

**A CRITIQUE OF THE LICENSE GRANT CLAUSE OF
THE ABA'S MODEL SOFTWARE LICENSING PROVISIONS WHERE
LICENSED TECHNOLOGY IS OR MAY BE COVERED BY A PATENT**

By:

**Proprietary Rights Committee
Computer Law Section
State Bar of Michigan**

Chairman

**David R. Syrowik
Brooks & Kushman P.C.
Southfield, Michigan**

Committee Members

**Daniel B. Burg
Mitchell A. Goodkin
Robert C. Hall
William M. Hanlon, Jr.
Mary I. Hiniker
Ronald M. Nabozny
Paul J. Raine**

***State Bar of Michigan
63rd Annual Meeting
September 16, 1998
Lansing, Michigan***

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
A. Background	1
B. ABA'S Model Software Licensing Provisions	2
C. Fact Situation	4
D. Five Different Licensing Situations	6
II. SELECTED MODEL PROVISIONS	7
A. Article 1	7
B. Article 10 - General Administrative Provisions	9
III. MODIFICATIONS OF THE MODEL LICENSE GRANT CLAUSE	10
A. First Situation	10
B. Second Situation	11
C. Third Situation	12
D. Fourth Situation	13
E. Fifth Situation	14
IV. CONCLUSION.....	15

I. INTRODUCTION

A. Background

Businesses and other entities that license software, especially small and medium size businesses, have traditionally relied upon copyright and trade secret protection to the exclusion of patents for their software products.¹ This fact is reflected by the absence of any patent terms in their license agreements.

In recent years, however, the number of software-related patents² sought from and issued by the United States Patent and Trademark Office (“the Patent Office”) has rapidly increased.³ Furthermore, recent actions and decisions by the Patent Office and the Court of Appeals for the Federal Circuit suggest a trend toward increasing the eligibility of software-based innovations for protection under U.S. patent laws.⁴ At the same time, a number of court decisions have cut back the scope of protection available for software under the copyright laws.⁵

In a recent case,⁶ the Court held that a conventional software license grant provision to use, modify and compile the plaintiff's source code in the defendants' voice recognition software product was a complete defense to a charge of infringement of plaintiff's patents. Obviously, such an implied license was not contemplated by the plaintiff, hence, the danger in using such conventional terms in software license agreements.

Such situations may arise more often in the future due to the increasing number of software patents as noted above, especially when the individuals licensing the software are not aware of the existence of any patent or patent application which covers the software.⁷

This Committee therefore saw a need for a license grant clause which would be used as a starting point in situations where licensed technology including software is or may be covered by a patent. The Committee's search for such a provision led to the ABA's Model Software Licensing Provisions which seek to cover situations where licensed software is covered by a patent. For example, intellectual property rights which are licensed include rights under the patent law as indicated under the definitional section entitled "Intellectual Property Rights." The license grant clause was studied by this Committee to see how it might be modified to cover five different licensing situations given the fact situation stated below. Obviously, other licensing situations are possible.

The modifications presented by the Committee are meant to illustrate one possible treatment for each general licensing situation described. They are not offered as legal advice about any specific licensing situation and should not be relied upon as such.

B. ABA'S Model Software Licensing Provisions

Working Draft 3.0 of the Model Software Licensing Provisions ("Model Provisions") was released in March, 1992. These Model Provisions were a project of the Committee on Computer Software of the ABA Section on Patent, Trademark, and Copyright Law and are designed to be incorporated by reference, in their entirety, in a software license agreement. The Model Provisions set forth a number of common and/or customary terms and conditions to govern a software license relationship in the absence of the parties' agreement otherwise. These terms and conditions represent synthesis of many software license agreements negotiated by big and small companies.

The Model Provisions set out express definitions for a number of standard terms commonly used in the software industry. The provisions also define some shorthand expressions in the interest of promoting a standard and convenient “lingua franca” for license drafters. Incorporation-by-reference can thus allow drafters to state quickly and succinctly many of the intentions of the parties, thus reducing the transaction costs of the license agreement.

