

THE LAW & ECONOMICS OF REVERSE ENGINEERING

by
Pamela Samuelson* & Suzanne Scotchmer**

- I. Introduction
- II. Reverse Engineering in Traditional Manufacturing Industries
 - A. A Legal Perspective on Reverse Engineering
 - B. An Economic Perspective on Reverse Engineering
 - C. Anti-patent Laws: An Exception to Reverse Engineering Rules?
- III. Reverse Engineering In the Semiconductor Industry
 - A. Perturbations in Product Life Cycles in the Chip Industry
 - B. Copyright or Sui Generis Protection for Chip Designs?
 - C. Economic Implications of the SCPA Rules
 - D. Post-SCPA Developments
- IV. Reverse Engineering in the Computer Software Industry
 - A. Reverse Engineering of Software And Copyright Law
 - B. The Economics of Interoperability and of Reverse Engineering
 - C. Reverse Engineering of Software And Patent Law
 - D. Reverse Engineering of Software And Contract Law
- V. Reverse Engineering of Technically Protected Digital Content
 - A. Legislative History of the DMCA Anti-Circumvention Rules
 - B. An Economic Rationale for the DMCA Rules
 - C. Collateral Damage of Overbroad DMCA Rules
 - D. Alternatives to the DMCA Anti-Circumvention Rules
- VI. Reverse Engineering as a Policy Lever
 - A. Ways to Regulate Reverse Engineering
 - 1. Regulating a Market-Destructive Means of Reverse Engineering
 - 2. A Breadth Requirement For Products of Reverse Engineering
 - 3. Purpose- And Necessity-Based Criteria for Determining the Legitimacy of Reverse Engineering

* Professor of Law and Information Management, University of California at Berkeley. I wish to thank Kirsten Roe, Eddan Katz, and Christine Duh for their excellent research assistance in connection with this article. Research support for this paper was provided by NSF Grant. No. SES 9979852. We are both grateful for insightful comments on an earlier draft by Rochelle Cooper Dreyfuss, Neil Gandal, Robert J. Glushko, Wendy J. Gordon, Mark A. Lemley, Ejan MacKaay, David McGowan, Michael Moradzadeh, Maureen O'Rourke, Eva Oglieska, J.H. Reichman, Hal Varian, Fred von Lohmann, and participants in the Boston University Intellectual Property Workshop Series and the University of Washington Law School's Innovation Workshop.

** Professor of Economics and Public Policy, University of California at Berkeley. Research support for this paper was provided by NSF Grant No. 98 18689.

4. Regulating Reverse Engineering Tools
5. Restricting Publication of Information Discovered By a Reverse Engineer

B. Policy Options When Innovators Try to Bypass Reverse Engineering

1. Avoiding the Threat of Reverse Engineering By Contract
2. Avoiding the Threat of Reverse Engineering by Technical Obfuscation

VII. Conclusion

I. Introduction

Reverse engineering has a long history as an accepted practice. Lawyers and economists have endorsed reverse engineering as an appropriate way for firms to obtain information about another firm's product, even if the intended result is to make a directly competing product that will draw away customers from the maker of the first product.¹ Given this acceptance, it may be surprising that reverse engineering has been under siege in a number of contexts in the past few decades.

Consider these examples: First, in the 1970's and 1980's some states forbade the use of a direct molding process to reverse engineer boat hulls.² Second, the semiconductor industry in the late 1970's and early 1980's sought legislation to protect chip layouts from competitive reverse engineering that industry leaders claimed had market-destructive effects.³ Third, in the mid-1980's and early 1990's, a controversy broke out in the computer software industry about whether a common form of reverse engineering of computer programs, namely, decompilation of machine-readable forms of programs to human-readable forms, was legal as a matter of copyright law.⁴ Fourth, even after U.S. courts decided that decompilation for a legitimate purpose such as achieving interoperability with other programs was lawful,⁵ a related controversy broke out as to whether clauses in software and other digital information licenses that forbade reverse engineering should be enforceable.⁶ Fifth, questions have more recently arisen about whether decompilation of computer programs may infringe patent rights in software components.⁷ Sixth, Congress decided to create a federal cause of action for trade secrecy misappropriation without providing for a reverse engineering defense when enacting the first federal trade secrecy protection statute, the Economic Espionage Act of

¹ See, e.g., JAMES H.A. POOLEY, *TRADE SECRET LAW* sec. 5.02 at 5-16 (1999); David Friedman, William M. Landes, & Richard A. Posner, *Some Economics of Trade Secret Law*, 5 J. Econ. Persp. 61, 71 (1991).

² See Paul Heald, *Federal Intellectual Property Law and the Economics of Preemption*, 76 Iowa L. Rev. 959, 960 (1991). Anti-plug mold laws are discussed *infra* Section II-C.

³ See Section III.

⁴ See Section IV-A.

⁵ See, e.g., *Sega Enterp. Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), discussed in Section IV-A.

⁶ See Section IV-D.

⁷ See, e.g., Julie E. Cohen and Mark A. Lemley, *Patent Scope and Innovation in the Software Industry*, 89 Calif. L. Rev. 1, 6 (2000), discussed in Section IV-C.

1996 (EEA).⁸ Rights granted under the EEA arguably implicate certain reverse engineering activities previously thought to be lawful.⁹ Seventh, in 1998, Congress outlawed reverse engineering of technical protection measures used by copyright owners to protect digital versions of their works; this law also outlaws manufacture or distribution of tools for engaging in such reverse engineering (except in very limited circumstances) and disclosure of information obtained in the course of lawful reverse engineering.¹⁰ Eighth, the international treaty known as the Agreement on Trade-Related Aspects of Intellectual Property Rights (TRIPS) establishes an obligation on the part of member states of the World Trade Organization to protect trade secrets, yet it too lacks a reverse engineering privilege.¹¹

Our motivation in this article is to understand why there have been so many proposals to restrict reverse engineering in recent years and whether actual or proposed restrictions on reverse engineering are economically justifiable. We conclude that the legal rule favoring reverse engineering has been an economically sound rule in the context of a manufacturing economy in which reverse engineering has, in general, been costly and time-consuming, allowing incremental innovators a reasonable lead-time in which they could recoup their initial research and development (R&D) expenditures.¹² Much of the know-how required to make manufactured goods typically remained within the factory when the product went to market. It was, moreover, often difficult, and sometimes impossible, to discern know-how necessary to make the product from disassembly or testing of the product. Not only is reverse engineering time-consuming and costly—perhaps high enough to induce second comers to license know-how from the

⁸ Pub. L. No. 104-294, 110 Stat. 3488 (codified at 18 U.S.C. ss. 1831-1839). The “troubling” absence of a specific reverse engineering privilege in the EEA has been noted in James H.A. Pooley, Mark A. Lemley, & Peter J. Toren, *Understanding the Economic Espionage Act of 1996*, 5 Tex. Intell. Prop. L.J. 177, 195 (1997). See also Rochelle C. Dreyfuss, *Trade Secrets: How Well Should We Be Able to Hide Them? The Economic Espionage Act of 1996*, 9 Fordham Intell. Prop., Media, & Ent. L.J. 1, 15 (1998).

⁹ See Pooley et al., *supra* note 8, at 195-96. Specifically, the concern is that decompilation and disassembly of computer programs, which are now considered to be fair means of obtaining trade secret information in programs, may run afoul of the new EEA rules. *Id.* See also Craig L. Urich, *The Economic Espionage Act: Reverse Engineering and the Intellectual Property Public Policy*, 7 Mich. Telecomm. Tech. L. Rev. 147 (2001)(recommending amendments to the EEA to privilege legitimate reverse engineering activities).

¹⁰ 17 U.S.C. sec. 1201. There is a limited exception to enable bypassing technical controls and making tools to enable this when necessary to achieve interoperability among programs. See 17 U.S.C. sec. 1201(f). This law is discussed in Section V.

¹¹ See Final Act Embodying the Results of the Uruguay Round of Multilateral Trade Negotiations, Apr. 15, 1994, reprinted in THE RESULTS OF THE URUGUAY ROUND OF MULTILATERAL TRADE NEGOTIATIONS--THE LEGAL TEXTS 2-3 (Gatt Secretariat ed. 1994); Marrakesh Agreement Establishing the World Trade Organization, Apr. 15, 1994, Annex 1C: Agreement on Trade-Related Aspects of Intellectual Property Rights (hereinafter “TRIPS Agreement”), reprinted in RESULTS OF URUGUAY ROUND, *supra* at 6-19, 365-403. The trade secrecy provision of the TRIPS Agreement is Article 39. For Congressional approval of the TRIPS and WTO Agreements, see Uruguay Round Agreements Act, Pub. L. No. 103-465, §§101-103, 108 Stat. 4809 (1994). But see Charles R. McManis, *Taking TRIPS on the Information Superhighway: International Intellectual Property Protection And Emerging Computer Technology*, 41 Vill. L. Rev. 207 (1996)(arguing that reverse engineering of software is accepted within the TRIPS framework).

¹² See, e.g., J.H. Reichman, *Legal Hybrids Between the Patent and Copyright Paradigms*, 94 Colum. L. Rev. 2432, 2438-40, 2506-11 (1994). Reichman has been a pioneer among intellectual property scholars in probing the tacit role of trade secret law in providing lead-time to innovators.

innovator—but so is the second comer’s development of a product based on know-how obtained from reverse engineering.¹³ Hence, reverse engineering does not destroy, even if it somewhat erodes, the lead-time an innovator enjoys from introducing a new or improved product into the marketplace.

But are legal rules in favor of reverse engineering economically sound for an information economy? It is noticeable that most of the challenges to reverse engineering mentioned above involve information economy products. If, as some commentators have suggested, information-rich products—that is, products that bear much or all of the know-how required to make them on or near the face of the product sold in the marketplace—are more vulnerable to market-destructive appropriations than manufactured products have been,¹⁴ this might explain why there have been so many efforts to restrict reverse engineering in recent years. Perhaps such vulnerabilities justify some restrictions on reverse engineering that were unnecessary in the manufacturing economy.

The article begins in Section II with an assessment of the law and economics of reverse engineering in traditional manufacturing industries. In Sections III, IV and V, it moves on to consider the law and economics of reverse engineering in three information-based industries: the semiconductor chip industry, the computer software industry, and the emerging market in technically protected entertainment products, such as DVD movies. In all three contexts, rules restricting reverse engineering have been adopted or proposed.

In Section VI we consider reverse engineering as one of the important policy levers of intellectual property law, along with term and scope of protection rules. The most obvious settings for the reverse engineering policy lever are “on” (reverse engineering is permissible) or “off” (reverse engineering is impermissible). However, our study reveals five additional strategies for regulating reverse engineering in the four industrial contexts studied. We distinguish in this discussion between regulations on the act of reverse engineering itself and regulations as to what the reverse engineer can do with information derived in the reverse engineering process. We also consider possible policy responses when innovators seek to thwart reverse engineering rights by contract or by technical obfuscation.

¹³ See *infra* Section II-B.

¹⁴ See, e.g., J.H. Reichman, *Computer Programs as Applied Scientific Know-How: Implications of Copyright Protection for Commercialized University Research*, 42 Vand. L. Rev. 639, 660 (1989) (“[T]oday’s most productive and refined technical innovations are among the easiest of all forms of industrial know-how to duplicate. Because each product of the new technologies tends to bear its know-how on its face, like an artistic work, each is exposed to instant predation when successful and is likely to enjoy zero lead time after being launched on the market.”). See also Reichman, *Legal Hybrids*, *supra* note 12, at 2511-18; Pamela Samuelson, Randall Davis, Mitchell D. Kapor, & J.H. Reichman, *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 Colum. L. Rev. 2308, 2314 (1994) (characterizing software as an information product that is more vulnerable than traditional manufactured goods to market-destructive appropriations) (cited hereinafter as “Manifesto”). See also Rochelle Cooper Dreyfuss, *A Wiseguy’s Approach to Information Products: Muscling Copyright and Patent Into a Unitary Theory of Intellectual Property*, 1992 Sup. Ct. Rev. 195 (1993).

Our principal goals are, first, to assess whether the various mechanisms for regulating reverse engineering in particular industrial contexts are economically sound, and second, to provide an intellectual framework for analyzing the economic effects of reverse engineering rules to assist policymakers in fine-tuning the law or in considering future proposals to restrict reverse engineering. Intellectual property law in the United States has a deeply economic purpose of creating incentives for investments in innovation as a means of advancing consumer welfare.¹⁵ Economic analysis has much to offer as an aid to legal decision-makers in shaping the contours of intellectual property law.¹⁶ Legal limitations on reverse engineering should be tailored to achieve utilitarian goals of intellectual property law and extend no farther than necessary to achieve these goals.

II. Reverse Engineering in Traditional Manufacturing Industries

The law generally allows the reverse engineering of manufactured products. While reverse engineering may be undertaken for many reasons,¹⁷ we concentrate here on reverse engineering for the purposes of making a competing product because this is the most economically significant reason to reverse engineer in this industrial context.¹⁸ From an economic standpoint, a right to reverse engineer is typically sound because the innovator is nevertheless protected in two ways: by the lead-time it enjoys and by the costliness of reverse engineering. Lead-time serves the same function as a short-lived intellectual property right. Costliness may prevent reverse engineering entirely, especially if the first comer licenses its innovation as a strategy for preventing unlicensed entry. Provided the cost of reverse engineering is high enough, such licensing will be on terms that permit the innovator to recoup its R&D expenses, while at the same time constraining the exercise of market power in order to dissuade potential entrants. However, when a particular method of reverse engineering makes it so cheap, easy, and rapid to make competing products that innovators will find it difficult to recoup their R&D expenses, such as with the plug-molding of boat hulls, it may be economically sound to regulate this means of reverse engineering.

A. A Legal Perspective on Reverse Engineering

The law of reverse engineering in the traditional manufacturing industry context can be simply stated. Reverse engineering—i.e., “starting with the known product and

¹⁵ See, e.g., *Mazer v. Stein*, 347 U.S. 201, 219 (1954) (“The economic philosophy behind the clause empowering Congress to grant patents and copyrights is the conviction that encouragement of individual effort by personal gain is the best way to advance public welfare through the talents of authors and inventors.”)

