The Changing Roles of Patent and Copyright Protection for Software and Communication Technology In the Internet Age

GREGORY J. KIRSCH, TIM TINGKANG XIA
Needle & Rosenberg, P.C.

Key words: Patent, Copyright & Internet

Abstract: The roles are changing dramatically for patent and copyright protection as legal mechanisms for protecting software and communication technologies. While copyright protection has traditionally been the primary protection method for software, recent case law in the United States and elsewhere has reduced the scope of protection provided by the copyright laws. At the same time, courts, including the U.S. Supreme Court and the U.S. Court of Appeals for the Federal Circuit, have increasingly allowed broader patent protection for these technologies. Given that patents protect the useful, functional features of an invention, as opposed to copyrights protecting only the way a work is tangibly and non-functionally expressed, patents are increasingly becoming the legal protection tool of choice for software and communication technology developers. The explosive growth of E-Commence further strengthens the trend to use software patents as business assets and to provide a company with significant leverage and defensive rights.

1. INTRODUCTION

During the past several years, many computer software and communication technology companies have increasingly begun to use patents as a primary mechanism for legally protecting their computer software and other innovative techniques from misappropriation by others. At the same time, many software companies have grown to rely less on copyright protection to protect their innovation. This divergent, but possibly related, trend has caused a dramatic shift in thinking in the software and communication technology industries — a change destined to have a major

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35515-3_53

S. Qing et al. (eds.), *Information Security for Global Information Infrastructures*

impact on the continued advancement of these technologies. Regardless of whether this impact is deemed positive or negative, it will be a significant impact nonetheless.

Moreover, stemming from the recent State Street Bank & Trust v. Signature Financial Group Inc. and AT&T v. Excel decisions by the U.S. Court of Appeals for the Federal Circuit, more companies are rushing to get software and business method patents, and, consequently, Internet patent wars are on the rise. Thus, in this Internet age, can software, and especially e-commerce inventions, be patented and actually protected through patent enforcement actions? If so, how much protection does such a patent provide?

In order to answer these intriguing questions and understand how the changing roles of patent and copyright protection will affect the software and communication technology industries, at least a basic understanding of the underpinnings of the patent and copyright laws must be possessed. The present paper will focus on patent and copyright protection in the United States, although it will be understood that many of the basis principles discussed will be applicable to the intellectual property laws of other countries. Also, because of the many similarities between the application of intellectual property principles to both software and communication technologies (e.g., Internet, telecommunications, etc.), the below discussion treats these varying technologies in a like manner.

2. COPYRIGHT AND PATENT PROTECTION

2.1 U.S. Copyright Protection

In the United States, the copyright and patent laws were created by Congress under it's Constitutionally mandated power: to promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries.

With respect to the copyright law, the key constitutional language is "to promote the Progress of Science ..., by securing for limited Times to Authors ... the exclusive Right to their respective Writings...". The copyright laws serve to protect the "writings" of "authors" against unauthorized copying. Only those works that are "original" are protected, and therefore, independent creation by another without access to the copyrighted work would not be copyright infringement, and would, in fact, entitle the second work to copyright protection in its own right.

In the computer software context, a computer program, either in its original "source code" format, or in its machine-readable format, is generally deemed to fall within the definition of a "writing" for copyright purposes. In

fact, in 1980, the U.S. copyright laws were amended to make explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright. While there has been a considerable amount of debate on this point, case law supports this view. In short, with respect to a computer program, copyright does not protect the idea behind the program – i.e., what function the code performs. Copyright only protects the expression of the program - that is, how it is written.

2.2 U.S. Patent Protection

Unlike copyright law, the relevant constitutional language for implementing the patent laws is "to promote the Progress of ... useful Arts, by securing for limited Times to ... Inventors the exclusive Right to their ... Discoveries." Because it is the "useful arts" that are being protected, only a useful "process, machine, manufacture, or composition of matter, or any ... useful improvement thereof" can be protected by a patent. In addition to being useful, in order to obtain a patent, the invention must also be novel (new) and non-obvious.

