The Protection of Computer Software

A Comparative Study of the American and German Law

by

Daniela H. Birkenfeld

Thesis submitted to the Faculty of Graduate Studies and Research of McGill University in partial fulfillment of the requirements for the degree of Master of Laws (LL.M.).

Institute of Comparative Law

McGill University

Montreal, Quebec

Canada

ISBN 0-315-34424-5

With love to my mother and

in memory of my father

Abstract

This thesis deals with the legal protection of computer software in the Federal Republic of Germany and the United States. It focusses upon traditional forms of intellectual and industrial property protection namely, patent, trade secret, and copyright Laws.

In the patent area, the issue is whether software involves mental processes, mathematical algorithms, or fundamental laws of nature which are not protectable.

In the copyright area, the question is whether the form of software used on the computer is in any sense a "literary work", since it is only intended to control the operation of a machine.

Trade secret law has traditionally been used by software producers as a primary method of enforcing and preserving intellectual property rights.

Although the current market for software is large, it is none the less expected to grow dramatically in the next five or ten years. Since software piracy has become widespread, the protection of software owners is ultimately a question of international cooperation. The study will point out the possibilities for protection in both jurisdictions, and the international agreements to which both are parties.

Résumé

La présente thèse traite la protection du materiel informatique "software" en République Fédérale d'Allemagne et aux Etats - Unis. L'étude se referre aux formes traditionnelles de protection de la propriété intellectuelle et industrielle: Droit de brevets, secret commercial et Droit d'auteur.

Dans le domaine des brevets, le problème est de déterminer si le materiel "software" est composé de créations intellectuelles ou integre des algorithme mathématique ou encore des lois de la nature non susceptibles de protection.

Dans le domaine du Droit d'auteur, le problème est de savoir si la forme de "software" qui est actuellement utilisée sur les ordinateurs est, sous un certain aspect, un "materiau litteraire" puisqu'il est seulement destiné a l'exploitation d'une machine.

La loi sur le secret commercial est utilisée traditionnellement par les fabricants comme protection première de leurs droit de propriété intellectuelle.

Quoique le marché des materiaux "software" soit actuellement très large, on prénait qu'il va croître énormément dans les prochaines cinq ou dix années. Le travail portera sur les possibilités de protection dans les deux Etats et les accords internationaux auxquels ces pays sont parties prenantes.

Table of Contents

Table of Cases

List of Abbreviations

A.2d	Atlantic Reporter 2d Series
BAG	Bundesarbeitsgericht
BB	Betriebsberater
BCC	Berne Copyright Convention
BGH	Bundesgerichtshof
BPat G	Bundespatentgericht
CONTU	Commission on New Technological Uses of Copyright Works
e.g.	exempli gratia
F.2d	Federal Reporter 2d Series
F.Supp.	Federal Supplement
FRG	Federal Republic of Germany
GRUR	Gewerblicher Rechtsschutz und Urheberrecht
i.e.	id est
int.	international
LG	Landgericht
NJW	Neue Juristische Wochenschrift
OLG	Oberlandesgericht
Pat G	Patentgesetz
RAM	Random Access Memory
RG Z	Reichsgericht in Zivilsachen
RIW	Recht der internationalen Wirtschaft
ROM	Read Only Memory

Sec. or s.	Section
UCC	Universal Copyright Convention
Urh G	Urhebergesetz
U.S.	United States
UWG	Gesetz gegen den Unlauteren Wettbewerb
WIPO	World International Property Organization
WRP	Wettbewerb in Recht und Praxis
ZUM	Zeitschrift fuer Urheber- und Medienrecht

Chapter 1:

Introduction

This thesis examines the legal protection available for computer software in the Federal Republic of Germany and in the United States (1). Though computers were developed in the mid-1940s, the need for legal protection of computer programs did not arise until the 1960s. The initial impetus for the development of an independent software industry was provided in 1969 when International Business Machines (I.B.M.) in the United States stopped "bundling" its hardware and software. A worldwide information explosion causes a demand for new products and services to help manage vast amounts of information efficiently and effectively. The dataprocessing industry has provided a wide variety of computing machines and computer software in response to this demand. In 1976, the personal computer was developed, and with the revolution in the semiconductor technology making small and relatively inexpensive computers possible, the market for computer programs was expanded to include small-business and individual users.

During the past two decades, the foundation of the United States' economy has shifted from primary a dependence--on labour-intensive manufacturing industries, to dependence on

the development of advanced technology. The growth rate of the software industry has greatly exceeded that of the hardware industry and, due to decreasing hardware costs, the future profits of computer technology may well lie in software. Perhaps because the software industry has become a major commercial and technological contributor to the success of American business, the creators of software in the United States have established an undisputed worldwide lead.

Millions of computers are used in the United States, and all of these need programs in order to perform useful functions. Although the current market for software is already very large, it is expected to grow dramatically in the next five to ten years. Software industry revenue for 1983 is estimated at $ 17 billion worldwide; in 1987 it should reach $ 55 billion (2).

Legal problems have increased as changes in technology have caused the microcomputer to become part of the mass market. As Robert P. Merges has pointed out, legal rules change in response to the effects of new technologies, not in response to technical developments themselves (3).

Because programs in machine-readable form are as easy and cheap to copy as they are difficult and expensive to develop and refine, it is not surprising that software piracy has become widespread. To comprehend the extent of the piracy problem one has only to imagine that for the price of a blank disk, which is approximately four dollars, a computer operator

may copy a program retailing for hundreds of dollars on the open market.

The amount of piracy in this rapidly expanding market is difficult to estimate with any precision. However, some analysts have estimated that as much as thirty percent of potential software industry revenues are lost to piracy (4). As a result, vendors of original software are motivated to over-price their products initially in order to factor the likelihood of diminished revenues due to piracy. Even this tactic, however, may not fully compensate the originating vendor, and in this writer's opinion, may create a further incentive for piracy.

The software industry, authors, users, and government have shown an increasing interest in legal protection of programs. Hundreds of software-infringement cases are pending throughout the United States (5). And all three branches of the U.S. Government have taken actions which demonstrate their interest.

First, the U.S. Supreme Court handed down two significant decisions construing specific sections of the patent law (6). Second, Congress enacted a new Copyright Act and amended the patent law. Third, President Reagan urged Congress to modify the federal antitrust and intellectual property laws to improve the competitiveness of American industries in international markets.

Software is a valuable end-product of human endeavour, requiring substantial technical and financial resources. There is no doubt that it should enjoy the same degree of legal protection as any other property. The question is what form of protection it should have. This essay analyzes two different approaches to the protection of computer software, by comparing the protection schemes in the United States (which has a statute specifically providing protection under copyright) with those of Germany (which protects computer software by application of existing law) (7).

Traditionally there are three forms of intellectual property protection in both these jurisdictions: patent law; copyright law; and, trade secret law. The debate over the nature of legal protection available or appropriate to software centers on the question, what is the nature of software?

In the patent area, the issue is whether software states mental steps or incorporates mathematical algorithms or other fundamental laws of nature which traditionally are not patentable. It is interesting to note that a patent was granted in Germany for a program that was denied protection in the United States (8).

In the copyright area, the issue is whether the form of software which is actually used on the computer, ("binary form"), is in any sense a "literary work", since it is intended only to operate a machine.

Trade secret law, at least traditionally, has been used by

software producers as the primary method of enforcing and preserving intellectual property rights. In this area the main problem with which software owners have had to deal has been a more practical one: namely, how to keep the secret.

Since it is essential to understand programs in their entirety in order to realize the legal problems of protecting computer software, this paper commences with an introduction to computer terminology (9). A detailed description and analysis of the protection of computer software in the United States and Germany follows. The thesis also canvasses existing international agreements in order to place in context the different software protection schemes of each country. Finally, a comparison of the different forms of legal protection is undertaken, as well as an analysis of their advantages and disadvantages.

Chapter 2:

Computer - Terminology:
Computer Programs and How They Work

Familiarity with the terms and phrases used in the computer software industry is a prerequisite to a coherent discussion of the protection of computer software. This is especially important insofar as several relevant judicial decisions, as shown below, have turned on distinctions between the different types of computer programs.

I. Hardware and Software

The first generation of computers did not utilize what are now referred to as computer programs to carry out their computational tasks. The machine could perform only the particular function for which it had been wired. In other words, the first computers were all hardware. The term hardware, as used today, includes the television monitor, storage devices, and the microcircuit chip.

Software, on the other hand, is the set of instructions used directly or indirectly in a computer to bring about a certain result (10). Software manipulates and instructs the various

hardware components to perform desired functional tasks.

II. Computer Programs

A computer program, often called software, is a set of serial instructions that directs the computer to perform certain tasks. It is typically written in several steps.

Before the programmer writes any code, he develops the program flow logic, a logical sequence of steps which the program will perform to accomplish its functions. The programmer uses this flow logic as a guide for writing a code. It can be read and understood by individuals, but not by computers (11).

Often, the most innovative and valuable aspect of this logic is contained in a set of sequential calculations or procedures which will result in the desired output after the input has been submitted. This logic is known as an algorithm. The essential part of most programs consists of a series of algorithms (12). The next step is to express the algorithms and other parts of the program in a source code.

III. Source Code and Object Code

A source code is the most sophisticated type of software: it instructs the computer hardware to perform a variety of specific functional tasks, such as word and data processing. Source code generally refers to any mnemonic system representing large numbers of machine instructions, such as the high level languages BASIC, PASCAL, COBOL and FORTRAN, which use commands resembling English words and symbols (13). Source code is intelligible to human beings.

Source code is then converted to object code through the use of a "compiler" program. Afterwards, the program is written in digital language consisting of zeroes and ones. Each zero or one carries one piece of information which is called a bit; the bit merges in a series of bits, called bytes (14). The operation of the compiler is more than a human language translation; it is not really comparable to a conventionel translation of a book, as from French into English. Object code has syntax rules that require a program in object code to contain many more statements than does the program in source language. There is no simple one-to-one relationship between the numbers of source code and object code. Some source code statements wholly disappear, while some object code statements may have no counterpart in the source code version (15). What the source code and the object code have in common

is not the specific literary expression chosen by the programmer, but a general structure and specific strategies for bringing about an intended result.

The machine-readable "binary-form" in which the written code (object or source code) is prepared is unintelligible to a human being; its actual form is electrical (consisting of high and low voltages) and magnetic (consisting of different polarities of magnetization) (16).

IV. Application Program and Operating Program

Application programs are the software programs with which the public has come to be familiar. They are normally written in high level programming languages and are designed to perform specific user-orientated tasks, like creating word-processing capability. They accept input from the user (such as the numbers to be added) and provide the user with an output (such as the sum of those numbers) (17).

Operating programs, in contrast, perform functions internal to the computer, such as directing the output to the proper output terminal device, creating the video display, and storing programs on a casette, tape, or disk (18).

V. Read Only Memory (ROM)

A ROM is a silicon chip on which a computer program in object code form is photochemically imprinted as a pattern of binary on/off switches (19). When activated in a given sequence these switches serve to give machine instructions to the computer, to which the chip itself is permanently wired. This results in the execution of the program. The stored information cannot be changed by the user of the computer and is therefore called "read only memory".

Memory chips on which users can write are known as Random Access Memory (RAM) chips and operate as a transient form of storage designed for use during program operation. A more recent development is the EPROM, an erasable programmable memory that can have its memory contents erased and reprogrammed (20). To be used in a computer, object code must be stored in a memory device, such as a floppy disk or a ROM.

Chapter 3:

Protection of Computer Software in the United States

Three forms of legal protection are available for computer programs in general: trade secret law; copyright law; and, patent law. Each form of protection is attented by certain advantages and disadvantages.

I. Trade Secret Law

Since secrecy is important to software authors and vendors, one might think that trade secret law would offer the best form of protection for computer programs. It is indeed the favored form of protection at the moment (21). Trade secret protection is available both domestically and internationally, even though there is no international convention specifically dealing with trade secret law (22). There is no dispute that computer programs and software are protectible subject matter under trade secret law (23). Trade secret law can protect the idea, the information, the invention, the design, as well as the expression of the idea. Thus, trade secrecy may extend its protection to the entire program.

A. Elements

Trade secret law has its roots in the common law, so that its content varies somewhat from state to state. This lack of uniformity may prove particulary troublesome to those who market their computer programs on a national scale (24). Nevertheless, twenty-five states and all federal jurisdictions (25) have substantially adopted the definition of trade secrets contained in section 757 of the Restatement of Torts, comment b (1939):

> (a)ny 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. The subject matter of a trade secret must be.....secret so that, except by the use of improper means, there would be difficulty in acquiring the information. An exact definition of a trade secret is not possible. Some factors to be considered in determining whether given information is one's trade secret are:
> (1) the extent to which the information is known outside of his business;
> (2) the extent to which it is known by employees and others involved in his business;
> (3) the extent of measures taken by him to guard the secrecy of the information;
> (4) the value of the information to him and to his competitors;
> (5) the amount of effort or money expended by him in developing the information;
> (6) the ease or difficulty with which the information could be properly acquired or duplicated by others.

Many courts have expressly adopted these factors, some courts also have required the purported trade secret to have an element of novelty (26).

These criteria are examined in greater detail below. It should be noted that where computer software qualifies as a trade

secret, then protection extends to the computer program itself, to any documentation concerning the program, and even to formulae and algorithms that are incorporated into the program.

1. Secrecy

The virtual monopoly that trade secret protection provides to the owner of qualified software must be guarded quite carefully. Secrecy is by far the most important element of trade secret status. All protection is lost if the proprietary software is accidentally or illegally disclosed. This is not to say that the owner has to keep the secret absolutely; he may market the software as long as he takes reasonable steps to maintain its confidentiality (27).