¹⁶ See generally Symposium, *Taking Stock: The Law and Economics of Intellectual Property Rights*, 53 Vand. L. Rev. 1727 (2000). See also Pamela Samuelson, *Economic and Constitutional Influences on Copyright Law in the United States*, 23 Eur. Intell. Prop. Rev. (forthcoming 2001).

¹⁷ Pooley identifies six reasons for engaging in reverse engineering: 1) learning, 2) changing or repairing a product, 3) providing a related service, 4) developing a compatible product, 5) creating a clone of the product, and 6) improving the product. See Pooley, *supra* note 1, at 5-18 to 5-19.

¹⁸ Reverse engineering undertaken for purposes of repairing a purchased product may well affect the manufacturer’s aftermarkets (e.g., for spare parts or service), but this will generally have less of an economic effect on the manufacturer than if the reverse engineer makes a competing product. Reverse engineering to achieve compatibility will be discussed *infra* Section IV-B.

working backwards to divine the process which aided in its development or manufacture”¹⁹—has always been a lawful way to acquire a trade secret, as long as “acquisition of the known product...[is] by fair and honest means, such as purchase of the item on the open market.”²⁰ As the Restatement of Unfair Competition points out, “[t]he owner of a trade secret does not have an exclusive right to possession or use of the secret information. Protection is available only against a wrongful acquisition, use or disclosure of the trade secret,”²¹ as when the use or disclosure is in breach of a understanding between the parties or when improper means, such as trespass or deceit, are used to obtain the secret.²² Even when a firm has misappropriated another firm’s trade secret, injunctive relief may be limited in duration based in part on the court’s estimation of how long it would take a reverse engineer to discover the secret lawfully.²³

The legal “right” to reverse engineer a trade secret—that is, to take apart a product to discover information about the product’s composition and how to make it (that is, applied industrial know-how)²⁴—is so well-established that neither courts nor commentators have generally perceived a need to explain the rationale for this doctrine. A rare exception is the 1989 U.S. Supreme Court decision, *Bonito Boats, Inc. v. Thunder Craft Boats, Inc.*, which characterized reverse engineering as “an essential part of innovation,” likely to yield variations on the product that “could lead to significant advances in technology.”²⁵ The Court added that “the competitive reality of reverse engineering may act as a spur to the inventor” to develop patentable ideas.²⁶ Even when reverse engineering does not lead to additional innovation, the *Bonito Boats* decision suggests it may still promote consumer welfare by providing consumers with a competing product at a lower price.²⁷

Courts have also treated reverse engineering as an important factor in maintaining balance in intellectual property law. Federal patent law allows innovators to have up to twenty years of exclusive rights to make, use and sell the invention,²⁸ but only in

¹⁹ See *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 476 (1974) (defining reverse engineering).

²⁰ Official Comment on Sec. 1 of Uniform Trade Secrets Act (cited hereinafter as UTSA).

²¹ AMERICAN LAW INSTITUTE, RESTATEMENT OF THE LAW OF UNFAIR COMPETITION, comment a to Sec. 43 at 493 (1993) (cited hereinafter as “Restatement of Unfair Competition”).

²² AMERICAN LAW INSTITUTE, RESTATEMENT OF TORTS, sec. 757 (1939); UTSA, supra note 20, sec. 1.

²³ See, e.g., Heald, supra note 2, at 975.

²⁴ See, e.g., Reichman, *Programs As Know-How*, supra note 14, at 656-62 (discussing the economic significance of applied industrial know-how). James Pooley emphasizes that the “fundamental purpose of reverse engineering is discovery, albeit of a path already taken.” Pooley, supra note 1, at 5-19.

²⁵ 489 U.S. 141, 160 (1989), discussed infra Section II-C. See also MATTHEW JOSEPHSON, EDISON: A BIOGRAPHY 91 (1992) (“When the devices of others were brought before him for inspection, it was seldom that [Edison] could not contribute his own technical refinements or ideas for improved mechanical construction. As he worked constantly over such machines, certain original insights came to him; by dint of many trials, materials long known to others, constructions long accepted, were ‘put together in a different way’---and there you had an *invention*.”) (emphasis in original).

²⁶ *Bonito Boats*, 489 U.S. at 160.

²⁷ See, e.g., Heald, supra note 2, at 970. The Supreme Court did not make this point as directly as Heald, although it emphasized the right of the public to make use of unpatented designs in general circulation. See *Bonito Boats*, 489 U.S. at 164-65.

²⁸ 35 U.S.C. sec. 271(a).

exchange for disclosure of significant details about their inventions to the public.²⁹ This deal is attractive in part because if an innovator chooses to protect its invention as a trade secret, such protection may be short-lived if it can be reverse-engineered. If state legislatures tried to make trade secrets immune from reverse engineering, this would undermine federal patent policy because it would “convert the [] trade secret into a state-conferred monopoly akin to the absolute protection that a federal patent affords.”³⁰ Reverse engineering is, then, an important part of the balance implicit in trade secret law.

No reverse engineering right, as such, exists in patent law.³¹ In theory, there should be no need to reverse engineer a patented invention to get information about how to make it because the patent specification should inform the relevant technical community of how to make the invention, and indeed the best mode of making it.³² Insofar as a patent does not teach technologists everything they might want to know, it is clear that some reverse engineering activities will not infringe a patent. The purchaser of a machine embodying a patented invention, for example, is generally free to disassemble it to study how it works under the first sale principle of patent law.³³ In addition, a

²⁹ 35 U.S.C. sec. 112 (setting forth disclosure requirements). The Supreme Court in *Kewanee* spoke of patent law’s disclosure requirement as “the quid pro quo of the right to exclude.” *Kewanee*, 416 U.S. at 484. See also *id.* at 484-92 (emphasizing the importance of disclosure in achieving federal patent objectives and weaknesses in trade secrecy law, including the right to reverse engineer, as reasons why trade secrecy law does not conflict with federal patent policy).

³⁰ See *Chicago Lock Co. v. Fanberg*, 676 F.2d 400, 404 (9th Cir. 1981). *Fanberg* relies on the Supreme Court’s decision in *Kewanee*, 416 U.S. 470, in support of this position. *Kewanee* considered whether state trade secrecy law was in conflict with federal patent policy such that it should be preempted by this federal law. The majority in *Kewanee* concluded that no serious conflict existed because trade secrecy law was both weaker and different from patent law. Reverse engineering was one of the features of trade secrecy law that made it weaker and different from patent law. See *Kewanee*, 416 U.S. at 489-90. See also Pooley, *supra* note 1, at 5-16 (because reverse engineering makes trade secret law weaker than patent law, trade secret law is not preempted by patent law); 1 JAGER ON TRADE SECRETS sec. 5.04[3] at 5-39 (“The likelihood that unpatented objects in the public domain will be reverse engineered is part of the federal balance. It is an inducement to create patentable inventions.”). See also *Rockwell Graphic Systems, Inc. v. DEV Industries, Inc.*, 925 F.2d 174, 178-80 (7th Cir. 1991)(discussing reverse engineering as a limitation on trade secret protection).

³¹ See, e.g., Cohen & Lemley, *supra* note 7, at 6. Although there is no reverse engineering right, as such, in another U.S. intellectual property rights law, the Plant Variety Protection Act, 7 U.S.C. sec. 2321 et seq., there is a research exemption that serves a similar function: “The use and reproduction of a protected variety for plant breeding or other bona fide research shall not constitute an infringement of the protection provided under this Act.” *Id.* at sec. 2544. “The Research Exemption allows anyone to use the PVPA protected lines in a laboratory or field breeding research program to develop new lines. For example, a second comer may purchase a commercially available, PVPA protected soybean variety and use it to develop a new line. This new line can be sold or applied [sic] for protection of its own as long as it is new, distinct, uniform, and stable.” Memorandum of Christine Duh, pp. 2-3, Aug. 6, 2001 (on file with the authors).

³² 35 U.S.C. sec. 112.

³³ See, e.g., Cohen & Lemley, *supra* note 7, at 30-35. By purchasing a manufactured product, the owner acquires the right to use it. Since disassembling a manufactured product does not involve making or selling the invention, no patent rights are implicated by reverse engineering in this context. See *infra* notes xx and accompanying text for a discussion of special characteristics of computer software that suggest that disassembly of this kind of product may implicate patent rights.

While disassembly of a manufactured product is generally lawful, some courts have sometimes enforced a contractual restriction on reverse engineering. See *K&G Oil & Tool Service Co. v. G&G Fishing Tool Service*, 158 Tex. 94, 314 S.W.2d 782, 785-86 (1958)(enforcing a negotiated agreement not to

person who makes a patented invention to satisfy scientific curiosity may assert an experimental use defense to patent infringement.³⁴

Until quite recently, copyright law neither had nor had need for a reverse engineering privilege. The creative works this law had traditionally protected did not need to be reverse-engineered to be understood.³⁵ Books, paintings, and the like bear the know-how they contain on the face of the commercial product sold in the marketplace. To access this information, one simply needs to read or analyze the work. Moreover, copyright law in the United States, at least till the admission of computer programs to its domain, did not protect industrial products of the sort that firms typically reverse-engineer.³⁶

B. An Economic Perspective on Reverse Engineering

The economic effects of reverse engineering depend on a number of factors, including the purpose for which reverse engineering is undertaken, the industrial context within which such acts occur, the costs and time required to engage in reverse engineering, the amount of lead-time the innovator has before competitive entry occurs, whether second comers decide to reverse engineer or to license the innovation, and what the reverse engineer does with the information discerned in the reverse engineering process.³⁷ In this subsection, we concentrate on the economic effects of reverse engineering undertaken for the purpose of developing a similar competing product or an

disassemble K&G's magnetic fishing tool against competitor who then developed substantially the same tool). See also *Pioneer Hi-Bred Int'l, Inc. v. DeKalb Genetics Corp.*, 51 U.S.P.Q. 1987 (S.D. Iowa 1999)(enforcing "bag tag" prohibiting purchasers of PVPA-protected corn seed from using the seed for breeding or research purposes). For further discussion of the enforceability of contractual restrictions on reverse engineering in the computer software industry context, see *infra* Section IV-D.

³⁴ See, e.g., ROGER M. MILGRIM, *MILGRIM ON TRADE SECRETS* sec. 1.05[3] at 1-250 (2000). In U.S. patent law, the experimental use defense is quite narrow, not encompassing, for example, scientific or research use that may lead to development of a patentable invention or a commercial product. See, e.g., Rebecca S. Eisenberg, *Patents and the Progress of Science: Exclusive Rights and Experimental Use*, 56 U. Chi. L. Rev. 1017 (1989)(arguing for a broader experimental use defense in U.S. patent law). Exempting experimental uses of inventions from the scope of the patent right has achieved considerable acceptance in the international community. See, e.g., Janice M. Mueller, *No "Dilettante Affair": Rethinking the Experimental Use Exception to Patent Infringement for Biomedical Research Tools*, 76 Wash. L. Rev. 1, 37-39 (2001).

³⁵ See Section IV-A for a discussion of the controversy in copyright law over the legality of reverse engineering of computer software, a non-traditional copyright subject matter that does not reveal its know-how on the face of mass-market products.

³⁶ Pictorial, sculptural or graphic works can be protected by U.S. copyright law unless they have a utility beyond conveying information or displaying of an appearance. See 17 U.S.C. sec. 101 (definitions of pictorial, sculptural and graphic works and of useful article). Many industrial products (e.g., chairs, automobiles and toasters) have an aesthetic appearance, yet not be copyrightable in the U.S. because their aesthetic design is not separable from the utilitarian functions the products have. See, e.g., *Brandir Int'l, Inc. v. Cascade Pacific Lumber Co.*, 834 F.2d 1142 (2d Cir. 1987) (aesthetic design for bicycle rack uncopyrightable because of inseparability of functional considerations in final design).

³⁷ Reverse engineering does not of itself render the trade secret valueless because reverse engineers do not generally publish their discoveries, but instead maintain the discovered information as their own trade secret. See, e.g., Pooley, *supra* note 1, at 5-19. If reverse engineers do publish the information, this can erode an innovator's ability to recoup its R&D expenses because the secret will have gotten out.

improved product. In the traditional manufacturing context, this has been the most common and economically significant purpose for reverse engineering.³⁸

An economic assessment of the effects of reverse engineering undertaken to develop a competing manufactured product must take into account that reverse engineering is only one step in what is typically a four-stage development process. A second comer's development process as a whole will generally be sufficiently time-consuming, difficult, and costly that an innovator will have significant de facto lead-time protection within which to recoup R&D expenses.³⁹

The first stage of a second comer's development process is an awareness stage.⁴⁰ This involves a firm's recognition that another firm has introduced a product into the market that is potentially worth the time, expense and effort of reverse engineering. In some markets, recognition happen very rapidly; in others, it may take some time, during which the innovator can begin to recoup its R&D costs by selling its product and establishing good will with its customer base.⁴¹

Second is the reverse engineering stage. This begins when a second comer obtains the innovator's product and starts to disassemble and analyze it to discern of what and how it was made.⁴² The reverse engineering stage may be costly, time-consuming, and difficult,⁴³ although this varies considerably, depending mainly on how readily the

³⁸ The economic issues arising from reverse engineering for purposes of developing complementary products are explored infra Section IV-B.

³⁹ We emphasize lead-time as a critical factor in the economic analysis of the effects of reverse engineering because empirical studies conducted of manufacturing firms over a series of years demonstrate that such firms rely more on lead-time than on patents as the principal source of protection for their intellectual assets. See, e.g., Wesley M. Cohen, Richard R. Nelson, & John P. Walsh, *Protecting Their Intellectual Assets: Appropriability Conditions and Why U.S. Manufacturing Firms Patent (or Not)* (Feb. 2001)(manuscript on file with the author). See also Reichman, *Legal Hybrids*, supra note 12, at 2439-41 (explaining the importance of lead-time in trade secrecy law).

⁴⁰ The more innovative the product, the longer it may take for potential competitors to recognize the innovation and undertake to copy it. However, the innovator may also find it difficult to achieve initial market success. See, e.g., GEOFFREY A. MOORE & REGIS MCKENNA, *CROSSING THE CHASM: MARKETING AND SELLING HIGH TECHNOLOGY PRODUCTS TO MAINSTREAM CUSTOMERS* (1991). Because of this, the more innovative the product, the more economically sensible it will generally be to obtain patent protection for key aspects of the innovation to impede competitive imitation.

