baecker and William A.S. Buxton, *Readings in Human-Computer Interaction* (San Mateo, CA: Morgan Kaufmann, 1987), p. 427,

[11] Ibid., p. 428,



Photo credit: Library of Congress

An early user interface. Dr. J. Presper Eckert, Jr. examines the control panel of the ENIAC computer at the University of Pennsylvania in 1946.

deleting a file when a computer is using the Unix operating system is ''rm. ' The difficulty with this type of interface is that it is hard to learn. It may be difficult to remember the available commands, their exact syntax, and how they can be strung together to perform more complex tasks.[12]

A second type of interdiction style uses a *menu.* Instead of having to remember the commands, the user can select choices from a list of alternatives displayed on a screen.[13] The selection can be made by pressing a key that corresponds to a menu option, or by moving the cursor on the screen until the option is highlighted. Menu-based interfaces were made possible by the development of hardware technology that allowed large amounts of information to be quickly displayed on a screen.[14]

Newer interfaces are graphical, using images, in addition to text, to display information to users. *Icons* represent operations (much as menu options do) or data. Commands can be issued by *direct manipulation*[15]*:* instead of using a command language to describe operations on objects, the user ''manipulates' objects visible on the display.[16] The effect of the action is immediately apparent to the user. For example, a user could point to an icon representing a file with the mouse and then ask the system to delete the file; the icon could then disappear from the screen to show the user that the file has been deleted. Graphical user interfaces often allow users to view several activities simultaneously on the screen, through the use of *windows* that subdivide the screen area.

The Design Process

The user interface designer makes many design decisions.[17] Technological change is adding to the range of available choices-color, graphics, sound, video, and animation are only beginning to be explored or widely applied.[18] However, the user interface designer must also work within a set of constraints. Some of these constraints are imposed

12 Bill Curtis, "Engineering Computer 'Look and Feel': User Interface Technology and Human Factors Engineering, " *Jurimetrics,* vol. ***30, No.*** 1, fall 1989, p. 59.

13 Donald A. Norman, ' 'Design Principles for Human-Computer Interfaces, ' *Proceedings of CHI (Conference on Human Factors in Computing Systems), 1983 (New* York, NY: Association for Computing Machinery, 1983), p. 9.

14 John Walker, ''Through the Looking Glass, '' in Brenda Laurel (cd.), *The Art of Human-Computer Interface Design* (Reading, MA: Addison Wesley, 1990).

15 Ben Shneiderman, ' 'Direct Manipulation: A Step Beyond Programming Languages, " *IEEE Computer,* vol. 16, No. 8, August 1983, pp. 57-69.

16 Robert J.K. Jacob, ' 'Direct Manipulation, " in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (New* York, NY: Institute of Electrical and Electronics Engineers, 1986), pp. 384-388.

17 "Unless one clones an existing product, designing even one aspect of an interface—menu navigation, window operations, CO remand names, function key assignments, mouse button syntax, icon design, etc.—gives rise to a potentially endless series of decisions. " Jonathan Grudin, op. cit., footnote 9, p. 261.

18 Ibid.

by the needs of the user, and some are imposed by hardware or software capabilities.[19]

In developing a program, a developer decides which functions the program is to perform for the user. The user interface design reflects the designer's efforts to communicate this functionality to the user. Designing user interfaces is a "communications task like writing or filmmaking."[20] The user interface helps the user develop a ' 'mental model" of how the program works. This mental model does not necessarily reflect the internal 'engineering model' of the program.[21] For example, when a user Points with the mouse to an icon representing a file and asks the computer to delete the file, the user does not have to know where the file is stored or how the hardware and software perform the operation.

One component of the user interface design process is the choice of the interaction style. This choice is influenced by hardware and software constraints, the nature of the application, and the characteristics of the end user. For example, a command language interface may not be appropriate for users who use a program infrequently; a menu interaction style would be easier to use because it would provide the users with reminders of the available commands.

Another component of the design process requires that the functionality of the application be represented within the interaction style. This stage of design would include, for example, the choice of commands and their representation by icons or appropriate mnemonics. This stage of the design reflects the designer's judgment of how the user would want to accomplish certain tasks.[22] For example, the assignment of commands to various menus could reflect a judgment of the relative importance of each command.

Constraints

There are a number of constraints that the user interface designer must take into account. Hardware or software constraints may limit solutions that require too much processing power or are too time-cons uming to program. The capabilities of users present other constraints. Research in the field of human-computer interaction (HCI) tries to find a scientific base for understanding what makes a user interface design successful.[23] HCI research also focuses on methodologies for developing better interfaces.

One of the most important user interface design principles is that ' 'consistency' is important.[24] One type of consistency is internal consistency .25 Internal consistency means, for example, that operations common to several objects in a program have the same results on all of the objects. For example, there might be a single ''delete' command that deletes the selected object, whether it is a text string, a curve, or a file.[26]

From the perspective of intellectual property law, the most significant type of consistency is *external consistency* with the features of other interfaces, i.e., similarity of a given interface to those of other applications and systems. A system is said to be ''backwards compatible'' if it is compatible with an older version of the system, allowing users to benefit

[19] ''[Solutions are shaped by a multitude of problems that arc invisible to those outside of the design process. A wonderfully intuitive solution doesn't matter if the system architecture doesn't support it, or if the resulting code takes up too much memory or runs too slowly. Other problems stem from the basic capabilities of humans, and the requirements of tasks users wish to do. It doesn't matter if the interface responds instantly, if the user can't usc it. Solutions to an interface problem involve compromise. But how do designers determine what an acceptable compromise is? How do designers figure out acceptable tradeoffs between speed and intuitiveness and other seemingly contradictory values and requircmcnts? Thomas D. Erickson, ''Creativity and Design, ' m Brenda Laurel (cd.), *The Art of Human-Computer Interface Design* (Reading, MA: Addison-Wesley, 1990), p. 3.

[20] Paul Heckel, *The Elements {$ Friend/) Software Design* (Alameda, CA: Sybex, 199 1), p, xix. The Epilogue describes the author's experiences in applying for and enforcing a ' 'software patent.''

[21] Donald R. Gentner and Jonathan Grudin, '' Why Good Engineers (Sometimes) Create Bad Interfaces,' *Proceedings of CHI (Conference on Human Factors in Computing Systems), 1990 (New* York, NY: Association for Computing Machinery, 1990), p. 277.

[22] ' '... designers who have only a sketchy or partial understanding of users' tasks will tind it difficult to appreciate the dominant role tasks should play in interface design. in the absence of task analysis, the designer has little to go on and it *bccomcs* convenient to focus on properties of the interface. ." Jonathan Grudin, ' 'The Case Against User Interface Consistency', " *Communications of the ACM, vol.* 32, No IO, October 1989, p. 1165.

[23] Stuart K. Card and Thomas P. Moran, ' 'User Technology: From Pointing to Pondering, ' in Adele M. Goldberg (cd.), *A History of Personal Workstations (New* York, NY: ACM Press. 1988), p. 493.

[24] Ben Shneiderman, *Designing the User Interface* (Reading, MA: Addison-Wesley, 1987).

[25] Grudin, op. cit., footnote 22, p. 1165.

[26] Butler W. Lampson, ''Personal Distributed Computing, '' in Adele M. Goldberg (cd.), *A History of Personal Workstations (New* York, NY: ACM Press, 1 988), p 321

from new features or better performance without having to learn a completely new system. Interoperability and backwards compatibility requirements reflect aspects of the users' experience and environment that should be reflected in an interface design. They are increasingly important as computers are used by more people and in different application areas .27

Program Code

The program code is the implementation of the fiction and external design (including the user interface) of the program. Much effort has been devoted to developing new tools and methods for coping with the complexity of developing new software systems. In general terms, ''software engineering' is concerned with the practical aspects of developing software. It overlaps all of the various specialities of software research, including programming languages, operating systems, algorithms, data structures, databases, and file systems.[28]

The use of high-level programming languages, such as FORTRAN, C, and Pascal, is an important element in the development of complex programs. High-level languages are more powerful than machine languages-each statement in a high-level language usually does the same job as several machine language instructions. In addition, programmers need not be as concerned with the details of the computer's operation, and can write the program in a more natural way. For example, the Pascal-language statement "quantity := total - 5" does the same job as a series of machine language statements; the names of the *variables,* such as "total,' may also suggest something about why the operation is being done. Finally, the programmer does not need to know where the data is stored in memory or how it is represented by patterns of electronic signals.

Programs written in a high-level language must be translated into machine language for execution. This process is called "compilation." The high-level language version of a program is often referred to as ''source code, ' while the machine language version is referred to as "object code. " In the software debate, the relationship between source code and object code has been the subject of considerable discussion (see box 4-A and the "Recompilation" section later in this chapter).

High-level languages also encourage the construction of *procedures, data abstractions,* or *objects,* which allow the decomposition of large and complex programs into smaller pieces that can be attacked independently. Much of the internal structure of a program is due to decisions made by the programmer about how best to decompose the larger problem into smaller pieces. Breaking a larger problem into smaller, more manageable pieces is not unique to software:

> Well-designed computational systems, like well-designed automobiles or nuclear reactors, are designed in a modular manner, so that the parts can be constructed, replaced, and debugged separately.[29]

An example of a procedure might be a sequence of instructions for sorting numbers. This sequence of instructions could be a procedure called "Sort.' "Sort" in effect becomes a new instruction that the programmer can use as if it were an ordinary instruction. At every point in the program where it is necessary to sort some numbers, the programmer simply uses the new instruction, without worrying about the details of its implementation. The programmer only needs to know about the interface that specifies the name of the procedure, its function, and the format in which it exchanges information with other parts of a program.[30]

Software and the Application of Intellectual Property Laws

This section discusses the policy issues associated with the four elements of software outlined in the previous section—program function, external design, the user interface, and the program code. For each of these elements, the section outlines the courts' current approach to its protection, and then discusses arguments for maintaining or modifying the level of protection. To the extent possible, this

---

27 Grudin, op. cit., footnote 22, p. 1171.

28 Computer Science and Technology Bored, National Research Council, *Scaling Up: A Research Agenda for Software Engineering* (Washington, DC: National Academy Press, 1989), p. 18.

29 Harold Abelson and Gerald Sussman, *Structure and Interpretation of Computer Programs* (Cambridge, MA: MIT Press, 1985), p. 2.

30 ''Good engineers distinguish between *what* a component does (the abstraction seen by the user) and *how* it does it (the implementation inside the black box). " Jon Bentley, *Programming Pear/s* (Reading, MA: Addison Wesley, 1986), p. 134.

Box 4-A—Authorship

The copyright clause of the U.S. Constitution permits Congress to grant ''authors' exclusive rights to their ''writings.' Before the current Copyright Act and 1980 software amendments became law, there was considerable disagreement as to whether programs were copyrightable "writings." Even after the 1980 amendments were enacted some believed that only high-level language (or ''source code") programs were copyrightable subject matter, while machine language (or 'object code' programs were not protected by copyright law. Some arguments centered on whether code in lower-level languages was *human-readable;* according to one view, only higher-level languages expressed writings (for human readers) eligible for copyright protection.[1]

A corollary debate concerns requirements for ''authorship' of programs. There were two issues: 1) whether "original works of authorship' required a human author, or whether machine-generated works could also be eligible for copyright—the Copyright Office has maintained that the term ''authorship' in the Copyright Act implies a human originator and 2) determining authorship and copyright ownership for machine-generated or machine-assisted works.

Questions of machine authorship have arisen with respect to compilers.[3] When a program is compiled, some information is removed (e.g., comments), some information is added, and the code may then be rearranged to optimize execution speed. From a technological perspective, the end result of the compilation process could therefore be regarded as a ''derivative work' based on the source code program. However, the Copyright Office does not view the object code version as containing sufficient "originality' to be a derivative work, For this reason, the Copyright Office takes the position that the source-code and object code versions of a computer program are copies of the same work.[4]

The final report of the National Commission on New Technological Uses of Copyrighted Works (CONTU) addressed the question of computer-generated works and concluded that ''no special problem exists. The issue continues to be addressed, most recently at a symposium sponsored by the World Intellectual Property Organization (WJPO) on the intellectual property aspects of artificial intelligence. One conferee expressed the view that the real issue was not whether there is a human author, but rather who that author is. While CONTU had concluded that the computer was just a tool to assist a human being in creating a work, the conferee suggested that advances in artificial intelligence meant that the tools were becoming increasingly sophisticated, perhaps indicating a need to apportion authorship among the user and the author of a program used in creating a work.[6] A second participant expressed similar views when discussing the authorship of programs generated by a sophisticated "code generator" from functional specifications.[7]

---

1 The rule that a work must be readable *by a* human *audience* had its origins in *White-Smith Music Publishing CO.v. Apollo Music Co.*, *209* U.S. 1 (1908), which ruled that player-piano rolls could not be copyrighted. For a discussion of the readability requirement, see "Copyright Protection of Computer Program Object Code," Harvard Law *Review*, vol. 96, May 1983, pp. 1723-1744; Christopher M. Mislow, "Computer Microcode: Testing the Limits of Software Copyrightability,' *Boston University Law Review*, vol. 65, July 1985, pp. 733-805; and the dissent of Commissioner John Hersey in National Commission on New Technological Uses of Copyrighted Works (CONTU), *Final Report* (Washington, DC: Library of Congress, July 31, 1978).

2 Cary H. Sherman, Hamish R. Sandison, and Marc D. Guren, *Computer Software Protection Law* (Washington, DC: The Bureau of National Affairs, 1989, 1990), 204.3(d).

3 Ibid., *208.2(b)(5),* and discussion at OTA workshop on "Patent, Copyright and Trade Secret Protection for Software," June 20, 1991.

4 ''The Copyright Office considers source code and object code as two representations of the same computer program. For registration purposes, the claim is in the *computer program* rather than in any particular representation of the program. " Computer Program Practices, *Compendium II, Copyright Office Practices,* Section 321.03.

5 CONTU, op, cit., footnote 1, p. 46.

6 Arthur R. Miller, ''Computers and Authorship: The Copyrightability of Computer-Generated Works, '' in *WIPO Worldwide Symposium on the Intellectual Property Aspects of Artificial Intelligence,* WIPO Publication No. 698 (E) (Geneva, Switzerland: World Intellectual Property Organization 1991) , p. 241.

7 Robert Barr, "Computer-Produced Creations, " in *WIPO Worldwide Symposium on the Intellectual Property Aspects of Artificial Intelligence,* WIPO Publication No. 698 (E), p. 225.

SOURCE: OTA and cited sources.

section separates the question of the appropriate level of protection from the question of how existing patent, copyright, and trade secret laws should be interpreted. In the software debate, policy arguments for a certain level of protection are often characterized as consistent with a ''proper' interpretation of existing law. Legal arguments made in support of policy positions are outlined in the last section of the chapter.

Program Function

Intellectual Property Protection of' Program Function

Existing intellectual property laws are applied to program function in several ways. First, it can be argued that the program function is protected against copying because the implementation is protected by copyright law. Copyright law prevents others from acquiring the functionality of the programmed computer if it is obtained by copying the "expression' in the program code. However, copyright law does not prevent the independent development of a program that performs the same function,

Some aspects of the function of a programmed computer may be protected by maintaining them as trade secrets. For example, a program may be distributed with contractual restrictions on the extent to which it may be studied. Trade secret protection may also be maintained (in part) by distribution of the program in machine language, which is difficult for competitors to understand. This may be a valuable form of intellectual property protection; the scope of protection is the subject of the ' 'decompilation" debate discussed later in this chapter.

Finally, parts of the function of a program may be protected by patent law. The same program may embody many patentable inventions, or none at all, depending on how many parts of the program function are novel, nonobvious, and statutory subject matter. The inventions are claimed either as a series of functional steps carried out by the computer or as a system capable of performing certain functions. The U.S. Patent and Trademark Office (PTO) emphasizes that patents are granted for the functional steps or the system, not the program code .31

When the invention is being claimed as a series of functional steps, or *process, the* applicant does not specify each machine language operation carried out by the computer.[32] Instead, the claim usually describes the steps at a higher level of abstraction, independent of a specific implementation. For example, one patent recites the steps of:

> Identifying a plurality of overlapping working areas on said screen, associating each said working area with an independent computer program, selectively communicating data to each said program. . ,[33]

If, on the other hand, the invention is being claimed as a system or *apparatus, the* applicant describes the 'means' for performing the functions. For example, the apparatus claim corresponding to the process claim described above specifies:

> A computer terminal display system comprising a display surface, means for simultaneously displaying a plurality of overlapping rectangular graphic layers, . . . means for associating each of said graphic layers with an independent computer program. . .

The basis for claiming a software-related invention as an apparatus is that the programmed computer becomes anew machine, or at least a' 'new and useful improvement' of the unprogrammed machine.[34] In an early case that addressed the question of the patentability of software-related inventions, the Court of Customs and Patent Appeals (C. C.P.A.) wrote:

> If a machine is programmed in a certain new and unobvious way, it is physically different from the machine without that program; its memory elements are differently arranged. The fact that these physical

31 ''Recently some commentators have stated that the Office is issuing patents on computer programs or 'Software. ' This is not the case. A 'computer program' is a set of statements or instructions to be used directly or indirectly in a computer to bring about a certain result. A computer program is different from a 'computer process' which is defined as a series of process steps performed by a computer. This distinction may become blurred because some refer to both the series of process steps performed by the computer and the set of statements or instructions as computer programs.'' Jeffrey M. Samuels, Acting Commissioner of Patents and Trademarks, testimony at hearings before the House Subcommittee on Courts, Intellectual Property, and the Administration of Justice, Mar. 7, 1990, Serial No. 119, p. 334.

32 John P. Sumner, "The Copyright/Patent Interface: Patent Protection for the Structure of Program Code, " *Jurimetrics*, vol. 30, No. 1, fall 1989, p. 118.

33 U.S. Patent No. 4,555,775. In order to satisfy the ' 'enablement'' requirement of section 112 of the Patent Act, the specification would show in more detail how the process steps recited in the claim would be performed.

34 *In re Bernhart*, 417 F.2d 1400 (C'. C.P.A. 1969).

changes are invisible to the eye should not tempt us to conclude that the machine has not been changed.[35]

Statutory or Nonstatutory--Novel and nonobvious program function, whether claimed as an apparatus or process, is not necessarily ‘ ‘statutory subject matter for which patents may be granted. In determining whether a claimed computer-related invention is statutory, patent examiners apply the Freeman-Walter-Abele test, outlined in PTO guidelines issued in 1989.[36] This test is named after the appeals court decisions that contributed to developing and refining the test. The purpose of the Freeman-Walter-Abele test is to determine whether a claimed invention is a nonstatutory ‘‘mathematical algorithm’ or mere calculation. [37]

The Freeman-Walter-Abele test applies, in the context of computer-related inventions, the patent law doctrines that regard scientific principles, abstract ideas, and mathematics as nonstatutory. In the 1972 case, *Gottschalk v. Benson,*[38] the U.S, Supreme Court held that a method of converting binary coded decimal (BCD) numbers to binary numbers was not statutory. In C. C.P.A. decisions that followed Benson this method was characterized as a ‘ ‘mathematical algorithm. ’ The Supreme Court objected to the fact that the claimed process just converted numbers from one representation to another without applying the result of the calculation to any other task. Just as a law of nature by itself was not statutory subject matter, the mathematical algorithm was not statutory unless it was applied in some fashion .39

The Freeman-Walter-Abele test has two parts. The first part of the test asks examiners and the courts to determine whether a claim includes a mathematical algorithm. If there is no mathematical algorithm, the claim is for statutory subject matter: ‘‘nonmathematical’ algorithms are statutory.[40] If, on the other hand, a mathematical algorithm is part of the claim, then the examiner must apply the second part of the test and determine whether the algorithm is sufficiently ‘ ‘applied. ’ An invention that includes a mathematical algorithm is statutory only if the mathematical algorithm is ‘ ‘applied in any manner to physical elements or process steps or if the invention is ‘‘otherwise statutory’ without the algorithm.[41]

Mathematical Algorithms-The line between “mathematical algorithms’ and other types of program function is difficult to draw. PTO guidelines state that claims that include mathematical formulae or calculations expressed in mathematical symbols indicate that the program function is a mathematical algorithm. Terms in a claim such as ‘ ‘computing’ or ‘ ‘counting’ may also indicate the presence of a mathematical algorithm.[42] On the other hand, the claim does not recite a mathematical algorithm if the invention can be stated in terms of its operations on things in ‘‘the real world’ that are not conventionally considered ‘ ‘mathematical. For example, claims for inventions that would process architectural symbols[43] or translate languages were found to be ‘ ‘non-mathematical.

The distinction in patent law between mathematical algorithms and other software-related inventions was discussed by the C. C.P.A. in Bradley:

---

35 *In re Bernhart* 417 F.2d 1400 (C.C.P.A. 1969). ‘‘In onc sense, a general-purpose digital computer maybe regarded as but a storeroom of parts and/or electrical components. But once a program has been introduced, the general-purpose digital computer becomes a special-purpose digital computer (i. e., a specific electrical circuit with or without electro-mechanical components) which, along with the process by which it operates, may be patented subject, of course, to the requirements of novelty, utility, and non-obviousness ’ *In re Prater, 415* F.2d 1393, 1403.

36 ‘‘Mathematical Algorithms and Computer Programs,’ *Official Gazette of the Patent Office*, vol. 1106, No. 5, Sept. 5, 1989, pp. 5-12.

37 ‘ ‘The focus of the inquiry should be whether the claim, us *a whole, is* directed essentially to a method of calculation or mathematical formula. ’ *In re Diehr,* 602 F.2d 987,

38 409 U. S. 64.

39 ‘‘It is thus clear that the ‘nutshell’ language of *Benson* expressed the ancient rule that practical application remains key, Because it did not consider the performance of an algorithm by a computer as constituting a practical application o! that algorithm under the rule, the Court must have viewed Benson’s claims as effectively claiming the ‘effect, ’ principle, or law or force of nature (the algorithm) itself. ” *In re de Castelet*, 562 F.2d 1243 (C. C.P.A. 1977).

40 ‘‘The CCPA {has] . . . held that a computer algorithm, as opposed to a mathematical algorithm, is patentable subject matter. ’ *Paine, Webber, Jackson & Curtis v. Merrill Lynch, Pierce, Fenner & Smith, 564* F. Supp. 1358, 1367.

41 ‘‘Ma[hcmatical Algorithms and Computer Programs, ’ op. cit , footnote 36, p. 8.

42 Ibid.

43 *In re Phillips*, 608 F.2d 879 (C. C.P.A. 1979)

44 *In re Toma*, 57 s F.2d 872 (C.C p A.1978).

> [The data being manipulated] may represent the solution of the Pythagorean theorem, or a complex vector equation describing the behavior of a rocket in flight, in which case the computer is performing a mathematical algorithm and solving an equation. This is what was involved in *Benson* and *Flook.* On the other hand, it may be that the data and the manipulations performed thereon by the computer, when viewed on the human level, represent the contents of a page of the Milwaukee telephone directory, or the text of a court opinion retrieved by a computerized law service. Such information is utterly devoid of mathematical significance.[45]

At one point during the period in which it was uncertain how *Benson* was to be interpreted, PTO viewed all claims that involved the use of a computer as ''mathematical' in the *Benson* sense,[46] because computers perform logical and arithmetic operations. The C. C.P.A. responded to this by writing:

> The board's analysis confuses *what the* computer does with *how* it is done. It is of course true that a modem digital computer manipulates data, usually in binary form, by performing mathematical operations. . . But this is only *how the* computer does what it does. Of importance is the significance of the data and their manipulation in the real world, i.e. *what the* computer is doing.[47]

When Is an Invention That Uses a "Mathematical Algorithm " Statutory?—The second part of the Freeman-Walter-Abele test determines if a mathematical algorithm is ''applied' '—in which case the invention is statutory--or if the applicant is attempting to claim the nonstatutory mathematical algorithm. A software-related invention that includes a mathematical algorithm has been found to be applied and statutory if the computer is being used as part of an apparatus or process for transforming physical substances into a different physical state. For example, the invention that the Supreme Court held to be statutory in *Diamond v. Diehr* used the result of a calculation to control a process for curing rubber.

However, a distinction is made between transformations of physical substances and the mere manipulation of ''data. '[48] If the claim is for a series of calculations whose only result is a ' 'pure number,"[49] then the claimed invention is not statutory. For example, the purpose of a process found nonstatutory in *Parker v. Flook*[50] was to calculate an "alarm limit. " Because the alarm limit was just a ''number" and not clearly applied in a physical sense the Supreme Court ruled that the claimed process was not statutory.

At one time it was believed that a mathematical algorithm could become statutory subject matter if the claim were in apparatus form. The apparatus claim was thought to make the invention sufficiently ''applied' or ''non-abstract. ' '[51] However, PTO will no longer approve these applications, viewing them as attempts to circumvent the nonstatutory subject-matter rejection .52

Software Patents—The term "software patent" is frequently used in the policy debate to describe a class of inventions that some believe should not be statutory subject matter or should not be infringing. The policy debate is complicated by the fact that the term software patent' does not correspond directly to a PTO technology class or subclass. The term ''software patent, ' as used in the policy debate, appears to refer to those inventions that would usually be implemented using a program executing on a general-purpose computer.

One difficulty with the use of the term "software patent' is that terminology used by PTO, such as ''computer-related invention' or "computer process' does not refer only to inventions implemented in software. These terms are also used to refer to

---

45 *In re* &a&y, 600 F.2d 812 (C.C.P.A. 1979).

46 See, e.g., *In re Bradley*, 600 F.2d 807 (C. C.P.A. 1979).

47 Ibid., p. 811.

48 "Mathematical Algorithms and Computer Programs, " op. cit., footnote 36, p. 9.

49 *In* r. Waiter, 618 F.2d 767.

50 437 U.S. 584,

51 ''The instant claims, however, are drawn t. physical structure and not to an abstract 'law of nature, a mathematical formula or an algorithm. ' There is nothing abstract about the claimed invention. It comprises physical structure, including storage devices and electrical components uniquely configured to perform specified functions through the physical properties of electrical circuits to achieve controlled results. Appellant's programmed machine is structurally different from a machine without that program. ' *In re Noll*, 545 F.2d 148 (C. C.P.A. 1976).

52 ''Mathematical Algorithms and Computer programs, op. cit., footnote 36, p. 8.

inventions that could be implemented in hardware .53 Under current law, the form of implementation does not determine whether a computer-related invention is statutory-it is significant that the Freeman- Walter-Abele test for statutory subject matter only checks for the use of mathematical algorithms, not the use of software. OTA uses the term ''software-related invention' to refer to inventions implemented in software.

Another difficulty with defining the term "software patent'' is that software is used in a variety of inventions. Not all software-related inventions are products of the ''software industry. ' Many traditional industrial processes are now controlled by computers or use embedded processors. For example, the invention found statutory by the Supreme Court in *Diamond v. Diehr* used a computer to control a process for curing rubber.

''Software patent-type' inventions may be distinguished from other ''computer-related inventions' because they are not as well represented in the PTO'S database of prior art. This database consists mainly of issued patents. Many significant advances in software are not represented in the database because few ''software patents' were issued before the mid-1980s. While in theory these inventions could have been implemented in hardware, in practice they were not. As it was widely assumed that implementation in software precluded the issuance of a patent, few developers applied for patents on these inventions. The gaps in the PTO database of prior art make it more difficult for examiners to determine whether an invention is novel and nonobvious.

When Is ***a Patent for a Software-Related Invention Infringed?***-- There is uncertainty about the scope of protection available from a patent on a software-related invention. The breadth of protection is determined during infringement litigation, of which there has been little to date. One important issue will be the interpretation of claims. Before determining if a device is infringing, courts interpret a claim by looking at the specification and the prosecution history. For exampIe, "means-plus-function' claims do not cover all of the "means" for performing a function, only the "structure, material or acts described in the specification and equivalents thereof. ' '[54] One question might be whether the claims cover both hardware and software implementations of an invention.

"Literal infringement' occurs when an accused device or process incorporates each and every element of a claim. Even if there is no literal infringement, a device can still be found to be infringing-this is known as the "doctrine of equivalents, '[55] The doctrine of equivalents is applied when an accused device does not incorporate every element of the claim but is still ''close enough.'[56] Infringement occurs if the accused device or process accomplishes "substantially the same thing, in substantially the same way to achieve substantially the same result. ' '[57] Generally, ' 'pioneer' inventions that represent a substantial advance over the prior art are granted a broader range of equivalents by the courts. Observers have argued that some software-based patents, though claimed and allowed broadly due to a lack of knowledge of constraining prior art, may not be true pioneers in their fields .58

Protection of Program Function— "Software Patent" Policy Issues

There has been considerable debate about the granting of patents for software-related inventions. Some believe that no inventions that use software should be patentable or that only software-related inventions that are traditional industrial processes should be statutory subject matter.[59] Others believe that inventions that use software should not be

53 Whether Or not a software implementation infringes a "hardware" patent depends on the interpretation of "equivalent" in the [35 U. S. C.] section 112(6) sense and the "doctrine of equivalents. " See Ronald S. Laurie and Jorge Contreras, '*Application of the Doctrine of Equivalents to Software-Based Patents, '' in Michael S. Keplinger and Ronald S. Laurie (eds.), *Patent Protection for Computer Software* (Englewood Cliffs, NJ: Prentice Hall Law and Business, 1989), p. 161.

54 35 U.S.C. 112(6).

55 The ''doctrine of equivalents' should not be confused with the meaning of ''equivalent as used in interpreting means-p s-function claims. See Donald S. Chisum, *Patents* (New York, NY: Matthew Bender, 1991), vol. 4, pp. 18-60—18-63.

56 Laurie and Contreras, Op, cit., footnote 53, p. 163.

57 Chisum, op. cit., footnote 55.

58 Laurie and Contreras, op. **cit.,** footnote 53, p. 169.

59 See Pamela Samuelson, "Benson Revisited, " *Emory Law Journal, vol. 39, No. 4,* fall 1990, pp. 1133-1142.

treated differently from other types of inventions.[60] In fact, some who believe that software-related inventions should be patentable have argued that many of what are now deemed nonstatutory mathematical algorithms should be statutory subject matter.[61]

Some of the concerns about the patenting of inventions that use software are similar to those expressed about the patent system in general[62] or about the patent system's ability to accommodate any new technology. In some cases these questions have been brought into sharper focus in the context of software. For example, the appropriate length of the patent term has been a subject of discussion for many years (see chapter 6); many believe that 17 years is especially inappropriate for a fast-moving technology such as software.[63] The fact that patent applications are kept secret until the patent issues makes it impossible to be certain that a product under development will not 'be accused of infringing a patent; this ''kmdmine' problem may be exacerbated by the longer pendency for computer-related inventions. However, two areas of concern are more directly related to the question of software patentability: the effect of patents on industry structure and innovation, and the quality of the patents that have been granted.

First, it is argued that the widespread use of patents could change the structure of the software industry in a way that would actually reduce the rate of innovation. According to those who hold this view, patenting favors larger companies, not the small companies that have historically been the source of much innovation. The software industry has had a disproportionate number of smaller companies; in part, this industry structure was due to the fact that the limited use of patents and licenses kept barriers to entry modest. There is a concern that the widespread use of patents could reduce small-company-based innovation by raising barriers to entry, either as a result of the need to pay royalties or the added costs of searching and filing for patents,[64] In addition, *large companies could engage in* portfolio trading while small companies without extensive patent portfolios would have their freedom to develop products restricted.[65]

The alternative view is that the economics of the software industry is not that different from the economics of other industries, and that patents are therefore equally appropriate for encouraging software development. Some argue that software development is becoming increasingly expensive, and patents may provide the incentive needed to invest or attract venture capital funding.[66] In addition, it may be that patents in fact benefit small companies, by providing a means to protect their development effort against appropriation by a larger company.[67]

A second set of concerns focuses on the quality of patents that have issued. Some consider that many patents have issued that do not in fact represent significant advances.[68] From the developer's perspective, this increases the probability that a program could be accused of infringing patents. The developer would then have to decide whether to engage in costly litigation in an attempt to invalidate the patent. The perceived problems with examination quality have primarily been the result of the long period of time during which it was uncertain whether software-related inventions were statutory subject matter. Few patents issued for software-related inventions, leading to gaps in PTO'S database of prior art. Some believe that the the problems with the database of prior art can be resolved given enough time.[69] However, the burdens on the PTO of increasingly backlogged applications and external criticism may be exacerbated over the next several

60 Donald S. Chisum, "The Patentability of Algorithms, ' *University of Pittsburgh Law Review,* vol. 47, No. 4, summer 1986, pp. 1009-1019.

61 Ronald S. Laurie, ''The Patentability of Artificial Intelligence Under US Law, '' in Morgan Chu and Ronald S. Laurie (eds.), *Patent Protection for Computer Software* (Englewood Cliffs, NJ: Prentice Hall Law and Business, 1991), pp. 288-290.

62 For an overview of current concerns about the patent system, see ' 'The Patent Game: Raising the "rit e* " *Science, vol.* 253, July 5, 1991, pp. 20-24.

63 Mitchell D. Kapor, Chairman and Chief Executive Officer, ON Technology, Inc., testimony at hearings before the House Subcommittee On Courts, Intellectual Property, and the Administration of Justice, Nov. 8, 1989, Serial No. 119, p. 244.

64 Brian Kahin, ' 'The Software Patent Crisis, ' *Technology Review,* vol. 93, No. 3, April 1990, p. 53.

65 Ibid.

66 Chisum argues that software should not be a ''disfavored technology. " Donald S. Chisum, op. cit., footnote 60,

67 Elon Gasper, Ed Harris, Paul Heckel, William Hulbig, Larry Lightman, and Mike O'Malley, letter, *New York Times,* June 8, 1989, editorial page.

68 Kahin, op. cit., footnote 64.

69 David Bender, letter, New *York Times,* June 8, 1989, editorial page.

Box 4-B--Cryptography

The recorded history of cryptography as a means for securing and keeping private the content of communications is more that 4,000 years old. Manual encryption methods using code books, letter/number substitutions and transpositions, etc. have been used for hundreds of years-the Library of Congress has letters from Thomas Jefferson to James Madison containing encrypted passages. Modem (computer-based) cryptographic technologies began to develop in the World War II era, with the German *Enigma* cipher machine and the successful efforts to break the cipher computationally; cryptographic research and development in the United States has often proceeded under the aegis (or watchful eye) of the National Security Agency and, to some extent, the National Institute of Standards and Technology (NIST).

Encryption techniques can be used to maintain the secrecy/privacy of messages; they can also be used to authenticate the content and origin of messages. The latter function is of widespread commercial interest as a means of electronically authenticating and ' 'signing" commercial transactions like purchase orders and funds transfers, and ensuring that transmission errors or unauthorized modifications are detected. Encryption is a mathematical process and the descriptions of different techniques (e.g., the Federal Data Encryption Standard (DES), the Rivest-Shamir-Adelman (RSA) public-key cipher, the "Trapdoor Knapsack" cipher, etc.) are usually referred to as ''algorithms. ' Nevertheless, cryptographic systems have been successfully patented (usually as "means plus function" claims); the RSA system was patented by the Massachusetts Institute of Technology and licensed in 1982 to the inventors, who formed a private company to market the system. Results from other university research in cryptography have also been patented and licensed—for example, U.S. Patent No. 4,218,582 for a "Public Key Cryptographic Apparatus and Method" invented by Martin Hellman and Ralph Merkle of Stanford was granted in 1980----as have commercially developed systems.

Patents may be complicating development of a new Federal standard for a public-key cipher. In 1991, NIST proposed a digital signature standard (DSS) for unclassified use in digitally authenticating the integrity of data and the identity of the sender of the data. The proposed standard is intended to be suitable for use by corporations, as well as civilian agencies of the government. NIST has filed for a U.S. patent on the selected technique and plans to seek foreign patents. NIST has also announced its intention to make the DSS technique available worldwide on a royalty-free basis. According to press accounts, NIST has chosen the DSS algorithm as a standard to avoid royalties. Some critics of this choice (including the company marketing the RSA system) have asserted that the RSA algorithm is technologically superior and that NIST deliberately chose a weaker cipher. In late 1991, NIST'S Computer Security and Privacy Advisory Board went on record as opposing adoption of the proposed DSS.

SOURCES: U.S. Congress, Office of Technology Assessment, *Defending Secrets, Sharing Data: New Locks and Keys for Electronic Information,* OTA-CIT-31O (Washington, DC: U.S. Government printing Office, October 1987); Michael Alexander, "Data Security Plan Bashed,' *Computerworld,* vol. 25, No. 26, July 1, 1991, pp. 1, 80; Richard A. Danca "NIST Crypto Change Takes Fed Vendors by Surprise," *Federal Computer Week,* July 8, 1991, pp. 1,37; *Federal Register,* vol. 56, No. 169, Aug. 30, 1991, pp. 4298042982; Richard A. Danca "NIST Signature Standard Whips Up Storm of Controversy From Industry," *Federal Computer Week,* Sept. 2, 1991, p. 2; Darryl K. Taft "Board Finds NIST'S DSS Unacceptable, " *Government Computer News,* vol. 10, No. 26, Dec. 23, 1991, pp. 1,56.

years. Computer implemented processes will become more "commonplace and important in a wide variety of industries and applications (see box 4-B), ranging from home entertainment to scientific research to financial services.

There may be practical limitations on attempts to exclude ''software inventions from the patent system. First, many claims in computer-related invention patents issuing today cover both hardware and software implementations; if the software implementation were not an infringement, the value of a "hardware" invention could be appropriated. Some inventions that were initially "hardware" inventions are now being implemented in software, as faster processors have become available. Second, there are-many inventions that use software but are not the type of invention that has been the subject of concern in the policy debate. There does not appear to be much concern about the patenting of traditional industrial processes that happen to use software as part of the apparatus or to perform a step in the process.

The ***Freeman-Walter-Abele*** *Test—Another issue* is whether the Freeman-Walter-Abele test draws the line between statutory and nonstatutory subject matter in the right place. Some observers believe that some of what PTO and the Court of Appeals for the Federal Circuit regard as nonstatutory mathematical

algorithms should be statutory subject matter.[70] A variant of this opinion is that "field of use" limitations should be sufficient to demonstrate that a mathematical algorithm is ''applied' and the claimed invention statutory. In *Parker v. Flook*[71] *the* Supreme Court held that language in the claim that limited the field of use of a mathematical algorithm to processes comprising the catalytic conversion of hydrocarbons was not sufficient to make the invention statutory.

Supporters of an expansion in the scope of statutory subject matter argue that some mathematical algorithms fall within the ''useful arts' and that their invention should be encouraged by the patent system. Under the *Benson* analysis, more efficient methods of solving general ' 'mathematical' problems on a computer are not statutory subject matter. In its decision finding Benson's application to be statutory (later reversed by the Supreme Court), the C. C.P.A. listed a number of advances embodied by the invention:

> . . . reducing the number of steps required to be taken, dispensing with the repetitive storing and retrieval of partially converted information, eliminating the need for interchanging signals among various equipment components, and the need for auxiliary equipment, and decreasing the chance of error.[72]

The Freeman-Walter-Abele test may also be difficult to apply consistently .73 The distinction between "mathematical" and "nonmathematical" algorithms has been criticized by computer scientists as a creation of the case law that lacks a foundation in computer science.[74] It may be that:

> . . . any attempt to find a helpful or cutting distinction between mathematics and nonmathematics, as between numerical or nonnumerical, is doomed.[75]

Some commentators have suggested that patents have issued for inventions that do not appear to satisfy the conditions of the test, or at least indicate that the test is difficult to understand.[76] There is also a sense among some patent attorneys that PTO has recently changed its application of the Freeman-Walter-Abele test, resulting in an increase in the number of rejections for nonstatutory subject matter.[77]

The difficulty in distinguishing between mathematical and other algorithms has been used to support calls for both an expansion and a reduction in the scope of statutory subject matter. Those who would reduce the scope of statutory subject matter argue that, since the distinction cannot be easily made, all algorithms should be nonstatutory .78 Those who would expand the scope argue the opposite—if some algorithms are statutory and no distinction can be made, then statutory subject matter should include many of what are now called mathematical algorithms.[79]

External Design

When Is the External Design of a Program Protected?

The external design of a program includes its user interface and the conventions for communication with other programs. The design of a user interface can include the appearance of images on a screen, the choice of commands for a command language, or the design of a programming language. The external design may also include file formats and communi-

70 ''[P]olicy considerations indicate that patent protection is as appropriate for mathematical algorithms that are useful in computer programming as for other technological innovations. ' Chisum, op cit., footnote 60, p. 1020.

71 98 S. Ct. *2522 (1978).*

72 *In re Benson,* 441 F.2d 682,683 (C. C.P.A. 1971).

73 "Maintenance of such an arbitrary and unclear line between mathematical and nonmathematical algorithms is necessary only because of the assumption of the continued vitality of *Benson. Benson* held that 'something' is per se unpatentable but failed to provide reasoning that could bc applied to determine the scope of the per se rule. ' Chisum, op. cit., footnote 60, p. 1007.

