When Efforts To Conceal May Actually Reveal: Whether First Amendment Protection Of Encryption Source Code and the Open Source Movement Support Re-Drawing The Constitutional Line Between the First Amendment and Copyright[*]

Rod Dixon[**]

I. INTRODUCTION

Computer source code *is* speech -- that is the argument advanced by privacy advocates and cryptographers, who recently obtained the support of the Ninth Circuit in *Bernstein v. United States.* Courts are in general conflict over the appropriate standard of review to apply to the government's regulations regarding the use or export of encryption products. That notwithstanding, among the courts that have considered the question, there is remarkable accord supporting the view that in some

respects, computer source code represents the expression of ideas by those who understand the arcane languages of computer programming. While the noteworthy relevance of this conclusion to copyright jurisprudence has not been fully absorbed by lawyers or the laity, one impact of viewing source code as speech is certain: it becomes palpably apparent that within the context of copyright infringement actions, the scope of copyright protection for computer source code should be thin, indeed.

In its sum and substance, the law of copyright both advances and encumbers the manner in which an author may express himself. Copyright is a constitutional promise granted by the Federal government that must yield, in instances of conflict, to the First Amendment. To that end, courts have engrafted onto the law of copyright a presumptively dispositive test often referred to as the idea/expression dichotomy.

In applying the idea/expression dichotomy, courts have often struggled to draw principled distinctions between copyrightable expression and the so-called basic ideas underlying copyrightable expression, particularly when the "expression" subject to a court's analysis is contained in computer programs. Many commentators have criticized the analytical limitations of the dichotomy, and others have argued that courts frequently misunderstand the dichotomy. Perhaps, the long-time survival of the idea/expression dichotomy as an analytic tool is more due to the general tendency of some ill-fitting legal doctrine to remain vital as a bewildering type of phantasmagoria -- especially when there is a failure to find a useful substitute -- rather than due to a doctrine's conceptual vitality.

Whatever the reason for its survival, the encryption decisions are compelling indications that the time has come for courts to put aside and replace the amorphous and ineffective dichotomy. This is particularly true in the context of copyright infringement actions involving computer source code. Since the proverbial moment the law of copyright first recognized that computer programs could be subject to copyright protection, courts have struggled with setting the boundaries of what aspects of a computer program are copyrightable. Clearly, both the Copyright Act and the First Amendment prohibit the application of copyright protection to ideas, but applying that constitutional and statutory doctrine to actual allegations of copyright infringement is neither simple, nor precise.

Even so, recent changes in computer software development -- largely the result of a paradigm shift in programming initiated by copyleftists and the open source code movement in cyberspace -- and the recent approval of the argument that the nature of software development often involves the free expression of ideas should sufficiently set the groundwork to advance copyright jurisprudence by freeing courts from reliance on inconsistent and incoherent distinctions between copyrightable and uncopyrightable aspects of computer programs. Most important, viewing computer source code as an artifact of the public domain suitably reinforces an important goal of copyright; namely, that the government grant copyrights in works to meaningfully motivate the creativity of authors in a manner that ultimately ensures public access to authors' products. In this regard, copyright law should permit the unfettered access to public domain material by protecting source code authors from copyright infringement when the elements of a work at issue in an infringement action are the artifacts of the public domain. Thus, courts adjudicating copyright infringement actions involving computer software should undertake a thorough reassessment of the limiting principles of copyright law, recalibrate the boundaries and the scope of copyright protection for software, and rarely regard source code as a category of expression created as a result of independent, and hence, original authorship.

II. BACKGROUND

Computer source code[1] *is* speech[2] -- that is the argument advanced by privacy advocates and cryptographers, who recently

---

[1] Source code is a set of symbols governed by lexical rules that computer programmers use to instruct computers to perform certain actions. Joseph Weber, USING JAVA 1.1, THIRD EDITION, at 74 (1997).

[2] This is not a trivial statement; its fulcra turn on the many approaches to determinations of what it means to say that a given form of communication or communicative event has constitutional value. *See, e.g.,* Roth v. United States, 354 U.S. 476, 484, *reh'g. denied*, 355 U.S. 852 (1957) (essentially declaring that obscenity is a form of speech that lacks constitutional value). Although this article takes as its starting point the conclusion that source code is speech, the conflicting legal opinions in the encryption cases amply support that the question of what degree of constitutional value source code properly warrants contains nuances of complexity. Nonetheless, the focus, here, is in the context of copyright jurisprudence, where the application of copyright

obtained the support of the Ninth Circuit in *Bernstein v. United States.*[3] Courts are in general conflict over the appropriate standard of review to apply to the government's regulations regarding the use or export of encryption products. However, among the courts that have considered the question, there is remarkable accord supporting the view that in some significant respects, computer source code is speech and, as such, represents the expression of ideas by those who understand the arcane languages of computer programming.[4] That source code should be viewed as a marketplace of scientific and technological ideas is a remarkable determination. Notwithstanding that declaring that 'source code is speech' is simplistic in its formulation, the impact of the determination is far reaching and profound. In light of this recent formulation, protecting First Amendment values in computer source code

---

protection to expressions, but not to ideas, operates to limit the reach of copyright at the constitutional line of the First Amendment.

[3] A panel of the Ninth Circuit upheld the district court's finding that computer source code is speech protected by the First Amendment. Although the Ninth Circuit granted the Federal government's request for a rehearing *en banc* and subsequently withdrew the panel's opinion, the view that computer source code may represent the expression of ideas by those who can read and understand the language of computer programming remains the law of this case as well as the law of the only other two cases to have considered the question. *Cf.* Junger v. Daley, 8 F.Supp.2d 708 (N.D.OH. 1998) (acknowledging expressive elements in source code) and Karn v. U.S. Dep't of State, 925 F. Supp. 1, 9 (D. D.C. 1996) (assuming that computer source code is protected by the First Amendment) with Bernstein v. U. S. Dep't of State, 922 F. Supp. 1426, 1436 (N.D. Cal. 1996) (holding that source code is speech).

[4] A computer language is a systematized formation of signs and symbols used to construct a computer program that can send instructions to a machine. A computer program is a set of instructions to a computer. See generally, M. Keplinger, *Computer Software--Its Nature and its Protection*, 30 Emory L.J. 483, 484-85 (1984) (source code is "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result"). Source code ultimately controls the software and hardware that taken together function as computers. Computers are digital technologies, and, as such, can be used to efficiently express vastly different forms of information - such as factual databases, audio recordings, or electronic mail messages - using bits of data in the form of computer 0s and 1s. Bits (or binary digits) are essentially the smallest and most fundamental units of digital technology data; each bit has a value of 0 or 1. The bits 0 and 1 represent off and on switches, which measure the presence or absence of electrical voltage in any given memory register of the computer. Since binary digits enable fairly easy digital expression and digital technology significantly expands the amount of data that can be processed on a single silicon chip, computers have become the format of choice in electronics. Rod Dixon, *Profits in Cyberspace: Should Newspaper and Magazine Publishers Pay Freelance Writers for Digital Content? -- Tasini v. New York Times*, 4 Mich. Telecomm. & Tech. L. Rev. 127 (1998).

requires a reconsideration of the proper scope of copyright protection in source code.[5]

In its sum and substance, the law of copyright[6] both advances and encumbers the manner in which an author may express himself.[7] Copyright is a constitutional promise granted by the Federal government that must yield, in instances of conflict, to the First Amendment. To that end, courts have engrafted onto the law of copyright a presumptively dispositive test often referred to as the idea/expression dichotomy.[8]

---

[5] This is not to say that a Writing may not be subject to First Amendment and Copyright protection. Rather, the point is that the limiting doctrines of copyright circumscribe the conditions under which a work or aspects of a work may be protectable speech, but not (or no longer) copyrightable. *See, e.g.*, Bridge Publications, Inc., v. F.A.C.C.T. Net, Inc., 183 F.R.D. 254, 262 (D.Colo. 1998) (once a work is injected into the public domain, it remains there); Dow Jones & Co. v. Bd. Of Trade of City of Chicago, 546 F.Supp. 113, 116 n.5 (S.D.N.Y. 1982) (once a work enters the public domain, all of its copyright protection is lost permanently). More important, to say that source code is speech is to say that source code essentially is a caldron of ideas. In this respect, it is apparent why protecting source code under copyright intrudes upon First Amendment interests.

[6] Congress enacted the first federal copyright statute in 1790. Act of May 31, 1790, §§ 1 and 3, 1 Stat. 124-125. Of course, the source of the law of copyright is Art. I, § 8, cl. 8 of the U.S. Constitution.

[7] Although courts and even many commentators are often apt to repeat the Supreme Court's epigrammatic jingle that "copyright is the engine of free expression," in practical use, copyright is both a shield and sword. It limits and enhances free expression. Copyright is used to temporarily protect an author's expression from use by others – and, thus, disrupt the expressive efforts of others – through the granting to copyright owners of a government-protected monopoly on words, images, sounds, and similar modes of communications.

[8] Apparently, courts were convinced, without the guidance of Congress, that copyright protection extended to a computer program's source and object codes. *See, e.g.*, Stern Electronics, Inc. v. Kaufman, 669 F.2d 852, 855 n. 3 (2d Cir.1982) (noting prior protections of source code). In step with the lead of courts, Congress amended the Copyright Act in 1976 to include, *inter alia*, the First Amendment test limiting the reach of copyright to expressions, not ideas. 17 U.S.C. § 102(b). Section 102(b) provides: "[i]n no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of form in which it is described, explained, illustrated, or embodied in such work". This provision is generally considered Congress' expression of its desire to codify the idea/expression dichotomy. *See also*, Computer Associates Int'l v. Altai, Inc., 982 F.2d 693, 703 (2d Cir. 1992). Despite occasional evidence of the contrary, the idea/expression dichotomy – in name – is only a shorthand reference and should not be understood literally as extending copyright solely to expressions that are either devoid of content or that lack ideas.

In applying the idea/expression dichotomy, courts have often struggled to draw principled distinctions between copyrightable expression[9] and the so-called basic ideas[10] underlying copyrightable expression,[11] particularly when the "expression" subject to a court's analysis is contained in computer programs.[12] As a result of cyberspace-based trends[13] in computer programming[14] -- including recent successes

---

[9] 17 U.S.C. § 102(a)(1) extends copyright protection to "literary works," and computer programs are classified as literary works for the purposes of copyright. *See* H.R.Rep. No. 1476, 94th Cong., 2d Sess. 54, reprinted in 1976 U.S.Code Cong. & Ad.News 5659, 5667. Of course, copyright in a work can be infringed, without reference to the idea/expression dichotomy, when there is substantial similarity between two works' literal elements. One can also violate the copyright of a play or book by copying its plot or plot devices. *See, e.g.*, Twentieth Century-Fox Film Corp. v. MCA, Inc., 715 F.2d 1327, 1329 (9th Cir.1983) (plot similarities between Battlestar Galactica and Star Wars may be basis for a finding of copyright violation).

[10] To some degree, the idea/expression dichotomy is a doctrine that merely creates a definitional landscape by artificially drawing a curtain over "ideas" in order to maintain "expression" as the province of copyright. As such, the question of how to distinguish an idea from an expression often is not satisfactorily answered; instead, the question is recast as having to do with traditional notions or definitions of expressive output.

[11] 17 U.S.C. § 201(a) (1988 & Supp. IV 1993). A "work" is fixed "in a tangible medium of expression when its embodiment in a copy or phonorecord, by or under the authority of the author, is sufficiently permanent or stable to permit it to be perceived, reproduced, or otherwise communicated for a period of more than transitory duration." 17 U.S.C. § 101 (1996). Copyright does not extend to any "idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work." 17 U.S.C. § 102(b) (1996).

[12] The idea/expression dichotomy is by no means troublesome only in the context of software. More than just a few commentators have found the notion that an idea and its expression may be parsed with the precision required by the jurisprudence of copyright to be sophistic, at best, and, perhaps, more appropriately, intellectually objectionable. Perhaps, the better argument is that the meaning or idea conveyed by an expression may be rooted in a combination of linguistic and interpretative phenomena, including analytical pragmatics, semantics, and syntax, as well as other structural, contextual, and logical interpretative considerations. The intricacy in which meaning is tied to language renders the task of separating an idea from its expression a highly dubious endeavor. *See, e.g.*, Neil Smith and Dierdre Wilson, Modern Linguistics: The Results of Chomsky's Revolution, 170-171 (1980). No less an authority on legal doctrine than Judge Learned Hand is said to have concluded that a court's distinction between an idea and its expression will "inevitably be ad hoc." Peter Pan Fabrics, Inc. v. Martin Weiner Corp., 274 F.2d 487, 489 (2d Cir.1960). Thus, the doctrine may have no principled basis.

[13] Commentary on the impact of Cyberspace on speech rights and intellectual property interests is now abundant. *See, e.g.*, Douglas J. Masson, Comment, *Fixation On Fixation: Why Imposing Old Copyright Law On New Technology Will Not Work*, 71 Ind. L.J. 1049, 1052 (1996); Niva Elkin-Koren, *Copyright Law and Social Dialogue on the Information*

by copyleftists[15] and the open source code movement -- and judicial recognition of the communicative dimension of computer source code, the federal courts' struggles with the application of the idea/expression dichotomy could be in decline.[16] Many commentators have criticized[17]

Superhighway: The Case Against Copyright Liability of Bulletin Board Operators, 13 CARDOZO ARTS & ENT. L.J. 345 (1995); Jane C. Ginsburg, *Putting Cars on the "Information Superhighway": Authors, Exploiters, and Copyright in Cyberspace*, 95 COLUM. L. REV. 1466 (1995); I. Trotter Hardy, *The Proper Legal Regime for "Cyberspace,"* 55 U. PITT. L. REV. 993 (1994); Pamela Samuelson, *Legally Speaking: Copyright's Fair Use Doctrine and Digital Data*, Comm. ACM 37: 21 (Jan. 1994); Raymond T. Nimmer & Patricia Ann Krauthaus, *Copyright on the Information Superhighway: Requiem for a Middleweight*, 6 STAN. L. & POL'Y REV. 25 (1994); Information *Infra*structure Task Force, *Intellectual Property And The National Information Infrastructure: A Report Of The Working Group On Intellectual Property Rights*, <http://www.uspto.gov/web/offices/com/doc/ipnii/index.html> (visited Apr. 25, 1995) (in the world of intellectual property, this document is commonly referred to as The Green Paper).

[14] As noted more fully below, many of the recent changes in computer software development support the conclusion that courts need not rely upon inconsistent and incoherent distinctions between copyrightable and uncopyrightable aspects of computer programs in adjudicating copyright infringement actions. In this respect, viewing computer source code as an artifact of the public domain suitably reinforces an important goal of copyright; namely, the motivation of the creative activity of authors – through which public access to the products of an author's creativity may be assured by advancing public access to works and supporting the continued vitality of the public domain.

[15] "Copyleftist" is a short hand reference to the members of the faction of the open source code movement whose participants do not oppose proprietary use of open source code projects as long as the software applications are copylefted. As noted more fully below, copylefting a software application involves distributing source code with a so-called public license that essentially dislodges the exclusive rights granted to a work by copyright. The terms of the public license prematurely pushes the source code into a public commons. In other words, copyright is turned on its head, hence, the term, copyleft.

[16] Although the formulation of the idea/expression dichotomy is largely inadequate for the purpose for which it is directed, the principles upon which the dichotomy is based are not only consistent with recognizing a First Amendment limitation on the scope of copyright, but also consistent with the goals of copyright, itself. In this regard, it is noteworthy that a basic purpose underlying the idea/expression distinction – as it applies to software – is to allow copyright protection beyond the literal computer code, and provide the proper incentive for programmers by protecting their most valuable efforts, while not giving the copyright holder a stranglehold over the development of similar software programs that accomplish the same end. *See, e.g.*, Herbert Rosenthal Jewelry Corp. v. Kalpakian, 446 F.2d 738, 742 (9th Cir. 1971) (The idea of a jeweled bee pin was held to be inseparable from the expression of the idea and thus this 'stranglehold' by copyright was held invalid). Moreover, while the idea/expression dichotomy in its actual

the analytical limitations of the dichotomy, and others have argued that courts frequently misunderstand the dichotomy.[18] As noted previously, perhaps, the survival of the idea/expression dichotomy as an analytic tool is more due to a failure to find a useful substitute than due to its conceptual vitality.[19]

Whatever the reason for the long-time survival of the idea/expression dichotomy,[20] the encryption decisions[21] should be

---

application is of dubious use as a proxy for First Amendment analysis of the scope of copyright protection for computer source code, its value in other contexts is less open to doubt.

[17] *See e.g.*, Alfred C. Yen, *A First Amendment Perspective On The Idea/Expression Dichotomy And Copyright In A Work's "Total Concept And Feel,"* 38 EMORY L.J. 393 (1989) (reliance on the idea/expression dichotomy to reconcile copyright with the First Amendment is unjustified).

[18] Related to the idea/expression dichotomy is the *scenes a faire* doctrine. *Scenes a faire* are incidents, characters or settings, which are as a practical matter indispensable in the treatment of a given topic. Atari, Inc. v. North American Philips Consumer Elecs. Corp., 672 F.2d 607, 616 (7th Cir.), *cert. denied*, 459 U.S. 880 (1982). The *scenes a faire* doctrine, like the idea/expression dichotomy, is a limiting doctrine that essentially permits the artifacts protected by the doctrine to be freely used by an author. *Scenes a faire* are afforded no copyright protection. *Id.* (*citing* Reyher v. Children's Television Workshop, 533 F.2d 87, 91 (2d Cir. 1976), *cert. denied*, 429 U.S. 980 (1976).

[19] Indeed, the notion that a court (or anyone else, for that matter) can separate an idea from its expression seems to beg for judicial invention. It is a fundamental linguistic principle that we grasp ideas through expression; an idea cannot exist apart from expression. Although, in some metaphysical sense, expressions refer to ideas outside themselves, there is an intimate tie between expressions and ideas that the act of untying substantially disturbs. In other words, expressions are coefficients of ideas that are not easily subjected to the anachronistic tools of the idea/expression dichotomy. *See, e.g.*, GEORGES GUSDORF, NORTHWESTERN UNIVERSITY STUDIES IN PHENOMENOLOGY AND EXISTENTIAL PHILOSOPHY, SPEAKING (LA PAROLE), (1965) (for an interesting view on how existential phenomenologists consider the conjunction of expressions and ideas as a constitutive element of human reality that cannot be meaningfully separated in the context of the human experience).

[20] Under the dichotomy, copyrightability is considered in the context of an infringement action. Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1253 (3d Cir. 1983) (by excluding copyright expression in computer code dictated by the external factors of computer hardware from protection against infringement, copyright law secures for public use those elements of computer code incidental of ideas underlying the software).

