Questions and Answers About Software Protection

There is no end in sight to the rapid evolution of the market for software. In light of the hectic pace necessary simply to keep up with the market, the last thing many software developers, sellers, resellers or end-users may wish is to become enmeshed in the legal technicalities of software protection. That viewpoint is shortsighted, however, for two reasons.

First, it is always less expensive to do things right the first time, and taking care of these legal issues as they arise in the natural course is much easier than trying later to reconstruct or recall matters after documents have disappeared (or were never created) and memories have faded.

Second, addressing legal issues initially is all the more important for the software developer. The ultimate success of a software project, as measured by lasting economic value, often depends on being able to prove that the property rights in your software do, in fact, belong to you. The more clear and uncontestable the evidence of your property rights, the more easily you will be able to attract other investors, the less likely you will have expensive disputes with others over the ownership of those rights, and the greater the likelihood that the outstanding features of your software will prove themselves in the marketplace.

Thus, these "legal technicalities" are not in addition to the everyday problems of running a business; they are part of these problems, and must be met like any other obstacle. The more that is known about the potential legal problems involved with software development and marketing, the better able one is to prevent those problems or meet them as they arise.

1. **What legal protection is available for software?**

Software can be protected legally in a number of ways. Protection of software stems from legal rights (including so-called "intellectual property" rights) associated with the software, and/or by a contract providing what can and cannot be done with the software.

There are four major types of intellectual property: patents, trademarks, copyrights and trade secrets. The interesting thing about software is that all four types of protection may apply to one piece of software. Each type of property serves a unique and special purpose and relates to distinct aspects of software. Although protection under all four types of intellectual property may not always be justified, each option should be explored carefully.

Keep in mind that a software copy on a disk, for example, is not *per se* intellectual property. Selling or licensing a *copy* of software may enable the buyer or licensee to use the copy, but does not necessarily transfer any of the underlying intellectual property rights to the buyer or licensee. By way of an analogy, a buyer of a book in a bookstore does not acquire the copyright simply by buying the book.
2. **How is software copyrighted?**

Copyright in software arises automatically at the moment an original computer program is fixed in some tangible medium, such as on paper or electronically. No registration is needed to obtain the copyright itself.

Nevertheless, registration of a copyright with the U.S. Copyright Office provides important legal benefits and is ordinarily advisable. Generally, registration is a prerequisite to filing a copyright infringement suit. Also, special legal benefits can be obtained with respect to infringements which occur after (but not before) registration. Registration consists of completing a two-page form and sending the form and a nominal fee (currently $30) to the Copyright Office. Although the copyright registration process is not as involved as the patent prosecution or trademark registration processes, certain special requirements pertain to computer software. You can obtain additional information about copyrights through the U.S. Copyright Office’s website at http://www.copyright.gov, or by calling the U.S. Copyright Office at (202) 707-3000.

Placing a copyright notice on your software is *highly* advisable. One of the benefits of including a copyright notice is that an infringer cannot claim lack of knowledge that the software was copyrighted. The following is one form of a copyright notice:

```
© [NAME OF COPYRIGHT OWNER] 20xx
```

BEWARE: for published works, the year should indicate the year the software was first published -- that is, distributed to the public. Also, "(C)" is not a substitute for the © symbol. If for some reason the © symbol cannot be used, use the word COPYRIGHT rather than (C).

Depending on when a work was created, a copyright generally lasts for 70 years after the death of the last living author; or, if the computer program is a special type of work called a "work made for hire," the copyright lasts 120 years from the creation of the work or 95 years from the publication of the work, whichever period ends first.

3. **Can my company obtain a copyright when parts of the software are to be created by others?**

Yes. A copyright must vest in the name of the author or authors of the software. If an employee or employees of your company create the software as part of their job responsibilities, then the company -- the employer -- is considered the author for copyright ownership and registration purposes. A written employment contract is a good way to define an employee's job responsibilities.

If the software is created by someone other than an employee of the company as part of their job responsibilities, such as an independent contractor, a written contract should be entered into between the company and the independent contractor that specifies how ownership rights are divided between the company and the contractor.
A work created by more than one person or entity with the intention that authorship in the work is to be shared is known as a joint work, and it is registered in the names of all authors. Unless otherwise agreed upon, each of the authors may exploit the rights in the work and must account proportionately to the other authors for all profits received.