The novelty requirement may be contrasted with the originality requirement of copyright law. In order to be novel and non-obvious, the invention must not have been a part of the "prior art," and must not have been an obvious variation of the prior art, regardless of whether the invention was created independently from that of the prior art or not. This is markedly different from copyright law, where independent creation of even an identical work results in copyright protection in the second created work.

An issued patent allows the patent owner to exclude others from making, using or selling the patented invention. Such a right operates regardless of whether a third party copied the invention from the inventor of the patented subject matter.

2.3 Why Copyright and Patent Protection?

When developing computer software, the creator of the software usually wishes to ensure that the time and effort expended is somehow protected against misappropriation by, for example, a competitor. By the same token, the software developer not only wants to prevent others from making verbatim copies of the software, but also from copying as much of the innovation that went into the software as possible. Software operating on a computer causes the computer to perform a process, and the process can usually be represented by any one of a multitude of different (even if functionally equivalent) sequences of software code. Thus, a software developer does not merely wish to rely on the prevention of verbatim copying of the software, since a competitor may observe the functions

performed by the software, and without knowing the details of the software code underlying the functions, write equivalent code.

Thus, given the choice, most software developers want to be able to (1) prevent others from making, using and selling verbatim copies of the software, and (2) prevent others from utilizing the functionality by which the software operates. From the software developer's point of view, the broader such protection exists, the better.

Just as the software developer wants to protect its software from being copied, either exactly or functionally, society (you, me and everybody other than the software developer) wants to have broad and inexpensive access to software that performs useful functions in new and innovative ways. It would benefit society the greatest if it (1) could have access to a wide variety of innovative and useful software products, (2) while paying little for such software.

Because software lends itself well to being easily copied and distributed, it would be possible for the laws to allow for society to pay little for software -- by simply allowing for any and all copying and distribution of software by third parties. Of course, such a law would provide little incentive for software development, as few companies or individuals would be willing to go to the trouble to develop software knowing that it could be easily copied without recourse.

Thus, assuming that society wants to have wide access to innovative software, then it is safe to say that society is willing to pay for this access, such as by allowing software developers to have proprietary rights in their creations. However, unlike the software developer, who wishes to have broad rights which allow him or her to reap a maximum return on investment, society likely is only willing to grant the software developer enough of an incentive so as to create a minimum threshold of innovation -- or usefulness -- in the developed software. In the end, society wishes to reward the software developer, but not to the extent that the developer would optimally desire.

Obviously, a gap exists between the level of legal protection afforded software that the software developer ideally wants, and that which society is willing to grant. However, as with any other difficult legal determination, a line, albeit not always a bright line, can be created which is equally fair to both the creator of software and society.

When creating the ideal level of protection for software, a number of factors can be taken into account, including:

(1) At what minimum level of protection will a software developer be willing to develop a particular piece of software?

(2) What is the maximum price society is willing to pay for such a piece of software?

(3) How can the level of protection be created so that the maximum amount of innovation in software is generated, at a price suitable for the maximum number of members of society?

Of course, these are but a few of the types of policy questions that must be asked and answered in order to arrive at the fairest level of protection.

The succeeding paragraphs will attempt to provide an overview of the dramatically changing roles of patent and copyright protection for software and communication technology.

3. THE CHANGING ROLES OF PATENT AND COPYRIGHT PROTECTION

3.1 Copyright Protection

While software has been specifically identified by Congress and the courts as deserving of copyright protection, the scope of copyright protection afforded software has been in flux in recent years. The copyright laws specifically state that copyright protection does not "extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery." Thus, as explained previously, copyright protection extends only to specific expression, and not to the ideas behind this expression -- commonly referred to as the "idea/expression" dichotomy.