[21] This is a shorthand reference to the *Bernstein*, *Karn*, and *Junger* decisions. As noted *infra*, these decisions are not in agreement over the degree of First Amendment protection to extend to computer source code, nor do they agree as to the appropriate standard of review to apply to the Federal government's regulation of encryption source code. Each court's determination remarkably exposed that the source code of a computer

viewed as compelling indications that the time has arrived for courts to put aside and replace the amorphous and ineffective dichotomy;[22] this is particularly true in the context of copyright infringement actions involving computer source code.[23]

---

programmer has a communicative dimension. Notably, although the *Junger* and *Karn* courts suggest that some uses of source code may override the communicative dimension of source code -- such as when encryption source code is exported in digital form or as digital media -- those uses are not relevant in the adjudication of copyright infringement. Instead, the relevant inquiry, under copyright, is, initially, whether there are aspects of the plaintiff's source code that must be filtered out of the litigation because they are First Amendment ideas (not expressions) or artifacts of the public domain. In this regard, the encryption cases are instructive; these cases support the view that source code contains a great deal of stock ideas, perhaps far more than had been previously considered by courts adjudicating copyright claims and, as such, entirely or predominately consist of uncopyrightable expression. This follows, not just from the fact that source code largely contains uncopyrightable methods and procedures, but also from the fact that these procedures and methods are arranged by pre-established programming conventions.

[22] Although, conceptually, the application of the dichotomy to source code is suspect for reasons already noted, it is its use by courts that has been most troublesome. The idea/expression dichotomy, along with its distinct subsisting tests – such as the abstraction-filtration-comparison test, is a tool that courts once called upon to assist them to identify protectable copyright expression, but have since become beholden to as a proxy for the constitutional analysis. *See* Mitel, Inc. v. Iqtel, Inc., 896 F. Supp. 1050, 1055 (D.Colo. 1995).

[23] The Federal government sought and obtained a rehearing *en banc* of Bernstein before the Ninth Circuit. In addition, in September 1999, the Clinton administration announced that its policy toward restricting the export of encryption technologies will be "substantially relax[ed]" when the Department of Commerce issues new export regulations. *Curbs on Export of Secrecy Codes Ending*, WASH. POST, September 17, 1999, at A1. It may be that the Federal government is determined to achieve its objectives regarding the regulation of encryption products through the enactment of new legislation (the Department of Justice is supporting the proposed Cyberspace Electronic Security Act of 1999). *Id.* Notwithstanding these recent developments, the government continues to regulate encryption technologies, and it does so in a manner that formed the basis of Bernstein's complaint. More important, the question addressed, here, arises more directly from the assumptions by all of the encryption cases; namely, that source code creation is the result of mathematicians, software engineers, and computer programmers culling together from the public domain various functions, subroutines, algorithms, and statements in accordance with the syntactic rules of a given programming language. In this regard, the proper level of First Amendment protection that is accorded source code is not directly implicated by the focus of this article. Even so, whatever the outcome of the Bernstein litigation, it is doubtful that the Ninth Circuit will have the final say on this matter. Many commentators have opined that given the importance of the First Amendment question in *Bernstein* as well as the fact the Sixth Circuit and the D.C. Circuit will soon weigh in on the same issue, it is quite likely that the issue will come before the U.S. Supreme Court shortly.

Bernstein specifically resolves the question whether encryption source code is speech. *Karn v. United States Department of State* and *Junger v. Daley* avoid resolving this question by declaring the question presented as one requiring the court to determine whether the Federal regulation at issue is purposefully directed at speech.[24] Nonetheless, if the implicit assumptions of the encryption decisions remain the prevailing view or are upheld by the Supreme Court, their reach will necessarily extend beyond the debate over the use of encryption software. The proponents of unregulated use of encryption products have launched a moderately successful campaign toward protecting source code as an artifact of the First Amendment. As courts find it persuasive that some computer source code should be viewed as an expression of ideas, the protection of source code under the First Amendment may ultimately reshape the scope of its future protection under copyright.[25]

Viewed in this light*,* the idea/expression dichotomy may have no doctrinal application in the context of computer software copyright infringement litigation. This article concludes that although under copyright doctrine the boundary between an idea and an expression remains fuzzy and subject to semantic invention, viewing computer source code as speech[26] resolves some of the limitations in applying the idea/expression dichotomy by clearly fixating most of computer source

---

[24] As pointed out *infra*, whether the *Junger* and *Karn* courts relied upon a sleight of hand to side-step the admittedly complex constitutional question concerning the First Amendment status of computer source code is subject to dispute; undoubtedly, focusing a court's attention on whether the government may regulate the export of encryption source code rather than upon whether the regulation of encryption source code through a federal licensing scheme violates the First Amendment could lead to distinct results in a court's analysis of what is at issue. Though finding, as in *Bernstein*, that encryption source code is speech, the court, in *Karn*, rejected Karn's claims and upheld the constitutionality of the AECA and ITAR on the grounds that they furthered an important or substantial governmental interest. In addition, the court rejected his argument that the ITAR constituted a prior restraint on free speech since the regulations were content-neutral. At odds with *Bernstein*, *Junger* granted summary judgment in favor of the government, holding that the First Amendment did not protect the export of encryption source code on the Internet.

[25] Interestingly enough, the question of whether the source code is expressive or merely functional is reminiscent of the debate concerning whether source code is copyrightable. *See generally*, RAYMOND T. NIMMER, THE LAW OF COMPUTER TECHNOLOGY 1.0 (rev. ed. 1992).

[26] All of the encryption decisions, of course embrace this conclusion, but do so with differing levels of enthusiasm.

code in the marketplace of ideas or, in other words, outside the scope of copyright protection.[27]

At bottom, a given slice of computer source code cannot be protectable as both copyrightable expression and as expression belonging to the marketplace of ideas.[28] Source code should be viewed as a resource for other software authors to draw upon when writing source code for their own programs. Subsisting within the purposes of copyright is the purpose to allow the public unfettered access to the uncopyrightable aspects of a work.[29] The copyright law supports the progress of science and the useful arts by, among other things, withholding the grant of copyright, in the context of infringement actions,[30] from aspects of works that constitute ideas, merge ideas with expressions, that constitute scenes-a-faire, or singular modes of expression.[31] In this regard, it would be consistent with the objectives of copyright that source code be freely copied, distributed or used in the creation of derivative works.[32] Stated simply, the law of copyright would

---

[27] This is consistent with the Supreme Court's analysis in *Feist Publications Inc., v. Rural Tel. Serv. Co.*, where the Court required "some minimal degree of creativity," or a "minimal creative spark" before finding copyrightability in a compilation of a telephone book's white pages. 499 U.S. at 362, 363 (1991).

[28] Notably, excluding copyright protection in source code is not tantamount to eliminating software programs from the scope of copyright protection. Computer software, like a book or a screenplay, may contain both copyrightable and uncopyrightable aspects. A computer program's screen output may be copyrightable, although the source code would not be. *See, e.g.*, Gates Rubber Co., v. Bando Chem. Indus., Ltd., 9 F.3d 823 (10th Cir. 1993).

[29] This article does not address object code – except to say, as developed more fully below, for some computer languages, the presence of object code is a trivial or inconsequential matter; admittedly, object code may present its own set of uniquely difficult questions for copyright. 17 U.S.C. § 102(b).

[30] Section 102(b) of the Copyright Act also specifies limitations of the scope of copyright, however, due to the complexity of contemporary infringement actions, courts often must apply judicially crafted limiting doctrine, rather than rely upon the plain language of the statute.

[31] A singular mode of expression means there is one precise way to say something. Not surprisingly, as identified more fully, below, the need to be precise in the use of computer languages limits creative expression in source code; there may be only a singular precise way to code a given scientific expression.

[32] Since courts have long held that software programs contain both literal and nonliteral elements subject to copyright protection, it is highly doubtful that removing source code from the copyrightable aspect of a computer program would have a perceptible adverse impact on Congress' ability to promote the progress of computer science, should such congressional action be considered necessary.

provide sufficient incentive[33] for software developers to create works from a vast public domain; the public domain would provide access to source code for future authors and, in turn, those authors would create works[34] that could promote the progress of science, thereby, further enriching the public domain.[35]

III. COPYRIGHT AND FREE EXPRESSION

At the root of nearly every transaction in cyberspace is the transmission of an idea, and in cyberspace, one cannot not communicate.[36] In the borderless virtual space of cyberspace, the shift

---

[33] Some have argued that the incentives the law of copyright provides are solely those of the author, not the public. *See generally* Mitel, Inc. v. Iqtel, Inc., 896 F. Supp. 1050 (D. Colo. 1995). Nonetheless, there are more than sufficient instances demonstrating that in the context of technology, and perhaps beyond, authors would create works without the protection of copyright. Indeed, scientific works, although subject to patent protection, may be outside the scope of copyright entirely. Funk Bros. Seed Co. v. Kalo Inoculant Co., 333 U.S. 127, 130 (1948) (copyright law recognizes no claims for scientific inventions). Perhaps the best proof lies within the Copyright Act, itself, wherein it excludes expression such as business forms and type fonts from copyrightability, yet, authors continue to create such works for compensation.

[34] At first blush, some may find this conception of copyright to embrace a perverse notion of incentive. Why would an author, one might say, create a work without compensation for each copy? The short answer is that authors create such works frequently; notably, employees, under the work-for-hire doctrine, do not retain copyright interests in the works they create. In addition, the open source code movement challenges prior assumptions as to what establishes sufficient incentive for authors to create works. More directly, software programs contain other aspects that may be suitable to copyright protection, including output, screen interface, program design, and graphical images.

[35] *See e.g.,* Stephen Breyer, *The Uneasy Case for Copyright: A Study of Copyright in Books, Photocopies, and Computer Programs*, 84 HARV. L. REV. 281 (1970).

[36] It is not surprising that many disputes in Cyberspace involve claims arising from or in defense of freedom of speech. This article examines one aspect of the freedom of expression debate arising, in part, in Cyberspace; namely, whether the extension of First Amendment protection for source code will leave any challenge to the notion that future copyright protection of computer source code is largely in doubt. Perhaps, running contra to this position is a frequently mentioned and increasingly popular rationale for copyright: the property right rationale. It is based on the following assumptions: (1) granting property rights in a work will allow the author to earn a profit, (2) the ability to earn a profit will provide the author with an incentive to create, and (3) the more works that are created, the greater the benefit to the public. *See, e.g.*, Julie Cohen*, A Right to Read Anonymously: A Closer Look at "Copyright Management" in Cyberspace*, 28 CONN. L. REV. 981 (1996) (noting some of the implications of these assumptions). As

from mere idea to the *communication* of an idea occurs automatically almost as a transparent instinctive response.[37] Yet, the conceptual distinction between ideas and the communication or expression of ideas is fundamental in copyright doctrine.[38]

The United States Supreme Court has determined that First Amendment freedoms of speech include the collective interest in protecting an individual's right to freely express almost any idea known to man.[39] Copyright law directly affects the free expression of ideas[40] because the United States Constitution secures for "limited times" to copyright holders "the exclusive Right to their respective Writings and Discoveries."[41] The copyright statute gives copyright owners a variety of exclusive rights: the rights to make copies of their works, to create derivative works, to distribute the works, and to publicly perform or display them. In other words, the law of copyright grants to authors the

---

noted in Part II.A., *infra*, the open source code movement represents a frontal attack upon the continuing vitality of those assumptions.

[37]See 1 Paul Goldstein, COPYRIGHT § 4.4.1.2 (1989); 3 Melville B. Nimmer & David Nimmer, NIMMER ON COPYRIGHT 10.02, at 10-22 to 10-35 (1997).

[38] Principled and conceptually based distinctions in the law of copyright are not without their apparent contradictions and compromises. Any complete commentary on contemporary trends in the law of copyright must recognize that some doctrinal difficulty is due to political compromise. It is unremarkable to acknowledge that as a result of the growing demand for digital content, copyright owners are in a race to try and get Congress to pass laws that benefit one group over another. *See, e.g.*, David Landis, *Catching Some Entertainment, in Bits and Pieces*, USA TODAY, Aug. 25, 1994, at 8D (quoting Jay Berman, President of the Recording Industry Association of America).

[39] This fundamental right, like all rights, is not without its exception or counter-balancing interests. Nonetheless, the free expression of ideas is a First Amendment precept.

[40] Often, when copyright infringement is alleged, courts must balance the constitutionally competing aims of promoting human creativity and original expression through the strict enforcement of the copyright law with ensuring that broad copyright protections do not unfairly or unnecessarily prevent the development of our knowledge base -- particularly, the nation's development of practical uses of information. *See, e.g.*, Rod Dixon, *Profits in Cyberspace: Should Newspaper and Magazine Publishers Pay Freelance Writers for Digital Content? -- Tasini v. New York Times*, 4 MICH. TELECOMM. & TECH. L. REV. 127, 140 (1998).

[41] U.S. CONST. art I. § 8, cl. 8.

right to control, restrict, or thwart public access to their expressive product. [42]

Despite the compelling language of the Constitution's copyright provision, it is apparent that the founding fathers only intended to permit Congress to protect a copyright holder's right to her original *expression*.[43] In the clash of competing constitutional provisions and almost strictly as a conceptual matter, the First Amendment trumps Article I, Section 8, Clause 8, in its significant limitation upon the scope and function of the law of copyright.[44]

Recently, the encryption debates moved from public deliberations in popular media to significant controversies presented in various legal fora. As proponents of unregulated use of encryption

---

[42] The paradox is that the public can only benefit if it has access to a work. Access is restricted, at least for a limited time, by granting the author property rights in her work, for only by restricting access can the author charge users and earn a profit. *See generally* Jessica Litman, *The Public Domain*, 39 EMORY L.J. 965 (1990). The artifacts of Cyberspace are largely intellectual property, and the owners of the intellectual property have a right to control how their property is communicated. In this respect, despite the open and public nature of Cyberspace, it is the province of an inherent and basic tension flowing from the goal of access to information between those that communicate and those that own the artifacts of communication. Of course, reliance on copyright law is not the only manner an author may reliably restrict access to his work. She may, for instance, license use of her computer software. Or, she may rely upon technological barriers -- often referred to as digital copyright management systems -- to prevent unfettered access to her works. Notably, some authors use a variety of factors to restrict public access to a given work, but as noted in Part II.C., *infra*, the open source code movement is forestalling this previous trend.

[43] To be an author under the law of copyright, an artist or software developer must supply more than mere direction or ideas; he must 'translate an idea into a fixed tangible expression entitled to copyright protection.' Community for Creative Non-Violence v. Reid, 490 U.S. 730 (1989).

[44] "Original," is a term of art in copyright; it means only 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. 1 M. Nimmer & D. Nimmer, Copyright § § 2.01[A], [B] (1990). Originality does not signify novelty; a work may be original even though it closely resembles other works so long as the similarity is fortuitous, not the result of copying. *See* Sheldon v. Metro-Goldwyn Pictures Corp., 81 F. 2d 49, 54 (2d Cir. 1936). Originality is a constitutional requirement. The source of Congress' power to enact copyright laws is Article I, § 8, cl. 8, of the Constitution, which authorizes Congress to "secure for limited Times to Authors ... the exclusive Right to their respective Writings." In this regard, it is thought that the Constitution mandates some – albeit minimal – degree of creativity; an author who claims copyright infringement, therefore, must prove the existence of independent intellectual conception.

technologies have presented increasingly sophisticated arguments supporting their positions, the federal government also has been unrelenting in its opposition to an entirely unregulated regime of encryption use.[45]   At bottom, the encryption debates concern disagreements between the federal government and various privacy advocates -- who have been joined by a sizeable number of software companies -- over whether, and, if so, to what extent, the government should be permitted to continue its regulation of the manner in which access to data in any form may be blocked for all of those except authorized and intended recipients through the use of digital technology.[46]

Arguing that encryption technologies may present the most serious challenges to national security or local and national law enforcement when they are in digital form, the federal government has not been halting or languid in its efforts to regulate the actual *use* of encryption devices and the fruits of encryption products.[47] In this regard, the encryption debates have moved toward a wide-ranging area of disagreements, including the recent dispute concerning whether the very discussion of encryption technology may, in part, be significantly restricted or, at least, regulated by the Federal government.

The arguments on both sides have drawn toward framing one central question in the debates: whether encryption computer source code is speech and, therefore, protectable under the First Amendment. If the ultimate outcome of this dispute leads to protecting encryption source

---

[45] Once almost the exclusive province of military and governmental bodies, cryptography is now increasingly available to businesses and private individuals wishing to keep their communications confidential. *See* Bernstein v. United States Dep't of State, 974 F. Supp. 1288, 1292 (N.D. Cal. 1997).

[46] To keep their communications confidential, users encrypt and decrypt communications, records and other data. Through encryption, users seek to prevent the unauthorized interception, viewing, tampering, and forging of such data. Without encryption, those other than the intended recipient may view a sender's private or personal messages. Through encryption human-readable text of a message or document (also known as "plaintext") is transformed into a text (known as "ciphertext") that the sender and recipient intend third parties not to understand. "Decryption," simply stated, is the reverse process of transforming the ciphertext message or document into the original plaintext.

[47] Encryption software carries out a cryptographic algorithm, along with other computations, that directs computer hardware to encrypt plaintext into an encoded ciphertext. Mathematical functions or equations usually make up the source code.

code as speech, then the federal government's attempt to regulate the digital technology of encryption and the transmission or communication of encryption "speech" will be significantly impaired.

Quite apart from the impact of the current dispute on the future use of encryption, the ultimate resolution of the question will have an astonishing level of influence on the current jurisprudence governing the balance between copyright and the First Amendment in the context of computer programs. Since both the First Amendment and the law of copyright protect expressive conduct, a ruling upholding computer source code, as an essentially expressive artifact would significantly restrict the degree to which many software copyright owners could successfully challenge another developer's use of source code on the basis of copyright infringement. In other words, viewing computer source code as speech protectable by the First Amendment underwrites a new balance between copyright and the First Amendment.[48] In this respect, First Amendment analysis of encryption source code sharply illuminates the predominant failure of copyright doctrine -- the idea/expression dichotomy -- to adequately calibrate the boundaries of copyright and free speech in the context of digital technologies.

In the context of computer source code, a re-examination of the relationship between copyright, which places limits on the accessibility of certain speech, and the First Amendment, which, generally, is directed toward making speech as open and accessible as possible, may reveal a balanced, but more relevant, division between these contextually competing interests than the current murky and obscure distinctions of copyright doctrine.[49]

---

[48] Despite many significant efforts by Congress and the courts, copyright doctrine has never been suitably revised to faithfully accommodate the competing interests and the complex questions raised by allowing authors to claim copyright in digital technologies. *See, e.g.*, Arthur R. Miller, *Copyright Protection for Computer Programs, Databases, and Computer-Generated Works: Is Anything New Since CONTU?*, 106 HARV. L. REV. 977 (1993).

[49] *See, e.g.*, Computer Assocs. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 701 (2d Cir. 1992) (holding that in a suit for copyright infringement, the plaintiff must establish its ownership of a valid copyright, and prove that the defendant copied the copyrighted work). In this case, the court had to decide what non-literal elements of computer programs were eligible for copyright protection and whether the scenes a faire doctrine protected the elements copied by the defendant.