Data General Corp. v. Digital Computer Controls, Inc. demonstrates that trade secret protection is not lost as easy as some people might suppose (28). Data General at one time had distributed over eighty copies of a maintenance manual which contained trade secrets regarding the construction of its computers. These manuals were seen (or might have been seen) by more than 6000 persons. Data General was nevertheless able to enforce its trade secret rights, despite the broad potential dissemination, because it had taken steps

to preserve its rights, such as placing proprietary legends on the documents, and requiring its employees and customers to agree to confidentiality restrictions.

However, software must be kept relatively secret to be protected. The software market is a consumer market in which a large volume of programs are sold through mail business and retail stores. Maintaining a secret is therefore especially difficult, and expensive. Consequently, the software company should first institute procedures and take reasonable steps to protect the secret at the site of creation and marketing (29). The most common form of contractual software protection is the restrictive licence agreement, which restricts the licencee's use of the software and prohibits disclosure to third parties. Other standard procedures include nondisclosure agreements with employees, proprietary and confidentiality legends on all materials embodying the secrets, and restricted access to locations containing the materials or computer systems containing the software (30). If a trade secret is illegally obtained by industrial espionage, or is disclosed by a present or former employee in breach of a confidentiality agreement, anyone is then free to use the trade secret (31).

2. Novelty

Software which contains novel elements (for example, allowing new applications to be processed by computers) seems clearly to fall within the standard definition of a trade secret (32). "Novelty", in the sense in which courts use it, does not mean that the information must be new, or represent a significant advance over prior knowledge in the industry (33). All that is required is that the information or knowledge represent, in some considerable degree, the independent efforts of its claimant (34).

Trade secret novelty is merely intended to insure that the information is capable of being kept secret. Although matters of general knowledge in the industry are not protected by trade secret law (35), particular combinations of generally known concepts may be protected (36). Most computer programs fall under the particular combination theory, because although similiar programs may use the same algorithms and logic, they combine those algorithms and logic differently to produce programs of different speed, accuracy, cost, and commercial feasibility. The combination which produces the best speed, accuracy and commercial feasibility is considered to be sufficiently outside the general knowledge of the industry to be protectible (37).

B. Scope of Trade Secret Protection

Once protection attaches, it lasts for as long the secret is not discovered by legitimate means (38). The owner is protected against misappropriation of his secret through improper conduct. He is protected against:

(1) discovery of the secret by improper means;

(2) disclosure or use of the secret which constitutes a breach of an express or implied duty of confidence;

(3) use of the secret by one who learned the secret from a third person with notice that it was a secret and that the third person discovered it by improper means, or that the third person's disclosure of it was a breach of confidence;

(4) use of the secret by one who learned of the secret with notice that it was a secret and its disclosure was made to him by mistake; and

(5) use of the secret by one who learned of the secret without notice that it was a secret and that disclosure to him was a breach of confidence, or who learned of the secret through a mistake without notice of the secrecy and mistake, after he receives notice of such facts (39).

Consequently, any person who independently creates the same or similiar software is free to use it, since trade secret law only provides legal protection against someone who illegally obtains the proprietary information.

Even when the information is discovered by improper means and is therefore no longer secret from all competitors, the original holder is entitled to damages and protection from use by the improper discoverers for at least as long as it would have taken a competitor to discover the secret through proper means.

Since trade secret protection requires a limited and strictly controlled use of the protected software, trade secret protection is not in every case a very effective and sufficient method of protecting.

II. Copyright Law

A. History

The Copyright Act of 1909 was quite different from the Act of 1976 currently in force. The 1909 Act required publication in order for a work to receive statutory protection; expressions were also required to be directly readable by a person to qualify for protection (40). The human-readable form requirement was eliminated by the 1976 Act. While the 1976 Copyright Act, as originally enacted, did not specifically provide for copyright protection of computer software, its legislative history suggests that Congress considered computer programs copyrightable as "literary works" (41).

In Tandy Corp. v. Personal Micro Computers,Inc. the Court examined the legislative history and concluded that the 1976 Copyright Act extended federal copyright protection to computer programs (42). The Court decided that a computer program clearly satisfies the statutory requirements of "authorship" and "fixation".

In 1974, Congress set up the National Commission on New Technological Uses of Copyright Works (CONTU), whose mandate was to consider software and technology problems in the context of the Copyright Act. The several conclusions that

resulted were embodied in the 1978 CONTU Final Report and were, finally, considered in the 1980 Amendment.

It is undisputed that the 1976 Copyright Act, as amended by the Computer Software Act of 1980, extends copyright protection to software (43).The Computer Software Act of 1980 added a definition of what constitutes a computer program, and created a special exception for computer programs to the normal proscriptions against copying of copyrighted works. The 1980 Amendment was made in accordance with the recommendations of CONTU. The Amendments and the most important sections relating to the copyrightability of computer software are as follows:

Section 101 (Definitions):
A "computer program" is a set of statement of instructions to be used directly or indirectly in a computer in order to bring about a certain result.
A "copy" is defined to include a material object in which a work is fixed by any method now known or later developed, and from which the work can be perceived , reproduced, or otherwise communicated, either directly or with the aid of a machine or device.

Section 102 (b):
In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle or discovery, regardless of the form in which it is described, explained, illustrated or embodied in such work.

Section 117:
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 adaption of that computer program provided:
1. That such a new copy or adaption 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 a new copy or adaption is for archival purposes only and that all archival copies are destroyed in the event the continued position of the program should cease to be rightful.
Any exact copies prepared in accordance with the provision 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 transfer of all rights in the program. Adaptions so prepared may be transferred only with the authorization of the copyright owner.

The one change made by Congress to the CONTU recommendations was the substitution of the term "owner" in s. 117 for "rightful possessor". Congress gave no official explanation for this change, but it has been reported that Congress was concerned that otherwise s. 117 would not allow licensing of software (44).

Although it is undisputed that the Copyright Act extends copyright protection for softwarein general, there is some question whether this protection applies to all forms of software (45).

B. Elements Necessary for Copyrightability

Copyright protection attaches to works of authorship which comprise copyrightable subject matter, are originally created by the author, and are fixed in a tangible medium of expression (46).

1. Copyrightable Subject Matter

In order to be copyrightable subject matter, a work must be a "work of authorship". A computer program is considered a literary work of authorship to which copyright attaches (47). Only the expression of ideas in a work constitutes copyrightable subject matter. Copyright does not protect actual ideas, procedures, processes, systems, methods of operation, concepts, principles, discoveries, or utilitarian aspects of a work; it only protects the specific manner in which they are expressed.

2. Originality

A second requirement for copyright protection is originality. In order to satisfy this criterion the author must create the work independently, without copying from another work. The created work need not be different from previously existing works, so long as an independently created work expresses ideas which can be expressed in several ways, the originality requirement for copyright protection is satisfied.

3. Fixation

Finally, in order to be eligible for copyright protection, a work must be fixed in a tangible medium of expression. The medium may be known at the time a work is created or may be subsequently discovered, provided that it is one from which a work can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device (48). A work is "fixed" when its embodiment in a copy is sufficiently permanent or stable to permit it to be perceived, reproduced, or otherwise communicated for a period of more than transitory duration (49).

4. Procedural Prerequisites to Copyright Protection

A plaintiff attempting to prove copyright infringement must meet certain procedural requirements before proceeding to the necessary substantive elements. The plaintiff must have registered the work in the Copyright Office, or at least have attempted to register it and have been refused (50).

After the enactment of the 1976 Act, the requirement that a copy of the full text of the source code must be deposited with the Copyright Office was dropped. At present, the norm under the regulations is that the first and last twenty-five pages of the source code must be deposited with the Copyright Office (51). The Copyright Office prefers the deposit of a computer program's source code, but will accept object code under the so-called "rule of doubt". The doubt stated is not whether the program is within the subject matter of copyright, but rather that, since the Copyright Office examiners are not expert computer programmers, they have extreme difficulty determining whether the deposit actually contains copyrightable authorship, if the computer program is deposited in a format other than source code (52). Where the applicant is unable or unwilling to deposit a printout in source format, the Office proceeds with registration under their "rule of doubt", upon receipt of a letter from the applicant assuming that the work as deposited contains copyrightable authorship.

The purpose of the deposit is to allow the Copyright Office to recognize the computer program as a copyrightable work written by the applying author and to make the deposit available for public inspection.

The limited deposit of the source code has been criticized as not being a meaningful public disclosure of the copyrighted object code (53). Since the full text of a source code may amount to several thousand pages, the first and last pages are very likely to be "comments" which are not actually part of the program instructions. Therefore, the practise of the Copyright Office makes it very easy for someone to retain the secrecy of his source code. This is contrary to the general policy of copyright law (54), which is to make available to the public any published work which has been deposited (55).

C. Case Law

There has been some confusion as to whether machine language, or object code, is a form of expression covered by the Copyright Act.

1. Data Cash Systems, Inc. v. JS & A Group

The copyrightability of object code was first considered in Data Cash Systems, Inc. v. JS & A Group (56).

The District Court found that the object code form of a computer program stored in a ROM was not a "copy" in the purposes of the 1909 Copyright Act, because object code could not be seen and read by humans, and because it was simply a "mechanical tool or a machine part" (57). The Court stated that even if the 1976 Copyright Act had applied, copying a ROM would not be actionable (58). The Court also posited that the Copyright Act of 1976 covers computer programs in flowchart, source, and assembly phases, but not in their object code phase. The Court explained that the object code, when stored in a ROM, is a mechanical device which is outside the scope of copyright law. The Court of Appeal affirmed The District Court's decision without ruling on the question whether object code in a ROM was copyrightable (the ROM in question did not

contain the requisite statutory notice).

2. Tandy Corp. v. Personal Micro Computers, Inc.

Given the same issue and similar facts as the Court in Data Cash Systems, Inc. v. JS & A Group, the District Court in Tandy Corp. v. Personal Micro Computers, Inc., held that copyright law does protect a program fixed in the form of a ROM chip (59). Applying the Copyright Act of 1976, ss. 101 and 102, the Court held that a computer program is a work of authorship subject to copyright protection, and that a silicon chip is a "tangible medium of expression". Thus, a program fixed in such a form is subject to copyright laws (60). In the result it was held that the practise of unloading and reloading a program from a ROM chip is an infringement of a copyright in a source program. The Court pointed out that the amended s. 117 was not intended to provide a loophole which would allow a computer program fixed on a silicon chip to be duplicated.

3. Williams Electronics, Inc. v. Artic International, Inc.

The conclusion in Data Cash, supra, conflicts with the s. 101 definition of a "copy", and was specifically rejected in Williams Electronics, Inc. v. Artic International, Inc. (61). The Court in Williams held that a copy is a material object in which the copyrighted work is fixed, and that no further requirement exists that the copy perform some communicative role (62).

The defendant had argued that a distinction should be drawn between the "source code" version of a computer program and the "object code" stage. It was suggested that the former could be copyrighted whereas the latter could not be protected since a computer program stored in a ROM would not satisfy the statutory requirement of fixation. It was argued that because a "copy" must be intelligible to human beings and intended as a medium of communication to human beings, the "object code" did not qualify for copyright protection.

The Third Circuit Court rejected the defendant's arguments by interpreting the definition of a "copy" expansively. The Court ruled that it was not logical that an infringement action would apply to copying of the computer program text, but not to the duplication of a computer program fixed on a silicon chip. Additionally, the Court referred to the conclusion reached in Tandy Corp., supra, albeit in the context of

computers rather than video games (63).

4. Apple Computer v. Franklin Computer

The first appellate opinion to deal with the copyrightability of software in a comprehensive manner was the decision in Apple Computer, Inc. v. Franklin Computer Corp. (64).

In the early-1980s Franklin Computer, a challenger to the relatively well-established Apple Computer Company, sold an "Apple-compatible" computer. The point of marketing such a product was to allow buyers of the Franklin Ace 100 model to use software written for the popular Apple II computer. To make this possible, Franklin employees made copies of several of Apple's operating system programs. Apple filed a suit in Federal Court alleging in part that Franklin infringed its copyrights in these programs.

The District Court denied Apple's motion for a preliminary injunction. On appeal to the Court of Appeals for the Third Circuit, two major issues emerged.

(a) Copyrightability of Object Code

Since computer programs are routinely translated from source code into object code, copyrightability of source code alone does not offer sufficient protection against unauthorized duplication of programs. Franklin's two major arguments against copyrightability of object code were the fact that it is unintelligible to humans, and that it has a utilitarian purpose.

The source of the intelligibility argument was the decision in White - Smith Music Publishing Co. v. Apollo (65). The Court in Apple Computer rejected this argument, however, on the basis that the legislative history of the 1976 Act demonstrated that the Act "was intended to obliterate distinctions engendered by White - Smith" (66).
In addition, it was held that there was a logical inconsistency between the intelligibility argument and s. 101 of the Copyright Act of 1976, as amended in 1980. The definition of computer program in s. 101 refers to "instructions to be used directly or indirectly" in a computer. As source code instructions must be translated into object code before the computer can act upon them, only instructions expressed in object code can be used "directly" by the computer (67).

The defendant's second argument, the utilitarian purpose of the object code, was also rejected by the Court. The Court reasoned by analogy, using the example of an instruction booklet for a complex machine. Since such a booklet does something useful (i.e., explains how to operate the machine) but receives copyright protection none the less, the Court concluded that utility was not a bar to copyright protection (68). Therefore, it was held that since object code is just another means of expression, it could receive protection notwithstanding its utility.

(b) Copyrightability of Operating Systems

Apart from the copyrightability of object code, the Court considered Franklin's contention that copyright law could not protect operating system software.

The Franklin argument drew a distinction between application programs and operating systems. It was conceded that the former were copyrightable, but it was argued that the latter were not (69). Franklin underlined this statement by citing Baker v. Selden (70). Selden was the author of a copyrighted book about a novel bookkeeping system. The book contained an explanatory text and some sample ledger sheets. Baker's book on the same bookkeeping system included very similar sample

ledger sheets, but used somewhat different column headings and rearranged columns. Selden's heirs claimed infringement of the copyright. The United States Supreme Court held that copyright infringement had not occured.