⁴¹ For some consumers, a firm's reputation for innovation or quality service will make its product attractive even if second comers eventually copy it. To the extent there are switching costs associated with the product (e.g., owing to a steep learning curve in how to use it), the innovator may also benefit from "lock-in" of its initial customers and those who later value the innovator's product because others are using it. See, e.g., Mark A. Lemley & David McGowan, *The Law and Economics of Network Effects*, 86 Calif. L. Rev. 479 (1998).

⁴² The reverse engineer's purchase of a competitor's product to reverse engineer it does, of course, make some contribution toward recoupment of the innovator's costs; this may be trivial, however, in the case of many mass-market goods.

⁴³ Products vary considerably in the ease with which they can be reverse engineered. In general, the more difficult reverse engineering is, the greater value the secret will have, the longer lead-time advantage the trade secret holder will enjoy in the market, and the less incentive the holder may have to license the secret. See, e.g., Pooley, supra note 1, at 4-42. See also Restatement, supra note 22, sec. 39, com. f. Firms can sometimes make reverse engineering more difficult, and this may be an economically sensible thing to do if the secret is valuable. See, e.g., Pooley, supra note 1 at 5-25: "It may be possible to build products that are

innovator's product will yield the know-how required to make it when confronted by a determined and skilled reverse engineer.⁴⁴ However, a reverse engineer will generally spend less time and money to discern this know-how than the initial innovator spent in developing it, in part because the reverse engineer is able to avoid wasteful expenditures investigating approaches that don't work, and in part because advances in technology typically reduce the costs of rediscovery over time.⁴⁵

Third is the implementation stage.⁴⁶ After reverse engineering the innovator's product, a second comer must take the know-how obtained during the reverse engineering process and put it to work in designing and developing a product to compete in the same market. This may involve making prototypes, experimenting with them, retooling manufacturing facilities, and reiteration of the design and development process until it yields a satisfactory product. It may be necessary to return to the reverse engineering stage again if it becomes apparent in the implementation phase that some necessary know-how eluded the reverse engineer the first time. Information obtained during reverse engineering may, moreover, suggest possibilities for additional product

difficult to break down and copy. Hardware components can be encapsulated to make nondestructive disassembly almost impossible; components can be mislabeled...; custom parts can be used; 'locks' (often implemented in software) can be added. In any sort of complex product, nonfunctional features can be added to create a 'fingerprint' on any illegitimate copy, forcing copyists to invest in real reverse engineering efforts." Friedman, Landes & Posner regard the expenditures required to make the product more difficult to reverse engineer as costs of not prohibiting reverse engineering. Friedman et al., supra note 1, at 70. Professor Kitch discusses other reasons it is difficult to "steal" valuable information. See Edmund W. Kitch, *The Law and Economics of Rights in Valuable Information*, 9 J. Legal Stud. 683, 711-15 (1980). See also Steven N.S. Cheung, *Property Rights in Trade Secrets*, 20 Econ. Inquiry 40, 47 (1982)(discussing economics of trade secrecy law and various means by which trade secret rents may be dissipated).

⁴⁴ See, e.g., Pooley, supra note 1, at 4-41. The relative difficulty of reverse engineering does not, of course, match up perfectly with the difficulty and expense of developing the secret in the first place. Some trade secrets may have been serendipitously developed at low cost, yet are difficult to reverse engineer, while other expensive and time consuming innovations may be impossible to hide in the final product. Still, some commentators contend that "inventiveness often correlates with difficulty of reverse engineering, with the result that the more inventive the product, the longer its inventor enjoys the so-called 'first mover advantage,' and the more profit she earns." ROCHELLE C. DREYFUSS & ROBERTA R. KWALL, *INTELLECTUAL PROPERTY: CASES AND MATERIALS ON PATENTS, COPYRIGHTS, AND TRADEMARKS* 818 (1996).

A further consideration is how difficult or easy it is to detect whether another firm independently developed the same or a similar innovation or engaged in reverse engineering to discover it. Reverse engineering, after all, tends to occur behind closed doors. See Friedman et al., supra note 1, at 70; Kitch, supra note 43, at 690. However, it is sometimes possible to persuade courts that independent invention of the same trade secret was unlikely. See, e.g., *Pioneer Hi-Bred Int'l v. Holden Foundation Seeds*, 35 F.3d 1226 (8th Cir. 1994).

⁴⁵ See, e.g., Friedman et al., supra note 1 at 63. See also JARED DIAMOND, *GUNS, GERMS, AND STEEL* (1999)(giving examples of technologies, the reinvention of which occurred rapidly once it became known that the technology was possible).

⁴⁶ During both the reverse engineering and the implementation stages, the innovator may decide to license its know-how to the second comer. Over time, the innovator's willingness to license may increase, especially if it has reason to think that certain second comers are making progress toward developing a competing or improved product. The second comer's willingness to take a license may decline as his expenditures in reverse engineering and redevelopment rise and as it perceives these efforts to be bearing fruit. Yet, a license from the innovator may become attractive if fine details of implementation elude the reverse engineer.

innovation that will be investigated in the implementation stage.⁴⁷ For these reasons, the second comer's implementation stage may take considerable time and require significant expense.

The fourth stage in the second comer's development process is the introduction of its product to the market. How quickly the new product will erode the innovator's market share and force the innovator to reduce prices to be competitive with the new entrant will depend on various market factors.⁴⁸

An economic assessment of reverse engineering should balance four factors: 1) the effects of reverse engineering on incentives to innovate, 2) its effects on prices, 3) its potential for creating incentives to invest in improved products, and 4) its potential for duplicated or other socially wasteful expenditures of resources. Consider, for example, that a prohibition on reverse engineering would seem to have two beneficial effects: It increases incentives to introduce innovative products into the market, and it avoids wasteful expenditures on reverse engineering.⁴⁹ However, reverse engineering has beneficial effects that must also be considered: It can create competition in the marketplace, leading to lower prices, and it can spur second comers to introduce additional innovations into that market. While this first-cut analysis might suggest a stasis as to economic effects, it is misleading because it does not consider incentives to license know-how to second comers that can avoid some wasteful expenditures or duplicated costs if reverse engineering is legal.

The innovator's incentive to license is precisely due to the threat of reverse engineering. Under this threat, an innovator knows that its market position can be eroded by unlicensed entry. The firm can preempt this outcome by authorizing entry in a controlled way through licenses. Licensing should be in the interest of both the innovator and potential reverse engineers because it allows the former to recoup costs through licensing revenues and it imposes costs on the latter that will preclude them from pricing their products in a market-destructive way.⁵⁰ Licensing can achieve the same knowledge-sharing and market outcomes as reverse engineering without incurring the costs of reverse engineering. The cost savings can be shared through the terms of the license.⁵¹

⁴⁷ See, e.g., Richard C. Levin et al., *Appropriating Returns from Industrial Research and Development*, 1987 BROOKINGS PAPERS ON ECONOMIC ACTIVITY 783, 805-07 (improvements likely to result from reverse engineering).

⁴⁸ It bears repeating that an innovator may be able to hold on to its leading market share if it has a positive reputation for quality or service, it has a strong brand, or there are high switching costs.

⁴⁹ See, e.g., Martin J. Adelman, *Property Rights Theory and Patent-Antitrust: The Role of Compulsory Licensing*, 52 N.Y.U. L. Rev. 977, 982 (1977)(expressing concern about wasteful expenditures of reinvention). Another set of socially wasteful costs that may be incurred if reverse engineering is legal are the costs of making one's product difficult to reverse engineer. See supra note 43.

⁵⁰ See, e.g., Reichman, *Legal Hybrids*, supra note 12, at 2441. See also J.H. Reichman, *Of Green Tulips and Legal Kudzu: Repackaging Rights in Subpatentable Innovation*, 53 Vand. L. Rev. 1743 (2000) (proposing a compensatory system to enable developers of subpatentable innovations to recoup R&D expenses).

⁵¹ Similarly, the consequences for competition do not depend on whether reverse engineering is avoided by licensing. An unlicensed second comer's entry will have a direct downward effect on prices, but its pricing strategy will be affected by its need to recover costs incurred in reverse engineering and implementation of

Furthermore, although forbidding reverse engineering would enhance incentives to innovate, the important question is whether permitting reverse engineering so dissipates profit that it discourages innovation. The answer to this question will chiefly depend on the costs of reverse engineering relative to the innovator's development costs and on how long the process of reverse engineering takes. As we pointed out above, costs of reverse engineering are often substantial and take considerable time.⁵² Of course, if the costs of reverse engineering are very low in comparison with the costs of the initial development and if second comers can enter quickly because reverse engineering was easy, the second comer's entry will rapidly drive prices down due to the rival's cost advantage; this would likely deprive the innovator of revenues sufficient to cover its development costs.⁵³ The effect of a reverse engineering right on incentives to innovate will thus depend on the relative costs incurred by the innovator and potential reverse engineers and on the natural lead-time protection afforded by delay, not on whether reverse engineering is avoided by licensing.

Table 1 illustrates our analysis of the social welfare effects of reverse engineering in traditional manufacturing industries:

Table 1
Social Calculus of Reverse Engineering in Manufacturing Sector

	RE legal	RE illegal
Incentives to innovate	lower (but adequate)	higher (but excessive)

the innovation. A licensed second comer operates under similar constraints, for it must price its product so as to recoup a one-time license fee or cover the costs of an ongoing royalty owed to the innovator. If reverse engineering is a legal option, an innovator will set license fees low enough to attract licensees and discourage unlicensed entrants from reverse engineering. See Stephen Maurer & Suzanne Scotchmer, *The Independent Invention Defense in Intellectual Property*, John M. Olin Working Paper 98-11, U.C. Berkeley (Boalt Hall) (1998).

⁵² See supra notes xx and accompanying text. See also Dreyfuss & Kwall, supra note 44, at 818:

Because reverse engineering generally takes time (time to decide the product is worth figuring out as well as time to actually do the engineering and bring the product to market), the first inventor enjoys a period of exclusivity in which to recapture the costs of the invention, build a reputation, and establish a base of loyal customers. Furthermore, the copyist is not quite a free rider because reverse engineering is generally expensive. Thus, after the secret is discovered, the parties compete on a fairly level playing field.

⁵³ See, e.g., Wendy J. Gordon, *Asymmetric Market Failure and Prisoners' Dilemma in Intellectual Property*, 17 U. Dayton L. Rev. 853 (1992) (discussing conditions under which market failure may arise from appropriation of intellectual creations); Wendy J. Gordon, *On Owning Information: Intellectual Property and the Restitutory Impulse*, 78 Va. L. Rev. 149 (1992) (discussing the concept of "malcompetitive" copying). Douglas Lichtman has argued that incentives to develop subpatentable innovations such as boat hulls will be threatened if there is a right to engage in very low cost reverse engineering, as by use of plug molds. See Douglas Lichtman, *The Economics of Innovation: Protecting Unpatentable Goods*, 81 Minn. L. Rev. 693, 721-23 (1997). Maurer and Scotchmer, supra note 51, argue from the other direction: incentives to innovate will survive a rival's independent innovation whenever its costs are roughly "commensurate" with the innovator's development cost.

Price	lower	higher
Follow-on innovation	higher	lower
Duplicated/wasted costs	higher (but avoidable by licensing)	lower

On balance, we conclude that a legal rule against reverse engineering of traditional manufactured products would be economically unsound because it would, in effect, give firms perpetual exclusive rights in unpatented innovations by precluding access to the know-how embedded in the products.⁵⁴ Given that the costs of reverse engineering in the traditional manufacturing industry context are generally substantial and that innovators generally enjoy substantial lead-time because of reverse engineering takes time, a ban on reverse engineering is unnecessary to provide appropriate incentives to invest in incremental innovations that do not qualify for patent protection (that is, “subpatentable innovations”⁵⁵). On the positive side, a right to reverse engineer has a salutary effect on price competition and on the dissemination of know-how that can lead to new and improved products.

C. Anti-plug Mold Laws: An Exception to Reverse Engineering Rules?

In the late 1970’s and early 1980’s twelve states adopted laws to prohibit plug-molding as a means of reverse engineering existing manufactured products.⁵⁶ Anti-plug mold laws forbade use of a manufactured item, such as a boat hull, as a “plug” for a direct molding process which produced a mold that could then be used to manufacture identical products in direct competition with the plugged product. Florida’s legislature had apparently been convinced that plug molding of boat hulls was undermining incentives to invest in innovative boat designs, thereby harming a significant Florida industry.⁵⁷ California passed a more general anti-plug mold law.

In *Interpart Corp. v. Imos Italia, Vitaloni, S.p.A.*,⁵⁸ the defendant challenged the consistency of the California law with federal patent policy. The Court of Appeals for the Federal Circuit rejected this challenge, characterizing California’s anti-plug mold law as a regulation of a certain use of chattels (i.e., don’t use another firm’s product as a plug

⁵⁴ See, e.g., Friedman et al., supra note 1 at 70-71 (concurring in this view).

⁵⁵ See, e.g., Lichtman, supra note 53; Reichman, *Legal Hybrids*, supra note 12, at 2439.

⁵⁶ See, e.g., Heald, supra note 2, at 960, 962. In some countries, parasitical copying such as that conducted by a plug mold process is illegal as a matter of unfair competition law. See, e.g., Reichman, *Legal Hybrids*, supra note 12, at 2472-74.

⁵⁷ *Bonito Boats*, 489 U.S. at 158. See also Lichtman, supra note 53, at 719-20. The direct molding process was itself a relatively new technological innovation that had been patented in 1968. See *Bonito Boats*, 489 U.S. at 163-64. The patent specification asserted this advantage to the direct molding process: “It is a major object of the present invention to provide a method for making large molded boat hull molds at very low cost, once a prototype hull has been developed.” *Id.* at 164 (quoting from the patent).