Within the last decade, the pendulum of copyright protection for software has come full swing. Perhaps the most notable software copyright case in the mid-1980s was Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc., 797 F.2d 122 (3rd Cir. 1986), a case from the Third Circuit Federal Court of Appeals that was cited for support in many subsequent judicial opinions. In Whelan, the defendant obtained an unauthorized copy of the plaintiff's source code for its software, and developed its own competing version of the plaintiff's software application. In finding for the plaintiff, the Court held that "copyright protection of computer programs may extend beyond the programs' literal code to their structure, sequence and organization." In reaching this decision, the Court noted that the majority of the creative effort in developing a computer program involves the design rather than the mere coding of the program.

The Whelan Court therefore created a relatively broad definition for copyrightable subject of software -- everything that is not necessary to the computer program's purpose or function, including its "structure, sequence and organization."

After the Whelan case was decided, a number of subsequent cases cited to Whelan for support, including those cases that purportedly upheld copyright protection for the "look and feel" of software. While many of

these cases utilized the very general and broad test outlined in Whelan, a number of the cases took different approaches, resulting in much confusion as to the appropriate scope of copyright protection for software.

In 1992, the Second Circuit Federal Court of Appeals decided Computer Associates Int'l v. Altai, Inc., 982 F.2d 693 (2nd Cir. 1992), which specifically rejected the simplistic test regarding the scope of copyright protection formulated in Whelan. In Computer Associates, the Court developed a three-part test for determining whether software is infringed under the copyright laws. The test, which has become known as the "abstraction/ filtration/ comparison" test, is based upon a similar copyright infringement test enunciated by Judge Learned Hand in Nichols v. Universal Pictures Corp., 45 F.2d 119 (2nd Cir. 1930).

In the first step of the revised Computer Associates test, the computer program is divided into its various levels of abstraction. The second test, the "filtration step," entails examining the structural components of the software at each level of abstraction to determine (1) whether their particular inclusion at that level was "idea" or was dictated by considerations of efficiency, (2) whether their inclusion was required by factors external to the program itself, such as a required data input or output protocol, or (3) whether the structural components were taken from the public domain. If a particular structural component at each level of abstraction satisfies any of the three criteria, then it is non-protectable expression, and is not considered in the final step of the test, described below. The third and final step, the "comparison" step, involves comparing the expression left after the filtration step at each level of abstraction to the accused software in order to determine whether there is substantial similarity between the two. If there is substantial similarity, and it can be shown that the developer of the accused software had access to the original software, then copyright infringement may be found.

While the Computer Associates test has been adopted fairly uniformly in subsequent court decisions around the country, it is likely to result in a lesser scope of protection for computer software. The old Whelan test gave a more expansive view of protectable expression, while the Computer Associates test provides a more detailed approach whereby great care is taken in order to examine the software code on many levels, and to remove non-protectable expression before an infringement comparison is made. Regardless of whether the Computer Associates test represents a more sound test for infringement, the authors feel that it represents a general narrowing of protection afforded to software under the copyright laws.

3.2 Patent Protection

Like copyright protection, the allowable scope of patent protection for software-related inventions has changed considerably. However, unlike copyright protection, the allowable scope for software-related patents has enlarged in recent years.

Prior to the early 1980s, conventional wisdom held that software was not patentable. This view was furthered by two notable U.S. Supreme Court cases, Gottschalk v. Benson, 409 U.S. 63, 93 S.Ct. 253, 175 USPQ 673 (S.Ct. 1972) and Parker v. Flook, 437 U.S. 584, 98 S.Ct. 2522; 198 USPQ 193 (S.Ct. 1978), in which software-implemented inventions were held to not be patentable. Although Congress has never passed a patent statute expressly providing for federal patent protection of computer software, in 1981 the U.S. Supreme Court decided Diamond v. Diehr, 450 U.S. 175, 101 S.Ct. 1048, 209 USPQ 1 (S.Ct. 1981), which expressly held for the first time that computer software was patentable (or put another way, that an invention was not necessarily unpatentable simply because it utilized software). In Diehr, the Supreme Court found that computer software was patentable provided that any claims to the computer software were not merely a procedure for solving a mathematical formula. Though the significance of this case was not immediately appreciated by most computer software companies, certain companies recognized that the Supreme Court had provided a road map for patenting computer software. In addition to the Supreme Court's opening of the door for protecting computer software with patents, the Court of Appeals for the Federal Circuit was established in 1982 as the sole appellate court authorized to hear all patent cases. The Federal Circuit promptly made it clear that patent protection for software would assume greater prominence in the future.