74 Allen Newell, ' 'The Models Are Broken, The Models Are Broken, ' *University of Pittsburgh Law Review, vol. 47,* No. 4, summer 1986, pp. 1023-1035,

75 Ibid., p. 1024.

76 on, issued patent often cited involves the ''Karrnakar algorithm. ' See Pamela Samuelson, ''Benson Revisited, '' *Emory Law Journal, vol.* 39, No. 4, fall 1990, pp. 1099-1102.

77 Robert Greene Sterne and Edward J. Kessler, ''Worldwldc Patent Protection in the 1990's for Compulcr-Related Technology, ' in Morgan Chu and Ronald S. Laurie (eds.), *Patent Protection for Computer Software* (Englewood Cliffs, NJ: Prentice Hall Law and Business, 1991), p. 445.

78 See, e.g., Samuelson, op. cit., footnote 76, pp. 1139-1 140.

79 See, e.g., Chisum, op. cit., footnote 60, p. 959.

cations protocols. Modules of a program, such as a procedure, also have an interface.

Patent, copyright, and trade secret law have all been used to protect elements of external designs. Some external designs, such as communications protocols, may be patentable.[80] Patents can also be used to protect elements of user interfaces, if novel and nonobvious, and design patents may be available for some of the ''ornamental' aspects of user interfaces.[81] Trade secret law may provide some degree of protection, if a program is distributed in machine language form. To specify all of the externally observable behavior of an interface, one must generally know all permitted sequences of interface actions. Determining all of the possible sequences of interface actions may be difficult if it is not possible to study the assembly language or high-level language versions of the program code.

Copyright protection may be available for aspects of external designs, especially those that use screen displays. The screen displays of a video game are often protected through an ' 'audiovisual' copyright. Other user interfaces have also been found to be protected by copyright law, There are two different approaches to protecting user interfaces using copyright law. One approach is to protect the user interface through the copyright in the underlying program.[82] The other approach is to regard tie screen display as a separate work from the program code, and protect it as an audiovisual work or as a compilation of literary terms (for interfaces that use text). The scope of copyright protection for user interfaces that do not use a screen display, such as command languages or programming languages, has not been at issue in a decided case. However, there are some who feel that the legal reasoning used in the cases where the user interface used screen displays would protect these types of external design as well.[83]

Interfaces other than user interfaces have also been the subject of copyright litigation. The format for entering statistical data into a structural-analysis program has been found to be not protected.[84] There has been an effort to assert copyright protection for what the court described as ''minor content variations' in the bit pattern of a communications protocol,[85] *but the court did* not find in the variations ''choice and selection' beyond the content of an earlier protocol to evidence sufficient originality. Dictum in a 1985 case, *E.F. Johnson v. Uniden,*[86] indicates that the court viewed the development of a communications product (a radio) ''compatible' with an existing product as permissible behavior. The court emphasized, however, that permissible development of a compatible product requires that the implementation be done independently[87]—achieving compatibility in the external design is not an excuse for copying the program code.[88] No cases have addressed such issues as the interfaces in class libraries for object-oriented languages.

Perceptions of the scope of copyright protection for interfaces have changed over the past decade. In the early 1980s some had assumed that the external design was unprotected[89] and that the only issue was whether the implementation had been done independently—there are usually different ways of writing a program with the same interface. The view that copyright protection for interfaces was limited was reflected in the use of 'clean rooms, ' in which a specification of the program is given to program-

80 Sterne and Kessler, op. cit., footnote 77, p. 402.

81 Daniel J. Kluth and Steven M. Lundberg, ''Dcslgn Patents: A New Form of Intellectual Property Protection for Computer Software, ' *JPTOS*, December 1988, p. 847.

82 *Telemarketing* v. *Symantec*, 12 U, S.P.Q.2d1991 (N.D. Cal. *1989); Lotus v. Paperback*, 740 F.Supp. 37 (D. Mass. 1990).

83 Ronald L. Johnston and Allen R. Grogan, ' 'Copyright Protection for Command Driven Interfaces, ' *The Computer Lawyer*, vol. 8, No. 6, June 1991, p. 1.

84 *Engineering Dynamics* v. *Structural Software*, No. CV 89-1655, U.S. District Court, E.D. Louisiana, Aug. 29, 1991; *Synercom Technology* v. *University Computing*, 462 F. Supp. 1003 (N.D. Tcx. 1978).

85 *Secure Services Technology* v. *Time and Space Processing*, 722 F. Supp. 1354, 1362 (E.D. Va. 1989).

86 623 F.Supp. 1485 (D. Minn. 1985).

87 Ibid,, p. 1501, footnote 17.

88 S.. also *Apple v. Franklin*, 714 F.2d 1240, 1253 (3d Cir. 1983).

89 In a 1986 article Duncan Davidson wrote: ''It is striking that despite all the concerns raised over software copyrights, a patent-like monopoly does not exist m any area of software. Application environments like Lotus 123 have been both cloned and emulated by other spreadsheets. " Duncan M. Davidson, ''Common Law, Uncommon Software, *University of Pittsburgh Law Review*, *vol. 47, No. 4,* summer 1986, p. 1077. (In the 1990 case *Lotus* v, *Paperback* (740 F. Supp. 37) such a clone was found to be a copyright infringement.)

mers who have not seen the original program (the aim of the procedure being to make available ''clean' uncopyrighted ideas without the ''taint' of the program's copyrighted expression). The theory was that since the clean-room programmers had not seen the original program there could be no infringement. The most commonly cited example of a clean-room developed product is an operating system program used in a type of microcomputer.[90] The legal status of clean-room practices is still uncertain; however, in the 1991 case *Computer Associates v. Altai,* a program implemented using a clean-room-type process was found to be noninfringing.

Policy Issues-External Design

The economic effects of protecting interfaces are difficult to evaluate, requiring a determination of the appropriate level of incentives and the role of standards and network externalities. An evaluation of the economic effects of intellectual property protection may also be complicated by the fact that there are different types of interfaces. The value of a standard, and the balance between the cost of designing the interface and cost of its implementation, may both depend on the type of interface.

lncentives-One policy position is that intellectual property protection is required in order to provide the proper incentives for the development of software. It is argued that protection of the program code alone is not sufficient to provide this incentive. Because there are different ways of writing a program with the same interface, it may be possible to reimplement the same interface without a finding of infringement, If the cost of reimplementation were small when compared to the orginal developer's investment in designing the interface, it would be relatively easy to appropriate this investment. Without more direct intellectual property protection for the external design, it is argued, there would be less incentive to develop new interfaces.

An important factor in evaluating whether external designs should be protected is therefore the relative cost of design and implementation. Supporters of intellectual property protection for external designs argue that the cost of implementation is becoming less significant. In *Lotus the* court said:

> I credit the testimony of expert witnesses that the bulk of the creative work is in the conceptualization of a computer program and its user interface, rather than in its encoding.[91]

Similar considerations are said to apply to other types of interfaces: during an intellectual property panel at the 1990 Personal Computing Forum, one participant said ' 'the hard work in doing object-oriented technology is in the interface design, The implementation of an object is trite. ' '[92] The relative cost of design and implementation is also an important factor in the recompilation debate discussed later in this chapter—it has been argued that recompilation can make it significantly less expensive for a competitor to reimplement an existing program.

The alternative view is that there is sufficient incentive to engage in the design of interfaces, even without intellectual property protection. Those who argue for this position claim that reimplementation may be time consuming and expensive, providing the original developer with significant lead time.[93] Other factors may also provide a significant advantage to the original designer of the interface.[94] Long-range planning of enhancements may favor the interface originator, for example.[95]

---

90 Ibid.

91 *Lotus* v. *Paperback,* 740 F. Supp. 37,56 (D. Mass. 1990). Others also hold the view that the effort involved in dcsigrnng a user interface deserves protection. See, e.g., Ben Shneiderman, "Protecting Rights in User Interface Designs, ' *SIGCHI Bulletin,* October 1990, vol. 22, No. 2, p.18.

92 Adele Goldberg ParcPlace Systems, at Personal Computing Forum 1990, transcript in *Release 1.0,* vol. 90, p. 107.

93 ''Software is so complex and idiosyncratic that, unless the person is deliberately copying the internals of the code, a reproduction of a sophisticated application so flawless that it has equivalent quality and utility to the original is usually significantly difficult and expensive to produce that any firm with the economic and intellectual resources to do a good job at this prefers to create original products which represent a greater opportunity. ' Mitchell D. Kapor, Chairman and Chief Executive Officer, On Technology, Inc., testimony at hearings before the House Subcommittee on Courts, Intellectual Property and the Administration of Justice, Nov. 8, *1989,* Serial No. 119, p. 243.

94 ' 'And I must tell you that the de velopment of the software program, maintaining it, keeping it documented, porting it to other computers, evolving it, enhancing it, supporting it, answering service calls, and so on and so forth, this is the bigger picture, and it is really simplistic (o say that if one can actually just lake the looks of a program, they will be able to run with it and in fact surpass whoever originated the first program. " Richard Bezjian, President, Mosaic Software, at panel on ' 'Intellectual Property in Computing: (How) Should Software Be Protected" (Cambridge MA: Transcript, Oct. 30, 1990), p. 24.

95 Brett L. Reed, ''Obscrvatioms on the Economics of Copyright and User Interfaces, " *International Computer Law Adviser,* vol. 5, No. 10, July 1991, p. 4.

Network Externalities-There is a question as to whether the effect of intellectual property laws on standardization should affect an evaluation of the appropriate level of protection for interfaces,[96] Standards benefit users in a number of ways. For example, a greater variety of application programs will be developed if there is a standard operating system-developers will be able to sell to a larger market and more easily recover their development costs. Another example of an advantage of standards is that consistency among user interfaces makes it easier for users to learn to use a new program. The benefits to users that result from the wider use of an interface are known as ' 'network externalities' (see also ch. 6). Moreover, users may benefit from competition among suppliers of standard product. For example, suppliers of compilers for standard languages compete on the basis of the cost of the compiler and the efficiency of the machine language code generated.

De facto standards evolve through the actions of the market. If there is a dominant firm, the interface that it has developed is more likely to become the standard. Alternatively, a de facto standard can develop because of a ' 'bandwagon' effect. If consumers are faced with a choice between different interfaces, network externalities make the more widely used product more attractive. Consumers value the network externalities, not just the intrinsic value of the interface.

Standards may also be negotiated using standards committees. Firms engage in voluntary standards-setting when they determine that they are better off with a part of a larger market than if they were to continue trying to establish their interface as a de facto standard. Consumers may be less willing to buy a proprietary product. For example, it is not clear whether a computer language available from a single vendor would be widely used[97]—a developer might be unwilling to rely on a single supplier.

One view is that intellectual property protection may harm users by affecting standardization processes. It is argued that firms may not have the correct incentives to engage in voluntary standards setting because intellectual property protection can increase a fro's vested interest in seeing the interface it has developed chosen as a standard, slowing the standardization process.[98] This could harm users, until a standard is negotiated or one interface prevails in the marketplace, Users could also be harmed if new programs are not "backwards compatible" and require users to learn a new interface to take advantage of new features or better performance, In addition, it has been argued that network externality effects can complicate the balancing of incentives for software development, by resulting in "extra" revenues for firms that succeed in establishing their products as a de facto standard and making it more difficult for other firms to enter the market.[99]

The other view is that the question of standards should be kept separate from the basic issue of the proper incentives for software development. Furthermore, it is argued that voluntary standards efforts are sufficient,[100] and that there is a trend in the computer industry toward using more formal standardization and licensing processes. Consortia have formed in a number of areas, such as user interface design and operating systems. There are a variety of voluntary standards committees that are developing standards for data communications protocols,[101] operating system interfaces,[102] and principles for user interface design.[103]

---

[96] For discussions of standardization considerations, see Peter S. Menell, ''An Analysis of the Scope of Copyright protection for Application Programs, ' *Stanford Law Review, vol. 41, No. 5, May 1989,* pp. 1100-1 101; Richard H. Stem, ''Legal protection of Screen Display s,' *Columbia 'LA Journal of Law& the Arts,* vol. 14, pp. 291-292; Anthony L. Clapes, *Software, Copyright, & Competition* (Westport, CT: Quorum Books, 1989), p. 206.

[97] Alfred Z. Spector, "Software, Interface, and Implementation, " *Jurimetrics*, vol. *30, No.* 1, fall 1989, p. 89.

[98] Joseph Farrell, "Standardization and Intellectual Property, " *Jurimetrics*, *vol. 30, No.* 1, fall 1989, p. 44.

[99] The court in *Lotus* did not view this as affecting the determination of whether [he copyright had been infringed,

100, 'Th, exclusions [of ,nte-faces and limitations of 'decompilation' from copyright law] are unnecessary to permit development of 'interoperable' programs: thousands of such programs have been created under the existing copyright rules, thanks to the work of international standards organization and the voluntary sharing of necessary information. " William T. Lake, John H. Harwood, and Thomas P. Olson, "Tampering With Fundamentals: A Critique of Proposed Changes in EC Software Protection, ' The *Computer Lawyer*, vol. 6, No. 12, December 1989, p. 3.

[101] Steven Turner, "The Network Manager's Compendium of Standards, " *NetworkWorld*, vol. 8, No. 15, Apr. 15, 1991, p. 1.

[102] D. Richard Kuhn, ''IEEE's Posix: Making Progress, ' *IEEE Spectrum, vol.* 28, No, 12, December 1991, pp. 36-39.

[103] See, e.g., Pat BillingsIcy, ' 'The Standards Factor: Standards on the Horizon, " *SIGCHI Bulletin,* vol. 22, No. 2, p.10.

User Interface

Which Elements of a User Interface Design Have Been Protected?

The type of interface that has received the most attention in the software intellectual property debate is the user interface. Two factors have been taken into account by the courts when determining the scope of protection for user interfaces. First, standard ''building blocks' of user interface design such as the idea of using a menu have not been protected. Second, the courts have recognized constraints on the design; for example, commands that are necessary to the overall purpose of a program have not been protected. Hardware and software constraints on the way information is presented on the screen have also been recognized,

Having determined which elements of an interface are either standard building blocks or imposed by technical constraints, the courts look for design choices. When there are design choices remaining after the constraints have been taken into account, the courts generally protect the elaboration of these design choices into a user interface. The choice and organization of commands in a menu hierarchy, and the arrangement of command terms on a screen, for example, have been found to be protected.

Unprotected Elements.—*In* general, common interaction techniques have not been protected. The idea of using a menu has not been protected. Particular menu styles have also not been protected by copyright, on the grounds that they were common in the industry. The use of a pull-down menu was not protected expression in *Telemarketing.*[104] *The* use of a two-line moving cursor menu was described as "functional and obvious" (and not protected) in Lotus.[105] Also found unprotected have been standard ways of entering commands,[106] selecting menu entries,[107] and navigating on the screen.[108]

Commands and menu options required for the overall purpose of the program would probably not be protected. For example, in *Telemarketing* menu options that allowed the user to access existing files, edit work, and print the work were not protected.[109] Also not protected were functions that were likely to be found in any outlining program, or cost-estimation program.[110] The rimes chosen for individual menu entries have, in general, not been protected. For example, the use of "print" as the command name for printing would probably not be protected.111

The courts have also addressed the issue of the organization of information on the screen, and have generally recognized constraints. Centering the headings on a screen, locating program commands at the bottom of the screen, and the use of a columnar format have all been found to be either unprotected ''conventions' chosen from a narrow range of choices or not original.112

Protected Elements.—What *has* generally been protected is the overall set of command terms and their organization into menus. The designer's judgment of the way in which users would want to use a spreadsheet, as reflected in the ' 'menu structure, " including the overall structure, the order of commands in each menu line, and the choice of letters, words, or ''symbolic tokens' to represent each

---

104 ''Plaintiffs may not claim copyright protection of an idea and expression that is, if not standard, then commonplace in the computer software industry. " *Telemarketing v. Symantec, 12* U. S. P.Q.2d 1991, 1995 (N.D.Cal. 1989).

10574.0 F. Supp. 37, 65 (D. Mass. *1990).*

106 "...*the* typing of two symbols to activate a specific command is an 'idea. ' " *Digital v. Softklone, 659* F. Supp. 449,459 (N.D. Ga. 1987).

107 *MTI* v. *CAMS,* 706 F. Supp. 934, 9950. Conn. 1989).

108 '' [T]he idea at issue, the process or manner of navigating internally on any specific screen displays likewise is limited in the number of ways it may be simply achieved to facilitate user comfort. To give the plaintiff copyright protection for this aspect of its screen displays, would come dangerously close to allowing it to monopolize as ignificant portion of the easy-to-use internal navigation conventions for computers. *MTI* v. *CAMS,* 706 F. Supp. 984, 995 (D. Conn. 1989).

10912 U. S, P.Q.2d 1991, 1995 (N.D.Cal.1989).

110 ''Nor is the listing of items for which &U is supplied subject to copyright, because, in the language of the machining industry, speeds and feeds, machining times and costs, and data specific to the size, depth, and diameter of the hole is all closely related to and hence incident to the idea of displaying this data. . ." *MTI* v. CAMS, 706 F. Supp. 984, 998 (D. Corm 1989).

111 ' 'Obvious' command terms which merge with the idea of the command term were discussed by the court in *Lotus. Lotus v. Paperback, 740* F. Supp. 37,67 (D. Mass. 1990).

112 *MTI* v. *CAMS,* 706 F. Supp. 984, 994-5 and 998 (D. Conn. 1989).

command was found to be protected in *Lotus v. Paperback.*[113] *In MTI v. CAMS, the* designer's view of how a user would go through the process of cost-estimating, as reflected in the sequence of menu screens, was found to be protected.[114] The existence of design choices has frequently been shown by the existence of a third program that uses a different menu structure and has different commands.[115] This has been interpreted to show the absence of ' 'mechanical or utilitarian constraints' [116] on the designer.

In one case, the commands themselves were not protected, but the arrangement of the command terms on the screen was protected expression.[117] As a result, the defendant was forced to redesign the product to present the command options to the user in a different way. Instead of presenting the command options on a single screen, they were distributed over a sequence of screens.

Standards—Industry conventions such as the use of certain menu styles, or the use of the 'return' key to select a highlighted menu element have not been protected by copyright law. On the other hand, the choices made by the designer of a successful product in developing the menu structure have not been recognized as a constraint on later developers. In *Lotus* the defendants sought to show that while there may have once been a number of design choices, the success of the plaintiff's spreadsheet product in the market sharply limited the choices of later developers, due to network externality effects. This argument was not accepted; the court wrote:

> By arguing that 1-2-3 was so innovative that it occupied the field and set a de *facto* industry standard, and that, therefore, defendants were free to copy plaintiff's expression, defendants have flipped copyright on its head. [118]

User Interface—Issues

At one level the software intellectual property debate has been concerned with the question of whether user interfaces should be protected at all. The secondary issues have focused on the question of which elements should be protected. Intellectual property law establishes rules for competition in user interface design by drawing lines between protected and unprotected elements. The debate about ''look and feel' '[119] reflects a concern that a particular *style* of interface would be protected by copyright law. There is a concern that the protection of an interaction style would leave too little room for innovation by others within the general style, or for its use in a different program.

However, in cases decided so far, the courts have held the mere use of a menu-based interaction style to be unprotected. The use of the "spreadsheet metaphor' has also been held to be unprotected. In effect, the courts have viewed the use of these common types of interaction in the same way that they view the use of words and stock characters alone in the application of copyright law to literature: as building blocks that should not be protected. To give one creator a monopoly over these basic elements would effectively stunt the efforts of other creators to elaborate on these elements in the production of their own works.[120]

One difficulty is that technological change is continually adding new building blocks. The cases that have been decided all involved simple text-based menus that do not represent the state of the art in user interface design. Some of the cases now in the courts involve graphical user interfaces, and it is less clear what constitutes an unprotected ''building block" of graphical user interface design, and what constitutes an elaboration of building blocks into a

113740 F. Supp. 37,67 (D. Mass, 1990). The court said that '' [a]n example of distinctive details of expression is the precise 'structure, sequence, and organization'. .of the menu command system. '

114 706 F. Supp. 984, 994 (D. Corm. 1989).

115 ''In the present case, the Court has already noted that the existence of 'Stickybear Printer' [a third program] disproves defendant's argument that there are a very limited number of ways to express the idea underlying 'Print Shop. ' Thus, there is no danger in the present case that affording copyright protection to the 'instructions' of 'Print Shop' will amount to awarding plaintiff a monopoly over the idea of a menu-driven program that prints greeting cards, banners, signs and posters. " *Broderbund Software* v. *UnisonWorld, 648* F. Supp. 1127, 1134 (N.D. Cal. 1986).

116 *Broderbund Software v. UnisonWorld, 648* F. Supp. 1127, 1133 (N.D. M. 1986).

117 *Digital Communications Associates v. Softklone Distributing, 659 F.* Supp. 449 (N.D. Ga. 1987).

118 740 F. Supp. 37, 79 (D. Mass. 1990).

119 Despite its 'widespread use in public discourse, a court has said that the " 'look and feel' concept, standing alone [was not] significantly helpful" in distinguishing between uncopyrightable and copyrightable elements of a computer program. *Lotus* v. *Paperback, 740* F. Supp. 37,62 (D. Mass. 1990).

120 For a discussion of ''idcast '' see Paul Goldstein, *Copyright Principles, Law and Practice* (Boston, MA: Little, Brown, 1989), vol. I, pp. 76-79.

protected design. There is a concern that a‘ ‘building block” could be appropriated through copyright by the developer of the first program to use it.

One question is related to the role of user interface design principles in determining the scope of design choices. One of these principles is that interface designers should be aware of the benefits of ‘external consistency. [121] External consistency allows the ‘‘transfer of learning’ from one program to another and from one generation of a program to the next. Is it necessary for two spreadsheet programs to be identical in virtually every respect, or can there be significant transfer of learning if two spreadsheet programs share only some core similarity? Would this core similarity be viewed as an unprotected ‘‘idea’ in the context of copyright law? At the same time, there is a concern that intellectual property law will force “gratuitous” differences between interfaces. [122]

Program Code

How Is the Program Code Protected?

The copying of a computer program can be prevented in several ways. If a computer program is the implementation of a patented process, then copying the program and practicing the invention would infringe the patent. Copying could also be limited by a licensing agreement between the developer of the software and a licensee. However, the main vehicle for preventing the copying of the program has been copyright law. Computer programs have been copyrightable subject matter since 1978, when the Copyright Act of 1976 became fully effective. [123]

The Copyright Act states that copyright protection does not extend to the “procedure’ or “system’ or ‘‘method of operation’ described by a copyrighted work. This is to prevent copyright from being used to protect ‘‘utilitarian’ or ‘‘fictional’ articles. For example, an electronic circuit is not copyrightable subject matter, but the circuit diagram that describes the circuit is a copyrightable “pictorial’ work. The copyright only prevents someone from copying the pictorial work, not from building the circuit. In the case of computer programs, it is especially difficult to separate the description of the function from the function itself.

“Idea” is a metaphor used in copyright law for the elements of a work that copyright law does not protect. Procedures, systems and methods of operation are ideas. ‘‘Expression ‘‘ is a metaphor for the protected elements of a work. Infringement occurs under copyright law when a work is copied and, taken together, the elements copied amount to an improper appropriation of expression. Copying can be shown by direct evidence or by inference, if the defendant had access to the plaintiff’s work and the works have substantial similarity as to the protected expression. Improper appropriation is shown by the taking of a substantial amount of protected ‘ ‘expression.

Literal Code—The literal code of a program has consistently been shown to be protected expression, and verbatim copying a copyright infringement. As a result, copying a program from one disk to another clearly infringes the copyright in the program, except to the extent permitted by the Copyright Act (e.g., section 117). This is true regardless of the language used to write the program: the argument that a program in executable (machine language) form was not copyrightable subject matter because it could be considered a ‘ ‘machine part’ has been rejected by the courts.

“Nonliteral” Copying—In a series of cases, courts have held that the internal design of a program at a level of abstraction above that of the program code could not be copied, In one case a judge wrote:

> It would probably be a violation to take a detailed description of a particular problem solution, such as a flowchart or step-by-step set of prose instructions, written in human language, and program such a description in computer language.[124]

In other words, a finding of infringement could not be avoided by making small changes to a program or

---

121 See pp. 129-130.

122 Participant in discussion at the Massachusetts Institute of Technology on “Intellectual Property in Computing: (How) Should Software Be Protected” (Cambridge, MA: Transcript, Oct. 30, 1990), p. 21.

123 House Report 94-1476 says ‘‘liteW works’’ protected under section 102(a)(1) of the Copyright Act include computer programs. The protection of computer programs under the Copyright Act was confirmed by the Software Amendments of 1980.

124 *Synercom* v. *UCC*, 462 F. Supp. 1003, 1013 n.5 (N.D. Tex. 1978).

by translating the program from one language to another language.[125]

The higher levels of abstraction of the program code are often described as the ''structure, sequence, and organization' [126] (SS0) of the program, although this terminology has been criticized,[127] Protection of the structure, sequence, and organization has been described as consistent with the application of copyright to more traditional types of ' 'literary' works such as novels.[128] The main reason to limit copying at this higher level of abstraction is that it would otherwise be possible to avoid copyright infringement by making a few trivial changes to the program text. The courts have determined that this would allow the appropriation of a significant part of the value of a program.[129]

There were two seminal cases in the area of protection for the structure, sequence, and organization of computer programs: *Whelan v. Jaslow* and *SAS v. S&H.* In these cases the particular organization of the program into subroutines or modules was found to be protected expression. In *SAS v. S&H* the court stated that copying the organizational scheme of a program would be a taking of expression, even if the program code for the ''lowest level tasks' were written independently, [130] In *Whelan v. Jaslow* the two programs were found to be substantially similar because of similarities in the detailed structure of the five subroutines that the court found to be qualitatively important to the program and ' 'virtually identical file structures. [171]

Constraints on Program Structure-The courts have applied the "merger" doctrine of copyright law by looking for evidence that the structure of the program was dictated by engineering constraints. In cases such as Q-Co. v. *Hoffman* and *NEC v. Intel the courts* have found that the similarities between programs were due to constraints imposed by the overall purpose of the program or by the hardware. If there were different ways of writing the program to perform a particular function, however, the courts have found protected expression. In *SAS v. S&H,* for example, the court wrote that:

> [The defendants] presented no evidence that the functional abilities, ideas, methods, and processes of SAS could be expressed in only very limited ways. *[32]

The number of different ways of writing a program to perform a particular function was discussed at hearings conducted by the National Commission on New Technological Uses of Copyrighted Works (CONTU).[133]

Copies—Because computer programs are protected by copyright, the making of *any* copy is an infringement. Even the transfer of a program from disk to memory is thought to be the creation of a copy that would be infringing but for the special exemption contained in section 117 of the Copyright Act, which allows a computer program to be copied ''as an essential step in the utilization of the computer program in conjunction with a machine.

The exclusive rights granted to the copyright holder are also thought by some to limit disassembly or recompilation of programs-these procedures involve the making of reproductions or ''derivative works' of the machine language program. Limitations on disassembly and recompilation provide trade secret protection for aspects of a program

125 See *Whelan* v. *Jaslow* 797 F.2d 1222 (3d Cir. 1986) and *SAS v. S&H,* 605 F. Supp. 816 (M.D. Tenn. 1985).

126 *Whelan* v. *Jaslow, 797* F.2d 1222, 1224 (3d Cir. 1986).

127 *Computer Associates* v, *Altai, No.* CV 89-0811, U.S. District Court, E.D. New York, Aug. 9,1991.

128 ''As I have indicated, CONTU had no views, and made no recommendations which would negate the availability of copyright protection for the detailed design, structure, and flow of a program under the copyright principles that make copyright protection available, in appropriate circumstances, for the structure and flow of a novel, a play or a motion picture. ' Declaration of Melville B. Nirnmer (Vice Chairman of CONTU), appendix to Anthony L. Clapes, Patrick Lynch, and Mark R. Steinberg, ''Silicon Epics and Binary Bards: Determining the Proper Scope of Copyright Protection for Computer programs," *UCLA Law Review, vol.* 34, June-August 1987, p. 1493.

129 ''. . .among the more significant costs in computer programming arc those attributable to developing the structure and logic of the program. The rule proposed here, which allows copyright protection beyond the literal code, would provide the proper incentive for programmers by protecting their most valuable efforts, while not giving them a stranglehold over the development of new computer devices that accomplish the same end. *Whelan* v. *Jaslow, 797* F.2d 1222, 1237 (3d Cir. 1986).

130 605 F. Supp 816, 826 (M.D. Tenn. 1985).

131 *Whelan* v, *Jaslow,* 797 F.2d 1222, 1228 (3d Cir. 1986).

132 605 F, Supp 816, 825 (M.D. Tcm. 1985).

133 See transcript of CONTU Meeting No. 10, pp. 44-45, quoted in declaration of Melville B, Nimmer, appendix to Clapes et. al., op. cit., footnote 128, p. 1588.

because the machine language version of programs is difficult to understand, The recompilation issue is discussed in more detail in a later section of this chapter.

Policy Issues-Protection of Literal and Nonliteral Elements of Program Code

Literal Copying—The justification for restrictions on the copying of computer programs is economic: some form of legal protection is necessary to provide program developers with the incentives to develop software. Computer programs are easy to copy—they have the same intangible character as traditionally copyrightable works. The CONTU Final Report states:

> The Commission is, therefore, satisfied that some form of protection is necessary to encourage the creation and broad distribution of computer programs in a competitive market,[134]

The Commission viewed computer program copyrightability as consistent with the expansion of copyright to new technologies over the previous two centuries, [135]

Protected and Unprotected Elements of Program Code—One of the reasons for protecting nonliteral elements of a program is to prevent later developers from avoiding a finding of infringement by making small changes. The main issue is the degree of similarity two programs may have and the degree of independent development that a later developer will be forced to do. In other words, to what extent can the intellectual work in one program be used in a second program? In the two important structure, sequence, and organization cases, *Whelan v. Jaslow* and *SAS v. S&H, the infringing* program's code was similar at a low level of detail. The line drawn between protected and unprotected elements reflects a determination of the level of competition desirable. [136]rticulating this line has proven to be difficult.

In practice, there will rarely be access to the high-level language version of a competitor's program. The only access that one would normally have to a competitor's program would be to its machine language form. Disassembly would be possible, but there would still be considerable work involved in understanding the program and reimplementing it. It is not surprising that in the structure, sequence, and organization cases there has either been access to the source code or the programs were short enough to be disassembled and studied relatively easily.[137] The legal status of attempts to disassemble a program is a major issue associated with the protection of computer programs using copyright law, and is the subject of the next section.

Recompilation

''Recompilation' is a procedure for translating a machine language program into a more understandable form. It is thought by some to be a copyright infringement, and by others to be a necessary tool for software engineering. The recompilation issue intersects many of the policy issues outlined earlier in this chapter. For example, recompilation may be used in the development of functionally compatible products; whether or not the development of such products should be permitted is itself a policy issue (see ''External Design—Policy Issues" in this chapter).

Introduction

To understand a program, there are three things you can do: read about it (e.g., documentation), read it (e. g., source code), or run it (e.g., watch execution, get trace data, examine dynamic storage, etc.).[138] Understanding a program is made easier when the high-level language or assembly language representations are available. In most cases, however, only the machine language version is distributed. Decompilation is a procedure by which a high-level language representation of a program is derived from a machine language program, and ' 'disassembly" is a procedure for translating the machine language program into an assembly language program.

---

134 National Commission on New Technological Uses of Copyrighted Works (CONTU), *Final Report* (Washington DC: Library of Congress, July 31, 1978), p. 11.

135 Ibid.

136 ''LaST Frontier Conference Report on Copyright Protection of Computer Software, " *Jurimetrics*, vol. 30, No. 1, fall 1989, p. 20.

137 *E.F. Johnson v. Uniden*, 623 F.Supp. 1485 (D.Minn. 1985) (a radio communications product), *NEC v. Intel 10* U. S. P.Q.2d 1177 (N.D. Cal. 1989) (microcode).

138 Richard B. Butler and Thomas A. Corbi, "Program Understanding: Challenge for the 1990' s," *Scaling Up: A Research Agenda for Software Engineering* (Washington, DC: National Academy Press, 1989), p. 41.

The legal status of efforts to discover assembly language or high-level language representations of a program has become the subject of an intense debate.[139] Both recompilation and disassembly involve the making of at least a partial reproduction or derivative of the machine language program, and some people believe that reverse engineering using these techniques is a copyright infringement.[140] The policy question is the extent to which limiting access to information about someone else's program through the workings of the copyright law is socially desirable.

Important factors cited in the policy debate are the uses of recompilation, the ease with which it can be done, and the degree to which the information is available from sources other than recompilation. It has been argued that limits placed on recompilation are required to provide sufficient incentives for the development of original programs. Those who take this position claim that recompilation is a straightforward and routine process that allows clone programs to be implemented at much lower cost,[141] Programs are decompiled and then:

> ... without the necessity for the significant R&D expenditures made by the innovator, the pirate goes on to alter the program to disguise the copying, and create a second, similar program which it markets as an allegedly different product for a much lower price. '[42]

Others argue that recompilation is technically difficult, and is therefore unlikely to be used for piracy. They emphasize that disassembly and recompilation can be used for a variety of other purposes, many of which would have a less direct economic impact on the developer of the program being reverse engineered. For example, some of the information gained by recompilation may be used in developing an "attaching' product that is to exchange data with the program being reverse engineered. Recompilation also could be used for maintenance, debugging, detecting viruses, investigating safety or reliability concerns, or systems integration. Indeed, some of these uses of decompilation represent situations in which an organization might reverse engineer its own programs, not just those developed by someone else.[143]

Recompilation and Disassembly

The product of recompilation or disassembly would never be identical to the original source program.[144] At the very least, comments and the names of labels, variables, and procedures would be lost in the assembly or compilation process and could not be recovered. In addition, the structure of the decompiled program would not necessarily be the same as that of the original program, although this would depend on the compiler that had been used. Because of the loss of mnemonics and much of the structure of the program, considerable work is required to understand the decompiled or disassembled program.

Disassembly is easier than recompilation. There is essentially a one-to-one conversion between the machine language statements and assembly language statements, simplifying the process of translating the machine language program into a more readable form. However, it takes a great deal of effort to understand the disassembled code from a large program.[145] Because disassemblers are **widely** available,[146] some developers assume that their programs will be disassembled, and try to write sensitive parts of their code in ways which make disassembly more difficult or make the disassem-

---

139 Pamela Samuelson, "Reverse-Engineering Someone Else's Software: Is It Legal?" *IEEE Software, vol. 7, No.* 1, January 1990, pp. 90-96.

140 Victor Siber, Corporate Counsel, IBM Corp., "Interpreting Reverse Engineering Law, ' letter to *IEEE Software,* vol. 7, No. 4, July 1990, p. 8.

141 "Decompilation of a computer program does not provide an imitator with just a good start in producing a competing product; it gives him virtually everything necessary to produce a functionally identical product, ' William T. Lake, John H. Harwood, and Thomas P. Olson, "Tampering With Fundamentals: A Critique of Proposed Changes in EC Software Protection,' *The Computer Lawyer*, vol. 6, No. 12, December 1989, pp. 1-10.

142 Testimony of James M. Burger, Chief Counsel, Apple Computer, Inc., on behalf Of the Computer and Business Equipment Manufacturers Association, at hearings before the House Subcommittee on Intellectual Property and Judicial Administration, May 30, 1991.

143 This situation presents no infringement issues.

144 The lack of identity is not relevant to the legal question of unauthorized copying.

145 An 80,000-byte machine language program for an IBM PC-class computer would result in 32,000 lines of *assembly code.* Clark Calkins, ' 'Tailoring the MD86 Disassembler for Turbo Pascal,' *Tech Specialism, vol. 2, No. 6,* June 1991, pp. 41-46.

146 For a discussion of commercially available disassemblers, see Brett Glass, ' 'Disassembler Roundup, '' *Programmer' s Journal, vol. 9.2,* March/April 1991, pp. 66-71.

bled code more difficult to understand.[147] Figure 4-2 shows a high-level language program, the corresponding machine language (compiled) program, and the results of disassembling the machine language program.

Recompilation is much more difficult; at this time there appear to be no commercially available decompilers. For this reason, it is unclear whether decompilation is widely used by ‘ ‘pirates” to decompile entire programs and then rearrange the code in an attempt to hide the copying,[148] It is possible that the term “recompilation ‘‘ is being used in the policy debate to include disassembly,[149] as “decompilation’ is often characterized as any technique that is used to transform ‘‘machine readable’ code into ‘‘human readable’ code.[150]’

Today, any effort to decompile a program would start with disassembly. Then, if one knew something about the compiler that had been used, it might be possible to match certain patterns of assembly language statements to higher level constructs. However, recompilation would be much more difficult in cases where a sophisticated compiler had been used: *optimizing* compilers delete and rearrange some of the instructions in order to make the machine language program more efficient, and the correspondence between sequences of machine language instructions and high-level instructions becomes less direct. While a pseudo-source code program could still be derived, it would be less likely that the decompiled program would immediately reveal the original program structure.

Uses of Recompilation

The information gained by reverse engineering techniques such as recompilation can be put to a variety of uses, each with a different economic effect. The effect on the developer of the program being decompiled is most direct when the information is being used to develop a competitive product, In some cases the reverse engineer maybe interested in learning about a small part of a program, such as an algorithm, that gives the program’s owner a competitive advantage.[151] In other cases, decompilation could be used to develop the specifications for a program that is fictionally compatible—a clone program, Sometimes the specifications are used in a clean-room process that is intended to ensure that the new program does not share expression with the original, for it is the protected expression that is protected by copyright.[152]

Recompilation may also be used to develop a program or hardware device that is not competitive, but “complementary” or “attaching.” This would not affect the market for the original product directly, but would create more competition in the second market. For example, knowing interface information might allow the development of competition in the market for peripheral devices such as printers. Recompilation can also be used to confirm published interface specifications; for example, in the course of debugging a program an unexpected problem may arise with another program in the system, such as an operating system.

Finally, there are a variety of uses for decompilation for which no product is developed at all. First of

---

147 Bob Edgar, “Shielded Code: How To Protect Your Proprietary Code From Disassemblers, ” *Computer Language*, vol. *8, No. 6,* June 1991, pp. 65-71.

148 Following the testimony of the computer and Business Equipment Manufacturers Association at the May 30,1991 hearings (See footnote 142), OTA asked for specific examples of piracy using recompilation and descriptions of the state of the art in automated recompilation (Joan Winston, OTA, letters to James Burger, Apple Computer, July 5, 1991 and Sept. 23, 1991). To date, OTA has not been provided with this information.

149 ‘‘A computer program is generally written, in the first instance, in ‘source code’ —that is, in a relatively high-level language such as FORTRAN or Pascal. The program is then translated (or ‘compiled’) into ‘object code, ’ which consists of instructions to the computer in the form of O’s and 1‘s. Programs are frequently distributed to customers only in object code form; the source code is retained as an unpublished, copyrighted work. Recompilation and disassembly (*which* we calf *‘decompilation’ for short*) *are* methods of reconstructing the source code of a program through copying and manipulation of its object code. ’ Lake et al., op. cit., footnote 141, p. 4 [emphasis added].

150 The legal issue is the same, whether a program is “disassembled” or “decompiled.”

151 ‘‘The source code, which often contains the trade secrets of the software creator, remains unpublished. Many software companies go to great lengths to keep their proprietary source codes confidential. . . The right to decide not to publish in any form source code goes to the heart of most software companies’ strategies for retaining the confidentiality of their most valuable and carefully guarded trade secrets. William Neukom, Vice President, Law and Corporate Affairs, Microsoft Corp., on behalf of the Software Publishers Association at hearings before the House Subcommittee on Intellectual Property and Judicial Administration, May 30, 1991.

152 For discussion of clean room issues, see David L. Hayes, ‘‘Acquiring and Protecting Technology: The Intellectual Property Audit,’’ *The Computer Lawyer,* vol. *8, No. 4,* April 1991, pp. 1-20. The effectiveness of using a clean room to avoid copyright infringement depends on whether the specifications that are used are ideas and not expression.

Figure 4-2-High-Level Language, Machine Language, and Disassembled Versions of a Program

HIGH-LEVEL LANGUAGE PROGRAM

```
program sum–of–numbers;
{This program adds the numbers from first–number to last–number}
{The text between curly brackets is known as a "comment." Comments }
{make a program easier to read and understand, but do not affect the }
{execution of the program.                                           }
var first–number, last–number, i, sum: integer;
begin
        {initialize variables}
        first–number := 1,
        last–number := 5;
        sum :_O;
        {add numbers from first–number to last–number}
        for i := first–number to last–number do
        begin
                sum := sum + i;
        end;
        {print the sum}
        writeln('The sum is sum);
end.
```

The program shown above, written in the high-level language Pascal, adds the numbers from 1 to 5. High-level language programs have to be translated (compiled) into machine language in order to be executed on the computer. Part of the compiled program is shown below.