⁵⁸ 777 F.2d 678 (Fed. Cir. 1985).

in a direct molding process).⁵⁹ It perceived no conflict with federal patent law because the California law did not confer a right to exclude others from making, using, or selling the product.⁶⁰ Anyone could reverse engineer and copy a manufactured product by conventional means; they just couldn't do so by plug-molding.⁶¹

Four years later the U.S. Supreme Court overruled *Interpart* in *Bonito Boats, Inc. v. Thunder Craft Boats, Inc.*⁶², ruling that Florida's plug mold laws was in conflict with federal patent policy. One reason the Court gave for striking down this law was that it "prohibits the entire public from engaging in a form of reverse engineering of a product in the public domain."⁶³ The Court said that it was "difficult to conceive of a more effective method of creating substantial property rights in an intellectual creation than to eliminate the most efficient method for its exploitation."⁶⁴ The Court drew upon previous preemption rulings as protecting "more than the right of the public to contemplate the abstract beauty of an otherwise unprotected intellectual creation—they assure its efficient reduction to practice and sale in the marketplace."⁶⁵ It went on to say that "[w]here an item in general circulation is unprotected by a patent, '[r]eproduction of a functional attribute is legitimate competitive activity.'"⁶⁶

However, the economic consequences of plug-molding deserved more serious consideration.⁶⁷ The plug-mold process dramatically reduces the costs of and time

⁵⁹ See *Bonito Boats*, 489 U.S. at 163 (characterizing *Interpart* as resting on this theory)

⁶⁰ *Interpart*, 777 F.2d at 684.

⁶¹ *Interpart*, 777 F.2d at 685.

⁶² 489 U.S. 141 (1989).

⁶³ *Id.* at 160. *Bonito Boats* seems to elevate the principle of reverse engineering to a constitutionally protected interest. See, e.g., *Chicago Lock Co. v. Fanberg*, 676 F.2d 400, 404 (9th Cir. 1982) (for state law not to allow reverse engineering "would, in effect, convert the Company's trade secret into a state-conferred monopoly akin to the absolute protection that a federal patent affords. Such an extension of California trade secrets law would certainly be preempted by the federal scheme of patent regulation.") See also Reichman, *Legal Hybrids*, supra note 12, at 2473 (interpreting *Bonito Boats* as "endow[ing] the competitor's right to reverse engineer with constitutional underpinnings").

⁶⁴ *Bonito Boats*, 489 U.S. at 164.

⁶⁵ *Id.* The cases upon which the Court principally drew were the companion cases of *Sears, Roebuck & Co. v. Stiffel Co.*, 376 U.S. 225 (1964) and *Compco Co. v. Day-Brite Lighting, Inc.*, 376 U.S. 234 (1964). In these cases the Court ruled that state unfair competition law could not be used to protect unpatentable designs from competitive copying because this would interfere with federal patent policy. Although the courts have been consistently hostile to unfair competition-like claims as a means to protect unpatented designs since *Sears* and *Compco*, they have been far more receptive to protecting product configurations against copying under trade dress law. See, e.g., *Sunbeam Products, Inc. v. West Bend Co.*, 123 F.3d 246 (5th Cir. 1997). The Supreme Court has endorsed trade dress claims for product configurations or designs in appropriate cases; yet it has placed a heavy burden of proof on a trade dress claimant to show that the claimed configuration or design is nonfunctional if it was claimed in an expired patent. See *TrafFix Devices, Inc. v. Marketing Displays, Inc.*, 532 U.S. 23 (2001).

⁶⁶ *Bonito Boats*, 489 U.S. at 164. It should be noted that in 1998 Congress enacted a new form of intellectual property protection for vessel hulls. See 17 U.S.C. sec. 1301 et seq. Now they can neither be plug-molded nor copied by any other method.

⁶⁷ Like the Court, economists would be concerned about the distortions likely to arise from non-uniform state laws. As the framers of the U.S. Constitution understood very well, states are not well-equipped to provide effectual protection of publicly disclosed innovations. It is for this reason that the framers included Art. I, sec. 8, cl.8, in the U.S. Constitution. See THE FEDERALIST No. 43, at 338 (James Madison) (John C. Hamilton, ed., 1804) (1788). The non-uniformity problem was present in the *Bonito Boats* case

required to engage in reverse engineering and reimplementing of an innovation. If plug-molding undermines incentives to invest in innovative boat hulls or other manufactured goods,⁶⁸ a ban on the use of the plug-mold process might be economically sound, at least for some period of time. The germ of an argument that plug-molding may have market-destructive effects can be found in *Bonito Boats*. The Supreme Court, for instance, noted that Bonito Boats had expended substantial resources in developing the boat hull that it sought to protect in the litigation against Thunder Craft Boats,⁶⁹ and that the very purpose of the plug mold process was to “provide a method for making large molded boat hull molds at very low cost, once a prototype hull has been provided.”⁷⁰ Yet the Court gave very little attention to these details in its lengthy legal and policy analysis of the case.

The Supreme Court suggested in *Bonito Boats* that plug mold duplication of boat hulls was, “an essential part of innovation in the field of hydrodynamic design.”⁷¹ Professor Heald questioned this assertion, pointing out that the Florida law “primarily discriminates against those interested in reproduction not innovation”⁷² and that plug-molding might well “result in less innovation.”⁷³ Heald’s is the more economically sound view of the effects of plug-molding on follow-on innovation.⁷⁴

because Thunder Craft Boats was a Tennessee-based company and Tennessee had no anti-plug molding statute. See *Bonito Boats*, 489 U.S. at 145.

⁶⁸ It should not be enough for boat designers to testify that they needed such a law. Robert Kastenmeier, former head of the Intellectual Property Subcommittee of the House Judiciary Committee, recognized the danger of new laws to protect particular industries. It is very easy for special interest groups to claim that they need more legal protection, but this does not mean that adopting such a law is necessarily in the overall public interest. To guard against special interest lobbying, Kastenmeier articulated a multi-part test to determine when legislation of this sort would be warranted. See Robert W. Kastenmeier & Michael J. Remington, *The Semiconductor Chip Protection Act of 1984: Swamp or Firm Ground?*, 70 Minn. L. Rev. 417, 438-61 (1985).

⁶⁹ *Bonito Boats*, 489 U.S. at 144.

⁷⁰ Id. at 164 (quoting from the patent).

⁷¹ Id. at 160.

⁷² Heald, supra note 2, at 987. See also Reichman, *Legal Hybrids*, supra note 12, at 2473 (plug-molders merely duplicate originator’s product).

⁷³ Heald, supra note 2, at 987. The Court insisted that enactment of laws to give incentives to invest in innovation is reserved to the federal government, not to states. *Bonito Boats*, 489 U.S. at 157-58. Heald reinforces the Court’s position by asserting that “the Constitution grants Congress the right to experiment in the area. Congress’ intent is frustrated by state statutes whose incentives interfere with Congress’ experiments.” Heald, supra note 2, at 969. However, many state laws, including those that protect trade secrets, trademarks, and rights of publicity, aim, in part, at inducing investment in intellectual creations; yet, they are generally not preempted. See, e.g., John S. Wiley, Jr., *Bonito Boats: Uninformed But Mandatory Innovation Policy*, 1989 Sup. Ct. Rev. 283, 290-94 (1989)(discussing state intellectual property laws threatened by preemption analysis in *Bonito Boats*).

⁷⁴ If reverse engineering is a process that results in discovery of know-how, not just rapid, cheap copying of existing products, one might argue that plug-molding is not reverse engineering at all. As subsection A has shown, reverse engineering and competitive copying of a product are different activities, even if courts, as in *Bonito Boats*, sometimes conflate them. See, e.g., *Bonito Boats*, 489 U.S. at 160 (Florida law “prohibits the entire public from engaging in a form of reverse engineering of a product in the public domain”); *Traffix Devices, Inc. v. Marketing Displays, Inc.*, 532 U.S. 23 (2001)(seeming to conflate reverse engineering and copying). By pointing out this difference, we do not mean to suggest that cloning is always or necessarily economically harmful. As long as the costs of cloning are roughly commensurate

Of course, this does not mean that the laws enacted in Florida or California were chosen on the basis of economic merit. Some features of the Florida law suggest that it was the product of a rent-seeking special interest group lobby. Consider, for instance, that the law applied retroactively to boat hulls already in existence.⁷⁵ Moreover, it did not require any showing of originality, novelty, or improvement as a criterion for the grant of protection.⁷⁶ Nor was there any durational limit to the protection.⁷⁷ It is difficult to believe that perpetual rights are necessary to enable boat hull designers to recoup their R&D expenses.⁷⁸ An economically sound anti-plug mold law might, then, apply only prospectively, have a minimal creativity requirement, and a durational limitation aimed at providing a reasonable amount of lead-time to enable boat hull innovators to recoup its investments, but not more than that.⁷⁹ In 1998, Congress enacted a “sui generis” (of its own kind) form of intellectual property protection to protect boat hulls from unauthorized copying and not just from plug-molding.⁸⁰

From an economic perspective, anti-plug mold laws illustrate that even in the context of traditional manufacturing industries, a form of reverse engineering that produces cheap, rapid identical copies has the potential to have market-destructive consequences. “Quick imitation robs innovation of value.”⁸¹ Insofar as market-

with the costs of initial development or there is enough delay in the cloner’s entry so that the first comer can recoup R&D costs, introduction of an identical product can be economically beneficial.

⁷⁵ Retroactive application of the law cannot incent the creation of existing designs. It is worth pointing out that Bonito Boats developed the 5VBR boat more than six years before the Florida legislature passed the anti-plug mold law, yet the law protected this hull as well as all new designs. *Bonito Boats*, 489 U.S. at 144-45.

⁷⁶ *Id.* Heald was critical of the Florida plug mold law for the lack of a creativity requirement. See Heald, *supra* note 2, at 987.

⁷⁷ *Bonito Boats*, 489 U.S. at 144-45. Some commentators have been critical of the Florida plug mold law on this basis as well. See, e.g., Lichtman, *supra* note 53, at 718.

⁷⁸ By the time Thunder Craft copied the 5VBR boat hull and sold competing boats, Bonito Boats had already had eight years within which to recoup its R&D expenses on that hull. *Bonito Boats*, 489 U.S. at 144-45.

⁷⁹ The new form of intellectual property right Congress enacted in 1998 to protect boat hulls does have an originality requirement and a durational limitation. See 17 U.S.C. secs. 1302 (originality requirement), 1305 (duration limitation).

⁸⁰ See Vessel Hull Design Protection Act, which was Title V of the Digital Millennium Copyright Act, Pub. L. No. 105-304, 112 Stat. 2860 (1998), now codified at 17 U.S.C. sec. 1301 et seq. In protecting the configuration of boat hulls, the VHDPA most closely resembles utility model laws adopted in some countries. See, e.g., Reichman, *Legal Hybrids*, *supra* note 12, at 2455-59 (discussing utility model laws). For the moment, the VHDPA only covers vessel hulls, but only minor changes would be necessary to convert it to a more general intellectual property law to protect the configuration of manufactured products. Congress has rejected legislation of this sort in nearly every session during the 20th century because of concerns it would unduly impede competition in product markets. See, e.g., Richard Frenkel, *Intellectual Property Law in the Balance: Proposals for Improving Industrial Design Protection in the Post-TRIPS Era*, 32 Loy. L.A. Law Rev. 531 (1999). For a discussion of industrial design protection more generally and why it has been controversial over the years, see, e.g., J.H. Reichman, *Design Protection and the New Technologies: The United States Experience in a Transnational Perspective*, 19 U. Balt. L. Rev. 6 (1989-90). However, the expansion of state and federal trade dress protection for product configurations has had much the same effect as an industrial design would have in the U.S. *Id.* The functionality limitation on trade dress protection limits the utility of this law as a surrogate for a European-style utility model law.

⁸¹ Email communication from Michael Moradzadeh to Pamela Samuelson (on file with the authors).

destructive effects can be demonstrated, it may be economically sound for the law to restrict a market-destructive means of reverse engineering and reimplementing for a period of time sufficient to enable the innovator to recoup its R&D expenses.

III. Reverse Engineering In the Semiconductor Industry

Although the semiconductor industry is in many respects a traditional manufacturing industry, we give it separate treatment here for two reasons: first, because semiconductors are information technology products that in the late 1970's and early 1980's bore virtually all of the know-how required to make them on the face of the product sold in the market,⁸² rendering them vulnerable to rapid, cheap copying that was undermining the ability of innovative chip developers, such as Intel, to recoup the very high costs of R&D necessary to produce new chips,⁸³ and second, because Congress was so concerned about the effects that "chip piracy"⁸⁴ was having on incentives to invest in semiconductor innovation that it created a new form of intellectual property protection for semiconductor chip designs.⁸⁵

The Semiconductor Chip Protection Act (SCPA)⁸⁶ is noteworthy for a number of reasons,⁸⁷ among them, that it is an intellectual property law⁸⁸ with an express reverse

⁸² See, e.g., Reichman, *Legal Hybrids*, supra note 12, at 2479-80, Manifesto, supra note 14, at 2338 (discussing the vulnerability of information technology products to market-destructive appropriations because of the high quantum of know-how they bear on or near the face of products sold in the marketplace). See also Morton David Goldberg, *Semiconductor Chip Protection As a Case Study*, in GLOBAL DIMENSIONS OF INTELLECTUAL PROPERTY RIGHTS IN SCIENCE AND TECHNOLOGY (Mitchel B. Wallerstein, Mary Ellen Mogee, & Roberta A. Schoen, eds. 1993) at 333 ("Considerable skill and creativity are invested in the design of the mask works that determine the topography of those products, but this design work is easily appropriated since, in essence, each copy of the product carries its own blueprint with it.")

⁸³ See, e.g., Prepared Testimony of F. Thomas Dunlap, Jr., Corporate Counsel and Secretary of Intel Corp. (cited hereinafter as "Dunlap Statement"), Hearings Before the Subcommittee on Courts, Civil Liberties, and the Administration of Justice of the House Committee on the Judiciary, on H.R. 1028, 98th Cong., 1st Sess. (8/3/83) (cited hereinafter as "House Hearings") explaining the industry's need for this legislation.

⁸⁴ See, e.g., Prepared Statement of Sen. Charles McC. Mathias, Jr., House Hearings, supra note 83, at 3: "[Chip] innovators are being ripped off by onshore and offshore 'chip pirates' who, for a fraction of the developer's cost, can now legally appropriate and use these chip designs as their own." Of particular concern was the loss to Japanese industry of a substantial share of the market for random access memory chips to Japanese competitors whose superior quality control made their chips very competitive. See Stephen P. Kasch, *The Semiconductor Chip Protection Act: Past, Present, and Future*, 7 High Tech. L.J. 71, 79 (1993).