In interpreting the "original work of authorship" standard, the Court significantly limited the scope of copyright law's protection (71). It was held that copyright law did not provide protection against the appropriation of an idea or process utilized in an author's work, but rather protected only the particular expression adopted by the author to convey this idea or process. Thus, a book containing explanations and examples of an author's independently developed accounting procedure was copyrightable, but the copyright did not preclude others from publishing substantially different explanations and examples which utilized the author's procedure (72). Essentially, copyright prevents misappropriation of an idea's particular expression rather than the use of the idea itself.

This qualification has become significant in operating system infringenment disputes. Franklin contended that Apple's operating system programs were not copyrightable under s. 102 (b) of the 1976 Copyright Act, which codified Baker v. Selden. Section 102 (b) prohibits copyright protection for any "process, system or operation". Franklin argued that operating system programs are part of the "process, system, or

method of operation" of computer programs, and did not as such qualify for copyright protection.

The Court in Apple Computers refused to adopt this expansive reading of s. 102 (b), but held instead that Franklin could re-create, but not copy, Apple's operating system programs. It was stated that although the programs' method of operation was not protected, the instructions used as a particular expression of this method could be protected (73).

Franklin's argument that Apple's operating system programs monopolized the ideas on which they were based was dismissed by the Court, which formulated a "test of copyrightability" for such situations. Simply stated, the test asks whether other programs can be written or created that perform the same function as the program in question (74). If other programs are possible, then the program in question is an expression of an idea and therefore copyrightable. However, if other programs are not possible, then copyright protection is unavailable (75).

D. Protection for Algorithms

No judicial decision has yet addressed the question whether algorithms (the specific steps a computer follows in performing an assigned task) are protected by copyright law. The answer depends primarily on future interpretations of s. 102 (b) of the Copyright Act (76). Algorithms are not readily comparable to either the idea or expression side of the copyright law distinction. On one hand an algorithm is more than a mere idea because it involves developed thought; on the other hand, it differs from an expression because it is a technique that can be adopted and used in other programs (77).

As shown above, it was ruled that where an idea can be expressed in only one way, the expression of that idea cannot be protected, unless the idea and its expression are separable. Thus, protection of the expressionis not possible where it would, in effect, monopolize the idea.

Although most algorithms are more complex than the simple expression usually effected by the "expression/idea unity" rule, the rule extends to algorithms because the idea of the algorithm is indivisible from its expression (78). Algorithms may also be precluded from protection, as constituting methods, processes, or procedures as these terms are used in s. 102 (b).

Additionally, the legislative history, combined with the tone of the statute, reflects a fear of over-protection. Thus, the House Reports states:

"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 copyright law" (79).

For these reasons, one cannot expect that the courts will protect algorithms under the Copyright Act when a case comes up for consideration.

E. The Semiconductor Chip Protection Act of 1984

In 1984 Congress passed the Semiconductor Chip Protection Act in response to the demand of the semiconductor industry for special legal protection of innovation (80). Specifically, the Act provides in s. 905 for a ten-year term of protection from unauthorized reproduction, importation, or distribution of an original mask work, defined in s. 901:

A "mask work" is a series of related images, however fixed or enclosed
(a) having or representing the predetermined, threedimensional pattern of metallic, insulating, or semiconductor material present or removed from the layers of a semiconductor chip product; and
(b) in which series the relation of the images to one another

is that each image has the pattern of the surface of one form of the semiconductor chip product.

The right dates from design registration with the Register of Copyrights (81) or from the date on which the design is first commercially exploited, whichever occurs first (82).

Enforcement procedures and remedies are modelled on those of traditional Copyright Law. The chip monopoly, however, is limited. The Act prohibits protection for chip designs that consist of stable, familiar, or common-place designs, or a mask work that is not original (83). The Act allows reverse engineering of the mask work for analysis solely for the purpose of teaching, analyzing, or evaluating the concepts or techniques embodied in the mask works and further allows the results of this analysis to be used in designing a new mask work (84). Consequently, the Act permits the use of concepts and techniques from reverse engineered chips to be incorporated into new designs, but prohibits the exact replication of the original chip. Additionally, an innocent purchaser of an infringing semiconductor chip product is not liable with respect to the importation or distribution of the infringing semiconductor chip product that occurs before he has notice of protection; but is liable for a reasonable royalty after having notice of protection with respect to the mask work embodied in the semiconductor chip product (85).

Since the Act does not extend traditional copyright protection to chips, but places chip production under a new chapter of

the copyright statute, the United States are relieved of any obligation under the UCC (Universal Copyright Convention) to grant equivalent protection to foreign chips. Nevertheless, protection for foreign chips is granted, if the foreign nation extends protection to mask works of owners who are nationals or domiciliaries of the U.S., upon substantially the same basis as that on which the foreign nation extends protection to mask works of its own nationals and domiciliaries (86). Apart from that provision, protection for a foreign mask work is always available if the mask work is first commercially exploited in the U.S..

Whether or not the Semiconductor Chip Protection Act of 1984 is a model for new intellectual property laws, covering technological innovations, it is the first one in the world (87).

F. Conclusion

To evaluate or criticize the latest development of copyright protection for computer software in the U.S., it is necessary to examine carefully the copyright law. The subject matter covered by copyright law is described in s. 102 of the Copyright Act. Section 102 requires an original work of authorship and lists seven illustrative categories of

authorship including "literary works".

"'Literary works' are works...expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects...in which they are embodied" (88). The authorship in software is embodied in the original written computer instructions and consists of the symbolic manipulation of letters, numbers, and other symbols; therefore, this authorship falls within the definition.

An argument that has been asserted against copyright protection for certain programs, such as operating programs, is that they are a process, system or method of operation that is barred from copyright protection by s. 102 (b) (89). This argument relies on the principle of Baker v. Selden (90) which is codified in s. 102 (b). This argument shows a misunderstanding of the purpose and effect of the section. The purpose of s. 102 (b) is to distinguish copyrightability from patentability, that means that copyright protection does not extend to the underlying idea of the work of authorship (91). Consequently, the fact that a computer program represents or embodies a method is irrelevant, because only the form of expressing the method (and not the method itself) is within the scope of copyright protection.

Once it has been initially established that a computer program is an original work of authorship, s. 102 requires the work to be fixed in a tangible medium of expression, which is defined

to include any material object that permits the work of authorship to be perceived or communicated with or without the aid of a machine or device for more than a transitory duration.

There is a dispute whether object code embedded in a ROM is copyrightable, because it is argued that a ROM should be considered as a machine part (92). A ROM from the outside looks like any other chip in a computer, so that it is possible to argue that a ROM should be considered part of the hard-wiring of the computer. This approach leads in the wrong direction. The definition above shows clearly that the tangible medium of expression does not matter, so long as the authorship is at least reproducible from it with the aid of a machine or device. ROM is merely a memory device which means that the information in memory is reproducible with the aid of a computer, even if it consists of software (93). In none of the cases do the technological steps make the ROMs or diskettes so "hard-wired" that the software is not reproducible from them, otherwise the computer itself would be unable to read its own software.

It is essential that the medium itself is not copyrightable, and that utilization of a ROM in a computer does not render the ROM copyrightable. It always depends on the message whether the ROM is a utilitarian, mechanical part of the computer hardware, without any programed software instructions, and then void of any copyrightable "authorship"

characteristics.

An open question, which has to be answered by the courts in the future remains where to draw the line between a copyrightable computer program and a patentable utilitarian device.

Opponents of the ROM-embedded object code copyrightability also argue that the purpose of copyrightable work's creation must be direct communication with humans (94). Because ROM-embedded object code directly instructs and manages the internal hardware functions and facilitates source code operation, it is asserted that ROM-embedded object code lacks the requisite communicative purpose. It is maintained that ROM is an unintelligible medium, and is created for the purpose of communicating directly with the hardware rather than with the user (95). Both the language and the legislative history (96) of the 1976 Act indicate that the communicative requirement should not be interpreted to mandate direct communication with the user as a prerequisite to copyrightability. The statute itself allows object code programs to be communicated directly or indirectly with the aid of a machine or device.

Thus, under American copyright law all types of software (operating system or application), in all languages (high level or machine), in all codes (source or object), and in all forms (written, printed, in ROM) are copyrightable. Attempts

to distinguish a copyrightable source code program from its object code counterpart contradict the plain meaning and legislative history of the current Copyright Act; object code is copyrightable subject matter.

Nevertheless, the federal judiciary should develop meaningful guidelines for distinguishing between a copyrightable ROM-embedded object code from an uncopyrightable ROM-resident hardware circuitry pattern.

III. Interaction between Copyright and Trade Secret Protection

Another approach to computer software protection is the concurrent use of trade secret and copyright law. Because copyright protects only the specific logic and design of the program, the legal protection form most widely utilized by the software industry has been trade secret law. Trade secret law protects the unique logic and coherence of the program as well as the underlying programming techniques, routines and algorithms of the program, the input and output formats of the program and the ways in which the program interfaces with other parts of the computer. Protection under this branch of the law is perpetual, being terminable only by discovery of the secret by others; whereas the duration of copyright is statutorily limited to the author's life plus fifty years (97).

The system of intellectual property law in force in the United States seems to rest upon the assumption that all intellectual property will fit neatly within a particular category. Thus, obtaining one type of protection will often preclude recourse to another type of protection. For this reason, the compatibility of copyright law with trade secret protection must be examined.

When the Copyright Act of 1976 was passed, there was some question as to whether s. 301 preempted trade secret protection for items which were granted copyright protection (98). Section 301 states that copyright protection pre-empts all state law remedies which are "equivalent" to the exclusive rights given to a copyright holder in a copyrightable subject matter. The problem was whether the rights protected by copyright were equivalent to the rights protected by trade secret.

In Videotronics, Inc. v. Bend Electronics the Court held that trade secret protection was unavailable for computer software that had been made readily available to the public (99). The Court noted that once the property interest in the software was covered by the Copyright Act of 1976, relief was no longer available under state trade secret law. This is not to say that the Act pre-empted state trade secret law. Both the House of Representatives Report and the Final Report of the National Commission on New Technological Uses of Copyrighted Works concluded that section 301 does not pre-empt trade secret protection (100).

In Technicon Medical Syss., Inc. v. Green Bay Packaging, Inc. the Court recognized that copyright law seeks to protect "the form of the work", while trade secret law seeks to protect "contents or idea in a work" (101). The Court stated that, to a certain degree, the two respective rights in intellectual

property interact. Thus, to the extent that a work has been copyrighted and published, the chances of unprivileged disclosure may increase. But the mere fact that an expression is copyrighted does not, in and of itself, disclose the trade secret or eliminate its mantel of confidentiality. Since trade secret and copyright protect different things, they are not equivalent. Therefore, s. 301 does not pre-empt trade secret protection.

Even if copyright protection does not pre-empt state trade secret protection, copyright registration can create problems for the protection of software under trade secret law. Registration is a prerequisite to bring an infringement action, but this may result in disclosure of the trade secret. As pointed out earlier, a way to retain secrecy in spite of depositing the work in the Copyright Office is either to deposit the object code rather than the source code (102), or to apply for "special relief". This means of retaining secrecy results in the source code being inspected for evidence of authorship, and then returned except for a minimal identifying portion (103).

In connection with the application for special relief, another approach to retaining secrecy has been suggested. It is based on s. 411 (a) of the 1976 Act, which allows an applicant refused registration after complying with all formalities to institute an infringement action. Usually, registration is a

prerequisite to institute such an action, but the approach is to apply for registration with special relief and, if special relief is refused, gain return of all deposits, thereby allowing secrecy to be retained. When an infringement action is later instituted, one can claim jurisdiction despite lack of registration, on the basis of s. 411 (a).

As the study showed, the major disadvantage of trade secrecy as a protective device arises in its application to a mass-market product. The software owner may find that he must expend large sums of money and conduct business in an encumbered manner to maintain his trade secret. In contrast, copyright law as a proprietary device is neither expensive nor difficult to maintain. Therefore, concurrent use of trade secret and copyright protection is advisable where possible. Nevertheless, the utility of this combined approach will most likely be limited to proprietary software which is licensed on a confidential basis to a limited number of customers(104).

IV. Patent Protection

A. Historical Perspective and Statutory Requirements

The principle forums for the development of the law on the patentability of programming are the United States Patent and Trademark Office (USPTO), the Court of Customs and Patent Appeals (CCPA), and the United States Supreme Court (105).

Traditionally, patent law has been invoked to protect technological inventions. Under the Patent Act, creators of inventions are given a seventeen year period of exclusive use, which protects against independent creation, use, or sale of the invention or process by all others in the United States not party to an express licensing agreement with the patent holder (106).

Sections 101-3 of the Patent Act outline three requirements which must be met by any invention in order to obtain patent protection. The invention must fit into one of the statutory subject matter classifications (107), it must be novel (108), and, it must be non-obvious to one skilled in the art (109). While a mathematical or functional principle incorporated in a program may satisfy these requirements, courts have held that the software itself is patentable only if the manner in which

the principle is utilized is also new and useful (110). As applied by most federal courts, this criterion has operated to disqualify most claims for software patentability (111).

One could argue that patent law protects only those programs which are integral parts of inventions of matter-transforming processes, and excludes from protection the logic number of programs which simply calculate, keep records, or otherwise manage information (112). To evaluate whether patent protection deals with all aspects of mass-products like computer software requires a closer look at the statutory requirements.

1. Patentable Subject Matter

Section 101 of the Patent Act describes the subject matter protected by the Act:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefore, subject to the conditions and requirements of this title.

Attempts to patent software usually try to characterize it as a process or as a part of a new machine (113). All of the categories of statutory subject matter have been interpreted to exclude scientific principles, laws of nature, mathematical

formulas, and methods of calculation (114). Most of the attempts to patent computer-related inventions have centered around the presence and effect of a mathematical algorithm implemented by a computer through a computer program.

