frdm

Software Freedom
Law Center

1995 Broadway, 17th Floor
New York, NY 10023–5882
tel +1–212–580–0800
fax +1–212–580–0898
www.softwarefreedom.org

Originality Requirements under U.S. and E.U. Copyright Law

27 September 2007

1 Executive summary

Understanding how copyright law applies to software is essential to making informed decisions about how to develop and distribute code. The first step of this analysis is to consider whether the code in question is in fact copyrightable. This document discusses the requirements for software to be considered within the scope of copyright under U.S. law. Because open source and free software development is very often conducted worldwide, a brief discussion of the applicability of copyright laws in the European Union is also included.

The originality requirement under both U.S. and E.U. copyright law is minimal, such that courts have ruled computer programs insufficiently original to be eligible for copyright protection in only a very small number of cases. While the originality standard is low, it does exist. In particular, the laws stress that it is a programmer's *expression* of some functionality that may be protected by copyright, and not the functionality itself. If code embodies the only way (or one of very few ways) to express its underlying functionality, that code will be considered unoriginal because the expression is inseparable from the functionality. Similarly, if a program's expression is dictated entirely by practical or technical considerations, or other external constraints, it will also be considered unoriginal. The originality of a program's functionality is irrelevant to its eligibility for copyright. Code implementing a completely novel algorithm may not be copyrightable due to a dearth of expressive alternatives. Meanwhile, code which behaves identically to other existing programs may be copyrightable because of the original expression of its implementation. A program is also unoriginal to the extent that its expression (but not ideas or functionality) is taken from public domain or other existing code.

In addition to protecting wholly original programs, both E.U. and U.S. copyright laws protect programs based in part on preexisting code (i.e. "derivative works"), insofar as the resulting work incorporates an original expression of the later contributor. The size of the contribution in relation to the underlying work is irrelevant. Original contributions are copyrightable and unoriginal contributions are not. Furthermore, it is the contribution's *expression*, not its *functionality*, which must be original.

2 The U.S. copyright standard: originality plus fixation

No objective minimum amount of content is required for a work to be included within the scope of copyright. The Copyright Act defines only two requirements for copyrightability: original authorship ("originality") and fixation.[1] Though there remains controversy regarding the application of the fixation requirement to software

[1] 17 U.S.C. §102(a).

in some forms (in particular, with regard to copies in RAM [2]), source code in most incarnations is considered sufficiently "fixed" within the meaning of the statute.[3] Thus for the purposes of this analysis, fixation is not at issue, and the only express barrier to copyrightability is the originality requirement.

3 The originality standard (*Feist*)

In the U.S., originality is a constitutional requirement for copyright applicability, though it was first stated explicitly by statute only with the introduction of the 1976 Copyright Act.[4] "[T]he originality requirement is not particularly stringent," and, as discussed in the U.S. Supreme Court case *Feist Publications, Inc. v. Rural Telephone Service, Co.* which considered the copyrightability of an alphabetically organized phone book, is comprised of two elements: "that the work was independently created by the author (as opposed to copied from other works), and that it possesses at least some minimal degree of creativity."[5] A work satisfies the "independent creation" element so long as it was not literally copied from another, even if it is fortuitously identical to an existing work.[6] The "creativity" element sets an "extremely low" bar that is cleared "quite easily" by "[t]he vast majority of works."[7] It requires only that a work "possess some creative spark, 'no matter how crude, humble, or obvious it might be."'[8]

3.1 Originality and merger

The originality requirement is designed to withhold copyright not only from works that are too similar to existing original works, but also to prevent authors from appropriating that which is considered inherently unoriginal, hence uncopyrightable. Thus, the Copyright Act explicitly excludes from protection "any idea, procedure, process, system, method of operation, concept, principle, or discovery."[9] For example, when a work's "expression is essential to the statement of the idea" embodied therein, the author's expression of that idea is said to "merge" with the idea itself, and is rendered uncopyrightable.[10] This is so even if the idea, process, or other type of uncopyrightable element expressed is "novel" in the sense that it has never before been expressed.[11] Merger applies where the author's expression is the only way, or "one of very few ways," of expressing some uncopyrightable element.[12]