⁸⁵ Some commentators have suggested that the semiconductor industry "greatly overstated the severity of the chip piracy problem" in testimony before Congress. See, e.g., Robert Risberg, *Five Years Without Infringement Litigation Under the Semiconductor Chip Protection Act: Unmasking the Specter of Chip Piracy in an Era of Diverse and Incompatible Technologies*, 1990 Wisc. L. Rev. 241, 244-45 (1990). See also Kasch, supra note 84, at 92 (questioning evidence of chip piracy at legislative hearings).

⁸⁶ Pub. L. No. 98-620, 98 Stat. 3347 (1984), now codified at 17 U.S.C. sec. 901 et seq.

⁸⁷ SCPA has been the subject of much commentary. See, e.g., Kasch, supra note 84; John G. Rauch, *The Realities of Our Times: The Semiconductor Chip Protection Act of 1984 and the Evolution of the Semiconductor Industry*, 3 Fordham Ent., Media & Intell. Prop. L.F. 403 (1993); Risberg, supra note 85; Linda Samuels & Jeffrey M. Samuels, *Semiconductor Chip Protection Act of 1984: An Analytical Commentary*, 23 Am. Bus. L. J. 601 (1986); Terrill Lewis, *Comment, Semiconductor Chip Process Protection*, 32 Hous. L. Rev. 555 (1995). See also *Symposium: The Semiconductor Chip Protection Act of*

engineering privilege.⁸⁹ This privilege not only permits reverse engineers to make copies of protected chip designs in order to study the layouts of circuits, but also to incorporate know-how obtained during reverse engineering in a new chip design.⁹⁰ However, the statute also requires reverse engineers to engage in enough “forward-engineering” to develop an original chip design that itself qualifies for SCPA protection.⁹¹ This contrasts with the legal rule discussed in Section II that generally permits a reverse engineer to make and sell an identical or near-identical product.⁹² The economic justification for establishing a breadth requirement for products resulting from the reverse engineering process in the SCPA is to avoid the market-destructive practice of cloning of innovative chip designs and to promote investment in chip designs that will advance the state of the art.

A. Perturbations in Product Life Cycles in the Chip Industry

The typical product life cycle in the semiconductor industry was relatively constant in the 1970’s and 1980’s.⁹³ A pioneering firm (usually Intel Corp.) would develop an innovative new product and introduce it to the market priced handsomely so that the firm could recoup its investments. “Later, as the manufacture [became] more efficient [the innovator would cut] its prices to expand its market and discourage competition. Nonetheless, second-source products—chips electrically and mechanically compatible with the pioneering product—eventually appear[ed] on the market. The arrival of competition precipitate[d] further rounds of price cuts.”⁹⁴ Toward the end of this life cycle, the pioneer’s profit margins tailed off and the firm could only hope that the next round of innovation would allow it to regain market share and profits.

1984 and Its Lessons, 70 Minn. L. Rev. 263 (1985) (consisting of six articles); RICHARD H. STERN, SEMICONDUCTOR CHIP PROTECTION (1986); ANDREW CHRISTIE, INTEGRATED CIRCUITS AND THEIR CONTENTS: INTERNATIONAL PROTECTION (1995); and articles cited infra notes xx and yy.

⁸⁸ Although trade secrecy is sometimes characterized as a form of intellectual property protection, see, e.g., Stanley M. Besen & Leo J. Raskind, *An Introduction to the Law and Economics of Intellectual Property*, 5 J. Econ. Perspectives 3, 3 (1991), it is more appropriately understood as a branch of unfair competition law. See, e.g., Restatement of Unfair Competition, supra note 21, secs. 39-44. Trade secret law confers no exclusive rights on innovators, as intellectual property statutes typically do, but only protects holders from certain kinds of tortious acts, such as use of improper means or breach of confidence to acquire the secret.

⁸⁹ Professor Raskind has spoken of the reverse engineering privilege as the “capstone” of SCPA. See Leo J. Raskind, *Reverse Engineering, Unfair Competition, and Fair Use*, 70 Minn. L. Rev. 385, 385 (1985). The reverse engineering privilege of SCPA also received attention in other articles, including Harold R. Brown, *Fear and Loathing of the Paper Trail: Originality in Products of Reverse Engineering Under the Semiconductor Chip Protection Act As Analogized to the Fair Use of Nonfiction Literary Works*, 41 Syr. L. Rev. 985 (1990) and Lee Hsu, *Reverse Engineering Under the Semiconductor Chip Protection Act: Complications for Standard of Infringement*, 5 Albany L.J. Sci. & Tech. 249 (1996).

⁹⁰ 17 U.S.C. sec. 906(a).

⁹¹ See, e.g., E.J. Chikofsky & J.H. Cross, *Reverse Engineering and Design Recovery: A Taxonomy*, 7 IEEE Software 13 (1990) (defining forward engineering); Kasch, supra note 84, at 85 (discussing forward engineering in respect of the SCPA).

⁹² See supra Section II-A.

⁹³ Kasch, supra note 84, at 78.

⁹⁴ Id.

Semiconductor firms relied on lead-time and secrecy far more than on patents to protect their intellectual assets.⁹⁵ An innovator could rely not only on being first to market to provide some lead-time, but also on being further along the “yield curve” than imitating second comers. In the early stages of a new production process, chip makers could count on a lower yield of salable chips than in later stages of the production process when fine-tuning of the production would yield a higher quantity of high quality chips. Trade secrecy protection was especially important in the chip manufacturing process because considerable know-how was required to make commercially acceptable chips. However, trade secrecy law could obviously not protect the layout of chips sold in the marketplace because this information was readily ascertainable from examination of the marketed product (that is, it could be easily reverse engineered).⁹⁶

Several factors contributed to patents not playing as crucial a role in the early and mid-phases of this industry as one might expect.⁹⁷ For one thing, semiconductors are a cumulative system technology in which the interrelatedness of inventions has meant that extensive cross-licensing of patents was necessary for industry participants to make advanced chips.⁹⁸ Second, some major customers of this industry, including notably the U.S. government, insisted on “second-sourcing,” that is, in having competitive suppliers of compatible chips to reduce the risk of unforeseen supply problems.⁹⁹ This too contributed to widespread cross-licensing. Third, the rapid pace of innovation and short life cycles of many chip products lessened the utility of patents in this industry.¹⁰⁰ Fourth, during the 1970’s, when the semiconductor industry was becoming a major American industry, there was a widespread perception that courts were hostile to patents, and patents had, as a consequence, less economic significance than at other times.¹⁰¹ A fifth limitation of patents, much emphasized in the legislative history of SCPA, was that under then prevailing standards, the overall layout of chip circuits was rarely if ever patentable.¹⁰²

⁹⁵ See, e.g., Cohen et al., supra note 39, Tables A1 and A2 (showing that semiconductor firms regard trade secrecy and lead-time as far more effective patents in protecting firm intellectual assets from market-destructive appropriations). See also Levin et al., supra note 47.

⁹⁶ See, e.g., Reichman, *Legal Hybrids*, supra note 12, at 2478-80.

⁹⁷ See, e.g., Bronwyn H. Hall and Rosemarie H. Ziedonis, *The Patent Paradox Revisited: An Empirical Study of Patenting in the US Semiconductor Industry, 1979-95*, 32 *Rand J. Econ.* 101 (May 2000), Table B.2 (showing the pattern of patenting in the semiconductor industry over this period).

⁹⁸ See, e.g., Suzanne Scotchmer, *Standing on the Shoulders of Giants: Cumulative Innovation and the Patent Law*, 5 *J. Econ. Persp.* 29 (1991) (discussing cumulative system technologies); Robert P. Merges & Richard Nelson, *On the Complex Economics of Patent Scope*, 90 *Colum. L. Rev.* 839 (1990) (same); Hall & Ziedonis, supra note 97, at 3 (characterizing semiconductor industry as cumulative system technology and emphasizing importance of cross-licensing in this industry); Deepak Somaya & David J. Teece, *Combining Patented Inventions in Multi-Invention Products: Transactional Challenges & Organizational Choices*, draft dated Aug. 24, 2001 (on file with the authors). See also Risberg, supra note 85, at 249 (noting widespread licensing in semiconductor industry).

⁹⁹ See, e.g., Risberg, supra note 85, at 247, n. 29; Kasch, supra note 84, at 96-98 (discussing second sourcing).

¹⁰⁰ See, e.g., Hall & Ziedonis, supra note 97, at 2. See also Goldberg, supra note 82, at 330.

¹⁰¹ See, e.g., Risberg, supra note 85, at 267. As patents grew progressively stronger in the 1980’s, chip firms increased the rate of their patenting. See *id.* at 267-79; Hall & Ziedonis, supra note 97, at 4.

¹⁰² See, e.g., H. Rep. No. 98-781, 98th Cong., 2nd Sess. at 3-4 (May 15, 1984).

While the U.S. semiconductor industry thrived for years under these conditions, the life cycle pattern of chip products was so disturbed during the late 1970's and early 1980's that leading chip producers sought legislative help. Several factors contributed to this disturbance. One was a steep rise in the costs of developing and marketing new chips that had to be recouped.¹⁰³ Second, advances in manufacturing technologies reduced the costs and time required to make exact or near-exact competing chips, thereby shortening considerably the lead-time innovators could expect and reducing the costs of copying. Third, American firms were losing out to foreign—and in particular, to Japanese—competitors, raising the specter of a diminished U.S. presence in this very significant sector of the national and global economy (with potentially serious national security consequences).¹⁰⁴

B. Copyright or Sui Generis Protection for Chip Designs?

Intel Corp. initially sought to combat “chip piracy” with copyright law. Intel obtained copyright registration certificates for drawings of chip circuitry.¹⁰⁵ It then sought to register masks (that is, stencils used in manufacturing chips) and chips as derivative works of the drawings. This would help them claim that manufacturers of identical or near-identical chips were infringing copyrights in protected masks and/or chips. Intel's strategy was derailed when the U.S. Copyright Office rejected Intel's application to register masks because of their utilitarian function in the chip manufacturing process. Although Intel sued the Register of Copyright to compel registration of the masks,¹⁰⁶ it soon dropped the litigation and turned to Congress for legislative relief.¹⁰⁷

Intel's second strategy was also based on copyright. It asked Congress to amend copyright law to add “mask works” to the subject matter of copyright.¹⁰⁸ Intel argued that innovative chip designs, like literary works, were very expensive to develop and very cheap to copy, and unless the law intervened to stop rapid, cheap copying, innovators would be unable to recoup their R&D expenses and justify further investments in semiconductor innovation.¹⁰⁹ Contemporaneously, Congressional Commission was offering a very similar argument in support of copyright protection for computer

¹⁰³ See, e.g., Kasch, *supra* note 84, at 78-79.

¹⁰⁴ See, e.g., *id.* at 79. See also Raskind, *supra* note 89, at 385, 413-15.

¹⁰⁵ See Kasch, *supra* note 84, at 80.

¹⁰⁶ *Id.* See, e.g., Statement of Dorothy Schrader, Associate Registrar of Copyrights (cited hereinafter as “Schrader Statement”), House Hearings, *supra* note 83, at 88, n. 10. Apparently one copyright infringement suit was brought in the Northern District of California on a derivative mask copyright claim. *Id.*

¹⁰⁷ Stern gives a chronology of the legislative activity on the chip bills in Stern, *supra* note 87, in Appendix B. He reports that H.R. 14293, 95th Cong., 2d Sess. (1978) was the first bill introduced in Congress to protect chip designs through copyright law. The same bill was reintroduced the next session as H.R. 1007, 96th Cong., 1st Sess. (1979), and hearings were held on it, but no action was taken. Similar bills were introduced in the 97th Congress, but it was not until the 98th Congress that there was sufficient consensus on semiconductor chip protection for the legislation move forward and pass. See Stern, *supra* note 87, at 493-95.

¹⁰⁸ See Kasch, *supra* note 84, at 80.

¹⁰⁹ See, e.g., Dunlap Statement, *supra* note 83.

programs (another easily copied utilitarian information technology product that was designed to operate the circuits of semiconductor chips),¹¹⁰ and Congress acted on this recommendation.¹¹¹

During the first set of legislative hearings on the chip protection bills, some industry witnesses expressed concern about the use of copyright for chips or mask works because copyright's fair use doctrine seemed too uncertain a basis for ensuring that the common and competitively healthy industry practice of reverse engineering would be able to continue.¹¹² An explicit reverse engineering privilege was added to a later bill, but it only allowed reproducing a chip design for study and analysis and did not expressly allow reverse engineers to use what the engineer had learned in the course of reverse engineering in designing a new chip.¹¹³ Industry representatives pointed out that in order to comply with second-source "form, fit, and function" compatibility requirements, the chips resulting from reverse engineering would likely be similar to the chips being reverse engineered, although not in a competitively harmful way.¹¹⁴

Lack of industry consensus stalled enactment of chip protection bills until 1983. By that time, a fairly large number of compromise provisions had been added to the bills to satisfy various semiconductor industry concerns.¹¹⁵ Yet those compromises so deviated from traditional copyright rules that a new and different kind of opposition arose.¹¹⁶ As a representative of the Association of American Publishers explained at a 1983 hearing:

[T]he AAP is not questioning the creativity, skill, labor, or investment of chip designers, or their need for and entitlement to appropriate protection...Our concern lies...with the fundamental departures from the copyright system that accompany the proposal, e.g., the extension of Copyright Act protection to utilitarian objects that, it is acknowledged, may not be 'writings' under the Constitution...; the limitations on remedies against infringement and the extension of compulsory licensing; and most notably, the limitation imposed on the duration of this particular

¹¹⁰ See National Commission on New Technological Uses of Copyrighted Works, Final Report 12-13 (1978) (cited hereinafter as "CONTU Report"). But see Pamela Samuelson, *Creating a New Kind of Intellectual Property Law: Applying the Lessons of the Chip Law to Computer Programs*, 70 Minn. L. Rev. 471, 504-06 (1985) (arguing that the Congressional rationale for protecting chip designs by means of a sui generis law suggested that computer programs should also have been protected through a sui generis law).

¹¹¹ See Pub. L. No. 96-517, 94 Stat. 3015, 3028 (1980).