4. Is copyright enough protection for software?

Perhaps, depending on what you're trying to accomplish. Although copyright provides very important legal protection for software, it does have limits. Copyright does not extend to any idea, method or system underlying the software. The particular description, explanation, or illustration of the idea or system (generally referred to as the "expression") is what is protected. Further, copyright (unlike a patent) does not protect against independent creation. So, the independent creation of software that is substantially similar to your software does not infringe your copyright.

In general, the software copyright owner can stop others from reproducing (copying), deriving (adapting, translating or modifying), distributing, performing, displaying and renting the software, except as permitted by U.S. copyright law. Protection extends not just to software code (source code and object code), but also covers audiovisual, pictorial and graphic elements present in the user interface. The law concerning the scope of copyright protection is still evolving.

5. When is software patentable?

A computer program invention (which we will call "software" for the purposes of this question) is patentable if it meets certain requirements. In general, software must amount to more than merely an unapplied mathematical algorithm and must be "new," "useful" and "nonobvious" to qualify for "utility" patent protection. With a utility patent, one can exclude others from making, using or selling the patented software for a 20-year period from the earliest priority date of the patent.

Patent protection can provide very powerful legal protection for software. It provides stronger protection than other forms of intellectual property since, in general, software that performs the same work as the patented software in substantially the same way and accomplishes the same result would infringe the patent. In view of the cost and time required to obtain a patent, careful analysis should precede any decision to patent software. In addition, some foreign countries place significant restrictions on the scope of patent protection available for software.

BEWARE: Some uses of the software (such as publishing a Web page regarding its operation or offering the software for sale) starts a one-year period in which to file a U.S. patent application. Patent protection is barred thereafter. Moreover, patent protection in some foreign countries may be barred immediately upon a these types of uses.

Obtaining a software patent normally involves the following steps: performing a patentability search (optional, but usually advisable); preparing and filing a patent application; and convincing the U.S. Patent and Trademark Office that you are entitled to
a patent. Patent protection is not automatic. Patent applications are routinely rejected for failure to meet very strict legal requirements. It is also important to remember that most patent rights do not become effective until the patent issues, which may be years after you file your application. You can and should use "patent pending," "patent applied for," or abbreviations thereof, in connection with the software after (but not before) filing a patent application. You can obtain additional information about patents on the Internet at www.uspto.gov.

6. How can trademarks be used to protect software products and services?

In general, a mark is any word, name, symbol or design, or any combination of these, adopted and used to identify goods or services and which distinguishes those goods and services from those of others. A mark used to identify goods is called a "trademark" and a mark used to identify services is called a "service mark." The term "trademark" is often used to refer to both types of marks.

A name or logo used to identify the source of a software product or service may qualify as a trademark or service mark, respectively. In addition, distinctive elements of a user interface also may qualify, such as an icon or other symbol incorporated into the user interface. Further, some nonfunctional elements of both the packaging of the software product and the overall user interface may qualify as something called "trade dress" and therefore be eligible for legal protection.

7. Should I register a trademark with the government?

Unlike patent rights, trademark rights may be created without filing any paperwork with the federal or any state government. These rights are called "common law" rights and are created by using the mark in commerce to identify the source of goods or services. Nevertheless, the registration of a trademark with the federal and/or state governments provides important legal benefits and is ordinarily advisable. For example, federal registration on the Principal Register provides the following benefits:

- constructive notice to the public of the registrant's claim of ownership of the mark;
- a legal presumption of the registrant's ownership of the mark and the registrant's exclusive right to use the mark nationwide on or in connection with the goods and/or services listed in the registration;
- the ability to bring an action concerning the mark in federal court;
- the use of the U.S registration as a basis to obtain registration in foreign countries; and
- the ability to file the U.S. registration with the U.S. Customs Service to prevent importation of infringing foreign goods.

It is possible to start the federal registration process before actually using a mark; however, the mark must thereafter be used in commerce in order for registration to occur.
Registering a trademark normally involves the following steps: performing a trademark search (optional, but usually advisable); preparing and filing a (federal and/or state) trademark registration application; and convincing the applicable governmental office that you are entitled to registration. Federal registration allows one to use the ® symbol with the registered mark. The use of the ® symbol in conjunction with a mark not registered with the federal government is a violation of federal law.