When a work contains both original and unoriginal aspects, copyright protects the work but "is limited to those aspects of the work—termed 'expression'—that display the stamp of the author's originality."[13] The Supreme Court established the framework for analyzing such hybrid works in *Feist.*[14] Though *Feist* dealt in particular with mixed factual and expressive works, it applies with equal force to other hybrid works, and has been applied as such by lower courts.[15]

[2] *See* R. Anthony Reese, Assistant Professor, School of Law, The University of Texas at Austin, Address at the University of Illinois Law Review Symposium: The Public Display Right: The Copyright Act's Neglected Solution to the Controversy Over RAM "Copies" (2001).

[3] Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1243 (3d Cir. 1983).

[4] Feist Publ'ns, Inc. v. Rural Tel. Serv. Co., 499 U.S. 340, 355 (1991).

[5] *Id.* at 358–359.

[6] *Id.*

[7] *Id.* at 349.

[8] *Id.*

[9] 17 U.S.C. §102(b).

[10] Lexmark Int'l, Inc. v. Static Control Components, Inc., 387 F.3d 522, 535 (6th Cir. 2004). The earliest and quintessential expression of the merger doctrine is found in the Supreme Court's 1880 decision in Baker v. Selden, which denied copyright in a ledger sheet which merely embodied the system of accounting described in the plaintiff's book. Baker v. Selden, 101 U.S. 99 (1880).

[11] *See* Lexmark at 20.

[12] *See* Lexmark at 22.

[13] Harper & Row, Publrs. v. Nation Enters., 471 U.S. 539, 546 (1985).

[14] 499 U.S. at 342.

[15] *See, e.g.*, Computer Assocs. Int'l v. Altai, Inc., 982 F.2d 693, 711 (2d Cir. 1992).

Feist recognized that an author's original contribution to a hybrid work may comprise two types of expression: an author may "clothe[] facts with an original collocation of words. . . [and] claim a copyright in this written expression," and she may also "select[] and arrange[]" uncopyrightable elements in an original manner and claim copyright in that selection and arrangement.[16] *Feist* emphatically reiterated that mere collections of unoriginal elements are not copyrightable, stated that only a "thin" copyright subsists in "original selection[s] or arrangement[s]" of unoriginal elements,[17] and established that some arrangements undertaken by (and thus in a sense originating with) the author will nonetheless be "so mechanical or routine as to require no creativity whatsoever."[18] The Court held that alphabetical arrangement was such an arrangement, and described generically the indicia of unoriginal selections and arrangements—in addition to being "mechanical or routine," they are "entirely typical," of the "garden-variety," and "could not be more obvious."*Id.*

4 The originality standard and derivative works

The Copyright Act expressly recognizes copyright in derivative works, but this right covers only the original contribution of the author who prepares the derivative work, and does not extend to any portion of the underlying work upon which it is based.[19] Put differently, the later contribution must meet the standard for copyright independently, and thus must itself be original.[20] Beyond this minimum, some circuit courts of appeals have set the standard for copyrightability of derivative works higher than that for non-derivative works.

4.1 Non-trivial originality (*Durham*)

In *Durham Industries, Inc. v. Tomy Corp.*,[21] the Second Circuit Court of Appeals stated that copyright's applicability to derivative works is subject to the limitation that "the original aspects of a derivative work must be more than trivial."[22] This language derived from the court's earlier statement of the originality standard in the seminal case *Alfred Bell & Co. v. Catalda Fine Arts, Inc.*[23] However, the *Alfred Bell* court considered non-trivial variation to be a requirement of the originality standard, and merely a restatement of the principle that, to be original, a work must be "recognizably [the author's] own."[24] This accords with the originality standard as later stated by the Supreme Court in *Feist*, which held that the "minimal level of creativity" requirement would be met by any work in which "the creative spark is [not] utterly lacking or so trivial as to be virtually nonexistent."[25] The *Durham* court, in contrast, read non-triviality to be a separate bar to copyrightability of derivative works, requiring contributions to be both "original" *and* "non-trivial." While this formulation seems redundant upon review of the originality standard, courts and commentators have read *Durham* to establish a separate non-triviality requirement for derivative works.[26]