¹¹² See, e.g., Kasch, *supra* note 84, at 81 (reporting sharp industry divide at first hearing on chip legislation).

¹¹³ *Id.* at 82.

¹¹⁴ See, e.g., Brown, *supra* note 89, at 998-99; Stern, *supra* note 87, at secs. 1.2, 1.3, 5.5.

¹¹⁵ See, e.g., Kasch, *supra* note 84, at 81.

¹¹⁶ See, e.g., S. 1201, 98th Cong., 1st Sess. (1983), discussed Schrader Statement, *supra* note 106, at 128-33 (comparing the main features of the sui generis and copyright bills).

class, and the distortion of the fair use doctrine to accommodate reverse engineering.¹¹⁷

It would be better, he argued, to develop “sui generis” (of its own kind) legislation to protect semiconductor chip designs—which is what Congress ultimately did.¹¹⁸

Even so, the SCPA regime resembles copyright to a significant degree.¹¹⁹ One conceptual holdover from Intel’s copyright strategy was the subject matter chosen for SCPA protection, namely, “mask works.”¹²⁰ As with copyright, works must be “original” to qualify for protection.¹²¹ Rights attach automatically by operation of law, but registration with the Copyright Office brings benefits unavailable to non-registrants.¹²² The legislative history demonstrates that copyright-like concepts of substantial similarity and substantial identity were to be used in judging infringement of SCPA rights.¹²³ And SCPA relies, as copyright does, on a grant of exclusive rights to control reproductions and distributions of products embodying the protected work.¹²⁴

A notably sui generis feature of the SCPA¹²⁵ is its reverse engineering provision:

¹¹⁷ See Testimony of Jon A. Baumgarten, Copyright Counsel, Association of American Publishers, House Hearings, supra note 83, at 11-12. See also id. at 12, n. 2 (expressing doubt that reverse engineering would be fair use under traditional principles of that law).

¹¹⁸ Id. at 11.

¹¹⁹ See, e.g., Reichman, *Legal Hybrids*, supra note 12, at 2478-79 (discussing similarities between SCPA and copyright).

¹²⁰ 17 U.S.C. sec. 902. In retrospect, it would have been preferable for the subject matter of SCPA protection to be the layout, design, or topography of integrated circuits. Subsequent legislation in other countries has chosen the topography of integrated circuits as its subject matter. See, e.g., Council Directive of 16 December 1986 on the Legal Protection of Topographies of Semiconductor Products, 87/54/EEC, 30 O.J. L24 at 36 (Jan. 27, 1987) (cited hereinafter as “Council Directive”). A serious disadvantage of “mask works” as the protected subject matter under SCPA is that its technology-specific nature meant that SCPA would become obsolete if chip production moved beyond use of masks in the manufacturing process—as indeed occurred. See Goldberg, supra note 82, at 333.

¹²¹ See 17 U.S.C. sec. 902(b)(1). The SCPA denies protection to chip designs that are “staple, commonplace, or familiar in the semiconductor industry, or variations of such designs in such a way that, considered as a whole, is not original.” Id. at 902(b)(2). However, Congress offered very little guidance about the quantum of originality required for SCPA protection or how much difference must exist between the second comer’s and the innovator’s chips. See, e.g., Brown, supra note 89, at 991-92; Risberg, supra note 85, at 262.

¹²² 17 U.S.C. sec. 908 (rights under SCPA terminate unless the chip design is registered within two years). See also 17 U.S.C. sec. 412 (right to statutory damage awards and to recovery of attorney fees depends on prompt registration of copyright claims with the Copyright Office).

¹²³ See, e.g., H. R. Rep. No. 781, 98th Cong., 2d Sess., at 18, 22 (anticipating use of copyright-like concepts of substantial similarity and substantial identity in infringement decisions). A second comer cannot, however, hope to make a workable compatible chip merely by making minor variations on an innovative chip design in order to avoid infringement. See Brown, supra note 89, at 998.

¹²⁴ Compare 17 U.S.C. sec. 905 (SCPA’s exclusive rights provision) and 17 U.S.C. sec. 106 (copyright’s exclusive rights provision). One very significant difference between the exclusive rights provision of the SCPA and that of copyright is that the former does not include a derivative work right.

¹²⁵ The SCPA contains a number of novel and specially tailored provisions apart from the reverse engineering privilege. See, e.g., Samuelson, supra note 110, at 492-501 (discussing other sui generis features of SCPA).

[I]t is not an infringement of the exclusive rights of the owner of a mask work for (1) a person to reproduce the mask work solely for the purpose of teaching, analyzing, or evaluating the concepts or techniques embodied in the mask work or the circuitry, logic flow, or organization of components used in the mask work; or (2) a person who performs the analysis or evaluation described in paragraph (1) to incorporate the results of such conduct in an original mask work which is made to be distributed.¹²⁶

Industry witnesses distinguished between “legitimate” and “illegitimate” reverse engineering:

A reverse engineering firm should be allowed to analyze the chip, draw a circuit schematic of the chip, and then layout a different pattern. This pattern could be used to fabricate a version of the semiconductor which is functionally equivalent to the original chip but has different visual patterns on it. The reverse engineering firm could then improve the performance of the chip, reduce the size of the chip, and reduce the overall manufacturing costs of the chip....¹²⁷

A “legitimate” reverse engineer would not, for example, reproduce inefficiencies or mistakes in the innovator’s layout of circuits because careful study and analysis of the chip would identify these problems. The reverse engineer would then design around them, improving the chip design in various ways.¹²⁸

The House Report on SCPA explained the impact of this and similar testimony:

Based on testimony of industry representatives that it is an established industry practice to...make photo-reproductions of the mask work in order to analyze the existing chip so as to design a second chip with the same electrical and physical performance characteristics as the existing chip (so-called ‘form, fit and function’ compatibility), and that this practice fosters fair competition and provides a frequently needed ‘second source’ for chip products, it is the intent of the Committee to permit such reproduction by competitors...[and to make illegal] mere wholesale appropriation of the work and investment in the first chip. It is the intent of the Committee to permit, under the reverse engineering limitation, the...creation of a second mask work whose layout, in substantial part, is similar to the layout of the protected mask work—if the second mask work as the product of

¹²⁶ 17 U.S.C. sec. 906(a). Similar provisions exist in other laws protecting chip designs. See, e.g., Council Directive, *supra* note 120, Art. 5(2)-(4).

¹²⁷ Dunlap Statement, *supra* note 83, at 27-28.

¹²⁸ *Id.* Some industry witnesses also sought to distinguish “legitimate” from “illegitimate” reverse engineering in terms of differences in comparative development costs and time to market, see, e.g., *id.*, at 28, 32, or in terms of the “paper trail” that a legitimate reverse engineer would create, see *id.* at 36. The legislative history gives no weight to these factors. See, e.g., House Report, *supra* note 123.

substantial study and analysis and not the mere result of plagiarism accomplished without such study or analysis.¹²⁹

One commentator observed that the SCPA “accepts copying as the industry norm of competition. The industry spokespersons, while seeking protection from piracy as they perceived it, were insistent on preserving and encouraging the industry practices of creative copying, a practice known to them as reverse engineering.”¹³⁰

C. Economic Implications of the SCPA Rules

From an economic perspective, the SCPA reverse engineering rule would seem to have three beneficial effects: First, it could enhance incentives for second comers to license the innovator’s know-how by foreclosing the very cheap option of cloning. Second, insofar as second comers decided to reverse engineer instead of licensing from the innovator, SCPA’s requirement that a second comer engage in enough forward-engineering to produce a new, original chip design would lengthen the second comer’s development time and increase the second comer’s costs, thereby giving the innovator more lead-time within which to recoup its R&D expenses and making it necessary for the second comer to price its new product much higher than if cloning was an option. Third, the forward-engineering requirement also increased the likelihood that second comers would advance the state of the art in semiconductor design because they would have no choice but to engage in serious analysis when developing a competing chip.¹³¹

In this respect, the anti-cloning rules of the SCPA resemble the anti-plug mold rules discussed in Section II. Chip cloners were no more engaged in innovation-enhancing discovery of applied industrial know-how than were plug-molders. The disparity between initial costs of development and the costs of making identical competing products was sufficiently great as to have market-destructive potential, undermining incentives to invest in innovation in the first place. The SCPA rule aimed to induce second comers to join the ranks of innovation-enhancing firms just as the anti-plug mold laws induced second comers to engage in conventional reverse engineering that might advance the state of the art. Table 2 illustrates our economic assessment of the SCPA rules.

Table 2
Social Calculus of Reverse Engineering in the Chip Industry Pre- and Post-SCPA

¹²⁹ Id. at 22.

¹³⁰ Raskind, *supra* note 89, at 391. Shortly after the enactment of SCPA, Professor Raskind predicted that “[w]hen Congress introduced the concept of ‘reverse engineering’ as a limitation on the rights of an owner of protected industrial intellectual property in the Semiconductor Chip Protection Act of 1984 (‘the Chip Act’), it effected an innovation in the law of intellectual property that has ramifications wider and deeper than the Chip Act itself.” Id. at 385. As Section IV-A will show, this prediction has proved accurate.

¹³¹ See, e.g., Ted O’Donoghue & Suzanne Scotchmer, & Jacques-Francois Thisse, *Patent Breadth, Patent Life and the Pace of Technological Progress*, 7 J. Econ. & Mgmt. Strategy 1 (Spring 1998)(discussing the effective life of intellectual property protection in rapidly evolving sequential technologies and how the breadth of protection interacts with this).

Incentives to innovate	Pre-SCPA too low	Post-SCPA higher
Price	lower (short run)	higher
Follow-on innovation	lower (too low?)	higher
Duplicated costs	lower	higher (but avoidable by licensing)

D. Post-SCPA Developments

The failure of one second comer to do a sufficient amount of forward-engineering doomed its reverse engineering defense in the one reported judicial decision under SCPA, *Brooktree Corp. v. Advanced Micro Devices, Inc.*¹³² AMD produced a prodigious paper trail in support of its reverse engineering defense and pointed to the considerable time and expense it had spent on developing a chip compatible with the Brooktree chip.¹³³ It also emphasized many differences in the layout of its chip and Brooktree's.¹³⁴ However, the evidence showed that under pressure of an upcoming deadline, AMD's principal designer revisited the Brooktree chip layout and afterwards abandoned his plans for a six or eight transistor core cell design in favor of the same ten transistor design arrangement used in Brooktree's chip.¹³⁵ The Court of Appeals concluded that a reasonable jury "could have decided that AMD's paper trail, insofar as it related to the SRAM cell, related entirely to AMD's failures, and that as soon as the Brooktree chip was correctly deciphered by reverse engineering, AMD did not create its own design but copied the Brooktree design."¹³⁶ While AMD surely made a far greater investment in engineering than the cloning firms that the SCPA was principally aimed at, AMD did not, as SCPA required, develop its own original design of a key portion of the Brooktree chip, and hence, it was held liable for infringement of the SCPA right.

One way to interpret the scarcity of litigation under SCPA is as a sign that the law successfully deterred chip piracy, although most legal commentators have inferred from this that the SCPA is unimportant.¹³⁷ Some assert that SCPA was badly drafted, claiming that it is technologically obsolete or provides too thin a scope of legal protection.¹³⁸ Others explain SCPA's unimportance in terms of subsequent legal developments, such as the renewed importance of patents in the

¹³² 977 F.2d 1555 (Fed. Cir. 1992).

¹³³ *Id.* at 1566-67.

¹³⁴ *Id.* at 1566-67.

¹³⁵ *Id.* at 1567.

¹³⁶ *Id.* at 1568.

¹³⁷ See, e.g., Risberg, *supra* note 85, at 245 (describing SCPA as "a largely untested, if not impotent, piece of legislation"); Kasch, *supra* note 84, at 72 (SCPA of "largely academic interest").

¹³⁸ See, e.g., Goldberg, *supra* note 82, at 333-35 (making both complaints).

aftermath of creation of the Federal Circuit or a rise in second-source licensing agreements between pioneers and follow-on innovators.¹³⁹ Still others regard changes in the industry and technology, such as further miniaturization of chip circuitry, advances in process technology, mass customization of chip designs, and the increasing sophistication of CAD/CAM programs for generating that alternative layouts, as having rendered infeasible the kind of copying that gave rise to the SCPA.¹⁴⁰

One indication of some continuing interest among chip producers in SCPA rights can be found in the number of registered chip designs with the U.S. Copyright Office and elsewhere.¹⁴¹ Legal protection for the layout of integrated circuits was also deemed important enough internationally to warrant its inclusion in the TRIPS Agreement.¹⁴² TRIPS incorporates by reference a number of provisions of an earlier treaty on the legal protection for the layout of integrated circuits, including a reverse engineering privilege closely modeled on the SCPA rule.¹⁴³ The semiconductor chip industry, as a consequence, is the only industry

¹³⁹ See, e.g., See Hall & Ziedonis, supra note 97, at 4 (attributing a substantial increase in patenting in the semiconductor industry to a strong “pro-patent” shift in U.S. legal environment).

¹⁴⁰ See, e.g., Risberg, supra note 85, at 263-72; Kasch, supra note 84, at 73, 103. Kasch predicted that further changes in technology might cause the SCPA’s anti-cloning protection to have renewed importance in the future. Id. at 103-04.

¹⁴¹ See, e.g., Risberg, supra note 84, at 243, n. 16. See also Andy Y. Sun, *From Pirate King to Jungle King: Transformation of Taiwan’s Intellectual Property Protection*, 9 Fordham Intell. Prop., Media & Ent. L.J. 67, 138-39 (1998) (reporting a substantial number of chip protection registrations in Taiwan). Professor Rosemarie Ziedonis has collected data about chip registrations in the U.S. She reports that between 1985 and 1997, there were 6834 chip registrations with the U.S. Copyright Office, including 637 in 1996 and 471 in 1997. Ironically, Intel is noticeably absent from the list of U.S. registrants. Email communication from Rosemarie Ziedonis, May 18, 2001 (on file with author).