MACHINE LANGUAGE (COMPILED) PROGRAM

```
10111000 00000001 00000000 10100011 01100000 00000010 10111000 00000101
00000000 10100011 01100010 00000010 10111000 00000000 00000000 10100011
01100110 00000010 10100001 01100000 00000010 01010000 01000001 01100010
00000010 01011001 10010001 00101011 11001000 01111101 00000011 11101001
00011010 00000000 01000001 10100011 01100100 00000010 01010001 10100001
01100110 00000010 00000011 00000110 01100100 00000010 10100011 01100110
00000010 01011001 00101001 01110100 00000111 11111111 00000110 01100100
00000010 11101001 11101010 11111111 11101000 01111010 11110111 11101000
01111100
```

Machine language programs are difficult to read and understand. If the original high-level language program is not available, disassembler programs may be used to translate the machine language program into a more understandable form called assembly language. However, assembly language programs are still more difficult to understand than high-level language programs. Part of the assembly language program is shown below.

DISASSEMBLED PROGRAM

```
2D9F  MOV   Ax, 0001          2DC1  JMP   2DDB
      MOV   [0260], AX              INC   CX
      MOV   Ax, 0005          2DC5  MOV   [0264], AX
      MOV   [0262], AX              Push  CX
      MOV   Ax, 0000                MOV   AX, [0266]
      MOV   [0266], AX              ADD   AX, [0264]
      MOV   AX, [0260]              MOV   [0266], AX
      PUSH  Ax                      POP   CX
      MOV   Ax [0262]               DEC   CX
      POP                           JZ    2DDB
      XCHG  Cx, Ax                  INC   WORD PTR[0264]
      SUB   Cx, Ax                  JMP   2DC5
      JGE   2DC1              2DDB  CALL  2558
```

SOURCE: OTA.

all, someone may wish to ''maintain' a program for which no source code is available. Recompilation or disassembly would help in understanding the program so that it could be adapted to new requirements, or to fix bugs if no other support was available. Disassembly is also used to find viruses, to examine the output of a compiler to see what it had done, and finally to examine a competitor's program to see if they had taken any protected expression.

Other Methods of Reverse Engineering

There are other methods of reverse engineering programs whose legal status is less controversial because they do not involve the making of unauthorized reproductions of the machine language program.[153] In practice, a reverse engineer would probably employ a combination of methods, depending on the application, the information being sought, the effort involved, and legal considerations. Some information can be obtained by simply executing the program: it can be run with many different data sets and its behavior observed. There are a number of different software and hardware tools that could be used to follow the course of execution of the program. However, the program code is the best specification of the behavior of the program-it may be impossible to develop tests to explore all the cases that a program may have to handle.

Other information is available from published specifications, manuals, and standards documents. In some cases companies will publish interface specifications because it is in their commercial interest to do so. Even if the information is not published, they may be willing to make it available through contractual arrangements. However, in other cases, such as when a company is active in the market for both the primary product and a complementary product, it may want to limit competition in the secondary market by not making the interface information available. Published documents may not be at the appropriate level of detail. For example, there may be scope for differences between implementations of a standard and manuals may be inaccurate or out of date, or leave some elements undocumented.

Legal Arguments for Policy Positions

While there have been proposals that a new ***sui generis*** law be enacted to protect software, much of the discussion of software intellectual property policy issues has been based on interpretations of current law. Convincing legal arguments have been made for many of the policy positions discussed in the preceding sections. The two broadest legal questions are the proper interpretation of the ''mental steps" and "law of nature" exceptions to patentability in patent law, and the proper interpretation of the statement in section 102(b) of the Copyright Act that copyright protection does not extend to ''processes' or ''methods of operation. Both the exceptions to patentability and the meaning of section 102(b) have been given a number of different interpretations by legal scholars and the courts.

Patent Law

One, policy position is that inventions implemented in software should not be statutory subject matter. It has been argued that the "mental steps" doctrine can be used to exclude software implementations from the patent system.[154] Under this doctrine, processes that could be performed using pencil and paper are not statutory. The U.S. Supreme Court in its *Benson* opinion wrote that a computer does arithmetic ''as a person would do it by head and hand. ''155 In the late 1960s PTO used the mental steps doctrine to deny patents to inventions that used software.

The view that inventions that use software are only statutory if they are traditional industrial processes that transform matter may also be supported by the case law. In *Benson, the* Supreme Court, relying on a series of cases from the 1800s, wrote that "[t]ransformation and reduction of an article 'to a different state or thing' is the clue to patentability of a process claim. ''[156] However, the Court did goon to say that it was not holding that no

153 ''. . . the ideas and principles underlying a program ean frequently be discovered in other ways---ways that are legitimate. Examples are studying published documentation, performing timing tests and observing the inputs, outputs, and conditions of operation," Victor Siber, op. cit., footnote 140, p. 8.

154 Pamela Samuelson, "Benson Revisited, " *Emory Low Journal*, vol. 39, No. 4, fall 1990, p. 1047-1048.

155 *Gottschalk* v. *Benson,* 93 S. Ct. 253,254.

156 93 S. Ct. 253,256.

process patent could ever qualify if it did not operate to change articles or materials.[157]

Legal arguments can also be used to support the position that some of what are now deemed nonstatutory mathematical algorithms should be patentable. These arguments are based on the fact that the Supreme Court appeared to view the *Benson algo*rithm as a ‘‘law of nature. ’ Some have argued that the Benson algorithm was not the mathematical expression of a scientific truth, such as F=ma expresses the relationship between force, mass, and acceleration, but a man-made solution to a complex problem.[158] According to this interpretation of patent law, industrially useful processes should not become unpatentable merely because they can be described mathematically.[159]

Copyright Law

The scope of copyright protection for computer programs depends in part on the interpretation of the meaning of section 102(b) of the Copyright Act. With all works courts must engage in the process of drawing the line between protectable expression and unprotectable ‘‘ idea[s], procedures], processes], system[s], method[s] of operation, concepts], principles], or discoveries].’’[160] This e$_{xerc}$ise becomes more critical, and difficult, in the context of fact-based works, such as history texts and instruction manuals, and ‘‘functional’ works, such as blueprints or computer programs,

There are a number of different views of the application of existing law to user interfaces. One interpretation of the law is that user interfaces are inherently functional and therefore not copyrightable subject matter. According to this interpretation, user interfaces are in the domain of patent law,[161] protected only to the extent that elements are novel and nonobvious. This argument would support a policy position that sharply limits the scope of protection for user interfaces.

The other view is that user interfaces may be protected by copyright. One approach has been to protect the user interface screen displays as audiovisual works or compilations of literary terms.[162] The screen displays are considered a separate work from the program code. As for all works, the scope of protection for the audiovisual work or compilation is determined by an interpretation of section 102(b). One interpretation is that the command terms are ‘ideas’ and that only their arrangement on the screen is protected expression.[163] Protection of the command terms themselves can be supported by an interpretation of section 102@) in which the unprotected ‘ ‘idea’ is at a higher level of abstraction, such as the overall purpose of the program. Any design choices not necessary to the purpose of the program, including the choice of command terms, would then be protected expression.

A second approach to protecting user interfaces through copyright law is to consider the user interface as protected by the copyright in the program. The user interface is viewed as part of the ‘‘structure, sequence, and organization’ of the underlying program.[164] This arguably represents a different interpretation of the meaning of structure, sequence, and organization’ from the way in which the term was used in *Whelan:* `it` is possible to create two programs that have identical user interfaces but use different subroutines and data structures (the elements that contributed to the court’s finding of similarity of SS0 in *Whelan).*

The term ‘structure, sequence, and organization’ has been criticized for failing to distinguish between the ‘‘static’ structure of the program-the program code—and its ‘‘dynamic’ structure-the ‘ ‘behavior’ of the program when loaded into the computer

15793 S. Ct. 253,257.

158 Laurie, op.cit., footnote 61, p.257.

159 William L, Keefauver, ‘ ‘The Outer Limits of Software Patents, in Morgan Chu and Ronald S. Laurie (eds.), *Patent Protection for Computer Software* (Englewood Cliffs, NJ: Prentice Hall Law and Business, 1991), p. 83.

160 17 U.S.C. 102(b).

161 Steven M, Lundberg, Michelle M, Michel, and John P. Sumner, ‘ ‘The Copyright/Patent Interface: Why Utilitarian ‘Look and Feel’ Is Uncopyrightable Subject Matter, ” *The Computer Lawyer, vol. 6, No.* 1, January 1989.

162 *Digital Communications Associates* v. *Softklone*, 659 F. Supp. 449 (N.D. Ga. 1987).

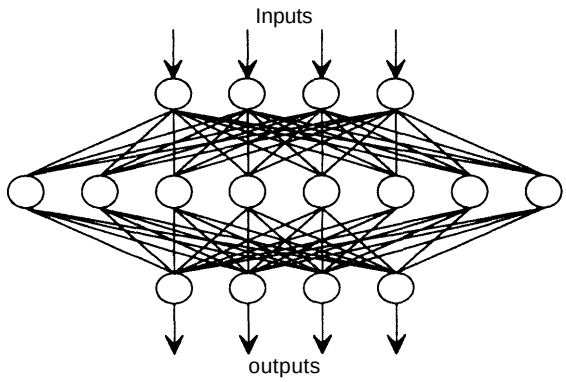
163 Ibid.

164 *Lotus* v. *Paperback, 740* F. Supp. 37, 80 (D. Mass. 1990).

Box 4-C—Neural Networks

Neural networks are a special kind of computing architecture.[1] The network consists of a large number of interconnected processing elements, arranged in layers (see figure 4-C-l). The relationship between the input and output of the network is determined by the internal details of the network. Signals passing between the layers of the network are modi.tied by multiplying them by "weights." These weighted signals are received as inputs by the processing elements. The processors compute an output value, which is a function of the sum of the inputs, and passes the output to the next layer of processors.

Figure 4-C-1—Neural Network



The circles represent processing elements that perform a weighted **sum on their inputs and compute an output value that is then sent to the next layer of processing elements.**

SOURCE: OTA.

The weights determine the overall behavior of the network, much as the program code determines the behavior of a conventional computer. However, neural networks are not programmed in the same way as conventional computers. Neural networks are ''trained. The network is presented with input values for which the desired output is known. The network then adjusts the weights until this output is achieved. This process is repeated for a large number of input and output examples, called a training set. Given enough examples, the desired behavior of the network can be achieved for a wide range of inputs. One focus of research on neural network applications has been "pattern recognition" problems such as recognizing handwritten characters: the input is image data, and the output indicates which letter has been "read."[2]

Because neural networks are different from conventional computers, there is some uncertainty about the application of intellectual property laws.[3] One issue is the copyrightability of the set of weights. For example, do the weights satisfy the Copyright Act's definition of a computer program? Can the set of weights be said to be a work of authorship? One could argue that the network, and not a human, actually authors the weights (see box 4-A on authorship). On the other hand, the network could be regarded simply as a tool used by a human author-the author chooses the training set and presents the data to the network. The first copyright registration for a set of neural network weights issued in October 1990.

A second question is whether protecting the weights alone is sufficient to protect the value embodied by the network. Much as two programs can have the same external design or input/output relationship but different program code, two networks can have the same input/output relationship but different sets of weights. The ability of a neural network to "learn" could make it easier to appropriate the value of the network—the input/output relationship-without actually copying the weights. An existing network or conventional program could be supplied with inputs and the outputs observed.[4] These sets of input and output data could then be used to train a second network, which would have similar behavior to the original network.

---

1 For an introduction to neural networks, see Judith Dayhoff, *Neural Network Achitectures* (New York, NY: Van Nostrand Reinhold, 1990).

2 Tim Studt, ''Neural Networks: Computer Toolbox for the 90's,'' *R&D Magazine*, vol. 33, No. 10, September 1991, p. 36.

3 Andy Johnson-Laird, "Neural Networks: The Next Intellectual Property Nightmare?" *Computer Lawyer*, vol. 7, No. 3, March 1990, p. 7; Gerald H. Robinson, "Protection of Intellectual Property Protection in Neural Networks," *Computer Lawyer*, vol. 7, No. 3, March 1990, p. 17; Donald L. Wenskay, ''Neural Networks: A Prescription for Effective Protection' *Computer Lawyer, vol. 8, No. 6,* August 1991, p. 12.

4 Johnson-Laird, op. cit., footnote 3, pp. 14-15.

SOURCE: OTA and cited sources.

and executed.[165] The possibility that the behavior of a program could be protected expression led to a discussion about the extent to which copyright protection might overlap with patent protection of the program function—what some have termed the "patent/copyright interface" problem.[165] The relationship between intellectual property protection of static and dynamic structure is also an issue in the context of neural networks (see box 4-C).

The same issue was addressed by the Copyright Office in 1988 when it addressed the nature of the relationship between the program code and screen displays. Hearings were held by the Office in response to the *Softklone court's* holding that the ''computer program' copyright did not extend to the screen displays. The *Softklone* court had noted that "the same screen can be created by a variety of separate and independent computer programs. [167] At the hearings the IEEE Computer Society supported separate registrations of the program code and screen displays, arguing that the nature of the ''authorship '' in the program code was fundamentally different from that in the screen displays.[168] However, the Copyright Office ruled that a single registration of a computer program covered any copyrightable authorship in the program code and the screen displays, writing that ''the computer program code and screen displays are integrally related and ordinarily form a single work. ''[169]

Software Development

Arguments about the proper interpretation of existing law also rely in part on characterizations of the software development process. Some emphasize ''creative' aspects of the development process. Just as with other copyrightable works, it is argued, this creative effort should be encouraged by limitations on copying, Others, however, characterize the development process as ''engineering, ' in an effort to limit the scope of copyright protection or to argue for the wider use of patents.[170]

Discussions of creativity and engineering can also be seen as related to the scope of available design choices. One of the goals of software engineering methodologies is to reduce the number of design decisions, as a way of managing the complexity of large projects. Elements of the development process have become more routine. High-level languages free programs from much of what Brooks calls "accidental complexity. ' '[171] Shaw points out that today "almost nobody believes that new kinds of loops should be invented as a routine practice. ' '[172] Subroutines, macros, and operating systems have also been used to avoid ''re-inventing the wheel. " The concept of reuse (see box 4-D) may also make parts of the development process more routine. The Federal Government, particularly the Department of Defense, has shown considerable interest in encouraging reuse (see box 4-E).

165 ''Ccn@d to Dr. Davis criticism of the Whelan 'structure, sequence, and organization' formulation is the fact that there is no necessary relationship between the sequence of operations in a program, which arc part of behavior, and the order or sequence in which these operations arc set forth in the text of the program—the source code and object code. As Dr. Davis pointed out, 'the order in which sub-routines appear in the program text is utterly irrelevant, ' and the two views of a computer program, as text and as behavior, arc 'quite distinct. ' '' *Computer Associates v. Altai*, op. cit., footnote 127, p. 14.

166 See Pamela Samuelson, "Survey on the Patent/Copyright Interface for Computer Programs, " *AIPLA QJ. vol.* **17,** p. **256.** See **also** *Computer Associates v. Altai*, op. cit., footnote 127, p. 15. A study prepared jointfy by the Patent and Trademark Office and the Copyright Office concluded that there is minimal overlap between the two areas with respect to computer software. U. S. Patent and Trademark Office and U. S. Copyright Office, *Patent-Cop yright Laws Overlap Study*, May 1991, pp. ii-iii.

167 *Digital Communications* Associates v. *Softklone* Distributing, 659 F. Supp. 449, 455-456. The court then concluded, ' 'Therefore, it is the court's opinion that a computer program's copyright protection docs not extend to the program's screen displays, and that copying of a program's screen displays, without evidence of copying of the program's source code, object code, sequence, organization, or structure does not state a claim of infringement.

168 Richard H. Stem, "Appropriate and Inappropriate Legal Protection of User Interfaces and Screen Displays, Part l," *IEEE Micro, vol. 9, No. 3,* June 1989, p. 84.

169 Copyright Office, ''Registration Decision; Registration and Deposit of Computer Screen Display s,' 53 *Federal Register* 21819 (June 10, 1988).

170 For one view of the relationship between ''software engineering" and intellectual property, see Clapes, op. cit., footnote 96, pp. 119-120. For extensive discussion of the nature of software development, see Susan Lammers, *Programmers at Work* (Redmond, WA: Microsoft Press, 1986).

171 Frederick p, Brooks, Jr., ' 'No Silver Bullet, ' *IEEE Computer*, April 1987, *p.* 12.

172 Mary Shaw, "Prospects for an Engineering Discipline of Software, ' *IEEE Software*, vol. 7, No. 6, November 1990, p. 22.

Box 4-D—Software Reuse

Productivity in software development is a concern in both the private and public sectors.[1] The relatively low productivity of software programmers is a difficult problem, so one way to improve programming productivity is to "reuse" program code.[2] This would eliminate much of the redundant work of many programmers writing code that does essentially the same thing. One source estimated that of 15.3 billion lines of code written in 1990, only 30 to 40 percent represent novel applications; 60 to 70 percent represent generic computer tasks like data entry, storage, and sorting.[3]

Reuse can be either accidental or systematic. Many programmers employ "accidental' reuse, making use of some elements of their own previous work or that of their colleagues. In systematic reuse, software is written from the beginning with the intention of making it more reusable; the components are documented and put in a library. This can be time consuming, and in the short run can be more costly than writing a specific program for the immediate need. This cost has to be seen as an investment that pays off in the long run if the component can be reused several times. Software development to facilitate systematic reuse could also streamline software maintenance, which accounts for a large and increasing portion of software life-cycle costs.

When reuse is being practiced within an organization and programmers only use components from their own organization's software library, intellectual property considerations are not an issue. However, questions of ownership become more important if there is to be development of a market in reusable components. This market is growing, but is still relatively small. For example, it is possible to license libraries of code for common functions, such as components used in developing graphical user interfaces.

Intellectual property considerations can affect reuse in three ways. First, a number of participants at an OTA workshop on software engineering indicated that uncertainty about the ownership of a component or the scope of intellectual property rights could discourage the development of programs composed of components from different sources. Second, some in the reuse community think that a stable system of ownership rights[4] is necessary to encourage the investment required for creating a commercial-quality library and to handle questions of liability. In some cases in the past, the investment for widely used libraries has come from sources other than potential licensing fees (e.g., the X-windows library developed in a university research and education environment, at MIT, and later used in commercial products). A final issue is whether the interfaces in libraries of reusable components are protected by copyright law: can a competitor offer a library with the same interfaces but different implementations?[6]

There are a number of' other factors which affect the degree of reuse:[7]

1. Development standards have not been established for software;
2. There is a pervasive belief that if it is "not developed here," it can't be trusted or used by "us";
3. Software is all too often developed with respect to a specific requirement with no consideration given to reuse in other environments;
4. Many languages encourage constructs that are not conducive to reuse;

---

1 See, e.g., *The Software Challenge* (Alexandria, VA: Time-Life Books, 1988); and Albert F. Case, Jr., *Information Systems Development: Principles of Computer Aided Software Engineering (New* York, NY: Prentice-Hall, 1986).

2 See Kazuo Matsumura et al., "Trend Toward Reusable Module Component: Design and Coding Technique 50SM," in *Proceedings of the Eleventh Annual International Computer Software and Applications Conference—COMPSAC '87* (Washington, DC: IEEE Computer Society Press, Oct. 7-9, 1987), p. 45 (cited in Michael Cusumano, *Japan's Software Factories: A Challenge to U.S. Management (New* York, *NY:* Oxford University Press, 1991), p. 258).

3 David Eichmann and John Atkins, "Design of a Lattice-Based Faceted Classification System," paper presented at Software Engineering and Knowledge Engineering (SEKE '90), Skokie, IL, June 21-23, 1990.

4 Constructing licenses or pricing structures may be difficult. As with any digital information (see ch. 5), it will be hard to control what a user does with a component once a copy has been obtained. To&y, some libraries are being sold on a per-copy basis as source code with no royalties or runtime licenses, However some believe that market forces under the classical copyright paradigm-where "copies' are priced and sold—will not work properly (Brad Cox (Washington, CT), personal communication, Aug. 1, 1991). To reduce individual transaction costs, Cox suggests that use-base fees be administered collectively, similar to the way in which performance royalties for musical compositions are administered.

5 Robert W. Scheifler and James Gettys, *X Window System* (Bedford, MA: Digital press, 1990), pp. 8-15.

6 Institute for Defense Analyses, *Proceedings of the Workshop on Legal Issues in Software Reuse*, IDA Document D-1OO4 (Alexandria, VA: Institute for Defense Analyses, July 1991).

7 From Eichman and Atkins, op. cit., footnote 3.

5. Software engineering principles are not widely practiced and consequently, requirements and design documents often are not available with the code; and
6. No widely accepted methodology has been developed to facilitate the identification and access of reusable components.

There is an ongoing body of research designed to: 1) identify characteristics of software components that make them suitable for reuse, *2)* identify techniques to translate a software component with marginal reuse potential into one that can be reused, and 3) develop systems for classifying and identifying software components to make it easy to retrieve them from databases when they are needed.[8] Among the systems being considered are artificial intelligence programs capable of browsing libraries of programs, rating their qualit y according to several reusability y criteria (e.g., modularity, cohesion, size, control structure), and indicating those most suitable for reuse.[9] Much of this work is being done for, or in conjunction with, the Department of Defense (DOD), especially the Defense Advanced Research Projects Agency. As a major user of software, DOD has an interest in improving its own and its contractors' productivity through fostering reuse of software.

Reuse is more common among some major software users in Japan. Of several firms surveyed, Toshiba reported the most reuse with 50 percent of its delivered custom applications software being made of reused components.[10] Toshiba has made software reuse a central strategy for increasing productivity and reliability while reducing costs. Reuse is a high priority for both managers and programmers, Managers are rated on how well their projects have met reuse targets, as well as more usual measures like schedules and customer requirements. Programmers are required to report periodically on how many components they have used from, or contributed to, the reuse database; the company rewards authors of successful components that are frequently reused by others. Toshiba has also developed a specialized tool, OKBL (object-oriented knowledge-based language), which helps users classify components for storage in, or retrieval from, departmental libraries. Users can also locate components using printed catalogs. Most reuse, even at Toshiba, is within families of related products: less than 10 percent of software is reused across departmental lines.

8 Ibid. See also V.R. Basili, H.D. Rombach, J. Bailey, A. Delis, F. Farhat, "Ada Reuse Metrics," and R. Gagliano, G.S. Owen, M.D. Fraser, K.N. King, P.A. Honhanen, "Tools for Managing a Library of Reusable Ada Components," paper presented at Ada Reuse and Metrics Workshop, Atlanta, GA, June 15-16, 1988.

9 Jan Carlos Esteva and Robert G. Reynolds, "Learning To Recognize Reusable Software by Induction," paper presented at Software Engineering and Knowledge Engineering (SEKE '90), Skokie, IL, June 21-23, 1990.

10 Cusumano, op. cit., footnote 2, p. 261.

SOURCE: OTA and cited sources.

However, despite these advances, Brooks argues that part of software development will always be a creative process.[173] After reviewing the development of software engineering, he concluded that while the difference between poor conceptual designs and good ones may lie in the soundness of the design method (and cart be addressed by progress in software engineering), changes in methodology cannot bridge the gap between a good design and a great one. This, Brooks concluded, requires great designers.[174]

Debate continues within the field concerning the extent to which computer science should be characterized as a science or as an engineering discipline, its maturity as a discipline, and the appropriate content of undergraduate education in the discipline.[175] Some recent efforts have presented a formal definition of the discipline, its methodologies, and

173 Brooks, op. cit., footnote 171, p. 18.

174 Ibid.

175 For some recent discussion of these topics, see David Gries et al., "The 1988 Snowbird Report: A Discipline Matures," *Communications of the ACM*, vol. 32, No. 3, March 1989, pp. 294-297; Nor-man E. Gibbs, "The SEI Education Program The Challenge of Teaching Future Software Engineers," *Communications of the ACM, vol.* 32, *No.* 5, May 1989, pp 594-605; and Edsger W Dijkstra et al., "A Debate on Teaching Computer Science," *Communications of the ACM, vol.* 32, No. 12, December 1989, pp. 1397-1414.

Box 4-E—Special Concerns of the Federal Government

As a major user and developerof software, the Federal Govemment has special concerns with regard to future trends in software development. Due to the variety of missions of government agencies, its software needs span the gamut from small standard packages (word processing, spreadsheets, graphics) to large, specialized mission-critical systems (air traffic control, hospital information systems, military command and control) and nearly everything in between. Concerns include procurement policies, development of large composed systems, and technology transfer.

Procurement

Government procurement of computer hardware and software has been a complex and controversial subject for a long time. The government strategies for acquiring and managing information technology have been in a state of flux since passage of the Brooks Act of 1965,[1] which was enacted to establish procurement and management policies. Among concerns that have generated this flux are: 1) tension between the rapid pace of change in agency needs and improvements in technology versus the slow pace of the planning and procurement process; and 2) the tension between agency desires to ensure compatibility between systems and congressional desires to ensure competition among vendors.[2]

Software Development

Many government agencies are supported by software systems that are critical to performance of the agency's mission. These large systems, to be successful, require a good match between planning and assessment of technology needs and the acquisition or development of the hardware and software to match those needs.[3] In creating their systems, agencies face the choice of developing their software in-house, attempting to purchase ‘ ‘off-the-shelf packages to meet their needs, contracting with outsiders to develop customized software for them, or some combination of the three.

Once systems are in place, the complexities of the procurement process often ensure that they stay in place a long time. For these complicated systems, modifications and updates over the years make the software extremely complex and difficult to maintain. For example, the Social Security Administration's (SSA) system, in place since the early 1960s, had to be modified to reflect changes in benefits mandated by 15 laws passed between 1972 and 1981. Time allowed to make the changes was always inadequate, many mistakes were made, and backlogs became a recurrent problem, By 1982 the SSA faced the possibility of a ‘‘potential disruption of service' due to software deficiencies, yet by 1986 a system modernization program was still mired in political and legal problems and had barely begun.[4]

1 Public Law 89-306.

2 A detailed study of options for management of government information resources is found in U.S. Congress, office Of Technology Assessment, *Federal Government Information Technology: Management, Security, and Congressional Oveersight*, OTA-CIT-297 (Washington, DC: Government Printing Office), February 1986.

3 OTA has taken a close look at software development/proc urement problems at several agencies, including Federal Aviation Administration% Social Security Administration and Veterans Administration: U.S. Congress, Office of Technology Assessment, *Review of FAA's 1982 National Airspace System Plan,* OTA-STI-176 (Washington, DC: U.S. Government Printing Office, August 1982); *The Social Security Administration and Information* Technology, OTA-CIT-311 (Washington, DC: U.S. Government Printing Office, October 1986); *Hospital Information Systems at the Veterans' Administration,* OTA-CIT-372 (Washington, DC: U.S. Government Printing Office, October 1987).

4 U.S. Congress, Office of Technology Assessment, *The Social Security Administration and Information Technology,* op.cit., footnote 3.

its characteristics (see box 4-F). In 1988 an Association for Computing Machinery/IEEE Computer Society Task Force on the Core of Computer Science developed its detailed definition of the discipline through three paradigms: *theory* (rooted in mathematics), *abstraction or modeling* (rooted in the experimental scientific method), and *design* (rooted in engineering).

Government has a particular need, in future generations of software systems for well--engineered, maintainable software. An additional need is for tools and methods to plan for future software needs and ability to match technology to those needs in a timely manner. Several government projects aim at bringing government, industry, and academic research to bear on these projects. For example, a program called Software Technology for Adaptable, Reliable Systems works with industry to develop new software tools and methods. Part of this multiyear effort was the establishment of the Software Engineering Institute at Carnegie Mellon University which has done research on software reuse and other ''software factory' methods.

Technology Transfer

Software developed by the Federal Government may not be copyrighted. Under section 105 of the Copyright Act, copyright protection is not available for any work created by the Federal Government. Section 105 was enacted to give the public unlimited access to important information, to prevent the government from exercising censorship, and to prevent the government from using copyright in government works as a shield that would prevent selected groups from acquiring information.[5] In addition, it is argued that the public has paid for the creation of the work through taxes and should not pay a second time by paying copyright royalties.

Some propose that exceptions to the provisions of section 105 be made for computer programs, arguing that copyright protection for government software would facilitate its transfer to the private sector.[6] According to this view, private sector firms that might be interested in developing and marketing products based on government-developed software would be more likely to invest in the ' 'commercialization' of the government software if they were assured of an exclusive license.[7] Similar considerations have motivated government policy with respect to patents granted to the Federal Government. Opponents of an exception being made for computer programs argue that the exception is the ''thin end of the wedge," which could lead to further exceptions to section 105. In addition, it has been suggested that the line between programs and information' or 'data' is not always clear, and that granting exclusive rights to 'programs' could have the effect of limiting access to 'data' which would be retrieved using the programs.[8]

Legislation introduced in the 102d Congress would permit limited copyrighting of government software. H.R. 191 and S. 1581 would allow Federal agencies to secure copyright in software prepared by Federal employees in the context of cooperative research and development agreements (CRADAs) with industry.

---

5 Ralph Oman, Register of Copyrights, testimony at hearings before the House Subcommittee on Science, Research and Technology, Apr. 26, 1990, Serial No. 117, p. 100.

6 John M, Ols, Jr., Director i, the Resources, Community, and Economic Development Division, General Accounting Office, testimony at hearings before the House Subcommittee on Science, Research, and Technology, Apr. 26, 1990, Serial No. 117, p. 44.

7 Ibid., p. 41.

8 Steven J. Metalitz, Vice President and General Counsel, Information Industry Association, testimony at hearings before the Senate Committee on Commerce, Science and Transportation, Sept. 13, 1991.

SOURCE: OTA and cited sources.

Box 4-F—The Discipline of Computer Science

In March 1991, the Association for Computing Machinery (ACM) and the IEEE Computer Society (IEEE-CS) published a joint report on recommendations for undergraduate curricula in computer science. The report, ***Computing Curricula 1991,*** was prepared by the ACM/IEEE-CS Joint Curriculum Task Force and was intended to present "current thinking on goals and objectives for computing curricula. " The curriculum recommendations in the report built upon nine areas comprising the subject matter of the discipline:

1. algorithms and data structures,
2. architecture,
3. artificial intelligence and robotics,
4. database and information retrieval,
5. human-computer communication,
6. numerical and symbolic computation,
7. operating systems,
8. programmingg languages, and
9. software methodology and engineering,

In preparing this report, the task force drew upon the comprehensive definition of the discipline of computer science presented in 1988 by the ACM/IEEE-CS Task Force on the Core of Computer Science. In its 1988 report, ***Computing as a Discipline, the*** Task Force on the Core of Computer Science noted that it had extended its task to include computer engineering, as well as computer science because there was not fundamental difference between the core material for the two fields; the difference between them is that "computer science focuses on analysis and abstraction; computer engineering on abstraction and design. " The task force's definition of the "discipline of computing" included all of computer science and engineering:

> The discipline of computing is the systematic study of algorithmic processes that describe and transform information: their theory, analysis, design, efficiency, implementation, and application. The fundamental question underlying all of computing is, "What can be (efficiently) automated?"

Concerning the role of programming languages, the Task Force on the Core of Computer Science had noted that the notion that "computer science equals programming" is misleading because many activities (such as hardware design, validating models, or designing a database application) are not "programming."Therefore, it concluded that computer science curricula should not be based on programming.Nonetheless, the task force did recommend that competence in programming be part of the curricula because:

> It is. . clear that access to the distinctions of any domain is given through language, and that most of the distinctions of computing are embodied in programming notations.

SOURCES: ACM/IEEE-CS Joint curriculum Task Force, *Computing Curricula 1991 (New York NY:* Association for Computing Machinery. 1991); "A Summary of the ACM/IEEE-CS Joint curriculum Task Force Report," Communications of the ACM, vo1. 34, No. 6, June 1991, pp. 69-84; and Peter J. Denning et al., "Computing as a Discipline,' *Communications of the ACM,* vol. 32, No. 1, January 1989, pp. 9-23.

Chapter 5

Digital Information and Copyright

Contents

	Page
Introduction	161
Electronic Publishing	161
Print-Based Electronic Publishing	162
Nonprint Electronic Publishing	165
Copyright Issues for Digital Information	170
What Is a “Work”?	171
Originality and Authorship	173
Use of Digital Information	175

Boxes

Box	Page
5-A. Storing and Retrieving Dada,	162
5-B. Storing Text and Images	163
5-C. Digital information and the Scholarly Publishing System	167
5-D. One User’s View	171

chapter 5

Digital Information and Copyright

Introduction

On an optical disc there might be recorded the sounds of a hit song, the pictures at an art exhibition, or the complete contents of an encyclopedia. In any case the disc looks the same--all that is visible to the naked eye is a silver disc with rainbow highlights. Information is encoded on the disc in digital form. Digital information is weightless, compact, and essentially invisible. To make it visible, people need tools—not just hardware, like a computer terminal or a disc reader, but also software tools. Some of the software runs the hardware to make the information visible or audible, but equally important is the software that sorts and navigates so that a person can find the particular fact that is wanted. A single disc might hold the equivalent of 275,000 pages of text.

People are becoming increasingly familiar with digital information and the software to manage it. Word-processing, spreadsheet, database, and graphics programs are among the most popular types of software used on personal computers; many people use them at home as well as on the job. Word-processing software helps a person create and manipulate text information, store it for future retrieval, or print it out on paper. Spreadsheets help people to manipulate numbers and perform calculations. Database management programs are used to organize collections of data—from something as small as a personal address file to something more complex like the billing records of a major corporation or the medical records of the patients in a hospital. Graphics software can be used to make free-hand drawings or graphs and charts illustrating mathematical data. Different kinds of digital information need to be stored and used in different ways, so many more sophisticated software tools are being developed to capture information, organize it, store it, and then find it, filter out irrelevancies, and put it in useful form. (See box 5-A on storing data and box 5-B on storing text and images.)

The line between "software" (or computer program) and the "data" to be manipulated by the software is sometimes hard to discern. For example, "knowledge bases," the databases used in artificial intelligence programs, are not mere lists or records of facts. The knowledge base itself includes structured rules and relationships needed for making decisions. New programming techniques, like object-oriented programming, further blur the distinction between program and data by putting more of the intelligence of a system into the organization of the data and by making the data "active." In object-oriented programming, an abstract data type, called an object, contains both data and methods (a method describes the algorithm the object uses to solve a problem).[1] Thus object-oriented databases can represent a blurring of the program and the data it manipulates (or better stated, object-oriented databases illustrate the essential equivalence of program and data).

Though the line between data and software is blurring, it is still possible to distinguish data or digital information and the concerns that have been raised with respect to intellectual property. Digital information is not just numbers or words, it can be text, motion pictures, or music. Some of the concerns closely parallel those raised for software in other chapters. Others are unique. To understand the issues, it is necessary to quickly review the growing field of digital information or electronic publishing to see the range of technologies and products that are becoming available.

Electronic Publishing

There is a growing commercial markct for digital information. The term "electronic publishing covers a wide range of processes, products, and services ranging from traditional books and printed materials to works that are available only in electronic form. Digital information and computer technology is revolutionizing the publishing industry. In addition to commercial producers, a growing number of businesses and government agencies are creating, storing, and using documents in digital form.

[1] Phil de Lisle, "Putting OOP to Work for You," *Data Based Adviser*, vol. 9, No. 3, March 1991, p. 87.

Box 5-A—Storing and Retrieving Data

A database management system is a collection of files and a set of computer programs that allow users to efficiently modify and retrieve the files. Data may be organized in a number of different ways, but the important point about a database management system is that it allows users an abstract view of the data; that is, most users need not understand the details of where and how each piece of data is stored and maintained. The software is designed to offer different users a view of the data that is appropriate to the task that user is performing. Many programming languages for database management systems use the notion of a ***record as*** **a** basic unit of organization. In a bank's database, for example, each customer would be represented by a record, and within each record would be several fields--name, account number, balance, address, zip code, etc. Even though the information is only stored once, different employees can retrieve and use it in many different ways; to notify all customers in a certain neighborhood of the opening of a new branch, to send overdraft notices to appropriate accounts, or to make an address change to a specific account.

There are a number of approaches to structuring database systems. The *relational model* represents data and relationships among data by a collection of tables, each with several columns with unique names such as name, account number, balance, etc. A number of separate computer programs are associated with the database to allow users to perform transactions involving the data, e.g., paying interest or deducting withdrawals.

An *object-oriented* database is based on a collection of *objects in* which are stored *instance variables* and *methods. In* a bank's database, an account object would contain instance variables for name, account number, balance, etc. An important difference between object-oriented and relational databases is that in object-oriented databases, the software instructions for making use of an object are contained in the object itself. These instructions are called ***methods.*** Methods are bodies of computer code that can act on the object or cause the object to behave in certain ways under appropriate circumstances. For example, an account object may contain within it a method ***called pay-interest*** which adds interest to the balance. If the interest rate changes, or the bank introduces a new policy on paying interest, the method called pay-interest can be modified without affecting other parts of the object.

SOURCE: Adapted from Henry F. Korth and Abraham Silberschatz, *Database System Concepts (New York, NY:* McGraw Hill, Inc., 1991), pp. 1-21.

Print-Based Electronic Publishing

Many books and periodicals are now written, edited, and typeset on computers; they are only committed to paper in time to be delivered to the end user. Computer-aided publishing systems offer many advantages to publishers: information only has to be typed once and then the captured keystrokes can be edited, corrected, rearranged, or updated with relative ease. Pages can be laid out and "pasted up" directly on the computer. Graphics can be inserted, either by using graphics software to draw them on the computer, or by using a scanner to make digital versions of printed photographs or drawings. Then all the text and graphic information can be converted to a form usable by computer-based typesetters and ***in*** some cases, computer-driven printing machinery.[3] Newspapers were at the forefront of implementing computer-aided publishing,[4] but the techniques have rapidly spread to magazine, journal, and book publishing as well. Fairly low-cost desktop publishing systems, using personal computers and laser printers, have even brought these abilities to small businesses, community groups, and schools.

The easier manipulation of digitized information means that many different products can be derived from basically the same information. For example, magazines and newspapers can more easily produce different regional editions of the same issue. While the bulk of the text may be the same, articles or advertisements of purely local interest can be

[2] Electronic publishing does not have a single definition. Some people insist that "print-based' electronic publishing does not exist, and that only processes that deliver information in electronic form directly to the end user should be called electronic publishing. More often, however, use of computer technology and digital information in early stages of creation of printed materials is considered to fall under the electronic publishing rubric. For a discussion, see Michael R. Gabriel, *A Guide to the Literature of Electronic Publishing* (Greenwich, CT: JAI Press, Inc., 1989), pp. 1-14, and Oldrich Standera, *The Electronic Era of Publishing (New* York, NY: Elsevier, 1987), pp. 6-10.

[3] Oldrich Standera, Th, *Electronic* Era *of Publishing: An Overview of Concepts, Technologies and Methods* (New *York, NY:* Elsevier Science Publishing, 1987), p. 157.

[4] Ibid., p. 110.

Box 5-B—Storing Text and Images

Text storage and retrieval: Text retrieval system, text base management system, or text data management system are terms for computer-based storage and retrieval systems that store documents in machine-readable, character-coded form so that they can be retrieved by a user or processed by a computer.[1] Text databases differ from other kinds of databases described in box 5-A mainly in that the fields are very large-often the size of a whole article or book. Text databases also differ from word processing systems, which also store text in character-coded form, in that text retrieval systems have much more powerful capabilities for creating indexes to significant words in the text and giving rapid access to text segments that contain the specified character strings. The software of text retrieval systems typically allows a user to search the entire text database (or text base) for all occurrences of a specific word or a phrase. More complex searches may be constructed using Boolean logic operators (e.g., using the terms AND, OR, and NOT to restrict or broaden a search), wild cards (e.g., searching for "creat**" to locate all instances of "creator", "creative", ''creativity", etc.), proximity matching (searching for ''nerve" only if it appears within 20 words of the word "brain' '), and other search tools.

Providers of on-line information developed proprietary text retrieval software for use on their own mainframe computers beginning in the mid-1960s and 1970s, with the earliest implementations limited to relatively brief documents like bibliographic citations and abstracts. Today text systems exist to handle documents of virtually any length, and software is commercially available to run on hardware of many kinds, including personal computers. Text retrieval systems are coming into use by corporations and government agencies to manage internal libraries of letters, reports, legal briefs, and other documents. Text retrieval software is also used in scholarly work to analyze electronic versions of books and other documents. Some systems allow users to create notes and annotations that can be electronically linked to specific areas of the text.

Documents may be put into the text base through direct keyboard entry, by transferring text files from other computers and word processors, or by converting printed documents to digital form through optical character recognition (OCR). Of these three input methods, transferring files is the easiest—most text retrieval systems are designed to ' 'import' digital text from any source. For documents that exist only on paper, scanning with OCR can often turn out to be less expensive than keyboarding, but it is not yet a problem-free method of converting printed text to digital text. Changes in type style or blemishes on the paper can cause scanning mistakes that must be found and corrected by keyboard. Despite advances in sc arming technology in recent years, the conversion problem prevents many organizations from replacing paper archives with digital text bases.