The basic agreement document is divided into 10 Articles covering broad aspects of a software license relationship, including:

- Article 1: Creation of license relationship-grant clause, payment of License Fee;
- Article 2: Start-up activities, e.g., delivery and installation;
- Article 3: Operating provisions, e.g., single-user restrictions, movement of licensed copies;
- Article 4: On-going licensor support, e.g. telephone support, routine maintenance;
- Article 5: Procedures for correction of errors in the licensed software;
- Article 6: Performance-type warranties (including a no-virus warranty) and remedies;
- Article 7: Intellectual property matters (including a non-infringement “knowledge rep”);
- Article 8: Risk allocation (e.g., exclusion of consequential damages);
- Article 9: Breach and termination provisions;
- Article 10: Administrative and general provisions.

Several of the above articles have optional “snap-in” provisions. These provisions have an “s” before their section numbers. For example, Article 2 can be augmented with detailed “snap-in” provisions for custom development of software; Article 4 can be supplemented with “snap-in” provisions for periodic maintenance fees, maintenance releases, and the like. Extensive work went into integrating and coordinating these provisions (and their arrangement in numerical sequence) to be readily mixed and matched in this way. Commentary immediately below the provisions provided by the ABA Committee briefly explains some of the motivations behind and/or implications of these provisions.

C. Fact Situation

The early days of the Internet were funded largely by grants from the National Science Foundation. On Wednesday, December 2, 1992, the National Science Foundation Network (NSFNET) was de-commissioned. NSFNET was a nationwide network of dedicated 56kbps phone circuits that were replaced by 1.5Mbps (T1) circuits during the course of the NSFNET project. In 1992, the NSFNET which was based on twisted copper wire ceased to exist and the new 45Mbps fiber-optic cable network marked the end of an era.

Since the commercialization of the Internet in 1992, worldwide thirst for Internet bandwidth has been seemingly unquenchable. Phone companies have installed high-capacity fiber-optic cable in an attempt to meet the high demand. As more fiber is installed to carry the data, the problem shifts to the ability of the routers at each end of the fiber to handle the massive number of packets of information. Routers are responsible for directing packets of data so that they can most effectively reach their Internet destination.

The speed of light transmission of data over fiber-optic cable strands is limited by the rate at which data bits can be stored and processed by routers. Capacity of fiber-optic cables can be increased to terabit transmission rates through manipulation of the wavelengths of light without violating the laws of physics. Regardless of the ultimate transmission rate of data over fiber, the routers must be capable of processing the data at phenomenally high transmission rates.

\$1.8 billion will be spent this year on research by just five router manufacturers: Cisco, Bay Networks, Inc., Ascend Communications, Inc., Fore Systems, Inc., and Newbridge Networks Corp. Uunet and other large Internet providers are also investing in start-up companies such as Juniper Networks, Inc., which is developing a router that is capable of directing one trillion bits of data per second.

A router is typically a combination of both hardware and software. Future advances in router technology will likely include faster hardware circuit designs which may also be implemented in software.

Alternatively, new routing algorithms which are developed in software can be implemented in electronic circuitry in order to increase speed. In other words, the control logic by which the router operates may be implemented in software with conventional hardware or special purpose hardware or electronics. A router manufacturer may wish to retain patent rights on hardware implementations of routing algorithms, but may also want to license the software to companies it does not compete with in order to provide an additional source of revenue.

The software that the manufacturer wishes to license may need to be further developed in order for the Licensee to use and/or market it. Such further development may be done solely by the Licensor, the Licensee or together.

For purposes of this report, it is assumed that the Licensor-Developer has come up with a new concept or approach for doing data routing. Software which runs on a general purpose computer has been developed to implement that concept. Hardware to implement the concept may or may not be in existence.

D. Five Different Licensing Situations

The first licensing situation closely resembles a conventional software license situation where the licensee has the right to manufacture and distribute the software which typically runs on a general purpose computer. It is for software only with no right to modify. Consequently, only the object code of the software changes hands. Rights under any patent are licensed only to the extent necessary to allow the Licensee to manufacture and distribute the object code.

In the second licensing situation, the Licensee is to get the rights granted in the first situation, but also the right to make one or more derivative works of the original program. The Licensee will typically receive some source code of the program to make such modifications. No other use of the source code is contemplated.

The Licensee obtains maximum rights in the third licensing situation. The Licensee gets source code, all documentation that the Licensor has in its possession, and even any related hardware. The Licensor grants full patent and software licenses to the Licensee to permit the Licensee to do most anything it wants to do to fully exploit the technology, even to the extent of making hardware implementations of the concept.