If a derivative work must satisfy two separate non-triviality standards, the question is raised as to what

[16] 499 U.S. at 348–349.

[17] *Id.* at 349. "Where the compilation author adds no written expression but rather lets the facts speak for themselves. . . . [t]he only conceivable expression is the manner in which the compiler has selected and arranged the facts." *Id.*

[18] *Id.* at 362.

[19] Durham Industries, Inc. v. Tomy Corp., 630 F.2d 905, 909 (2d Cir. 1980).

[20] Waldman Publishing Corp. v. Landoll, Inc., 43 F.3d 775, 782 (2d Cir. 1994).

[21] 630 F.2d 905 (2d Cir. 1980).

[22] *Id.* at 909.

[23] 191 F.2d 99 (2d Cir. 1951).

[24] *Id.* at 102–103

[25] 499 U.S. at 358.

[26] *See, e.g.*, Leo J. Raskind, Professor of Law, University of Minnesota, Address at the University of Pittsburgh Law Review Symposium: The Uncertain Case for Special Legislation Protecting Computer Software (1986) (citing *Durham* for the proposition that "A somewhat higher standard of merit is required for copyright protection of derivative works"); Entm't Research Group v. Genesis Creative Group, 122 F.3d 1211 (9th Cir. 1997) ("[T]he Durham test. . . ensure[s] that copyright protection is not given to derivative works whose originality is merely trivial.").

circumstances will a derivative work author's contribution be non-trivial under the originality standard yet too trivial a variation from the original for copyright eligibility. The cases citing the *Durham* test are largely consistent with pre-*Durham* case law, implying that there is no difference between variations which are original (hence non-trivial)[27] and variations which are non-trivially original.[28]

The *Durham* test has only been applied to a derivative computer program in a single unpublished opinion. In *E.F. Johnson Co. v. Uniden Corp. of America*[29], a Minnesota District Court cited *Durham*'s non-trivial variation standard in determining whether an application derived largely from public domain components contained sufficient originality for copyright. Though the court found that the algorithms and data structures used in the plaintiff's radio firmware were based on "textbook" techniques, it found that the "degree of originality contributed by plaintiff to [its implementation of them] is far from 'trivial."'[30] Specifically, it found the following elements probative evidence of non-trivial originality:

1. the arrangement of bits in a 7-bit binary code commonly used in radio pulse compression (the plaintiff inverted the string and flipped the bits),
2. a 56-bit table representing the output of the public-domain Barker correlation function as sampled at eight regular intervals, and
3. a binary matrix comprised of the output of a pseudo-random number generator (and which, the judge noted, could have been "configured any of 32 different ways").[31]

These are highly questionable indicia of originality even if original is taken only to mean "having its source in the author," and leaving to one side the creativity requirement. Furthermore, *E.F. Johnson Co.* was decided before *Altai*,[32] and the features identified by the judge as non-trivial original contributions would most likely be considered not copyrightable under *Altai* as elements "dictated by efficiency. . . [or] external factors," or as residing in the public domain.[33] For all of these reasons, the Minnesota District Court's interpretation of *Durham* can be considered incorrect under current law.