(a) Gottschalk v. Benson

In a 1972 case, Gottschalk v. Benson, the U.S. Supreme Court held that a patent claim relating to a simple mathematical formula without substantial practical application, except in a connection with digital computers, was not patentable subject matter (115). The claim in question related to a method for converting one type of internal representation of numbers to another. The theoretical background for this decision was the reference to the "mental processes doctrine", utilized by the USPTO, and first considered by the CCPA in In re Prater (116). The "mental processes doctrine", closely related to the principle that processes of nature are not patentable, specifies that patent claims which rely on mathematical formulae and expressions which are inherently not subject to patent are unpatentable. Computer programs which derive their utility from mathematical algorithms in general, and which are not confined or limited to a particular apparatus or end use, are particularly unlikely to be granted patent protection.

Prior to Benson v. Gottschalk, the CCPA had upheld the patentability of computer programs. The following statement appears in the Prater opinion, for example: "No reason is now apparent to us why, based on the Constitution, statute or case law, apparatus and process claims broad enough to encompass the operation of a programmed general-purpose digital computer are necessarily unpatentable" (117).

In a later case, In re Musgrave, the CCPA again found that programs were patentable subject matter (118). It was held in the Musgrave decision that method or process claims which could be performed on a programmable computer, or which were otherwise in the "technological arts" because they were not "purely mental", were patentable subject matter.

The Supreme Court decision in Gottschalk v. Benson appears to be a strong repudiation of the CCPA's "technological arts test". However, it is noteworthy that in Gottschalk the Court expressly restricted its holding to the particular facts of the case, and avoided stating a general rule.

(b) Parker v. Flook

In Parker v. Flook, the Supreme Court addressed the issue of patentability of a method for updating alarm limits (119). An alarm limit was disclosed in the patent application as a parameter in a catalytic hydrocarbon conversion process. The Supreme Court held (6:3) that Flook's process was unpatentable under s. 101: not because it contained a mathematical algorithm as one component, but because "once that algorithm is assumed to be within the prior art, the application considered as a whole, contains no patentable invention". Furthermore, the Supreme Court referred to the CCPA by stating that "if a claim is directed essentially to a method of calculating, using a mathematical formula, even if the solution is for a specific purpose, the claimed method is nonstatutory" (120). The Court rejected Flook's argument that the presence of a special 'post-solution' activity --the adjustment of an alarm limit to the figure computed according to the formula-- distinguishes the case from Benson, making the process patentable.

The Supreme Court's test in Flook "that (the) algorithm is assumed to be within the prior art" provided a basis for the USPTO to find most forms of programming unpatentable subject matter, and was soundly criticized with considerations of statutory subject matter (121).

(c) Diamond v. Diehr

In Diamond v. Diehr, Diehr applied for a patent on a process for molding rubber which ensured that the rubber would always be perfectly cured (122). A computer, operated by a program, took constant measurements of the temperature of the mold, and applied a known formula which constantly recalculated the remaining cure time based on changes in temperature during the elapsed cure time. When the computer calculated that the remaining cure time was zero, it automatically opened the mold.

Diehr did not attempt to claim patent protection for the formula itself, but claimed only the improved process for curing rubber (123). In an attempt to clarify the elements delineating patentable and unpatentable subject matter, the U.S. Supreme Court stated that the claim in question must initially be examined to ascertain whether it contained a mathematical formula. Then it must be determined whether the claim sought patent protection for that formula in the abstract (because protection is not available for a mathematical formula, scientific principle, or law of nature). The Court explained that unpatentable subject matter cannot be converted to patentable subject matter merely by limiting the use of the formula to a particular technological environment or by adding insignificant post-solution activity.

Since it was attempted in Diehr to patent only a total process, and neither the formula itself nor a method of calculation, the Court held that the presence of the formula, and the computer solution of it, did not destroy the statutory subject matter of the process as a whole (124).

(d) Post Diehr Decisions

After the Supreme Court decision in Diehr, the CCPA decided four cases dealing with the patentability of computer-related inventions: In re Taner (125); In re Abele (126); In re Pardo (127); and, In re Meyer (128).

In those cases the CCPA followed the two-step analysis which was originally set forth in In re Freeman (129), as modified by the Supreme Court in Diehr. Taken as a group, these decisions refined the two-step analysis into its present form. Under the present analysis, the claim in question must first be evaluated to determine whether it includes a mathematical algorithm, or whether it encompasses an application of the algorithm that includes statutory subject matter. In the former case the claim is not patentable, whereas in the latter it is. It should be noted, though, that for patentability to exist in the latter case, the statutory subject matter must be something other than a field-of-use limitation or a non-essential post-solution activity. The effect of these

rules is that, for the first time, many areas of programming are established as being clearly patentable subject matter. The previous uncertainty about patentability has been removed for most types of programming (130).

These judicial decisions have prompted a number of critical commentaries concerning patentable subject matter. Some commentators have proposed alterate criteria for evaluating the patentability of algorithms. For example, a court could require simply that the input and output of the device or program be concrete enough for the method by which the input data is to be determined, or the way in which the output is to be used, to be specified. This proposal was made by Michael Gemignani, who believes that the distinction the Supreme Court has drawn between patentable processes and mathematical algorithms is confusing and unnecessary (131). He has stated that, if the application and the input/output specifications are drawn sufficiently sharply, the dangers inherent in the grant of an excessively broad monopoly are averted.

Applied to the cases in question, one gets the result that Benson's invention was the most obscure in this regard, citing no application to which his process must specifically attach or to which it was restricted. Flook's invention was constrained to a particular context, catalytic conversion, as was Diehr's, the production of precision rubber products. However, there was a distinct degree in the specificity of

input and output, between the inventions in Flook (no indication of how to get specific data or how to use output) and Diehr (raw rubber in; cured product out) (132). The implementation of Gemignani's proposal would not overturn any of the cases the Supreme Court has decided; but would take care that the owner of computer software would be provided with more clarity, concerning patent protection.

2. Novelty

In addition to falling within one of the categories of statutory subject matter, an invention must also be novel to qualify for a patent (133). An earlier invention need not be identical to the applicant's invention to anticipate it. If there is an invention created before the applicant's invention that would have violated a patent granted on the applicant's invention if it had been invented later, the earlier invention anticipates the applicant's invention (134). Concepts and points of similarity from several prior inventions cannot be combined to anticipate the applicant's invention. This anticipation must relate to the whole invention claimed, not merely to parts of it, to destroy the element of novelty (135).

3. Nonobviousness

Although earlier inventions by others cannot be combined to destroy the novelty of the applicant's invention, earlier discoveries can be combined and viewed as a mosaic to defeat the applicant's claim that his invention is not obvious. A requirement of nonobviousness is prescribed by s. 103 of the Patent Act. An invention may not be patented "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 said subject matter pertains".

Since most computer programs result from the application of combinations of generally known programming skills and/or general known data processing concepts, the requirement of nonobviousness will preclude patent protection for the majority of programs.

B. Scope and Enforcement

Obtaining a patent is generally expensive and requires a lengthy period of time (typically several years). The delay seriously limits the usefulness of patent protection, since in the high technology area of computers, the invention's commercial life may have already expired by the time patent is granted.

Another problem with patent protection of software is the risk of taking an infringement action to enforce the patent, which may result in a judicial determination that the patent is invalid. The percentage of concluded lawsuits for patent infringement in which the patentee has emerged the winner has been low, and continues to decline (136). Moreover, since proof of infringement is so uncertain, taking an action may not be advisable in every case. To prove infringement, the patentee must demonstrate that his invention was made, used, or sold during the term of a patent by one without authority to do so. In the unusual case where software can meet the requirements of statutory subject matter, novelty, and nonobviousness, and where the expected lifespan of the software is relatively long and the value of the software relatively high, patent protection may be worth the time and expense.

All in all, the patent system, as it presently exists, is clearly not the best way to protect innovative developments in computer software.

C. Patent and Trade Secret

Patent and Trade secret cannot coexist for the same material, because the disclosure requirements for patent protection are diametrically opposed to the secrecy requirements for trade secret protection (137). Furthermore, the uncertainty surrounding patent protection for computer software can leave a publisher or programmer without either patent or trade secret protection, after a long and expensive application process in which some or all his secrets disclosed in the application are printed in the public record of the opinions written by the courts when the denial of an application is appealed. Thus, when a patent application is denied, the applicant either forgoes his right to an appeal and give up all hope of ever obtaining a patent, or takes his appeal and hopes that the secrets will not be disclosed in an opinion.

D. Patent and Copyright

Patent and Copyright protection can coexist for software, but the two types of protection protect different parts of the software. Patent protection covers the process which the software contains while copyright protects the expression of the idea embodied in the software. None of the requirements for either type of protection preclude the existence of any of the requirements for the other type of protection. A combination of patent and copyright protection does not give software that is patentable any greater protection than would patent protection alone because, except for the period of protection, patent protection is much broader than copyright protection. On the other hand the chances of winning a copyright infringement action seem to be better, and the expense involved is much less.

V. Conclusion

The tremendous growth of computer technology and the resulting need for software has magnified the importance of intellectual property law for protecting computer software. There are three forms of protection available. While trade secrecy was once the chief means of protecting software, the mass marketing of multiple copies of computer program --especially for the personal computer market-- has rendered trade secret law an ineffective mechanism for protecting most software.

The Copyright Act of 1976, as amended by the Computer Software Act of 1980, is now the chief vehicle for protecting software. If possible, dual protection under both copyright and trade secret law should be utilized to protect software, but even the combination does not provide the software owner with fully comprehensive legal coverage. The software remains susceptible to honest or accidental discovery. The added measure of copyright provides the trade secret holder with relief in the event of accidental injunction of the trade secret into the public domain with attendant loss of trade secret protection. However, the American copyright law does not protect algorithms, and in addition there is an uncertainty concerning the question of where to draw the line between a copyrightable ROM-embedded object code and an uncopyrightable ROM-resident hardware

circuitry pattern.

Patent law, on the other hand, provides questionable means of protecting computer software, because the procedure for obtaining a patent is expensive and time-consuming. Moreover, the distinction the Supreme Court has drawn between patentable processes and mathematical algorithm is not really obvious, and therefore less convincing.

All in all, the protection of computer software under American law is not available in every case, and seems occasionally doubtful.

Chapter 4:

Protection of Computer Software in the Federal Republic of Germany

I. Introduction

Since the 1960s the controversy concerning protection of computer software has been growing in West Germany. Characteristic of the legal debate in Germany is the analysis of computer programs not as an integral whole, but rather in their component aspects.

At the beginning the question turned mainly on the possibilities of protection by patents (138), and even the German Federal Patent Court indicated a preoccupation with this particular form of protection (139). The Amendment of the Patent Act in 1978 has put an end to speculations. Preceded by an European Patent Agreement (140) which was ratified by the Federal Republic of Germany, the Patent Act now expressly excludes the protection of computer programs (141). However, in 1985, the Copyright Act was amended; and it now explicitly includes the protection of computer programs explicitly. Because of the late development in the legislation, all the crucial decisions were rendered before the Amendment was in effect; they deal therefore with the

question of protection of computer programs in a fundamental manner.

II. Patent Protection

Under the German Patent Act, creators of inventions are given a time-limited exclusive right, provided that the invention is true and new (Section 9 and 16 PatG). The German Federal Patent Court first approved a digital computer program in 1970 (142). In 1973 the invention which had just been the subject of patent rejection by the United States Supreme Court in Gottschalk v. Benson (143) was approved in Germany (144). The next year, however, the Federal Patent Court began retreating from this position, and in 1976 the Federal Supreme Court held that programs relying upon mathematical formulae and algorithms are not patentable (145). Since then the German Patent Act has been amended to exclude "programs for data processing installations":

Sec.1 I PatG

Patente werden fuer Erfindungen erteilt, die neu sind, auf einer erfinderischen Taetigkeit beruhen und gewerblich anwendbar sind.

Sec.1 II PatG

Als Erfindungen im Sinne des Absatzes 1 werden insbesondere nicht angesehen:...

3. Plaene, Regeln und Verfahren fuer gedankliche Taetigkeiten, fuer Spiele oder fuer geschaeftliche Taetigkeiten sowie Programme fuer Datenverarbeitungsanlagen.

Section 1 II 3 PatG is applicable if the program in question is an application - program, which means, that it is used by the data processing installation whenever and as often as it

is necessary. However, patent protection is possible if the program, besides containing instructions for the calculations, would also involve the construction of a new data processing system, or would at least use a known arrangement in a completely new way (the criterion of novel usefulness) (146). Consequently, patent protection is reserved for technical solutions of problems in the area of data processing.

III. Copyright Law

A. History

When the German Empire ("Reich") was created in 1871 and Germany thus became one country, copyright was made a matter for the central government and the first German copyright law was promulgated in the same year. It was superseded in 1901 by the Law on Copyright in Literary and Musical Works, and supplemented by a statute relating to publishing law.

On January 1st, 1966, the first comprehensive Copyright Act became effective. The Act was intended to reflect the technical advances made in the area of reproduction, distribution, and formation of words, sounds, and pictures. The 1966 Act did not protect computer programs explicitly, but in general protection was granted under s. 2 I 3 or s. 2 I 7 UrhG to works of literature, and/or scientific or technical representations. Since July 1st, 1985, the Copyright Act protects programs for data processing installations explicitly in s. 2 I 1 UrhG. Section 2 II of the Act provides that the work of authorship has to be an individual intellectual creation.

Section 2 I

Zu den geschuetzten Werken der Literatur, Wissenschaft und Kunst gehoeren insbesondere:
1. Sprachwerke, wie Schriftwerke und Reden, sowie Programme fuer die Datenverarbeitung....
7. Darstellungen wissenschaftlicher oder technischer Art, wie Zeichnungen, Plaene, Karten, Skizzen, Tabellen und plastische Darstellungen.

Section 2 II

Werke im Sinne dieses Gesetzes sind nur persoenliche geistige Schoepfungen.