Although online document storage saves space compared to storage of paper letters, legal documents, books, or reports, it still requires disc space. Data compression algorithms and more sophisticated indexing algorithms are aiding in reducing these storage requirements. Some compression techniques can reduce data storage space requirements by as much as 75 Percent.* Improved indexing algorithms allow both for faster searches and reduced storage needs for the index database (a complete index of all occurrences of all words in a document can be nearly as long as the document itself unless space-efficient methods are used).[3] Text bases can be stored on both magnetic and optical discs.

Image storage and retrieval: Image storage captures a document's appearance, rather than its content. An electronic document imaging system uses a scanner to convert documents to a form that can be stored digitally. The most widely used scanners divide the document into many tiny areas called pixels (picture elements), measure the light reflected from each pixel, and send a corresponding electrical signal to image processing circuitry which converts the signal to a stream of digital code.[4] The scanners are fairly reliable and simple to operate compared to OCR. This easier input task is an advantage, as is the ability to copy the exact appearance of a page of text, including any associated photos or graphics. A disadvantage is that text stored in image form usually can not be directly manipulated, analyzed, or searched by text retrieval software; each image must be properly categorized and linked with index terms when it is entered into the system. Images also require a great deal of disc storage space even after

1 William Saffady, *Text Storage and Retrieval Systems:* A *Technology Survey and Product* Directory (Westport, CT: Meckler Corp., 1989), p. 3.

2 Ibid., p. 25.

3 See, e.g., Dennis Allen, ''Text Retrieval Witb a Twist," *Byte,* July 1989, pp. 201-204.

4 For more information on image storage, see William Saffady, *Personal Computer Systems for Automated Document Storage and Retrieval* (Silver Spring, MD: Association for Information and Image Management, 1989), pp. 19-27.

Continued on next page

Box 5-B—Storing Text and Images-Continued

processing by compression algorithms. High-capacity optical discs are often the storage medium of choice for image databases. Not only pictures of documents but also drawings, photographs, and images produced by some medical diagnostic equipment can be stored in this way. Images are retrieved by searching the index database in many of the same ways mentioned above (words, phrases, Boolean search strategies, etc.). Retrieved images may be viewed on a high-resolution screen or printed out on a laser printer or other appropriate printer.

Storing and exchanging compound documents: There are many different types of documents and different formats for storing data. Documents produced by one word processing software package, for example, may not be readable by another because they use different conventions for indicating format changes (type styles, paragraph indents, boldface, etc.). Format becomes increasingly important for documents produced by sophisticated publishing software or by multimedia systems—these documents may include extensive format information about type fonts, photos, graphics and other non-text information. If a document is stripped of this format information and reduced to its simplest terms (e.g., when a text document is converted to a file of ASCII[5] characters) it may be read by a wide variety of software packages, but so much formatting information maybe lost that the document is useless for certain purposes. Even if it is still readable, a document so treated may no longer be revisable, that is, it cannot be edited or updated.

There are a growing number of situations in which fully formatted, revisable digital documents need to be exchanged by organizations that use different software. Much architectural and mechanical design work, for example, is done on computers. Exchanging and storing drawings, specifications, manuals, and other documents in electronic form rather than on paper can reduce storage and maintenance costs. (For example the B-1 Bomber reportedly has over 1 million pages of documentation, 200,000 pages of which must be updated yearly. The Department of Defense, through its Computer-Aided Acquisition and Logistics Support strategy—CALS--is in the process of requiring that all information submitted by contractors developing weapons systems be submitted electronically. [6])

Interchange of complex documents, independent of particular software or hardware environments requires another level of standardization-standard ways to describe how data was handled by the originating software so that the receiving software can handle it in an appropriate and compatible manner. An approach to this problem is use of a‘ ‘document description language’ or ‘metalanguage’; the ‘‘tags’ or ‘‘labels’ generated by the document description language are included in the document, A person reading the text mayor may not see these tags; their purpose is to describe the document to the receiving software. Several such languages have been developed and are competing in the marketplace; no clear standard has yet emerged. Use of document description tags within the document greatly increases the size of the document and thus the storage space requirement on disc or other storage medium,

5 ASCII stands for American Standard Code for Information Interchange. It is the standard 7-bit code for transferring information On local and long distance telecommunications lines.

6 Brooke Stoddard, ‘‘Shedding the Burden: The Federal Government Spent $130 Million on Electronic Image Processing Last Year, ” *Government Computer News,* Apr. 29, 1991, p. S4.

inserted in the different editions. The completed electronic text can be sent via telephone lines to several printing and distribution facilities throughout the country, thus reducing mailing costs and speeding delivery to subscribers.

In the book publishing arena, materials can be customized ‘ ‘on demand. For example, publishers can respond to requests from college professors to create textbooks that only include those chapters that are actually to be used in their courses. By selecting text chapters from an electronic database, publishers can make several different versions of a standard text for different universities. Authors and publishers also find it easier to keep books up to date or to add new chapters if the books are stored electronically until it is necessary to print them.

Once information has been placed in digital form for publishing, it is also possible to use it for other purposes. Some newspaper and magazine publishers also sell electronic versions of their publications in some of the forms discussed below.

Nonprint Electronic Publishing

The other type of electronic publishing, and the one that will get the most attention in this chapter, is the type in which information is delivered to the end user in electronic form.

On-line Information

One segment of this market is the on-line information retrieval service where end users retrieve information from databases housed in some distant computer via telephone lines or computer networks. In 1988 it is estimated that the four largest such services had a total of 1.3 million subscribers and annual sales of over $560 million.[5] Costs to subscribers are usually based on a subscription fee plus usage charges for actual use of particular databases, citation charges for printing or downloading some results, and telephone line charges.

The vast majority of subscribers to these commercial services are organizations—businesses, libraries, universities, etc.—whose employees need access to such services. Ten minutes of on-line searching might cost anywhere from $3.50 to $14.00[6] or more. Costs can mount up quickly if searches are complex, if expensive databases are used, or if users are not thoroughly trained. Other services aiming to give home users affordable on-line information services (as well as home shopping and entertainment) are currently a small part of the market in the United States. In Europe, Australia, and a few other countries, on-line information services intended for the home user are provided by the telephone company or a government agency and are usually called videotex.[7]

Most on-line information services offer their subscribers access to databases provided by a variety of publishers. The services negotiate contracts with the owners of the databases for the right to distribute them, and pay royalties based primarily on how much the database is used. Having access to many databases through one service is an advantage to users, allowing them to search more economically. For example, the largest service, DIALOG Information Retrieval Service, offers over 400 databases in four categories numeric data, directories bibl bibliographic records. and full-text records. Examples of numeric databases include, for example. stock and bond price quotations as well as many kinds of statistical and financial information, some based on government statistics. Directory databases include many standard reference works and handbooks, many of which are also published in hard copy. Bibliographic databases have citations to journal, magazine, or newspaper articles and sometimes also include abstracts of the articles cited. Full-text databases contain electronic versions of magazine, journal, and newsletter articles; they may be collections of articles from many sources, or actual electronic editions of full journall or magazine,

While many elcectrnically published journals are digital versions of the 'paper" journal, electronic publishing is gradually growing and changing the way informal ion is exchanged among research communities. Ten refereed electronic journals are now available on the Intcrnet,[8] a collection of research and information networks through which many universities. businesses. and government agencies share services 1ike electronic mail, file exchange, and access to distant computers,[9] Within the Internet community, and on other computer networks, them is a growing camp of researchers who view electronic publishing as an ongoing interactive process. Some experimental journals on the lnternet use publication of text as a tool to e1icit comment from other researchers.) One on-line project on the Genome Project at the Welch Library at Johns Hopkins University mounts text related to the Genome Project on an on-line database. Geneticicists students, and critics from around the country can

[5] Mary Lu Carnevale and Julie Amperano Lopez, "In Electronic Publishing, All Eyes Are Fixed on AT&T," *The Wall Street Journal*, Aug. 2, 1989. The four services are Lexis-Nexis-Medis, owned by Mead Corp.; Dialog Information Service, owned by Knight-Ridder; Compuserve Information Service, owned by H&R Block; and Dow Jones News/Retrieval Service, owned by Dow Jones & Co.

[6] These examples include discounts for high volume use. From *The DIALOG Service*, user catalog, spring 1991 (New York, NY: Knight Ridder, Inc., spring 1991).

[7] Gabriel, op. cit., footnote 2, pp. 77-88.

[8] Association of Research Libraries, "Statement of the Association of Research Libraries to the Subcommittee on Science, Technology, and Space, Senate Committee on Commerce, Science and Transportation for the Hearing record of March 5, 1991 on S. 272—The High-Performance Computing Act of 1991," March 1991, Washington, DC. In "refereed" journals, submitted articles are reviewed by a committee of scholars before publication. Publication in refereed journals generally carries more prestige in scholarly communities than publication in other journals or magazines.

[9] For more information on the Internet and on the future evolution of a National Research and Education Network, see U.S. Congress, Office of Technology Assessment, *The National Research and Education Network* (Washington, DC: U.S. Government Printing Office, forthcoming).

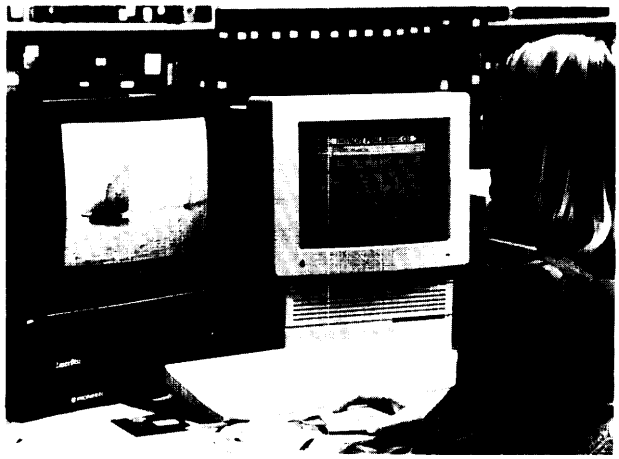


Photo credit: U.S. Library of Congress, American Memory Project

Multimedia databases can give users access to information in several forms-e. g., text, graphics, motion pictures, or sound recordings.

access the text and respond to the authors via electronic mail. The output in this case is an on-line textbook which changes constantly to reflect advances in the field.l" (See box 5-C for a discussion of how electronic publishing may change scholarly publishing.)

CD-ROM

Another increasingly popular way of marketing digital information is through publication of CD-ROMS. CD-ROM (compact disc, read-only memory) is an optical storage medium. CD-ROMs are essentially the same as the compact discs now used for musical recordings, although the data are stored in a different format and require a different player.[11]

Many of the databases available through on-line services are also available on CD-ROM. CD-ROMs are often made available to library patrons. Some of the databases on CD-ROM offer more ''hiendly" user interfaces than do on-line services and inexperienced or occasional users can search at their own pace without accruing huge bills for 'connect time' and telephone usage. However, the user might have to worry about holding up another user, since licensing requirements or equipment limitations often mean that only one person at a time can use a particular disc. Licenses are becoming available that allow multiple users to access CD-ROMs on local area networks; while a few vendors do not charge extra for network licenses, in most cases these are much more expensive than single user licenses.

CD-ROMs will become increasingly popular as more titles are offered and as the price of players and discs fall. Recent developments that simplify creation of CD masters (from which individual discs are stamped) will probably make this format more accessible to small publishers.[12]

Using Digital Information

Using digital information has both disadvantages and advantages compared to books or magazines. Information displayed on a computer screen is often not as comfortable or convenient to read as the printed counterpart. Certainly the computers most people use today cannot be conveniently used at a bus stop or on the beach, and research shows that people read 20 to 30 percent more slowly from screen than they do from a printed page.[13] Perhaps more important, the traditional search and retrieval aids we use with books (page numbers, indexes, tables of contents, visual memories of how something looked on the page) are not used in the same way with digital information.

But information in digital form has powerful advantages over printed documents. For example, retrieval software can search through and sort information to help a user find the specific information he/she wants, rather than reading through a whole book or using a (usually inadequate) printed table of contents. With a few keystrokes, a user can use an electronic index, receive a report on how many times a requested term appears in the text, and then actually look at each instance in context. This feature alone may not always be sufficiently helpful if it turns out that there are many instances (hundreds or even thousands) in which the desired term

[10] Ibid.

[11] For a discussion of different optical storage formats, see U.S. Congress, Office of Technology Assessment *Copyright and Home Copying: Technology Challenges the Law, OTA-CIT-422* (Washington, DC: U.S. Government Printing Office, October 1989), pp. 45-48.

[12] Tom McCusker, "CD-ROM Production Power!'' *Datamation*, vol. 37, No. 4, Feb. 15, 1991, pp. 26-29.

[13] For a discussion of the problems of typographic design for computer screens, see Richard Rubenstein, *Digital Typography: An Introduction to Type and Composition for Computer System Design (Reading, MA:* Addison-Wesley Publishing Co., 1988), pp. 189-193. For research on human factors of reading from screens, see also John D. Gould, ''Reading Is Slower From CRT Displays Than Paper: Attempts To Isolate a Single-Variable Explanation, ' *Human Factors*, vol. 29, No. 3, 1987, pp. 269-299.

Box 5-C—Digital Information and the Scholarly Publishing System

This view of the effects of electronic publishing on the current system of scholarly publication was taken from an OTA contractor report:

In the scholarly world today, the printed version of knowledge has the function of creating an archive of knowledge more than it serves the function of the exchange of knowledge. This is more true in scientific and technical fields that have access to networks and computers than to the humanities that do not. Scholarly exchange on the network occurs through ''affiity group computing, " such as the 2,000 Listserv protocols on Bitnet. This kind of exchange is a very good example of gift giving, since it creates a scholarly community (remembering that the function of a gift is to create a social bond, not a profit).

The exchange of knowledge as a gift exchange system among scholars creates "the invisible college" of researchers. The strength of this culture is that it is governed by a search for truth; the weakness is that access to it is restricted. The invisible college traditionally occurred through ''old boy networks' meeting in face-to-face interactions, such as conventions. But today, the use of digital networks has expanded it greatly.

(It must be stated that this "gift culture" is possible because the rewards of scholarly research are not given by a market, but nonetheless they exist in the economic rewards of promotion and tenure. However, this is deliberate. The system of scholarly communication was setup through a deliberate system of subsidies, such as the creation of university presses and the higher rates that libraries paid to subscribe to journals. This system of scholarly communication has been destroyed as book and journal publishing moved out of universities and became profitmaking enterprises in the marketplace. Today no research library can afford to pay for the full range of scholarly journals; the price of scholarly journals is rising twice as fast as any other research cost. The destruction of the print-based system of scholarly communication is an excellent case study of what happens when the fine balance between a gift exchange system and a commercial market exchange system is destroyed. However, this very destruction may be driving the development of digital-based scholarly communication. See below.)

The invisible college of research activities today exists on the network. Since nearly every scientific and technical field is growing and changing much faster than the print publication process can reflect, the real exchange of knowledge occurs long before the publication process. Most scientists must actively seek "preprints" in order to find out the current state of research in their field; the actual publication in printed form only validates the contribution for historical reasons and creates an archive. The most interesting experiments in digital publication reflect this; for example, the Online Mendelian Inheritance in Man (OMIM) and Genome Data Bank (GDB) projects at Johns Hopkins University, in which current research findings are peer-reviewed online in databases, are available electronically throughout the world. In essence these projects have done away with the print publication process altogether. Similarly, there are now about a dozen online peer-reviewed scholarly journals which essentially do away with the print publication process.

This evolution has several causes. One is the rapid increase in the cost of scholarly print publications, remembering that scholarly communication was conceived of as a gift exchange process that has been distorted by the marketplace. But another is that digital media have some of the qualities of an oral culture, and oral cultures have traditionally been more effective in providing natural homes for gift activities.

The example of scholarly journals shows that if the marketplace is allowed to define knowledge solely as a commodity, the system of dissemination of ideas and subsequent intellectual innovation can break down. This is happening now, as libraries cancel journal subscriptions, and are not allowed to share subscriptions because of the limitations of print copyright. Exchange systems governing intellectual products on the network must be devised that encourage use by "the invisible college. "

SOURCE: Steven W. Gilbert and Frank W. Connolly, ''A Wealth of Notions: Regaining Balance as New Information Technologies Collide With Traditional Controls and Incentives for Intellectual Work" contractor report prepared for Office of Technology Assessment, July 31, 1991.

appears. Users of databases usually find that a major problem with digital retrieval is 'getting too much information. Most on-line database retrieval systems allow the use of several strategies to narrow the search to retrieve a reasonable amount of information. (See box 5-B.)

Retrieval and text analysis software are becoming increasingly sophisticated. Still, getting the precise information one wants without reading through lots of irrelevancies, or worse yet, missing something important, can sometimes be a daunting challenge; the challenge can be even greater if one doesn't

know which of several databases offered by different publishers has the information. In today's market, different publishers have different user interfaces and search protocols. Learning to use them all can be time consuming and expensive. A class of software tools called ' 'agents' or '' filters are coming on the market. These can collect information from multiple sources, including electronic mail, on-line news services, and internal corporate databases, and sort it according to the user's priorities and interests.[14] Some database providers are offering more sophisticatcd software tools that w will allow users specify a search and then have it automatically performed on a number of different databases offered by that provider. In some cases these new tools controvcrsial because~ owners of the databases disagree about how royalties should be calculated when such software is used. [15] As use of digital information grows, people are going to need even more sophisticatcd search tools. One group envisions ''knowbot programs' that will act as personal librarians in the future. These artificial intelligence tools would accept the users requests for information, search many different sources, and then ret urn the results in a form (hat would be most useful to the reader. [16]

Hypertext

One great advantage of digital information is that it does not have to appear to the user in the same order in which it was written by the author, nor indeed does it need to appear to different users in the same way if they have different needs. ' [7] The ability of computer software to link different pieces of information also allows information to be presented in innovative formats, ' 'Hypertext and "hypermedia' are generic terms for systems that link related pieces of information for presentation in a nonsequential manner. Hyperlinks give the reader the power of ''subjective linearity."[18] When the material is read, the reader chooses the particular items to be presented, and the order of presentation, depending on his or her needs or level of interest.

Dictionaries and encyclopedias, with their many discrete entries and heavy dependence cm crossreferencing are obvious candidates for hypertext, and some have been published in this format. With a hypertext encyclopedia, the user may begin reading an entry on ''elephants and upon seeing the mention of ''ivory may ' 'click on' (select) that topic. The hypertext software will then usually open a window with a brief discussion of the term; by clicking again, the user may then choose to get more details on the new topic or go back to the original topic. In this way, the user can search through many related topics without toting half a dozen volumes from shelf to table, and without trying to mark a place in several books at once. Electronic bookmarks and other aids help the reader navigate through the information.

Hypertext is useful for on-line help systems for computer users. Hypertext provides the ability for a user to go directly from an error message to the relevant section of the on-line user's manual or even to a tutorial program giving computer-aided instruction on how to avoid the problem in the future. CASE (Computer-Aided Software Engineering) systems have been created in hypertext, allowing soft ware developers to link and navigate through the various versions of the reports, documents, and code objects developed during a major software development project.[19] In the commercial world, prototype hypertext systems have been developed for financial auditing, a field that is heavily dependent on crossindexing and relating information from different sources. Hypertext also lends itself to advertising, product catalogs, and tourist guides.

Hypertext is also used for educational and scholarly work. For example, Harvard's Perseus project brings together information from ancient Greek

---

14 David S. Marshak, ''Fi]tcrs: Separating the Wheat From the Chaff, ' *Patricia Seybold's Office Computing Report*, vol. 13, No. 11, November 1990, pp. 1-15.

15 For example, see Mick O'Leary, ''Dialog and the American Chemical Society Play a High Stakes Game, *Online*, January 1991, pp. 15-20.

16 Robert E, Kahn and Vinton G, Cerf, Th *Digital Library Project, Volume 1:* The *World of Knowbots* (Mc Lean, VA: Corporation for National Research Initiatives, 1988), p. 60. ''Knowbot" is a registered trademark of CNRI.

17 A few novelists, e.g., Juno Cortazar in *Hopscotch* (New York, NY: Macmillan, 1972), have experimented with giving readers alternative sequences in which to read chapters in printed books. Nonfiction writers often give prefatory *warnings,* such as those cautioning lay readers to review explanatory appendices before starting difficult text. In general, however, it is difficult to overcome the linear nature of printed information.

18 Rick Gessner, "Building a Hypertext System: Hypertext for Every Programmer's Toolbox, " *Dr Dobb's Journal, vol. 15, No. 6,* June 1990, p. 22.

19 Ibid., pp. 44-45.

history, archaeology, and literature, including the original Greek text, translations, dictionaries, and scholarly articles. Hypertext links allow students to open windows where the original Greek writing and its translation can be viewed simultaneously, or to check the validity of scholarly interpretation by going directly from the citation to the underlying text.[20]

Mixed-Media Databases

Nor is digital information limited to data or texts. Anything that can be seen or heard can be digitized, including voices, music, photographs, and moving pictures. Databases that combine items from several such sources are called mixed-media or multimedia databases. In many cases, multimedia packages are complex affairs that may link together computers, compact disc readers, and several different display screens. Mixed-media databases are still in their infancy, but the software to make them work and the display technology to make them appealing and easy to use are becoming available.

One example of a mixed-media database is the *American Memory Project*, a prototype digital library developed by the Library of Congress. *American Memory* is a collection of photographs, pamphlets, films, and sound recordings dating from the late 1800s and early 1900s. The information is stored on a set of video discs (with photos and films in analog form) and CD-ROMs (with all other material in digital form) and packaged together with hypertext search software.[21]

When mixed-media databases are combined with hypertext techniques the result is the ability to combine and link many different types of information on the same topic. The Perseus system mentioned above, for example, contains many photographs of ancient Greek art and architecture, in addition to text. In some cases, students can see more detail in the screen image of the photos than they would be able to if they had access to the original items. Further, the hypertext links allow related works to be compared. A hypertext treatment of Beethoven's Ninth Symphony includes a CD recording of the Vienna Philharmonic performing the symphony along with different versions of the score, discussions of the music and of Beethoven's life, examples of his writings, translations of the choral movement, and a quiz in the form of a video game. Students can listen to the whole symphony, study one part in depth, or listen note by note as they move the cursor through the score.[22]

Digitized video and audio require a lot of storage spacc compared with text. Optical media such as CD-ROM are good storage media for such types of information, but even here there are limits. The capacity of one CD-ROM easily accommodates a 26-volume encyclopedia and leaves room for the text retrieval software, 15,000 illustrations. 45 animation sequences, and one hour of audio.[23] A C can normally hold about 540 megabytes or 275,000 pages of text,[24] but only about 74 minutes of high-fidelity digital audio and far less of full-motion digitail video .[25] Several digital compression methods can be used to reduce the space required for digital audio or video by 2 to 10 times, though most of these methods are ''lossy,'' that is, the playback version is not of the same quality as the original because some information is lost.[26]

Digital Libraries

If digital technology is changing publishing it is also changing libraries. Libraries have been experimenting with and investing in computer technology for 25 years. Patrons at many libraries across the country have been either thrilled or dismayed to find that computer terminals have replaced the card catalog. High school students no longer thumb through musty index volumes at the public library in

[20] Ibid, p 68.

[21] GaryH. Anthes, "Llt)riU} Releases Data via Laser Disc, ' *Computerworld*, Sept. 10, 1990, p. 53. Also Carl Fleishaur, Director, American Memory Project, [~. S Library of Congress, personal communication, January 1991.

[22] Robert Haawind, "Hypertext" The Smart Tool for Information Overload,'' *Technology Review*, November-Dcccmber 1990, p. 47.

[23] Jakob Nielson, *Hypertext and Hypermedia* {Boston, MA: Academic Press, Inc , 1990), p. 53

[24] David C. Miller, *Publishers, Libraries & CD-ROM Implications of Digital Optical Printing* (Portland, OR: Library and Information Resources for the Northwest, 19X7), p.7.

[25] For a discussion of compact disc, CD-ROM, and other optical formats, see (-1. S. Congress, Office of Technology Assessment, *Copyright and Home Copying*, op. cit., footnote 11, pp. 44-48 and Nielsen, op. cit., footnote 23, pp. 123-126.

[26] Edward A. Fox, ''ACM Press Database and Electronic Products--New Services for the Information Age,'' *Communications of the ACM*, vol. 31, N{). 8, p. 948.

search of magazine articles; they search CD-ROM databases.

For most libraries, the first step was to use computers to streamline acquisitions, circulation, and other administrative operations. An early development was the Library of Congress's MARC (Machine Readable Cataloging) system, begun in 1966. The MARC system provides standards for encoding bibliographic information in machine readable form; many libraries use it to create their own machine readable records as a basis for on-line catalogs. By the early 1980s, many research libraries had some on-line cataloging capability; in same cases it was available for patron use. Cooperative agreements among libraries have led to major bibliographic databases such as OCLC (On-line Computer Library Center) and RLIN (Research Libraries Information Network) which receive contributions from, and process inquiries for, thousands of libraries.

On-line catalogs are a great aid to library users, allowing them to search large collections, or multiple collections, for the resources they need. For example, a user can now search the catalog of the entire University of California system at any time, from home or office. Ten years ago, the task would have required visits to over 1100 card catalogs on nine different campuses across the State. In addition, the cross-referencing possible with an on-line catalog is far superior to what was possible with cards. For example, a user can easily identify all holdings of the library published in a given foreign language during a given time period—a search which would be impossible with a card catalog.[27] Some university and research libraries have also begun providing users with on-line access to journal literature, either citations or full-text form, by obtaining licenses to some of the commercial databases or on-line services discussed above. (See box 5-D.) The next step for some research libraries in the forefront of automation is the automated delivery of library materials that have been identified through a search of the on-line catalog; materials might be delivered either in either in hard copy or in electronic form. An experimental project at Carnegie Mellon University will provide links from catalog citations to either full-text records or digitized images of printed pages available on-line .28 The same project is also attempting to make extensive digital resources such as databases and electronically published journals available at any terminal on campus. Part of this project is the enhancement of the university's electronic catalogs to give more useful search information. Examples of enhancements include listing of chapter titles, separate listing of authors of stories, essays, or chapters in books, and abstracts of technical papers.

What will be the digital library of the future? Though many of the building blocks are in place, many say that its potential has not been realized, primarily because so much of the world's knowledge is still not in digital form and will not be for some time to come.

Copyright Issues for Digital Information

The previous sections pointed up some major differences between digital information and information in more traditional forms. These differences have been summarized as a list of six characteristics of digital information[29]:

1. Works are easily copied.
2. They can be easily transmitted to other users or be accessed by multiple users
3. They can be easily manipulated and modified
4. Works are essentially equivalent: text, video, or music are all reduced to a series of bits and stored in the same medium.
5. Works are inaccessible to the user without hardware and software tools for retrieval, decoding, and navigation.
6. Software also allows for new kinds of search and linking activities that can produce works that can be experienced in new ways.

These characteristics of works in digital form have implications for copyright because they change how authors create, the kinds of works they can

[27] Clifford A. Lynch, "Library Automation and the National Research Network, ' *EDUCOM Review*, fall 1989, pp. 21-27.

[28] Denise A. Troll, *Library Information System II: Progress Report and Technical Plan,* Mercury Technical Report Series, No. 3 (Pittsburgh, PA: Carnegie Mellon University, 1990), pp. 5-16.

[29] Adapted f'rem: Pamela Samuelson, ''Digital Media and the Changing Face of Intellectual Property Law, '' *Rutgers Computer & Technology Journal*, vol. 16, *No. 2, 1990*, pp. 323-340 and from discussion at the OTA workshop on ''Digital Libraries, Electronic Publishing, and Intellectual Property, " Feb. 11, 1991, Washington, DC. A similar set of issues was also developed in *Copyright and Home Copying*, op. cit., footnote 1 I, especially ch. 2.

Box 5-D-One User's View

The availability of digital information offers many new opportunities for people of all kinds, but it also gives rise to some uncertainties and confusions, as discussed in this first-hand account taken from an OTA contractor report:

When I was in high school and college, I took handwritten notes of the library material I would use for research papers. Often the teacher would put material on reserve and I would trudge over to the library and laboriously copy stuff out by hand. By the end of college in the late 1960s, copy machines were installed in college libraries, but the copy quality was not particularly good and the price was pretty high, at least compared to handwritten note taking. A few years later, in graduate school, copy machines were common in college libraries, copy quality was pretty good, and the price of a copy was coming down. Colleges had even put in copy service centers in the library so I didn't have to stand at the copy machine; I could take my stuff to a central location and someone else would copy it for me. Still, as in high school and undergraduate days, I did my research manually using print indexes. But with the cheaper price of making copies and the better quality, my professors began to compile packets of material which lessened the number of trips to the Reserve Room.

When I began my doctorate, on-line search services were available but expensive for a graduate student without a research grant, so again, most of my research was done manually, using print indexes. In contrast to my undergraduate days, I did little manual note taking, using the copy machine instead. Most of my doctoral research was done on ERIC (Educational Resources Information Center—a bibliographic database), which I poured through volume by volume, year by year, About a month before I finished my research, my library installed ERIC on CD-ROM—too late to be of much use to me. Now, however, I'm not even sure where the paper copy of ERIC is in my university library. Should I so choose, I can download from CD-ROM ERIC and use the information in my home or office computer. In addition, I can access the university/consortium catalog of holdings from my home or office computer. No longer am I bound to the physical location of the library as I was as an undergraduate. Nor am I bound to paper. Should I want to compile a bibliography from ERIC, I can simply reformat the information I've downloaded to a format acceptable for whatever purpose I'm using it for.

To me, at this point, that 'stuff' that I've gotten from ERIC is similar to music coming over the radio or a television program. Either the music or the television program I can tape-it's coming into my home; I have the technology to tape it and so I do. And that's legal (I think). But what if I use that tape in my classroom? Is that legal? What if I simply reformat the information from the ERIC CD-ROM I've downloaded, and distribute that information to my students? Is that legal? Do I even stop to wonder whether it's legal? Suppose I'm working on a video disc presentation using Hypercard. I have these television programs I've taped, radio music I've taped, the bibliography I've downloaded, and the technology to put it all together into a video disk that will be used only by my students to help them learn better. Do I stop to wonder whether it's legal? Probably not, I have the tools (technical) to do the job and the information and material to put into the tools, so I probably just go ahead and make the video disc without too much concern about legality. I want to use the most effective tools and resources available as quickly as I can to help my students learn—and I could be frustrated waiting for every legal clarification and permission-if I could even find the sources.

SOURCE: Essay by Judy Ann Pearce in Steven W. Gilbert and Frank W. Connolly, "A Wealth of Notions: Regaining Balance as New Information Technologies Collide With Traditional Controls and Incentives for Intellectual Work", contractor report prepared for Office of Technology Assessment, July 31, 1991.

create, and the ways that readers (or users) read (or use) the works.

What Is a "Work"?

As mentioned in chapter 2, copyright protection attaches to an ''original work of authorship' when it is ' 'fixed in any tangible medium of expression, Thus, when an author writes a novel on a computer or word processor, it is clear that a printout is fixed and tangible and protected by copyright. It is also fairly clear that the words on the cathode-ray tube are evanescent and therefore unprotectable.[30]

A new kind of work that is increasingly being produced today is the electronic mail message,

30 Stanley M. Besen and Leo J. Raskind, '' An Introduction to the Law and Economics of Intellectual Property, ' *The Journal of Economic Perspectives*, winter 1991, vol. 5, No. 1, Case law has held that the fixation requirement for computer programs is met when the source code is written on paper or when the object code or microcode is fixed in a computer chip.

which usually exists only in digital form (fixed in the magnetic disc of the computers where it is sent or received) unless it is printed out. Users of electronic mail on nationwide systems like Internet, Bitnet or CompuServe send millions of messages a year. In addition, many agencies, corporations, and universities have internal electronic mail systems. Some types of electronic mail communication are intended by their senders to be private, others are public. However, there are currently no well-established rules of etiquette for electronic mail nor is there a clear distinction between public and private communications. Most messages are of an ephemeral nature and their writers may or may not care whether their rights under copyright are protected. Other users of electronic mail use this medium to contact and collaborate with colleagues, to express ideas, to exchange drafts of work in progress. In these cases, people would be interested in retaining the rights to their writings.

Technology allows a person to forward an electronic message received from someone else very easily to any number of other people, Is this kind of distribution the same as ''publishing' (a right which the copyright law gives exclusively to the author)? A message can also be modified before forwarding; does this create a derivative work (for which permission from the author should be gained)? Most people would probably agree that mail messages belong to the writer and that publishing them without attribution or modifying them without permission is a breach of manners, at best. However, whether it is an infringement of copyright has not yet been tested.

A further complication in the definition of a work arises because computers make collaboration and multiple authorship so easy. Many electronic mail messages are generated as a part of computer ''conferences. Conferencing is a method whereby people can communicate about topics of mutual interest, even though they are geographically separated. Conferencing software on the host computer records and organizes incoming messages so that each participant can read what has been written by others and then add his or her own responses. Conferences can be of short duration (a day or two) or they can go on for years; they can be limited to a few authorized members or open to anyone with access to the host computer.

Are the "proceedings" of a computer conference one joint or collective work, or many separate works? If it is a collective work with may contributors, the individual contributors can claim authorship in their respective contributions, but who can claim authorship in the collection as a whole? If it is not a joint work, does each individual message constitute a separate work, or do all the contributions of one author constitute a work? The question of what constitutes the work, and the identity of the author or authors, will determine the right of various contributors. For example, if the conference is a joint work, each contributor would have the right to publish the whole conference (subject to accounting to the other joint authors for their pro-rata shares of any royalty). Each joint author would have the right to sue for infringement of any portion of the conference. On the other hand, if the conference is composed of separate works of authorship, each individual author could exercise exclusive rights only over his or her own portion.[31]

In addition, the question of the size of a work might be important in determining if infringement has taken place and if a "fair use" defense against infringement is appropriate. Fair use is determined by four criteria (discussed in chapter 2), one of which is the amount and substantiality of material used with respect to the whole. If a computer conference is one work, then using a single message *in toto* is a small part of the whole; if each message is a work in itself, then copying a single message would be appropriation of the entire work and the fair use defense would be on shakier ground.

Mixed-Media Works

The fact that digital storage makes all works essentially equivalent complicates the definition of a digital work. Current copyright law treats works according to the category to which the work belongs. Categories defined by the law include: literary works; dramatic works; pictorial, graphic, and sculptural works; audiovisual works; motion pictures; musical compositions; computer programs; and sound recordings. These different categories sometimes have different implications for protection of the work. There is no category for a mixed-media work that combines examples from each of these categories.

[31] Morton David Goldberg, partner, Schwab Goldberg Price & Dannay, personal communication, Oct. 3, 1991.

One school of thought holds that a mixed-media work should be considered to be a series of different works, with each type of work being treated according to its class. Another approach would be to consider the whole package as if all the parts were of the same category.[32]

Converting Works to Digital Form

Developers of mixed-media products encounter copyright questions not only in protecting their works, but in trying to create them. Getting permission to put copyrighted works into digital databases has sometimes been so difficult as to prevent projects from getting underway. Because the medium is new, most people have never dealt with it before and the channels for copyright clearance and agreed upon rates for royalties have not yet been worked out. And because many mixed-media projects are large collective works, many different rights owners often need to be satisfied.

In the field of music, songwriters and music publishers collect royalties through ASCAP (American Society of Composers, Authors and Publishers) and BMI (Broadcast Music International), who grant licenses for public performances of music, and through the Harry Fox Agency, which grants mechanical licenses for incorporating music into recordings or movies .[33] The fee structures of these organizations are geared to traditional uses of music, and permission to use music typically costs a few percent of the expected sale price of a published disc. The price structure is reasonable for traditional recordings that use a small number of complete songs on each disc, but prohibitive, for example, for a multimedia library intending to use small parts of hundreds of songs. Until new structures are developed, mixed-media producers are generally limited to works in the public domain, or works composed especially for use in multimedia presentations. Several companies are developing libraries of such works.[34]

Another layer of complication arises if multiple licenses are needed for each work, for example, if multimedia presentations are deemed to be public performances of the copyrighted works they include. It is fairly common in the music industry for performance rights to reside with one entity, while reproduction and derivative use rights rest with another. If one user sits before a computer terminal and hears part of a copyrighted song in the presentation, has a public performance occurred? One royalties-collection agency has taken the position that each use of computer-based presentation is a public performance.[35] The definition of public performance is not clear in the case of computer-based works.

Similar, or worse, difficulties exist in other areas, particularly in areas like images and writings where there are no collective organizations like ASCAP, and negotiations must be made with many individuals. A project to copy baseball cards on CD-ROM was scrapped when it was realized that the publishing company did not have, and could not grant to a licensee, the rights to make digital versions of its own printed cards. The rights would have to be obtained through separate negotiations with more than 500 individual players or their lawyers, but the royalty that could reasonably be expected from sales of the CD-ROM product would be far too small to justify such an undertaking.[36]

Getting permission to convert whole works, such as books, into digital form is generally easier. A number of books have been converted to digital form (some packaged with text analysis software to facilitate scholarly research). Many of these are in the public domain, but for those that are copyrighted, a typical contract follows the model of a contract granting translation rights.[37]

Originality and Authorship

Copyright attaches to ‘ ‘original works of authorship. . . ."38 Original, in this case, means that the work was independently created by the author and

[32] American Association of Law Libraries, ‘ ‘Copyright Considerations for the Use of Mixed Media in Libraries' discussion draft, appeared as an appendix to *A-V & Micrographics SIS Newsletter*, vol. 10, No. 2 , May 1990, and *Automation,* vol. 9, No. 2, winter 1990, pp. 12-23.

[33] For more discussion of royalty structures for music, see *Copyright and Home Copying*, op. cit., footnote 11, especially ch. 5.

[34] Jack Shandle, ‘ ‘Multimedia Computing Hits a Sour Note, ’ *Electronics,* June 1991, pp. 48-53.

[35] Ibid.

[36] Ibid., p. 50.

[37] Michael Newman, Director, Center for Text and Technology, Georgetown University, personal communication, June 21, 1991.

[38] 17 U.S.C.102(a).

not copied from another work. Original does not mean novel—two writers could conceivably create identical works, but as long as neither copied from the other, the works would be original. In earlier cases, the U.S. Supreme Court has also held that some degree of creativity must be involved, and that protectable writings are the "fruits of intellectual labor.' '[39] The U.S. Supreme Court has also defined ''author' as ''he to whom anything owes its origin; originator; maker. ' '[40]

A lot of digital information is in the form of compilations of facts. Facts themselves are not copyrightable. However, an author's selection, arrangement, and organization of facts may be sufficiently original to make the compilation copyrightable. Many publishers, for example, compile and resell information available from the Federal Government (e.g., court decisions, laws and regulations, economic and financial statistics). The database publishers add value to the government material by organizing it, adding indexes, packaging it with search and retrieval software, etc. Government information is in the public domain, not covered by copyright. Yet the publisher's selection and arrangement of the public domain information can be copyrighted .41

How much of the publisher's contribution should be protected is sometimes subject to controversy. In a lawsuit involving two major legal publishers, West Publishing Co., claimed that a competitor, Mead Data, infringed by offering Mead's subscribers an electronic information service with citations including the page numbers on which legal opinions appear in West's publications. The district and circuit courts found that the organization of the information, including page and section numbers, was copyrightable expression and that a competitor's unauthorized use of them was an infringement.[42] The case ultimatey ended in settlement and the decisions remain controversial.[43]

A recent case decided by the U.S. Supreme Court, *Feist Publications v. Rural Telephone Service Co.*, found that telephone White Pages are not copyrightable, and that copying them into another compilation (a regional telephone book) was not an infringement. Lower courts had ruled, in accordance with earlier ''sweat of the brow' or ''industrious collection' tests, that Rural Telephone was entitled to copyright because of the effort it expended to created the White Pages directory. However, the U.S. Supreme Court held that the proper test for copyrightability of a compilation is originality-in this case, the intellectual work of selection and arrangement of facts. Rural Telephone did not select the facts (it was required to list all subscribers with published numbers) and the arrangement was the same as is traditionally used in White Pages telephone directories. Thus, this compilation did not have the minimal spark of creativity to warrant being called an original, copyrightable work.[44]

Database publishers also consider their user interfaces and search and retrieval aids to be copyrightable expression, but it is not always clear how much of this is actually protectable.[45] The same controversies apply here as with the discussions over protectability of user interfaces as discussed in chapter 5.

''Authoring' is a technical term used in the process of constructing works in hypertext. In this case the ''author' is the one who turns a lot of different objects (words, paragraphs, pictures, sounds) into hypertext by establishing the links among them. This ''author" may or may not be the

39 *The Trade-Mark Cases,* 100 *U. S., at* 94.

40 *Burrow-Giles,* 111 U. S., at 58.