4.2 Substantial difference (*Gracen*)

A still higher standard was set by Judge Posner in the controversial[34] Seventh Circuit case *Gracen v. Bradford Exchange*.[35] Where a derivative work very closely resembles the original (and even when the derivative is rendered in a wholly different medium), the court's reasoning goes, it will be difficult for a court to determine whether a different derivative work was copied from the first derivative or is based upon the original.[36] In order to avoid this result, the court denied copyright to the first derivative work for its failure to exhibit "a sufficiently gross difference between the underlying and the derivative work to avoid entangling subsequent artists depicting the underlying work in copyright problems."[37]

The *Gracen* standard has never been directly applied to source code, so no guidance is available on what the appropriate application would be. *Gracen* involved a painting depicting a scene from a film, and the

[27] *See, e.g.*, L. Batlin & Son, Inc. v. Snyder, 536 F.2d 486, 487 (2d Cir. 1976) (Cast-iron toy bank in the public domain substantially copied in plastic form insufficient variation).

[28] *See, e.g.*, Winfield Collection, Ltd. v. Gemmy Indus., Corp., 147 Fed. Appx. 547 (6th Cir. 2005) (Three-dimensional crafts constructed from two-dimensional designs demonstrate only trivial variation).

[29] 1985 U.S. Dist. LEXIS 12800 (D. Minn. 1985).

[30] *Id.* at 1499.

[31] *Id.*

[32] *See infra* notes 49–61 and accompanying text.

[33] *See* Computer Associates Intern., Inc. v. Altai, Inc., 982 F.2d 693, 707–710 (2d Cir. 1992).

[34] *See, e.g.*, Steven S. Boyd, Deriving Originality in Derivative Works: Considering the Quantum of Originality Needed to Attain Copyright Protection in a Derivative Work, 40 Santa Clara L. Rev. 325, 362–365

[35] 698 F.2d 300 (7th Cir. 1983)

[36] *See id.* at 304.

[37] *See id.* at 305.

court's hypothetical "subsequent artist" refers to another painter who wishes to depict the same scene. An analogous situation within the software development context might involve two developers wishing adapt an existing application to a different programming language or coding paradigm. Applying *Gracen* to the issue of whether the first developer's adaptation was sufficiently original to qualify as a derivative work, a court might focus on the likelihood that a later developer undertaking the same sort of adaptation would produce a work confusingly similar to the first developer's.

5 Originality as applied in software copyright infringement cases

There generally have been two types of cases which have considered the originality of some specific piece of software. The first line of cases (*Sega*, *Lexmark*) involves the literal copying of a program or other portion of code so insubstantial that it is questionable whether it could possibly contain expressive content or structure. The other line of cases (*Whelan*, *Altai*) contemplates the originality of structural and other non-literal components from programs whose literal components are clearly or concededly copyrighted.

For programs between these two types of cases that are of some minimum length or complexity, the originality of the program's literal components becomes a foregone conclusion. However, apart from early cases establishing the copyrightability of source code as a general proposition, there are no federal cases on record which rule directly that a program's literal components are sufficiently substantial to deserve protection. The question of how much literal content a program must contain to qualify for copyright, however, was addressed in the Sixth Circuit's decision of *Lexmark Int'l, Inc. v. Static Control Components, Inc.*[38]

5.1 Minimum standard for copyright in literal elements (*Lexmark*)

The *Lexmark* court identified two doctrines—the merger doctrine discussed *supra*, and the *scenes a faire* doctrine—as the appropriate tools for determining how little expression was too little. Applying the merger doctrine to source code, the court stated that "if the [program's unprotected underlying] process is embodied inextricably in the line-by-line instructions of the computer program, then the process merges with the expression and precludes copyright protection."[39] If a work represents one of only a few possible means of accomplishing a task, it is not copyrightable. The *scenes a faire* doctrine has its origins in narrative works, and means that expressions which are "standard, stock, or that necessarily follow from a common theme or setting" cannot be protected.[40] "In the computer-software context, the doctrine means that the elements of a program dictated by practical realities—e.g., by hardware standards and mechanical specifications, software standards and compatibility requirements, computer manufacturer design standards, target industry practices, and standard computer programming practices—may not obtain protection."[41]