Prior to the Amendment of s. 2 I in 1985, the first Court to be engaged with the issue of protection of computer software, the Higher District Court of Mannheim (147), held that programs for data processsing installations --even if they fall under s. 2 I UrhG-- never fulfill the requirement of s. 2 II UrhG. The Court stated that a computer program, even if it is created with imagination and uncommon combinations, will always be beyond the sensual perception of a viewer, because of its abstraction and its lack of any aesthetic characteristics. Therefore, it is not surprising that the debate about protection of computer software in Germany is primarily a question of the interpretation of s. 2 II UrhG.

B. Elements

After the Amendment of the Copyright Act in 1985, the debate concerning protection of computer software has changed, from an initial discussion of s. 2 I UrhG to a problem of interpretation concerning the notion of the requirement of individual intellectual creativity in s. 2 II UrhG. The interpretation of s. 2 II UrhG goes back to the very beginning of German jurisdiction. The Supreme Court of the German Reich, the pre-war equivalent of the Federal Supreme Court, interpreted s. 2 II UrhG by using the concept of aesthetics. The Court argued that an individual intellectual creation is a work of authorship that is produced by means of formation of art, and is intended for the stimulation of the aesthetic sensation of the viewer (148).

The Federal Supreme Court (BGH) took over most of the jurisdiction of the Supreme Court of the German Reich. In 1971, the former pointed out that a work of art has to reach such a high standard of aesthetic substance that from an average point of view society, or someone who is familiar with the prevalent evaluation of art, would speak of an artistic performance (149).

Considering the recent jurisprudence of the Federal Supreme Court, one can ascertain that the Court now renounces the

requirement of the aesthetic intellectual substance or the stimulation of aesthetic sensation (150). In BGH BB 1985, 1750 the Court argued that s. 2 II UrhG does not demand an aesthetic substance, in the sense of beauty. At present, the Court rather calls for individual characteristics in the actual result. The evaluation of the required degree of distinguishing characteristics is now judged by the entire impression of that creation, and by an overall comparison to prior creations. It must be possible thereby to recognize individual intellectual creativity. Thus, the author of the work must realize his own ideas in his creation.

In the course of time, legal authorities have developed some indicia to determine when a work of authorship is a protectable subject matter. First of all, the creator of the work has to have some kind of scope, because a dependency on instructions would necessarily exclude a process of creation (151). Secondly, the product should not be earmarked only for a specific purpose. There has to be latitude for individual creativity (152). Another indication is the proof of novelty (153); and finally, an expert should be consulted to evaluate the degree of creativity.

C. Case Law

1. OLG Frankfurt and OLG Karlsruhe

In the summer of 1983, the Higher Regional Court of Frankfurt had to settle three cases concerning the protection of computer software (154). Ruling in the proceeding of June 13th, 1983, the Court determined that, as a rule the starting point in the determination of copyright grants would be the individual impression of the creation of the "frame", disregarding the content of the program (155). The Court argued that the source code, its printed form, or its projection on the screen in particular, is a proper means of determinining the qualification of the selection and arrangement of the used material as an individual intellectual creation.

All cases were essentially based upon similar facts. In the proceeding of June 13th, 1983, both parties were competitors in the market for video games (156). The plaintiff sold a game program called "Pengo", which was copied and sold on the German market by the defendant under the name "Pento". First of all, the Court stated that a computer program is either a work of literature or a scientific or technical representation, and therefore copyrightable subject matter

under s. 2 I UrhG. Furthermore, the Court concluded that it is impossible to evaluate a computer program by looking at its artistic or aesthetic effect, because a computer program could never have an artistic form of representation. The Court expanded upon this conclusion by emphasizing that the crucial point, ie. the external train of thought, is necessarily influenced by the content of the information. But, the content itself is not taken into account for the determination of the work of authorship.

Consequently, one can find individual intellectual creativity in the sense of s. 2 II UrhG in the selection, arrangement, and attachment of existing or new material used to write the software (157). This creativity must always be found in the form of the work of authorship, disregarding whether the different steps of development have been conceptualized or carried out with great intellectual effort.

Ruling in the proceeding on July 21st, 1983, the Court found that in the determination of authorship it is not decisive that various writers of programs would develop different programs, although they all had the same problem to solve (158). Such a requirement would be a limitation to individual performance, and would not take the element of intellectual creative activity into account.

Ruling in the proceeding on August 4th, 1983, the Court added

that the quantity of steps involved for the process of software production is not a criterion for the copyrightability of the software (159). Accordingly, the Court dismissed the action in all three cases on the ground that video games on their own are not protectable under the Copyright Act because the visible game is embodied in the underlying source code, which is already protected by the Copyright Act.

Like the Higher Regional Court of Frankfurt, the Higher Regional Court of Karlsruhe affirmed the fundamental copyrightability of computer programs (160). The issue in that case was a collection program, which condidered the different risks arising during the enforcement of summary proceedings for the recovery of debts or liqiuidated demands (default actions). The crucial point was that the personal data of the creditor and debtor, and the data of the District Court had been combined with a cost-benefit and debtor-structure analysis. That data was used with a program to calculate a variable scale of charges. This particular combination resulted in the Court evaluating the collection program as an individual intellectual creation.

2. OLG Koblenz and LG Muenchen I

The Higher Regional Court of Koblenz, contrary to the Higher Regional Court of Frankfurt and the Higher Regional Court of Karlsruhe, argued that an individual intellectual creation in the sense of s. 2 II UrhG could result from the content of the program, as well as from its external appearance (form) (161). The dispute before the Court was the question of copyrightability of a program carrying information about the construction of buildings. The program made use of about a hundred other computer programs. In determining the copyrightability of the program, the Court looked at its content, and in particular at the selection and structure of the information and the disclosure of the facts.

The reason given by the Court for including the internal elements in its evaluation of the program as an individual intellectual creation was that the Federal Supreme Court had decided that exclusion of content applied only to problems concerning technical drawings. The Court explained furthermore that the reason for so doing, was that technical theories are protected by the Patent Act, but that this argument is not applicable to data processing installments because these are explicitly excluded from patent protection by s. 1 II 3 PatG.

The Regional Court of Muenchen I, also came to the conclusion

that the individual intellectual creation can be found in both the content and in the external form of representation (162). The controversial program was a comprehensive data program with about tenthousand program commands. The Court found that the content is expressed in the formation and direction of thoughts, and/or in the particular and ingenious form the presented material was collected, distributed, and arranged. Without going into details, the Court resolved the question of copyrightability of the particular computer program under discussion in the affirmative. The Court held that every program which is not too simple would show individual characteristics, all the more so as the program increases in complexity. Another reason given by the Court was the plaintiff's reference to newspaper articles in which precisely the high quality and originality of the program was affirmed.

3. BGH

Recently, the Federal Supreme Court settled a case concerning the copyrightability of computer programs (163). Subject matter of that appellate decision was the collection program which had already been brought before the Higher Regional Court of Karlsruhe. Like the lower Courts, the Federal Supreme Court confirmed the copyrightability of computer programs. Nevertheless, in this particular instance, the Federal Supreme

Court remanded the case to the Higher Regional Court of Karlsruhe for the purpose of clarifying evidence.

The first concern of the Federal Supreme Court was the problem of legally classifying the different steps in the creation of a computer program (164). The Court stated that the first step of writing a program is the general solution of the problem, which is based on mathematical premises and logical reasoning. The result is described in a study called a duties record book. This is undoubtly a work of literature, and therefore falls under s. 2 I 1 UrhG.

In the course of the second step, the projected solution of the problem is described in detail, and presented in a graphic called a flow sheet or a data flow plan. The flow sheet is a scientific or technical presentation, and for that reason is protected by s. 2 I 7 UrhG.

The code itself, the source code followed by the object code, is written during the third step. This completed program falls under the protection of s. 2 I 1 UrhG as a work of literature, even if the program is only intelligible to a machine (165).

Additionally, the Federal Supreme Court ascertained that the crucial point for the copyrightability of computer programs and the steps taken in its creation are only the formation and the way of collection, distribution, and arrangement of the used material. Within that area there would be enough latitude

to develop an argument based on individual intellectual creation in the sense of s. 2 II UrhG; and this is so in any or all of the three steps. Thus, the Federal Supreme Court maintained that it is even possible to have an individual intellectual creation within the selection and arrangement of the source code because of a lack of concretness concerning the flow sheet (166).

Finally, the Court affirmed the lower Courts' decisions in saying that s. 2 II UrhG does not require an aesthetic content in order to permit the copyrightability of computer programs. In the abstract then, the question of copyrightability of computer programs is, as a matter of principle, to be answered in the affirmative.

In discussing the particular case before them, the Court found that one can discern characteristics of a work of authorship which satisfy the requirement of s. 2 II UrhG by considering the entire impression of the intellectual creation, and comparing it to prior programs. The result of such a comparison must reveal characteristics specific to the new program. Accordingly, it is not important that the program be of high quantity, or that the process of development has been expensive or time-consuming, or that the problem itself has been a completely new one. In addition, the Court affirmed the Higher Regional Court of Frankfurt in holding that it is insignificant whether different writers of the program,

although they all had to solve the same problem, would construct different programs. The crucial point is that the writer of the program has shown unique efforts in the selection, collection, distribution, and arrangement of the information, as compared with an average performance (167). The Federal Supreme Court retains the principle that it developed concerning technical drawings: namely that the protection of a work of authorship follows from its external formation and representation, and not from its content.

If, in a particular case, one can determine a certain uniqueness only in an early stage of the creation of the program, and not during the subsequent writing of the code, a denial of the copyrightability is not a necessary consequence. As a result of considering a work of authorship as a unity, it is enough to show an individual intellectual creation in the preparatory work, because those precedings are merged in the final work.

D. Protection of Algorithms

Concerning the copyrightability of computer programs, it is of great significance that the calculation itself, and all the underlying mathematical principles such as the algorithm, are not protectable subject matter under the Copyright Act (168).

The reason for this exclusion is that an algorithm is an order to the machine, consequently a principle which is not protectable under the Copyright Act (169). Scientific perceptions have to be accessible to the public; the purpose of Copyright Law is not to grant a monopoly. If the algorithm as a scientific order is not copyrightable, the consequence is that it cannot be taken into account to determine the copyrightability of the program as a whole (170). Nevertheless, it is argued that single elements of the algorithm which are part of the program in question should be taken into consideration for the determination of the individuality of the program, because of their specific fusion with it (171).

Under the Patent Act there is no protection available for an algorithm either. Indeed, though the algorithm can be the foundation of a technical invention there is no protection obtainable independent of the final product or process (172).

E. Scope of Copyright Protection

The copyrightability of a computer program substantiates an exclusive right for the author of the work or his legal successor with regard to its exploitation. The author has the absolute right to exploit the work in its original or in its

treated form, in particular to reproduce and distribute it (ss. 12, 15, 16 and 17 UrhG). As a result of that absolute right, every use of the protected program by a third person without permission of the author or his legal successor is an infringement of the copyright.

Nevertheless, the Copyright Act does not protect against independent reinvention. Although one could argue that there is not as much scope to the creation of computer program as there is to writing a book or to painting, the danger of independent reinvention is no more likely than for any other comparable work of authorship (173). In a case in which the Court determines that a computer program is identical or very similar to one already protected, the burden of proof shifts from the plaintiff to the defendant.

Finally, the Copyright Act permits the distribution and publication of a work which was developed with the aid of someone else's program. However, the German legislation concerned with that problem is very exacting. Thus one has to show individual characteristics of the new program, so that the other elements used will be recognized as merely stimulation.

The right of reproduction allows the owner of the computer program to control all essential possibilities of application and exploitation. Accordingly, the translation of the source code into the object code, the transformation of the source

code or object code from one data carrier to another, as well as the keeping of the computer program in an internal machine-store, falls under the right of reproduction. The same protection applies to a program stored in a ROM (174).

The right of distribution enables the author to publish his work; in particular to exploit the program by renting, confering or licensing.

In contrast to the American Copyright Protection, there are no formalities to be observed, such as registration of the work of authorship; rather the right automatically arises with the invention.

F. Protection of Mask Works

In contrast to the United States, where the Semiconductor Protection Act was already put into force, the discussion about protection of "mask works" in Germany has just begun. In Germany, the question turns on the possibilities of protection of the masks under existing law (175). First of all, one could consider protection under the Utility Models Act (Gebrauchsmustergesetz). Protection under the Utility Models Act is granted on the premises that the mask work is novel and shows a certain extent of inventive activity. Copyright protection is in principle possible for technical drawings and other graphic representations which

arise from the development of the chips and their preceding products, much as data lists are a preceding product of the masks. However, one has to take into consideration that presently there exists no protection against the utilization of masks which are produced with the help of those drawings (176).

All in all, there exists as yet no pertinent legislation in Germany, so that the protection of mask works is largely unsettled.

G. Conclusion

The distinction drawn by the Federal Supreme Court between the external form and the content is not convincing. As one could see by looking at the definitions relating to the individual intellectual creation given by the Higher Regional Court of Frankfurt and the Higher Regional Court of Koblenz, it is impossible to draw a clear distinction between these elements, since they do interact with each other. An idea of the difficulties the Courts have with the determination of the individual intellectual creation can be seen by looking at the judgement of the Regional Court of Muenchen I, which accepted newspaper articles as admissible evidence. Moreover, an author of a work should not be without any protection if

someone else produces a program with the same content but in a different form (177).

The classical notion of what constitutes "creation" as defined by s. 2 II UrhG is in particular questionable when applied to the protection of computer software. Thus, it is characteristic of the writer of a computer program that he has a certain problem to solve. As a rule, his intent is not to realize his own personality --as a musician or a sculptor might do-- but simply to solve the problem in an adequate way. Furthermore, the view of the Federal Supreme Court that various writers need not construct different programs, although they have the same problem to solve, is surprising because that view is equivalent to rejecting the required element of individuality.