Although the purpose of writing source code is not to draft letters to lovers or communicate contractual terms that may bind two parties, source code can be read or understood by computer programmers, computer hobbyists, mathematicians, scientists, and other professionals who are trained in the particular programming language in which the source code is written.[50] Notably, it is not necessary for an individual to have the capacity to read or understand any given source code in order to compile or execute software that may control the operation of a computer. Even so, some computer users or programmers may find it useful, if not necessary, to obtain access to the underlying computer source code of a given software application to make full use of the program or extend the program's utility beyond the program's original design.[51]

Part I of this article sets the technological backdrop for establishing why this is the proper time to recalibrate the balance between copyright and free expression for computer source code. Specifically, Part I describes and examines the goals of copyleftists and the open source code movement and sets forth the pertinent issues from the encryption debates that have led privacy advocates to claim source code as subject to the protection of the First Amendment. These major technology-driven predicaments undergird the current arguments and

---

[50] A notable exception to many general statements on source code would apply to source code written in the various versions of Assembly computer language. Assembly is actually a rather powerful, but cryptic, computer language that may be used to send terse instructions directly to the computer's hardware. An unskillful Assembly language programmer could do a great deal of damage to a computer because Assembly source code can execute instructions directly affecting the hardware of a computer, which are not easily available to programmers using high-level languages. Most important, unlike source code written in high-level languages, Assembly language source code closely approximates machine language and is low-level. A program written in an Assembly language may be written so tightly that it may run on only the machine the program was developed on. In this regard, the expressive qualities of any source code necessarily should be considered dubious, when the computer language used to write the source code closely approximates machine language. *See* PETER NORTON, PETER NORTON'S INSIDE THE PC (6th ed. 1995) (machine language is binary).

[51] Indeed, adding to the functionality of computer software by end users has become so common that software developers encourage this practice through the use of macros or programming interfaces like Microsoft's Visual Basic for Applications (VBA), which permits computer users to freely extend the functionality of Microsoft's desktop applications by altering the program's source code without the risk of infringing Microsoft's copyright interests. *See generally* Scott, Shannon, Font, Hatfield, et al., VISUAL BASIC 4 UNLEASHED, at 20–21 (SAMS Publishing, 1995).

court disputes over whether the elements of expressive content in computer source code warrant protection under the First Amendment.

Part II provides an overview of the prevailing copyright law doctrine limiting the expansion of copyright into First Amendment areas; namely, the idea/expression dichotomy, and concludes that the doctrine, in general, and its application to digital artifacts and computer technology, specifically, obfuscates rather than elucidates the critical First Amendment question.

Part III assesses the notion that computer source code should be subject to copyright protection in light of the encryption cases' recognition that in some respects, source code represents the expression of ideas by those who understand the arcane languages of computer programming. Part III concludes that if source code remains within the province of full First Amendment protection, then, consistent with the First Amendment limitation on copyright and the open source code movement, source code should be regarded rarely, if ever, as a category of expression created as a result of independent, and hence, original authorship.

IV. PART I A DEBATE AND A MOVEMENT

A. *The Encryption Debates*

Encryption is the process of converting a message from its original form (commonly known as "plaintext") into a scrambled form (known as "ciphertext").[52] Generally, the strength of any encryption depends on how rapidly it can be decrypted.[53] Most encryption methods

---

[52] *See* Stimson Garfinkle & Gene Spafford, WEB SECURITY AND COMMERCE 187-208 (1997).

[53] There are two basic and widely available types of encryption systems: private key – or   secret key or symmetric key – and public key (or asymmetric key) encryption. In a private key encryption system, the key used to encode the information that is sent to the recipient, is the same key used to decode the encrypted message. The principal drawback to a private key encryption is the risk incurred in sending the key to the intended recipient. Key management is essential; the sender and recipient must use another, secure channel, or protocol, to agree on and exchange a common key. *Id.* In a public key encryption system, there are two mathematically related keys: a private key and a public key. Using a private key, one can encode a message that can only be decrypted with that person's public key. Alternatively, the person can use the recipient's public key to

rely on having a reusable "key," which generates a series of substitutions or transpositions of the plaintext to create ciphertext.[54] Ciphertext can be sent to the recipient, who has an identical or complementary key that can be used to reverse the process ("decrypt") and produce the original plaintext.[55]

An individual who intercepts the ciphertext cannot understand it without breaking the code. One way to do this, assuming one knows the formula used to encrypt, is to try every possible key combination until one obtains a readable text (a process called "brute-force" search).[56] For a sufficiently large key, a brute-force search on even a series of the most powerful supercomputers working in tandem could take longer than the life of the universe.[57]

It is in this regard that the convergence of computers and cryptography is ostensibly threatening one of the most fundamental tools of the national security and law enforcement communities: the ability to eavesdrop on private or secret communication.[58]

Today almost any computer user through the use of encryption software can encrypt electronic messages and other forms of digital data. In order for a personal computer to be used to encrypt data, it must use

---

encrypt a message, which can only be decoded with the recipient's private key and no other. The risk to key security is thereby reduced.

[54] Simon Singh, The Code Book: The Evolution of Secrecy from Mary, Queen of Scots to Quantum Cryptography, 10-11 (1999).

[55] Wayne Madsen, et al., *Cryptography and Liberty: An International Survey of Encryption Policy*, 16 J. Marshall J. Computer & Info. L. 475 (1998).

[56] Bruce Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C 151 (1996) (Brute-force attacks usually require knowledge or access to comparative samples of ciphertext and plaintext).

[57] *See generally* Kenneth W. Dam, T*he Cryptography Wars*, Wash. Post, July 23, 1996, at A17.

[58] Horst Feistel, *Cryptography and Computer Privacy*, Sci. Am., May 1973, at 15. Computer cryptography is treated as a dual-use technology. Dual-use technologies are those that have both military and civilian use. Many encryption technologies were once defined as "munitions" and their export prohibited, but in 1996, encryption technologies were transferred from the Munitions List of the AECA to the Commerce Control List under the EAA. *See* Exec. Order No. 13,026, 61 Fed. Reg. 58,767, 68,572 (1996). Presently, the Commerce Department regulates all encryption technologies, except those developed exclusively for the use of the military.

encryption software that makes the computer's circuitry carry out the encoding process.[59]

Under federal munitions regulations and in the lingo of technologists, with few exceptions, individuals and American companies can export encryption products no stronger than 56-bits.[60] Since encryption products are often measured in terms of key strength,[61] the length of the key -- which is actually produced by a mathematical formula -- will predict the ease or difficulty which an individual trying to crack the encryption by simply trying different key combinations will encounter before he is able to read the stolen encrypted e-mail message or discover the details of the hijacked medical records.[62] In other words, a 56-bit key is longer and, hence, stronger than a 40-bit key primarily because it may take longer to crack.[63]

---

[59] Singh, *supra* note 54, at 272–79.

[60] Although recently subject to change, the current maximum penalties for violating export licensing requirements for encryption source code range from a one-million dollar criminal penalty and ten years in prison to a five-hundred thousand dollar civil penalty and a three year export ban. *See* Steve Higgins, *Breaking U.S. Encryption Statute Could Be Costly*, PC WK., Feb. 8, 1993, at 1; *see also* 22 U.S.C. 2778 (c)(1988); 50 U.S.C. 2410 (c)(1988).

[61] A key system may be public or private. Public key algorithms are designed to enhance the difficulty of deducing the secret key from the public key and in deducing the plaintext from the ciphertext. Although not a likely concern for most individuals, one relative weakness in public key cryptosystems is the potential that a private key may be derived by a cryptanalyst, who has studied your ciphertext and has successfully implemented a ciphertext attack on your messages. In such a scenario, your messages will not only be able to be read at will, but the attacker can use the generated private key to take on your identity and issue messages or engage in other nefarious activities on your behalf.

[62] *See* Rod Dixon, *The Feds Should Act to Ensure Our Privacy Online*, COMPUTER WORLD, April 5, 1999. One troublesome aspect of the government's encryption policy is that it may be easily vitiated because of the borderless nature of cyberspace. A foreign company may purchase American- produced 56-bit encryption technology off the Internet and then upgrade it in their own country to 128-bit technology.

[63] BRUCE SCHNEIER, APPLIED CRYPTOGRAPHY: PROTOCOLS, ALGORITHMS, AND SOURCE CODE IN C, (1996). Key lengths in modern encryption algorithms range in size from 40 to 128 bits. The approximate difficulty of breaking an encrypted message by "guessing" the right key is proportional to the number of possible key values. If the key is 8-bits long, there are 256 possible keys. Therefore, it will take 256 attempts to find the correct key, with an expected number of attempts of 128. If the key is 40-bits long, the total number of keys is very large. Schneier estimates that a network of 400 computers with fast commercially available chips, each capable of performing 32,000 encryptions per second, can complete a "brute force" attack against a 40-bit key in a single day. By

Perhaps enthralled by the Dionysian-like beauty of sophisticated technology, a few privacy advocates[64] modified the battle for stronger privacy laws to include political arguments challenging the Federal government's encryption policy.[65] These privacy advocates were viewed by some as pushing an ill-fated agenda[66] by blurring the distinction between the conception of privacy[67] as an aspect of data protection[68] and

---

comparison a 56-bit key provides 65,636 times as many possible key values as a 40-bit key. *See* Schneier, at 129-138.

[64] Organizations in support of electronic communications privacy issues include, but are not limited to, the American Civil Liberties Union, the Electronic Frontier Foundation, the Privacy Coalition, the Electronic Privacy Information Center, the Privacy Clearing House, Human Rights Watch, as well as numerous encryption scientists. *See also* Anne Meredith Fulton, *Cyberspace and the Internet: Who Will be the Privacy Police?*, 3 COMM. L. CONSPECTUS 63 (1995).

[65] As noted earlier, although the federal government's encryption policy is presumably based on the government's interest in keeping the most powerful data-scrambling, or encryption, software out of the hands of foreign criminals or terrorists by setting low limits on the strength of encryption software that can be licensed for export by software companies subject to U.S. laws, the policy seems rather incoherent since more powerful encryption technology is readily available from foreign software companies. *See* Jeri Clausing, *Online Groups Mount an Effort to Fight Clinton on Encryption*, NY TIMES, January 14, 1999, (discussing the battle over the effects of encryption technology) *but cf.*, Dan Lehrer, *Clipper Chips and Cypherpunks*, 259 NATION 376, 376 (1994) (describing the use of a popular encryption program by an alleged child pornographer to encrypt potential evidence of traffic in child pornography).

[66] Many privacy advocates have attacked the administration's encryption policy as harmful to U.S. software companies and in contravention of basic principles of personal privacy. While these points do not seem to be misdirected, generally, privacy advocates have devoted a deal energy and resources to thwarting the administration's encryption policy. *See* Jeri Clausing, *F.B.I., Security Chiefs Ask Senate for all Keys to all Encrypted Data* (visited July 10, 1997) <http:// www.nytimes.com/library/cyber/week/071097encrypt.ht> (stating that the director of the FBI, Louis B. Freeh, argued before the Senate in support of a Clinton Administration plan for regulation of encryption by a "key recovery" system or "trap door" mechanism to alleviate virtually uncrackable codes for law enforcement in its effort to protect the nation from "terrorism and organized crime in the next century"). *Cf.*, Jeri Clausing, *U.S. Official Says Clinton Wants Market-Driven Encryption Policy* (visited Oct. 9, 1997) <http:// www.nytimes.com/books/sea...yberlib+17609+8+ wAAA+market-driv> (stating that the Administration's key escrow plan is calling for a key recovery mechanism to be built into government used software).

[67] In a poll of one thousand Americans, two-thirds found it more important to protect the privacy of phone calls than to preserve the ability of police to conduct wiretaps. When informed about the Clipper Chip, 80 percent said they opposed it. *See* Philip Elmer-Dewitt, *Who Should Keep the Keys?*, TIME, Mar. 14, 1994, at 90; John Mintz & John Schwartz, *Chipping Away at Privacy?*, WASH. POST, May 30, 1993, at H1 (describing the Administration's contingency plan to ban unescrowed encryption).

the conception of privacy as a fundamental principle underlying the value for maintaining personal control over when, how and whether intimate facts or personal secrets should be publicly revealed.[69] At the heart of the encryption debate and the federal government's dramatic concerns over new technologies is the tension between protecting national security and other economic interests while also safeguarding basic rights in personal privacy in the current Information Age.[70]

In September 1999, the government revised its federal policy on encryption for the second time in two years. The modified policy allows American companies to use encryption programs of virtually unlimited strength when exporting to foreign banks, insurance companies, and certain health and medical service companies, if the foreign company is located in a country on an approved list. In addition, other requirements regarding licensing foreign-bound encryption products and key recovery plans have been scaled back.[71] There is still an export limit on the 56-key strength of encryption products for uses not directly involving financial services or health and medical services.[72] As such, the

---

[68] See John Markoff, *Electronics Plan Aims to Balance Government Access with Privacy*, N.Y. TIMES, Apr. 16, 1993, at A1, A18.

[69] This point often is lost in the vituperative debates on encryption. *See* Rod Dixon, *The Feds Should Act to Ensure Our Privacy Online*, COMPUTER WORLD, April 5, 1999; Jed Rubenfeld, *The Right of Privacy*, 102 HARV. L. REV. 737, (1989) (urging that privacy is a fundamental right limiting the power of government).

[70] Law enforcement agencies argue that unregulated encryption hinders their ability to prevent crime. *See* Jill M. Ryan, *Freedom to Speak Unintelligibly: The First Amendment Implications of Government Controlled Encryption*, 4 WM. & MARY BILL RTS. J. 1165 (1996). But, note that these concerns are not new and, instead, represent a long-held position by the Federal government, which significantly predates contemporary Cyberspace-based security issues. *See, e.g.*, *Cryptographic Algorithms for Protection of Computer Data During Transmission and Dormant Storage*, 38 Fed. Reg. 12,763 (1973) ("The increasing volume, value and confidentiality of these records regularly transmitted and stored by commercial and government agencies has led to heightened recognition and concern over their exposure to unauthorized access and use").

[71] On Jan. 14, 2000 the *Federal Register* published EAR amendments that implement the White House's Sep. 16, 1999 announcement on encryption; although any encryption commodity or software, including components, of any key length can now be exported under a license exception, a technical review is still required for any non-government end-user in any country except for the seven state supporters of terrorism. *See* 65 Fed. Reg. 2492 (Jan. 14, 2000). The new rules simplify U.S. encryption export rules.

[72] *See* Evan R. Berlack & Cecil Hunt, *Overview of U.S. Export Controls*, *in* Coping with U.S. Export Controls 1994, at 11, 26 (PLI Com. Law & Practice Course Handbook Series No. A-705, 1994) (arguing that the technically complex applications, and the many layers of the federal bureaucracy that controls national encryption export policy, properly

government's encryption policy remains substantially unchanged.[73] The critical point, however, is that the encryption debate commenced as a disagreement over how to counter-balance the interests of privacy and security. Now, the debate encompasses First Amendment freedom of speech concerns and, ultimately, will influence copyright jurisprudence.

Quite apart from the impact of the current dispute on the future use of encryption, the ultimate resolution of the question concerning whether source code is speech will have an astonishing level of influence on the current jurisprudence governing the balance between copyright and the First Amendment in the context of computer programs. Similarly, the open source code movement represents an affirmative indication that copyright jurisprudence must be modified.

B. *The Open Source Code Movement*[74]

The phrase open[75] source[76] represents a paradigm shift[77] in computer programming.[78]   Generally, the words refer to the "source

characterizes the licensing scheme currently relied upon to manage the export of encryption products). The export of articles or services on the U.S. Munitions List is regulated by the DTC under the ITAR. *See* 22 C.F.R. § 120.5 (1994). The DTC settles disputes regarding whether an item is on the U.S. Munitions List according to the commodity jurisdiction procedure, which determines whether the ITAR or the EAR will apply. *See* 22 C.F.R. § 120.4 (1994).

[73] More importantly, the Clinton Administration still strongly favors an escrow public key-based encryption standard. Escrowed Encryption Standard (EES), which theoretically will make it easier for the NSA to monitor Internet communications as well as access computer stored information, involves three keys: the session key, the chip key, and the family key. A. Michael Froomkin, *The Metaphor is the Key: Cryptography, the Clipper Chip, and the Constitution*, 143 U. Pa. L. Rev. 709 (1995).   If an individual communicates by the Internet with another individual with corresponding equipment, both individuals select a session key. The session key is then transmitted by sending a data stream known as a Law Enforcement Access Field (LEAF). *See Approval of Federal Information Processing Standards Publication 185, Escrowed Encryption Standard* (EES), 59 Fed. Reg. 5997 (Nat'l Inst. of Standards & Tech. 1994); *See also* Dorothy E. Denning & Miles Smid, *Key Escrowing Today*, IEEE Comm., Sept. 1994, at 58.

[74] As noted below, the movement includes a range of viewpoints and alternative labels. The label "open source" accurately captures the salient conceptual basis of the movement without risking unnecessary confusion presented by the use of other terms.

[75] Some adherents to the open source movement discourage the use of "open" as the appropriate label to describe the goals and purposes of the movement. For some, the significance of using the term "open" rather than "free" or perhaps "free(dom)" highlights two diverging views of what open source is really about.   There is no dispute that open source challenges the proprietary framework of software development and

asserts that the current intellectual property regime is misapplied with regard to software; those conjoining goals notwithstanding, some open source supporters prefer to emphasize their objective of developing software that grants users and other developers freedom to use the works as they wish. These views are most commonly associated with the Free Software Foundation, which is managed by Richard Stallman. To some extent, the free software movement, unlike the open source movement, is attempting to do more than shift the economic model of proprietary software development toward a more open framework. According to Stallman, the concept of "copyleft" grew out of the Free Software Movement. *See* E-mail correspondence between Rod Dixon and Richard Stallman, (Feb. 4-8, 2000) (on file with author). The distinction is important because "Free Software is a political stand; Open Source is a development methodology. That gives a clear idea of the difference." *Id.* Even so, the movement's factionalism represents differences of degree, rather than kind. Semantics aside, although Stallman's group rightly emphasizes that the new software paradigm is about a great deal more than deconstructing inaccurate and archaic views on software development, no one in the movement would suggest that software should be free, not sold. In this respect, it does seem more useful to label the movement as an open source movement rather than a "free(dom)" software movement.

[76] Open source software has essentially three important features distinguishing it from other forms of software distribution like shareware, freeware, shrink-wrap or general-off-the-shelf consumer software. Open source software is distributed with the source code open to the public for free use and carries a so-called public license precluding a potential software developer from capturing the source code by "closing" the source code. The general public license (known as a "GPL") allows users to sell, copy, and change "copylefted" software programs - which can also be copyright protected - but the author must pass along the same freedom to sell or copy her modifications and change them further. The author must also make the source code of her modifications freely available. In other words, open source software removes the usual restrictions on what a user may do with the program imposed by copyright by (1) requiring that the products developed as open source code software be distributed with a GPL, (2) requiring that derivative or any product developed by modifiying the original software product be distributed with access to the source code, hence, open source, and (3) requiring that the derived program be distributed with a provision in the GPL offering some degree of copyleft protection. To date, the GPL is not known to have been subject to legal challenge.