If a portion of a program's expression merges with its underlying idea or is dictated by external technical considerations, that portion is not copyrightable under *Lexmark*. In applying this rule to code, the *Lexmark* decision directs courts to "ask whether the ideas, methods of operation and facts of the program could have been expressed in any form other than that chosen by the programmer, taking into consideration the functionality, compatibility and efficiency demanded of the program."[42] The court clearly implies that the capacity for originality in a computer program is to some degree a function of the program's size, stating that for a very large and complex program, "it would have been exceedingly difficult to say that practical alternative means of expression did not exist,"[43] and that a small program's "size. . . [may] dictate the content

[38] 387 F.3d 522 (6th Cir. 2004).
[39] *Id.* at 535.
[40] *Id.*
[41] *Id.*
[42] *Id.* at 554.
[43] *Id.* at 539.

of the. . . [p]rogram."[44]

Lexmark provides some specific guidance as to what sorts of available variations are insufficient to demonstrate a program's originality: that there exist "different ideas or methods of operation altogether" for achieving comparable functionality, because these are not "copyright-protectable expression;"[45] representing a formula with a look-up table;[46] and re-ordering constituent formulae in a manner analogous to paraphrasing.[47] More generally, the court exhibited disregard for conceivable variations that would be "trivial," and would not constitute "material changes" or "make any 'substantial difference' to the nature of the program."[48]

5.2 Minimum standard for copyright in non-literal elements (*Altai*)

The *Altai*[49] decision preceded *Lexmark*, and the two cases employ similar approaches to somewhat different issues. Whereas *Lexmark* involved the wholesale copying of an entire program, and turned mostly on whether the entirety of the program's expression merged with its functionality, *Altai* was concerned "only with [the program's] non-literal components" and whether its structure as copied by the defendants was sufficiently original to be afforded a selection and arrangement copyright. In *Altai*, the Court of Appeals for the Second Circuit expounded its "abstraction-filtration-comparison" test for ascertaining which structural or non-literal elements of a program are sufficiently original to qualify for copyright protection.[50]

5.3 The *Altai* Abstraction-Filtration-Comparison test

The Altai Abstraction-Filtration-Comparison is typically applied by using a three step process, each step of which is included in the name of the test.

5.3.1 Abstraction

The first step in *Altai*'s Abstraction-Filtration-Comparison test, Abstraction, requires a court to separate a program into conceptual layers or stages so as to evaluate the originality invested by the author in the design of each. The analysis begins at the "lowest" conceptual layer (the source code) and works to higher levels of generality, identifying the structural elements at each level and mindful that at some "point in this series of abstractions," the program's structure will "no longer [be] protected."[51]

[44] *Id.* at 541.

[45] *Id.* at 540.

[46] *Id.*

[47] *Id.*

[48] *Id.* at 539–540.

[49] Computer Associates Intern., Inc. v. Altai, Inc., 982 F.2d 693 (2d Cir. 1992).

[50] The court summarized the test:

> In ascertaining substantial similarity. . . a court would first break down the allegedly infringed program into its constituent structural parts[, t]hen. . . sift out all non-protectable material. . . , [then] compare [the remaining] material with the structure of an allegedly infringing program.

Id. at 706.

[51] *Id.* (quoting Nichols v. Universal Pictures Co., 45 F.2d 119, 121 (2d Cir. 1930).

5.3.2 Filtration

The second step, Filtration, defines a method of determining, at each level of abstraction, which elements of a computer program may be copied. Appropriable elements are broken down into three categories: those "dictated by efficiency," "dictated by external factors," or "taken from the public domain."[52] The first two relate closely to the merger doctrine.