The development costs of a computer program increase with its complexity, a notion applicable not only to the form but to the content as well. After all, one of the main elements of the program is a logical arrangement of the commands within it. Therefore the mere protection of the external form of a computer program does not meet the purpose of the Copyright Act, which is to assure the author of the benefits of his work. That does not mean that there should be protection granted for simple programs or algorithms, but rather, that the interests of the author should be taken more into account.

IV. Trade Mark Protection

In addition to the copyright protection, trade mark protection is available in certain cases. A prerequisite to protection is that the program contains a trade mark which can be seen on the screen whenever the program is used. The production and sale of copies without the permission of the owner are infringements of the protected program. The best example is a program running a video game where, as a rule, the trade name is projected on the screen at the beginning (178). Nevertheless, the efficiency of trade mark protection is mainly a question of whether the removal of the trade mark demands a substantial effort.

In addition, the protection of a trade mark is not comparable to copyright protection. Under the Trade Marks Act there is no prohibition to copying programs, which means that trade mark protection does not become effective unless the copy is brought into public. Moreover, there is no bona fide rights protection against third parties, as there would be under the copyright provision.

V. Unfair Competition

Usually, an infringement of the German Unfair Competition Act arises from the use of other people's work without having made personal efforts.

Section 1 UWG

Wer im geschaeftlichen Verkehr zu Zwecken des Wettbewerbs Handlungen vornimmt, die gegen die guten Sitten verstossen, kann auf Unterlassung und Schadensersatz in Anspruch genommen werden.

Section 1 UWG requires a violation of moral principles. As a rule, such violations result from the fact that the original person providing the achievement lost the benefits of his work in an inequitable way (179). Thus, the Higher Regional Court of Frankfurt (180) affirmed the application of s. 1 UWG concerning copies of video games, and the Higher District Court of Mannheim (181) has agreed that the sale of unauthorized copies of programs is a violation of fair trade.

Concerning the protection of computer software under Unfair Competition Act, there is one important point that must be recognized. The Federal Supreme Court has decided that the Unfair Competition Act assures the original producer the exploitation of his work during a limited period of time (182). There is no protection for programs against copying after this limitation period. The Higher Regional Court of

Frankfurt has found that the attraction of video games diminishes after six to twelve months. Consequently, the Court limited unfair competion protection to that period of time.

Like American trade secret protection, the German Unfair Competition Act provides protection against the unauthorized use of trade secrets in s. 17 of that Act. A prerequisite to protection is that the secret is not obvious, and the secret-holder has expressed his will to maintain secrecy. Thus the problem is the same as under American law.

The scope of the unfair competition protection for computer software is not as extensive as copyright protection. This is because, the statutory period of limitation in the former (6 months after knowledge of the unauthorized use; and no longer than 3 years without) is not as long as the one established in the latter (70 years from the death of the author). In addition, there is no right to the destruction of infringing copies as provided by s. 98 II UrhG.

VI. Conclusion

In Germany, general protection of computer software is now established law. Nevertheless, the debate about the classical notion of creation has assumed a new importance, and in the end the protection of computer programs will depend on the result of that academic dispute. Protection under the Trade Marks Act and the Unfair Competition Act can augment copyright protection, but taken on their own, they are ineffective.

Chapter 5:

International Agreements

An important consideration is whether a program will be eligible for protection overseas. The relations of the Federal Republic of Germany and the United States of America, with each other and other countries, are based on bilateral treaties and various international conventions. The two major copyright conventions are the Universal Copyright Convention (UCC) (183), and the Berne Copyright Convention for the Protection of Literary and Artistic Works (BCC) (184). Both ensure crossborder copyright protection among a large number of countries, and include provisions that define a minimum protection either directly under the ratified convention, as in the BCC, or to be established by the local law of the member state, as in the UCC.

I. Universal Copyright Convention (UCC)

Both West Germany and the United States are signatories of the UCC (185). The UCC is the result of a compromise between the European concept of copyright protection and the American view (186). The compromise was reached primarly to include the United States in an international convention on copyright.

Under the UCC, works by nationals of the United States, wherever published, must be granted in each member country at least "the same copyright protection as that other (nation) accords to work of its nationals first published in its own territory" (UCC, Art.II(1)).

The result of that national treatment requirement is that, if a national of a foreign member country publishes a computer program in the United States, that program is afforded the degree of protection granted by the United States Copyright Act, and vice versa.

II. Berne Copyright Convention (BCC)

Most nations have acceeded to the Berne Convention, either in its original form or with respect to one or more revisions of the Convention (187). The United States, in contrast to West Germany, is not a signatory of the Berne Convention. The major

reasons for the American position are the Berne abandonment of formalities, in view of the formalities of notice required under the U.S.law, and the existence of the U.S. manufacture clauses in the U.S.law. Furthermore, the American position results from the provisions of differing terms of protection under U.S.law and under the Berne agreement, and from objections in the United States to provisions in recent versions of the BCC regarding the "moral rights" (right of publication, right of paternity, right of integrity) of authors (188). Even though the United States is not a signatory of the BCC, American nationals can obtain Berne protection through the "backdoor to Berne", a device which allows works by American nationals to achieve Berne protection by being published either first or simultaneously in a Berne country, such as Canada or the United Kingdom.

Both the UCC and the BCC do not appear to exclude computer software from the protection that they afford; this applies to computer software in source and object code. However, these conventions provide protection for software only if the country in which protection is desired recognizes that protection.

III. World Intellectual Property Organization

The World Intellectual Property Organization (WIPO), established in 1967 to coordinate the activities of the Paris and Berne Unions, has produced a number of proposals concerning computer software.

First, an advisory group of non-governmental experts met in 1974 to study the subject. In view of the large degree of uncertainty generally related to the existence and form of protection under copyright, this group recommended that a special system of protection of software, similar to copyright, should be set up at national and international levels. Thus, in 1978, the International Bureau of WIPO presented the draft of a model law for national protection of software, and a draft treaty for the international protection and international deposit of software (189). The draft treaty specifies that states party to the agreement will constitute a union for the protection of computer software. Protection of software is not granted under the agreement, but must be undertaken by the contracting states through domestic legislation. Each contracting state must provide the same protection to others that it grants to its own nationals.

The WIPO Draft Treaty does not mention formalities such as registration requirements, or "moral rights" (supra), the two major reasons for the American non-adherence to the

Berne Convention. It is likely that one or both of these concepts will be incorporated in any final agreement (190). These two provisions would thus stand as major obstacles to any U.S. adherence to the treaty.

In any case, the positions of the United States and West Germany regarding a new treaty were rather ambivalent because, from their points of view, an additional agreement was not necessary (191).

Chapter 6:

Conclusion

Having presented an overview of the various problems related to the protection of computer software in West Germany and in the United States, it is now appropriate to analyze the main similarities and differences between them. As this essay has shown, the centre of gravity of protection in the U.S. and in Germany can be found in the copyright area. Protection under the Trade Marks Act, Unfair Competition Act and under trade secret law is secondary to copyright protection.

Concerning the protection of computer software under copyright law in both jurisdictions, there is one fundamental difference to be recognized, which leads to the result that one cannot find any general discussion about the nature of computer software in the U.S.; notions like "individual intellectual creation", "degree of creation" or "aestetic effect" are not part of the debate. This is because of the existence of four major arguments in favour of copyright: the principle of natural justice; the economic argument; the cultural argument; and, the social argument (192). The different legal systems give different priorities to these arguments, and put greater emphasis on some than on others.

The "droit d'auteur system" puts the emphasis on the

principles of natural justice, while an "Anglo-Saxon system" like that of the U.S. relies mainly on the economic argument.

In the "droit d'auteur" or "continental European system" the work which by the process of its creation becomes the property of the author gives him the right to exploit it economically (economic right); but the work also has an intellectual and moral link with its author, as being his brainchild, which gives him the right to publish it or not as he wishes, when he wishes, and in such form as he wishes; and the right to defend it against any distortions or abuses (moral rights). Perhaps the most far reaching consequence in twentieth century terms is that, because the "droit d'auteur" is a natural and therefore individual right, it can only originate in an individual and not in a company or corporation.

The Germanic jurisdictions (Germany, Austria, Switzerland), whilst based on the concept of the "droit d'auteur", are in some respect closer to the copyright system than to the French system. This affinity is particularly evidenced by the granting of "neighbouring rights", i.e. rights neighbouring on author's rights, such as the rights of performers, producers of phonograms and broadcasting organisations, which may originate in legal entities as opposed to natural persons as an inhibiting factor.

The philosophical foundation of the "Anglo-Saxon" or

"copyright system" is simply that whoever takes the initiative in creating the material and makes the investment to produce it or market it, taking the financial risks that such activities involve, should be allowed to reap the benefit. This philosophy is expressed in practical terms as the right to prevent the copying of physical material, with the object of producting the owner of the copyright against any reproduction of that material which he has not authorized. As the economic argument has always been in the foreground, it is never been contemplated that the first copyright owner might be a company or a corporation.

Another result of the different philosophical background is that in all Anglo-Saxon jurisdictions a work of authorship, such as a computer program, does not acquire copyright protection until it is fixed in writing or in any other material form; in other words fixation is a condition precedent to the existence of copyright. In Germany, on the other hand, copyright protection is available as soon as the work exists in the mind of the author; no further step is necessary to qualify for copyright. Thus, the work is an intellectual creation, not a material thing.

Although the discussion of copyrightability of computer programs in the U.S. has never extended to the question of their intrinsic legal nature, the opinion expressed by professor Nimmer in the Final Report of CONTU, is worth

mentioning (193). Professor Nimmer criticizes the Copyright Act as a general misappropriation law. Thus, he is of the view that the Act's open-ended copyright protection for all computer software, and its failure to articulate any rationale which would not equally justify copyright protection for the tangible expression of any and all original ideas, are fundamental flaws.

"... I should like to suggest a possible line of demarcation which would distinguish between protectible and nonprotectible software in a manner more consistent with limiting such protection to the convential copyright area".

"... it may prove desirable to limit copyright protection of software to those computer programs which produce works which themselves qualify for copyright protection" (194).

However, professor Nimmer's approach, seen within the context of the present American legislation, would not constitute an as extreme departure as might have been the case in 1983. The Semiconductor Chip Protection Act of 1984 would close the rising gap of copyrightability.

Philosophically the difference between the personal, individualistic and idealistic "droit d'auteur system", and the more commercially orientated "copyright system", may be fundamental; but in practice, as the present thesis shows, the outcomes do not really differ. This is because both systems presuppose a free economic market economy and grant a high level of protection. However, the "copyright system" seems to adopt more easily to the demands of new technology. The Software Act of 1980, amending the Copyright Act of 1976 was

an important milestone in copyright protection for computer software. With the Semiconductor Chip Protection Act of 1984, the United States has again shown the way, as the first country to protect technological innovations.

Endnotes

(1) All further references to "Germany" denote the Federal Republic of Germany.

(2) Victor Siber, "Clear trend: Copyright Law applies, The worldwide legal status of Software Protection" (1985) 82 col in v. 7 Nat'l L.J. 20; hereinafter "Worldwide Legal Status".

(3) Robert P. Merges, "Apple v. Franklin: An Essay on technology and judicial Competence" (1983) Yale L. & Policy Rev. 62; hereinafter "Judicial Competence".

(4) "Battling the Computer Pirates", N.Y. Times, Jan.5 1983, at D1, vol.3.

(5) Stan Soocher, "Software Infringement ruling prompt debate" (1985) 32 col in v. 7 Nat'l L.J. 11; hereinafter "Debate".

(6) Diamond v. Diehr, 450 U.S. 175, 187 (1981); Diamond v. Chakrabarty, 447 U.S. 303, 309 (1980).

(7) Similar protection schemes prevail in Australia, Hungary, the Philippines and India.

(8) Western Elec. Co. Patent Appl. p-14-74-091.1-53 (Federal Republic of Germany Patent Court, May 28, 1973); same program as in Gottschalk v. Benson, 409 U.S. 63 (1973).

(9) The study does not deal with legal problems such as employer-employee works made for hire; for greater detail relating to that aspect of software see:
Carol Ann Surrel, "The treatment of computer software works for hire and the Copyright Act of 1976" (1985) 5 Comp. L.J. 579, hereinafter "Treatment".
Debra S. Wilson "Software Rental, Piracy and Copyright Protection"(1984) 5 Comp.L.J. 125, hereinafter "Software Rental".

(10) 17 U.S.C. Section 101, as amended by the Computer Software Act of 1980.

(11) Andrew G. Rodau, "Protecting Computer Software: After Apple Computer, Inc. v. Franklin Computer Corp., 174 F.2d 1240 (3 Cir.1983), Does Copyright provide the best protection?" (1984) 57 Temple Law Quarterly 527, 539; hereinafter "Best Protection".

(12) Pamela Samuelson, "CONTU revisited: The Case against Copyright Protection for Computer Programs in machine-readable form" (1984) Duke Law Journal 663, 686, hereinafter "CONTU".

(13) Leslie Wharton, "Use and Expression: The scope of copyright protection for Computer Software"(1985) 5. Comp.L.J. 433, 436; hereinafter "Use and Expression".

(14) See "Best Protection", supra, note 11 at 539.

(15) Mickey T. Mihm, "Software Piracy and the Personal Computer: Is the 1980 Software Copyright Act effective?" (1983) 4 Comp.L.J. 171, 179; hereinafter "Software Piracy".

(16)For a more detailed distinction between binary form and object code, see
Duncan M. Davidson, "Protecting Computer Software: A comprehensive analysis" (1983) 23 Jurimetrics J. 237, 341; hereinafter "A comprehensive Analysis".

(17) James S. Altman, "Copyright Protection of Computer Software" (1985) 5 Comp.L.J. 413, 414; hereinafter "Computer Software".

(18) Apple Computer, Inc. v. Franklin Computer Corp. 545 F.Supp. 812, 814 (E.D.Pa 1982), rev'd, 714 F.2d 1240 (3.Cir.1983).