¹⁴² See TRIPS, supra note 11, arts. 35-38. While on the subject of international protection for chip designs, it is worth noting that the United States made what in retrospect can be seen as a tactical mistake in its approach to gaining international acceptance of SCPA-like protection. Rather than adopt a national treatment-based approach, as most international treaties do, under which chip designs of foreign producers would be protected under U.S. law regardless of whether their nations protected chip designs, SCPA adopted a material reciprocity approach under which the chips of foreign nationals would not be protected under U.S. law unless their nations had adopted “equivalent” laws. SCPA established a process under which U.S. officials could judge whether other nations had adopted sufficient laws. See 17 U.S.C. sec. 914. Although the U.S. was able to persuade many other nations to adopt chip protection laws, see, e.g., Council Directive, supra note 120, and Stern, supra note 87, chap. 10, there has been some resentment among intellectual property professionals in other countries about the U.S. reciprocity approach. This also came back to haunt the U.S. when the European Commission decided to adopt a new form of legal protection to the contents of databases on a material reciprocity basis. This was of concern to U.S. database developers because of their substantial market share in the European market. See, e.g., J.H. Reichman & Pamela Samuelson, *Intellectual Property Rights in Data?*, 50 Vand. L. Rev. 51, 96-97 (1997)(discussing the European database directive and the initial U.S. response to it).

¹⁴³ See TRIPS, supra note 11, art. 35 (incorporating various provisions of the Treaty on Intellectual Property in Respect of Integrated Circuits, commonly known as the “Washington Treaty,” including Art. 6(2)’s reverse engineering privilege). The U.S. and other developed chip-producing countries objected to some provisions of the Washington treaty and refused to sign it. See, e.g., Goldberg, supra note 82, at 335-36 (discussing various dissatisfactions with the Washington Treaty). To “fix” the perceived weaknesses in the Washington Treaty, the TRIPS Agreement added some new substantive requirements as minimum standards for protecting the layouts of integrated circuits. See TRIPS, supra note 11, arts. 36-37.

whose reverse engineering activities are expressly protected in an international intellectual property treaty.

In the years since the SCPA enactment, the semiconductor industry has enjoyed very considerable growth and U.S. firms dominate a much-enlarged global chip market.¹⁴⁴ Interestingly, in the post-SCPA era, there has been a partial bifurcation of design and fabrication components of the chip industry.¹⁴⁵ That is, some firms now design chip layouts and other firms fabricate chips of that design. This has been accompanied by a rise in the rate of patenting in this industry and a more aggressive enforcement of patent rights, especially by the design firms.¹⁴⁶ From an economic perspective, if SCPA contributed to the rise in second-source licensing agreements (and it probably did) and if it contributed to the cessation of cloning of innovative chip designs, it has had a beneficial effect on this market.

IV. Reverse Engineering in the Computer Software Industry

Reverse engineering is as standard an industry practice in the computer software industry as in the traditional manufacturing and semiconductor industries.¹⁴⁷ Yet, for much of the past two decades, the legality of two common forms of reverse engineering of programs, namely, decompilation and disassembly of object code,¹⁴⁸ has been challenged on trade secret, copyright, patent, and contract law theories. This section will consider the doctrinal and policy issues that have dominated the legal debate and the economic effects of decompilation and disassembly, particularly when undertaken for purposes of making an interoperable program. The legal and economic considerations pertinent to the decompilation/disassembly debate overlap to some degree but diverge in some respects. The economic case for interoperability may not be as open and shut as some legal commentators have suggested, although, we conclude, on balance, interoperability has more beneficial than harmful consequences. Hence, the legal rule permitting these forms of reverse engineering of programs to achieve interoperability is sound.

A. Reverse Engineering of Software And Copyright Law

Commercial developers of computer programs generally distribute software in object code form. They do so for two principal reasons: first, because users mainly want

¹⁴⁴ See, e.g., Risberg, *supra* note 85, at 273-76; Hall & Ziedonis, *supra* note 97. See generally U.S. DEPT. OF COMMERCE, *THE EMERGING DIGITAL ECONOMY*, Appendix 1 (1998) (reporting on high growth of information technology industries, including the semiconductor industry, in the U.S.).

¹⁴⁵ See, e.g., Hall & Ziedonis, *supra* note 97, at 4-5.

¹⁴⁶ See, e.g., *id.* One would expect design firms to rely not only on patents (as they apparently do, see *id.* at 5), but also on the legal protection SCPA provides against copying of chip layouts, although the latter has not been documented.

¹⁴⁷ See, e.g., Andrew Johnson-Laird, *Reverse Engineering of Software: Separating Legal Mythology From Actual Technology*, 5 *Software L.J.* 331, 354 (1992) (“Reverse engineering is practiced by all programmers....”).

¹⁴⁸ See OFFICE OF TECHNOLOGY ASSESSMENT, U.S. CONGRESS, *FINDING A BALANCE: COMPUTER SOFTWARE, INTELLECTUAL PROPERTY, AND THE CHALLENGE OF TECHNOLOGICAL CHANGE* 7 (1992) (explaining disassembly and decompilation).

the functionality that object code forms of programs provide, not to read the code, and second, because the developers want to maintain source code forms of their products and other human-readable documentation as trade secrets.¹⁴⁹ Decompilation or disassembly of object code provides a way for reverse engineers to “work[] backwards from object code to produce a simulcrum of the original source code.”¹⁵⁰ From this approximation of source code, reverse engineers can discern or deduce internal design details of the program, such as information necessary to enable the development of a program that will interoperate with the decompiled or disassembled program. Lawyers for some major software producers argued that decompilation and disassembly should be illegal as a matter of copyright and trade secrecy law because decompilation and disassembly inevitably require a reverse engineer to make unauthorized (and arguably infringing) copies of the program, and this infringement makes decompilation and disassembly an improper means to obtain trade secrets.¹⁵¹

¹⁴⁹ See, e.g., Jessica Litman, *Copyright and Information Policy*, 55 *Law & Contemp. Probs.* 185, 196-201 (1992) (discussing strategies of software industry lawyers for maintaining program internals as trade secrets). See also Reichman, *Programs as Know-How*, supra note 14, at 701 (describing nondisclosure of program internals as a business imperative, although concluding that second comers ought to be able to reverse engineer object code).

¹⁵⁰ See Cohen & Lemley, supra note 7, at 16, n. 52 (2001). See also Litman, supra note 149, at 197-98: Decompilation is a species of reverse engineering that involves translating the object code into a human-readable form, or ‘pseudo source code,’ largely through trial and error. Part of the decompilation process can be computer-assisted: there are, for example, disassembly programs that will translate the object code into an intermediate assembly language form that is more decipherable to skilled readers. Other computer software can assist the developer in the laborious process of translating the assembly language into pseudo source code form. The decompilation process does not generate source code as originally written but rather a plausible reconstruction of what portions of the original source code could have been. Of course, the produce of such reverse engineering will include not only the parts of the program that were compiled into object code in the first instance; the English comments and descriptions were never compiled and cannot be retrieved or recreated. Pseudo source code is nonetheless a useful tool that can assist a software developer in analyzing how a computer program works.

¹⁵¹ See, e.g., Allen R. Grogan, *Decompilation and Disassembly: Undoing Software Protection*, *Computer Law.*, Jan. 1984, at 1. Grogan’s argument wove trade secret, copyright, and contract together in a tight mesh. He asserted that reverse engineering of object code by decompilation or disassembly was trade secret misappropriation because the reverse engineer used improper means to obtain the trade secret information embedded in the program by making unauthorized copies of the program in the course of the reverse engineering process (thereby infringing copyright) and/or by violating anti-reverse engineering clauses of shrinkwrap license contracts under which they were distributed. At that time, there was much uncertainty about the enforceability of shrinkwrap licenses as a matter of contract law and about the enforceability of anti-reverse engineering clauses in particular. See infra Section IV-D for further discussion of the shrinkwrap license issues pertaining to reverse engineering of software. For similar arguments, see also Anthony L. Clapes, *Confessions of an Amicus Curiae: Technophobia, Law and Creativity in the Digital Arts*, 19 *U. Dayton L. Rev.* 903 (1994); Duncan Davidson, *Common Law, Uncommon Software*, 47 *U. Pitt. L. Rev.* 1037 (1985); Arthur R. Miller, *Copyright Protection for Computer Programs, Databases and Computer-Generated Works: Is Anything New Since CONTU?*, 106 *Harv. L. Rev.* 977 (1993).

However, the predominant view among legal commentators supports a right to reverse engineer software under copyright law. See, e.g., JONATHAN BAND & MASANOBU KATOH, *INTERFACES ON TRIAL: INTELLECTUAL PROPERTY AND INTEROPERABILITY IN THE GLOBAL SOFTWARE INDUSTRY* 167-225 (1995); *Brief Amicus Curiae of Eleven Copyright Professors*, *Sega Enter. Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), published in 33 *Jurimetrics J.* 147 (1992); Julie

The principal U.S. case testing this legal theory was *Sega Enterprises Ltd. v. Accolade, Inc.*¹⁵² Accolade, a small U.S. computer game company, disassembled Sega game programs in order to get information necessary to make its games compatible with the Sega Genesis console. Accolade then sold its independently developed games in competition with those made by Sega and third-party developers licensed by Sega. Accolade raised a fair use defense to Sega's claims that the disassembly copies were infringing.¹⁵³ The Ninth Circuit gave little weight to the commercial purpose of Accolade's copying because it regarded the copying as having been done "solely in order to discover the functional requirements for compatibility with the Genesis console— aspects of Sega's programs that are not protected by copyright."¹⁵⁴ Reverse engineering was, moreover, the only way that Accolade could gain access to this information.¹⁵⁵ Although Accolade had copied the whole of Sega's programs in the course of its reverse analysis, the court discounted this because it occurred in an intermediate stage of Accolade's software development process. Although the court recognized that Accolade's games affected the market for Sega games, it did not do so in a way in which

E. Cohen, *Reverse Engineering and the Rise of Electronic Vigilantism: Intellectual Property Implications of 'Lock-out' Programs*, 68 S. Cal. L. Rev. 1091 (1995); Lawrence Graham & Richard O. Zerbe, Jr., *Economically Efficient Treatment of Computer Software: Reverse Engineering, Protection and Disclosure*, 22 Rutg. Comp. & Tech. L.J. 61 (1996); Dennis S. Karjala, *Copyright Protection of Computer Software, Reverse Engineering and Professor Miller*, 19 U. Dayton L. Rev. 975 (1994); Robert A. Kreiss, *Accessibility and Commercialization in Copyright Theory*, 43 UCLA L. Rev. 1 (1995); *LaST Frontier Conference Report on Copyright Protection for Computer Software*, 30 Jurim. J. 15 (1989); Ronald S. Laurie & Stephen M. Everett, *Protection of Trade Secrets in Object Form Software: The Case for Reverse Engineering*, Computer Law., July 1984, at 1; Lemley & McGowan, *supra* note 41; Litman, *supra* note 149; Charles R. McManis, *Intellectual Property Protection and Reverse Engineering of Computer Programs in the United States and the European Community*, 8 High Tech. L. J. 25 (1993); Reichman, *Programs As Know-How*, *supra* note 14; David A. Rice, *Sega and Beyond: A Beacon for Fair Use Analysis...At Least As Far As It Goes*, 19 U. Dayton L. Rev. 1131 (1994); Pamela Samuelson, *Fair Use for Computer Programs and Other Copyrightable Works in Digital Form: The Implications of Sony, Galoob, and Sega*, 1 J. Intell. Prop. L. 49 (1993); Manifesto, *supra* note 14; Timothy Teter, Note, *Merger and the Machines: An Analysis of the Pro-Compatibility Trend in Computer Software Copyright Cases*, 45 Stan. L. Rev. 1061 (1993).

¹⁵² 977 F.2d 1510 (9th Cir. 1992). *Sega v. Accolade* was not the first appellate court decision on whether decompilation or disassembly of a program could in appropriate circumstances be fair use. *Atari Games Corp. v. Nintendo of Am., Inc.*, 975 F.2d 832 (Fed. Cir. 1992) was decided shortly before the Ninth Circuit decision. The *Atari v. Nintendo* analysis of fair use is similar to the Ninth Circuit's analysis, although somewhat less extensive. In *Atari Games*, the fair use issue was complicated by the fact that Atari Games' lawyers lied to the U.S. Copyright Office to get the registration copy of Nintendo source code so that the firm's engineers could use it to finalize the development of compatible games. *Id.* at 837. The Federal Circuit ruled that the initial decompilation copying was fair use. *Id.* at 843.

¹⁵³ Courts generally consider four factors in considering whether a use is fair: the purpose of the defendant's use of the work, the nature of the copyrighted work, the amount and substantiality of the defendant's appropriation, and the harm or potential harm to the market if the defendant's use is permitted. See 17 U.S.C. sec. 107. It is interesting to note that Sega relied in part on the legislative history of SCPA in which some witnesses had expressed doubt that reverse engineering could be fair use as a matter of copyright law. See, e.g., Baumgarten Testimony, *supra* note 117 and accompanying text; *Sega*, 977 F.2d at 1517.

¹⁵⁴ 977 F.2d at 1522.

¹⁵⁵ *Id.* The unprotected aspects of most copyrighted works, the court pointed out, "are readily accessible to the human eye....Computer programs, however, are typically distributed for public use in object code form, embedded in a silicon chip, or on a floppy disk. For that reason, humans often cannot gain access to the unprotected ideas and functional concepts without disassembling that code." *Id.* at 1525.

copyright is concerned.¹⁵⁶ Accolade's decompilation "led to an increase in the number of independently designed video game programs offered for use with the Genesis console. It is precisely this growth in creative expression...that the Copyright Act was intended to promote."¹⁵⁷ An important policy consideration derived from the court's recognition that if it ruled that disassembling computer programs was unlawful, this would confer on Sega "a de facto monopoly over [the unprotected] ideas and functional concepts [in the program]."¹⁵⁸ To get a monopoly on such ideas and functional concepts, a creator needs to seek patent protection.¹⁵⁹

Still, the court did not give a green light to all reverse engineering of program code, but only to that undertaken for a "legitimate purpose," such as to gain access to the functional specifications necessary to make a compatible program, and then only if it "provides the only means of access to those elements of the code that are not protected by copyright."¹⁶⁰

Sega v. Accolade has been followed in virtually all subsequent cases,¹⁶¹ as well as being widely praised by legal commentators¹⁶² and being consistent with the rules of other nations.¹⁶³ The Ninth Circuit Court of Appeals recently reaffirmed this ruling in

¹⁵⁶ Id. Copyright law is concerned with infringing copies that compete with the author's works, not with competition on the merits among noninfringing works.