Elements dictated by efficiency: *Altai* holds that when "efficiency concerns. . . so narrow the practical range of choice as to make only one or two forms of expression workable options," there is merger.[53] Consistent with the court's "abstraction" inquiry, this doctrine applies to design choices embodied at various layers of the application, from the structure of the code to the operation and visual layout of the interface.[54] The question central to this inquiry, according to the court, is "whether the use of *this particular set* of modules is necessary to efficiently implement that part of the program's process being implemented." If so, "it should be disregarded in the overall substantial similarity analysis."[55]

Elements dictated by external factors: By analogy to the *scenes a faire* doctrine[56], the *Altai* court directed that "elements dictated by external factors" should be "filtered out of the infringement analysis," noting that it would be "virtually impossible to write a program to perform particular functions in a specific computing environment without employing standard techniques."[57] The decision also lists "the mechanical specifications of the computer on which a particular program is intended to run; [] compatibility requirements of other programs with which a program is designed to operate in conjunction; [] computer manufacturers' design standards;[] demands of the industry being serviced; and [] widely accepted programming practices within the computer industry" as examples of "extrinsic considerations" which constrain "a programmer's freedom of design choice" and may negate an inference of copying.[58]

Elements taken from the public domain: In addition to functionally necessary elements, the Second Circuit directed in *Altai* that "elements taken from the public domain" should be excluded from the infringement analysis. The decision refers in particular to "computer program[s] that have entered the public domain by virtue of freely accessible program exchanges and the like" and "expression that is, if not standard, then commonplace in the computer software industry."[59] The structural elements of a program which survive filtration are those which are original.

5.3.3 Comparison

The third step, Comparison, directs the court evaluating an infringement claim to compare only these elements with the allegedly infringing program, to determine whether any were copied. If so, the court must then consider "the copied portion's relative importance with respect to the plaintiff's overall program" to determine whether the similarities are "substantial."[60]

[52] *Id.* at 707–710.
[53] *Id.* at 708.
[54] *Id.* at 708–709.
[55] *Id.* at 709.
[56] This doctrine also informs the court's third unprotected category, "elements in the public domain." *Altai*, 982 F.2d at 710.
[57] *Id.* at 709 (quoting 3 Melville B. Nimmer & David Nimmer, *Nimmer on Copyright* §13.03[F][3], at 13-65.).
[58] *Id.* at 709–10.
[59] *Id.* at 710 (quoting *Brown Bag Software v. Symantec Corp.*, 960 F.2d 1465, (9th Cir. 1992).
[60] *Id.* at 710–711.

5.4 Applying *Altai* to derivative works

Altai holds that "the non-literal structures of computer programs are protected by copyright."[61] It follows that a program which is derivative of and restructures another existing program will be eligible for a copyright to the extent that the structure it contributes is original within the meaning of the Copyright Act. *Altai*'s abstraction-filtration-comparison test can be applied to a derivative work, as to any other, to determine whether the new structural contributions are sufficiently original. However, when dealing with a derivative work, the entirety of the underlying work must be put to the side at the "filtration" stage of the analysis.

6 Size and substantiality: affirmative defenses to infringement

A work's size is examined in the context of two affirmative defenses to infringement. The first, the affirmative defense of fair use, does not implicate the infringed work's copyrightability. A court ruling on a fair use claim is required to consider, *inter alia*, the amount and substantiality of the portion of a work used by the defendant, in relation to the work as a whole. However, a finding of fair use assumes infringement (and by implication, copyrightability) and merely reflects the court's judgment that the infringement at issue was "fair" in light of the relevant factors.

The second infringement defense that implicates the size of the copied work is that of "*de minimis*" copying. The *de minimis* analysis touches on two of the factors in the fair use analysis: whether a substantial portion of the work was copied, and also whether the copying would "diminish the value of the original."[62] Some courts have held that *de minimis* copying is insufficient to constitute "substantial similarity," a necessary element for a finding of infringement, and thus would preclude a finding of infringement, unlike a finding of fair use.[63] The classic formulation of *de minimis*, however, involves "a technical violation of a right so trivial that the law will not impose legal consequences,"[64] and would suggest a finding of no liability despite a finding of infringement. Both theories have been applied by the circuit courts.