[77] In some ways, this paradigm shift could be predicated on the last shift; namely, the adoption of object-oriented programming (OOP) re-oriented programming away from procedures and toward objects. OOP saved programmers time by increasing a programmer's ability to create multiple uses of pre-written code. *See, e.g.*, Joseph Weber, USING JAVA 1.1 at 74 (3d ed. 1997). Regardless of a software author's programming philosophy, the interplay among economic, cultural, and technological forces of Cyberspace is reshaping the course of how one does computer programming.

[78] Some examples of successfully launched open source software include well-regarded applications used in Cyberspace such as: Sendmail, the program that routes over 80% of all email on the Internet; Perl, the programming language that is used to write most of the common gateway interfaces (also called "cgi") or applications that enable most electronic commerce features on many web sites; Apache, the most popular web server software run on web servers connected to the Internet; BIND (or "Berkeley

code," or programming, of various pieces of software, wherein the end user is guaranteed[79] free[80] and open access to the software code.[81] Many off-the-shelf software developers try to keep their source code secret, mistakenly assuming that copyright and secrecy were coordinate.[82] But a

---

Internet Name Daemon"), the de facto software used to run the entire DNS (the "Domain Name System") server on the Internet; and perhaps the most popular, Mozilla, the open code software used in the well known and widely used Netscape browser. *See generally* Eric S. Raymond, The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary at 21-24 (O'Reilly ed. 1999). Although all of the aforementioned programs are examples of open source projects, there is considerable debate as to which programs are in substantial compliance with the terms of the GNU GPL, which could be described as the constitution of open source. The greater a program's public license departs from the terms of the GNU GPL, the more likely that the program's license will restrict rather than broaden the freedoms associated with open source code distribution and copyleft.

[79] This "guarantee" is supported by the use of a license. The license is called a general public license or GPL, and it binds anyone who consents to its provisions by downloading or purchasing an open source program. GPLs vary widely, but most have their genesis in the GNU GPL drafted by the Free Software Foundation, which, in many respects, is notably at the forefront of the open source movement. Generally, a GPL achieves three goals: it designates ownership of copyright in the open source project; it grants everyone the right to modify, copy, and distribute the source code and the derivative program; and it sets distribution terms. The distribution terms require software developers, who produce programs using the original source code, to re-distribute all source code along with the GPL, and to distribute the source code in a form that is open to others or made available to others. In this regard, the phrase "open source code" is an appropriate reference to the GPL.

[80] "Free" is not a reference to the cost of the software. Instead, it refers to the free(dom) to change the source code and re-distribute it as a derivative program. It is entirely permissible to charge a price for use of software produced by the open source movement. *See* Berkman Center for Internet and Society, *The Power of Openness: Why Citizens, Education, Government and Business Should Care About the Coming Revolution in Open Source Code Software (A Critique and a Proposal for the H2O Project)*, 1999 <http://opencode.org/h2o/> (visited Feb. 8, 2000). Since free software may cost money, it is confusing to use the label "free software movement" to denote what is going on. Hence, "open source movement" is preferred.

[81] *See* David Betz and Jon Edwards, *Richard Stallman discusses his public-domain UNIX-compatible software system*, BYTE Magazine, July 1986; John Perry Barlow, *The Economy of Ideas: A Framework for Rethinking Patents and Copyright in the Digital Age*, Wired 2.03, Mar. 1994, at 85.

[82] Although one could argue that some software authors keep their source code secret because they view the source code as a trade secret, the principle argument, here, is that this too is a mistaken notion, if not a dubious proposition. As explained *infra*, most source code contains freely expressed public domain material or basic ideas. Keeping these ideas secret or hidden under the guise of copyright may lead to devastating results far afield from the objectives of copyright; to use a well-worn example, Microsoft may

growing number of organizations are bucking the trend,[83] especially in cyberspace.[84] Source code creation is now, largely, an open and shared process in cyberspace. Programmers engage in code sharing efforts on web sites, on Internet bulletin boards and newsgroups, and in e-mail exchanges.[85] In this regard, source code is similar to the pieces of a jigsaw puzzle; the pieces of any given puzzle are rather non-unique and available for use by anyone with access to the puzzle, but, to some, the pieces will remain a puzzlement. They will not be able to put the pieces together meaningfully. To others, the pieces will fit neatly together again and again. Even so, jigsaw pieces, like subroutines in a given

---

have used its copyright interests in its operating systems software as a devastating corporate armamentarium to suppress the progress of science by anyone but, itself. *See also* Charles C. Mann, *Who will Own Your Next Good Idea?*, ATLANTIC UNBOUND, (Roundtable: Life, Liberty and the Pursuit of Copyright) September 10, 1998.

[83] An influential paper subsequently published as a book by an open source software advocate—Eric Raymond—was first published in May 1997. *The Cathedral and the Bazaar*, <http://www.tuxedo.org/~esr/writings/cathedral-bazaar/> (visited November 13, 1999). Raymond's paper was reportedly expressly cited by Netscape management as a motivation for their decision to release browser source code. The new programming paradigm acknowledges that "good programmers know what to write, and great programmers know what to rewrite" (and reuse). *Id.* In this regard, open source code programmers are more likely to efficiently and openly reuse code than traditional programmers not simply because they are always guaranteed access to the entire source code, but also because they need not waste resources keeping their code secret to either protect an intellectual property interest or avoid apparent notice that their software creation efforts violate the intellectual property interests of others.

[84] *Id.*

[85] Interestingly, at least one author, Ellen Ullman, has noted that the use of "pre-built" or shared code has resulted in the dumbing-down of programming. *See* Ellen Ullman, *The Culture Of Technology, The Technology Of Culture: The Dumbing-Down Of Programming*, SALON MAGAZINE, June 8, 1998 <http://www.salon.com/21st/feature/1998/05/cov_12feature.html> (visited Oct. 27, 1999). According to Ullman, code sharing ensures that programmers do not understand the code they are using and, perhaps much worse, the "knowledge" accessible by programming disappears into the source code unknown or unknowable by the programmer. While this trend, undoubtedly, could have an alarming impact on the goals of the open source code movement, which consider the aims of empowering programmers and computer users as primary objectives of the movement, it is noteworthy that the separation of knowledge from programming seems to be supported by those that view the open source code movement as tremendously dangerous to their own objectives: namely, programming tool makers who support a legal regime that privileges proprietary source code. Admittedly, due to the scope and vast range of programming projects, the line between these two competing trends becomes blurred at critical points.

section of source code, build upon each other, and are largely worthless outside of their respective framework.

The open source movement[86] is an ad hoc, loosely defined, but pervasive, global community of software authors who create software in cyberspace that provides software users with greater control over their computing environment than software developers who produce proprietary source code products.[87]

    1.   *The new trend: reveal the code to the world.*

The new trend: reveal the code to the world's programming community,[88] and let everyone[89] try their hands at improving it.[90]

---

[86] The success of open source is well established in Cyberspace. Indeed, its success in Cyberspace may signify that open source is a viable model of electronic commerce that could extend far beyond the software industry. In this respect, what may be called "open source theory" will provide the core for examining how a dramatic shift in conventional business models will occur over the next decade. *See, e.g.*, Bill Gates, BUSINESS AT THE SPEED OF THOUGHT (1999).

[87] As noted, *infra*, like any movement, the open source code movement is not monolithic; some factions' abhorrence of proprietary source code does not seem to run as deep as their distaste for software produced by ubiquitous software developers. In this regard, the free software faction tends to view the goals of the open source movement as more than merely challenging the economic model of proprietary software development.

[88] Interestingly enough, the specifications for the software protocol that controls the flow of information on the World-Wide-Web, HTTP (Hypertext Transfer Protocol) and the specifications that allow website authors create websites, HTML (Hypertext Markup Language), were developed as open source technologies by the father of the World-Wide-Web, Tim Berners-Lee. Berners-Lee, who is often wrongly referred to as a physicist, released the specifications for web browsers and the web page programming language, HTML, so that others could adapt or improve the specifications for uses beyond the needs of his employer, CERN, an international research center in Geneva. E-mail correspondence between Rod Dixon and Tim Berners-Lee, (Apr. 9 - 21, 1999) (on file with author).

[89] On January 22, 1998, Netscape Communications Corporation (Netscape) made the source code for its popular web browser software available for free licensing on the Internet. Netscape joined the open source code movement in an attempt to harness the creative power of thousands of programmers on the Internet. The open source code movement generally is viewed as a successful way to stimulate the creative energies of the Cyberspace-based programming community. It has been credited with having inspired unprecedented levels of innovation in software development. Netscape manages its open source code project by using a distribution license called Mozilla Public License (MPL) that allows source code modification and redistribution and provides for free availability of source code versions, but has no copyleft provision. Although Microsoft disputes Netscape's claims, Netscape has indicated that it has an install base of more than 68 million users. See generally, Netscape Press Release, *Bold Move To Harness Creative*

Organizations which follow this trend are so numerous on the Internet[91] that the open source movement represents a global movement that is successfully challenging the contemporary proprietary model of software development with a model in which "openness"[92] is considered a virtue.[93] The open source model produces a superior product[94] from input

Power Of Thousands Of Internet Developers; Company Makes Netscape Navigator And Communicator 4.0 Immediately Free For All Users, *Seeding Market For Enterprise And Netcenter Businesses*, <http://www.netscape.com/newsref/pr/newsrelease558.html> (visited November 29, 1999).

[90] This movement unquestionably deviates from the once prevailing view of computer source code as a trade secret. *See* Delta Filter Corp. v. Morin, 108 A.D.2d 991, 992, 485 N.Y.S.2d 143, 144 (3d 1985); Support Sys. Assocs. v. Tavolacci, 135 A.D.2d 704, 706, 522 N.Y.S.2d 604, 606 (2d 1987).

[91] One of Harvard Law School's research centers, the Berkman Center for Internet and Society, was established, in part, to provide a useful focal point for research, discussion, policy analysis and strategic planning for the open source movement. *See generally* <http://cyber.law.harvard.edu/projects/opencode.html> (visited November 29, 1999). For an engaging assessment of the limitations of the open source code movement, see Lawrence Lessig, Code: And other Laws of Cyberspace, 101-110 (1999).

[92] Open source is in some limited respects similar to the "freeware" tradition of distributing software on the Internet. Freeware describes software that is distributed to users at no cost while the software application is under development. In other words, users were granted free use of a software program in exchange for comments about whether the software performed according to expectations. Once a program had been sufficiently improved, many programmers would abandon earlier versions of the software and begin selling the more refined product. In this respect, the open source movement significantly extends the freeware development process far beyond cheap labor and smart marketing by allowing others to freely develop new programs using original source code.

[93] The open source movement is not without its thoughtful critics. Aside from the many critics of the Free Softtware Foundation, the seminal paper, "The Cathedral and the Bazaar," written by Eric S. Raymond and viewed as a manifesto of the non-free software faction of the open source movement, also has been criticized as simplistic and far too idealistic for the commercial enterprise of software development. In it, Raymond seems to assume an open-is-good/commercial-is-bad software development environment, and claims that the goals of software development are pitted between these diametrically opposed alternatives. *The Cathedral and the Bazaar*, <http://www.tuxedo.org/~esr/writings/cathedral-bazaar/> (visited November 13, 1999). According to Raymond, "The Cathedral" represents a monopolistic, property rights-centered style of software development, while "the Bazaar" represents freedom, community, and information sharing. Although these characterizations do not seem far off the mark there are notable exceptions, including the Netscape Navigator web browser, which has roots in both open source and proprietary software development. Charis DiBona, et al., eds., Open Sources: Voices from the Open Source Revolution (1999).

[94] See Eric Raymond, *Homesteading on the Noosphere* <http://www.tuxedo.org/~esr/writings/homesteading/homesteading.html> (visited April

from potentially hundreds of programmers as well as reinforces market competition by precluding "lock in" to proprietary[95] technology.[96] In other words, open source programming[97] eschews the use of software

---

24, 2000). Raymond discusses the difference between economically motivated exchange (e.g. commercial software development for money) and "gift exchange". "Homesteading" is acquiring property by being the first to 'discover' it or by being the first to make a significant contribution to it. The "Noosphere" is loosely defined as the "space of all work" or the community of open source code programmers. In gift cultures, social status is determined not by what you control but by what you give away. In this regard, Noosphere is a gift culture, wherein source code and software is freely shared. This communal sharing shifts the primary measure of success from income to reputation among peers as a gifted and highly regarded programmer.

[95] Since the technology of the Internet is based largely on open standards, the open source code movement views issues of Internet governance a primary concern to the movement. Therefore, attempts by Microsoft and Sun Microsystems to privatize the technology of Cyberspace have not been well received. Similarly, efforts by the Internet Corporation for Assigned Names and Numbers (ICANN) to influence or establish Internet protocol standards have been viewed with suspicion. See Lawrence Lessig, *The Code of Cyberspace*, THE INDUSTRY STANDARD, Dec. 6, 1999, at 184-188.

[96] In this regard, open source programming may weaken the anti-competitive effects of the proprietary software development practice, where developers invest creative efforts in proprietary software design methods in order to create or strengthen position in markets, lock in their technology, and limit consumer choice. Indeed, many commentators have noted that the current intellectual property regime, which has granted Microsoft "property rights" in its immensely successful operating system software, is a significant factor supporting Microsoft's continued market power in the operating system software market. *See* Mark A. Lemley, *Antitrust and the Internet Standardization Problem*, 28 CONN. L. REV. 1041 (1996); S.J. Liebowitz & Stephen E. Margolis, *Network Externality: An Uncommon Tragedy*, 8 J. ECON. PERSP. 133(1994); see also Pamela Samuelson, *The Copyright Grab*, WIRED 4.01 (Jan. 1996). It should also be acknowledged that to some degree Microsoft's ability to squeeze nearly $8 billion dollars of net profit from $20 billion dollars in corporate sales may be due to the increasing returns to scale that seems to accompany market leadership in software development. *See, e.g.*, James Fallows, *Billion-Dollar Babies*, THE NEW YORK REVIEW OF BOOKS Vol. XLVI, No. 20, Dec. 16 1999, at 9.

[97] "Open source" does not just refer to free or open access to source code; a public license must accompany source code distributed as open source code and the license must comport with the standards of the open source movement, containing terms that ensure that those who use the software may do so freely (that is, there is no restriction on copying or modifying the source code) and those who distribute the software do so in accordance with the terms of the original public license. See *Frequently Asked Questions About Open Source*, <http://www.opensource.org/faq.html> (visited Apr. 16, 2000). It should be noted that there is considerable debate among those within and outside the open source code community as to what terms of the public license are enforceable. Although the issues raised by that debate are interesting questions, there are outside the scope of this article.

development to strategically control markets without regard to the production of superior software applications.[98]

The most famous open source project is probably Linux, a version of the Unix operating system.[99] Its proponents distributed a version the code on the Internet several years ago, and hundreds of programmers have added their own refinements.[100] The result is claimed to be a much faster, less crash-prone operating system than Microsoft's Windows operating systems.[101]

The link between open source and public domain should not be overstated. Although open source projects have the attributes of the public domain, they are not public domain works. One open source project can rarely begat another. The individual or organization managing the open source project retains the exclusive copyrights to the

---

[98] Some software developers have argued that source code encapsulated in compiled object code actually is kept private in order to protect trade secrets. However, copyright law exists to aid authors to protect their creative, original expression, not to withhold that expression under claims of trade secrecy.

[99] Unmistakably, the open source code movement is about a great deal more than a preferred method or style of programming; the movement, if successful, could challenge conventional economic principles of marketplace competition and traditional political notions of democratic decision-making. In this respect, openness is a virtue not just because it may produce a superior software product, but also because it reinforces free market competition and progressive principles. Linux, for example, represents a fundamental attack on the traditional association between technology and property. It is attempting to infuse principles of openness and shared intellectual property into common conceptions regarding property.

[100] *See Open Source Movement*, <http://www.opensource.org/links.html> (visited Nov. 13, 1999).

[101] One of the leading successes of the open source movement is that the Apache software, a totally free Web server program that runs on several different operating systems, controls more than half the world's Web servers. According to Netcraft, a networking analyst in Bath, England, claiming to have surveyed more than 4.3 million sites, almost 55 percent of Web sites use Apache, followed by Redmond, Wash.-based Microsoft with 24 percent; Mountain View, Calif.-based Netscape with about 4 percent; and Sebastopol, Calif.- based O'Reilly & Associates Inc. with less than 2 percent. The survey is available at <www.netcraft.com/survey/>. Interestingly, none of the commercial Web server vendors is terribly concerned about Apache, instead stressing their market segments, channel programs and support. Indications are that large consumer-oriented sites are moving to commercial products to take advantage of better support, transaction abilities, security and, most importantly, administration, according to Forrester Research Inc., Cambridge, Mass. "It may be free to acquire the bytes, but over the long term, it is probably cheaper to buy Netscape for the support.

works created or derived from the original code. The General Public License (GPL), on which open source projects are based, grants non-exclusive rights to distribute or copy the original source code or to make derivative works based on the original source code. In this regard, works created from open source do not provide the same benefit to authors that works created from public domain could provide. Of course, the point of the open source movement is that copyright is not necessary to promote the progress of science -- at least, in so far as computer programs are concerned. Authors create works due to reputational benefits and other economic advantages.

Even so, the one obvious, practical, and destabilizing effect of an open source project based on GPLs is the potential for the project to implacably acquire copyright interests in all the works for which the original source code is based.[102] (Genuine public domain projects could not lead to the same result because copyright would not aggregate in one original "author."). In this regard, open source could, ironically, distort the goals or aims of copyright. Since the GPLs do not preclude the original copyright owner or the open source code project, itself, from altering the terms of the GPL at the some later time, the original owner could attempt to extract royalty fees from all participants who fail to abandon their "free" efforts.[103] Moreover, the meaning of "derivative" work may be strained under the terms of the GPL. Those issues notwithstanding, the economic model of open source makes available a powerful disincentive to those who would attempt to distort the goals of the movement. More important, to date, open source has not been noticeably abused, and it has provided a compelling challenge to the rarely challenged notion that the grant of copyright is necessary (albeit, not sufficient) to incent authors to create works that promote the progress of science and the useful arts.

---

[102] Some refer to this phenomenon as the "viral" effect of the GPL. If the source code of two programs are combined, one a non-GPL program and the other a GPL program, in most circumstances, the terms of the GPL infect the non-GPL program by imposing the distribution and other licensing restrictions on the entire derivative work.

[103] Notably, the open source movement does not, generally, view revocable licenses as genuine GPLs and, hence, not as open source software. Nonetheless, there are open source projects with GPLs that may be lawfully altered by the original copyright owner. In one well-known instance, the "Open Group," which manages the X-Window open source project, attempted to collect royalties for its copyright grants by altering its GPL. The open source community quite characteristically opposed this "copyright grab" and convinced the Open Group to abandon the unfavorable terms of its new public license.