(19) Anderson L. Baldy III, "Computer Copyright Law: An emerging form of protection of object code software after Apple v. Franklin" (1984) 5 Comp.L.J. 233, 237; hereinafter "Emerging Form".

(20) S. Wyhron Semple, "The legal incidents of Computer Software and its use as collateral in secured transactions (1982-1983) 7 Can.Bus.L.J. 450, 454; hereinafter "Legal Incidents".

(21) See Robert H. Solomon, "The Copyrightability of Computer Software containing trade secrets" (1984) 63 Wash.U.L.Q. 131; hereinafter "Copyrightability".

(22) Trade Secret Protection is generally available in the industrial nations of Western Europe, see
Ray A. Mantle, "Trade Secret and Copyright Protection of Computer Software" (1984) 4 Comp.L.J. 669, 673; hereinafter "Trade Secret".
Cynthia L. Mellema, "Copyright Protection for Computer Software: An international view" (1984) 11 Syracuse J. Int'l L. & Com. 87, 97; hereinafter "Copyright Protection".

(23) See University Computing Co. v. Lykes-Youngston Corp., 504 F.2d 518 (5.Cir.1974);
Hancock v. Decker, 379 F.2d 552 (5.Cir.1967);
Warrington Assocs. v. Real-Time Eng'g Sys., 522 F.Supp. 367 (N.D.Ill.1981);
Management Science Am., Inc. v. Cyborg Sys., 6 Comp.L.Serv.Rep. 921 (N.D.Ill.1978).

(24) State Unfair Competition Law usually protects trade secrets. A 1976 survey of state trade secret statutes by a committee of the American Bar Association Section of Patent, Trademark, and Copyright Law reveded that twelve states do not have any laws protecting trade secrets and that the other 38 states have statuotory provisions offering various degrees of legal protection to trade secrets; reported in "Copyrightability", supra, note 21 at 132.

(25) Donald E. Stout, "Protecting of Programming in the Aftermath of Diamond v. Diehr" (1983) 4 Comp.L.J. 207, 237; hereinafter "Aftermath".

(26) See e.g. Cataphote Corp. v. Hudson, 444 F.2d 1313, 1315 (5.Cir.1971).

(27) See "Copyrightability", supra, note 21 at 134.

(28) 357 A.2d 105, 108, 110-111 (Del.Ch.1975).

(29) Motorola, Inc. v. Fairchild Camera and Instrument Corp., 366 F.Supp. 1173, 1186 (D.Ariz.1973), no trade secrets existed when a company gave guided tours through sensitive areas without restrictions.

(30) One can think of having a sign-in/sign-out procedure, escorting any guests, or having password protection on the computer system.

(31) See "Best Protection", supra, note 11 at 534.

(32) See "A comprehensive Analysis", supra, note 16 at 396.

(33) Greenberg v. Groydon Plastics Co., 378 F.Supp. 806, 812 (E.D.Pa.1974).

(34) Smith v. Dravo Corp., 203 F.2d 369, 373 (7.Cir.1953).

(35) Automated Sys. v. Service Bureau Corp., 401 F.2d 619, 625 (10.Cir.1968).

(36) Telex Corp. v. IBM Corp., 367 F.Supp. 258, 320 (N.D.Okla.1973), aff'd, 510 F.2d 894, 928-930 (10.Cir.), cert. dismissed, 423 U.S. 802 (1975) (affirming trade secret aspects).

(37) Cybertex Computer Prod. v. Whitfield, 203 U.S.P.Q. 1020, 1024 (Cal.Super Ct.1977).

(38) University Computing Co. v. Lykes-Youngston Corp., 504 F.2d 518, 534 (5.Cir.1974).

(39) Restatements of Torts Sections 757-758 (1938).

(40) Copyright Act of 1909, ch.320, 35 Stat.1075, 17 U.S.C. Sections 1-215 (repealed 1978).

(41) See "Trade Secret", supra, note 22 at 678.

(42) 524 F.Supp. 171 (N.D.Cal.1981).

(43) See, e.g. Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3.Cir.1983); Williams Elecs., Inc. v. Artic Int'l Inc., 685 F.2d 870 (3.Cir.1982);
see also CONTU "Final Report of the National Commission on New

Technological Uses of Copyrighted Works" (1981) 3 Comp.L.J. 53, 54, hereinafter "Final Report".

(44) See "A comprehensive Analysis", supra, note 16 at 363.

(45) See, e.g. Data Cash Syss., Inc. v. JS & A Group., Inc., 480 F.Supp. 1063, 1067-1068 n.4 (N.D.Ill.1979): "the 1976 Act applies to computer programs in their flowchart, source, and assembly phases but not in their object phase";
compare with:
Richard H. Stern, "Another look at copyright protection of software: Did the 1980 Act do anything for object code?" (1981) 3 Comp.L.J. 1 (object code not copyrightable), hereinafter "Another Look"; with note
"Copyright Protection of Computer Program object code" (1983) 96 Harv.L.Rev. 1723, 1732 (object code copyrightable), hereinafter "Object Code".

(46) 17 U.S.C. Section 102 (1982).

(47) Apple Computer, Inc. v. Formula Int'l Inc., 562 F.Supp. 775 (C.D.Cal. 1983);
Tandy Corp. v. Personal Micro Computers, 524 F.Supp. 171 (N.D.Cal.1981).

(48) 17 U.S.C. Section 102 (a) (1982).

(49) 17 U.S.C. Section 101 (1982).

(50) 17 U.S.C. Section 411 (a) (1982).

(51) 37 U.S.C. Section 202.20 (c)(2)(vii) (1984).

(52) John M. Conley & Robert M. Bryan, "A unifying theory for the litigation of computer software copyright cases" (1985) 63 North Carolina L.Rev. 563, 578, hereinafter "Unifying theory".

(53) See CONTU, supra, note 12 at 715.

(54) 17 U.S.C. Sections 705-706 (1982).

(55) See 37 C.F.R. Section 220.20 (d) (1984).

(56) 480 F.Supp.1063 (N.D.Ill 1979), aff'd because of lack of notice, 628 F.2d 1038 (7.Cir.1980).

(57) Id., p.1069.

(58) Id., p.1066.

(59) 524 F.Supp. 171 (N.D.Cal. 1981).

(60) Id., p.173.

(61) 685 F.2d 870 (3.Cir.1982).

(62) Id., p.877.

(63) Id.

(64) 714 F.2d 1240 (3.Cir.1983).

(65) 209 U.S. 1 (1908).

(66) 714 F.2d 1240, 1248 (3.Cir.1983).

(67) Id.

(68) Id., p.1252.

(69) Id., p.1249.

(70) 101 U.S. 99 (1879).

(71) Id., p. 104.

(72) Id., p. 103.

(73) 714 F.2d 1240, 1251 (3.Cir.1983).

(74) For more details on the question of tests: Howard Root, "Copyright Infringement of Computer Programs: A modification of the substantial similarity test", (1984) 68 Minn.L.Rev. 1264, 1276 et seq., hereinafter "Copyright Infringement".

(75) 714 F.2d 1240, 1252 (3.Cir.1983).

(76) Supra, at p. 19.

(77) Virginia Johnson, "Copyright Protection for Computer Flow Logic and Algorithms" (1984) 5 Comp.L.J. 257, 281, hereinafter "Algorithms".

(78) Id., nevertheless the author asks for copyright protection for particular combinations of algorithms comprising the program at the end, at 285.

(79) H.R.Rep.No.1476, 94th Cong., 2nd Sess.57, reprinted in 1976 U.S. code Cong. & Ad.News 5659, 5670.

(80) Ingrid Arckens & Jeff Keustermans "Der U.S.-amerikanische Semiconductor Chip Protection Act of 1984" RIW 1985, 280, 282. See also 17 U.S.C. Section 901 et seq..

(81) 17 U.S.C. Section 910 (b) (1).

(82) 17 U.S.C. Section 904 (a).

(83) 17 U.S.C. Section 902 (b).

(84) 17 U.S.C. Section 906.

(85) 17 U.S.C. Section 907.

(86) James Chesser, "Semiconductor Chip Protection: Changing Roles for Copyright and Competition" 71 Virginia L.Rev. 249, 294.

(87) 17 U.S.C. Section 902.

(88) 17 U.S.C. Section 101 (1982).

(89) Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1251 (3.Cir.1983).

(90) Baker v. Selden 101 U.S. 99 (1879).

(91) id.

(92) Data Cash Syss., Inc. v. JS & A Group, Inc. 480 F.Supp. 1063, 1066 n.4 (N.D.Ill.1979), aff'd 628 F.2d 1038 (7.Cir.1980).

(93) see to that ongoing dispute:
Richard A. Jordan "Concerning Davidson" (1984) 23 Jurimetrics 337, 24 Jurimetrics 397, hereinafter "Concerning Davidson";
and the reply
Duncan M. Davidson "Reply to letters by Merges and Jordan" (1984) 24 Jurimetrics 400, 402, hereinafter "Reply".

(94) see "Another Look", supra, note 45 at 4-5;
Jon O. Webster, "Copyright Protection of System Control Software stored in a read only memory chip: Into the World of Gulliver's Travels" (1984) 33 Buffalo L.Rev. 193, 222-223, hereinafter "Read Only Memory".

(95) i.d.

(96) see H.R. No.1476, 94th Cong., 2d Sess. 116, reprinted in 1976 U.S. Code Cong. & Ad.News 5659, 5664-5665.

(97) 17 U.S.C. Section 302 (a) (1982).

(98) Warrington Assoc., Inc. v. Real-Time Eng'g Sys., 522 F.Supp. 367, 369 (N.D.Ill 1981).

(99) 564 F.Supp. 1471 (D.Nev.1983).

(100) H.R.Report, No.1307, 96th Cong. 2d Sess. 23-24 (1980); "Final Report", supra, note 43 at 70.

(101) Technicon Medical Syss., Inc. v. Green Bay Packaging, Inc., 687 F. 2d 1032 at 1038 (7.Cir.1982).

(102) See "A comprehensive Analysis", supra, note 16 at 403, refering to the impossibility of copying the object code out of the Copyright Office or decompiling it.

(103) id., p.404.

(104) See, e.g. Management Science Am., Inc. v. Cyborg Syss., Inc. 6 Comp.L.Serv.Rep. 921 (N.D.Ill.1978).

(105) On October 1, 1982, the United States Court of Appeal for the Federal Circuit replaced the United States Court of Customs and Patent Appeals and assumed appellate jurisdiction of patent matters. All prior decisions were adopted as binding precedent on new court, see, e.g. South Corp. v. United States 690 F.2d 1368, 1371.

(106) 35 U.S.C. Section 154 (1982).

(107) 35 U.S.C. Section 101 (1982).

(108) 35 U.S.C. Section 102 (1982).

(109) 35 U.S.C. Section 103 (1982).

(110) See, e.g. Parker v. Flook 437 U.S. 584 (1978).

(111) See, e.g. Gottschalk v. Benson 409 U.S. 63 (1972), but see
Diamond v. Diehr 450 U.S. 175 (1981).

(112) See, e.g. "Algorithms", supra, note 76 at 261.

(113) See "A comprehensive Analysis", supra, note 16 at 349.

(114) Diamond v. Diehr 450 U.S. 175, 185-187 (1981); Parker v. Flook 437 U.S. 584, 589 (1978); Gottschalk v. Benson 409 U.S. 63, 67 (1972).

(115) 409 U.S. 63 (1972).

(116) 415 F.2d 1393 (1969).

(117) 415 U.S. 1393, 1403 n. 29 (1969).

(118) 431 F.2d 882 (1970).

(119) 437 U.S. 584 (1978).

(120) Id., p. 594-595.

(121) See "Aftermath", supra, note 25 at 216.

(122) 450 U.S. 175 (1981).

(123) Id., p. 191.

(124) Id., p. 192-193.

(125) 681 F.2d 787 (1982).

(126) 684 F.2d 902 (1982).

(127) 684 F.2d 912 (1982).

(128) 688 F.2d 789 (1982).

(129) 573 F.2d 1237 (1978).

(130) See "Aftermath", supra, note 25 at 229-231, stating patentability for system program (like operating programs) and application programs which do not implement mathematical algorithms.

(131) Michael Gemignani "Should algorithms be patentable" (1982) 22 Jurimetrics 326, 335, hereinafter "Patentability".

(132) For greater detail see "Patentability", supra, note 131 at 333 - 335.

(133) 35 U.S.C. Section 102 (1982).

(134) Aerotec Indus. v. Pacific Scientific Co., 381 F.2d 795 (9.Cir. 1967), cert. denied, 389 U.S. 1049 (1968).

(135) Tee-Pak, Inc. v. St. Regis Paper Co., 491 F.2d 1193 (6.Cir. 1974).

(136) See Mary Brandt Jensen, "Softright: A legislative solution to the problems of users' and producers' rights in computer software" (1984) 44 Lou.L.Rev. 1413, 1444, hereinafter "Softright".

(137) 35 U.S.C. Section 112 (1982) requires an applicant to describe the invention "in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same".

(138) Oskar Kienzle, "Patentierbarkeit von Computerprogrammen", Berlin 1975, hereinafter "Kienzle".

(139) Gert Kolle, "Der Rechtsschutz der Computersoftware in der Bundesrepublik Deutschland", GRUR 1982, 443; hereinafter "Kolle GRUR 1982".

(140) Europaeisches Patentuebereinkommen, EPU.

(141) Ernst Karl Pakuschker, "Rechtsschutz der Computerprogramme", UFITA 80, 35, hereinafter "Pakuschker UFITA 80".

(142) Philips Elektrologica GmbH Application, BPatG E 17 W6/68 (May 14, 1970).

(143) Gottschalk v. Benson 409 U.S. 63 (1972).

(144) Western Electric BPatG E 17 W71/70 (May 28, 1973); BPatG in Mitteilungen der deutschen Patentanwaelte 1973, 171.

(145) BGH Z XZB 23/74 (June 22, 1976).