In the fourth licensing situation, the Licensor begins to cut back on the software part of the license with increasing focus on the patent part of the license. Any software given to the Licensee is highly restricted in its use. For example, any software given may be given solely for study purposes. In other words, the Licensee can only use the provided software to guide it in writing its own software. In such a situation, the Licensor is attempting to limit its liability in any future products liability lawsuit involving the resulting software developed by the Licensee.

The fifth and final licensing situation closely resembles a conventional patent license where no software, either in source or object form, is transferred from the Licensor to the Licensee. The Licensee, however, is typically free to exploit or develop the technology in either hardware or software.

II. SELECTED MODEL PROVISIONS

A. Article 1

102. License Grant

102.1 *"Licensed Materials"*, as used in this Agreement, means one copy of the Executable Code and one copy of the Licensed Documentation.

102.2 *License Rights Granted*. Subject to the terms and conditions of this Agreement, LICENSOR hereby grants to LICENSEE and LICENSEE hereby accepts a license under any and all Intellectual Property Rights owned or otherwise assertable by LICENSOR to engage in the following Licensable Activities (the "License"):

[COMMENT: Delete and/or edit the paragraphs below]

- a. Internal Use (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation;
- b. Manufacture of Copies of the Executable Code of the Licensed Programs and of the User Documentation (see generally Sections et seq.) for purposes of Distribution in accordance with this Agreement (see generally Sections et seq.);
- c. Distribution of Copies (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation;
- d. Trademark Use of the Licensed Marks defined in, and in accordance with, Section .

102.3 *Scope of License Grant.* The License is non-exclusive and extends to the United States and its territories only.

102.4 *No Other Rights Granted.* Apart from the License Rights enumerated in this Agreement, the License does not include a grant to LICENSEE of any right to engage in any Licensable Activity, nor any ownership right, title, or interest, nor any security interest or other interest, in any Intellectual Property Rights relating to the Licensed Software nor in any Copy of any part of the Licensed Software.

ABA Commentary

Section 102.3 defines the scope and exclusivity of a license grant. This will often be the subject of negotiation. Consider whether it may be appropriate to treat various License Rights separately in their geographic territories, fields of use, and/or exclusivity.

B.

Article 10 - General Administrative Provisions

1001.50 Intellectual Property Rights means any and all rights to exclude existing from time to time in a specified jurisdiction under patent law, copyright law, moral rights law, trade-secret law, semiconductor chip protection law, trademark law, unfair competition law, or other similar rights.

ABA Commentary

The term "Intellectual Property Rights" is given a blanket definition that encompasses many different kinds of IP rights, commonly known as "rights to exclude." When reference is made to such rights, e.g., in connection with infringement of third-party rights, care should be taken to specify (i) what kind of IP rights, and (ii) the jurisdiction in which the IP rights in question exist, e.g., anywhere in the world, in the U.S. only, or in the Geographic Territory only.

1001.57 Licensable Activity means an activity encompassed by one or more Intellectual Property Rights, i.e., an activity which, absent a license, would give rise to liability for infringement (or inducement of infringement or contributory infringement) of the Intellectual Property Right(s).

III. MODIFICATIONS OF THE MODEL LICENSE GRANT CLAUSE

§ 102.1 and § 102.2 of the license grant clause can be modified as follows to cover the five licensing situations. Brackets, [], indicate deleted items whereas underlined items represent added material. § 102.2(d) has been eliminated in each situation because in each of the situations, the Licensor is not providing any materials that can be further distributed.

A. First Situation

102. License Grant

102.1 *"Licensed Materials"*, as used in this Agreement, means one copy of the Executable Code and one copy of the Licensed Documentation.

102.2 *License Rights Granted*. Subject to the terms and conditions of this Agreement, LICENSOR hereby grants to LICENSEE and LICENSEE hereby accepts a license under any and all Intellectual Property Rights owned or otherwise assertable by LICENSOR to engage in the following Licensable Activities (the "License"):

a. Internal Use (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation;

b. Manufacture of Copies of the Executable Code of the Licensed Programs and of the User Documentation (see generally Sections et seq.) for purposes of Distribution in accordance with this Agreement (see generally Sections et seq.); and

c. Distribution of Copies (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation[;].

[d. Trademark Use of the Licensed Marks defined in, and in accordance with, Section s721.]

B. Second Situation

102. License Grant

102.1 "*Licensed Materials*", as used in this Agreement, means one copy of the Source Code, one copy of the Executable Code and one copy of the Licensed Documentation.