The leading case on *de minimis* copying of source code, *Dun & Bradstreet Software Servs. v. Grace Consulting, Inc.*,[65] provides little guidance in tailoring the doctrine to software. Pointing out that the quantitative amount of code copied (in this case, 27 out of 525,000 lines) is "irrelevant as a matter of law" and that the substantiality element refers to whether the copied code is qualitatively "material," the court found the copied portion was "highly critical" and dispensed with the *de minimis* claim.[66]

No federal court case on record has considered when a derivative work's original contribution may be so insignificant that in appropriating the entire derivative work, the author of the underlying work will be considered to have engaged in only *de minimis* copying. However, to apply the elements of the *de minimis* doctrine, the appropriation would be greater than *de minimis*, and infringing, if either of two tests were met:

1. if the contributions of the author who prepared the derivative work were a substantial portion of the derivative work taken as a whole, or
2. if the original author's appropriation of the entire derivative work diminished the value of the derivative work.

This latter part of the test implies that appropriation of a derivative work by the original author would

[61] *Id.* at 701.

[62] Mathews Conveyer Co. v. Palmerbee Co., 135 F.2d 73, 85 (6th Cir. 1943).

[63] Gordon v. Nextel Communs., 345 F.3d 922, 924 (6th Cir. 2003).

[64] Ringgold v. Black Ent. TV, 126 F.3d 70, 74 (2d Cir. 1997).

[65] 307 F.3d 197 (3d Cir. 2002)

[66] *See id.* at 208.

constitute more than *de minimis* copying if the additions of the later contributor were of any value at all.

7 Transnational considerations: European Union

In 1991, the European Community issued the Software Copyright Directive,[67] establishing unified legal standards for copyright in computer programs across the E.C. member countries, and largely harmonizing European law with that of the United States. The Directive established a single requirement that a program must meet in order to be eligible for copyright protection: it must be original. As in the U.S., the originality requirement is not a stringent one, and is met merely if the "program. . . is the author's own intellectual creation." The E.C. Software Directive expressly prohibits member countries from establishing any other criteria by which to determine software's eligibility for copyright. It is irrelevant, then, whether a program (or contribution thereto) is small or large; courts may consider only whether it is original in the sense that the author created it.

On a more international scale, the WIPO Copyright Treaty (WCT) and the World Trade Organization Agreement on Trade-Related Aspects of Intellectual Property Rights (TRIPS) have made explicit that the standards set forth in the Berne Convention for the Protection of Literary and Artistic Works are also applicable to software. The Berne Convention does not provide for a uniform originality test but refers to the laws as established by each country.[68]. Nevertheless, the Conventions make clear that no other requirements have to be met by software to be protected under international copyright law. In addition, the Berne Convention specifically states that "Translations, adaptations. . . and other alterations of a literary or artistic work shall be protected as original works without prejudice to the copyright in the original work."[69] This provision is also applicable to software. Therefore, derivative works of software are protected under the Berne Convention if they meet the normal standards for derivative works of literary and artistic works.

For E.U. signatories to the Berne Convention, these agreements ensure that works of software, including derivative works, containing the requisite originality will be eligible for copyright. Because the E.C. standard is so similar to that of the U.S., for practical purposes, originality under one indicates originality under the other. Consequently, copyright for software will be coextensive under both sets of laws.

[67] Council Directive 91/250, 1991 O.J. (L 122) 42 (E.C.) [*hereinafter* "E.C. Software Directive"].

[68] Berne Convention for the Protection of Literary and Artistic Works, Sept. 9, 1886, art. 2, para. 7, as last revised July 24, 1971, 25 U.S.T. 1341, 828 U.N.T.S. 221.

[69] Berne Convention for the Protection of Literary and Artistic Works, *supra* note 68, art. 2, para. 3.