41 For a thorough discussion of electronic publishing and government information see U.S. Congress, Office of Technology Assessment *Informing the Nation: Federal Information Dissemination in an Electronic Age,* OTA-CIT-396 (Washington, DC: U.S. Government Printing Office, October 1988).

42 *West Publishing Co.* v. *Mead Data Central, Inc.*, 616 F. Supp. 1571 (D. Mire. 1985) (grant of preliminary injunction "copyright "sSue")* *aff d,* 799F.2d 1219 (8th Cir. 1986), *cert. denied 479* U.S. 1070 (1987). Trial was held on April 5-15, 1988 in the U.S. District Court for the District of Minnesota. Prior to a decision on the merits, parties resolved their dispute and entered into a confidential settlement with approval of the District Court. Order No, 4-85-931, (D.Minn. July 21, 1988).

43 For example, see L. Ray Patterson and Craig Joyce, ''Monopolizing the Law: The Scope of Copyright Protection for Law Reports and Statutory Compilations, " *UCLA Law Review*, vol 36. April 1989, pp. 719-814. The authors note that the defendant did not copy the numbering system, but merely cited it; in addition they believe the courts gave too much weight to economic effects on the plaintiff rather than to the purposes of copyright.

44 *Feist Publications, Inc. v. Rural Telephone Service Company, Inc ,* *No.* 89-1909,59 U. S.L.W. 3243 (U.S. Oct. 1, 1990).

45 Pamela Samuelson, "Some New Kinds of Authorship Made Possible by Computers and Some Intellectual property Questions They Raise, " presented at the Intellectual Property and Authorship Conference, Case Western Reserve University, Cleveland, OH, Apr. 19-21, 1991

same as the one who actually wrote the words that appear on the screen. Yet the establishment of the hyperlinks can be a significant intellectual effort; one that greatly contributes to the usability of the final product.

What are the implications of "authoring" if one person establishes new hypertext linkages within a system to which copyright is already held by someone else? If a scholar, working with ordinary print materials, were to make a new discovery, he or she would report it by writing an article; the article would undoubtedly be a copyrightable work of authorship, even though it contained many quotations from other works. An annotated bibliography, in which a scholar cites many references and adds his or her own comments, is also considered a copyrightable work.

However, an electronic library offers scholars new ways of publishing articles and guiding readers through relevant literature. A scholar could develop a set of hyperlinks that directly leads readers through the referenced materials in just the order the author wishes to make a point or demonstrate a discovery. Such a work might represent considerable intellectual effort, and might be considered a work of scholarship (the scholarly communities will have to work out their own standards about publishing in an electronic environment), But is it a writing? Could a set of hyperlinks be considered a copyrightable work? Or is it an idea or discovery, and therefore unprotectable? Could an electronic article consisting of a set of hyperlinks be considered a "derivative work' based on the underlying works, in which case permission should be obtained before it is created?[46]

Use of Digital Information

Book authors ultimately seek to collect financial rewards for their work by selling copies of their work to readers (often through publishers). A reader who has purchased a copy of a book is free to do whatever he or she wants with it—read it aloud to a child, make notes on it, give it to a friend, or return it undamaged to the store for a refund. The book is property that the reader owns, and under the "frost sale doctrine, ' the owner is free to sell it to someone else.

Electronic publishing is also about delivering works to readers and returning royalties to copyright holders. However, several characteristics of digital information make the delivery system different and also lead copyright owners and their publishers to want more control over the readers' uses of the information.



Photo credit: Mark G. Young

A CD-ROM database containing images of magazine and journal articles. Users can read from the screen or printout an authorized copy.

When Is Information Used?

In using an on-line information service, a reader does not purchase any piece of property; rather he or she buys *access* to the electronic information. Once that access is permitted, the information is out of the control of the copyright owner and the publisher. The user might decide the information is useless and do nothing further with it; on the other hand, he or she may download it (store it in the user's own computer) for future use. For the most part, publishers have no way of knowing the final disposition of the material. For this reason, publishers consider information as "used' as soon as it reaches the reader. They wish to be paid in advance. In the case of on-line vendors today, most fees from users are paid as periodic subscription fees plus use charges related to the amount of time spent searching each database, and sometimes charges for specific documents retrieved. The various schemes for digital libraries usually postulate charging for use of

46 *Ibid*

documents based on how much information a user has retrieved.

From the user's point of view this means that some amount of useless material is paid for. A partial remedy for this is to improve search and retrieval software and to offer means to browse through information (with tables of contents, abstracts, free or low-cost views of a portion of the document, etc.) before a reader coremits to requesting a whole document.

To get access, users generally have to agree to certain limitations on their use of the information. People sometimes purchase a copy of a work on CD-ROM or floppy disc;, but in many instances, close reading of the wrapper may show that it is leased or licensed, not purchased. In these cases, the first sale doctrine does not apply; the use of the material is subject to the terms of the license agreement. The license may state, for example, that users may not resell the disc, or alter it, or place it on network where more than one person can use it. Users may have to return old discs when new ones are supplied, or when the subscription period ends.

Digital information often comes to the end user through a long chain of intermediaries-the publisher, the database service, a library. Contracts govern the rights and responsibilities at each link of this distribution chain.

If there is a long chain of suppliers, there can additionally be many "layers" of users. It is sometimes hard to actually identify the end user of information in a real sense. A student researcher downloads an article from a CD-ROM database to a floppy disc and gives the disc to a teacher who posts it to an electronic bulletin board. Someone sees it on the bulletin board, makes a printout, and faxes a copy to a colleague who hangs it on a physical bulletin board. Each layer of use here has adapted the article to a new medium and involved a new end user. Users may or may not be aware of how the article got to them or what happened to it after it passed from their hands. Issues like copyright infringement and breach of contract may be involved, but who is at fault—the maker of the first copy or of the second or the person who received the last one? Often infringement is so widespread and diffuse that it is difficult to determine damage.[47]

Traditionally, copyright law does not give copyright owners rights to control the access that readers have to information.[48] Copyright owners in the electronic world use contracts to impose restrictions to make sure that they are paid for every instance of access or use. Still, as a practical matter, these restrictions do not prevent unauthorized copying. Once a user has paid for one legitimate copy of something, there is often not much except moral suasion to prevent his or her making other copies. Digital information is easily copied and easily transmitted to many locations. These characteristics make electronic distribution an attractive publishing medium; but they have a flip side; almost any reader is a potential ''publisher' of unauthorized copies.

Unauthorized Copying

Unauthorized copying and distribution is not a problem unique to digital information. Over the past 20 years, the photocopy machine has made copying of books, articles, and other printed works very easy. The introduction of the fax machine has even made it easier to deliver photocopies over long distances. Still, there are limitations to the distribution of unauthorized copies on paper: copy quality degrades with each generation; fax machines, at least at the present time, take some effort to program for large distribution lists; a copied document is still in the same format as the original and can be easily identified as a copyrighted work; photocopying large amounts of material can be inconvenient and time-consming.

Digital copies, on the other hand, do not degrade; each copy is of the same quality as the original. Distribution is easy; the copy could be posted on a computer bulletin board, or distributed to a list of users on a computer network. If one wants to disguise the origins or authorship of the document, format changes can be made with a few keystrokes, Scanning technology now allows one to turn information on paper into digital information so that it can be changed or manipulated.

Some proposals have been put forward to use technology to control unauthorized copying in the

47 Thanks to Rosemary Talab for articulating the concept of layers of use. Rosemary Talab, Kansas State University, personal communication, Nov. 28, 1991

48 Samuelson, "Digital Media and the Changing Face of Intellectual Property Law,' op. cit., footnote 29, pp. 323-340.

context of a digital library. One option is to assign intelligent software agents, for example the knowbot programs mentioned above, to the job of representing the copyright holder's interests. A special type of knowbot program is called a courier. A courier is assigned to a specific item of information (a database, a document, or a paragraph). Depending on the wishes of the owner, the courier can record all uses of the information so that charges can be applied or it can immediately request permission before releasing the information to the user (and deny access if permission is not granted.) The system could also allow for users to make derivative works or to extract parts from a protected work while still giving full credit (and paying royalties) to the original owner. When a user includes a piece of protected information in another document, the courier will create another version of itself to accompany the extract and to represent the owner's potential interest in the new work.[49] Some proposed systems hope to encourage users to do all their reading, writing, and adapting electronically and to discourage unauthorized copying with a pricing structure that makes working within the system and using authorized copies less costly than making unauthorized copies.[50]

In any case, technological proposals for limiting unauthorized copying generally seem to work only within a closed system. Once a user moves an authorized copy out of the system there seems to be no way to prevent further copying. Some writers suggest that there is no solution to the problem of unauthorized copying and that the problem is sufficiently grave that electronic publishing will never thrive as an industry because authors and publishers will not release works in digital form.[51] However it is possible that, as in the case of the photocopying of books or home taping of musical recordings, a viable market will persist despite the presence of unauthorized copies.[52]

Special Concerns of Libraries

Libraries, as mentioned earlier, have been actively making use of computers and digital information for two decades. Digital information allows libraries new ways to offer services, and completely new services to offer, but some uncertainties still need to be worked out.

Many of the rules under the copyright law regarding lending and sharing library materials or making preservation copies or replacement copies of damaged works were developed with printed books and journals in mind. For example, for purposes of preservation or security (or to deposit with another Library) a library has the right to make a "copy in facsimile form" of an unpublished work.[53] Neither the law nor the legislative history define "facsimile, ' but the dictionary definition is ' 'an exact copy, ' ' which may indicate that conversion of a printed work to machine readable digital text is not permitted.

Some provisions in the copyright law also deal with copying and other use of ''computer programs," but do not specifically extend to digital information. For example, the copyright law gives the owner of a computer program the right to make an archival (backup) copy under certain conditions. There are two points here. In the first place, the library may not be the owner of the computer program. Vendors often say that programs are licensed, not sold. The library, as a licensee rather than an owner, does not have the rights described in the copyright law; these are abrogated by the terms of the license. There is considerable controversy over the enforceability of many of these contracts where the vendor has enough barg aining power to force terms on the user.[54] At present, there is a wide variety in the terms and conditions of software and database licenses. An institutional user like a library or university computer center often uses hundreds of different program and data packages, and to ensure

49 Cerf and Kahn, op. cit., footnote 16.

50 Daniel Gross, Magnetic Press, Inc., discussing the pricing structure of the proposed Xanadu system under development by Autodesk and its partners at the OTA workshop on "Digital Libraries, Electronic Publishing, and Intellectual Property, " Feb. 11, 1991.

51 See, e.g., Robert Weber, "The Clouded Future of Electronic Publishing, " *Publishers Weekly*, vol. *237, No. 26,* June 29, 1990, pp. 76-80.

52 *Copyright and Home Copying,* op. cit., footnote 11, especially ch. 7.

53 See 17 U.S.C. 108(b).

54 American Association of Law Libraries, op. cit., footnote 32.

compliance with all of their different requirements is difficult.[55] (For more on licenses, see chapter 2.)

The second point is that the copyright law currently refers only to computer programs and not to data or digital information. Computer data is stored in the same medium as computer programs, and it would seem logical to treat them in the same way, but the argument remains that digital data does not fit the definitions currently set out in section 101 of the Copyright Act so owners have no right to make archival copies.[56] The two points raised here become even more complicated for libraries in the case of mixed-media works where printed material, digital data, computer programs, microfiche, and other forms might be packaged and used together.

Libraries have a long tradition of resource sharing. Several libraries may cooperatively purchase material, and some libraries may refrain from making certain purchases in the knowledge that the material can be obtained through interlibrary loan when needed, Resource sharing practices have long been viewed as prudent use of both funds and storage space, especially for items for which demand is low. Lending of materials among libraries is institutionalized both by tradition and under the provisions of the Copyright Act (section 108), and interlibrary loan usage has increased dramatically in recent years. However, resource sharing practices have recently come under fire from some publishers who see them as depriving information providers of sales.[57] Publishers strengthen their position by leasing, rather than selling materials, thus denying libraries the rights which ownership permits under the ''fust sale doctrine.' Contracts with electronic information providers sometimes limit or forbid sharing or lending of materials. Libraries, particularly public ones, have an obligation to balance the interests of users and producers—a balance which the Copyright Act is intended to maintain.

It has been suggested that the growing use of electronic information, and the tendency of information providers to control the uses of this material through contracts, will lead to greater distinctions between for-profit and not-for-profit libraries, in terms of their library operations, cost differentials, and access .58 Not-for-profit libraries may find themselves placing heavier reliance on free or lower-cost databases, and there may be less ability to share materials between libraries. Profit-based libraries will have access to more expensive information resources, but will also have great controls on their abilities to share resources or to network with other libraries.

Many libraries are examining their own role in offering digital information services to patrons. The shift to digital information introduces new kinds of costs. Public libraries are struggling to determine fair allocation of resources between digital information and printed library materials; in addition there is the question of whether, or how much, to charge patrons using some expensive data services. Public libraries have traditionally been free (supported by taxes) so taxpaying users have already paid for services. In addition, public libraries have an obligation to provide information services to those who cannot get them otherwise. Some libraries are developing usage charges for access to some databases, or are trying to allocate use of scarce resources among users by imposing time limits on the use of workstations with access to certain databases. Over the years the balance in cross subsidy between traditional and electronic services may change several times.

Another question is remote access to library services. The technology exists to allow users at home, office, school to use essentially any computer-based service they could use within the library walls. That many libraries are not now offering such services reflects both the costs of starting up such a service as well as questions that must be resolved concerning license policies. One observer points out that remote access could be a ''boon to the user and a bane to the supplier. ''[59] In many cases, libraries would be passing on to users access to data that has been obtained through a contract between the library and an information vendor. Many contracts now forbid remote access or make it prohibitively expen-

55 Comments at OTA Advisory Panel meeting, Sept. 11, 1991.

56 American Association of Law Libraries, op. cit., footnote *32.*

57 Rosemary Talab, "General Trends in New Technology Usage: Stages of Copyright Development on a National Level, " in *Advances in Library Resource Sharing,* vol. 3 (Greenwich CT: Meckler, in press), pp. 82-83.

58 Ibid., p. 78.

59 Ibid., p. 79.

sive. Libraries and vendors will have to work out the pricing of such services.

Also to be worked out are policies about the use of material obtained by patrons. Some libraries already offer on-line information, and other services such as access to electronic bulletin boards,[60] to their patrons; they therefore become an additional link in a complex of transactions. To what extent are libraries responsible if users make unauthorized copies, post copyrighted material on electronic bulletin boards, send obscene messages, or otherwise infringe copyrights, violate contracts, or break laws? These problems are not new. The advent of the photocopier eventually caused libraries to evolve a policy of providing copiers, posting a notice about relevant aspects of the copyright law, and then leaving users unsupervised to follow their own consciences. Policies regarding digital information—what can be downloaded, number of printouts allowed, etc.—will also need to be worked out, but the policy setting process may be more complex since contracts with information vendors will also be involved.

Another area of uncertainty is in the creation of information based on library holdings. On-line catalogs can be made more useful by adding more information about the works being cataloged—tables of contents, lists of illustrations, etc., but there may come a point where enhancements to the catalog infringe the copyright on the underlying works. As libraries increasingly work to create new information these questions may arise. Some envision libraries, especially research libraries of major universities, as eventually becoming electronic publishing centers for scholarly work.[61] If this is to happen some of these questions will have to be settled.

[60] See Nancy Cline, ''Information Resources," *EDUCOM Review*, summer 1990, pp. 30-34.

[61] Ibid.

Chapter 6

Economics, Intellectual Property, and Software

Contents

Page

Introduction 183
Public Goods and the Underlying Rationale for Granting Rights 185
Intellectual Property Systems and Net Social Benefits 186
"Strength" of Protection and Social Benefits 187
Industry Differences in the Perceived Efficacy of Intellectual Property 190
Cumulative Technological Progress 192
Patents and Trade Secrets 194
Choices in the Optimal Design of Patents 194
Optimal Patent Term 195
Optimal Patent Term and Breadth 196
Compatibility, Network Externalities, and the Installed Base 197
The Economics of Copyright and Home Copying 199

Boxes

Box *Page*

6-A. Some Software Industry and Technology Changes Since CONTU 188
6-B. Literature on Home Copying 200

Chapter 6
Economics, Intellectual Property, and Software

Introduction

Unfortunately, economists do not yet have a final word--or words—for us concerning the optimal intellectual property system for computer software. The topic of intellectual property applied to software has attracted increasing attention over the last decade, but the literature on the economics of software is still evolving, as is the broader economic literature on intellectual property and innovation.[1] Economists' efforts to understand these issues are made all the more difficult by the rapid changes in software technologies and markets. The discussion in this chapter, therefore, is only a 'snapshot' at this time, rather than an exposition of economists' ''solution' to the problem of how best to balance private incentives and social benefits in a rapidly moving area of technology. The following sections offer a perspective on the development of this literature and of economists' understanding of innovation and technological change.

In the ' 'U.S. intellectual property system, ' copyright, patent, and trademark are administered under Federal jurisdiction, as is protection for the topography of semiconductor chips. Laws concerning trade secrets and the misappropriation of confidential business information, trademarks not federally registered, and certain limited kinds of 'unfair competition' are under State jurisdiction.[2] The economics literature on intellectual property concentrates on Federal grants of patent and copyright. *In large part, this focus stems from the nature of patent and copyright: these exclusive rights (for limited periods of time) have been designed within a framework involving an economic tradeoff between private incentives and social benefits.*[3] *The* laws governing trade secrets do not incorporate this kind of explicit tradeoff.[4] Therefore, patent and copyright offer more established economic bases for theoretical and empirical analyses of markets for intellectual property. The resulting focus in the literature is reflected in the following discussion,[5] which spotlights some of the issues on intellectual property and innovation that economists have explored, including:

- the economic rationale for granting intellectual property rights-linkages between these rights and economic benefits to society as a whole;
- the balance between the benefits and costs accruing to intellectual property holders and to the public at large;
- factors affecting this balance and the socially "optimal' degree of protection;
- interindustry differences in the efficacy of patent protection;
- implications of dynamic models with cumulative innovation;
- choices in "optimal" patent design (term, breadth); and
- network externalities and compatibility.

[1] *OTA note:* Unless otherwise specified, OTA uses ''innovation' and ''innovative activity' in this chapter to refer to research and development (R&D) and other creative processes producing scientific and technological advances, whether the form of these advances would legally be considered copyrightable, patentable, or neither. Precisely speaking, ' 'innovation' is the technological *implementation* of a new idea, method, or device discovered by the process of ' 'invention. " The economics literature on R&D, intellectual property, and technological progress has focused mainly on patents—computer software is one of the few copyrightable works where ''authorship' directly con.. titutes technological progress—and has tended to use the terms ''imovation' and ''invention' synonymously. In reviewing the economics literature on this topic, OTA uses the authors' terminology.

[2] See Stanley M. Besen and Leo J. Raskind, ' 'An Introduction to the Law and Economics of Intellectual Property, '' *Journal ofEconomic Perspectives*, vol. 5, No. 1, winter 1991, pp. 3-27. Besen and Raskind review the basic economics of intellectual property and look at each of the modes in some detail.

[3] Private incentives are expected t. arise from the right holder's limited monopoly powers; social benefits are expected to include additional benefits to society from the induced disclosure and/or dissemination of innovations and technological advances.
For discussions of this balancing between private incentives and social benefits in the ''intellectual property bargain,'' see 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). See also Paul Goldstein, *Copyright—Principles, Law and Practice* (Boston, MA: Little, Brown and Co., 1989), secs. 1.1 and 1.2.

[4] *See* Besen and Raskind, op. cit., footnote 2, p. 23.
The rightful possessor of a trade secret does not have an exclusive right to usc the secret information, and the law only provides for legal remedies when the secret is lost through breach of contract or ''improper' means of discovery (e.g., industrial espionage). A trade secret may be maintained indefinitely. See ch. 2 and the section on trade secrets below; see also David Friedman et al., ''Some Economics of Trade Secret Law, ' *Journal of Economic Perspectives*, vol. 5, No. 1, winter 1991, pp. 61-72.

[5] This OTA report does not deal with trademark issues except as they relate to counterfeiting (see section on piracy in ch. 3).

Kenneth Arrow's seminal 1962 paper examined resource allocation problems in markets for information and invention and the obstacles to efficient functioning that information markets face.[6] He concluded that in a free-enterprise economy, there will be underinvestment in invention and research and underutilization of the resulting information:

> To sum up, we expect a free enterprise economy to underinvest in invention and research (as compared with an ideal) because it is risky, because the product can be appropriated only to a limited extent, and because of increasing returns in use. This underinvestment will be greater for more basic research. Further, to the extent that a firm succeeds in engrossing the economic value of its inventive activity, there will be an underutilization of that information as compared with an ideal allocation.[7]

The bulk of economic analysis on linkages among technological progress, economic welfare, and intellectual property has dealt with the patent system, rather than copyright-software is remarkable in being a technology for which copyright is so crucial. However, there is also a well-developed literature dealing with economic welfare, copyright, and consumer copying of journal articles, music, software, etc. (see the final section on home copying below). Moreover, many of the arguments concerning patents and duplication of innovations can be applied to software copyright, especially to issues like copyright protection of interfaces and the appropriate breadth of copyright protection.

As noted previously, this is an evolving literature. Sometimes, the analyses discussed in the following sections will differ in conclusions or policy implications. In particular, some of the economic research done since the National Commission on New Technological Uses of Copyrighted Works (CONTU) suggests policy implications that differ from those of earlier work in the 1950s, 1960s and early 1970s.[8] These differences result because the economic models depict different industry conditions, differ-

"One of the hallmarks of a competitive industry is the ease with which entrepreneurs may enter into competition with firms already doing business. The absence of significant barriers to entering the program-writing market is striking. There are several hundred independent firms whose stock in trade is computer programs. New software firms may be formed with few people and little money; entry into the market has thus far been fairly easy. None of the evidence received by the Commission suggests that affording copyright to programs would in any way permit program authors to monopolize the market for their products. Nor is there any indication that any firm is even remotely close to dominating the programming industry."

Final Report of the National Commission on New Technological Uses of Copyrighted Works (CONTU), **July 31, 1978, p. 23.**

"It seems to take courage even to register doubts about the net benefits oft he patent system. Some of the faithful, ardent believers in the patent system in its present form as an inherently moral institution, as a necessary component of private property, as an integral part of a free-enterprise economy, and as an indispensable spur to economic progress, have been quick to bear down on unbelievers with invectives and innuendoes. Perhaps this sort of pressure has something to do with the fact that agnostics on t he econom ics of patents often preamble their apprehensions about the consequences of patent protection in our time with affirmations of faith in the achievements of the past."

Fritz Machlup, An *Economic Review of the Patent System,* **Study of the Subcommittee on Patents, Trademarks, and Copyrights of the Committee on the Judiciary, U.S. Senate, Study No. 15 Pursuant to S. Res. 236, 85th Congress, 2d Sess. (Washington, DC: U.S. Government Printing Office, 1958), p. 43.**

---

6 Kenneth J. Arrow, ''~onomic Welfare and the Allocation of Resources for Invention " in National Bureau of Economic Research, *The* Rate *and Direction of Inventive Activity: Economic and Social Factors* (Princeton, NJ: Princeton University Press, 1962), pp. 609-624.

Because information is intangible, even with legal protections, sellers cannot fully appropriate its value. On the demand side, potential buyers find it difficult to value information correctly, unless they have already acquired it. (Ibid., p. 615.)

7 Arrow, op. cit., footnote 6, p. 619.

In his analysis of incentives to invent in both monopolistic and competitive markets, Arrow found that although the incentive to invent was greater under competitive conditions, even then it was less than optimal, especially for major inventions (ibid., pp. 619 and 622). Arrow concluded that, while a preinvention monopoly might increase appropriability, the advantages of this additionat incentive would have to be "offset against the monopolist's disincentive created by his preinvention monopoly profits" (ibid., p. 622). Therefore, he concluded, government subsidies or other nonprofit financing for research and invention will be needed to compensate for the underallocation of resources to these activities (ibid., pp. 623-624).

8 As described by Sidney Winter, the pendulum of opinion on the "optimal' term of protection (e.g., whether increasing or decreasing the term of patent protection would be more socially desirable) has swung back and forth over the years. For his discussion of changes in economic thinking about the term and strength of protection see Sidney G. Winter, "Patents in Complex Contexts: Incentives and Effectiveness, " in Vivian Weil and John W. Snapper (eds.), *Owning Scientific and Technical Information (New* Brunswick NJ: Rutgers University Press, 1989), pp. 41-43.

For another discussion of the economics literature on innovation, see Robert P. Merges, "Commercial Success and Patent Standards: Economic Perspectives on Innovation," *California Law Review, vol. 76,* pp. 803-876, 1988.

ent types of innovation, and different timeframes. Much of the earlier economic work on intellectual property (mainly patents) focused on cost-saving *process* innovations, while later work looked at *product* innovations. Until after the mid- 1970s, most analyses of (socially) optimaI patent design focused on *patent term* and assumed static (one-shot) models of innovation. The more recent work focuses on *breadth of protection,* as well as term; dynamic models of innovation include the possibilities of multiple inventors,[9] cumulative innovation, and network externalities. The more recent work, using dynamic models for innovation, supports shorter, rather than longer, terms of protection.[10]

Public Goods and the Underlying Rationale for Granting Rights

In economic terms, a "public good" is one that has the property of *nonexclusivity:* once the good has been produced, it is impossible (or prohibitively costly) to exclude any individual from benefiting from it, whether or not he or she pays.[11] Indeed, individuals have an incentive not to pay for the good, or to undervalue it, in hopes of getting access as ''free riders.' The inability to exclude free riders distorts market signals and is thought to result in inefficient allocation of resources to nonexclusive goods and underproduction of them, relative to socially optimal quantities.

Public support is provided for some activities in order to overcome this imperfection in the market. Thus, some public goods—like national defense—are produced publicly by government and paid for by collective taxation. Some goods that are produced *privately also* face the problem of (at least partial) nonexclusivity. For these also, ordinary market forces may not produce the most desirable social outcomes.[12] Goods that have the property of nonexclusivity will tend to be produced in insufficient quantity or variety because producers are unable to fully benefit from investments in creating them.[13]

Establishment of a system of defined "intellectual property rights" can help alleviate this difficulty.[14] The Framers of the U.S. Constitution dealt with this market imperfection by giving Congress the power to grant copyrights and patents:

> The Congress shall have Power. . .To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries. (U.S. Constitution, art. I, sec. 8, cl. 8.)

In granting a limited monopoly through copyright or patent, government attempts to compensate for distortions arising from nonexclusivity. According to this rationale, without the counterbalancing grants of monopoly power bestowed through copyright and patent, the inability of authors and inventors to appropriate economic returns from their labors would result in the underproduction of new works and inventions.

Using government grants of exclusive rights to produce countervailing distortions is not the only means to improve the efficiency of intellectual property markets; indeed, using a balanced set of public and private measures, rather than relying on expanded or strengthened intellectual property rights alone, may be desirable: [15]

---

[9] In multiple-inventor ''patent races,' the rate of R&D spending affects the probability of invention.

[10] See Winter, op. cit., footnote 8, and below.

[11] For a discussion of public goods, see John Eatwell et al. (eds.), *The* New *Palgrave: A Dictionary of Economics* (The Stockton Press, NY: 1987), pp. *1061-1066.*

[12] This happens both because *nonpaying* individuals can gain access as free riders and because the information at the heart of the intellectual property can be ''consumed' without supply being exhausted. In contrast to markets for houses or antiques, consumers don't have to bid to exclude each other in order to use information.

[13] The National Commission on New Uses of Copyrighted Works (CONTU) recognized that the information comprising software innovation in computer programs 'is a prime example of a public good' (Peter Menell, ''An Analysis of the Scope of Copyright Protection for Application Programs,' *Stanford Law Review*, vol. 41, No. 5, May 1989, pp. 1045-1104, citing pp. 9-12 of the CONTU report on p. 1059).

[14] Inappropriability tends t. result in underallocation of resources to the production of these goods. Legally defined ''rights can alleviate the problem of inappropriability. See J.A. Schumpeter, *Capitalism, Socialism, and Democracy (New* York, NY: Harper, 1950); and R.H. Cease, ''The Problem of Social Cost,'' *Journal of Law and Economics,* vol. 3, No. 1, October 1960, pp. 1-44.

[15] See Peter S. Menell, ''Tailoring Legal Protection for Computer Software,'' *Stanford Law Review,* vol. 39, No. 6, July 1987, pp. 1329-1372.
In this article, Menell concludes that copyright protection for computer software does promote some types of innovation by prohibiting piracy, but with some diminution of benefits from increased diffusion of innovations, cumulative innovation and network externalities. Menell suggests a new congressional commission to study software protections. He also suggests that Congress should consider tailoring special protections for software to avoid overextending protection under copyright, and that protection be tailored differently for operating systems and application programs.

- The introduction of monopoly rights[16] can have adverse effects on the market in question. Economic theory holds that a monopolistic producer will tend to produce less of a good and charge a higher price, relative to competitive levels.[17] Moreover, the prospect of monopoly rights can create excessive incentives for certain classes of innovative activity, diverting resources from more socially beneficial ones.
- The monopoly rights can produce unintended, '' spillover' effects in other markets (e.g., transfer market power[18] from the software market to the hardware market or vice-versa).
- Establishing and administering intellectual property systems impose costs on the private and public sectors (e.g., costs to secure rights, litigation and/or negotiation costs, costs to ''invent around' or otherwise avoid infringing others' rights).

Therefore, as Peter Menell has noted, in assessing the need for new (or stronger) intellectual property protections:

> [I]t is important to assess the extent to which other forces-both market and non market-tend to offset the adverse effects of the public goods problem.[19]

These ''other" forces can include leadtime advantages from being first to introduce a product, licensing and service agreements, anticopying technologies, secrecy, government policies concerning antitrust and standards, research support, etc.[20] (For how and why some of these forces may have changed since CONTU, see box 6-A.)

Intellectual Property Systems and Net Social Benefits

The linkage between intellectual property rights and economic benefits to society as a whole has traditionally followed this logic: intellectual property rights increase innovators' ability to obtain returns from their intellectual labors; the resulting potential for increased private gains to innovators stimulates additional innovation; and, *because of increased innovation,* additional benefits accrue to society as a whole.[21] In *Mazer v. Stein, the* U.S. Supreme Court stated this rationale as follows:

> The economic philosophy behind the clause empowering the Congress to grant patents and copyrights is the conviction that encouragement of individual efforts by personal gain is the best way to advance public welfare through the talents of authors and inventors in Science and the useful Arts.[22]

The U.S. system of patents and copyrights is intended to strike a balance between the rights of intellectual property holders and the public at large. It is important to recognize that systems of legal protections for intellectual property impose costs on a society, as well as benefits. These costs include the costs of the additional innovation resulting from the intellectual property incentive, plus administrative

---

16 *OTA note:* In this report, ''monopoly' is used in the economic sense and should *not* be taken as synonymous with illegal monopolization of a market or markets. For discussion, see F.M. Scherer, *Industrial Market Structure and Economic Performance,* 2d Ed. (Chicago, IL: Rand McNally College Publishing Co., 1980), pp. 527-594. As Scherer notes, "Congress [chose] the word 'monopolize' to describe what it condemned, and not some more conventional phrase such as 'obtain ox possess monopoly power' '' (ibid., p. 527),

17 See Scherer (1980), op. cit., footnote 16, pp. *14-20;* and Walter Nicholson, *Intermediate Microeconomics and Its Application, 2d* ed. @retie, IL: The Dryden Press, 1979), ch. 2 ("Pricing in Monopoly Markets), pp. 305-354.

''Monopoly rents' are the long-term economic profits resulting from prices that exceed average total cost: "These profits can be regarded as a return to that factor that forms the basis of the monopoly (a patent, a favorable location, a dynamic entrepreneur), hence another possible owner might be willing to pay that amount in rent for the right to the monopoly" (Nicholson, ibid., p. 309).

18 For a theoretical analysis of leveraging market power, see Michael D. Whinston, '' Tying, Foreclosure, and Exclusion, ' *The American Economic Review*, vol. 80, *No, 4,* September 1990, pp. 837-859.

For a discussion of market power and the practice of 'bundling' software and support with computer hardware, see Franklin M. Fisher et al., *Folded, Spindled and Mutilated: Economic Analysis and U S. vs. IBM* (Cambridge, MA: The MIT Press, 1983), pp. 204-215.

19 Menell (1987), op. cit., footnote 15, p. 1339.

20 Ibid., pp. 1339-1345.

21 For an excellent, comprehensive review of the development and functioning of the patent system, including historical perspectives on institutional issues and economic opinion on the social advantages and disadvantages of patent monopolies, see Fritz Machlup, *An Economic Review of the Patent System,* Study of the Subcommittee on Patents, Trademarks, and Copyrights of the Committee on the Judiciary, U.S. Senate, Study No. 15 Pursuant to S. Res. 236, 85th Congress, 2nd Sess. (Washington DC: U.S. Government Printing Office, 1958). Ch. IV reviews economic theory on patents from the mid- 18th to mid-20th centuries.

22 *Mazer* v. *Stein,* 347 U.S. **201,219** (1954).

and transaction costs[23] and foregone benefits to society from other innovations that would have taken place, without such protection.[24] Strengthening" the system or expanding its scope can increase these costs, and also increase economic losses to society from monopolistic exploitation of innovations:

> From the perspective of the public interest, the optimal system for protecting intellectual work equates the marginal benefit of enhancing the scope of intellectual work protection with the marginal cost of greater protection. . .That is, it equates the benefits of the availability of more and better products with research costs, losses due to monopolistic exploitation, administrative costs, and inhibiting effects on inventive activity. [n]

Nevertheless, much discussion about the U.S. intellectual property system is based on the assumption that, from an economic perspective, "better" or ‘‘stronger’ intellectual property protection is unequivocally ‘good. ’ This is based on the presumption that ‘ ‘stronger’ rights increase private incentives for innovation, producing additional technological progress and increased benefits to society overall.[26] But this is not necessarily the case.

"An economic evaluation of the patent system as a whole implies an analysis of the differences between its existence and nonexistence-perhaps a hopeless task,"

Fritz Machlup, An *Economic Review of the Patent System,* **Study of the Subcommittee on Patents, Trademarks, and Copyrights of the Committee on the Judicary, U.S. Senate, Study No. 15 Pursuant to S. Res. 236, 85th Congress, 2d Sess. (Washington, DC: U.S. Government Printing Office, 1958), p. 76.**

‘The analysis of the ‘increment of invention’ attributable tot he operation of t he patent system, or to certain changes in the patent system, can only be highly speculative, because no experimental tests can be devised to isolate the effects of patent protection from all other changes that are going on in the economy."

Fritz Machlup, *An Economic Review of the Patent System,* **Study of the Subcommittee on Patents, Trademarks, and Copyrights of the Committee on the Judiciary, U.S. Senate, Study No. 15 Pursuant to S. Res. 236, 85th Congress, 2d Sess. (Washington, DC: U.S. Government Printing Office, 1958), p. 62.**

‘The point is that *inventive* activity must precede the patent, whereas *innovating* activity may follow it. But the justification of the patent system as an incentive for innovating enterprise and for entrepreneurial investment would call for different supporting arguments than the justification as an incentive for invention. . .Moreover, there would be the additional question whether the promotion of innovating enterprise and of entrepreneurial investment can be held to be subsumed in the promotion of ‘science and t he useful arts’ which the Constitution *of* the United States stipulated as the sole objective of patent legislation."

Fritz Machlup, An *Economic Review of the Patent System,* Study of the Subcommittee on Patents, Trademarks, and Copyrights of the Committee on the Judiciary, U.S. Senate, Study No. 15 Pursuant to S. Res. 236, 85th Congress, 2d Sess. (Washington, DC: U.S. Government Printing Office, 1958), p. 56.

"Strength" of Protection and Social Benefits

In his recent review of theoretical issues and empirical findings regarding the patent system, Sidney Winter explains why there is now no general theoretical presumption in economics that the consequences of "stronger’ protections are favorable for society as a whole.[27] According to Winter, this contrasts with earlier thinking, when economists

23 Sometimes these transaction costs can be reduced by collective administration of licensing. In copyright markets for music and literary material, where the transaction costs of licensing are large compared to the value of the work to be licensed, collectives have been established. For example, the American Society of Composers, Authors, and Publishers (ASCAP) and Broadcast Music Inc. (BMI) collect and distribute performing rights royalties for composers, songwriters, lyricists, and music publishers; and the Copyright Clearance Center (CCC) collects and distributes photocopying royalties for journal and book publishers.

See Besen and Raskind, op. cit., footnote 2, pp. 14-16; and S.M. Besen and S.N. Kirby, *Compensating Creators of Intellectual Property: Collectives That Collect* (Washington, DC: Rand Corp., 1989). For descriptions of ASCAP and BMI, see U.S. Congress, Office of Technology Assessment, *Copyright and Home Copying:* Technology Challenges *the Law,* OTA-CIT-422 (Washington, DC: U.S. Government Printing Office, October 1989), ch. 4.

24 *Menell* (1987), op. *cit.,* footnote 15, pp. 1339-1345.

25 Ibid., p. 1340.

26 For an important exception (from the law literature) see Stephen Breyer, "The Uneasy Case for Copyright: A Study of Copyright in Books, Photocopies, and Computer Programs, ’ *Harvard Law Review,* vol. 84, No. 2, December 1970, pp. 281-351. In this article, although unable to conclude that copyright should be abolished, Breyer argued against extending the boundaries of copyright. He also concluded that advocates of copyright protection for computer programs had not yet made their case (in terms of empirical evidence and argument) and that if a need for program copyright was shown, efforts should be made to tailor protection to minimize economic harms to society (pp. 349-350).

27 Winter, op. cit., footnote 8, pp. 41-60.

Box 6-A—Some Software Industry and Technology Changes Since CONTU

Explosive growth in personal computers and packaged software: When Congress created the National commission on New Technological Uses (CONTU) in 1974, the "PC revolution" had not yet begun to bring desktop computing power to the millions of individuals that now use it. But, by the time CONTU issued its final report in 1978, the PC revolution was under way, creating a new 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. Large communities of users are now able to obtain powerful hardware and software tools. This broad base of users permits realization of network externalities; users increasingly exhibit preferences for programs with similar (and/or easy-to-learn) user interfaces.

Because of timing, CONTU and the analyses prepared for it could not really foresee the time when powerful computers could be in every office or every home, or when individuals would be able to create sophisticated and valuable software outside large organizations. There was an intimation of the possibilities, but no way to comprehend their full impact a decade hence:

> [T]here are some individuals who believe that future technological changes will permit individuals to do some programming useful to themselves and potentially valuable to others in their own homes. . .Changes in computer and programming technology may also make possible a degree of do-it-yourself programming by consumers. (Report to CONTU prepared by the Public Interest Economics Center (PIE), June 1977, pp. IV-4, IV-6.)

Increased barriers to entry by small firms: The CONTU report and the economic analysis prepared for it in 1977 by the Public Interest Economics Center noted rapid growth in the "independent" software industry, characterized by ''many small firms and rapid entry into the market, " and also noted the likelihood of continued viability for software as a ''cottage" industry. Today, the software industry remains vigorous and there certainly are many successful and innovative small firms. However—unlike when Lotus and Microsoft were founded-the packaged software market is a huge mass market, not a collection of small networks of hobbyists and aficionados. Moreover, as hardware has improved, packaged software programs have become larger and more complex, increasing the cost of writing a commercial program,

A cottage entrepreneur may still be able to develop an innovative and important program with few resources but his or her own wit and time. But, for that program to have substantial commercial success in the retail market, substantial marketing and distribution resources, including national advertising campaigns and customer-support services, will usually be required. Therefore, there has been a trend toward centralized software-publishing houses that acquire rights to software and then distribute and market it, paying royalties to the program authors.

Maturity of industry and increasing firm size: Once a small firm becomes successful, it appears to be relatively difficult to develop the firm to a meaningful size with the capacity to capture a significant fraction of the

considered that stronger protection for intellectual property was desirable to promote innovation.[28] Theoretical and empirical work from the mid- 1970s to present has challenged the assumption that stronger intellectual property regimes yield positive economic consequences. Moreover, industries have been found to vary in the extent to which rights holders are able to acquire competitive benefits from their ''properties" under the current regime. As Winter puts it:

> . . the desirability of strengthening or weakening invention incentives in a particular context depends on the existing balance (in that context) between (1) the joint effectiveness of a variety of means of appropriating returns and (2) the extent to which the advances in question are actually a net contribution to societv. rather than a capture of wealth from the public domain.[29]

Sometimes strengthening patent protection can produce *excessive incentives* for certain types of

[28] ''Arrow's [1962] article also contained a simple (but seminal) formal model of process innovation. The message of this model was unambiguously that even an ideal system of patents (of infinite duration and costlessly enforceable) might well provide an "inadequate' incentive to invent. . .To the extent that any policy implication is inferable from this very simple economic model of invention, it is clearly in the direction of stronger protection for intellectual property rights, and perhaps other measures to reduce transaction costs in markets for such rights. ' (Winter, op. cit., footnote 8, p. 42.)