8. What are trade secrets?

In general, a trade secret relates to information and includes any formula, pattern, device or compilation of information that is used in one's business and which gives an opportunity for competitive advantage. One of the most famous examples of a trade secret is the formula for Coca-Cola, which has been guarded for years. To qualify as a trade secret, (1) the owner must take reasonable measures to keep the information secret; and (2) the information must derive independent economic value from not being generally known to the public. Trade secret protection does not protect against independent creation and may not be effective against persons who innocently acquire the trade secret information.

Software may contain a number of trade secrets. For example, the executable version of the software may perform a subtask that is part of a larger task that is itself a trade secret. Also, the software source code is almost always considered a trade secret since underlying algorithms and various programming methods are discoverable upon access to the source code. Further, the software may directly include, or indirectly access, database information that is highly confidential and may constitute a trade secret.

In determining whether information constitutes a protectable trade secret, the North Carolina courts look at six factors: (1) the extent to which the information is known outside the business; (2) the extent to which it is known to employees and others involved in the business; (3) the extent of the measures taken to guard the secrecy of the information; (4) the value of information to the business and its competitors; (5) the amount of effort or money expended in developing the information; and (6) the ease or difficulty with which the information could properly be acquired or duplicated by others.

9. How can I maintain protection of my software as a trade secret?

You must make reasonable and affirmative steps to protect your trade secrets both within the company, with your employees, and outside the company, with your customers. The specific measures adopted for customers will depend in large part on the type of software and its manner of distribution. Mass marketed software, for example, is typically distributed in object code form only, often with a so-called "shrink wrap" license. On the other hand, transactions involving delivery of source code to an end user ordinarily merit a signed, written license agreement. In past cases, the North Carolina Court of Appeals has held that where software is distributed only in its object code form and the source code is not available to the general public, and access to the software is limited, it may be properly deemed a trade secret. However, in cases where the software was readily reverse engineered, the courts have typically declined to find the existence of a protectable trade secret.
In order to minimize the possibility of disclosure of trade secret information by employees or others who may obtain access to such information, and consequent loss of the trade secret, a protection program should include at least the following procedures: restricted access to the software and related materials (disclosures on a "need-to-know" basis only, physical security systems, etc.); placement of proprietary and confidentiality legends on all materials containing trade secrets; nondisclosure agreements for employees, visitors and outside consultants; performing exit interviews with departing employees reviewing their obligations to protect the trade secrets; and restrictive license agreements. Specifically, companies may also control software source code by limiting access to narrow pieces of code based on need; monitor internet activity and limit file export sizes; copyright code and insert identifiers into code strings; and restrict and monitor access to labs.

Sometimes the duty to keep a trade secret confidential is automatic. An employee privy to sensitive business information may have a legal duty not to disclose the information, even though no written contract says so. It is ordinarily advisable, however, to have written confidentiality agreements in place with employees and others who come into contact with trade secrets.

The North Carolina Trade Secrets Protection Act, prohibits the acquisition, disclosure or use of confidential business or technical information (including formulae, programs, data, techniques and the like) as long as the rightful owner takes reasonable precautions under the circumstances to safeguard these trade secrets. Some safeguards may include marking the information as a trade secret, storing it in a limited access area, limiting the number of copies available and keeping a log of those who actually use or view each copy of the information.

In order to show that a trade secret has been misappropriated, the plaintiff must show that (1) the defendant knew or should have known of the trade secret (thus emphasizing the importance of marking and identifying trade secrets); and (2) that the defendant has had a specific opportunity to acquire the trade secret. It is also important that any suit for misappropriation of trade secrets be brought within three years of the time when the misappropriation is or reasonably should have been discovered. Successful litigants may be able to enjoin others from use of their trade secrets, or may collect monetary damages.

10. How do I ensure that the techniques that give my software a competitive advantage remain confidential within my company?

Safeguarding valuable information or techniques that are not generally known is of prime importance to a developing technology company. Employers usually protect this confidentiality through employment contracts with their key personnel and by taking reasonable steps to keep the information private, so that it qualifies as a trade secret under the law of North Carolina.

The North Carolina Trade Secrets Protection Act prohibits the acquisition, disclosure or use of confidential business or technical information (including formulae, programs, data, techniques and the like) as long as the rightful owner takes reasonable precautions under the circumstances to safeguard these trade secrets. Some safeguards may include marking the information as a trade secret, storing it in a limited access area,
limiting the number of copies available and keeping a log of those who actually use or view each copy of the information.