(146) BPatG E 19, 97 and 102.

(147) LG Mannheim BB 1981, 1543.

(148) RG Z 124, 68.

(149) BGH GRUR 1971, 266, 267.

(150) BGH BB 1985, 1750.

(151) Eugen Ulmer, "Urheber -und Verlagsrecht", Berlin, Heidelberg, New York 1980, p. 133, hereinafter "Urheber- und Verlagsrecht".

(152) Id.

(153) Wilhelm Nordemann, "Der urheberrechtliche Schutz von Computersoftware ZUM 1985, 10 at 14, hereinafter "Nordemann ZUM 1985".

(154) OLG Frankfurt GRUR 1983, 753;
OLG Frankfurt GRUR 1983, 757;
OLG Frankfurt WRP 1984, 79.

(155) OLG Frankfurt GRUR 1983, 753.

(156) Id.

(157) Compare with BGH GRUR 1979, 464, 465.

(158) OLG Frankfurt GRUR 1983, 757.

(159) OLG Frankfurt WRP 1984, 79.

(160) OLG Karlsruhe BB 1983, 986.

(161) OLG Koblenz BB 1983, 992.

(162) LG Muenchen I BB 1983, 273.

(163) BGH BB 1985, 1747.

(164) For more details on the different steps:
Reimer Koehler, "Zum Urheberrechtsschutz fuer Programme elektronischer Datenverarbeitungsanlagen" BB 1969, 1114; hereinafter "Koehler BB 1969".

(165) compare with:
Otto-Friedrich v. Gamm, " Der Urheber- und Wettbewerbsrechtliche Schutz von Rechenprogrammen" WRP 1969, 96, 97, hereinafter "v.Gamm WRP 1969";
Wilhelm Nordemenn, "Das Computerprogramm als urheberrechtlich geschuetztes Werk" in Festschrift fuer Georg Roeber, Freiburg 1982, 297, hereinafter "Nordemann Festschrift".

(166) Rejecting in the case of object code, see
Eugen Ulmer, "Der Urheberrechtsschutz wissenschaftlicher Werke unter besonderer Berueecksichtigung der Programme elektronischer Rechenanlagen" Muenchen 1967, p.17, hereinafter "Ulmer Urheberrechtsschutz";
Eugen Ulmer, "Anmerkungen zu BAG" GRUR 1984, 429, 433, hereinafter "Ulmer GRUR 1984";
Gert Kolle, " Der Rechtsschutz von Computerprogrammen aus nationaler und internationaler Sicht" GRUR 1974, 7, 9; hereinafter "Kolle GRUR 1974";
Kolle GRUR 1982, supra, note 139 at 453.

(167) BGH BB 1985, 1747, 1750.

(168) See Eugen Ulmer & Gert Kolle "Der Urheberrechtsschutz von Computerprogrammen" GRUR Int. 1982, 489, 497, hereinafter "Ulmer/Kolle GRUR Int. 1982".

(169) BGH GRUR 1981, 520, 521.

(170) OLG Frankfurt GRUR 1983, 755;
BGH BB 1985, 1747, 1750.

(171) Ulrich Loewenheim, "Der urheberrechtliche Schutz von Computersoftware" ZUM 1985, 26, 30, hereinafter "Loewenheim

ZUM 1985".

(172) Gert Kolle, "Bericht der deutschen Landesgruppe fuer die Tagung des geschaeftsfuehrenden Ausschusses von 13.05.1985 - 18.05.1985 in Rio de Janeiro" GRUR Int. 1985, 29, 30, hereinafter "Kolle GRUR Int. 1985".

(173) Kolle GRUR Int. 1985, supra, note 172 at 29.

(174) Kolle GRUR Int. 1985, supra, note 172 at 31.

(175) Klaus Albert Bauer, "Urheberrechtsschutz von Computerprogrammen in den USA" GRUR Int. 1984, 136, 145, hereinafter "Bauer GRUR Int.1984".

(176) Kolle GRUR Int. 1985, supra, note 172 at 33.

(177) Ulrich Loewenheim, "Urheberrechtsschutz von Videospielen" in Festschrift fuer Hubmann, Frankfurt 1985, 307, 317, hereinafter "Loewenheim Festschrift";
Ulrich Sieber, "Der urheberrechtliche Schutz von Computerprogrammen" BB 1983, 977, 980, hereinafter "Sieber BB 1983".

(178) OLG Frankfurt GRUR 1983, 756.

(179) v. Gamm WRP 1969, supra, note 139 at 99;
Kolle GRUR 1982, 456.

(180) OLG Frankfurt GRUR 1983, 758;
OLG Frankfurt WRP 1984, 85.

(181) LG Mannheim BB 1981, 1544.

(182) BGH GRUR 1973, 478
BGH GRUR 1984, 453

(183) Universal Copyright Convention, done Sept.6, 1952 (1955) 6 U.S.T. 2731, T.I.A.S. No.3324, 216 U.N.T.S.132 (effective Sept.16, 1955) (Geneva Act), revised July 24, 1971 (1974) (Paris Act).

(184) Berne Convention for the Protection of Literary and Artistic Works, Sept.9, 1886, as amended by the Paris Additional Act and Declaration (1896), the Berne Convention (1908), the Berne Additional Protocol (1914), the Rome Convention (1928), the Brussels Convention (1948), and the Paris Convention (1971), 331 U.N.T.S. 217, T.S. No. 4757 (1948 text). For application that contains English translation of copyright laws and treaties of all nations of the world, see UNESCO, Copyright Laws & Treaties of the World (1982).

(185) See Arpad Bogsch, "The Law of Copyright under the Universal Convention"New York 1970, p.327 and 634.

(186) "Copyright Protection", supra, note 22 at 99.

(187) M.M. Boguslavsky, "Copyright in internatiuonal Relations: International Protection of Literary and Scientific Works", Sidney 1979, p. 74.

(188) "Copyright Protection", supra, note 22 at 99 n.50.

(189) See Walter E. Schmidt, "Legal Proprietary interest in Computer Programs: The American Experience" (1981) 21 Jurimetrics 345 at 403, hereinafter "The American Experience"; and
"WIPO: Legal Protection of Computer Software" (1983) 17 J.World Trade Law 537, hereinafter "WIPO".

(190) "Copyright Protection", supra, note 22 at 117.

(191) "WIPO", supra, note 189 at 539.

(192) Steven M. Steward, "International Copyright and Neighbouring Rights", 1983, p.3.

(193) "Final Report", supra, note 43 at 85.

(194) Id.

Bibliography

Altman, James S. "Copyright Protection of Computer Software" (1985) 5 Comp.L.J. 413

Arckens, Ingrid
Keustermans, Jeff "Der US - amerikanische Semiconductor Chip Protection Act of 1984", RIW 1985, 280

Baldy III, Anderson L. "Computer Copyright Law: An emerging form of protection for object code software after Apple v. Franklin" (1984) 5 Comp.L.J. 233

Bauer, Klaus Albert "Urheberrectsschutz von Computerprogrammen in den USA", GRUR Int. 1984, 136

Bogsch, Arpad The Law of Copyright under the Universal Convention, New York 1970

Boguslavsky, M.M. Copyright in International Relations: International Protection of Literary and Scientific Works, Sidney 1979

Brandt Jensen, Mary — "Softright: A legislative solution to the problem of users' and producers' rights in computer software" (1984) 44 Lou.L.Rev. 1413

Chesser, James — "Semiconductor Chip Protection: Changing Roles for Copyright and Competition", 71 Va.L.Rev. 249

Conley, John M.
Bryan, Robert M. — "A unifying theory for the litigation of computer software copyright cases" (1985) 63 North Carolina L.Rev. 563

CONTU — "Final Report of the National Commission on New Technological Uses of Copyrighted Works" (1981) 3 Comp.L.J. 53

Davidson, Duncan M. — "Protecting Computer Software: A comprehensive Analysis" (1983) 23 Jurimetrics 337

Davidson, Duncan M. "Reply to letters by Merges and Jordan" (1984) 24 Jurimetrics 400

Dorr, Robert C.
Eigles, William P. "Resolving claims to ownership of Software and Computer -stored data- The importance of temporary restraining orders and preliminary injunctions" (1984) 5 Comp.L.J. 1

Freed, N. Roy "Legal interest related to software program" (1985) 25 Jurimetrics 347

Harris, John R. "Apple Computer, Inc. v. Franklin Computer Corp. - Does a ROM a computer program make?" (1984) 24 Jurimetrics 248

Gemignani, Michael "Should algorithms be patentable?" (1982) 22 Jurimetrics 326

Johnson, Virginia "Copyright Protection for Computer Flow Logic and Algorithms" (1984) 5 Comp.L.J. 257

Jordan, Richard A. "Concerning Davidson (1983) 23 Jurimetrics 337" (1984) 24 Jurimetrics 397

Kienzle, Oskar Patentierbarkeit von Computerprogrammen, Berlin 1975

Koehler, Reimer "Zum Urheberschutz fuer Programme elektronischerDatenverarbeitungs-anlagen", BB 1969, 1114

Kolle, Gert "Der Rechtsschutz von Computerprogrammen aus nationaler und internationaler Sicht", GRUR 1974, 7

Kolle, Gert "Bericht der deutschen Landesgruppe fuer die Tagung des geschaeftsfuehrenden Ausschusses vom 13.05.1985 - 18.05.1985 in Rio de Janeiro", GRUR Int. 1985, 29

Kolle, Gert "Der Rechtsschutz der Computersoftware in der Bundesrepublik Deutschland", GRUR 1982, 443

Loewenheim, Ulrich — "Urheberrechtliche Schutz von Videospielen" in: Festschrift fuer Hubmann, Frankfurt 1985, 307

Loewenheim, Ulrich — "Der urheberrechtliche Schutz von Computersoftware", ZUM 1985, 26

Luccarelli, Peter A. — "The Supremacy of federal copyright law over state trade secret law for copyrightable computer programs marked with a copyright notice" (1981) 3 Comp.L.J. 19

Mantle, Roy A. — "Trade Secret and Copyright Protection of Computer Software" (1984) 4 Comp.L.J. 669

Mellema, Cynthia L. — "Copyright Protection for Computer Software: An international view" (1984) 11 Syracuse J.Int'l L.&Com. 87

Merges, Robert P — "Apple v. Franklin: An essay on technology and judicial competence" (1983) Yale L. & Policy Rev. 62

Mihm, Mickey T. "Software Piracy and the Personal Computer: Is the 1980 Software Copyright Act effective?" (1983) 4 Comp.L.J. 171

Nordemann, Wilhelm "Das Computerprogram als urheberrechtlich geschuetztes Werk" in: Festschrift fuer Georg Roeber, Freiburg 1982, 297

Nordemannn, Wilhelm "Der urheberrechtliche Schutz von Computersoftware", ZUM 1985, 10

Pakuschker, Ernst Karl "Rechtsschutz der Computerprogramme", UFITA 80, 35

Rodau, Andrew G. "Protecting Computer Software: After Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3Cir.1983), Does copyright provide the best protection?" (1984) 57 Temple Law Quarterly 527

Root, Howard — "Copyright Infringement of Computer Programs: A modification of the substantial similarity test" (1984) 68 Minn.L.Rev. 1264

Samuelson, Pamela — "CONTU revisited: The case against copyright protection for computer programs in machine-readable form" (1984) Duke Law Journal 663

Schmidt, Walter E. — "Legal Proprietary interests in Computer Programs: The American Experience" (1981) 21 Jurimetrics 345

Semple, S. Wynron — "The legal incidents of computer software and its use as collateral in secured transactions" (1982-1983) 7 Can.Bu.L.J. 450

Siber, Victor — "Clear Trend: Copyright Law applies, The worldwide legal status of Software Protection" (1985) 82 col in 7 Nat'l L.J. 20

Sieber, Ulrich — "Der urheberrechtliche Schutz von Computerprogrammen", BB 1983, 977

Solomon, Robert H. "The Copyrightability of Computer Software containing trade secrets" (1984) 63 Wash.U.L.Q. 131

Soocher, Stan "Software Infringement ruling prompt debate" (1985) 32 col in v. 7 Nat'l L.J. 11

Stern, Richard H. "Another look of copyright protection of software: did the 1980 Act do anything for object code?" (1981) 3 Comp.L.J. 1

Note to Stern, Richard H. "Copyright Protection of Computer Program object code" (1983) 96 Harv.L.R. 1723

Steward, Steven M. International Copyright and Neighbouring Rights, London 1983

Stout, Donald E. "Protection of Programming in the Aftermath of Diamond v. Diehr" (1983) 4 Comp.L.J. 207

Surrel, Carol Ann — "The Treatment of Computer Software works made for hire under the Copyright Act of 1976" (1985) 5 Comp.L.J. 579

Ulmer, Eugen — Der Urheberrechtsschutz wissenschaftlicher Werke unter besonderer Berueecksichtigung der Programme der elektronischen Rechenanlagen, Muenchen 1967, 171

Ulmer, Eugen — Urheber- und Verlagsrecht, Berlin, Heidelberg, New York, 1980

Ulmer, Eugen — "Anmerkungen zu BAG", GRUR 1984, 429

Ulmer, Eugen
Kolle, Gert — "Der Urheberrechtsschutz von Computerprogrammen", GRUR Int. 1982, 489

v. Gamm, Otto-Friedrich — "Der Urheber- und Wettbewerbsrechtliche Schutz von Rechenprogrammen", WRP 1969, 96

Webster, Jon.O. "Copyright Protection of Systems Control Software stored in read only memory chips: Into the world of Gulliver's Travels" (1984) 33 Buffalo L.Rev. 193

Wharton, Leslie "Use and Expression: The scope of copyright protection for computer programs" (1985) 5 Comp.L.J. 433

Wilson, Debra S. "Software Rental, Piracy, and Copyright Protection" (1984) 5 Comp.L.J. 125

WIPO "Legal Protection of Computer Software" (1983) 17 J.World Trade Law 537