[29] Ibid., pp. 41-60 (quote from p. 45). Winter draws his empirical evidence concerning functioning of the patent system from a Yale University study conducted by Richard Levin, Alvin Klevorick, Richard Nelson, and Winter. See R, Levin et al., "Survey Research on R&D Appropriability and Technological Opportunity, ' *Brookings Papers on Economic Activity, 1987,* pp. 783-820.

market and to expand internationally. Part of the difficulty seems to come from the technological nature of the enterprise, A startup is often technology-oriented: the founder may be chief programmer, perhaps designer of a single product. As the firm grows, it needs to shift to a market (rather than technology) focus: advertising, packaging, and user support become increasingly essential. A corresponding increase in personnel may also require the firm to acquire new managerial expertise. The notion that a firm must achieve a certain size and have a certain amount of marketing and managerial savvy in order to be a significant player in the software industry seems to receive support from the increasing number of software mergers and acquisitions.

Acquiring or merging with a company with similar product offerings can be efficient. It maybe less costly to acquire a small firm with a niche product, rather than developing it internally; conversely, once a small niche company has peaked, its logical evolution may lie in being acquired by an established firm, rather than trying to expand. Business alliances with firms that are "complementary" in terms of products, technologies, and market focus can also be very advantageous—the 1991 alliance between IBM and Apple is a notable example.

Increasing concentration: It still seems to be true—but perhaps not so true as in the early 1980s--that the software industry is a haven for small entrepreneurs. One estimate is that there are 9,000 to 12,000 software companies in the United States. However, a closer look shows that the industry is relatively concentrated, dominated by the top 200 companies or so. Distinct segments of the industry illustrate this relative concentration: according to Soft-Letter, the "Top 20" PC software companies reported 1990 revenues of about $3.3 billion. The "Top 5," however, accounted for two-thirds of these revenues, with the "No. l" firm accounting for almost 30 percent of the total (for the "Top 10," see table 3-2 inch. 3).

Prominence of hardware manufacturers as software producers: CONTU saw the software-market shares of hardware producers being ''steadily eroded" by independents and concluded that, with software protection, competition in software would be enhanced in the future due to entry. *(CONTU Report,* p. 24). Despite vigorous growth by the independent software sector, the hardware firms retain a major share of the software market overall. In terms of revenues, IBM is the leading U.S. software producer overall and is the largest packaged software vendor in the world (see ch. 3).

SOURCES: OTA workshop on "Software Developers' Business Needs and Concerns, " Sept. 25, 1990; also Everett M. Rogers and Judith K. Larsen *Silicon Valley Fever (New York:*Basic Books, Inc., 1984); Robert Schware, *The World Software Industry and Software Engineering* (Washington DC: The World Bank, 1989); *Final Report of the National Commission on New Technological Uses of Copyrighted Works (CONTU),* July 31,1978 (referred to by OTA as *CONTU Report); ''AnAnalysis* of Computer and Photocopying Copyright Issues From the Point of View of the General Public and the Ultimate Consumer,' Public Interest Economics Center (Washington DC: June 1977); INPUT, *U.S. Software Products Market, 1988-1993 (Mountain* View, CA: INPUT 1988); Organization for Economic Cooperation and Development *Internationalization of Software and Computer Services* (Paris, France: OECD, 1989); data compiled by *Soft-Letter* (Watertown, MA: 1991).

inventive effort. Private returns from an invention are usually expected to be lower than the benefits accruing to society.[30] However, there are exceptions. For certain types of inventive efforts, private returns may exceed social returns. This might occur, for example, if the inventor is able to invest in complementary assets whose prices will tend to increase after his or her invention is disclosed and implemented. The inventor's gain from these complementary assets need not be related to (or bounded by) the social value of the invention.[31] The possibility for extraordinary returns from inventions with valuable complementary assets might divert inventive efforts from other areas that might be more beneficial to society as a whole, although less rewarding to the individual rights holder.

Another source of excessive incentives for inventive activities comes about from the possibility of multiple inventors. In a dynamic ''race to patent, '

30 Mansfield's 1977 studies comparing average social and private rates of return from investments in innovation in a variety Of industries found that estimated social returns exceeded private ones, usually quite substantially. *IOTA note: This* aggregate approach did not compare public and private returns for particular inventions or inventors.] See Edwin Mansfield, ''Intellectual Property, Technology and Economic Growth,'' in *Intellectual Property Rights in Science, Technology. and Economic Performance,* Francis W. Rushing and Carole Ganz Brown (eds.) (Boulder, CO: Westview Press, 1990), ch. 2, esp. pp. 20-22.

31 Winter, op. cit., footnote 8, pp. 42-43. See also J. Hirshleifer, ' 'The Private and Social Value of Information and the Reward to Inventive Activity, ' *American Economic Review,* vol. 61, 1971, pp. 561-574; and Whinston, op. cit., footnote 18.

new technologies may be implemented too early-it might be better from a social perspective to wait and allow costs to decline due to general technological advance. Also, assuming the possibility of multiple independent discoveries, a patent might be granted for an invention that would otherwise appear in the public domain during the useful lifetime of the patent.[32]

Looking at the interplay of economic growth and patent and antitrust policies, Janusz Ordover has concluded that "weak" patent protection need not hamper economic growth, if appropriate industry cooperation is allowed. At the same time, ''strong' protection need not hinder diffusion of advances:

> Appropriately structured patent law and antitrust rules can together ensure incentives for R&D [research and development] and also induce cooperation among firms in diffusing R&D results through licensing and other means. . .At the same time, cooperation among firms at the R&D stage can counterbalance weak patent protection by internalizing spillovers from ongoing R&D programs, and such cooperation may also produce additional spillovers from existing knowledge.[33]

Ordover's examination of intellectual property and antitrust regimes in the United States, Japan, and the European Community, suggests, however, that the U.S. policy mix may have gone too far in favoring the interests of the inventor:

> . . the current U.S. policy thrust of strengthening patent protection while reducing antitrust restrictions on the exercise of intellectual property rights is not a precondition for high rates of growth and for the maintenance of R&D incentives.[34]

Industry Differences in the Perceived Efficacy of Intellectual Property

The extent to which the effects of intellectual property regimes on market behavior are manifested, as well as the social desirability of these outcomes, may vary depending on the characteristics of the market and technology.[35] In theory, a patent confers perfect appropriability from a technological advance of requisite quality in exchange for public disclosure and widespread use of the advance after the patent term has expired. In practice, patents often do not confer perfect appropriability (e.g., because they can be invented around or are too hard to enforce) and public disclosure of a patent claim need not assure eventual diffusion of the *knowledge* required to make economic use of an advance. Substantial resources may be required to imitate an innovation, even one in the public domain.[36] Moreover, firms may be able to ''pyramid' the benefits they derive from their patents by "fencing in" a field of technology through systematic patenting.[37]

In general, patents are thought to discourage entry into a market by raising the costs of entering with a "close" substitute or by deterring entry entirely, when the costs of licensing or inventing around the patent are too high. By comparison, copyright (which is not intended to bar independent creation)

---

32 Winter, op. cit., footnote 8, pp. 43-44. See also Y. Barzel, "Optimal Timing of Innovations," *Review of Economics and Statistics,* vol. *50, 1968,* pp. 348-355; and R.R. Nelson and S.G. Winter, *An Evolutionary Theory of Economic Change* (Cambridge, MA: Harvard University Press, 1982).

A contrasting view is that in some cases, early delimitations of patent rights can be used to make R&D resource allocation more efficient. This is called the ''prospect theory" of patents. See E. W. Kitch, "The Nature and Function of the Patent System," *Journal of Law and Economics,* vol. 20, 1977, pp. 265-290. But see Robert P'. Merges and Richard R. Nelson, ''On the Complex Economics of Patent Scope, ' *Columbia Law Review, vol. 90, No. 4,* pp. 839-916, esp. pp. 871-878.

33JanuszA.Ordover,''A Patent System for Both Diffusion and Exclusion,'' *Journal of Economic Perspectives,* vol. *5, No.* 1, winter 1991, pp. 43-60 (quote at p. 44). Ordover suggests that dichotomies like "long-term v. short-term" and ''exclusion v. diffusion' are too simple to be used as tools to shape public policy concerning patents.

34 Ibid., pp. 44,59.

35 For an equilibrium analysis of the relationship between market structure (including patent terms) and firms' decisions to invest in R&D, see Glenn C. Loury, "Market Structure and Innovation" *The Quarterly Journal of Economics,* vol. *93, No. 3,* August 1979, pp. 395-410.

For analysis of the effects of patent scope on invention, taking into account differences in industrial patterns of technological advance, especially in cumulative technologies, see Merges and Nelson, op. cit., footnote 32, pp. 880-908.

36 Richard C. Law@ ''A New Look at the patent System," *AEA Papers and Proceedings,* vol. 76, No. 2, May 1, 1986, pp. 199-202.

37 See Scherer, op. Cit., footnote 16, ch. 16, esp. pp. 450-452. Scherer also discusses institutional aspects of patent prosecution and administration.

is generally thought to impose lower barriers to entry.[38] In terns of economi, welfare, barriers to entry are not uniformly ‘ ‘bad’ in some cases, they will be beneficial. Analysis of which condition pertains can be quite complex and depends, in part, on whether goods will be produced in excessive variety (relative to the socially optimal amount of variety), absent barriers to entry,[39]

In instances where consumers care little about product variety, theory suggests that barriers substantial enough to reduce the number of very similar products (e.g., patents) will be beneficial. Conversely, when consumers value variety highly, an intellectual property regime that allows multiple products with similar features (e.g., copyright) will be preferred.[40] When consumers place a high value on products with similar features, the value of a copyright (to the copyright owner) will increase to the extent that it allows control over preparation of works with similar features (e.g., allows control of user interfaces).[41]

Under the present patent system, the ability of innovators to appropriate returns via patents appears to differ across industries and technologies.[42] In the 1980s, a multi-industry study on R&D appropriability (the ‘ ‘Yale study’ was conducted by Richard Levin, Alvin Klevorick, Richard Nelson, and Sidney Winter. This research explored industry differences in the perceived effectiveness of patents in gaining and maintaining competitive advantages from new or improved products and processes.[43] The researchers surveyed industry R&D managers concerning the relative effectiveness of:

- patents to prevent competitors from duplicating the new product or process;
- patents to earn royalty income through licensing;
- maintaining secrecy about the new product or process;
- leadtime advantage (from being first out with a new product or process);
- moving quickly down the learning curve (in order to reduce costs); and
- superior sales or service capabilities.

The Yale study concluded that the role of a patent in gaining and maintaining competitive advantage depends in large part on specifics of the relevant industry and technology. For example, in an industry where inventions tend not to be technologically independent of other patented inventions, firms with patent portfolios may participate in patent cross licensing. Even under these circumstances, breakthrough, “pioneer” patents can yield large advantages through excluding others and/or producing royalty income.[44]

The Yale study found a great deal of inter-industry variation in the perceived effectiveness of patents as a means of securing innovative gains, even in ‘ ‘high-tech, ’ R&D-intensive industries:

38 However, some legal observers warn that overly-broad copyright protection for software—e.g., copyright proteCtion for user interfaces—is equivalent to patent protection, except that the protection has been obtained for an extremely long term and without patent criteria of novelty and nonobviousness. (See Pamela Samuelson, ‘ ‘Why the Look and Feel of Software User Interfaces Should Not Be Protected by Copyright Law, ” *Communications of the ACM*, vol. 23, No, 5, May 1989, pp. 563-572.) Others consider that software faces a more general problem in that attempts to apply traditional copyright principles and precedents may well restrict efficient technology development. (See Dennis S. Karjala, ‘‘Copyright, Computer Software, and the New Protectionism,” *Jurimetrics Journal*, fall 1987, pp. 33-96.)

Disagreeing with these views, many other legal observers consider that the courts can be (and generally have been) successful in adapting traditional copyright principles to software cases. (See Morton David Goldberg and John F. Burleigh, ‘‘Copyright Protection for Computer Programs: Is the Sky Falling?’ *AIPLA Quarterly Journal*, vol. 17, No. 3, 1989, pp. 296-297; and Anthony L. Clapes, Patrick Lynch, and Mark R. Steinberg, “Silicon Epics and Binary Bards, ” *UCLA Law Review*, vol. 34, June-August 1987, pp. 1493-1594.

39 See, e.g., Michael Waterson, ‘‘The Economics of Product Patents, ’ *The American Economic Review*, vol. *80, No. 4*, September 1990, pp. 860-869.

40 Waterson, op. cit., footnote 39, pp. 867-869. For analysis of the economic effects of increased copyright protection where consumers vary in their tastes and in their costs of copying, scc Ian E. Novos and Michael Waldman, ‘ ‘The Effects of Increased Copyright Protection: An Analytic Approach, ’ *Journal of Political Economy*, April 1984, pp. 236-246.

41 Michael O’Hare, “Copyright: When Is Monopoly Efficient?” *Journal of Policy Analysis and Management, vol. 4, No. 3, 1985,* pp. 407418. Copyright gives exclusive rights over derivative works, to the extent that protected expression is used.

42 Fo, theoretical analysis of how the preferred form of research incentive can vary depending on market structure, see Brian D. Wright, ‘‘ne Economics of Invention Incentives: Patents, Prizes, and Research Contracts, ” *The American Economic Review*, vol. *73,* September 1983, pp. 691-707. Wright finds that when the probability of success of a project is high, contracts are preferred; when the probability of success is low to moderate and the supply of research is elastic (increases in expected financial rewards attract new researchers), patents are best (ibid., p. 703).

43 Winter, op. cit., footnote 8, pp. 45-56. Sec also R. Levin et al., op. cit., footnote 29, pp. 783-820.

44 In 1958, Machlup noted that strategic patents and/or restrictive licensing can permit substantial control of an industry by blocking or excluding rivals; accumulation or aggregation of patents can bar entry by newcomers. He also noted that ‘‘patent pooling’ (cross licensing) has ‘‘often been the vehicle for cartel agreements of the most restrictive sort. (Machlup, op. cit., footnote 21, pp. 10-12.)

- Patents were perceived as highly effective in the pharmaceutical industry[45] but were rated as being only moderately effective in the semiconductor, computer, and communications equipment industries;
- Most industries reported that using patents to secure royalty income! was less effective than using patents to prevent duplication;
- Except in petroleum refining, patents were considered more effective in preventing product duplication than process duplication;
- With the exception of product patents in the organic chemical and pharmaceutical industries, using patents to prevent duplication was perceived as only moderately effective in securing gains from an innovation.[46]

The Yale study found that the most important perceived limitation to the effectiveness of patents was "inventing around" by rivals able to invent a substitute product or process or narrowly skirt the edges of the patent claims. The fast pace of technological advance (''technology moving so fast that patents are irrelevant") was considered much less of a limitation. Perceived problems of "not readily patentable" subject matter and ''patents disclose too much proprietary information" were more severe for new or improved processes, rather than products. These considerations help explain the importance of other tools—like lead time and secrecy—in securing and preserving competitive advantages .47

The terms under which a patent is licensed depend on a number of factors, including the relative bargaining powers of the patent holder and prospective licensee, and how much information each has about market conditions, the invention, and the economic value of the patent.[48] Patent licensing can provide licensers with other advantages besides royalty income. In some cases, strategic licensing will permit a patent holder to structure the industry so it consists of relatively 'weak' competitors. This structure enables the patent holder to prolong its dominant position after the patent term has expired.[49]

Cumulative Technological Progress

The literature on "patent races" focuses on analyzing situations where multiple innovators are vying to achieve the same goal.[50] A somewhat different set of circumstances arises when innovation is cumulative--one firm develops an initial innovation but others can build on it. In this model, the incentives to develop both the initial and subsequent innovations depend on the scope (breadth) of patent protection.[51] The traditional literature did not focus on the dynamics of cumulative progress and multiple inventors.

Suzanne Scotchmer has looked at use of (product) patent protection and cooperative agreements to protect incentives for cumulative research where initial innovations facilitate subsequent ones.[52] She finds that with ***broad patent protection,*** economic incentives for outside firms to develop second-generation products (under license) can be too weak. The incentives for the frost innovator to develop the second-generation products (assuming he or she has the insight and resources to do so) are much stronger; in some cases, the prospect of licensing revenues for second-generation products may inefficiently inflate

45 In the pharmaceutical industry, a patent often corresponds to one product.

46 Winter, op. Cit., footnote 8, pp. 46-49, especially table 2.1. The study included the computer, semiconductor, and communications equipment industries, but not a separate "software" industry. Winter notes that these findings support both Arrow's view that transaction costs present serious problems for information markets (i.e., patent licensing) and also Kitch's view on the value of patents in making R&D markets more efficient.

47 Ibid., pp. 48-56, esp. table 2.2 and figure 2.1.

4S For discussion of literature on licensing arrangements and analysis of arrangements when the patent holder must decide how much information to share with the licensee, see Nancy T. Gallini and Brian D. Wright, ''Technology Transfer Under Asymmetric Information" *RAND Journal of Economics*, vol. *21, No.* 1, spring 1990, pp. 147-160.

49 For analysis of how a dominant patent holder may prefer to license to technologically weaker competitors, whose presence in the industry may deter entry by other fins, even after the patent has expired, see Katharine E. Rockett, "Choosing the Competition and Patent Licensing, ' RAND *Journal of Economics*, vol. 21, No. 1, spring 1990, pp. 161-171. Rockett cites examples of the development and licensing of polyester, cellophane, and nylon as evidence to support the general assumption that powerful patentholders can use licensing to structure their competition and preserve competitive advantages beyond the patent term.

50 The ''race'' metaphor assumes that only one firm can win the patent and that inventions are discrete.

51 Besen and Raskind, op. cit., footnote 2, p. 4.

52 Suzanne Scotchmer, "Standing on the Shoulders of Giants: Cumulative Research and the Patent Law," *Journal of Economic Perspectives*, vol. *5, No.* 1, winter 1991, pp. 29-41.

incentives for the first innovation.[53] With *narrow patent protection,* first-generation innovators may be discouraged from patenting and disclosing their products (if they fear that second-generation products developed by others will hurt their profits). They might, instead, hold frost-generation products off the market until they had developed their own second-generation products; alternatively, they might rely on trade secret protection to avoid disclosing the nature of their technological advances.[54]

Scotchmer explores two possible remedies for these defects:

> *narrowing the scope of protection* for the first patent (so that 'different enough' second-generation products do not infringe), and
>
> *prior agreements,* whereby second-generation innovators can sell their ideas to innovators of the first products or integrate with them.

She finds that:

> [N]o system of narrower patent protection and licensing can give the right incentives to both the first innovator and other firms that develop improvements, even if collusive licensing among noninfringing products were allowed. . ,[this] result depends on my premise that the breadth of the underlying patent cannot be separately tailored to the costs and benefits of each second generation product.[55]

Scotchmer also finds that—assuming that parties to the agreement can collude in using the resulting patents—prior agreements (e.g., research joint ventures) are:

> [A] social improvement over licensing because they can improve incentives to invest in second generation products, whatever the breadth of patent protection,[56]

She finds, therefore, that there are no simple conclusions to draw about the optimal breadth of patents; moreover, questions about patent breadth must be considered jointly with questions about the extent to which firms will be permitted to cooperate (under antitrust rules). Therefore, Scotchmer concludes that prospects for fine-tuning the patent system for particular technologies seem limited,[57]

Considering further the tradeoffs between disclosure and profitability, Suzanne Scotchmer and Jerry Green examine the effect of the novelty requirement in patenting on information sharing and economic welfare in industries where progress is cumulative .58 Scotchmer and Green proceed from the premise that the *effective* lifetime of a patent may be much shorter than its statutory term—the patented technology may be displaced by a (noninfringing) and superior innovation. Thus the novelty requirement is important in patent design. A high standard of novelty makes displacement harder and, in principle, gives a larger return to the patent holder, However, a high standard of novelty also bears a social cost in that relatively minor incremental innovations may not be patented—thus, not disclosed. High standards may also encourage firms who fall behind technologically to drop out of the patent race. This can be beneficial if it reduces duplicative research by technologically less-advanced fins. When the novelty requirement is weak, these firms may have too strong an incentive to stay in the patent race.[59]

---

53 Ibid., pp. 32-33. The prospect of licensing revenues may overinflate incentives for the first product innovation when it is not the only possible way to achieve the second-generation innovations. Scotchmer's analysis assumes that the second-generation product infringes the patent on the first and therefore, some of the second innovation's revenues must be transferred to the first innovator under licensing.

54 Ibid., pp. 38-39.

55 Ibid., pp. 33-35.

56 Ibid., p. 36, Social benefits would be largest if all possible innovators could be integrated via an agreement prior to the first innovation; however, this is likely to be infeasible, particularly in fields where second-generation innovations are serendipitous.
In Scotchmer's model, collusion in use of the patents is an important way to protect incentives to innovate. When research outcomes are unpredictable, innovators cannot know beforehand whether particular projects will result in innovations that infringe an earlier patent; therefore, rules that permitted agreements only if patents turned out to be infringing would be difficult to implement (pp. 36-37).

57 Ibid., pp. 39-40. In Scotchmer's model, patent breadth and term may sometimes be chosen independently; in other cases, breadth and effective term are correlated.

58 Suzanne Scotchmer and Jerry Green, "Novelty and Disclosure in Patent Law," *RAND Journal of Economics,* vol. 21, No, I, spring 1990, pp. 131-146. In their analysis, the patent system's criteria for novelty and nonobviousness determine both how broad the claims of a patent can be and how different subsequent innovations must be not to infringe. Thus, these criteria determine the value of a patent, the incentives for innovative activity, and how much technological information is disclosed and diffused.

59 Ibid., pp. 144-145; and Garth Saloner, "Introduction to Symposium on Patents and Technology Licensing," *RAND Journal of Economics,* vol.21, No. 1, spring 1990, p. 104. Another result from a weak novelty requirement is that firms may prefer not to disclose all of their technological progress, unless they are able to enter into cooperative licensing agreements with firms producing similar products.

In the copyright regime, the dilemma of 'cumulative progress'' is manifested in controversy over the breadth of protection extended to derivative works, Some economists have concluded that narrower protection for the original work will better serve the public interest by providing incentives for others to create derivative works. In the software arena, these commentators consider that making it illegal under copyright to copy in order to tailor programs to users' special needs, or to copy in order to analyze programs for the purpose of improving and enhancing functions, raises the cost of subsequent innovation.[60] Others, however, conclude that granting control of derivative works to the creator of the original work encourages early release of the original work. Otherwise, they argue, proprietors would attempt to secure market advantages by delaying release until they prepared derivatives.[61]

Patents and Trade Secrets

David Friedman, William Landes, and Richard Posner have examined the economics of trade secret law, focusing on the choice between patent and trade secret protection and why the law does not protect against loss of trade secrets through reverse engineering.[62] In their view, although the courts have sometimes thought that trade secret law protects a "lesser' class of inventions (because "no rational person with a reasonable invention would fail to seek a patent' '), this is not the case. Instead, they argue, trade secret law supplements the patent system. Innovators choose to maintain an innovation as a trade secret instead of applying for patent protection when patent protection is too costly relative to the value of the innovation, when the expected economic lifetime of the innovation is expected to be longer than the patent term, or when the innovation may not be sufficiently novel or nonobvious to qualify for patent protection. There are some social costs to this-with trade secrecy, there is no finite term of protection or disclosure in return for protection. However, these costs are somewhat ameliorated because independent invention (and patenting) of the "secret'' is permitted.[63]

Not prohibiting reverse engineering of trade secrets is costly to the extent that proprietors incur additional costs to maintain the secret or make their products hard to reverse engineer. Nevertheless, these authors conclude, the respective costs and benefits weigh against protecting trade secrets from reverse engineering: For one thing, the social costs of enforcing secrecy through the legal system would be high: for example, it could be difficult to prove that a competitor learned a trade secret through reverse engineering rather than through independent research.[64] Perhaps even more importantly, the information disclosed through reverse engineering facilitates incremental innovation.[65] On balance, Friedman, Landes, and Posner consider that more comprehensive legal protection of trade secrets as property, ''would be tantamount to a perpetual patent law without public disclosure," without the economic efficiency advantages of disclosure and limited terms.[66]

Choices in the Optimal Design of Patents

Rewards to innovation can be granted by broad patents of short duration or by patents designed with narrow scope (breadth) but long duration (term). The supply of R&D and other innovative activities is usually assumed to respond favorably to improved prospects for financial reward-i. e., increased incentives are thought to induce additional innovative

60 Besen and Raskind, op. cit., footnote 2, pp. 16-17 (citing Y.M. Braunstein et al., *Economics of Property Rights as Applied to Computer Software and Data Bases, PB-286 787* (Washington, DC: U.S. Department of Commerce, 1977); and R.H. Stem, "Section 117 of the Copyright Act: Charter of Software Users' Rights or an Illusory promise?' *Western New England Law Review*, vol. *7, 1985*, pp. 459-489).

61 Besen and Raskind, op. cit., footnote 2, p. 16 (citing W.M. Landes and R.A. Posner, ''An Economic Analysis of Copyright Law, *Journal of Legal Studies*, vol. 18, 1989, pp. 325-366).

62 Friedman et al., op. cit., footnote 4.

63 Ibid., pp. 62-66. Friedman, Landes, and Posner note that secrecy contrasts with the ''prospect'' theory of patenting, where disclosure serves to head off wasteful duplication of inventive effort (p. 65). See Kitch, op. cit., footnote 32.

64 For discussion of the some of the difficulties of trade secret protection, including contractual and evidentiary problems in enforcement, see Steven N.S. Cheung, "Property Rights in Trade Secrets, " *Economic Inquiry, vol. 20*, January 1982, pp. 40-53.

65 Friedman et al., op. cit., footnote 4, pp. 69-71.

66 Ibid., p. 71.

activities.[67] As Garth Saloner observes, determining the optimal size of the reward that a patent should confer and the optimal design of the patent (breadth, term) depends on the extent to which society wishes to encourage additional innovative activities beyond those that would take place otherwise, absent patents. There is a tradeoff between the social benefits realized through economic stimulation of innovative activities provided by the promise of a patent and the social costs later induced by the market power that a patent confers; the magnitude of these social costs depends on the manner in which innovators choose to patent, license, and otherwise exploit their discoveries.[68] Therefore, Saloner *notes,* normative questions of optimal patent design and positive questions of how firms behave are closely related; both must inform public policy.[69]

Aggregate data on R&D, patenting, and technological progress generally support theoretical findings that patents encourage innovative activities and increase economic welfare .70 But the more detailed theoretical models of optimal patent design will yield different-even contradictory-conclusions when based on different assumptions about the value of patents[71] and the behavior of innovators and markets (see below). Disaggregate empirical evidence to distinguish among these for the purposes of optimizing patent design is lacking. In terms of empirical evidence to support unambiguous ‘ ‘improvements ‘‘ in patent design, we have not moved very far from Fritz Machlup’s conclusions:

> One **important** moral of the argument [about the effects of an extension of patent protection by 1 year] is that no one who thinks it through can be very sanguine concerning the effects to be expected in ‘‘reality’ and certainly, no one cart be at all sure about any of these matters.[72]

Optimal Patent Term

Along with its breadth, the term of a patent is related to its ‘‘strength. (As was discussed earlier, thinking about whether ‘‘stronger’ rights are unequivocally ‘‘better’ has changed.) Economists have a long history of participation in the debate on ‘‘optimal’ patent terms.[73] As Fritz Machlup related in his 1958 review of the patent system:

> The duration of patents has been determined by historical precedent and political compromise. The 14-year term of the English patents after 1624 was based on the idea that 2 sets of apprentices should, in 7 years each, be trained in the new techniques. . . There were all sorts of arguments in later years in favor of a longer period of protection: it should be long enough to protect the inventor for the rest of his life; to protect him for the average length of time for which a user of the invention might succeed in keeping it secret; or for the average time it would take others to come up with the same invention; or for the average period in which investments of this kind can be amortized; and some pleas were made for perpetual coverage.
>
> Economists usually argued for shortening the period of protection: the bulk of inventions are not so costly as to require the stimulus provided by

67 However, Merges and Nelson conclude that, ‘‘Ultimately it is important to bear in mind that every potential inventor is also a potential infringer. Thus a ‘ ‘strengthening” of property rights will not always increase incentives to invent; it may do so for some pioneers, but it will also greatly increase an improver’s chances of becoming enmeshed in litigation. .When a broad patent is granted or expanded via the doctrine of equivalents, its scope diminishes incentives for others to stay in the invention game, compared again with a patent whose claims are trimmed more closely to the inventor’s actual results. ’ (Merges and Nelson, op. cit., footnote 32, p. 916.)

68 Saloner (1990), op. cit., footnote 59, pp. 103-105.

For a development and analysis of different mechanisms to secure rights for innovators (the “innovation patents’’ and the ‘‘l~ovation warrant’’), see William Kingston (cd.), *Direct Protection of Innovation* (Dordrecht, The Netherlands: Kluwer Academic Publishers, 1987). This study was prepared under contract for the Commission of the European Communities, Directorate-General Telecommunications, Information Industries and Innovation.

69 Ibid,

70 For example, see William Nordhaus, *Invention, Growth, and Welfare: A Theoretical Treatment of Technological Change (Cambridge!* MA: MIT Press, 1969); Zvi Griliches (cd.), *R&D, Patents and Productivity (Chicago,* IL: Chicago University Press, 1984); Zvi Griliches, ‘‘*patent* Statistics as Economic Indicators: A Survey, ” *Journal of Economic Literature,* vol. 28, No. 4, December 1990, pp. 1661-1707; and Machlup, op. cit., footnote 21, pp. 76-80.

71 There is extremely wide variance in the economic value of Patents, Griliches reports that, although a few smaller-scale studies have been done subsequently, the only detailed and extensive survey of patent holders was conducted in the late 1950s by Barkev Sanders and associates at the Patent and Trademarks Foundation. Economic gains reported from patents then in current usc were widely dispersed, with a mean value of $577,000 per patent and a median value of $25,000 (current dollars). Economic returns from all patents (including those not in usc or with negative returns) had a mean of $112,000 and a median c1osc to Zero. (Griliches (1990), op. cit., footnote 70, pp. 1679 -1680.)

72 Machlup, op. cit., footnote 21, pp. 70--73 (quote from P. 73).

73 As described by Sidney Winter, the pendulum of opinion **011 the “Optfi~’ term** of protection (e.g., whether increasing or decreasing the term of patent protection would be more socially desirable) has swung back and forth over the years, For his discussion of changes in economics thinking about the term and strength of protection, see Winter, op. cit., footnote 8, pp. 41-43.

> protection for such a long time, and not important enough to deserve the reward that it affords; a much shorter period would provide sufficient incentive for almost the same amount of inventive activity; the period should not be so long as to allow patentees to get entrenched in their market positions; "technology moves now with a speed once undreamed of—its swift march dictates a shortening in the life of a patent. '[74]

The cyclical debate has continued into our time. In the 1960s, following At-row's seminal paper, economists turned to mathematical analyses of the optimal patent term. William Nordhaus extended Arrow's model of incentives for invention, exploring the relationship between the ''size' of an invention (measured as the decrease in unit production cost of a good) and the patent term. Nordhaus then formulated the socially optimal patent term by maximizing social welfare subject to the inventor's behavior,[75] showing that shorter patent lives were preferable to longer ones.[76]

F.M. Scherer interpreted Nordhaus' theory geometrically and showed that (process) patents induce investments in R&D in two ways. The first of these, the ''stimulus effect, was emphasized by Nordhaus. In this role, the optimal patent provides just enough incentive for additional R&D so that the marginal social gain from further cost reductions just equals the marginal social cost. In its second role, the patent persuades investors that competitive imitation will be deferred long enough so that the stream of (discounted) monopoly rents will exceed outlays for investments in additional R&D.[77]

Scherer's model indicated that 'easy' inventions—those yielding big costs savings in relation to R&D resources invested-warranted shorter protection than "hard' inventions, unless the inventor had faced extraordinary risks or had exhibited extraordinary creativity. Scherer concluded that policies tailoring the life of each patent to the economic characteristics of the invention might be accomplished by a flexible system of compulsory licensing, where a patent would expire or be licensed openly at modest royalties after 3 to 5 years, unless the patent holder could show that special conditions warranted longer terms (e.g., market is small relative to research costs, the invention will yield small cost savings relative to research costs). For patents held by dominant corporations with large market shares, Scherer suggested that a presumption of early expiration or open licensing would be appropriate because the firm would have other ways of appropriating innovation gains.[78]

Empirical studies of R&D incentives and rewards suggest that the *useful economic lifetime* of a patent is often far shorter than the statutory term. From a multi-industry survey of R&D fins, Edwin Mansfield found that the effective economic lifetime of most patents is much less that 17 years; moreover, he found that about 60 percent of patented products were successfully imitated by others within 4 years.[79]

Optimal Patent Term and Breadth

More recent analyses have considered breadth, as well as length, as tools with which to craft optimal patent designs. Gilbert and Shapiro examine the tradeoffs between patent term and breadth in designing a patent to provide a reward of a given size,[80] Their model of the flow of profits available to the patent holder assumes a predictable underlying environment and homogeneous products, where the breadth of the patent corresponds to market power—the broader the patent, the greater the ability to increase price over marginal cost. With this model, increasing patent breadth yields a greater flow of profits but also increases the dead weight losses

---

74 Machlup, op. Cit., footnote 21, p. 9. The quote about the *swift* pace of technological advance is from 1941. Machlup discusses the implications of changes in patent term on pp. 66-73.

75 See also Morton J. Kamien and Nancy L. Schwartz, "Patent Life and R&D Rivalry," *The American Economic Review,* vol. 64, March 1974, pp. 183-187.

76 Nordhaus, op. cit., footnote 70.

77 Scherer calls this the ''Lebensraum effect.''

78 F.M. Scherer, ''Nordhaus' Theory of Optimal Patent Life: A Geometric Reinterpretation" *The American Economic Review,* vol. *62,* June 1972, pp. 422427.

79 Edwin Mansfield, ' 'R&D and Innovation: Some Empirical Findings, " in Zvi Griliches (cd.), *R&D, Patents and Productivity (Chicago, IL:* Chicago University Press, 1984). (Cited in Scotchmer and Green, op. cit, footnote 58.)

some the question of how much t. reward patent holders requires some assumptions about the elasticity of supply of inventions, designing an economically efficient patent with a given reward does not. See below.

stemming from the patent monopoly .81 Under these conditions, Gilbert and Shapiro conclude that longer patent terms are preferable to broader patents. However, they warn, their assumptions of a stable environment (with predictable supply and demand) and their focus on a single product is critical to this conclusion. Under other circumstances-for example in markets where research is cumulative—an overly long patent might block subsequent innovations by establishing monopoly rights over an entire line of research. In that case, optimal patents would be tailored differently.[82]

Paul Klemperer uses a different model to explore the tradeoff between patent term and breadth in designing a patent yielding a reward of a given size.[83] In his model, entry into the market is free and substitute goods are available. Consumers are assumed to prefer the patented good, but they vary in their demands and their costs of switching to less-preferred goods (either close rivals or goods in a different product class). As the breadth of the original patent decreases, rivals are able to position their substitute goods closer and closer in product space. Conversely, in Klemperer's model, a broader patent corresponds to fewer close substitutes, more market power for the original patent holder, and larger welfare losses.

Klemperer's model yields differing results, depending on how consumers vary in their costs of substituting to close rivals and substituting out of the product class entirely. If consumers face similar costs of substituting rival products (close substitutes) but vary in their costs of switching out of the product class, then Klemperer also shows that the optimal patent is very narrow and long-lived. However, if consumers have similar costs of switching out of the product class but vary in their costs of substituting rival products, then his model shows that optimal patents are broad and short-lived.[84]

"Protection of the general good is found in the limited term and stringent standards associated with patents, the proscription of the protection of ideas under copyright, and the refusal to allow the extension of patents and copyrights beyond their limited scopes. This last matter may be the heart of the concern about the economic effects of program copyright."

Final Report of the National Commission on New Technological Uses of Copyrighted Works (CONTU), July 31, 1978, p. 23.

Compatibility, Network Externalities, and the Installed Base[85]

For many products, the satisfaction a user derives from consuming the good increases as the good becomes more popular. As Michael Katz and Carl Shapiro explain, these positive externalities can arise for a number of reasons, besides the purely psychological, "bandwagon" effect": market size and share may signal product quality to consumers, or product information may be more plentiful or accessible for popular products. Some products, such as computer software and hardware, also exhibit what are called *network externalities,* where user satisfaction is greater, the more popular the network of users. Network externalities-manifested, for example, in users' preferences for 'popular' programs with established user communities, or for programs performing different functions but having similar user interfaces-are much more important now than at the time of CONTU. As the "PC revolution' has taken place, "networks' and their corollary benefits have become much larger.[86] (See discussion of software network externalities in ch. 4.)

Network externalities sometimes arise from a direct physical effect on product quality (e.g., telephones and facsimile machines become more

81 Richard Gilbert and Carl Shapiro, "Optimal Patent Length and Breadth, " *RAND Journal of Economics,* vol. 21, No. 1, spring 1990, pp. 106-112. The dead weight loss comes about because monopoly power permits the patent holder to maximize profits by producing the patented good in fewer quantities, and charging higher prices, than under competition.

82 Ibid,, pp. 107, 111-112.

83 Paul Klemperer, ' 'How Broad Should the Scope of a Patent Be?'' *RAND Journal of* Economics, vol. 21, No. 1, spring 1990, pp. 113-130.

84 Ibid., pp. 126-127; and Saloner, op. cit., footnote 59, p. 104.

85 *OTA note.* A separate OTA report on *Global Standards: Building Blocks for the Future* will be published in early 1992.

86 Peter Menell notes that CONTU ' 'entirely overlooked' network externalities in its analysis and recommendations (Menell, May 1989, op. cit., footnote 13, p. 1066).

In fairness, this is not surprising because of: 1) the relative paucity of economic analysis of network effects prior to the completion of the CONTU report, and 2) the fact that the main sources of these network effects in the computer industry-microcomputers and mass-marketed software-had yet to come into prominence.

valuable, the larger the network of users). Network externalities can also arise from indirect effects—for example, the availability and quality of service can depend on the number of units that have been sold (e.g., well-established products or brands are more likely to have well-developed service departments). In some cases, network externalities occur for complementary products like computer hardware and software-the amount and variety of software available for use with a particular type of computer will increase as the computer itself becomes more popular.

The size of the network giving rise to these externalities depends on whether products produced by different firms can be used together.[87] Looking at markets where network externalities are present, Katz and Shapiro find that firms' choices whether or not to make their products compatible will be one of the most important dimensions of market performance. In many cases, firms will disagree on the desirability of making their products compatible: doing so may raise the profits of some firms while lowering the profits of others, even if total output increases. Katz and Shapiro also conclude that intellectual property laws help determine how compatibility is achieved—through joint adoption of an industry standard (e.g., when patents are strictly and broadly enforced) or through unilateral actions of adapters (e.g., when patents are loosely enforced or narrowly applied) .88

For example, in the absence of legal protection for user interfaces, a firm's private benefits from introducing a new interface may be short-lived, If it is a market success, other firms will be free to adopt it. When they do so, they will share in the network externalities (which are largely due to the originating firm's market success), With protection for user interfaces, a firm can introduce a new interface into the market and begin benefiting from network externalities, without allowing its rivals to participate in the new network.[89] Looking at user interface standards for PC application programs, Menell finds that these considerations might encourage firms with brand recognition to introduce proprietary (noncompatible) product standards, even though adopting compatible standard would increase net social welfare.[90]

Joseph Farrell finds that compatibility and standardization raise difficult economic and policy issues for intellectual property, particularly when network externalities are present. Looking at formal and informal standards-setting processes for computers and software, he concluded that:

> [I]ntellectual-property rules determine the boundaries of what is protected, and thus determine the borders at which fighting, or competition, takes place.[91]

Farrell considers that copyright law protects the useful and valuable ideas in computer software only indirectly, by protecting "ancillary features" (i.e., expression). Moreover, he argues, it can be economically inefficient to protect the latter—sometimes they should be imitated in order to take advantage of network externalities. Therefore, he concludes that a different mix of protection regimes, to protect the useful aspects of software but permit compatibility, would be more economically efficient.[92]

When compatibility is important, especially when network externalities are present, the *installed base* of products and/or training can affect the pace of innovation. Joseph Farrell and Garth Saloner find that when an installed base exists and transition to a new standard must be gradual, the installed base can sometimes inhibit adoption of the new standards by creating ''excess inertia. ''[93] In this situation, early adopters of the new standard will bear a disproportionate share of transition costs. As a result, when

---

87 For computer hardware-software markets, the issue depends on whether software produced for use with one brand of computer (or operating system) will run properly on another brand-if so, the brands can be said to "compatible."

88 Michael L. Katz and Carl Shapiro, ''Network Externalities, Competition and Compatibility,' *The American Economic Review,* vol. 75, No. 3, pp. 424-440.