102.2 *License Rights Granted*. Subject to the terms and conditions of this Agreement, LICENSOR hereby grants to LICENSEE and LICENSEE hereby accepts a license under any and all Intellectual Property Rights owned or otherwise assertable by LICENSOR to engage in the following Licensable Activities (the " License"):

a. Create Derivative Works of the Licensed Programs and the Licensed Documentation;

[a] b. Internal Use (see generally Sections et seq.) of the Source Code and the Executable Code of the Licensed Programs and of the [User] Licensed Documentation and of the Derivative Works Thereof;

[b] c. Manufacture of Copies of the Executable Code of the Licensed Programs and of the User Documentation (see generally Sections et seq.) and of the

Derivative Works Thereof, for purposes of Distribution in accordance with this Agreement (see generally Sections et seq.); and

[c] d. Distribution of Copies (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation[;] and of the Derivative Works Thereof.

[d. Trademark Use of the Licensed Marks defined in, and in accordance with, Section s721.]

C. Third Situation

102. License Grant

102.1 *"Licensed Materials"*, as used in this Agreement, means one copy of the Source Code, one copy of the Executable Code and one copy of the Licensed Documentation.

102.2 *License Rights Granted.* Subject to the terms and conditions of this Agreement, LICENSOR hereby grants to LICENSEE and LICENSEE hereby accepts a license under any and all Intellectual Property Rights owned or otherwise assertable by LICENSOR to engage in the following Licensable Activities (the " License"):

a. Engage in any Licensable Activity, including but not limited to make, use, sell and import Products in computer software form or in specially designed hardware form or any combination of software and hardware;

b. Create Derivative Works of the Licensed Programs and the Licensed Documentation;

[a] c. Internal Use (see generally Sections et seq.) of the Source Code and the Executable Code of the Licensed Programs and of the [User] Licensed Documentation and of the Derivative Works Thereof;

[b] d. Manufacture of Copies of the Executable Code of the Licensed Programs and of the User Documentation (see generally Sections et seq.) and of the Derivative Works Thereof, for purposes of Distribution in accordance with this Agreement (see generally Sections et seq.); and

[c] e. Distribution of Copies (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation[;] and of the Derivative Works Thereof.

[d. Trademark Use of the Licensed Marks defined in, and in accordance with, Section s721.]

D. Fourth Situation

102. License Grant

102.1 *"Licensed Materials"*, as used in this Agreement, means one copy of the Source Code, the Executable Code and one copy of the Licensed Documentation.

102.2 *License Rights Granted*. Subject to the terms and conditions of this Agreement, LICENSOR hereby grants to LICENSEE and LICENSEE hereby accepts a license under any and all Intellectual Property Rights owned or otherwise assertable by LICENSOR to engage in the following Licensable Activities (the " License"):

a. Subject to the limitation described in Subsection 102.2b below, engage in any Licensable Activity, including but not limited to make, use, sell and import Products in computer software form or in specially designed hardware form or any combination of software and hardware; and

[a] b. Internal Use only (see generally Sections et seq.) of the Source Code, the Executable Code of the Licensed Programs and of the User Documentation[;] solely to gain a better understanding of how to implement the technology, with no right to distribute any substantial copy of any portion of Licensed Materials in any form.

[b. Manufacture of Copies of the Executable Code of the Licensed Programs and of the User Documentation (see generally Sections s321 et seq.) for purposes of Distribution in accordance with this Agreement (see generally Sections s311 et seq.);]

[c. Distribution of Copies (see generally Sections s311 et seq.) of the Executable Code of the Licensed Programs and of the User Documentation;]

[d. Trademark Use of the Licensed Marks defined in, and in accordance with, Section s721.]

E.

Fifth Situation

102. License Grant

102.1 [*"Licensed Materials"*, as used in this Agreement, means one copy of the Executable Code and one copy of the Licensed Documentation] Licensor will not provide to Licensee any "Licensed Materials."

102.2 *License Rights Granted.* Subject to the terms and conditions of this Agreement, LICENSOR hereby grants to LICENSEE and LICENSEE hereby accepts a license under any and all Intellectual Property Rights owned or otherwise assertable by LICENSOR to engage in the following Licensable Activities (the "License"):

a. Engage in any Licensable Activity, including but not limited to make, use, sell and import Products in computer software form or in specially designed hardware form or any combination of software and hardware.