During the mid and late 1980's, the preferred means of protecting computer software began to shift from copyrights to patents. Specifically, more companies began to file patent applications for their computer software inventions as the U.S. Patent Office began to relax its standards somewhat for issuing computer software patents and the federal courts upheld the majority of patents issued. However, it was not until the early 1990s that software vendors clearly favored patents for computer software. The reason for the change stemmed in part from a series of federal court decisions, led by Computer Associates v. Altai as well as Lotus v. Borland, which held that copyright protection for computer software should be applied narrowly in order to essentially protect little more than the exact copying of software code. These courts sent the message that broad protection for the functionality of computer software should be sought under patent laws. Many legal scholars agree with this view.

The trend towards broader patents directed to software inventions was arguably furthered in cases such as In re Alappat, 33 F.3d 1526, 31 USPQ2d 1545 (Fed. Cir. 1994), decided by the Court of Appeals for the Federal

Circuit. In Alappat, the Federal Circuit judges decided that the patent application presented proper "useful" statutory subject matter, even though the patent claims merely presented the software process in terms of different physical "elements" within a machine to perform the functions.

More recently, in State Street Bank & Trust v. Signature Financial Group, 1998 U.S.App. LEXIS 16869 (Fed. Cir. July 1998), the U.S. Court of Appeals for the Federal Circuit held that a patent directed to a "hub and spoke" data processing system that allows mutual funds (spokes) to pool their assets into an investment portfolio (hub) organized as a partnership was valid. The claimed invention made several calculations on a daily basis related to the financial configuration of the mutual funds, including each fund's percentage share of the investment's assets and expenses.

In holding that the subject patent was valid, the court held that a long-standing doctrine, known as the "business method exception", was ill-conceived and not appropriate. The business method exception to patentability has traditionally been used to strike down patents directed to methods of doing business. The court ruled that this exception to patentability is not only ill-conceived, but is not appropriate to computer-implemented inventions which clearly otherwise satisfy the tests of patentability. For the first time, the court made it clear that a mathematical algorithm that produces numbers as its only output constitutes patentable subject matter if it produces a result that is "useful, connected and tangible."

As a result, culminating with the State Street Bank and AT&T v. Excel (which affirmed the State Street Bank holding) cases, the courts and the U.S. Patent Office have now made it abundantly clear that virtually any kind of software or other computer-implemented invention is subject to patent protection so long as the standard tests for patentability (useful, new and not obvious) are met. While many software companies in a variety of fields have already begun to file for patent protection in increasing volume, the State Street Bank case will undoubtedly further this trend, among companies developing all types of software and Internet technology.

3.3 The Changing Roles

Figure 1 below is a simplified graphical representation of the changing roles of patent and copyright protection. The horizontal axis represents the time domain, whereas the vertical axis represents the scope and perceived "strength" of protection. The dashed line represents the diminishing "strength" of copyright protection for software and communication technologies, whereas the solid line represents the increasing "strength" of patent protection for these technologies. While of course Figure 1 is a simplistic representation of the change that has occurred, and which continues to occur, it may be useful to illustrate the point that the intellectual

property paradigm for software and communication technology has shifted, and continues to shift.