2. *Copyleft: a clear paradigm shift in programming.*

Cyberspace is the location of the birth of a subset of the open source movement called Copyleft. In the Copyleft or free software movement, the emphasis is not solely on increasing the amount of open source code available for general use, but also includes the objective of disenabling incentives to copyright software.[104] Software is distributed freely with a GPL, but the GPL includes a "copyleft" provision rendering it difficult for individual authors to freely convert open source software into proprietary programs. Although Copyleft allows anyone who redistributes the software, with or without changes, to pass along the freedom to further copy and change the source code, the copyleft provision requires that distribution terms in the GPL remain unchanged when a derivative work is distributed,[105] and that all code added to the program or distributed as part of the program be free(dom) software.

To copyleft a program, the author first must hold the lawful copyright to the source code. Then, distribution terms are attached to the source code, which usually grants anyone the right to use, modify, and redistribute the program's code or any program derived from it, but only if the distribution terms are unchanged. Thus, as noted, the code and the freedom to decide how one may use the code are viewed as legally inseparable. In this regard, Copyleftists[106] use copyright to guarantee an

---

[104] What is Copyleft?, (visited November 13, 1999) <http://www.fsf.org/copyleft/copyleft.html>.

[105] In this manner, copyleft ensures that "code and freedoms become legally inseparable." *Id.* In the open source movement, copyleft is controversial and is often not included in the GPLs used by the non-free software faction of the movement. The controversy arises, in part, on the boldness of the copyleft. Copyleft precludes the distribution of open source software with proprietary source code and it halts the conversion of free software to non-free software. In some respects, copyleft provisions are viewed as necessary to thwart the erosion of open source projects by Triple E tactics (embrace, extend and extinguish) often, but not solely, used by Microsoft Corporation. Whether copyleft provisions are lawfully enforceable or may effectively prevent pernicious code-forking is still an unanswered question. The irony of the copyleft provision is that it seems to have a tendency to weaken an author's freedom to develop software almost as much as it may temporarily enhance some of those freedoms. More importantly, without revising copyright law as it applies to software, copyleft provisions ultimately may have the same adverse impact upon the public domain of source code as proprietary software development models. A detailed analysis of copyleft provisions is certainly warranted.

[106] A short-hand reference to the faction of the open source movement supporting the use of copyleft provisions in GPLs.

author's freedom, rather than to limit it.[107] Accordingly, the proponents of the movement viewed it appropriate to reverse the term "copyright" to "copyleft."[108] Licenses that accompany source code and allow for privatization of publicly developed open source software products are licenses that permit what open source code programmers call "code-forking."[109] Code-forking allows a software author to take publicly available source code, and create a proprietary product out of the code. Copyleftists attempt to limit the practice of code-forking, but do not necessarily preclude it entirely.[110]

The open source code movement represents a clear paradigm shift in programming.[111] Traditionally, computer programming was a solitary task performed by a programmer on a single machine.[112] Other programmers or other machines could not understand most of the programmer's code. The instructions to a computer, or program, must be given to the computer in the form of 'machine language' notation. Machine language is, however, difficult for humans to comprehend. Generally, instead of writing machine language instructions that the

---

[107] *Id.*

[108] *Id.*; OPEN SOURCES: VOICES FROM THE OPEN SOURCE REVOLUTION (Charis DiBona et al. eds., 1999).

[109] OPEN SOURCES, *supra* note 108. The factionalism in the open source movement was enhanced by the increasing tendency of some GPLs without copyleft provisions to ostensibly permit code-forking.

[110]Since code-forking can aid Copyleftists overcome the attempts of others to obstruct the objectives of open source by "privatizing" the code, Copyleftists do not endeavor to preclude all avenues to code-forking. For example, an open source program may avert "death" from a Triple E attack by code-forking its own source code away from a company like Microsoft, and subsequently, re-opening the code as a re-claimed open source project that competes along side the proprietary product(s). Code-forking is by no means the only way to undermine an open source project. Even though most open source projects may level, at least, a soft blow to the proprietary model of software development, to a significant extent, the failure to copyleft an open source project renders the capacity of the GPL to safeguard freedom software dubious. The most notable example of weak capacity of non-copyleft GPLs is Netscape Corporation's (now, America Online and soon-to-be Time Warner AOL) public license for its browser software, Netscape Navigator. The Netscape Navigator is currently being developed as an open source project governed by a so-called public license, Mozzila Public License (MPL). The MPL, however, has no copyleft provision, and contains considerable restrictions on freedom software. Indeed, the MPL renders Netscape Navigator's development model more like a beta freeware software project than an open source project.

[111] *See generally* OPEN SOURCES, *supra* note 108.

[112] Peter Norton, PETER NORTON'S INSIDE THE PC, 454 – 456 (6th ed. 1995).

processor can execute directly, programmers write programs in a programming language, which then is translated mechanically to machine language by a compiler program.[113]   Some programmers write in assembly languages rather than in high-level programming languages.[114] Assembly language is a form of machine code that humans can read.[115] A programmer who writes a program in assembly language must approach the problem-solving task in the same systematic fashion that the computer will use.[116]

A program written in an Assembly language is converted into machine code by an assembler program. Assembly language programs generally run faster than higher-level language programs.   Programs written in either high-level programming language or assembly language are called 'source code' programs.   Machine-readable object code, incomprehensible to people, consists of a string of ones and zeros, which are the only two symbols a digital computer can understand.[117]

In some sense, software design is a creative process that programmers learn more through practice than from books, a process that cannot be formulated as a set of rules.   In this respect, the end product, the program, is generally the result of numerous conscious choices by the programmer[s].[118]

Microcomputers significantly changed the task of the programmer. Software engineers designed programming languages that could be understood by others and run on computers built under specifications.[119]   As cyberspace has become an increasingly useful environment for computer programming, programmers are developing a

---

[113] *Id.*

[114] *Id.*

[115] *Id.*

[116] *Id.*

[117] *Id.*

[118] In the 1970s Congress and the Commission on the New Technological Uses of Copyrighted Works (CONTU) included computer programs in the category of works that contain expression that may be protected without granting a monopoly on underlying methods or ideas. See CONTU, *supra*, and generally, *see*, Fred H. Cate, *Law in Cyberspace*, 39 How. L.J. 565, 575-77 (1996); Margaret Chon, *New Wine Bursting from Old Bottles: Collaborative Internet Art, Joint Works, and Entrepreneurship*, 75 Or. L. Rev. 257, 260-61 (1996).

[119] *See also* Raymond T. Nimmer & Patricia Ann Krauthaus, *Copyright on the Information Superhighway: Requiem for a Middleweight*, 6 Stan. L. & Pol'y Rev. 25 (1994).

reluctance to systematically "hide" their source code from each other.[120] Instead, programming began undergoing a paradigm shift[121] away from viewing source code as the province of secrecy and toward sharing source code in the form of reusable modules or objects.[122] In an attempt to build more complex applications that could run on desktop computers and local networks, programmers routinely share and exchange code that can be used and re-used for various projects.[123] In this regard, source code modules are distributed as works in the public domain.[124]

Programmers write computer code in one of several high level languages, such as Visual Basic, COBOL, BASIC, FORTRAN, JAVA, C++ or a low level language such as an ASSEMBLY language.[125] Many software programs that run on computer networks and on the Internet are actually written in scripting languages like PERL. Scripting languages blur the traditional distinctions between source code and object code because most programs written in scripting languages are executed and

---

[120] Indeed, source code written in Cyberspace-based programming languages like PERL and JAVASCRIPT is, generally, easily accessible by others, and can be read in any text editor. This open "feature" has enhanced code sharing even outside of the open source code movement. On the other hand, some programmers may have unknowingly sacrificed enforcement of their copyright interests as a result of using programming tools unsuitable for proprietary source code development. See Edward A. Cavazos & Gavino Morin, Cyberspace and the Law 47-48 (1994); James Gleick, *I'll Take the Money, Thanks*, N.Y. TIMES Magazine, Aug. 4, 1996, at 16; Thomas K. Landry, *Roundtable on Electronic Rights*, 20 COLUM.-VLA J.L. & ARTS 605, 658 (1996); Pamela Samuelson, *The Copyright Grab*, WIRED, Jan. 1996.

[121] White Paper, The Power Of Openness: Why Citizens, Education, Government and Business
Should Care About the Coming Revolution in Open Source Code Software; A Critique and a Proposal for The H20 Project, Berkman Center For Internet and Society, 1999, (visited Nov. 15, 1999) <http://opencode.org/h2o/>.

[122] Po Bronson, NUDIST ON THE LATE SHIFT (AND OTHER TRUE TALES OF SILICON VALLEY), (Random House 1999) 112-114; see also Lawrence Lessig, *Intellectual Property and Code*, 11 ST. JOHN'S J. LEGAL COMMENT 635 (1996).

[123] See Charis DiBona, Sam Ockman & Mark Stone, editors, OPEN SOURCES: VOICES FROM THE OPEN SOURCE REVOLUTION (O"Reilly & Associates, 1999).

[124] Technically speaking, modules and subroutines are slightly different. Compare R. Andree, J. Andree & D. Andree, Computer Programming 336 (1973) (defining subroutines) with R. Coats, Software Engineering for Small Computers 12 (1983) (modules).

[125] See Peter Norton, PETER NORTON'S INSIDE THE PC, at 454 – 456 (6th ed. 1995); Apple Computer, Inc. v. Franklin Computer Corp, 714 F.2d 1240, 1243 (3d Cir.1983), *cert. dismissed*, 464 U.S. 1033 (1984).

compiled at the same time. While this distinction has unclear relevance to copyright, it very well may be a significant factor in assessing the government's regulation of the export of an encryption program.

The most significant constraint on an open source code project may involve finding enough programmers available and interested in contributing their time jointly authoring freely available software projects.[126] In this respect, cyberspace has provided the tools necessary to bring together enough people to harness the intellectual efforts required to create serious software programs sufficient to support the paradigm shift in programming.[127] It is quite possible that the growth of the Internet will complete the programming paradigm shift since enough cyberspace-based programmers will be available to make both large scale projects and small programming alliances viable and routine.[128] Open source code collaborative programming efforts may become standard.[129] In this regard, the existence of the open source code movement amply supports a conception of copyright that provides sufficient incentive for software developers to create works from a vast public domain; a public domain that would provide access to source code for authors and, in turn, encourage software authors to create works that could promote the progress of science, thereby, further enrich the public domain.

V. PART II LIMITING THE EXPANSION OF COPYRIGHT

Copyright law protects original authorship, but sets up a dichotomy between the protected work and the idea within the work. Any original work of authorship that exists in tangible form is copyrightable.[130] Copyright protection, however, extends only to the

---

[126] *See generally* William C. Taylor, *Inspired by Work: Working on Open source Software*, FAST COMPANY, 200-208, (Nov. 1999).

[127] Josh McHugh, *For the Love of Hacking*, FORBES MAGAZINE, (August 10, 1998).

[128] In its present mode, the open source movement is hardly inconsequential. All Internet users rely upon at least one open source code software application to navigate through Cyberspace.

[129] *See*, John Perry Barlow, *The Economy of Ideas*, WIRED, Mar. 1994, at 84, 85.

[130] The Supreme Court has found that the idea/expression dichotomy strikes a definitional balance between the First Amendment and the Copyright Act by permitting free communication of facts while still protecting an author's expression. Harper & Row Publishers, Inc. v. Nation Enter., 471 U.S. 539, 556 (1985) (quoting the Second Circuit in 723 F.2d 195, 203 (1983)). In Harper & Row, the Supreme Court held that Time

particular expression of the ideas contained within the work, not to the ideas themselves.[131] Under the law of copyright, an idea is thought to ordinarily encompass many means of expression. Consequently, the idea/expression dichotomy, along with other limiting doctrine, thwarts the unintended effect of copyright to allow an author to gain control over an idea simply by expressing it in one tangible form. In this respect, the First Amendment should cut off the implacable drive to diminish, if only "temporarily,"[132] the stock of raw materials available to other authors.[133]

The distinction between ideas and expression is supposed to provide a way of reconciling two competing interests -- the interest in rewarding ingenuity and the interest in allowing the public to benefit from new works by other authors on the same subject. Since the function of copyright is to promote creativity so that the public may benefit from the labor of authors, the Federal government provides authors with an incentive to create by granting them the exclusive right to profit from and control specified uses of their works.[134]

Magazine infringed on a copyright when it published, without authorization, verbatim quotes, totally 300 words, from President Ford's memoirs. *Id.* at 569.

[131] Synercom Technology, Inc. v. University Computing Co., 462 F.Supp. 1003 (N.D. Tex. 1978), dealt with the question whether the "input formats" of a computer program – the configurations and collations of the information entered into the program – were idea or expression. The court held that the input formats were ideas, not expressions, and thus not protectable.

[132] Congress has extended the limited time period for which copyright interests may be claimed. The Sonny Bono Copyright Term Extension Act ("CTEA"), Pub. L. 105-298, 112 Stat. 2827 (1998) (codified in scattered sections of 17 U.S.C.) extends by twenty years the basic term of copyright protection for both newly created works of authorship and most pre-existing works with subsisting copyrights. This recent Congressional extension of copyright is under challenge before the Court of Appeals for the District of Columbia Circuit in Eldred v. Reno, No. 99-5430 (the case is scheduled for oral argument on October 5, 2000).

[133] Of course, within the limits of the constitutional grant, Congress has broad flexibility "to implement the stated purpose of the Framers by selecting the policy which in its judgment best effectuates the constitutional aim." Graham v. John Deere Co., 383 U.S. 1, 6 (1966); That notwithstanding, stimulating creative activity by authors is, of course, one of the principal purposes of copyright protection. Sony Corp. of America v. Universal City Studios, 464 U.S. 417, 429 (1984).

[134] In general, where a particular expression is common to the treatment of a particular idea, process or discovery, it is lacking in the originality required for copyright. Gates Rubber Co. v. Bando Chemical Industries, Ltd., 9 F.3d 823, 838 (10th Cir. 1993).

As noted, copyright also has the powerful capacity to diminish the potential for creativity.[135] The exclusive rights granted by copyright may hinder the efforts of new authors who seek to build on the creativity of the past.[136] It is in this regard that the idea/expression dichotomy helps copyright strike a balance between providing incentives to create and maintaining the store of raw materials needed for new creations.[137] However, under the dichotomy, the boundary between unprotectable ideas and protectable expression is often difficult to discern.[138]

---

[135] Feist Publications, Inc. v. Rural Tel. Serv. Co., Inc., 499 U.S. 340, 358 (1991) (originality requires that the selection and arrangement exhibit some minimal level of creativity).

[136] In the context of a copyright infringement claim that includes an allegation about source code, the elements of a copyright infringement action include a showing that the plaintiff owned the copyright in the source code and that the defendant copied the source code in developing its program. Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp., 562 F.2d 1157, 1162 (9th Cir.1977); Reyher v. Children's Television Workshop, 533 F.2d 87, 90 (2d Cir.), *cert. denied*, 429 U.S. 980 (1976); 3 NIMMER ON COPYRIGHT § 13.01 (1985). As it is rarely possible to prove copying through direct evidence, Roth Greeting Cards v. United Card Co., 429 F.2d 1106, 1110 (9th Cir.1970), copying may be proved inferentially by showing that the defendant had access to the allegedly infringed copyrighted work and that the allegedly infringing work is substantially similar to the copyrighted work. Ferguson v. National Broadcasting Co., 584 F.2d 111, 113 (5th Cir.1978); Sid & Marty Krofft Television Prods. Inc., *supra*; Universal Athletic Sales Co. v. Salkeld, 511 F.2d 904, 907 (3d Cir.), *cert. denied*, 423 U.S. 863 (1975); Midway Mfg. Co. v. Strohon, 564 F.Supp. 741, 752 (N.D. Ill. 1983).

[137] The idea/expression dichotomy is presumed to strike a definitional balance between the First Amendment and the Copyright Act. No author may copyright her ideas or the facts she narrates. 17 U.S.C. § 102(b); *See, e.g.,* New York Times Co. v. United States, 403 U.S. 713, 726 (1971) (Brennan, J., concurring) (summarily concluding that Copyright laws are not restrictions on freedom of speech as copyright protects only form of expression and not the ideas expressed).

[138] The idea/expression dichotomy has been further refined as the abstraction-filtration-comparison method of determining copyright protection for computer programs. Gates Rubber Co. v. Bando Chemical Indus., 9 F.3d 823 (10th Cir. 1993). The Tenth Circuit defined the method as follows: "First, in order to provide a framework for analysis, we conclude that a court should dissect the program according to its varying levels of generality as provided in the abstractions test. Second, poised with this framework, the court should examine each level of abstraction in order to filter out those elements of the program that are unprotectable. Filtration should eliminate from comparison the unprotectable elements of ideas, processes, facts, public domain information, merger material, scenes a faire material, and other unprotectable elements suggested by the particular facts of the program under examination. Third, the court should then compare the remaining protectable elements with the allegedly infringing program to determine whether the defendants have misappropriated substantial elements of the plaintiff's program."

Presumably, when copying is literal, an idea can easily be isolated from its expression.[139] Perhaps, the most important part of the public domain constitutes those works comprising copyrighted material and material that copyright does not protect. In other words, the public domain includes works that contain both copyrightable and uncopyrightable aspects. It is in this regard that the highly regarded Judge Learned Hand noted the conundrum that such works pose for the judge reviewing a copyright infringement action, in a seminal case involving a play entitled Abie's Irish Rose:

> "We assume that the plaintiff's play is altogether original, even to an extent that in fact it is hard to believe. We assume further that, so far as it has been anticipated by earlier plays of which she knew nothing, that fact is immaterial. Still, as we have already said, her copyright did not cover everything that might be drawn from her play; its content went to some extent into the public domain." [140]

The concept that portions of works protected by copyright are owned by no one and are available for any member of the public to use is such a fundamental one that it receives attention only when something seems to have gone awry. Although the public domain is implicit in all commentary on intellectual property, it rarely takes center stage. But a vigorous public domain is a crucial buttress to the copyright system; without the public domain, it might be impossible to tolerate copyright at all.

A. *The statutory definition of a computer program.*

The Copyright Act defines a computer program as "a set of statements or instructions to be used directly or indirectly in a computer

---

[139] Even if that is not so, the dichotomy could not help. It does not apply to infringement actions based on literal copying.