[a. Internal Use (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation;]

[b. Manufacture of Copies of the Executable Code of the Licensed Programs and of the User Documentation (see generally Sections et seq.) for purposes of Distribution in accordance with this Agreement (see generally Sections et seq.);]

[c. Distribution of Copies (see generally Sections et seq.) of the Executable Code of the Licensed Programs and of the User Documentation;]

[d. Trademark Use of the Licensed Marks defined in, and in accordance with, Section .]

IV. CONCLUSION

The editing of the model License Grant clause in each of the last four licensing situations illustrates not only the flexibility of the clause but also the need to expressly state the intentions of the parties in a technology licensing agreement where software is or may be covered by a patent as indicated by the Fact Situation. Boiler plate and model provisions and clauses cannot be taken on their face, but rather provide a starting point for initial study and then drafting.

-
1. Until 1981, the Patent Office routinely fought the issuance of such patents. Until that time, Supreme Court decisions cast considerable doubt over the statutory authority for the patentability of software. *Gottschalk v. Benson*, 409 U.S. 64 (1972); *Parker v. Flook*, 437 U.S. 584 (1978). Consequently, the public and even most patent attorneys held the view that “software isn't patentable”. Trade secret and, later, copyright protection were generally recognized as “tried and true” forms for protecting one's software. *Com-Share, Inc. v. Computer Complex, Inc.*, 338 F.Supp. 1229 (E.D. Mich. 1971); *aff'd* 458 F.2d 1341 (6th Cir. 1972)(“Unique engineering, logic and coherence” of the computer program protected); *Telex Corp. v. IBM Corp.*, 510 F.2d 894,911 (10th Cir. 1975). Computer Software Copyright Act of 1980, amending 17 U.S.C. § § 101, 117; *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240 (3rd Cir. 1983) (computer programs protectable under the copyright laws whether in source or object form, whether an application or operating system program, or whether stored in ROM or other media).

The U.S. Supreme Court's 1981 opinion, *Diamond v. Diehr*, 450 U.S. 175 (1981) opened the gates for patent protection for computer software. Because so many previously relied on trade secret law to protect their software technology, there is a lack of a comprehensive patent database for examiners to search. This has been one of the primary criticisms against the patentability of software-related inventions. It is believed that this will change as the software industry becomes more aware of the value and applicability of patents and the patent database continues to expand.

-
2. The term “software” is used somewhat differently by various authors. However, for purposes of this report, “software” is defined to be computer programs (“programs”). A program can be thought of as the intelligence or control logic communicated to a computer in order to enable the computer to perform a particular function. Typically, a computer program is prepared to solve a problem or to automate a mundane or complex task (i.e., to “teach” a general purpose machine to perform a particular desired function). Initially, specific methods for solving the problem are fashioned in a “logic” or “algorithm” stage. A block diagram “flowchart” may be prepared. The program is then written in a “coding” stage. This typically results in a sequence of instructions in a rigidly defined language which may have a resemblance to English. This is a “source program” which must be translated into an “object program” which is a sequence of “bits” (i.e., zeros and ones) that the computer understands. Consequently, a program is a set of instructions which embodies the logic or algorithm used to solve the initial problem.

The fact that there is a functional equivalence between hardware and software is important to an understanding of software-related patents. That is, technology has blurred the distinction between hardware (wired circuits) and software (program) solutions to the same problem such that there is generally a functional equivalence between a program executed by a computer and a wired electrical/electronic logic circuit.

A software-related patent is defined herein as a patent that claims some feature, function or process embodied in a program that is executed on a computer as a substantially significant portion of its invention. *See*, for example, Sumner and Lundberg, “The Versatility of Software Patent Protection: From Subroutines to Look and Feel”, *THE COMPUTER LAWYER*, Vol. 3, No. 6, June 1986.

3. Classes 364 and 395 are generally acknowledged to be those in which the Patent Office classifies most software-related patents. Class 395 was created in 1991 and replaced selected art areas of Class 364. The creation of Class 395 seems to at least partially remove one reason that the Supreme Court relied upon in denying patent protection for the software-related invention it reviewed – The Patent Office's lack of a reliable classification system. *Gottschalk v. Benson*, 409 U.S. 63, 72 (1972).

It is difficult to determine how many of the patents granted in these classes cover software inventions because the classes include other types of inventions as well. However, 3270 and 1174 patents were sought and issued, respectively, in class 364 in 1987. This is to be contrasted with 11,714 and 4,199 patents sought and issued, respectively, in combined Classes 364 and 195 in 1995.