11. How do I keep my employees and contractors from using what they learn from me to compete against me?

One way this protection may be obtained is by stating in an initial written employment contract that the employee is prohibited from using your confidential information to compete against you. In addition, the employee can be prohibited from disclosing publicly your confidential information, and even precluded from contacting your customers for a certain amount of time after employment ends.

These and other restrictions in an employment agreement must be drafted carefully so that they meet, but not exceed, your legitimate business interests. North Carolina law contains several technical requirements for the validity of these agreements, and courts do not hesitate to strike down overly restrictive employment agreements, especially those that seek to restrict unreasonably in time or in location--a worker's right to seek employment elsewhere.

A company generally is less constrained in negotiations with independent contractors, since it is assumed that these commonly are more "arms-length" transactions than a company's dealings with its own employees. Covenants not to compete, however, are nevertheless strictly construed and must be drafted carefully to be enforceable.

12. How do I ensure that what my employees or contractors bring with them from their former employment does not infringe the rights of others?

Just as your own employees owe a duty to you not to disclose your confidential information if they decide to work elsewhere, new potential workers similarly may have prior existing duties to their former employers. When you hire these workers, you may be exposing you and your company to claims by the worker's former employer that you have aided that employee in misappropriating the former employer’s confidential or trade secret information.

Consequently, the safest course of action is for you to talk with each prospective employee before you hire the employee to understand whether he or she may have obtained confidential information from a former employer. Obviously, not everything learned from the former employer will be considered confidential. In addition, your inquiry should not be limited to determining whether the employee has a “written contract” with the former employer. Although written "non-disclosure" agreements are preferable, an employee nonetheless has a duty to keep secret information that was disclosed to him or her in confidence even though no employment agreement was formerly memorialized. If you have any questions concerning the extent of the obligations a worker may have to a former employer, it is advisable to consult an attorney immediately. This is particularly important if you and the former employer are competitors.
13. How do I keep my employees and contractors from claiming rights in the software they develop or work on for me?

Who owns what rights in software is best answered before the software is developed. A written agreement, which can be included in an employment contract, can set forth what intellectual property rights in any software developed for you will be owned by you as the employer, jointly by you and the worker, or entirely by the worker.

In situations where no written agreement has been made, the rights that a worker can claim in software developed by you can depend upon whether the worker is classified as an "employee" or "independent contractor." Determining which classification applies in a given situation is not always easy and requires careful legal analysis.

In general, a worker is an "employee" if the employer has the power or right to control and direct the employee in the details of how the work is performed. Employees usually work out of the employer's premises and for a set salary.

An independent contractor, on the other hand, is someone who performs assigned tasks according to his or her personally preferred methods and is subject to the control of the party engaging the contractor only as to the end product or final result of the work. Independent contractors may receive compensation only on a project basis and work out of their own offices or homes. Independent contractors, again absent a specific written agreement providing otherwise, are considered the author of the software they develop and, therefore, own the copyright for that software.

Claims by employees to rights in software usually come from three sources: works developed prior to their employment; works developed while employed but unrelated to their job duties; or works developed after they terminate their employment. With regard to the first category, each new employee should be asked to describe briefly, in writing, all software projects on which they have worked and believe they may be entitled to development or ownership rights. Be sure to retain this written statement as a record of the employee's claims to rights in the software existing as of the time of initial employment.

As to software developed by an individual while employed, copyright law provides that the employer is considered the author of its employees' works created in connection with the employees' job duties. Therefore, be sure to maintain detailed records of each employee's projects, assignments and responsibilities. These records will aid in proving that software that the employee later claims was developed by him or her before or after employment, was actually developed for you during the employee's employment. Finally, as to claims to rights in the software based on development work after ending employment, an exit interview will help to lessen potential future claims by the employee. The interview should include discussions of the employment agreement signed at the beginning of employment and the extent of development work done while employed by the company. Having a written understanding and good records about what was accomplished during employment is the best way to avoid differing interpretations once the employee leaves the employer.
14. **How do I allow potential investors to evaluate my software without permitting them to steal it or to disclose too much to others?**

You should be hesitant to disclose too much about your software to others without having some agreement in place to limit how they may use what they see and to bind them to obligations of confidentiality. The terms of this agreement often depend on the relative bargaining power of the parties. For example, a larger company, with far-flung and often uncoordinated development and acquisition efforts, simply may refuse to review or accept investment ideas or opportunities without a written acknowledgment that it has no duty to keep confidential the information it receives from a developer, even though it has no intention to disclose any of it.