[140] In Nichols v. Universal Pictures Corp., 45 F.2d 119, 122 (2d Cir. 1930), *cert. denied*, 282 U.S. 902 (1931), the court found that no more than the idea of plaintiff's play, "Abie's Irish Rose," had been used in defendant's motion picture "The Cohens and the Kellys." Although common to both works were a "quarrel between a Jewish and an Irish father, the marriage of their children, the birth of grandchildren and a reconciliation," the court held that these were ideas, not expressions, and, therefore, not subject to copyright protection. *Id.* at 122.

in order to bring about a certain result."[141] Indeed, software authors can and do bring copyright infringement suits against other software authors.[142] In such a case, the defending author is likely to argue that, if any copying at all occurred, he took merely unprotected ideas from the other work. Under current copyright doctrine, a copyright infringement claim involving source code entails an allegation of literal copying. As such, a claim by the defendant that he only copied unprotected ideas would require the reviewing court to either apply the idea/expression dichotomy or to reject the defense on its face as inconsistent with the evidence of literal copying. It is in this regard that courts that apply the dichotomy are required to separate a software program's protected expression from its unprotected ideas, including the relevant aspects of the source code.[143]

Although the legislative history of the 1976 Copyright Act indicates that Congress intended for the revised copyright statute to protect computer programs, courts did not agree on the contours of what constituted a computer program under the Copyright Act until Congress amended the Copyright Act through its enactment of the Computer Software Copyright Act of 1980. This amendment to the 1976 copyright law added the definition of "computer program".[144]

---

[141] In writing these directions, the programmer works "from the general to the specific." Whelan Assoc., Inc. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 1229 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987). *See generally* Steven R. Englund, Note, *Idea, Process, or Protected Expression?: Determining the Scope of Copyright Protection of the Structure of Computer Programs*, 88 MICH. L. REV. 866, 867-73 (1990).

[142] Before a computer program can qualify for copyright protection, it must meet a number of general requirements outlined in the copyright statutes. First, the work (computer program) must be "fixed" in a tangible form "in which, or by means of which, other people can perceive it." The moment of "fixation" marks the start of federal copyright protection, assuming the program meets the Copyright Act's requirement of originality. "Originality," in this regard, is a term of art. If a computer programmer developed a particular program that was identical in all respects to a previously copyrighted program, and the programmer had not had access to or knowledge of the previously copyrighted program, the second programmer could still obtain a copyright for the work. However, certain elements-such as facts, scientific discoveries, mathematical equations, and historical theories-although newly discovered by an author, are not "original" within the meaning of the copyright statutes. *See, e.g.*, Feist v. Rural Telephone Services, 499 U.S. 340 (1991).

[143] *See also* Circular 61 Copyright Registration for Computer Programs, Register of Copyrights, Library of Congress.

[144] A 1976 House of Representatives committee report explained the applicability to computer programs of Section 102(b) of the 1976 Copyright Act, which codified the rule

B. *The Distinction between an Idea and its Expression.*

As an analytic tool, the idea/expression dichotomy is viewed as a useful way of not only defining the scope of the plaintiff's copyright protection, but also as an aid for the court's ultimate determination of whether the plaintiff's case is missing an essential element of a copyright infringement claim.[145] In addition, the idea/expression dichotomy may help courts avoid overprotection of a copyrighted work, while at the same time aid courts in not losing sight of expression that might be present in a work consisting largely of licensed and/or unprotectable elements.

The 1879 case of *Baker v. Selden* laid the foundation for the idea/expression dichotomy.[146] In *Baker*, the plaintiff sought protection for bookkeeping forms contained in a book that explained a novel method of bookkeeping.[147] The Supreme Court held in Baker that the copyright on a book does not give the copyright holder "an exclusive property in the art described therein."[148] Furthermore, the Court ruled, where use of the idea necessarily requires copying of the work itself, that copying is not infringement.[149] Accordingly, an author may not use the copyright laws to obtain a monopoly on a system or method.[150] Subsequently, the Court began interpreting *Baker* as unequivocally having held that copying an idea without copying the expression does not

---

that copyright protection does not extend to ideas. The committee report stated that Section 102(b) would clarify that a computer programmer's expression is a copyrightable element of a computer program, but that the actual processes or methods embodied within the program are outside the scope of copyright law *See, e.g.*, Michael E. Johnson, Note, *The Uncertain Future of Computer Software Users' Rights in the Aftermath of MAI Systems*, 44 DUKE L.J. 327 (1994).

[145] Baker v. Selden, 101 U.S. 99 (1879) (allowing free borrowing of accounting system described in a copyrighted book). The ambiguities and unstated assumptions in the idea/expression dichotomy are specifically discussed, *infra*.

[146] Amy B. Cohen, *Copyright Law and the Myth of Objectivity: The Idea-Expression Dichotomy and the Inevitability of Artistic Value Judgments*, 66 IND. L.J. 175, 184-95 (1990).; *see* Baker v. Selden, 101 U.S. 99 (1879). The Supreme Court found nothing copyrightable in Selden's bookkeeping system, and rejected his infringement claim regarding the ledger sheets.

[147] *Id*.

[148] *Id.* at 101.

[149] *Id*.

[150] *Id*.

constitute infringement.[151] Because copyright does not protect against the taking of ideas, the substantial similarity that serves as indirect proof of copying must be expression-level similarity, not merely idea-level similarity.[152]

C. *Copyright Protection for Source Code*

By 1983 the Copyright Office had issued a circular stating that '[c]opyright protection extends to the literary or textual expression contained in [a] computer program.'[153] Courts, then, began summarily applying the idea/expression dichotomy to software programs as a proxy for constitutional analysis of the inherent First Amendment question implicated by the idea/expression dichotomy.[154] In courts' inconsistent application of the idea/expression dichotomy it becomes palpably apparent that the broad concept of "expression" used in traditional literary works infringement cases seems strained when applied to software. This may be particularly true because computer programs are, in significant respects, artifacts of technology. Within the U.S. regime of intellectual property, copyright has not been the traditional province of protection for technological devices; that domain belonged to the law of trade secrets and patents.

A seminal software copyright case, *Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc*., extended copyright protection to the non-literal elements of a computer program including, *inter alia*, the program's structure, sequence, and organization. At issue was whether the defendants, in copying file structures, screen outputs, and five subroutines of a dental office software program, copied unprotected ideas or protected expression. The court deemed a program's unprotected ideas

---

[151] 3 Nimmer, § 13.03[A][1], at 13-24.

[152] Copyright protection extends beyond a literary work's strictly textual form to its non-literal components. Nichols v. Universal Pictures Co., 45 F.2d 119, 121 (2d Cir. 1930) (L. Hand, J.), cert. denied, 282 U.S. 902 (1931). Where "the fundamental essence or structure of one work is duplicated in another," 3 Nimmer, § 13.03[A][1], at 13-24, courts have found copyright infringement.

[153] *See, e.g.*, Pamela Samuelson, *Digital Media and the Changing Face of Intellectual Property Law*, 16 RUTGERS COMPUTER & TECH. L.J. 323, 332-34 (1990).

[154] *Id.* (noting those that support continued support of the dichotomy in the context of computer programming). On the other hand, the open source code movement is one compelling example of why source code, in particular, and perhaps software, in general, should be treated differently from traditional literary works by copyright jurisprudence.

to be its purpose or function along with anything in the program necessary to that purpose.[155] Although it is apparent that in some sense any aspect of a computer program could be deemed essential or necessary to its overall purpose,[156] the *Whelan* court presumably considered its articulation of the idea/expression dichotomy meaningful because "[w]here there are various means of achieving the desired purpose…the particular means chosen is…expression, not idea."[157]

The *Whelan* test is unsatisfying because of its circularity.[158] To say that an idea may be distinguished from its expression in a software program by isolating the program's purpose, in addition to 'whatever else is necessary to that purpose', offers little principled distinction. Instead of relying upon a lucid application of the dichotomy, the *Whelan* court recasts the dichotomy in broad and ambiguous terms as an assessment of purpose(s). It may be that the court had determined that at a high-level abstraction a typical software application has only one general purpose.[159] Nonetheless, this determination is hardly persuasive in the context of today's increasingly complex software programs that purposefully serve multiple functions. Indeed, in light of contemporary programming practices, the *Whelan* analysis devours the utility of the idea/expression dichotomy by summarily limiting the conception of an "idea" and, hence, broadly supporting an expansive scope of copyright protection for computer programs. Even if the court's analysis appropriately applied to present-day programming practices, it is not

---

[155] Whelan Assocs. v. Jaslow Dental Laboratory, Inc., 797 F.2d 1222 (3d Cir.1986), *cert. denied*, 479 U.S. 1031 (1987).

[156] 797 F.2d at 1225.

[157] *Id*.

[158] The Whelan test has received a mixed reception. While some decisions have adopted its reasoning, *See, e.g.*, Bull HN Info. Sys., Inc. v. American Express Bank, Ltd., 1990 Copyright Law Dec. (CCH) P 26,555 at 23,278 (S.D.N.Y. 1990); Dynamic Solutions, Inc. v. Planning & Control, Inc., 1987 Copyright Law Dec. (CCH) P 26,062 at 20,912 (S.D.N.Y. 1987); Broderbund Software Inc. v. Unison World, Inc., 648 F. Supp. 1127, 1133 (N.D.Cal. 1986), others have rejected it. See Plains Cotton Co-op v. Goodpasture Computer Serv, Inc., 807 F.2d 1256, 1262 (5th Cir.), *cert. denied*, 484 U.S. 821 (1987).

[159] For example, at some abstract level one could argue that the purpose of Microsoft's Windows operating systems is to manage input/output functions and disk drive access on a personal computer. Obviously, such over-generalized descriptions of complex software programs like operating systems would render the utility of the idea/expression dichotomy of dubious worth since it would seem to offer very little First Amendment limitation on the scope of copyright.

apparent why the concept "purpose" is a notable constitutional substitute for "idea" in drawing the line between the First Amendment and copyright.[160] a computer program necessarily contains many ideas that expand far beyond the program's general purpose. Why should a limiting doctrine, set up to distinguish between ideas and expressions, by its own terms ignore the majority of ideas contained in a work before being applied? In this regard, the *Whelan* court leaves open the question why a software program's ideas beyond an assumed singular program purpose cannot be unprotected ideas.[161]

Broderbund Software, Inc. v. Unison World, Inc., was decided a few months after *Whelan.* The court held that two programs' sequences and layout of screens, and method of user feedback were substantially similar. Accordingly, defendant's program was found to be infringing.[162] The plaintiff did not claim that the defendant copied the code or even the structure of its program.[163] Instead, the plaintiff contended that the overall appearance, structure, and sequence of the computer screen displays of the defendant's program infringed the plaintiff's copyright. The court, following *Whelan*, identified the program's overall function -- to create greeting cards -- as the unprotected idea, and held that other expressions of the function were not only possible, they were currently available off-the-shelf in stores. The *Broderbund* court seems to have eviscerated the First Amendment limitation by abandoning any meaningful attempt to distinguish ideas from expression.[164] While the issue was not directly before the court, it

---

[160] Certainly, it is the position of some encryption software programmers that their source code expresses at least two distinct ideas; namely, that encryption algorithms should be made public through access to the source code as a way to conduct certain forms of cryptanalysis and that the encryption program use a cipher of a certain key length. Indeed, these ideas are not only expressed in source code, but are debated in the scholarly literature of cryptology; to wit, encryption software programmers and other cryptographers dispute in the marketplace of ideas the principle strengths and weaknesses of cipher key lengths and various publicly known algorithms.

[161] *See, e.g.*, Stillman v. Leo Burnett Co., 720 F. Supp. 1353, 1358 (N.D. Ill. 1989) (referring to dual usages of term "substantial similarity"); Alan Latman, *"Probative Similarity" as Proof of Copying: Toward Dispelling Some Myths in Copyright Infringement*, 90 Col. L. Rev. 1187, 1189-90 (1990).

[162] Broderbund Software, Inc. v. Unison World, Inc, 648 F.Supp. 1127, 1137 (N.D. Cal. 1986).

[163] *Id.* at 1132.

[164] This is not to say that the Broderbund court is hostile to First Amendment claims in the context of copyright infringement actions. Rather, it may be that the court understood

would logically follow from the court's analysis that the court would "protect" Broderbund's source code in the same manner it did the nonliteral elements of the software.[165]

Perhaps responding to the defects in the *Whelan* analysis, the Second Circuit adopted its own idea/expression test for computer programs in *Computer Associates International, Inc. v. Altai, Inc*. -- the so-called "abstraction-filtration-comparison" test.[166] The court flatly rejected the *Whelan* test and explicitly applied the abstraction test first expounded by Judge Learned Hand.[167]

The court first separated the copyright protected computer program into parts of increasing "abstraction": source code, parameter lists, services, and charts. After applying this part of the test, the court "filtered" the abstracted elements against its idea-expression distinction, and then applied the limiting doctrine of merger, scenes a faire, and public domain.[168] This process resulted in determining which portions of the program were protectable -- specifically, the literal code of the program.[169] To assess infringement, the court "compared" the protected portions of the computer program to the allegedly infringing program and found that the two programs were not substantially similar.[170] Although the *Altai* test does not characterize "ideas" as broadly as the *Whelan* test, some literal elements of a program essentially *filter into the comparison* simply because they are not sufficiently abstract to be determined ideas. In other words, in the context of source code, both the

---

the First Amendment question to be fully resolved by simple application of the *Whelan* court's peculiar interpretation of the idea/expression dichotomy.

[165] Courts, of course, also have noted troubles with the general application of the dichotomy. *See* Denker v. Uhry, 820 F. Supp. 722, 728 (S.D.N.Y. 1992) (noting that "courts have been reluctant to make subjective determinations regarding the similarity between two works"); Steinberg v. Columbia Pictures Indus., 663 F. Supp. 706, 709 (S.D.N.Y. 1987).

[166] Computer Associates International, Inc. v. Altai, Inc. 982 F.2d 693, 706 (2d Cir. 1992).

[167] *Id.* at 703-05.

[168] *Id.*

[169] In this regard, the abstraction-filtration-comparison test seems to fail to filter out any source code. This is a notable failure of analysis. The encryption decisions each support the proposition that source code contains ideas, some of which are protectable by the First Amendment.

[170] Finally, the court held that the list of services and the charts were unprotected under the scenes a faire doctrine. Based on the above analysis, the court affirmed the district court's holding that the copyright-protected program had not been infringed. *Id.*

Whelan test and the *Altai* test fail to sufficiently distinguish ideas from expression.[171]

In applying the idea-expression dichotomy explicitly to source code, the court, in *Autoskill, Inc. v. National Educational Support Sys.,*[172] provided copyright holders with a very broad scope of copyright protection for software, thereby diminishing the public domain attributes of source code.[173] The plaintiff owned the copyright in a program called "Autoskill" for testing, diagnosing and training reading skills. The defendant studied the Autoskill program in detail, and then wrote its own computer program for reading skills called "NESS."[174] Each of the programs was based on the identification of three reading sub-types of students.[175] The programs administered tests to determine a student's sub-type by presenting thirteen categories of word form types based upon different combinations of consonants and vowels, ranging from one letter to four letters.[176]

The plaintiff alleged that the defendant's program infringed the copyright in the Autoskill program, and sought a preliminary injunction.[177] The plaintiff did not claim that the defendant's program copied source code, but argued that the defendant's program copied the structure, sequence and organization and "total concept and feel" of the plaintiff's program.

The district court found a number of similarities between the programs that related more to "the important pedagogical aspects of the reading program" than to "the logic flow between the display screens." Based on these findings, the district court concluded that the plaintiff had

---

[171] In *Gates Rubber Co. v. Bando Chemical Industries, Ltd*., the Court of Appeals for the Tenth Circuit also expressly adopted the abstraction-filtration-comparison test as the proper test to apply in computer program copyright infringement cases. *See* Gates Rubber Co. v. Bando Chemical Industries, Ltd. , 9 F3d 823, 834 (10th Cir. 1993).

[172] Autoskill, Inc. v. National Educational Support Sys*.,* 994 F.2d 1476 (10TH Cir.), *cert.denied*, 114 S.Ct. 307 (1993).

[173] *Id*.

[174] Autoskill, Inc. v. National Educational Support Sys., 793 F. Supp. 1557, 1558 (D.N.M. 1992).

[175] *Id.* at 1570.

[176] *Id.* at 1570-71.

[177] *Id.*

established a substantial likelihood of success on its claim of copyright infringement, and issued a preliminary injunction.[178]

On appeal, the Tenth Circuit affirmed.[179] The court noted that the district court had adopted a three-step filtration analysis for judging substantial similarity (as part of the access-plus-substantial-similarity test for proving copying), which was very similar to the Second Circuit's abstraction/filtration/comparison analysis in the *Altai* case.[180]

In the Tenth Circuit's view, a substantial similarity analysis must compare "portions of the alleged infringer's works with the portions of the complaining party's works which are determined to be legally protectable under the Copyright Act."[181] The court noted that the district court had used an abstractions analysis to determine which portions of the plaintiff's works were unprotectable ideas, and which were potentially expression that must be subjected to the filtration analysis.[182]

The Tenth Circuit rejected the defendant's argument that the features of the Autoskill program the plaintiff sought to protect were not protectable.[183] The court did not, however, explain why the functional choice of which sub skills to test for constituted copyrightable expression. As such, the Tenth Circuit's application of its test, like the *Whelan* test, provided copyright holders with a very broad scope of copyright protection for software. Although among courts there may be increasing support for the suggestion that a computer program may contain many "ideas" at many levels of abstraction (or specificity), the various tests used are too divergent conceptually to conclude that courts are beginning to appropriately restrict the scope of copyright protection afforded to computer programs.

Overly broad copyright protection for computer programs may actually hamper advancement in the field of computer programming. In addition, excessive grants of copyright protection are not necessary to promote advancement in the art and science of computer programming. Allowing a programmer to obtain copyright protection for elements of

---

[178] *Id.*
[179] Autoskill, Inc. v. National Educational Support Sys., 994 F.2d 1476 (10TH Cir.), cert.denied, 114 S.Ct. 307 (1993).
[180] *Id.* at 1493-94.
[181] *Id.* at 1492.
[182] *Id.* at 1493-94.
[183] *Id.*

the program available in the public domain and other elements necessitated by industry standards and hardware compatibility could limit both the ability and the incentive for software authors to create or market competitive products.[184]

VI. PART III PROTECTING COMPUTER SOURCE CODE UNDER THE FIRST AMENDMENT

A. *Posting a computer program on a web site may require a government-issued license*.

In November 1996, President Clinton issued an Executive Order and Presidential memorandum transferring regulatory authority over the export of most encryption products from the Department of State to the Department of Commerce, which is now responsible for administering the EAR.[185] In December 1996, the Department of Commerce amended

---

[184] Some commentators have noticed this very effect in the operating system market for personal computers.