89 Menell (May 1989), op, cit., footnote 13, pp. 1067-1068.

90 Ibid.

91 Joseph Farrell, "Standardization and Intellectual Property," *Jurimetrics Journal*, vol. 30, No. 1, fall 1989, pp. 35-50 (quote from p. 49). As compatibility becomes more important in an industry, Farrell sees formal standardization beginning to predominate, *even* though intellectual property protections may slow this process by increasing vested interests.

92 Ibid., pp. 49-50. Farrell suggests that (depend@ on court interpretation of laws) this mix might consist of a larger role for patent and less reliance on copyright to protect the most valuable aspects of software. See also Menell (July 1987), op. cit., footnote 15.

93 For discussion of the role of the installed base of hardware and software in earlier markets, see Fisher et al., op. cit., footnote 18, pp. 197-204.

important network externalities are present, the switch to a technologically superior new standard may be slower than socially desirable; even when all users intend to switch, they may prefer to wait for others to switch first.[94] But the opposite distortion is equally possible: when the new standard offers early adopters sufficient advantages over the old technology, they may be willing to switch long before a ''network' of users is well-established. As they switch, the new technology becomes increasingly attractive for later adopters, and ''excess momenturn' '-the inefficient adoption of the new technology—may occur, ''stranding' the old standard and hurting the old installed base's remaining users.[95]

In Farrell and Saloner's model, the presence of an installed base and network of users tied to the old technology creates a bias against a new (even superior) technology. Additionally, users who intend to switch may prefer not to be early adopters of the new technology. Farrell and Saloner show that incumbents can exploit these biases for anticompetitive purposes via *anticompetitive product preannouncements* and *predatory pricing.* Premature preannouncements of improved products using the old technology may prevent a new technology from gaining momentum. An incumbent firm may also be able to deter entry by a credible threat of price cuts in response to entry.[96]

The Economics of Copyright and Home Copying

Almost all of the literature discussed so far in this section has operated in a *business* context in examining linkages between intellectual property systems and technological advance, appropriability of rewards to creativity, etc. As we have seen, these linkages are complex. Therefore, the net effects on rights holders and on society from stronger/weaker, broader/narrower, longer/shorter protections are difficult to predict, let alone measure.

Similarly, the effects of noncommercial, private copying by consumers (which OTA calls ''home copying' are also complex and ambiguous. Economic theory suggests that the possible effects of home copying on resource allocation and on society's economic welfare will vary qualitatively, as well as quantitatively, depending on the economic and technological specifics of the industries and markets. Even for one specific type of home copying (e.g., home audiotaping), empirical effects are difficult to estimate with certainty: using survey data to estimate effects on industry revenues or consumer benefits necessarily involves many assumptions and approximations. Because many critical factors are difficult to measure and because choices among assumptions about underlying factors are subjective, even the same survey data can support disparate estimates of the economic effects of home copying. A 1989 OTA study, *Copyright and Home Copying: Technology Challenges the Law, examin*ed issues related to home copying in general, especially home video- and audiotaping. Chapter 7 of that report discussed the economics literature on home copying (i.e., of music and computer software) and reported on empirical analysis of home audiotaping based on an OTA survey of consumer taping practices (see box 6-B for a summary).

94 Joseph Farrell and Garth Saloner, "Installed Base and Compatibility: Innovation, Product Preannouncements, and Predation, ' *The American Economic Review,* vol. 76, No. 5, December 1986, pp. 940-955, Farrell and Saloner use as an example the persistence of the "QWERTY' keyboard in the face of the ergonomically more efficient Dvorak keyboard.

95 Ibid., pp. 941-942.

96 Ibid,, 954. Predation and anticompetitive preannouncements have been alleged in antitrust litigation, but are hard to prove (p. 942). Also, several economists maintain that truthful preannouncements arc procompetitive because they provide new information to the marketplace. (See Fisher et al., op. cit., footnote 18, p. 289.)

325-225 0 -- 92 - 14

Box 6-B—Literature on Home Copying

Prompted in large part by the copyright debates of the 1970s and 1980s concerning home audio- and videotaping, several economists have examined the economics of home copying: the effects of copying on the supply of copyrighted works (e.g., prices, quantities, variety), on consumer demand (e.g., whether copies substitute for purchases), and on society's net economic welfare (e.g., whether copying is economically inefficient, whether there would be an "excessive" variety of works absent copying). Because the economic effects of copying are complex and often ambiguous, each of these analyses relies on simplifying assumptions or specific conditions in order to reduce ambiguity. Therefore, the literature must be interpreted in light of the corresponding assumptions and conditions.

The Effect of Private **Copying on Economic Welfare**

Because intellectual property is a public good, ordinary market forces will not necessarily produce the most desirable social outcomes. Chanting a limited monopoly (e.g., via copyright) attempts to balance distortions arising from the partial inability of creators to exclude all nonpayers from obtaining their works. According to this rationale, the inability of creators to appropriate returns from their intellectual property would otherwise result in the underproduction of new works.

In the long run, the effect of unlimited private copying on society's economic welfare is ambiguous. As Johnson (1985) discusses, the long-run effect depends on a number of factors, including relative costs (of producing a home copy versus another "original"), the degree to which copying affects the demand for originals, the degree to which copying affects the production of new works, and the degree to which consumers value additional variety. The net social welfare effect of copying has two components: the effect on producers and the effect on consumers. Changing the amount of private copying (either increasing or restricting it) will affect not only the net level of society's economic welfare, but also the relative balance between producer and consumer welfares.

For example, Besen (1986) notes that copying can increase consumer welfare and producer profits in the short run-if private copying is efficient and the price of originals can be raised to reflect the value of the copies. On the other hand, copying may cause producers to reduce prices; this decreases both consumers' and producers' welfare. If, however, copying (by reducing the number of originals produced) reduces 'excessive' variety, this can increase welfare in the long run. This balance between gains and losses for producers and consumers is often the most visible and most hotly contested issue in debates over copyright scope and enforcement.

Will increased copyright protection for goods like musical recordings and software increase or decrease society's economic welfare? Overall, the economics literature indicates that the implications of increasing copyright protection are complex, and the policy tradeoffs are not simple. Some claim that stronger copyright protection will decrease the loss to society from the ***underproduction*** of works but will increase the loss to society from ***underutilization*** of these works. As Novos and Waldman show, the net result depends on the specifics of each situation. In some cases, market outcomes—where different classes of consumers are charged different prices of a good (e.g., individual and institutional subscription rates for journals) or where copyable and noncopyable goods (e.g., computers and software) are bundled-may be preferable to increased government enforcement, from an economic perspective. In some instances, as Katz notes, home copying might generate benefits from "network externalities" relating to the fact that consumers tend to value a hardware/software system more, the more popular that system and compatible ones are (a larger user base can increase the amount of information available about the system, enhance the image of a popular product, etc.)

Appropriability and Pricing

Private copying need not be harmful to producers, if copying is efficient and if producers can increase prices to take into account the value of the copies that will be made. If not all consumers copy, or if consumers vary in the number of copies each makes from an original, then efficient pricing would require discriminating among these groups, charging them different prices according to their valuations of the originals, based on their ability to make copies. This type of price discrimination is usually infeasible, however, because it is costly and difficult to gather the necessary information on users' valuations of originals and because resale is allowed. The inability to practice perfect price discrimination among users can produce imperfections in markets for intellectual property.

A simplified form of price discrimination is two-tiered pricing, in which producers are able to segment their customers into two classes and maximize profits by charging each a different price. Looking at the effect of photocopying on the number of scholarly journals purchased, Liebowitz has examined journal publishers' ability to indirectly appropriate copiers' true valuation of originals through higher subscription prices to libraries and institutions. He concluded that publishers can indirectly appropriate revenues from copiers who do not directly purchase journals. Since copying may have different effects on other media, however, case-by-case empirical investigation of the institutions and markets involved may be necessary.

Price Discrimination, Resource Allocation, and Variety

The inability to charge different classes of consumers different prices for a good in intellectual property markets means that the prices consumers pay need not reflect their actual valuations of the good: some value the good more, and will be willing to pay more. Those who do not value the good at a given price will not consume it. If they could be offered a lower price reflecting their valuation, however, then they would purchase it and both producers and consumers would be better off. Moreover, the decoupling of prices and valuations makes resource allocation—decisions about what to produce—more difficult and markets less efficient. Besen's analysis for the 1986 OTA report noted that where there are many producers of competing types of intellectual property, the resulting market structure is one of monopolistic competition: firms will have some control over the prices they can charge because their products are differentiated (e.g., music by different recording artists or groups). When firms are unable to charge different consumers different prices, however, there may be either excessive or insufficient variety. Under these conditions, when private copying serves to reduce the variety of products being offered, it does not necessarily reduce the efficiency of supply or make consumers worse off.

SOURCES: U.S. Congress, Office of Technology Assessment, *Copyright and Home Copying: Technology Challenges the Law, OTA-CIT-422 (Washington, DC:* U.S. Government Printing Office, October **1989),** ch. 7; William R. Johnson, "Estimating the Effect of Copying on the Demand for Original Creative Works," contractor report prepared for the Office of Technology Assessment (Springfield, VA: National Technical Information Service, October 1989); Michael L. Katz, "Home Copying and Its Economic Effects: An Approach for Analyzing the Home Copying Survey,' contractor report prepared for the Office of Technology Assessment (Springfield, VA: National Technical Information Service, October 1989); Ian E. Novos and Michael Waldman "The Emergence of Copying Technologies: What Have We Learned?" *contemporary Policy Issues,* vol. 5, July 1987, pp. 34-43; Stanley M. Be- "Economic Issues Relating to New Technologies and Intellectual Property," contractor report prepared for the Office of Technology Assessment by the Rand Corp. (Springfield, VA: National Technical Information Service, 1986); Stanley M. Be- "Private Copying, Reproduction Costs, and the Supply of Intellectual Property," *Information Economics and Policy, vol. 2, 1986,* pp. 5-22; William R. Johnson "The Economics of Copying," *Journal of Political Economy,* vol. 93, No. 11, 1985, pp. 158-174; S.J. Liebowitz, "Copying and Indirect Appropriability: Photocopying of Journals," *Journal of Political Economy,* vol. 93, No. 5, 1985, pp. 945-957; Ian E. Novos and MichaelWal&naq''The Effects of Increased Copyright Protection: An Analytic Approach" *Journal of Political Economy,* vol. 92, No. 2, April 1984, pp. 236-246; and Walter Y. Oi, "A Disneyland Dilemma: Two-Part Tariffs for a Mickey-Mouse Monopoly," *Quarterly Journal of Economics,* February 1971, pp. 77-94.

Appendixes

Appendix A
Selected Computer Hardware and Software Initiatives Overseas

Introduction

European producers have long faced competition from U.S. fins; they now face increased competition from Japanese firms positioning themselves in Europe in anticipation of the single European market. At the same time, the European software market is growing rapidly and packaged software+-long a U.S. strength-is becoming more popular. European research in computing is fragmented; market unification is expected to permit more integrated research in information and telecommunication technologies.[1]

Japanese firms are positioning themselves for rapid expansion in the United States, Europe, and Asia; Japanese computing research emphasizes massively parallel and distributed computing, optical computing, neural nets, and applications of fuzzy logic; software areas receiving attention include supercomputer software and graphical-display software for use in simulation and animation, The Japanese approach for "manufacturing' software has received much attention. Custom software currently dominates the Japanese market but some consider that Japanese "software factories' can extend Japan's advantage in quality "embedded" software to packaged software as well.[2]

Taiwan and Singapore are both developing information-technology industries, During the 1980s, a number of government measures facilitated development of Taiwan's microcomputer industry; the government is now focusing on nurturing a software industry. Over the past decade, Singapore has actively pursued a national goal of developing a software and services industry. It is now targeting strategic computing technologies like commercial applications of expert systems, neural nets, and fuzzy logic.

Europe

Computer Hardware

Europe makes up about one-third of the world computer market. In 1989, U.S. firms accounted for more than half of all computer sales in Europe and were even more dominant in the mainframe market; Japan held only about 10 percent of the computer market.[3] According to International Data Corp., the total European mainframe market was valued at $8,440 million in 1989 and $9,069 million in 1990. IBM, which earned 37 percent of its revenues in Europe in 1989, accounted for 52 percent of mainframe sales in 1989 and 61 percent in 1990.[4]

Computer hardware sales are slowing in Europe, due in part to economic conditions. Mainframe and minicomputer sales have slowed the most, despite price discounts, while microcomputer and workstation sales are more robust.[5] The latter reflects a trend in Europe toward smaller computers—annual growth in the microcomputer market is estimated at 25 percent for 1990 and 22 percent for 1991.[6]

During the downturn, European manufacturers like Olivetti and Groupe Bull have been more vulnerable than U.S. firms.[7] But now, in addition to slower sales, U.S. and European manufacturers are facing increasing competition from Japanese manufacturers, especially in the

---

1 Another expected outcome is more uniformity in European standards. See U.S. Congress, Office of Technology Assessment, Global *Standards: Building Blocks for the Future,* forthcoming 1992.

2 Sometimes a computer, its memory, and often, its programs, are embedded in another device such as an automobile engine, videocassette recorder, microwave oven, or television set. Such a computer is called an embedded computer and its programs are embedded applications programs. Some of the market estimates cited in this chapter may include the value of some embedded software but these were not reported separately.

3 Richard L. Hudson, ''Japanese Set To Do Battle in Europe's Computer Market,' *The Asian Wall Street Journal Weekly,* vol. XII, No. 34, Aug. 20, 1990, p. 1.

4 Data cited in Amiel Kornel, "Fujitsu Move Rattles Europe,' *Computerworld,* vol. 24, No. 32, Aug. 6, 1990, p. 1.

5 Before the slowdown, computer sales in Europe were growing at a rate almost twice that in the United States. The slowdown came after 5 years of heavy computer buying in Europe, reflected in 30-percent growth rates in computer sales in the late 1980s. Growth rates from Jonathan Levine et al., "Europe Ain't No Bonanza Anymore," *Business Week,* Aug. 6, 1990, pp. 26-28.

6 By comparison, growth in the U.S. microcomputer market is estimated at 10 to 15 percent in 1990 and 1991, compared to 50-Percent annual growth rates in the mid- 1980s. (Levine et al., op. cit., footnote 5, pp. 27-28.)

7 France's state-owned Groupe Bull's strategy to compete in the market of the 1990s depends in part on "open systems' based on AT&T Unix. Bull had revenues of $6.5 billion in 1989. Bull purchased Zenith Data Systems (laptop and personal computers) in 1988; Bull and NEC bought into Honeywell's computer business in 1987 and Bull took majority control of what is now Bull HN in 1988. Seventy percent of Bull's revenues come from outside France, including 33 percent from the rest of Europe and 30 percent from the United States. (Jonathan Levine and Gary McWilliams, ''Francis Lorentz' Scheme To Get Groupe Bull Charging," *Business Week,* July 16, 1990, pp. 154,156.)

mainframe market. Previously, m ffainframes made by Fujitsu, NEC, and Hitachi have been resold by European manufacturers like Siemens A.G. and Comparex.[8] Recently, however, Japanese firms have been establishing European bases of operation in anticipation of 1993's single market. For example, Fujitsu has acquired controlling interest in ICL (Great Britain's largest mainframe manufacturer)[9] and Mitsubishi has acquired Apricot Computers Ltd. (Great Britain's largest personal-computer manufacturer), now called ACT Group P. L.C.[10] Nevertheless, despite slower sales and increasing competition from Japan, the European computer market is still a lucrative and important one for U.S. manufacturers (see table A-l).

Table A-l —U.S. Computer Manufacturers' Domestic and European Sales (1989)

Company	Revenue ($ billions)	Percentage of sales: United States	Percentage of sales: Europe
IBM	$62.7	41 .0%	37.0%
DEC	12.7	46.0	40.0
Hewlett-Packard	11.9	46.7	34.7
Apple	5.3	64.4	22.9
Compaq	2.9	54.6	41.8
Sun	**1.8**	**57.9**	**24.2**
Tandem	**1.6**	**50.1**	**32.5**

SOURCE: 1989 data from Salomon Brothers, Inc. and company reports, cited in: Jonathan Levine et al., "Europe Ain't No Bonanza Anymore," *Business Week,* Aug. 6, 1990, p. 27.

Software

The software market in Europe, valued at $32 billion in 1990, is growing faster than the U.S. market and faster than the European hardware market.[11] At the same time, the composition of the market is shifting. Computer software-and-services companies, producing custom (''bespoke' software tailored to clients' operations and needs, are facing growing competition from packaged software. Packaged software's share of the market is increasing, accounting for 29 percent of software sales in 1989, up from 11 percent in 1979.[12]

According to the market research organization Ovum, the top 40 packaged software vendors in Europe (including 21 U. S., 6 French, 5 German, and 4 British companies) accounted for 64 percent of packaged software sales valued at $11 billion in 1989.13 Most of the major European packaged software vendors have been either large hardware manufacturers or systems houses focusing on sales of hardware or computer services, rather than specializing in packaged software (see table 3-3 inch. 3). With sales of both system and applications software, the largest vendor of packaged software overall in Europe is IBM, with 1989 sales worth $2.12 billion. Almost all of the major packaged software vendors that specialize in software are U.S. firms like Microsoft or Computer Associates International.[14]

Research and Technology Initiatives

Historically, research in Europe has been fragmented by company and country. Beginning in 1993, market unification in the European Community (EC)[15] will permit more integration of research in member countries, with the intention of strengthening the technological base of industry in the EC and improving EC industry's global competitiveness. The Research and Development (R&D) Title to the EC Treaty provides a firmer legal basis for cooperation in R&D and calls for the EC to adopt a multiyear framework laying out all its R&D activities. In 1987, the EC adopted the "New Framework Program," after debate concerning content of, and funding for, cooperative R&D, and the relative merits of coordinated research versus competition in stimulating commercially productive innovations.[16]

There have been three Framework Programs, providing R&D funding in overlapping 4-year periods: Framework

---

8 Kornel, op. cit., footnote 4.

9 Levine et al., op. cit., footnote 5, and David E. Sanger, "Fujitsu To Buy ICL Stake, " *The New York Times,* July 31, 1990, pp. D1, D6.

10 Richard L. Hudson, ''Japanese Set To DO Battle in Europe's Computer Market,'' The Asian *Wall Street Journal Weekly,* Aug. 20, 1990, pp. 1, 6.

11 *Estimate from:* 'European Software Industry 1992 Market Changes Response Profile (Executive Summary),' prepared by Ernst& Young on behalf of the Scottish Development Agency (Boston, MA: Clarke & Co., 1990).

12 ''Salvation in Services?'' *The Economist,* Aug. 11, 1990, pp. 70-71.

13 Ralph Bancroft, "Europe Struggling in Software, " *Computerworld,* July 23, 1990, p. 97.

14 Ibid.

15 For a summary of the impacts of the EC92 initiative on science and technology issues, including industry R&D and standards, see Glenn J. McLaughlin (coordinator), "The Europe 1992 Plan: Science and Technology Issues, " Congressional Research Service Report for Congress, revised Aug. 23, 1989.

16 Discussion from Candice Stevens, " 1992: The European Technology ChaUenge,' *Research & Technology Management, vol. 33, No.* 1, January-February 1990, pp. 17-23.

For more discussion on science and technology programs in Western Europe, see Congressional Research Service, Library of Congress, *Transfer of Technology From Publicly Funded Research Institutions to the Private Sector,* prepared for the Subcommittee on Oversight and Investigations of the Committee on Energy and Commerce, U.S. House of Representatives, Committee Print 102-G, 102d Congress, 1st Sess., July 1991, pp. 51-68.

One (1987-91), Framework Two (1988-92), and Framework Three (1990-94). This is intended to allow continuation of current R&D programs while providing a way to reevaluate them. The Framework Programs provide support for many fields of science and technology; support for R&D in information technology and communication technology is provided through the ESPRIT and RACE programs, respectively .17 Phase I of ESPRIT began in 1984, before the Single European Act, and continued through 1988; it addressed basic technologies and standards in microelectronics and software, as well as computer applications like computer-integrated manufacturing and office systems. Phase II of ESPRIT (1988-92) continues the initiatives begun in phase I and also includes new basic research projects; currently about 3,000 researchers are involved in 200 ESPRIT projects.[18] ESPRIT's software accomplishments to date include development of software for optimizing designs of application-specific integrated circuits (ASICS) and software-development and software-integration tools developed under the Portable Common Tool Environment project.[19] The EC contribution for ESPRIT II during 1988-92 is about 1,600 million European Currency Units (ECUs), nearly $2 billion.[20]

The RACE program includes research and standards projects intended to move the EC toward combined communications (voice, data, video, electronic mail) based on integrated services digital network (ISDN) standards. The EC contribution for RACE during 1987-92 is 550 million ECUs, or $660 million.[21]

Japan

Computer Hardware

Beginning in the mid- 1950s, the Japanese Government, which recognized the strategic importance of the computer industry, used capital, subsidized R&D, and protective regulations and procurement policies to nurture a developing Japanese computer hardware industry.[22] By 1990, these policies had paid off in an industry with compelling strengths in memory chips, flat-panel displays, and optical disks. In 1980, the Japanese industry had a 10 percent share of the world computer market; a decade later, Japan's share of the world computer market was expected to exceed 40 percent by 1992.[23] Japan already dominates the laptop and notebook computer markets.

In 1990, IBM held almost 52 percent of the world mainframe market, followed by Fujitsu (9.5 percent), Hitachi (6.8 percent), Unisys **(6.6** percent), Amdahl (4.6 percent), Siemens (2.8 percent), and NEC (2.7 percent) .24 Japanese mainframe computer manufacturers have been positioning themselves via product development, marketing strategies, and acquisitions for rapid expansion in the U. S., Asian, and European markets now dominated by IBM.[25] For example, Fujitsu has acquired **ICI P. L.C.,** Britain's largest mainframe computer manufacturer[26] and the world's ninth-largest producer of mainframe computers. Fujitsu acquired an 80 percent share of ICL for about $1.4 billion and thereby became the world's second-largest computer manufacturer (in terms of sales), behind

17 ESPRIT is the acronym for European Strategic Programme for Research and Development in Information Technologies; RACE is the acronym for R&D in Advanced Communications Technologies in Europe. See Congressional Research Service (July 1991), op. cit., footnote 16, pp. 56-58.

18 Ibid., pp. 57-58.

19 Gadi Kaplan and Alfred Rosenblatt (@.), "The Expanding World of R& D,' *IEEE Spectrum, October* 1990, *pp.* 28-33.

20 Congressional *Research* Service (July 1991), op. cit., footnote 16, pp. 57-58.

21 Ibid., pp. 59-60.

22 Marie Anchordoguy, "How Japan Built a Computer Industry, " in Charles H. Ferguson, "Computers and the Coming of the U.S. Keiretsu, " *Harvard Business Review,* July-August 1990, p. 65.

Between 1965 and 1985, Japanese R&D expenditures as a percentage of gross national product (GNP) nearly doubled, from 1.27 percent of GNP in 1965 to 2.53 percent in 1985. Over the same period, the Japanese Government's share of R&D expenditures fell from 31 to 19 percent; although totat government funding increased, industry R&D grew more rapidly. (National Science Foundation and Japan Science and Technology Agency data cited in: Leonard H. Lynn, "Technology Policy in Japan," *Forum for Applied Research and Public Policy,* fall 1990, pp. 57-61.)

For a discussion of Japan's basic research initiatives, see "CanJapan Make Einsteins Too?" *The Economist,* Aug. 11, 1990, pp. 81-83. Japan's R&D expenditures now total over 2.9 percent of GNP, with government expenditures amounting to 20 percent of this sum and expected to double during the 1990s (ibid., p. 81).

23 "Computers and Other Targets: How Japan Learns, and Wins, Even by Losing," *The New York Times,* May 11, 1990, p. A32 (editorial).

24 Nikkei, IDC data cited in Jean S. Bozman and Lori Valigra, "Long-Term Global Strategies Unfold," *Computerworld, Nov. 26,* 1990, p. 101.

25 For example, in 1989 Hitachi acquired 80 percent of National Advanced Systems, now called Hitachi Data Systems (Electronic Data Systems owns the other *20* percent), and Fujitsu has owned 43 percent of Amdahl since 1984. See Robert D. Hof and John W. Verity, "The Japanese Threat in Mainframes Has Finally Arrived, " *Business Week,* Apr. 9, 1990, p. 24; Jean S. Bozman and Lori Valigra, "Long-Term Global Strategies Unfold, " *Computerworld, Nov. 26,* 1990, pp. 1, 101.

26 Jeff Shear, "Japan Upsets Computer Applecart," *Insight,* Oct. 22, 1990, pp. 44-45. Fujitsu was already manufacturing ICL hardware.

IBM and **just ahead** of Digital Equipment Corp.[27] By the time that Fujitsu acquired ICL, the British firm had grown dependent on Fujitsu for crucial technologies (e.g., computer chips, circuit boards, cooling systems) .28 Fujitsu's acquisition of ICL was of special concern to the European information-technology community because ICL was a partner in about 40 hardware and software projects being conducted under the European ESPRIT prograrn.[29] Fujitsu also has a majority interest in a mainframe computer factory in Spain and an agreement to sell powerful mainframes through Siemens A.G. Coupled with concentration in the European computer industry due to financial pressures, such acquisitions position Japanese firms like Fujitsu or Mitsubishi Electric (which has acquired the manufacturing facilities of Apricot, a British PC company) to participate in the EC's single market after 1992.[30]

At the same time, Japanese investments in computer and software firms in the United States have continued to grow through minority equity positions as well as acquisitions. In 1990, Mitsui & Co. purchased a minority interest in Unisys with $150 million of preferred stock that can be converted into 4.6 percent of Unisys' common stock outstanding. Unisys also arranged a $50-million, 5-year loan from Mitsui.[31] Also in 1990, Mitsubishi Kasei Corp. acquired Verbatim Corp. for $200 million.[32] Canon has invested $100 million in NeXT Computer (multimedia computing), Canon Sales Co. has invested in FPS Computing Inc. (64-bit computers), and Fujitsu has purchased 46 percent of Poqet Computer (notebook computers) .33

Computing Research Focus

Japan's highly publicized Fifth-Generation project (1982-91) was intended to create "intelligent" machines that could support expert systems to emulate human reasoning and could communicate with humans through natural (rather than computer programming) languages. MITI initiated the Fifth-Generation project in 1982 and anticipated three phases: 1) study of existing knowledge in logic processing and parallel processing and development of prototype hardware and software systems; 2) construction of small-scale subsystems for logic processing and parallel computation; and 3) completion of a full-scale prototype computer implementing inference and knowledge-based functions in hardware (for speed) and using software for knowledge-base acquisition and management, natural-language interfaces, and "intelligent" (user-friendly) programming tools.[34] Today's Japanese artificial intelligence (AD) market reflects industry's commitment to increase the use of AI technology, especially expert systems in the financial and manufacturing industries, in support of the Fifth-Generation project.[35]

Although Japan, along with other countries, made progress in AI research, the Fifth-Generation project fell far short of its original goals. However, the project has had two important consequences. First, it stimulated research in the United States and Europe, as well as Japan, and enabled Japan to build up a basic-research infrastructure in computing, training, and influencing thousands of Japanese computer professionals. Second, it gave credence to, and focused Japanese Government and industry attention on, parallel processing.[36] Emphasis on massively parallel computing, part of what is referred to as the

27"Fuji@u's Purchase of ICL Illustrates Its Aggressive Pursuit of Europe Market,'' *The Asian Wall Street Journal* Weekly, Aug. 6, 1990, p. 5.

28 Sanger (July 1990), op. *Cit.,* fo'emote 9.

29 Kaplan and Rosenblatt (eds.), op. cit., footnote 19, pp. 28-33.

30 Singer (July 1990), Op. cit., footnote 9.

31 Paul B. Carroll, ''Unisys, s@=@ With Big Debt Load, Sells Preferred Stock to Japanese Firm,'' The *Wall Street Journal,* June 27, 1990. Mitsui and Unisys have had long-standing business relations: each owns one-third of Nihon Unisys Ltd., a computer marketer that does $2 billion of annual business in Japan, and Sperry (which merged with Burroughs to form Unisys in 1986) began working with Mitsui in the 1950s.

32 Michael R. Sesit, ''Japanese Are More Willing To Buy Minority Stakes in U.S. Companies, " *The Asian Wall Street Journal Weekly,* Aug. 13,1990, p. 21.

33 NeoConcepts da@ reported in: Sheridan Tatsuno, *''U.S.* Threatened by Rash of High-Tech Buyouts," New *Technology Week,* Aug. 6, 1990, p. 7. Fujitsu also owns 44 percent of the Amdahl Corp. (mainframes). (Amiel Kernel, op. cit., footnote 4.)

34 See Michael A. Cusumano, *Japan' s Software Factories: A Challenge to Japanese Management (New York, NY: Oxford* University Press, 1991), pp. 410-417. In addition to describing the Fifth-Generation project and its outcomes, Cusumano examines Japanese approaches to software development in detail.

35 *An Assessment of the Japanese Artificial Intelligence Market,* U.S. Department of Commerce, International Trade Administration (Springfield, VA: National Technical Information Service, May 1989), p. xvii. According to ITA, most Japanese expert system tools used to build expert systems are less expensive and less complex than U.S. counterparts; Japanese users prefer easy-to-use products with Japanese language capability.

The 1990 market for AI hardware and software was about $2.5 billion but the market is expected to reach $10.6 billion by 1995 (MITI estimates, ibid., p. vii). According to ITA, most Japanese AI software runs only on the hardware for which it was develop@ the eight largest Japanese computer firms (Hitachi, Fujitsu, Toshiba, NEC, Oki, Sharp, Mitsubisbi and Matsushita) control 60 percent of the total Japanese AI hardware and software market (ibid,, pp. xix, 42).

36 see Cusumano, op. cit., foomote 34, and David E. Sanger, "Japan Is Planning To Challenge U.S. in Computer Field," *The New York Times,* Apr. 30, 1990, pp. Al, D14. The Japanese Government provided about $250 million in funding from 1981 to 1990 and is expected to spend another $40 to $50 million before the project ends in 1992.

Sixth-Generation project, is a change in direction from earlier Japanese efforts to produce very fast supercomputers that relied on speed (rather than parallelism) for computational power.

Parallel processing is thought to hold great promise for applications like picture, sound (voice), and character recognition. To this end, advances in parallel and fault-tolerant architectures, as well as in chip technology, will be required. But, interestingly, the hardware to build massively parallel machines is simpler to design and develop than is the complex software required to coordinate parallel processing and efficiently exploit the capabilities of the hardware. Therefore, MITI has convinced government officials and industry to devote resources to a large effort to develop parallel-processing software.[37]

In April 1990 MITI published a broad outline of how to proceed from the Fifth-Generation project; one of the objectives outlined was a concerted effort to master massively parallel processing. MITI's plans for this New Information Processing Technologies (NIPT) program called for leapfrogging the evolutionary technology-development paths that most U.S. firms and research programs are following, concentrating instead on massively parallel and distributed computing systems, neural nets,[38] optical computing, and applications of fuzzy logic.[39] By spring 1991, the focus of the program appeared to be shifting to a more interdisciplinary, basic research focus that could eventually lead to development of massively parallel, distributed processing systems based on optoelectronics.[40]

Software

According to a 1990 American Electronics Association report, the software market in Japan is only about one-third the size of the U.S. market. However, it is growing much more rapidly-in part, because it is less mature. The Japanese market (for packaged and custom software), estimated at about $18 billion in 1990, is expected to grow to some $33 billion (about 20 percent of the worldwide market for software) by 1995.[41] (In 1986, the Japanese software market was estimated at only about $5.4 billion.[42])

The composition of the Japanese market differs significantly from its United States and European counterparts in that custom software (rather than packaged software) accounts for 80 percent of software sales. By contrast, packaged software is more common in Europe and predominates in the United States, accounting for 75 percent of the U.S. market, according to the American Electronics Association.[43] The prevalence of custom software in Japan **accounts** for the situation that, although imported software accounted for about half of packaged software sales in Japan in 1988, imports amounted to less than 10 percent of the total market.[44]

Much attention has been focused on the Japanese approach to software development through ''information systems" factories bringing manufacturing-style production and quality controls to software development. U.S. observers have raised concern that this approach will extend Japanese productivity and success in embedded software (in electronics) and in custom programming to the packaged software market worldwide.[45]

---

37 Sanger (April 1990), op. cit., footnote 36.

38 For example, Toshiba Corp. is developing a character-recognition device that uses a neural network to identify both katakana (a Japanese phonetic alphabet) and numerical characters. The system, which will be marketed in 5 years for applications in automatic-teller machines and optical character readers, reportedly has a 95-percent accuracy rate. (''Neural Network Device Can Read Handwriting, ' *The Asian Wall Street Journal Weekly,* Feb. 11, 1991, p. 8.)

39 Richard McCormack, ''Ameri~ Scramble To Figure Out Japan's Next Thrust in Computing, " New *Technology Week,* vol. *4, No. 48,* Dec. *3,* 1990, pp. 1-2.

Fuzzy logic can be used to reduce superfluous software coding in massively parallel computers. A fuzzy-logic chip can be used to reduce the number of conventional logic circuits needed, making it possible to install simpler controllers that require less-complex software to run them.

40 MITI officials seeking U.S. Government support for participation in NIPT by U.S. industry and academic researchers were describing the program as focused on basic research and development, with a special focus on the brain and neural research. (Richard McCormack, ''U.S. Chilly to Japan on 6th Generation Initiative," New *Technology Week,* vol. 5, No. 22, May 28, 1991, pp. 1-4.)

41 Japan Information Service Industry Association estimates cited in: American Electronics Association Industry Committee in Japan, *Soft Landing in Japan,* (Tokyo, Japan: American Electronics Association Japan Office, June 1990), p. 7. *OTA note:* These *figures may* include some ''services' as part of custom software revenues.

42 ADAPSO (The Computer Software and Services Industry Association) estimate in: Jeff Shear, ''Competitive Software Industry Suits Up for Global Hardball," *Insight,* July 10, 1989, p. 39.

43 American Electronics Association, op. cit., footnote 41, pp. 7-8. *OTA note:* These figures may include some "services' as part of custom software.

44 *Japanese Software: The Next Competitive Challenge,* prepared by Dewy, Ballantine, Bushby, Palmer & Wood for ADAPSO's International Marketing Seminar (Arlington, VA: ADAPSO, January 1989), p. vi.

45 See, e.g., Cusumano, op. cit., footnote 34, and Neil Gross, "Now Software Isn't Safe From Japan,'' *Business Week,* Feb. 11, 1991, p. 84; Shlomo Maital, ''Why Not Software Factories,' *Across the Board,* October 1990, pp. 5-6; and Jacob M. Schlesinger, ''Japanese Concept of Software Factory Could Reshape Industry World-Wide,' *The Asian Wall Street Journal Weekly,* Feb. 11, 1991, p. 4.

Moreover, in mid-1991 MITI announced the formation of a new R&D committee to study numerical simulation using supercomputers and high-end workstations. The purpose of this new MITI group, with participants from academia and 20 companies including Fujitsu, Hitachi, IBM Japan, and NEC, is to put together experimental and theoretical underpinninggs for developing advanced supercomputer hardware and software for use in simulation and animation. In concert with other Japanese technology initiatives, such as the Sixth-Generation project and MITIs' new international project on next-generation structural models for large computer systems, the supercomputer simulation initiative may give Japan strong capabilities in supercomputer software, as well as hardware.[46] However, the United States still leads in supercomputers, as well as high-speed networks,[47]

Rapid growth in the Japanese market has led more than 60 U.S. software companies, including Lotus Development Corp., Microsoft, Computer Associates International, and Adobe Systems Inc., to establish subsidiaries or offices in Japan.[48] At the same time, Japanese companies are establishing themselves in the United States. For example, last fall an engineering group in Sony Corp. 's U.S. unit began work developing a new version of Unix as a basis for developing Unix application programs. Sony is one of the first Japanese companies to develop software applications (in this case, with a staff consisting mainly of U.S. software engineers) in the United States.[49] Unix microcomputer and workstation software is expected to grow in importance in Japan during the 1990s.[50]

Taiwan

In Taiwan, government measures, including research funding and tax advantages, spurred development of Taiwan's computer-hardware industries during the 1980s. At present, Taiwan has relatively strong microcomputer and semiconductor chip industries made up of domestic manufacturers as well as original equipment manufacturers (OEMS) from the United States and Japan.[51] In 1990, revenues for these industries amounted to $1,566 million for microcomputers and $450 million for semiconductor chips.[52]

For the long term, however, Taiwan faces a transition from a manufacturing to a service and knowledge-based economy.[53] Accordingly, interest in the *domestic and* global software markets has increased sharply, and industry leaders called for the government to institute policies and programs to assist the software industry, as it did for the hardware industry. To this end, the government of Taiwan has provided research funding and, in some cases, investment and tax incentives for software developers.[54] This software activity takes place in a larger context of increasing total R&D expenditures as a percentage of gross national product (GNP), increasing business expenditures for R&D; and emphasis on target industries including microelectronics, computers, computer peripherals, materials, automation and robotics, as well as software and information science.[55]

By the mid-1980s, according to the U.S. Department of Commerce, such government efforts had given rise to over 100 small software houses in Taiwan, most with fewer than 20 employees. These software houses were

---

46 Sheridan Tatsuno, "The Latest MITI Thrust: Supercomputer Simulation Hardware and Software," New *Technology Week,* June 10, 1991, p. 3.

47 See U.S. Congress, Office of Technology Assessment, *Seeking Solutions: High-Performance computing for Science*, OTA-BP-TCT-77 (Washington DC: U.S. Government PrintingOffice, April 1991); and U.S. Congress, Office of Technology Assessment, *Networking the Nation: The National Research and Education Network*, forthcoming 1992.

48 Gross and Schwartz, op. cit., pp. 56-57.

For case studies of six U.S. software firms in Japan (Autodesk, Lotus, Comshare, Ingres, Oracle, and SDC), see American Electronics Association, op. cit., footnote 41, pp. 77-89.

49 "Engineering Group Set Up To Work on UNIX Software," *The Asian Wall Street Journal Weekly*, Aug. 20, 1990.

50 American Electronics Association, op. cit., footnote 41, pp. 10-11. IDC data cited by the American Electronics Association indicate thatwhile MS/DOS *will* continue to dominate microcomputer operating systems in the early 1990s, use of Unix systems is expected to grow at a compound annual growth rate of 140 percent during this period, compared to 25 percent annually for MS/DOS.

51 Paul C.B. Liu, ''Computer Software and Intellectual property Law in the Pacific Rim counties, ' contractor report prepared for the Office of Technology Assessment, March 1991, p. 4.

52 Jullian Wu, Market Intelligence Center, Institute fo, Information Industry, personal communication (letter), June 13, 1991.

53 According t. Taiwan's Council of Economic Plarming and Development, 50 percent of Taiwan's GNP will come from services by the year 2000. Chris Brown, "Taiwan Software Firms Ponder World Market," *Computerworld*, vol. 24, No. 29, July 16, 1990, pp. 110-111.

54 According t. the Institute for Information Industry's Market Intelligence Center, conditions under which software firms may receive investment and tax incentives are strict; from 1985 to 1989 only 15 software companies qualified for the tax incentive. (Jullian Wu, Market Intelligence Center, Institute for Information Industry, personal communication (letter), June 14, 1991.)

55 Dennis Fred Simon, ''Technology Policy on the Pacific Rim,'' *Forum for Applied Research and Public Policy*, fall 1990, pp. 67-72. In 1978, total R&D expenditures in Taiwan amounted to 0.5 percent of GNP ($1 11 million); by 1987, R&D spending had increased to 1.2 percent of GNP; Taiwan's 10-Year Science and Technology Development Plan projects R&D expenditures of 2 percent of GNP by 1995.

primarily developing application-software packages; by contrast, systems-software development in Taiwan was mostly being done in publicly supported research institutes like the Institute for Information Industry (III). Tax incentives and the availability of skilled, lower-cost labor led major U.S. corporations like Hewlett Packard, IBM, and Wang to establish software-development groups in Taiwan .56 Computerization of whole industries in Taiwan contributed to the rapid growth of Taiwan's software industry. By 1990, it had grown to about 300 software firms; 1990 revenues from sales of Taiwanese application-software packages amounted to $149.5 million.[57]

The government-sponsored III has been charged with promoting progress in the software industry through a variety of technical and institutional means. These include development of Chinese versions of Unix and the X Windows interface for use in workstations[58] and educational efforts to promote intellectual property concepts to the public,[59] as well as market-intelligence, economic, and legal research for the industry.[60] III has also signed a cooperation agreement with the Justice Yuan for information exchanges and joint research in computer law.[61]

Singapore

Over the past decade, Singapore has actively pursued a national goal of developing a vital software and services industry. This has largely been done under the auspices of the National Computer Board (NCB), established in 1981.[62] According to the U.S. Department of Commerce, in 1982 the government of Singapore initiated a 5-year, *$80-million* program under the direction of NCB to train computer specialists and provide financial incentives for local software development.[63] Three government-run training institutes were established under this program; of the three, one was a joint venture with IBM, and another was provided with significant funding and equipment from the Nippon Electric Co. (NEC) of Japan.[64] By late 1984, Singapore's financial incentive packages resulted in the establishment of software centers by Nixdorf, the Digital Equipment Corp., Hewlett Packard, and Sperry .65

In 1983, the revenues for Singapore's software industry amounted to $24 million; about $4 million came from exports.[66] AS of 1990, according to the U.S. International Trade Administration (ITA), software and services were a billion-dollar industry in Singapore.[67]

According to the ITA, having accomplished its initial objectives, the NCB is now *using* its applied-research arm, the Information Technology Institute (ITI) to strategically target R&D and commercial uses of emerging technologies like AI and fuzzy logic.[68] At the Nanyang Technical Institute, collaborative R&D by government and industry targets computer integrated manufacturing,

---

56 U.S. Department of Commerce, office of Computers and Business Equipment/Science and Electronics, *A Competitive Assessment of the United States Software Industry* (Washington, DC: U.S. Government Printing Office, December 1984), p. D-2.