On the other hand, many companies -- particularly if they have a specific interest in your software -- will agree to a nondisclosure and non-use agreement before receiving any confidential information about your software. To protect the developer, this agreement should include a general description of the software involved, the type of disclosure to be made, an acknowledgment of the confidentiality and value of the software, a limitation on the use and disclosure of the software by the potential investor, a reservation of all your property rights, and an agreement to return the original and all copies of whatever is provided for evaluation or inspection.

Whatever sort of agreement the potential investor is willing to enter into, the important thing is to ensure that everyone has a clear idea of the parameters of the relationship before the disclosure occurs.

15. **How do I demonstrate to potential investors that I own the rights to my software and the extent of those rights?**

There is never a good substitute for adequate recordkeeping. Your goal should be to provide potential investors with an unbroken, step-by-step record of how you obtained the rights in each portion of your finished software development project. You should be able to show which of your employees and contractors worked on particular components of the software and, ideally, produce employment and development agreements signed by those employees and contractors acknowledging your company’s rights in the software. Do be sure to incur the relatively small expense and short amount of time it takes to register your copyright in the software, including all subsequent versions or revisions. Finally, obtain written assignments, licenses or other documentation of the transfer of any rights that others may have in portions of the software, in order to clearly demonstrate that you have the right to incorporate these portions into your finished product.

16. **How do I ensure that the software I license or distribute is not used illegally?**

If you are the owner of the copyright, copyright law generally prohibits your customers from making and distributing multiple copies of your software after they have received only one copy from you. In addition, copyright law prohibits the renting or leasing of software to others by a licensee or user. Violators can be sued for copyright infringement.
Most people, however, do not rely solely upon copyright law in distributing their software. Written contracts, usually in the form of license agreements, are the best method of establishing exactly what one is permitted to do or is restricted from doing with the software at issue. In addition to claims for copyright infringement, claims of breach of contract can then also be brought against the persons who use the software in violation of the written contract.

17. What other methods can I use to prevent people from illegally distributing or using my software?

Software developers have created many practical methods that encourage users to purchase legitimate copies of software. These include copy protection mechanisms and the providing of user manuals, software updates, and customer support only to authorized users. However, it should be emphasized that however much a software pirate may deserve punishment in the form of destroyed data or an unusable computer system, you should seek competent legal counsel before including hidden “time bombs” or "booby traps" in software that cause unanticipated and unbargained-for damage to what may be essential business or personal information and files. Any mechanism, such as product authorization codes or the like, that is required to use the software or prevents use of the software (or worse) on the occurrence, or non-occurrence, of some event should be agreed to by the user prior to the user's license.

18. How do I minimize my liability if the software I license or distribute does not work correctly?

Generally, your liability for bugs or defects in the software you distribute is governed by the terms of the license or distribution agreements entered into by you and the licensee of the software. It is important to remember that any statement about the software may be viewed as a warranty, even if it does not mention the words "warranty" or "guarantee," so you should be particularly careful about the description and claims made about the software in written materials and oral presentations.

Further, it is possible to disclaim all, or some, warranties and limit your liability for damages caused by your software. Often, the largest part of the expense when software malfunctions is not due to the cost of the software, but comes from other indirect or consequential losses. Examples include the cost of extra workers to complete a job manually that your software was intended to automate, and the expense of recovering lost data. You should also be sure to always include a provision in your license or distribution agreements stating that the software is not error-free.

The law permits your agreement to contain a disclaimer or limitation on damages to a certain maximum dollar amount (e.g., the amount paid for the software under the agreement), and to disclaim indirect and consequential damages.

If you are a software reseller or distributor, you nonetheless may be liable for defects in the software, unless the written agreements between you and the software vendor and between you and the licensee specifically state otherwise. Therefore, you should spell out clearly in those written agreements who shall be liable for defects in the
software and what remedies (refund, repair, replacement, etc.) are available if a problem occurs.

Not all terms and conditions which you may put in a contract are necessarily enforceable in court. The law on what is enforceable varies from state to state and sometimes depends on federal law.