[185] 15 C.F.R. Parts 730-774. See Executive Order 13206, 61 Fed. Reg. 58,767 (Nov. 19, 1996); 32 Weekly Comp. Pres. Doc. 2397 (Nov. 15, 1996). The principal purpose of the EAR is to regulate the export of "dual use items -- items that can be used both for military and for civilian purposes. See 15 C.F.R. 6 730.3; see also *Id*. Part 772 (definition of "dual use"). Broadly speaking, and with various exceptions (see *Id*. Part 740), the EAR prohibits the export of dual use items to specified foreign destinations without a license from the Department of Commerce's Bureau of Export Administration (BXA). See *Id*. § 736.2(b)(1).   The core of the EAR's regulatory scheme is the Commerce Control List (CCL). See 15 C.F.R. 774 Supplement No. 1. The Commerce Control List establishes ten general categories of controlled items, such as nuclear materials (Category 0), computers (Category 4), telecommunications and information security items (Category 5), and lasers and sensors (Category 6). Each of these general categories encompasses "commodities," "software," and "technology." A "commodity" is any item other than software or technology. See   15 C.F.R. Part 772 (definition of "commodity"). "Software" is defined in its conventional sense. See ibid. (definitions of "software" and "program"). "Technology" is defined as specific information necessary for the 'development,' 'production,' or 'use' of a product," including technical data (such as   blueprints and diagrams) and technical assistance (such as instruction and consulting services). Every item on the Commerce Control List is assigned an Export Control Classification Number (ECCN). See generally 15 C.F.R. § 738.2(d)(1)-(2). An item's ECCN specifies (among other things) the particular reasons, such as national security or anti-terrorism, why the government controls the export of the item. see *Id*. § 738.2(d)(2)(i). The reasons for control affect the nature and scope of the export controls that apply to the item. see *Id*. §§ 738.4(a), 742.2-742.16. Certain items are not "subject to the EAR," a term of art meaning that they are not within the regulatory jurisdiction of

the EAR to include the encryption items transferred by the President from the regulatory jurisdiction of the Department of State.[186] In his Executive Order and memorandum, the President prescribed the basic policies governing the EAR's export controls on encryption items. The President determined that "[e]ncryption products, when used outside the United States, can jeopardize our foreign policy and national security interests" and can "threaten the safety of U.S. citizens here and abroad."[187] The President therefore directed that applications for licenses to export encryption products be reviewed by the Department of Commerce, in conjunction with other agencies, "to ensure that export would be consistent with U.S. foreign policy and national security interests."[188]

A license is required to export encryption items subject to the EAR to all foreign destinations other than Canada.   Certain encryption items whose export poses fewer risks to national security and foreign policy are eligible for liberalized licensing requirements.[189]

---

the BXA and may be exported without regard to the EAR's export controls, even if the Commerce Control List otherwise would cover them. See   15 C.F.R. §§ 734.1(a), 734.2(a), 734.3(b). Among other things, printed materials such as newspapers, books, and periodicals are not subject to the EAR, regardless of their subject matter or contents. See *Id*. § 734.3(b)(2). In addition, "publicly available" software and technology generally are not subject to the EAR. *Id.* § 734.3(b)(3), 734.7-734.10.

[186] See   61 Fed. Reg.68,572 (Dec.30, 1996); see also 63 Fed. Reg;.72,156 (Dec.31,1998) (amendments). As amended, the Commerce Control List includes encryption commodities (e.g., circuitry and other hardware), encryption software, and encryption technology. See 15 C.F.R. Part 774 Supplement No. 1 (CCL), ECCN 5A002 (commodities), 5D002 (software), 5E002 (technology). The regulatory scheme under the EAR creates two important classifications: Encryption Products, which are subject to the EAR, and Cryptographic Information, which are not subject to the EAR. In this regard, the EAR imposes no licensing requirement or other limitation on the export of Cryptographic Information. § 734.3(b)(3). This is an important distinction because it forms part of the basis for why the Federal government argues that its regulations are content neutral. Nonetheless, encryption source code, by regulatory definition, cannot be considered Cryptographic Information and, in this regard, although a given encryption software product may not require a license, it remains subject to the EAR. § 742.15.

[187] 32 Weekly Comp. Pres. Doc. 2397; see 15 C.F.R. § 742.15.
[188] 32   Weekly Comp. Pres. Doc. 2398. See also Exec. Order No. 13,026, 61 Fed. Reg. 58,767 (1996).
[189] See 15 C.F.R. §§ 736.1(b)(1), 742.15(a). 740.8, 740.17, 742.15(b)(2)-(7).

Peter Junger, a law professor, who teaches a course titled "Computers and the Law" at Case Western Reserve University Law School in Cleveland, Ohio,[190] maintains a website on the World Wide Web that contains information about courses that he teaches, including a computers and law course.[191] Professor Junger wanted to post to his web site various encryption programs that he has written to show how computers work. The U.S. Department of Commerce determined that such a posting is an export[192] under the Export Administration Regulations ("EAR").[193]

On June 12, 1997, Professor Junger submitted three applications to the Commerce Department requesting determination of commodity classifications for encryption software programs and other items. With

---

[190] Junger v. Daly, 8 F. Supp. 2d. 708 (N.D. Ohio 1998). On April 4, 2000, the Sixth Circuit ruled in Junger's favor on the First Amendment question by holding that source code is an articfact of expression entitled to First Amendment protection. The court remanded the case to the district court for further proceedings on the revised regulations issued y BXA subsequent to the district court's initial ruling. _ F3d _ (April 4, 2000) (2000 FED App. 0117P (6th Cir.)). While it is possible that the district court's analysis and fact-finding may result in the Federal government ultimately prevailing, the Circuit Court opinion follows the trend characterized in this article.

[191] During the litigation Professor Junger has continued to teach the computer law course, but, due to the government's position on the use and export of his cryptographic software program, Professor Junger has not used his website as a resource for the course and during the Fall 1999 semester "for the first time" has not used his own material as the casebook. See Rod Dixon and Peter Junger, Electronic Mail Correspondence, September 19 - 20, 1999 (on file with author).

[192] The EAR defines an export as "an actual shipment or transmission of items subject to the EAR out of the United States, or release of technology or software subject to the EAR to a foreign national in the United States." 15 C.F.R. § 734.2(b)(1) (1998). With regard to the exportation of encryption source code, the word "exporting" includes downloading, or causing the downloading of, such software to locations (including electronic bulletin boards, Internet file transfer protocol, and World Wide Web sites) outside the U.S., or making such software available for transfer outside the United States, over wire, cable, radio, electromagnetic, photo-optical, photoelectric or other comparable communications facilities accessible to persons outside the United States, including transfers from electronic bulletin boards, Internet file transfer protocol and World Wide Web sites. 15 C.F.R. § 734.2(b)(9)(ii) (1998).

[193] See 15 C.F.R. § 734.2(b)(9). See also Arms Export Control Act of 1976, Pub. L. No. 94-329, title II, 212(a)(1), 90 Stat. 744 (1976) (codified at 22 U.S.C. 2778 (1994)). Since source code posted on a website can be accessed from any country in the world, posting software on a website (or, possibly, even listing software as a hypertext link on a webpage) may constitute an export of encryption that requires a license.

these applications, Professor Junger sought a Commerce Department determination whether the agency restricted the materials from export.[194]

On July 4, 1997, the Commerce Department's Bureau of Export Administration told Professor Junger that Export Classification Control Number 5D002 covered four of the five software programs he had submitted, therefore making them subject to the EAR.[195] The Commerce Department found that the first chapter of Professor Junger 's textbook, Computers and the Law, was an allowed unlicensed export,[196] but that export of his software programs would require a license. After receiving the classification determination, Professor Junger did not apply for a license to export his classified encryption software.[197]

In filing his lawsuit, Professor Junger claimed the EAR[198] violate rights protected by the First Amendment.[199] According to Professor Junger, the Export Regulations engage in unconstitutional content

---

[194] *Id.* at 712.

[195] *Id.* at 712 - 714.

[196] *Id.* at 712 - 714.

[197] *Id.* While the mere posting of a program to a file transfer protocol (FTP) or web server located in the United States constitutes an export according to the definition, the mere posting of such a program does not by itself result in any sending, taking, disclosure or transmission of the program to a foreign person. Though the definition classifies the mere posting of encryption as an export, it does include a caveat that permits encryption products to be placed on Internet sites within the United States as long as the provider implements safeguards that are adequate to prevent unauthorized transfer of such code outside the United States. The required precautions include making sure that access to and transfer of the software is controlled through such measures as: (1) checking the address of every system attempting to obtain the software to make sure that the system is located within the United States; (2) providing a requesting party with notice that the transfer of the software is subject to export controls and that it cannot be exported without a license; and (3) requiring every party requesting a transfer to acknowledge that they understand that the software is subject to export controls. 15 C.F.R. § 734.2(b)(9)(ii)(A)(1-3) (1998).

[198] Essentially, the EAR establishes a voluntary procedure for submitting dual-use export classification requests; establishes a mandatory procedure for export licensing applications; establishes categories of subject matter that is subject to encryption export controls; and authorizes the Department of Commerce's Bureau of Export Administration (or BXA) to decline issuance of export licenses. The BXA's final determination may be challenged by filing a notice with the Office of the President.

[199] Chaplinsky v. State of New Hampshire, 315 U.S. 568, 572 (1942).

discrimination by subjecting certain types of encryption software to more stringent export regulations than other types of software.[200]

In addressing these claims, the court recognized that it needed to decide whether encryption source code is sufficiently expressive to merit heightened First Amendment protection.[201] In doing so, the court examined whether the Export Regulations are a prior restraint on speech subject to greater First Amendment scrutiny. The court determined that the Export Regulations are constitutional because encryption source code is inherently functional, because the Export Regulations are not directed at source code's expressive elements,[202] and because the Export Regulations do not reach academic discussions of software, or software in print form.[203]

The *Junger* court concluded that in the overwhelming majority of circumstances, encryption source code is exported to transfer functions, not to communicate ideas.[204] In exporting functioning capability, encryption source code is like other encryption devices. In the court's view, for the broad majority of persons receiving such source code, the value comes from the function the source code does. The court criticized the district court's analysis in *Bernstein* by noting that the

---

[200] *Id.* at 712. See, Mary M. Cheh, *Government Control of Private Ideas--Striking a Balance Between Scientific Freedom and National Security*, 23 JURIMETRICS J. 1, 22 (1982) (arguing that cryptographic information is protected by the First Amendment); James R. Ferguson, *Scientific Inquiry and the First Amendment*, 64 CORNELL L. REV. 639, 654-56 (1979) (arguing that scientific inquiry merits some degree of protection by the First Amendment).

[201] Without regard to a principled distinction, the Court suggested that some software is inherently expressive, while other software is "inherently functional." According to the Court, an indication of "inherently functional" software is software that users look to the performance of tasks with scant concern for the methods employed or the software language used to control such methods. *Id.*

[202] It is beyond dispute that prior restraints on expressive materials bear a heavy presumption against their constitutional validity, and are subject to the strictest judicial scrutiny. See New York Times Co. v. United States, 403 U.S. 713, 714 (1971). If a law distinguishes among types of speech based on their content of ideas, the Court reviews it under strict scrutiny. To survive strict scrutiny, the government must employ narrowly tailored means that are necessary to advance a compelling government interest. See Turner Broadcasting System, Inc. v. FCC, 512 U.S. 622, 642 (1994). However, if a law does not distinguish among types of speech based upon the content of the speech, it will not be subject to strict scrutiny.

[203] The Court granted the government's motion for summary judgment.

[204] *Id.* at 721.

Bernstein court's interpretation of *Texas v. Johnson*[205] was misguided. *Johnson* does not "strongly imply" that the First Amendment extends to anything written in language regardless of its expressiveness.[206] Rather, in the *Junger* court's view, it simply observes that the First Amendment's "protection does not end at the spoken or written word."[207]

Finally, the *Junger* court determined that the district court in *Bernstein* misunderstood the significance of source code's functionality. As *Junger* explained, source code is "purely functional" in a way that the *Bernstein* court's examples of instructions, manuals, and recipes are not.[208]

In this regard, *Junger* upheld the government's position that exporting source code is conduct that can occasionally have communicative elements, but that First Amendment protection is not necessarily warranted merely because conduct is occasionally expressive.[209] In other words, the court felt it evident that exporting encryption source code is not sufficiently communicative.[210] Apparently, the court was unconvinced that the licensing scheme used by the EAR had a close enough nexus to speech or that the export of encryption source code was conduct sufficiently associated with expression to pose an unconstitutional threat of censorship.

Notably, *Junger* did not specifically disregard the practical reality that source code may be used as speech or to speak. Rather, the court determined that notwithstanding the expressive elements of some forms of source code, whether a regulation that burdens a programmer's use of source code is subject to the First Amendment's highest level of

---

[205] 491 U.S. 397, 404 (1989),

[206] Bernstein I, 922 F. Supp. at 1434. In a preliminary ruling, Judge Marilyn Patel denied the government's motion to dismiss the case, and held that cryptographic computer source code is speech. Thus, Bernstein had asserted a "colorable" claim to First Amendment protection. *Id.* Accordingly, Judge Patel became the first to explicitly recognize in a judicial opinion that computer programmers maintained a protected speech interest in their computer source code.

[207] Johnson, 491 U.S. at 404. In addition, the Court observed that Bernstein's assertion that "language equals protected speech" is unsound. Junger at 716. "Speech" is not protected simply because we write it in a language. Instead, what determines whether the First Amendment protects something is whether it expresses ideas. *See* Roth v. United States, 354 U.S. at 484; Virginia Citizens Consumer Counsel, 425 U.S. at 762.

[208] Junger at 1111.

[209] *Id*.

[210] *See Id*. at 1113.

scrutiny depends upon more than summarily ascribing the qualities of a human language to a programmer's notations. Since source code is a set of instructions to a computer that is commonly distributed for the primary non-expressive purpose of controlling a computer's operation, *Junger* determined that the prior restraint doctrine is not implicated. In the court's view, simply because an activity may have an expressive element is not sufficient to transform an otherwise functional activity into something of constitutional significance. As such, the Court rejected Professor Junger's facial challenge to the Export Regulations' licensing scheme. Professor Junger's overbreadth challenge was rejected as well.[211]

B. *Using an encryption program to teach on the Net.*

In *Daniel J. Bernstein v. United States Department of State*, a former graduate student claimed that the government violated his right to free speech by restricting his right to post an encryption program on the Internet.[212] The district court ruled that encryption source code specifically, and computer programming languages generally, constitute speech.[213] As such, according to the court, the First Amendment protects communications in computer languages, like communications in other

---

[211] The overbreadth rule arises from the purpose of the doctrine. The overbreadth doctrine allows a challenge to laws having the potential to repeatedly chill the exercise of expressive activity by many individuals. To make the overbreadth challenge, there must be a realistic danger that the statute will significantly compromise recognized First Amendment protections of parties not before the Court. Members of City Council of City of Los Angeles v. Taxpayers for Vincent, 466 U.S. 789 (1984). Under Vincent, to prevail on a facial overbreadth challenge, the plaintiff must show that the challenged law is "substantially overbroad." *Id.* at 801. To establish substantial overbreadth, a plaintiff must show that the law will have a significant and different impact on third parties' free speech interests than it has on his own. See *Id*.

Professor Junger's overbreadth challenge failed because he did not show that the Export Regulations injure parties not before the Court in a manner different from the way they affect Professor Junger. The Court also determined, without helpful explanation, that the Export Regulations are not vague.

[212] 945 F. Supp. at 1289-90.
[213] In Bernstein v. United States Department of State, the District Court for the Northern District of California ruled that licensing requirements for the export of cryptographic software under the International Traffic in Arms Regulations ("ITAR") an unconstitutional prior restraint of protected speech. 945 F. Supp. 1279 (N.D. Cal. 1996), enforcing 922 F. Supp. 1426 (N.D. Cal. 1996) (ruling that computer code was protected speech under the First Amendment).

forms of language.[214] In her ruling, Judge Patel pointed out that the court could, "find no meaningful difference between computer languages...and German or French. All participate in a complex system of understood meanings within specific communities."[215]

The court began by stating what its view of source code is; namely, that source code is the text of a program written in a "high-level" programming language, such as "PASCAL" or "C."[216] A critical factual determination by the panel decision included that a distinguishing feature of source code is that it is "meant to be read and understood by humans and that it can be used to express an idea or a method."[217]

The court recognized that since source code is destined for the "maw of an automated, ruthlessly literal translator" -- the compiler -- a programmer must follow stringent grammatical, syntactical, formatting, and punctuation conventions. As a result, only those trained in programming can easily understand source code.[218] For example, the following is an excerpt from Bernstein's Snuffle source code written in the programming language C:[219]

---

[214] Bernstein had also submitted commodity jurisdiction requests for several written texts that contained the Snuffle algorithm and description. Initially, the State Department denied him permission to distribute the texts, but retracted this decision after Bernstein filed suit. Bernstein v. Dep't of State, 922 F. Supp. at 1433-34.

[215] A similar analysis is undertaken under the law of copyright with respect to examining the limitations of copyright in the First Amendment context. There, the legal regime of copyright has come to rely upon the so-called idea/expression dichotomy, wherein ideas are associated and fixated as protectable artifacts of the First Amendment while expressions are protectable artifacts of the law of copyright. In this respect, the law of copyright yields to the First Amendment by excluding from copyrightable "expression" abstract ideas. The empowerment of a tautology is at work here. Courts use as definitional line drawing between objects that are in many, if not most, respects indistinguishable.

[216] 945 F. Supp. at 1289.

[217] 945 F. Supp. at 1289-90.

[218] Interestingly enough, although the rigid grammar of software programming renders the code unintelligible to "outsiders," the strict structure of programming language similarly limits creative expression, not in style, but in fact. In other words, since there are so few ways to write an "If…then" decision structure, the level originality cannot possibly represent copyrightable expression. *Id.* at 1296.

[219] Interestingly enough, the snuffle program seems to rely upon a known Hash algorithm, despite the Ninth Circuit's conclusion about the distinct use of encryption source code to publicly expose untested algorithms. The court may have been responding to arguments of counsel rather than making its own determination or accepting the

```
for (; ;)
(
uch = gtchr();
if (!(n & 31))
(
for (i = 0; i64; i++)
l [ ctr[i] = k[i] + h[n - 64 + i]
Hash512 (wm, wl, level, 8);
)
```

The court also noted that since the chief task for cryptographers is the development of secure methods of encryption, the expression of algorithmic ideas with precision and methodological rigor in source code has the added benefit of facilitating peer review. In this regard, by compiling the source code, a cryptographer can create a working model subject to rigorous security tests.[220] According to the court, cryptographers use source code to express their scientific ideas in much the same way that mathematicians use equations or economists use graphs. Of course, both mathematical equations and graphs are used in other fields for many purposes, not all of which are expressive.[221] But mathematicians and economists have adopted these modes of expression in order to facilitate the precise and rigorous expression of complex scientific ideas.[222]

The court also determined that Snuffle was intended, in part, as political expression.[223] Bernstein discovered that the ITAR regulations

---

district court's finding about what the snuffle program actually does. An encryption software application can initialize a cryptographic algorithm contained in another source like an API (application programming interface) in an operating system, a programming environment – like JAVA, or another application – like Cyberspace-based SSL encryption transactions or specify – through programming notation – its own unique algorithm. *See, e.g.*, Jonathan Knudsen, JAVA CRYPTOGRAPHY at 5-7 (O'Reilly 1998). Rather astonishingly, under the Federal government's encryption regulations, the Ninth Circuit's panel decision, and the district court's analysis, whether the source code, itself, contains a cryptographic algorithm or merely makes a call to an "external" cryptographic software library is apparently a relevant factor under both the licensing scheme and the assessment of the expressive qualities of the source code. Only in the Karn case was source code, itself, containing a unique cryptographic algorithm at issue.