57 Jullian WU, Market Intelligence Center, Institute for Information Industry, personal communication (letters), June 13 and 14, 1991.

58 Chris Brown, op. cit., footnote 53.

59 Carol K.N. Chang, Planning Engineer, Market Intelligence center, Institute for Information Industry, personal communication (meeting), Nov. 9, 1990.

60 Jullian S.L. Wu, Project Manager, and Josephine L.L. Houng, Legal Research Fellow, Market Intelligence Center, Institute fOr Information Industry, personat communication (meeting), Apr. 23, 1991.

61 Under the terns of the agreement, III will provide the Yuan with computer law and market information and the Yuan will provide the III with court decisions related to intellectual property. The Justice Yuan and III held their first legal research meeting in June 1991. Jullian Wu, op. cit., footnote 57, June 1991.

62 The National Computer Board was established with three principal objectives: 1) to computerize the gov ernmental services and departments, 2) to train software professionals in sufficient numbers to meet Singapore's needs, and 3) to develop the software and services industry. For more information see Victoria Kader, Office of Computers and Business Equipment, International Trade Administration, "Singapore Moves Into the Advanced Information Age," *Business America,* Aug. 13, 1990, p. 9.

63 See U.S. Department of Commerce, op. cit., footnote 56, p. D-1.

Government assistance for Singapore software development gave rise to controversy in 1989-90, when the U.S. Department of Commerce investigated whether a Singapore firm's software product (being marketed in the United States) was an "unfair" competitor in that the Singapore Government had subsidized the development of a commercial software product. In March 1990 the Department of Commerce's ITA reversed its original ruling that the product, a computer-aided software engineering tool, had been subsidized. The ITA ruled that software on a disk or tape, including software on a master disk, can be subject to import duties. (David A. Ludlum, "Commerce Department Revokes Singapore Ruling," *Computerworld, vol. 24, No. 14,* Apr. 2, 1990, p. 119.)

64 Department of Commerce, op. cit., footnote 56, pp. D-1, D-2.

65 Ibid.

66 Department of Commerce, op. cit., footnote 56, p. D-2.

67 Kader, op. cit., footnote 62, p. 9.

68 Ibid. According to ITA, NCB accomplished its goal Of@aining 8,000 new software professionals (Singapore started with 1,800 in 1982) in 1990, 2 years ahead of schedule. ITA also reports that computerization in the government resulted in cost savings amounting to 1.7 times the investment in information technology.

often using U.S. expert system development tools.[69] With industry partnership, the Institute of Systems Science (ISI) at the National Institute of Singapore is working on AI and fuzzy-logic applications, including a neural network for foreign-exchange trading and a logistics system incorporating fuzzy-logic principles for the Singapore seaport.[70]

@Ibid., p. 9.

70 Ibid., p. 10. According to ITA, the 1S1 is active in training programs, such as a degree program in knowledge engineering, that are tailored to meet the needs of industry. The NCB also finances training programs in information technology and provides assistance programs for computerization of company business operations.

Appendix B
The European Economic Community Treaty: Structure and Function

The European Economic Community Treaty determine the institutions of the European Community (EC) and their basic functions. These institutions have broad legislative, judicial, and administrative powers, which enable them to render direct and binding orders to member states and to their nationals. This extensive delegation of duties in the treaty is founded on the belief that the future economic development of the EC would introduce changed circumstances that could not be anticipated in the treaty, and also that the creation of a strong central organization would lay the groundwork for future political unity. The powers of these EC institutions-the European Parliament, the Council, and the Commission-are allocated to maintain balanced distribution of powers in the European Community.

The Parliament

Composition of the Parliament

The Parliament consists of representatives of the nationals of the countries of the European Community. The function of the Parliament is to exercise the advisory and supervisory powers conferred upon it by the European Economic Community Treaty. The European Parliament officially came into existence in March of 1958. Its members are appointed by and from among the members of the national parliaments of the member states. The number of delegates from each country is as follows.

France	81
Germany	81
Italy	81
United Kingdom	81
Spain	60
The Netherlands	25
Belgium	24
Greece	24
Portugal	24
Denmark	16
Ireland	15
Luxembourg	6

Parliament members sit by party rather than by nation. Political groups act according to political perspective rather than on a national basis. As of January 1, 1986, groups constituting the Parliament were, in order of size, Socialists, European People's Party, European Democratic Group, Communists and Allies, Liberal, Democratic and Reformist Group, European Renewal and Democratic Alliance, Rainbow Group, and European Right.

The members of the European Parliament are thus representatives of their peoples, rather than representatives of their governments. They may not receive instructions from the national parliaments, political parties, or interest groups, and they are expected to act in the interest of the EC as a whole.

Functions of the Parliament

The European Parliament functions in an advisory and supervisory capacity.

The Parliament's advisory power is a means of involving the peoples of the EC in the drafting of EC legal measures, and allows the Parliament to play a part in the legislative process of the community. The treaty provides for consultation of the Parliament by the Council as an essential formality before the Council makes a final decision as to certain treaty matters. Disregard of this formality renders the measure void. Simply presenting a proposal accompanied by a request for an opinion of the Parliament is not sufficient; if the opinion is never actually rendered by the Parliament, the measure is void.

The treaty provides for the Parliament's supervisory functions through which it can force the Commission to resign by a motion of censure. The Parliament and its members may question the Council and the Commission to obtain the information necessary to make such a decision.

Parliament has also sought a greater voice in the negotiation and conclusions of international agreements to which the EC is to be party, and of treaties on the accession of new member states. To that end, a cooperation agreement called the Single European Act (SEA) was adopted. This agreement provides that the Council's common position, reached by a qualified majority, must be communicated to the Parliament accompanied by an explanation of the Council's reasons for taking that position. The Parliament must make a decision on the Council's position in 3 months; its failure to do so allows the Council to pass the measure. Should the Parliament propose amendments or reject the Council's position by a majority of its membership, the Council can overrule the Parliament's amendments or reject them only by a unanimous vote.

All members of the Commission and Council may attend sessions of Parliament. Those speaking in the name of the Commission or the Council must be heard at these sessions.

The Council

The European Council strengthens the decisionmaking capacity of the EC since many of the Council's decisions can be adopted by majority, rather than unanimous, vote.

Composition of the Council

Representatives of **member states make up the Council of the European Communities. The members of the Council represent the member states according to instructions of the individual State. The members also area part of an EC institution and can take decisions that may beat variance with the instructions of their governments.** Such variance does not in any way affect the validity of a Council decision. As the council acts as a federal institution, its decisions are valid notwithstanding any deviation from instructions.

Representatives of the member states are required to be members of the government of the member state they represent. Member states may decide which member of the government it wishes to appoint, and different members of the government maybe delegated, depending on the matter before the Council.

Functions of the Council

The role of the Council is to ensure the coordination of the economic policies of the member states. Under the treaty, the Council has the power to take decision, or adopt legislation proposed by the Commission. The treaty also requires the Council to confer on the Commission certain powers for the implementation of the rules the Council has adopted, and gives the Council the power to exercise implementing power itself in certain circumstances.

Coordination of Economic Policies—It is the particular task of the Council to coordinate the "general economic policies' of the member states. Thus, general economic policy in principle remains within the competence of the individual member states except to the extent that the treaty confers powers upon the EC. Coordination is to be achieved through consultations in the Council and through the recommendations made by it. The Council's powers to enforce the coordination of the economic policies reach no further than this, except insofar as it has expressly been given such powers by specific treaty provisions.

Power To Take Decisions—The power to take decisions or adopt legislation encompasses all acts that are to be undertaken under the provisions of the treaty—for example, on the budget of the EC, on the bringing of suit, or on the appointment of officials. The power to take decisions is limited in that most decisions require a prior proposal of the Commission and often the consultation of the Parliament or of the Economic and Social Committee.

Implementation of Council Acts—The Commission has the power to implement rules the Council has made. This power may be curtailed by certain requirements relating to the manner in which it is exercised. The Council may also reserve the right to directly exercise implementing powers itself. These procedures must be in accordance with rules and principles the Council will specify.

The Commission

Functions of the Commission

The Commission consists of 17 independent members. Its primary function is to see that the EEC treaty provisions are executed. The Commission also monitors the application of legislation promulgated by the institutions for purposes of implementing the treaty.

The Commission supervises the orderly functioning and development of the Common Market. This includes the removal of barriers between economies of Member States, the smooth conduct of economic life within the area in which a single market is to be established, and increasing economic activity and the standard of living. Member States have a duty to facilitate the achievement of the Commission's tasks, i.e., to ensure that the provisions of the treaty and the implementing measures adopted by the institutions are applied. Member States have a duty to consult the Commission and they must keep the Commission informed of measures they have taken in fulfillment of their obligations to implement EC rules.

The Commission may formulate recommendations or opinions on all matters covered by the treaty. This includes all matters that are dealt with in the treaty or subjects having a direct connction with treaty rules. While in some cases the treaty expressly provides for issuance of recommendations or opinions, the Commission is not limited to the provisions of the treaty and may issue recommendations or opinions any time it considers necessary. Recommendations and opinions may be addressed to anyone, and are not binding.

Finally, the Commission exercises powers conferred by the Council for the implementation of the treaty. This cooperation between the two EC institutions consists primarily of the Commission's right to make proposals for almost all important measures adopted by the Council. These powers of implementation are exercised within the framework of general rules made by the Council, and the limits of the power are to be inferred from the particular wording of the provision in question.

The Cooperation Procedure

The Council and Commission

The Commission has the right to initiate action on most issues under the provisions of the treaty. The Commission exercises that right to a large extent by making proposals to the Council. Article 149 restricts the power of the Council to amend the proposal of the Commission. Under that article, any amendment of a proposal of the Commission must be approved by unanimous vote, even though the adoption of the proposal itself may require a different majority. The effect of this provision strengthens the position of the Commission and makes the amendment of its proposals less likely. If a single member state objects to an amendment, and if the Commission fails to make changes desired by a majority of the Council, the Council can then either accept or reject the proposal as made by the Commission. If, on the other hand, the Council adopts an amendment by unanimous vote, the acceptance or rejection of the amended proposal must still be voted on, and the outcome of that vote depends upon whether the amended proposal is approved by the majority required under the applicable provision of the treaty, The Council must at all times observe the other rules of the treaty.

The European Parliament

When the Council has received a proposal from the Commission and obtained the Opinion of the European Parliament it may adopt a "common position. " This ''common position' must be communicated to the European Parliament, along with the reasons that led to its adoption as well as the Commission's position on the draft. The act may be definitively adopted if within 3 months (subject to a l-month extension) the Parliament approves the measure or has not acted on it. The Parliament may, alternatively, within this 3-month period propose amendments to the Council's common position, or it may reject the position. If the Parliament rejects the Council's common position, unanimity is required for the Council to act on a second reading.

The proposal on which the Council has taken a common position must be reexamined by the Commission within 1 month, and the Commission must take into account the amendments proposed by the European Parliament. Any parliamentary amendments which the Commission has chosen not to accept must be forwarded to the Council, together with the Commission's opinion on them. The Council may adopt these amendments, notwithstanding the Commission's rejection, but its vote must be unanimous. The Council must adopt any proposal that has been reexamined by the Commission; if the Council wishes to make any amendments, it can do so only by unanimous vote. If the Council does not take any action within 3 months, the Commission's proposal is deemed not to have been adopted. If the Council and the Parliament agree, this 3-month period may be extended for a maximum of 1 month.

Paragraph three of article 149 gives some flexibility to Commission proposals by authorizing the Commission to amend its original proposal as long as it has not been acted upon by the Council.

SOURCE: Common Market Reports.

320-226 [1 92 15

Appendix C

Workshop Participants and Reviewers

Software Engineering Technology and Intellectual Property Issues Workshop

Richard A. Belgard
Consultant

Bruce I. Blum
The Johns Hopkins University

James Chandler
The George Washington University

G. Gervaise Davis III
Schroeder, Davis & Orliss, Inc.

David Eichmann
West Virginia University

Gideon Frieder
Syracuse University

Andrew Johnson-Laird
Johnson-Laird, Inc.

Lindsey Kiang
Digital Equipment Corp.

Tom Kirkland
MCC

John F. Kramer
DARPA/ISTO

Peter Menell
University of California at Berkeley

Bryan Pflug
Boeing Commercial Airplane Co.

Hasan Sayani
Advanced Systems Technology Corp.

Software-Developer Issues Workshop

Eleanor Arita
Sunburst Communications, Inc.

Douglas K. Brotz
Adobe Systems, Inc.

Chris Byrnes
Landmark Systems Corp.

David Curtis
Microsoft Corp.

Bernard Goldstein
Broadview Associates

Jacqueline C. Morby
TA Associates

David P. Reed
Lotus Development Corp.

John Shore
Entropic Research Laboratory, Inc.

D. Ezra Sidran
Intergalactic Development, Inc.

Dennis Snow
Plimouth Research

Helene Stewart
Oracle Complex Systems Corp.

Public Interest Issues Workshop

David Bender
White & Case

Stanley M. Besen
The RAND Corp.

Anne W. Branscomb
Harvard University

Lynn Robert Carter
Software Engineering Institute

Steven W. Gilbert
EDUCOM

Janlori Goldman
ACLU-National Headquarters

Bruce A. Lehman
Swidler & Berlin, Chartered

Arthur Levine
Finnegan, Henderson, Farrabow, Garrett & Dunner

Paul Mayer
ZPAY Payroll System, Inc.

John McCarthy
Stanford University

Steven J. Metalitz
Information Industry Association

Robert L. Oakley
Georgetown University Law Center

Marybeth Peters
U.S. Copyright Office

Linda Roberts (Chair)
Office of Technology Assessment

Marc Rotenberg
Computer Professionals for Social Responsibility

Irving Samuels
Capital P.C. Users Group

G. Lee Skillington
U.S. Patent and Trademark Office

Richard Stallman
Free Software Foundation

Richard H. Stern
Oblon, Spivak, McClelland, Maier, & Neustadt, PC

Digital Libraries, Electronic Publishing, and Intellectual Property Workshop

David Bearman
Archive and Museum Informatics

Alan Bogage
Howard County Central Library

Daniel Gross
Flow Research

Steven L. Haynes
West Publishing Co.

Lee A. Hollaar
University of Utah

Mary Jensen
University of South Dakota

William Johnson
University of Virginia

Brian Kahin
Harvard University

Robert Lynch
McGraw Hill Inc.

Susan Saltrick
John Wiley and Sons

Harold Schoolman
National Library of Medicine

Marvin Sirbu
Carnegie Mellon University

Suzanne Thorin
Library of Congress

Ted K. Yamashita
CONTEL Technology Center

User Interface Technologies and Intellectual Property Workshop

Bernard Galler
University of Michigan

Michael Good
Digital Equipment Corp.

Paul Heckel
Hyperracks

Robert Jacob
Naval Research Laboratory

Nancy Kendzierski
Hewlett-Packard Laboratories

Michael Lesk
Bell Communications Research

Aaron Marcus
Aaron Marcus and Associates

David C. Nagel
Apple Computer

Paul Reed
AT&T Bell Laboratories

Victor Rosenberg
Personal Bibliographic Software

Robert Scheifler
MIT X Consortium

The Present Copyright/Patent/Trade Secret System of Protection for Computer Software Workshop

Richard Bernacchi
Irell & Manella

Allen Grogan
Blanc, Gilburne, Williams & Johnston

David L. Hayes
Fenwick & West

Dennis S. Karjala
Arizona State University

Raymond Nimmer
The University of Houston

Kevin O'Connor (Chair)
Office of Technology Assessment

Charles Ortner
Milgrim, Thomajan & Lee

Pamela Samuelson
The University of Pittsburgh

Robert Greene Sterne
Sterne, Kessler, Goldstein, & Fox

Reviewers

Prue Adler
Association of Research Libraries

Richard A. Belgard
Consultant

David Bender
White & Case

Richard Bland
Storage Technology Corp.

Bruce I. Blum
The Johns Hopkins University

Robert S. Bramson
Robert S. Bramson & Associates

Anne W. Branscomb
Harvard University

Dan Bricklin
Slate Corp.

Sara Brown
Software Publishers Association

Clark Calkins
CC Software

Lynn Robert Carter
Software Engineering Institute

Donald S. Chisum.
University of Washington

Stephen Chow
Cesari & McKenna

Eileen D. Cooke
American Library Association

Brad Cox
Information Age Consulting

Bridget Czamota
European Community Commission

Paul I. Davis
Martin Marietta Energy Systems

Joseph Ebersole
Attorney

Michael Ernst
MIT Laboratory for Computer Science

Joseph Farrell
University of California at Berkeley

Nicole Field
Software Publishers Association

Bernard A. Galler
The University of Michigan

Paul Goldstein
Stanford University

Gregory E. Gorman
Computer & Communications Industry Association

Stephen Gould
Congressional Research Service

Seth Greenstein
McDermott, Will & Emery

June Hargrove
University of Maryland

David L. Hayes
Fenwick & West

Steven L. Haynes
West Publishing Co.

Paul Heckel
Hyperracks

Wayne Herrington
U.S. International Trade Commission

George Hersey
Yale University

Edwin Hettinger
College of Charleston

Heidi Hijikata
U.S. Department of Commerce

Lee A. Hollaar
University of Utah

Robert Holleyman
Business Software Alliance

Robert Jacob
Naval Research Laboratory

Jan Jancin
American Bar Association

Mary Jensen
University of South Dakota

William Johnson
University of Virginia

Andy Johnson-bird
Johnson-Laird, Inc.

Brian Kahin
Harvard University

Dennis S. Karjala
Arizona State University

Michael S. Keplinger
U.S. Patent and Trademark Office

Tom Kirkland
Microelectronics and Computer Corp.

Glenn J. McLaughlin
Congressional Research Service

Peter Menell
University of California at Berkeley

Steven J. Metalitz
Information Industry Association

John S. Morrison
Technology Transfer International

Glenn Myers
University of Iowa

Jeff Neuburger
Weil, Gotshal & Manges

Raymond Nimmer
The University of Houston Law Center

Alessandro Nova
Stanford University

Robert L. Oakley
Georgetown University Law Center

Charles Ortner
Milgrim, Thomasan & Lee

Ronald J. Palenski
ADAPSO

Linda Pellecchia
University of Delaware

Marybeth Peters
U.S. Copyright Office

Shirley M. Radack
National Institute of Standards and Technology

David P. Reed
Lotus Development Corp.

Pamela Samuelson
The University of Pittsburgh

Glen Self
Electronic Data Systems, Inc.

Sandra Sellers
Willian, Brinks, Olds, Hofer, Gilson & Lione

Robert Sherwood
Consultant

G. Lee Skillington
U.S. Patent and Trademark Office

Mary Smolenski
U.S. Department of Commerce

Oliver R. Smoot
CBEMA

Anne Staines
European Community Commission

Richard Stallman
Free Software Foundation

Joshua Stem
Datatrope Corp.

Robert Greene Sterne
Sterne, Kessler, Goldstein & Fox

Rosemary Talab
Kansas State University

Suzanne Thorin
Library of Congress

Kenneth A. Wasch
Software Publishers Association

Philip Webre
Congressional Budget Office

Douglas R. Weimer
Library of Congress

Suzanne P. Weisband
University of Arizona

Milton Wessel*
Georgetown University Law Center

Alice Zalik
U.S. Patent and Trademark Office

*Deceased.

Index

Index

ADAPSO. See The Computer Software and Services Industry Association
Advisory Commission on Patent Law Reform, 10-11,55-56
Advisory Committee for Trade Policy and Negotiations, 109
American Memory Project, 169
Amnesty programs, 101
Anticompetitive product preannouncements, 199
Antitrust policies, 190
Apple v. Franklin, 13,68-69
Association for Computing Machinery/IEEE Computer Society Task Force on the Core of Computer Science, 156
Attaching programs, 127
Authorship issues, 112-113, 131, 174-175

Background of issues, 3-4
Baker v. Selden, 69
Barriers to entry, 190-191
Berne Convention, 16, 104-107, 110
Berne Convention Implementation Act of 1988, 106-107
Biotechnology issues, 40-41
Board of Patent Appeals and Interferences, 44-45
Breadth of a patent, 194-197
Broad patent protection, 192-193
Broderbund Sofware, Inc. v. Unison World, Inc., 72
BSA. See Business Software Alliance
Bureau of National Affairs, 109
Business Software Alliance, 95, 98
  overseas piracy, 101-103

CASE. See Computer-Aided Software Engineering
Case law
  copyright, 13, 68-73
  patents, 45-52
CBEMA. See Computer and Business Equipment Manufacturers Association
CCPA. See Court of Customs and Patent Appeals
CD-ROM. See Compact disc, read-only memory
Circular R61, 65-66
Clean room microcode, 71-72
Clean rooms, 139-140
Command language dialogue, 127-128
Compact disc, read-only memory, 166
Compatibility
  network externalities and, 198-199
Compendium of Copyright Office Practices, 66
Compilations
  copyright infringement and, 174
  databases and, 73-77
Computer-Aided Software Engineering, 168
Computer and Business Equipment Manufacturers Association, 96
Computer Associates International, Inc. v. Altai, Inc., 13, 72, 140
Computer science, undergraduate curricula, 158
Computer Software and Intellectual Property, 4
The Computer Software and Services Industry Association, 98
Congress. See U.S. Congress
Consistency issues, 129-130
Contract law protection of electronic and computer technology, 83-84

CONTU. See National Commission on New Technological Uses of Copyrighted Works
Cooling Systems & Flexibles, Inc. v. Stuart Radiator, Inc., 74
Cooperative agreements, 192-193
Copyright. See *also* Berne Convention; Copyright Act; Copyright law; Piracy
  authorship, 131, 174-175
  clean rooms, 139-140
  copying in Renaissance art, 62-63
  copyrightable expression, 60-61, 68-69, 70, 72-73, 74, 87, 139-140, 144
  recompilation, 6,7, 19,25, 146, 147-148, 150
  digital information issue, 19, 170-179
  early protection, 58-59
  economics literature, 183-185
  electronic mail messages, 171-172
  external design protection, 139
  home copying, 199-201
  intention of U.S. system, 186-187
  international issues, 25
  multimedia works, 36
  originality, 173-174
  ownership issue, 67-68
  permissions and royalty collection, 36
  policy areas and options, 29-31
  scope of protection, 9-10, 60-61, 151, 194
  size of *a* work, *172*
  statutory subject matter, 13
  technical challenges, 22-23
  unauthorized copying, 176-177
  "work" definition, 171-173
Copyright Act, 27,87, 111, 112, 144, 178
  fair use and, 61-65
  1980 amendment, 67-68
Copyright and Home Copying: Technology Challenges the Law, 4, 199-201
Copyright law. See also Copyright; Copyright Act; *Sui generis*
  backup copies of computer programs, 177
  case law, 13, 68-73
  categories of works, 172
  control of access to information, 176
  Copyright Office and, 65-66
  databases, 73-77
  development of, 58-60
  digital information and, 178
  expression definition, 144
  fair use issue, 31,35-36,61-65
  in foreign countries, 80-82
  goals, 56-58
  hybrid design protection, 77-78
  industrial design protection, 76-77
  levels of abstractions test, 72
  limitations to owner's exclusive rights, 61
  mask work protection, 76
  merger doctrine, 70, 145
  mixed-media work protection, 172-173
  object code and source code protection, 68-69
  patent/trade secret/copyright laws relationships, 86-88
  policy position arguments, 151, 153
  program code protection, 144

rule of doubt, 66
scope of protection, 9-10, 60-61, 151, 198
structure, sequence and organization protectability, 69-72
user interface, 72-73
Copyright Office
program code and screen display relationship, 153
responsibilities, 65-66, 109
rule of doubt, 66
Copyrightable expression, 60-61
clean rooms, 139-140
definition, 144
distinguishing from unprotected elements, 72-73
expression of facts, 74
form of, 87
idea/expression merger, 70
nonliteral elements test, 73
object code and source code protection, 68-69
Council Directive on legal protection of computer programs, 16-17
authorship of programs, 118
beneficiaries of protection, 118
recompilation, 119-120
exceptions to the restricted acts, 119
object of protection, 115
other legal provisions, 121
restricted acts, 118-119
special measures of protection, 120
term of protection, 120-121
Council of the European Economic Community, 112-113
Counterfeiting, 99-100
Court of Customs and Patent Appeals
best mode requirement, 44
mathematical algorithms guidelines, 133-134
mental steps doctrine decisions, 46-52
patentability of software-dated inventions, 132-133
point of novelty test, 49-50
Cryptography, 137
Czechoslovakia
bilateral treaty, 115

Databases
copyright law and, 73-77, 113
mixed-media, 169
Recompilation, 6,7, 19,25, 119-120
definition, 146
disassembly and, 147-148
USes of, **148, 150**
Design Innovation and Technology Act of 1991,77
Design patents, 42
protection, 76
statutory subject matter, 12
Diamond v. Chakrabarty, 40
Diamond v. Diehr, 12,24,32,51-52, 134
Digital Communications Associates v. Softklone Distributing Corporation, 72-73, 153
Digital information
advantages and disadvantages, 166-168
characteristics, 170-171
controlling use of, 175-179
copying, 176-177
copyright issues, 19-20, 170-179)
digital libraries, 169-170
electronic publishing, 161-170
hypertext, 168-169
libraries and, 177-179
mixed-media, 169, 172-173
scholarly publishing system and, 167
storing and retrieving data, 162
storing text and images, 163-164
user's view, 171
Digital libraries, 169-170
Disassembly
recompilation and, 147-148
definition, 146
''Draft Final Act Embodying the Results of the Uruguay Round of Multilateral Trade Negotiations, " 114

Economic issues, **183-201**
he-enterprise economy, 184
intellectual property, 20-21
literature on, 183-185
social benefits, 186-194
technology changes, 22
E.F. Johnson v. Uniden, 139
Electronic mail messages, **171-172**
Electronic publishing
CD-ROM, 166
definition, 171
nonprint, 165-170
on-line information retrieval service, 165-166
print-based, 162, 164
royalties, 175
End-user piracy, 97-98, 100-101
Engineering Dynamics, Inc. v. Structural Software, Inc. and S. Rao Guntur, 73
European Economic Community
antitrust regimes, 190
Council Directive on legal protection of computer programs, 16-17, 115, 118-121
Green Paper, 115, 116-118
trade agreement, 108
Expression. See Copyrightable expression
External design
copyright protection, 139
definition, 17-18
intellectual property protection of, 138-140
patent protection, 139
policy issues, 140-141
technology, **126-127**
trade secret law protection, 139

Fair use
copyright issues, 31, 35-36,61-65
libraries and, 35
patents, 31
Fax machines, 176
Federal government concerns, 156-157
Feist Publications Inc. v. Rural Telephone Service Co., Inc., 74-75, 174
France
software piracy, 102
Free-enterprise economy, 184
Free riders, 185
Freeman-Walter-Abele test, 133, 134, 137-138

GATT See General Agreement on Tariffs and Trade
GCA Corp. v. Chance, 69
General Agreement on Tariffs and Trade, 16, 107-114
  Office of the U.S. Trade Representative, 109-110
  trade-related intellectual property rights, 110-114
Global software industry
  data collection, 93-94
  markets, 94-97
  technology, 94-97
Gottschalk v. Benson, 12,32,4749, 133, 150-151

Healthcare Affiliated Services, Inc. v. Lippany, 71
Home copying
  effect on economic welfare, 199, 200
  piracy, 97
  price discrimination, 201
  resource allocation, 201
  variety, 201
Hybrid design protection, 77-78
Hypermedia, 168-169
Hypertext, 168-169

IIPA. See International Intellectual Property Alliance
Illegal copying. See Piracy
In re Abele, 52, 53
In re Abrams, 46
In re Benson, 47
In re Bernhart, 47,48
In re Bradley, 51
In re Chatfield, 50
In re Christensen, 49
In re deCastelet, 50
In re Deutsch, 50
In re Freeman, 50, 52
In re Grams, 54
In re Iwahashi, 54
In re Johnson, 51
In re Johnston, 50
In re MacIlroy, 49
In re Mahoney, 49
In re Meyer, 53
In re Musgrave, 47,48
In re Nell, 50
In re Pardo, 53
In re Prater, 46-47
In re Richman, 50
In re Sherwood, 44
In re Taner, 53
In re Toma, 50
In re Waldbaum, 49,50
In re Walter, 51, 52
In re Yuan, 46
Incentives
  for cumulative research, 192-194
  excessive, 188-190
India
  copyright and patent violations, 103
Industrial design protection, 76-77
Industry
  perception of patent effectiveness, 190-192
Installed base, 198-199
Intellectual property
  economic issues, 20-21
  program function, 132-135
Intellectual Property Rights in an Age of Electronics and Information, 3
International Intellectual Property Alliance, 98, 103
International issues. See *also* European Economic Community
  Berne Convention, 16, 104-107
  copyright law, 80-82
  copyright provisions, 25
  Council Directive on legal protection of computer programs, 16-17, 115-121
  design protection, 77
  "Draft Final Act Embodying the Results of the Uruguay Round of Multilateral Trade Negotiations," 114
  General Agreement on Tariffs and Trade, 16, 107-114
  global software industry, 14-15,93-97
  level of patent protection, 25-26
  patent laws, 57
  software piracy, 15-16, 97-104
  software revenues, 14-15
  trade secret law, 89-90
  treaties and agreements, 16-17
International Trade Commission, 95
Inventions. See *also* Patent law; Patents
  excessive incentives, 188-190
  incentives, 196
  private returns, 189
  underinvestment in, 184
Issues and options
  copyright law, 22-23
  digital information and copyright, 26
  future options, 36
  international aspects, 25-26
  patents, 23-25
  policy areas and options, 28-36, 184
  software industry structure, 26-28
Italy
  software piracy, 102

Japan
  antitrust regimes, 190
  competition with U.S. software producers, 15
  markets, 96-97
Judicial system
  software 'protection process, 11-12

Kewanee Oil case, 86
Key Publications Inc. v. Chinatown Today Publishing Enterprises Inc., 75
Kregos v. Associated Press, 75

Landmine patents, 12, 35, 136
Levels of abstractions test, 72
Libraries. See *also* Digital libraries
  digital information and, 177-179
  effect of copyright, 19-20
  fair use issues, 35-36
  on-line catalogs, 170, 179
  remote access to services, 178-179
  resource sharing, 178
Library of Congress
  digital library prototype, 169
  MARC system, 170
Licenses, 107, 192

Literal code, 144
Lotus *Development Corporation v. Paperback Software International, 13, 73, 142-143*

Machine Readable Cataloging, 170
MARC. See Machine Readable Cataloging
Market entry, 190-191, 197
Mask work protection, 76
Mathematical algorithms, 10, 17, 23, 32-33,51-54, 133-134
Mazer v. Stein, 186
Mental steps doctrine, 46-52, 150
Menus, 128
Midway Mfg. Co. v. Strohon, 69
Mixed-media databases, 169
Mixed-media works
  converting works to digital form, 173
  copyright protection, 36, 172-173
Monopoly rights, 185-186
Motion Picture Association of America, 103
Multimedia databases, 169
Multimedia works
  converting works to digital form, 173
  copyright protection, 36, 172-173
Multiple inventors, 189-190
Multivendor suit, 101

Narrow patent protection, 193
National Commission on New Technological Uses of Copyrighted Works
  computer-aided or computer-generated works, 112-113
  computer program copyright recommendations, 60
  Copyright Act amendment recommendations, 67-68
  objectives, 6
  rule of doubt recommendations, 66
  writing a program to perform a particular function, 145
NEC *Corp. v. Intel Corp.*, 71-72, 145
Network externalities, 197-199
Neural networks, 152
New *Developments in Biotechnology; Patenting Life, 3,4041*
New Zealand
  software piracy, 101
Noncommercial copying. See Home copying
Nonexclusivity, 185
Nonobviousness, 43, 76
Novelty, 4243,49-50,76, 193

Object code, 130
  and source code protection, 68-69
Obviousness, 43
Omnibus Trade Act
  "Special 301," 104
On-line catalogs, 170, 179
On-line Computer Library Center,, 170
On-line information retrieval service, 165-166
Options. See Issues and options; Policy areas and options

Parker v. Flook, 12, 32,42,50-51, 134, 138
Patent Act of 1952,39
Patent and Trademark Office, 109
Patent design, 185, 194-197
Patent law. See *also* Patents; U.S. Patent and Trademark Office
  case law evolution, 45-52
  mental steps doctrine, 150
  patent/trade secret/copyright laws relationships, 86-88
  policy position arguments, 150-151
  response to court cases, 52-56
Patent races, 192-194
Patent Reform Act of 1967,45
Patents. See *also* Patent law; Software patents; *specific types of patents by name;* U.S. Patent and Trademark Office
  "best mode" requirement, 43-44
  biotechnology issues, 40-41
  doctrine of equivalents, 135
  economics literature, 183-185
  examination quality and speed, 9, 24, 34-35
  external design protection, 139
  fair use, 31
  infringement issues, 135
  intention of U.S. system, 186-187
  licenses, 192
  mathematical algorithms, 10, 17, 23, 32-33, 51-54, 133-134
  nonobviousness, 43, 76
  novelty, 42-43, 49-50, 76, 193
  obviousness test, 43
  patent system description, 39-45
  policy areas and options, 31-35
  prior art, 6-7,8,33-35
  process, 43-45
  protection, 187-190, 192-193
  rejection conditions, 42-43
  statutory subject matter, 12, 32-33,4142, 133
  technical challenges, 23-25
  term and breadth, 45, 194-197
  trade secret law, 194
  utility, 42
  Yale study, 191-192
People's Republic of China
  copyright and patent violations, 103
Permissions
  copyright and, 36
Perseus project, 168-169
Photocopies, 176
Piracy, 15-16, 148
  counterfeiting, 99-100
  end-user, 100-101
  loss estimates, 98-99
  overseas, 101-103
  retail, 97-98, 99-100
  trade issues and, 103-104
Plains Cotton Cooperative Association v. Goodpasture Computer Serv., Inc., 13,71
Plant patents, 42
Plant Variety Protection Act of 1970,41
Point of novelty test, 49-50
Policy areas and options
  CONTU research, 184
  copyright protection scope, 29-31
  libraries, 35-36
  patent protection scope, 31-35
  principal policy areas, 28-36
PRC. See People's Republic of China
Predatory pricing, 199
President's Coremission on the Patent System, 45
Prior art issue, 6-7, 8, 33-35
Private copying. See Home copying
Program code

copies, 145-146
definition, 18-19
literal code, 144
literal copying, 146
nonliteral copying, 144-145
policy issues, 146
protected and unprotected elements, 146
protection, 144-146
screen display relationship, 153
technology, 130
Program function
definition, 17
intellectual property protection of, 132-135
software patent policy issues, 135-138
technology, 126
Programm ing languages, 7, 127, 130
PTO. See U.S. Patent and Trademark Office
Public goods
rationale for granting rights, 185-186
PVPA. See Plant Variety Protection Act of 1970

Q-Co. *Industries, Inc. v. Hoffman, 70, 145*

Reciprocity protection, 105
Recording Industry Association of America, 103
Research and development
optimal patent and, 194-196
underinvestment in, 184
Research Libraries Information Network, 170
Restatement of Torts, 79
Retail piracy. See Piracy
Reverse *engineering,* 31, 76, 148, 150. *See also* Decompilation
Romania
bilateral treaty, 115
Royalties
collecting, 36
electronic publishing and, 175

SAS *Institute, Inc. v. S&H Computer Systems, Inc., 70, 145, 146*
Schroeder v. William Morrow & Company, 74
SCPA. See Semiconductor Chip Protection Act of 1984
Semiconductor Chip Protection Act of 1984, 13, 27, 75
Sentry Market Research, 95
Small fins, 188
Societal benefits
cumulative technological progress, 192-194
intellectual property systems and, 186-194
market entry, 190-191
stronger patent protections and, 187-190
Yale study, 191-192
Software debate
complexity, 4-5
evolution, 5-12
stakeholder groups, 8-11
Software development
policy position legal arguments, 153, 155-156
Software industry
changes since CONTU, 188-189
global, 93-97
structure, 26-28
Software Patent Institute, 34, 56
Software patents, 32-33, 134-138
Software Publishers Association, 96, 98, 100, 101
Software reuse, 154-155
Software technology
changes since CONTU, 188-189
elements, 17-19, 125-126
external design, 126-127
program code, 130
program function, 126
user interface, 127-130
Source code, 130
South Korea
software piracy, 102
Soviet Union
bilateral treaty, 115
software piracy, 102-103
SPA. See Software Publishers' Association
Spain
software piracy, 102
SPI. See Software Patent Institute
SS0. See Structure, sequence, and organization
Standardization
network externalities and, 198-199
Structure, sequence, and organization, 69-72, 145, 151, 153
Study approach, 4-12
Substituting programs, 127
Sui generis approaches, 7-8, 26-28, 75-76,78
Supreme Court. See *also specific cases by name*
authorship definition, 174
obviousness test, 43
statement on patents and copyright, 57
Synercom Technology, Inc. v. University Computing Co., 72

Technology transfer, 98-99
Telemarketing v. Symantec, 142
Term of a patent, 194-197
Thailand
copyright and patent violations, 103
Trade Act of 1988, 103
Trade-related intellectual property rights
databases, 113
economic rights, 113
protection, 111-113
rental rights, 113
U.S. proposal to the GATT, 16, 110-114
Trade secret law
basis, 79
external design protection, 139
in foreign countries, 89-90
object of, 78-79
patent/trade secret/copyright laws relationships, 86-88, 183
protection indicia, 82
shrink wrap license, 84-86
software and, 82, 84-86
Trade secrets, See *also* Trade secret law
characteristics, 79, 82
definition, 79
economics of law, 194
efforts to retain secrecy, 82
statutory subject matter, 12-13
Treaties. See *also specific treaties by name*
bilateral, 114-115
TRIPS. See Trade-related intellectual property rights

UCC. See Universal Copyright Convention
UNESCO. See United Nations Educational, Scientific and Cultural Organization
Uniform Trade Secrets Act, 79
United Kingdom
  software piracy, 102
United Nations Educational, Scientific and Cultural Organization, 106
Universal Copyright Convention, 16, 106, 115
Uruguay Round, 107-109
U.S. Congress. See also Copyright law; Patent law; Trade secret law; *specific laws by name*
  "fast-track" negotiating authority, 109, 110
  power to grant copyrights and patents, 185
U.S. Department of Commerce, 109
U.S. Department of State, 109
U.S. Economic Policy Council, 103
U.S. Patent and Trademark Office. See *also* Patent law; Patents; *specific types of patents by name*
  administrative problems, 54-55
  examination quality and speed, 9, 24, 34-35, 40
  Freeman-Walter-Abele test, 133, 134, 137-138
  mathematical algorithms guidelines, 133-134
  patent protection scope, 31-35
  program code, 132
  program examination guidelines, 45
  response to court cases, 52-56
  technological and institutional changes, 23-25
  utility patent classes, 42
U.S. Trade Representative, 103, 109-110
User interface
  case law, 72-73
  constraints, 129-130
  copyright law protection, 151
  design, 18, 128-129
  and network externalities, 198
  policy issues, 143-144
  protected and unprotected elements, 142-143
  standards, 143
  technology, **127-130**
User satisfaction, 197
USTR. See U.S. Trade Representative
Utility, 42
Utility patents, 42
  statutory subject matter, 12
UTSA. See Uniform Trade Secrets Act

Vault Corp. v. Quaid Software Ltd., 86

West Publishing Co. v. Mead Data Central, 74
Whelan Assocs., Inc. v. Jaslow Dental Laboratory, Inc., 13, 69-70, 145, 146
White Consolidated Industries, Inc. v. Vega Servo-Control, Inc., 44
Williams & Wilkins Co. v. United States, 64
Williams Electronics, Inc. v. Artic International Inc., 68
Windows, 128
WIPO. See World Intellectual Property Organization
Workalike programs, 127
Workshop issues, 5
World Intellectual Property Organization, 106, 108
  draft treaty, 11,56
  patent system modification treaty, 26
Wright v. Warner Books, 65

Yale study, 191-192