19. **What protection do I have if I wish to distribute my software in other countries?**

Establishing intellectual property rights in the United States does not necessarily create any rights in foreign countries. Each nation has its own laws concerning patents, trademarks, copyrights, trade secrets and the distribution of software. Some of these laws in different countries are similar to those in the U.S. whereas others can be very different. Although there have been efforts to harmonize the intellectual property laws internationally, and some recent international treaties have helped, significant inconsistency still exists. Some countries may place tight restrictions on the distribution of foreign software or may exclude software from protection altogether. For example, questions have been raised in Europe as to whether software should be patentable at all. Few generally applicable principles can be formulated, except that there is no substitute for understanding the laws and customs of the particular country into which distribution is considered prior to any activity within that country.

20. **How do I minimize the damage from mistakes made in the past in failing to protect my rights?**

Sometimes it is just impossible to recover completely from past mistakes. For example, public disclosure of confidential information ends trade secret protection. Also, certain activities can permanently prevent patent protection. In other cases, however, the damage is reversible with reasonable effort. An excellent method to halt the further erosion of rights is to conduct an intellectual property "audit."

First, identify what intellectual property you own and what rights you have in each piece of property. Second, determine what steps need to be taken to protect each of those rights best. Last, implement the steps or procedures required to secure the rights.

This audit method depends upon the software owner's willingness to invest a fair amount of effort and expense to determine and implement the necessary procedures—whether the task involves registering the copyright, obtaining a confidentiality or other employment agreement, entering into a license or other transfer agreement—but the ultimate cost will only grow greater as the delay gets longer.
Summary

The following table summarizes the four major types of intellectual property.

From left to right, the columns in the chart summarize:

the individual types of Intellectual Property (i.e., Patents, Trademarks, Copyrights and Trade Secrets);

the basic rights afforded by each type of intellectual property ("Right to exclude others from" doing certain acts);

the "Thing" protected by each type of intellectual property (i.e., an invention, a mark, a work of authorship and information, respectively);

the Types of each intellectual property (e.g., types of patents include Design Patents and Utility Patents);

an Example of how each type of intellectual property might apply in a computer industry context (e.g., a Design Patent might be employed to protect the ornamental design of an icon, and a Utility Patent might be employed to protect a user interface invention or an expert system invention);

the Duration in Years of each type of intellectual property (note that the duration of a Design Patent is 14 years from the date of issue of the patent, whereas the duration of a Utility Patent is 20 years from the earliest priority date of the patent);

the basic Requirements for protection; and

a Sample Notice for each intellectual property.

Chart on next page.
<table>
<thead>
<tr>
<th>INTELLECTUAL PROPERTY</th>
<th>Right to exclude others from</th>
<th>“Thing”</th>
<th>Types</th>
<th>Example</th>
<th>Duration in Years</th>
<th>Requirements</th>
<th>Sample Notice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patent</td>
<td>making, using or selling</td>
<td>invention</td>
<td>Design, Utility</td>
<td>Icon, user interface, expert system</td>
<td>14 years</td>
<td>new ornamental nonobvious, new useful nonobvious</td>
<td>Design Pat. No. 123,456</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>20 years from earliest priority date</td>
<td></td>
<td>U.S. Pat. No. 1,234,567 Pat. Pend.</td>
</tr>
<tr>
<td>Trademark</td>
<td>using confusingly similar</td>
<td>mark</td>
<td>Trade, Service</td>
<td>Software product, Programming Services</td>
<td>until abandoned*</td>
<td>distinctive, actually used</td>
<td>Microsoft® software Windows™ software IBM® services Prodigy™ services</td>
</tr>
<tr>
<td>Copyright</td>
<td>reproducing, deriving, distributing, performing or displaying</td>
<td>work of authorship</td>
<td>Literary, Graphic, Audiovisual</td>
<td>Computer program, screen graphic, video game</td>
<td>120 years from creation or 95 years from publication</td>
<td>original, fixed</td>
<td>©IBM Corp. 1993 ALL RIGHTS RESERVED</td>
</tr>
<tr>
<td>Trade Secret</td>
<td>misappropriating</td>
<td>information</td>
<td>Formula, Pattern, Device, etc.</td>
<td>algorithm, circuitry, widget, source code</td>
<td>until disclosed</td>
<td>valuable, confidential</td>
<td>CONFIDENTIAL</td>
</tr>
</tbody>
</table>

* Federal registration lasts for 10 years, renewable in 10-year increments. State registration varies depending on state law.

Intellectual Property Law Section
North Carolina Bar Association