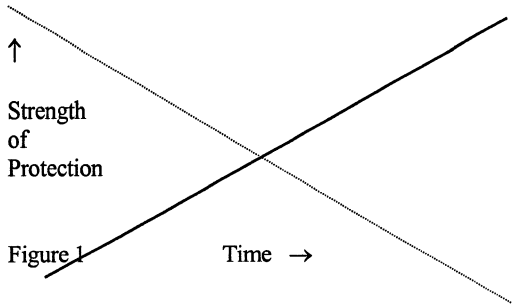

Figure 1

While patents are generally viewed as providing broader legal protection than copyrights, patents almost always cost much more to procure and perfect than copyrights. Additionally, while copyright protection automatically subsists in software immediately upon its expression in a tangible form, one must apply for a patent with the U.S. Patent Office. A patent application may not result in the issuance of a patent for many years, during which time the applicant cannot enforce any patent rights. Nevertheless, many computer software industry leaders commonly agree that patents will play a major role in the future of most software companies, regardless of whether they agree with the policies surrounding patent law. Therefore, it should come as no surprise that the U.S. Patent Office reportedly has tens of thousands of software patent applications pending, and in response to public pressure, in 1996 prepared a manual entitled "Examination Guidelines for Computer-Implemented Inventions" in order to assist its patent examiners and the public in prosecuting software patent applications.

3.4 The Internet Patent Goldrush

Riding the wave of seeking patent protection for software and Internet related technologies, the U.S. Patent and Trademark Office has now issued thousands of patents in the area of business management, finance and accounting. The State Street Bank case has opened the floodgates to the aggressive use of these and future patents as a strong competitive weapon and valuable assets in the Internet age.

For example, priceline.com recently sued Microsoft for software patent infringement. Priceline claims that its popular website at www.priceline.com, which utilizes "name-your-price" or "reverse auction" business method, is protected by U.S. Patent No. 5,794,207 and that Microsoft infringes that patent. It is interesting to note that Priceline now has filed more than 20 patent applications in the U.S. Patent Office (at least

seven of them have been granted) and is aggressively building a portfolio of Internet patents.

Priceline.com is not the only Internet powerhouse to protect its technology using patents. Amazon.com also recently sued Barnesandnoble.com for infringing its patented "1-click" technology, which allows a user to enter billing and shipping information once and to thereafter easily purchase products with just one click of the mouse. According to Amazon, its 1-click feature is covered by U.S. Patent No. 5,960,411, which just recently issued on September 28, 1999. Amazon filed the suit in a federal court in Seattle on October 20, 1999. Amazingly, U.S. District Court Judge Marsha J. Pechman granted Amazon a preliminary injunction against Barnes & Noble, forcing Barnesandnoble.com to remove the infringing functionality. Barnesandnoble.com appealed the decision. The fight goes on.

Thus, the State Street Bank decision and recent developments in Internet related patent battles directly challenge the widely prevalent assumption of most software developers that best way to protect business software is to rely on trade secret and copyright protection. Business entities that have significant investment in software assets, either as developers or as users, should take proper steps to seek appropriate intellectual property protection for their innovative software products, both to gain a competitive advantage and to preserve their freedom to operate in the e-commerce space.

4. CONCLUSION

Much debate regarding software patents continues within the U.S. Patent Office, the federal government in general and the public at large. For instance, in order to legislate a new prior user rights defense to business method patents in view of the recent State Street Bank and AT&T Excel decisions, the U.S. House of Representatives recently passed H.R. 1907, which would provide a limited defense to these types of patents. While many may question the appropriateness of patent protection for software, in terms of the appropriateness of the incentives it provides to software developers, few question the impact that software patents are having on the software marketplace. For software patent owners, software patents represent a worthwhile way in which to protect the fruits of their labors. For would-be infringers, software patents represent a nuisance at best, and at least an impediment to the expansion into new or existing marketplaces. For everyone, software patents represent powerful legal rights that one cannot ignore. Those who fail to recognize the importance of software patents -- either as protection of their proprietary knowledge or as a barrier to their use of the proprietary knowledge of others -- undoubtedly do so at their own peril.