4. As reported at 61 Fed. Reg. 7,478-7,492, effective March 29, 1996, the Patent Office adopted new examination guidelines for evaluating the patentability of computer-related inventions. The Patent Office's examination guidelines are intended to assist examiners in their review of applications for software-related inventions. Although the guidelines do not have the status of rules, they govern the Patent Office's internal examination

policies. These come in the wake of the Patent Office's decision to reverse its own ruling in *In re Beauregard*, 53 F.3d 1583 (Fed. Cir. 1995), on the patentability of software embodied on a floppy disk. Moreover, the Federal Circuit's en banc decision at 60 F.3d 807 (Fed. Cir. 1995) to vacate its prior panel decision in *In re Trovato*, 42 F.3d 1376 (Fed. Cir. 1995) and reconsider the case in light of *In re Alappat*, 33 F.3d 1526 (Fed. Cir. 1994) (en banc.) and the guidelines indicate that court's willingness to extend patent coverage for software-related inventions. Finally, as reported at 47 U.S.P.Q.2d 1596, the Federal Circuit on July 23, 1998 went even further by holding that a machine that performs mathematical calculations to transform data for administering mutual funds is statutory subject matter under Section 101 of Title 35 in *State Street Bank & Trust Co. v. Signature Financial Group, Inc.* The court reversed a summary judgment to the contrary, explaining that the claimed system constitutes a “useful” application of a mathematical algorithm. Along the way, the court found the *Freeman-Walter-Abele* test inapplicable and discarded the “ill-conceived” business method exception to statutory subject matter.

5. The *Gates Rubber Co. v. Bando Chemical Industries, Ltd.*, 9 F.3d 823 (10th Cir. 1993) (unprotectable elements of a copyrighted computer program were not sufficiently filtered out by the district court in its finding of infringement). In *Gates*, the Court adopted the abstraction-filtration-comparison test under *Computer Associates International, Inc. v. Altai, Inc.*, 982 F.2d 693 (2nd Cir. 1992). This test is more restricted in determining “substantial similarity” in copyright infringement controversies than the “one idea” approach of *Whelan Assoc. v. Jaslow Dental Laboratory, Inc.*, 797 F.2d 1222 (3rd Cir. 1986).

Two Federal appellate courts recently stated that intermediate copying of computer software for reverse engineering may be “fair use” as long as the copying is not more than necessary to understand the unprotected ideas of the work. *Sega Enterprises, Ltd. v. Accolade, Inc.*, 977 F.2d 1522 (9th Cir. 1992), and *Atari Games Corp. v. Nintendo of America, Inc.*, 975 F.2d 832 (Fed. Cir. 1992).

The First Circuit overturned Judge Keeton's decision of infringement in *Lotus Development Corp. v. Borland International, Inc.*, 49 F.3d 807 (1st Cir. 1995). The Court rules that *Borland's* Quattro Pro spreadsheet couldn't infringe *Lotus's* copyright on its popular 1-2-3 spreadsheet by copying its menu command hierarchy because *Lotus's* menu tree was a “method of operation” and, as such, was “uncopyrightable subject matter”. *Lotus*, now a unit of IBM, had argued that the order of menu commands represented a creative decision by a programmer that was copyrightable. As reported in the January 17, 1996 issue of the WALL STREET JOURNAL, the Supreme Court, on January 16, 1996 split 4-4 with Justice Stevens abstaining, thereby affirming the appellate decision for the First Circuit; however, the decision by an equally divided court is not precedential for any other Circuit.

The “virtual identity standard” was the standard adopted by the Ninth Circuit with respect to Apple Computer's graphical interface. The Court refused to view the work as a whole and apply the “total concept and feel” standard. *Apple Computer, Inc. v. Microsoft Corp.*, 35 F.3d 1435 (9th Cir. 1994). The Court held that *Apple* was entitled to “thin” protection

only, in view of the large proportion of the interface that was licensed and the limited number of ways that the interface could be expressed.

6. *Apple Computer, Inc. v. Articulate Systems, Inc.*, 44 U.S.P.Q.2d 1369 (N.D. Calif. 1997).
7. This problem can be latent because the filing of a patent application with the Patent Office can be delayed up to 1 year after first commercialization of the software. 35 U.S.C. § 102 (1988).