[220] 922 F. Supp. at 1435.

[221] *See, e.g.,* Simon Singh, THE CODE BOOK.

[222] 922 F. Supp. at 1435.

[223] 922 F. Supp. at 1435 – 1442.

controlled encryption exports, but not one-way hash functions.[224] Since Bernstein believed that an encryption system could easily be fashioned from any of a number of publicly available one-way hash functions, he viewed the distinction made by the ITAR regulations as absurd.[225]

The panel decision flatly rejected what it termed the government's argument "distilled to its essence" -- that even one drop of "direct functionality" overwhelms any constitutional protections that expression might otherwise enjoy.[226] In the court's view, the government's argument proved too much in the era of rapidly evolving computer capabilities, wherein computers will soon be able to respond directly to spoken commands.[227] In this regard, the court noted that to confer upon the government the unfettered power to impose prior restraints on speech in an effort to control the "functional" aspects of a communications technology.[228]

In the Ninth Circuit's view, the challenged regulations allowed the government to restrain speech indefinitely with no clear criteria for review.[229] As a result, Bernstein and other scientists were effectively chilled from engaging in valuable scientific expression.[230] Bernstein's experience, itself, demonstrates the enormous uncertainty that exists over the scope of the regulations and their chilling potential.[231]

The court's holding was narrow; it did not hold that all software is expressive, but recognized that much software is not.[232] Nor did the court assess whether the challenged regulations constitute content-based restrictions, subject to the strictest constitutional scrutiny, or whether they are, instead, content-neutral restrictions meriting less exacting

---

[224] See International Traffic in Arms Regulations, 22 C.F.R. § 120.4(a) (specifying a procedure for determining if an article is covered under the Munition List).

[225] Interestingly enough, Bernstein's argument lays bare the analytic similarity between the two questions: whether source code is speech and whether source code is copyrightable. Generally, an affirmative answer to the former question requires a negative answer to the latter. Encryption algorithms that are publicly available -- and, therefore, not trade secrets or national secrets -- are scenes a faire. Source code constituting artifacts of the public domain should not be subject to copyright protection.

[226] 922 F. Supp. at 1435 – 1442.

[227] *Id*.

[228] *Id*.

[229] 922 F. Supp. at 1435 – 1442.

[230] *Id.*

[231] *Id.* at 1441-42.

[232] *Id.* at 1442.

scrutiny.[233] Instead, the holding was limited to the determination that because the prepublication licensing regime applies directly to scientific expression, vests boundless discretion in government officials, and lacks adequate procedural safeguards, it constitutes an impermissible prior restraint on speech.[234]

Despite the professed narrowness of its holding, the Ninth Circuit opined that insofar as the EAR were intended to slow the spread of secure encryption methods to foreign nations, the government's policy was intentionally retarding the progress of the "flourishing science of cryptography." Additionally, the court found that the EAR had not sufficiently countenanced the growing importance of the need to protect personal privacy in the digital age. In the court's opinion, the pervasive use of digital technology has resulted in a dramatic diminution of our ability to communicate privately.[235] However, in view of the court's dicta, it is difficult to estimate how much the privacy considerations really played a role in the court's rejection of the regulations. [236]

One troubling aspect of the *Bernstein* analysis is that the court erroneously focuses too narrowly on the putatively communicative nature of source code. Generally, the First Amendment protection of speech attaches to activities where the free exchange of information or content is principally at issue, but *Bernstein* stretches its analysis of the communicative qualities of source code by distorting the task of the computer programmer.[237] In the court's view, programmers do not just program technological devices; rather, they write code that expresses ideas about technology.[238] In this light, one could hardly logically

---

[233] *Id.* at 1441-42.

[234] *Id.*

[235] The court noted that Cellular phones are subject to monitoring, email is easily intercepted, and transactions over the Internet are often less than secure.

[236] According to the Court, the government's efforts to retard progress in cryptography may implicate the Fourth Amendment, as well as the right to speak anonymously, *see* McIntyre v. Ohio Elections Comm'n, 514 U.S. 334, (1995), the right against compelled speech, *see* Wooley v. Maynard, 430 U.S. 705, 714 (1977), and the right to informational privacy, *see* Whalen v. Roe, 429 U.S. 589, 599-600 (1977).

[237] *See* Martin Redish, *The Value of Free Speech*, 130 U. PA. L. REV. 591, 628 (1982).

[238] Although programmers occasionally can be found describing an individual's source code (or more likely there own code) as elegant. This reference does not describe stylistic splendor or refined and graceful code that communicates in clearly expressed language. Instead, elegant code refers to the scientific precision and simplicity of source code. In the world of computer science, succinctness, brevity and precision in source code are valued highly. Precise source code usually yields robust, well performing software

disagree that the First Amendment should protect the programmer's task.[239]   Even so, attributes of source code are analytically distinct from the programmer's task, and the Court's analysis may have fallen short of recognizing the distinction.

C. *Computer diskettes are munitions.*

The *Karn* case arose out of the Federal government's designation of Philip Karn's computer diskette as a "defense article" pursuant to the Arms Export Control Act (AECA), *22 U.S.C. § § 2751*-2796d, and the International Traffic in Arms Regulations (ITAR), 22 C.F.R. § § 120-130.[240]

Karn alleged that the government's designation of a diskette containing source codes for cryptographic algorithms as a defense article subject to the export controls set forth in the ITAR, when the defendant deemed a book containing the same source codes not subject to said export controls, is arbitrary and capricious and an abuse of discretion in violation of the Administrative Procedure Act (APA), *5 U.S.C. § 706*(2)(a).[241] Karn also alleged that the regulation of the diskette violated his First Amendment right to freedom of speech and arbitrarily treated the diskette differently than the book in violation of the plaintiff's Fifth Amendment right to substantive due process.[242]

On February 12, 1994, Karn submitted to the Department of State a commodity jurisdiction request for the book Applied

---

applications that function efficiently while using minimal resources of the microprocessor and few amounts of random access memory.

[239] Wooley v. Maynard, 430 U.S. 705, 714 (1977) ("[T]he right of freedom of thought protected by the First Amendment against state action includes both the right to speak freely and the right to refrain from speaking at all.").

[240] Philip Karn filed suit challenging the State Department's denial of permission to export a diskette containing the source code for several encryption algorithms printed in the book Applied Cryptography by Bruce Schneier. *See* Bruce Schneier, Applied Cryptography (2d ed. 1996).   The State Department approved the export of the book itself, but not the diskette containing identical information. Karn sought review in federal district court of the government's denial claiming that the Arms Export Control Act (AECA) and the International Traffic in Arms Regulations (ITAR) were unconstitutional under the First and Fifth Amendments. 925 F. Supp. 1 (D.D.C. 1996), *remanded*, 107 F.3d 923 (D.C. Cir. 1997).

[241] 925 F. Supp. 1-7.

[242] 925 F. Supp. 1-7.

Cryptography, by Bruce Schneier.[243] The book, Applied Cryptography, provides, among other things, information on cryptographic protocols, cryptographic techniques, cryptographic algorithms, the history of cryptography, and the politics of cryptography. Part Five of Applied Cryptography contains source code for a number of cryptographic algorithms.[244]

On March 9, 1994, Karn submitted a commodity jurisdiction request for a diskette containing the source code printed in Part Five of the book, Applied Cryptography.[245] The request stated that "the diskette contains source code for encryption software that provides data confidentiality" and that "the software on this diskette is provided for those who wish to incorporate encryption into their applications."[246]

Karn contended that pursuant to sections 125.1 and 120.11, the diskette is in the "public domain" and therefore is not subject to the ITAR. The government contended that the diskette does not fall within the "public domain" exemption because said exemption only applies to "technical data" which, according to the defendants, does not include cryptographic software.[247] In this regard, the government's position was consistent with the positions in *Junger* and *Bernstein*. The Court determined that Karn's interpretation of the regulations at issue was "strained and unreasonable."[248]

In the Court's view, it is far more reasonable to read § 2778(a)(1) and (h) to preclude judicial review for the designation of items as defense articles pursuant to the language of the munitions list and the procedures provided for interpreting the list, all set forth in the ITAR -- in other words, if the defendants follow the procedures set forth in the ITAR and authorized by the AECA for designating an item as a defense article, such item is a part of the munitions list. As such, the Court concluded that the legislative scheme supported its view that

---

[243] 925 F. Supp. 1-7.
[244] See 22 C.F.R. § § 120.4 and 121.1, category XIII(b)(1), Note.

[245] 925 F. Supp. 1-3.
[246] 925 F. Supp. 1-3.
[247] 22 C.F.R. § § 120.10(a)(4), n10 120.11 and 121.8(f). (a) Public domain is a term of art under the ITAR that means information which is published and which is generally accessible or available to the public. 10 22 C.F.R. § 120.10.
[248] 925 F. Supp. 1-3.

Congress had precluded judicial review of the commodity jurisdiction procedure.

According to the Court, judicial non-reviewability of the defendants' commodity control decision was consistent with the structure of the United State's export control scheme.[249]

Likewise, the Court found the regulation of Karn's diskette as cryptographic software is rational, even when considered in conjunction with the government's decision not to subject the book Applied Cryptography to the ITAR. Karn conceded that using the source code in Part Five of Applied Cryptography to encode material takes greater effort and time than using the Karn diskette. In this respect, the Court concluded that treating the book and diskette differently is not in violation of the plaintiff's substantive due process rights.[250]

Despite their inconsistent outcomes, the encryption cases do agree on a few significant and relevant concerns. They all agree that books, academic writings, and papers can be exported without a government license, because they are "protected speech" under the First Amendment. *Karn* and *Bernstein* agreed that although the AECA and the EAA/ EAR bar judicial review of the designation of encryption items as defense items, they do not bar constitutional claims regarding the regulations themselves. More important, all three cases have maintained the view that source code contains some expressive qualities that warrant government restrictions affecting source code being subject to intermediate level scrutiny by courts.

Most notably, the courts disagreed over the outcome of whether the government regulations constituted valid restrictions upon the plaintiffs' First Amendment rights and whether the exports of encryption were expressive or functional. *Karn* found that the ITAR regulations

---

[249] At this stage in the proceedings, Karn was not challenging the licensing scheme. Instead, Kan challenged the discretionary classification scheme. One rationale for the Court's reliance on this general rule is that such "facial" challenges typically raise "a discrete issue, unrelated to the facts of the case, that only needs to be resolved once," and therefore, entertaining the challenge does not "open the floodgates to litigation." *Id.*; *see also*, Bowen v. Michigan Acad. of Family Phys., 476 U.S. 667 at 677, 680 n.11, (1986); Johnson v. Robison, 415 U.S. 361, 370 (1974). As such, the Court may have rightly rejected Karn's challenge.

[250] The Court dismissed Karn's APA claim, and ruled that the government was entitled to summary judgment on the First and Fifth Amendment claims. Karn subsequently challenged the licensing scheme under the EAR.

were content neutral and justifiable restrictions under the O'Brien test and that it did not have to decide whether the regulations constituted a prior restraint on the First Amendment. Likewise, *Junger* found that encryption exports were only occasionally expressive and therefore not protected by the First Amendment and also that the regulations themselves were content neutral and satisfied the O'Brien test. *Bernstein* however, decided that even if the regulations were content neutral, the encryption system was pure speech that could not be limited by a prior restraint without establishing sufficient safeguards.[251]

Although the encryption cases leave important questions unresolved, they support the narrowing of the scope of copyright protection for computer source code. The cases reinforce this conclusion at a time when it is becoming more apparent than ever that the idea/expression dichotomy of copyright law has failed to provide adequate protection to free expression and the public domain in the context of digital technologies like computer software. The idea/expression dichotomy neither serves as a proper limiting doctrine for source code nor fulfills the overriding objective of copyright to promote broad public availability of computer software by encouraging and rewarding creativity and innovation.[252]

Despite its deficiencies, the idea-expression distinction has been particularly popular as a judicial tool for alleviating perceived tension between first amendment and copyright interests. As noted *supra*, many courts have used the distinction to deny first amendment claims by simply stating the proposition that ideas are not copyrightable and summarily dismissing the apparent conflict without elaboration.

---

[251] *See also* Laura M. Pilkington, Comment, *First and Fifth Amendment Challenges to Export Controls on Encryption: Bernstein and Karn*, 37 SANTA CLARA L.REV. 159 (1996).

[252] There may be no better example of how copyright protection of source code has stood on its head the objectives of copyright to encourage the dissemination of creative and useful works than the manner in which Microsoft Corporation uses its copyright in the source code of its operating systems software to forestall or completely eliminate opportunities for others to create useful works in the marketplace of personal computers. *See, e.g.*, A guide to Act II in the Antitrust Trial, U.S. NEWS & WORLD REPORT, December 15, 1999 at 50 (noting that the district judge's over 400 findings of fact issued in the Federal government's antitrust litigation against Microsoft supported the court's suggestion that the software developer used its government granted monopoly over its source code to erect an impenetrable barrier to others who wished to create similar works).

Nonetheless, judicial analysis has often failed to acknowledge that idea and expression often merge, becoming virtually indistinguishable. This problem is particularly acute in the area of graphic works, where the visual impact of a photograph, for example, may be inseparable from the idea.

More important, both case and comment have ignored the situation in which no degree of creativity or effort can substitute for the duplication of the particular expression of another. Of course, the first amendment has not been construed as absolute. However, the idea of a government sanctioning burdensome licensing regime on free speech is repugnant to the very notions of lively and robust debate, which underlie the first amendment. Information is not a commodity for ransom, but a resource for societal progress and personal edification. Viewed in this light, compulsory licensing is objectionable when the full weight of the licensing scheme is directed toward an artifact of free speech, and has the effect of becoming so onerous that it prevents a speaker from communicating with an audience.

Courts can no longer rely on a variety of flawed exceptions to copyright law to ensure the free flow of information in today's technologically-oriented world.[253] By granting authors the exclusive right to reproduce and distribute their original expression, the Copyright Act allows some authors and copyright holders to use copyright as a means to suppress facts as well as expression.[254] The limiting doctrines of copyright should not be distorted to permit government-issued monopolies on what is supposed to be original expression to implacably continue to define out of or remove from the public domain and

---

[253] See Laurence H. Tribe, American Constitutional Law § 12-7 825-832 (2d ed. 1988) (noting that the distinction arises from labor picketing cases such as Thornhill v. Alabama, 310 U.S. 88 (1940) and arguing that the dichotomy is too oversimplified to be applied consistently or to have much determinate content); Cass R. Sunstein, *Words, Conduct, Caste*, 60 U. Chi. L. Rev. 795 (1993) (arguing that regulation of speech should be evaluated against goals of fostering democracy and equality, not on the speech/conduct distinction); *see also* Stephanie M. Kaufman, *The Speech/Conduct Distinction and First Amendment Protection of Begging in Subways*, 79 Geo. L.J. 1803 (1990).

[254] *See, e.g.*, Harper & Row Publishers, Inc. v. Nation Enters., 471 U.S. 539, 558-59 (1985) (public interest in the content of copyrighted subject matter does not necessarily override the author's right to first publication or other exclusive rights granted to copyright owners by the Copyright Act).

marketplace of ideas common methods of expressing computer instructions in source code.

VII.   CONCLUSION

A. *Courts should be freed from reliance on incoherent distinctions between copyrightable and uncopyrightable aspects of computer programs.*

Courts apply the so-called "idea/expression dichotomy" to limit the expansion of copyright from crossing into the province of the First Amendment. In doing so, courts have used the test to deny awarding damages in copyright infringement actions when the elements of the disputed work constitute ideas, not expression. In applying the dichotomy, courts have often struggled to draw principled distinctions between copyrightable expression and the so-called basic ideas underlying copyrightable expression, particularly when the "expression" subject to a court's analysis is contained in computer programs. The analytic limitations of the idea/expression dichotomy are considerable and courts frequently misunderstand how to apply its various manifestations.

Whatever the reasons for the prior reliance on the dichotomy, the encryption decisions are compelling indications that the time has come for courts to put aside and replace the amorphous and ineffective dichotomy; this is particularly true in the context of copyright infringement actions involving computer source code.

B. *Open source code is an artifact of the public domain*

The open source code movement represents a global movement that is successfully challenging the contemporary proprietary model of software development with a model in which "openness" is considered a virtue in software development.

Open source code programming both may produce superior products (as a result of input from potentially hundreds of programmers) as well as reinforce market competition by precluding "lock in" to proprietary technology. In this respect, open source programming eschews the development of proprietary source code products in favor of software products that contain freely available source code. Software

development is becoming, largely, an open and shared process in Cyberspace.

This new programming paradigm acknowledges that: good programmers know what to write, and great programmers know what to rewrite (or reuse). Open source code programmers are more likely to efficiently and openly reuse code than traditional programmers, not simply because they are always guaranteed access to the entire source code, but also because they need not waste resources keeping their code secret either to protect an intellectual property interest or, worse, to avoid apparent notice that their software creation efforts violate the intellectual property interests of others. As such, source code is suitably recognized as an artifact of the public domain.

C. *Source code rarely should be regarded as a category of expression created as a result of independent, and hence, original authorship*

The digital age has brought along a notably critical challenge for copyright: how to continue the vitality of copyright protection in an environment where violations of copyright are not only rampant and disgorging, but also undermining of the very basis of the copyright regime. To date, the short answer to this perplexing question has been the support of an implacably expanding reach of copyright.[255] Like a nine-headed Hydra, the reach of copyright is growing in parallel to the presumed threat that digital technologies and Cyberspace seem to place upon the legal regime. Since digital works must be copied to be used, these technologies will inevitably require courts and Congress to confront the conflict between copyright and the First Amendment in a straightforward manner. To date, no clearer example of this confrontation has arisen than in the context of the encryption debates.

Viewing computer source code as an artifact of the public domain suitably reinforces an important goal of copyright; namely, that the government grant copyrights in works to meaningfully motivate the creative activity of authors in a manner that ultimately ensures public

---

[255] The Clinton administration is viewed by some as attempting to circumvent significant Congressional scrutiny by obtaining increased copyright protection through new international copyright treaties. *See* Stephen Fraser, *The Copyright Battle: Emerging International Rules and Roadblocks on the Global Information Infrastructure*, 15 J. Marshall J. of Computer & Info. L. 759 (1997).

access to the products of an author's creativity. In this regard, copyright law should permit the unfettered access to public domain material by protecting source code authors from copyright infringement when the elements of a work at issue in an infringement action are the artifacts of the public domain. Thus, courts adjudicating copyright infringement actions involving computer software should undertake a thoroughgoing reassessment of the limiting principles of copyright law, recalibrate the boundaries and the scope of copyright protection for software, and rarely regard source code as a category of expression created as a result of independent, and hence, original authorship.