¹⁵⁷ Id. at 1523-24.

¹⁵⁸ Id. at 1527.

¹⁵⁹ Id. at 1526.

¹⁶⁰ Id. at 1518.

¹⁶¹ See, e.g., *DSC Communications Corp. v. DGI Techs., Inc.*, 81 F.3d 597, 601 (5th Cir. 1996); *Bateman v. Mnemonics, Inc.*, 79 F.3d 1532, 1539 n.18 (11th Cir. 1996); *Mitel, Inc. v. Iqtel, Inc.*, 896 F.Supp. 1050, 1056-57 (D. Colo. 1995), *aff'd* on other grounds, 124 F.3d 1366 (10th Cir. 1997).

¹⁶² See, e.g., sources cited *supra* note 151.

¹⁶³ The decompilation for interoperability issue was addressed legislatively in the European Union. . . In 1989, the European Commission published a proposed directive on the legal protection of computer programs to harmonize the laws of member states of the EU; it did not contain a decompilation or interoperability exception. See Proposal for a Council Directive on the Legal Protection of Computer Programs, 1989 O.J. (C91) 4. U.S. trade negotiators and representatives of some U.S. computer companies argued that this was as it should be. See, e.g., Victor Siber, *Interpreting Reverse Engineering Law*, IEEE Software 4 (July 1990) (explaining IBM's position against reverse engineering of software). See also Band & Katoh, *supra* note xx, at 229-241 (discussing U.S. industry lobbying and government officials' positions about the software directive); Thomas C. Vinje, *The Legislative History of the EC Software Directive*, in *A HANDBOOK OF EUROPEAN SOFTWARE LAW* (Michael Lehmann & Colin Tapper, eds. 1993). However, the Commission's competition directorate worried that unless the directive allowed decompilation for purposes of developing interoperable programs, European software developers would be at a serious disadvantage in the global software market. See, e.g., Pamela Samuelson, *Comparing U.S. and E.C. Copyright Protection For Computer Programs: Are They More Different Than They Seem?*, 13 J. Law & Comm. 279, 287-88 (1994) (discussing the concerns of the European Commission's competition directorate about the software directive).

In a response to these concerns, the final Directive contained a decompilation-for-interoperability privilege akin to that in *Sega v. Accolade*. See Council Directive 91/250 on the Legal Protection of Computer Programs, 1991 O.J. (L122) 42 (cited hereinafter as "Software Directive"). See, e.g., BRIDGET CZARNOTA & ROBERT J. HART, *LEGAL PROTECTION OF COMPUTER PROGRAMS IN EUROPE: A GUIDE TO THE EC DIRECTIVE* (1991); *HANDBOOK OF EUROPEAN SOFTWARE LAW*, *supra*. Achieving interoperability would seem to be the only legitimate purpose for decompilation under the European Software Directive. *Sega v. Accolade*, by contrast, contemplates that there may be other legitimate purposes for decompilation,

Sony Computer Entertainment, Inc. v. Connectix Corp.,¹⁶⁴ in which Connectix disassembled Sony code to make a substitute for the Sony platform, not compatible games. That is, Connectix developed emulation software that allowed owners of Apple computers to play Sony Playstation games on their iMacs, rather than having to buy the Sony platform. The Court of Appeals perceived no legal difference in the decompilation-for-interoperability considerations pertinent to development of competing platforms than as to games.

Although those who argued that decompilation should be illegal predicted grievous harm to the software industry if this form of reverse engineering was deemed lawful, American software industry continues to be strong.¹⁶⁵

B. The Economics of Interoperability and Software Reverse Engineering

Sega v. Accolade and *Sony v. Connectix* demonstrate that reverse engineering is undertaken for different reasons in the software industry than in other industrial contexts studied in this article. The main reason to reverse engineer in manufacturing industries is

although not saying what they might be. Error correction and detecting infringement are two other legitimate reasons to decompile programs. See, e.g., *E.F. Johnson Co. v. Uniden Corp. of Am.*, 623 F. Supp. 1485 (D. Minn. 1985) (decompilation to detect infringement); Samuelson, *supra* note 151, at 289 n. 59.

The European Software Directive also limits follow-on uses that can be made of information obtained in the course of decompilation. See European Software Directive, *supra*, Art. 6(2). One cannot, for example, publish information learned during reverse engineering. This puts at risk authors of books such as ANDREW SCHULMAN, DAVID MAXEY, & MATT PIETREK, UNDOCUMENTED WINDOWS: A PROGRAMMERS' GUIDE TO RESERVED MICROSOFT WINDOWS API FUNCTIONS (1992). Under Article 6(2), European decompilers are at risk if they try to recoup their reverse engineering expenses by licensing the information it learned in the course of its reverse engineering efforts. The Official Commentary to the Software Directive asserts that Article 6(2)(b) "prevents the publication or trafficking in information by those who have decompiled existing programs, since it would be inequitable to impose conditions on the decompiler but allows others access to the information which he had then made public." Czarnota & Hart, *supra*, at 81. The European software directive, in essence, converts copyright into a trade secrecy law as to internal elements of programs.

Europe's adoption of a decompilation-for-interoperability privilege and the *Sega v. Accolade* decision in the U.S. did not end the international debate about decompilation. U.S. officials continued to insist that decompilation should be unlawful. In the mid-1990's, for example, Japan was considering a proposal to amend its copyright law to allow reverse engineering of software, but dropped the proposal after intense pressure from U.S. officials. See, e.g., T.R. Reid, A Software Fight's Blurred Battle Lines: U.S. Companies Are On Both Sides as Japan Considers Copyright Law Changes, *Wash. Post*, Jan. 11, 1994, at D1. However, some Japanese commentators believe that Japanese copyright law would permit decompilation for interoperability purposes. See, e.g., Band & Katoh, *supra* note 151, at 294-97; Keiji Sugiyama, *Reverse Engineering and Other Issues of Software Protection in Japan*, 11 *Eur. Intell. Prop. Rev.* 395 (1991). A number of jurisdictions have, however, adopted similar decompilation-for-interoperability exceptions to the European software directive. See, e.g., Band & Katoh, *supra* note 151, at 271-82.

¹⁶⁴ 203 F.3d 596 (9th Cir. 2000)(reaffirming and extending *Sega v. Accolade* as to defendant who reverse engineered Sony games in order to develop software to enable users to play Sony games on Apple computers);

¹⁶⁵ See, e.g., Contributions of the Packaged Software Industry to the Global Economy (April 1999) (study conducted by Pricewaterhouse Coopers, commissioned by the Business Software Alliance).

to make directly competing stand-alone products.¹⁶⁶ While this is not unknown as to software,¹⁶⁷ reverse engineering of computer programs is generally so difficult, time-consuming, and resource-intensive that it is not an efficient way to develop competing programs.¹⁶⁸ The principal motivation for reverse engineering in the software industry, as reflected in *Sega v. Accolade* and other litigated cases, has been to develop a program that will interoperate with another program or with a hardware system.¹⁶⁹ Our economic assessment of reverse engineering in software industry will consequently focus on this purpose. As will become apparent, the economics of interoperability are more complicated than some previous commentators have suggested.¹⁷⁰

¹⁶⁶ See *supra* Section II-B.

¹⁶⁷ See, e.g., *Secure Services Techn., Inc. v. Time & Space Processing, Inc.*, 722 F.Supp. 1354 (E.D. Va. 1989)(reverse engineering of embedded software in secure facsimile machines for purposes of making competing, compatible facsimile machine); *Alcatel USA, Inc. v. DGI Techns., Inc.*, 166 F.3d 772 (5th Cir. 1999)(reverse engineering of telecommunications switching software to make competing product). Notice that both of these examples involve embedded software in a traditional manufactured product.

¹⁶⁸ See, e.g., Andrew Johnson-Laird, *Software Reverse-Engineering in the Real World*, 19 U. Dayton L. Rev. 843, 843 (1994). Johnson-Laird points out that reverse engineering does not “lay bare a program’s inner secrets. Indeed, it cannot. The inner secrets of a program, the real crown jewels, are embodied in the higher levels of abstraction material such as the source code commentary and the specification. This material never survives the process of being converted to object code.” *Id.* at 896. The reverse engineer, in other words, has to do considerable work to extract from the source code approximation those higher level abstraction concepts and information, and even more work to incorporate what he or she has learned from this analysis in a new program. This has caused other commentators to conclude that “decompilation should be regulated by the law—although not necessarily by copyright law—only if and to the extent that it permits competitors to acquire behavioral equivalence [with the target program] with only trivial effort, and therefore induces market failure.” See Manifesto, *supra* note 14, at 2392. Because the present state of decompilation technology does not permit trivial acquisition of equivalence, the Manifesto authors concluded that there is no economically sound reason to regulate decompilation. *Id.* If technological change shifted the balance and enabled rapid inexpensive copying that would be market-destructive, decompilation might need to be regulated to some degree. *Id.* at 2392-93. But see COMPUTER SCIENCE & TELECOMMUNICATIONS BOARD, NATIONAL RESEARCH COUNCIL, INTELLECTUAL PROPERTY ISSUES IN SOFTWARE 78 (1991) (quoting IBM executive as expressing concern that reverse analysis of programs could allow illegal copying of program internals that would escape easy detection).

¹⁶⁹ *Accolade*, for example, reverse engineered Sega programs in order to discover the precise details about how Sega programs exchanged data with the Sega console software so that *Accolade* could make games that would successfully interoperate with the Sega console. See *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992). *Connectix* reverse engineered the Sony PlayStation to learn enough about its interfaces to develop an emulator program that would run Sony PlayStation games. See *Sony Computer Entertainment, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000). See also *Atari Games Corp. v. Nintendo of Am, Inc.*, 975 F.2d 832 (Fed. Cir. 1992)(reverse engineering to develop games that could be played on Nintendo console); *Vault Corp. v. Quaid Software Ltd.*, 847 F.2d 255 (5th Cir. 1988)(reverse engineering of copy-protection software to make spoofing software). Reverse engineering of software has sometimes been done to develop a complementary service. See, e.g., *Allen-Myland v. IBM Corp.*, 746 F.Supp. 520 (W.D. Pa. 1990)(engineering service reverse engineered IBM software to aid in reconfiguration of leased computers for subsequent lease customers); *Hubco Data Prods. Corp. v. Management Assistance, Inc.*, 219 U.S.P.Q. 450 (D. Idaho 1983)(reverse engineering to discover portions of MAI code that blocked use of advanced features to enable reverse engineer to provide service of providing MAI customers with cheaper way to access the advanced features).

¹⁷⁰ Other legal commentators have considered the economic consequences of reverse engineering in the software industry and have concluded the economic effects are benign. See, e.g., Graham & Zerbe, *supra* note 151; Lemley & McGowan, *supra* note 41. However, those articles do not consider the systems competition issues literature below.

Before considering the role reverse engineering plays in the interoperability debate, it is first important to grasp some basic concepts about the incentives of firms to design their systems to be interoperable or non-interoperable. A system, for these purposes, consists of two complementary pieces, such as a platform (e.g., the Sega Genesis machine or Microsoft's Windows operating system program) and applications designed to run on it (e.g., Sega's Sonic Hedgehog game or Lotus 1-2-3). In the software industry context, platforms and applications are not just complementary products, they are complementary parts of a system by virtue of their conformity to interfaces necessary to achieving interoperability. Platforms are typically designed first. If an applications developer wants to make a program for a particular platform, it must have access to very precise specifications about the manner in which platform services can be requested and provided, collectively referred to as application programming interfaces (APIs).¹⁷¹ Some platform developers publish interfaces; some license them freely; and others maintain their APIs as closely held trade secrets.¹⁷²

The developer of a new platform, for example, might decide to publish its interfaces or make them available under open license terms—acts that make reverse engineering unnecessary—in order to make it easy for applications developers to adapt existing applications or make new applications for the platform. An important incentive to open interfaces is to drive demand for the new platform. Only if desirable applications are available for the platform will consumer demand for the platform skyrocket. In the 1980's, for example, IBM, then a new entrant into the personal computer (PC) market, published technical specifications for the PC and required Microsoft to broadly license the APIs to its operating system to enable applications developers to write programs for the IBM PC.¹⁷³ This resulted in “[a] large library of off-the-shelf IBM PC compatible applications software (particularly Lotus 1-2-3) [that] made the IBM PC an attractive platform.”¹⁷⁴ This allowed the IBM PC to rapidly achieve a substantial market success.¹⁷⁵

Publishing or broadly licensing interfaces can, however, be risky for platform developers, even if beneficial for consumers and competitors. Hewlett-Packard and Dell are among the makers of IBM-compatible PCs who took advantage of IBM's decision to embrace open architectures in the PC market. Consumers benefited from competition in among IBM-compatible PCs and from a wide array of applications for this standard

¹⁷¹ See Band & Katoh, *supra* note 151, at 6-7.

¹⁷² See, e.g., Cohen, *supra* note 151, at 1094; Manifesto, *supra* note 14, at 2402-03.

¹⁷³ “The IBM PC was also the first deliberately open computer architecture, a fundamental insight that shaped the future of personal computing. From the very start, Boca Roton [where IBM developed the PC] recognized that the best way to make the PC the industry standards was to publish all its technical specifications and make it easy for third parties to build add-on devices or write PC software applications, a principle that too Apple years to understand.” CHARLES H. FERGUSON & CHARLES R. MORRIS, *COMPUTER WARS : HOW THE WEST CAN WIN IN A POST-IBM WORLD* 52 (1993). Band and Katoh emphasize IBM's insistence on requiring Microsoft to broad license APIs for its operating system for the PC. Band & Kato, *supra* note 151, at 30.

¹⁷⁴ *Id.*

¹⁷⁵ See, e.g., *id.* (“in 1984 alone, IBM's PC revenues were \$4 billion”). IBM was far more proprietary about the interfaces to its mainframe computers, a market in which they long been dominant. *Id.*

system. However, IBM lost market share in the PC market in part because the openness of its PC architecture enabled cloning.¹⁷⁶

Some platform developers opt for a more restrictive approach toward access to interfaces and toward interoperability.¹⁷⁷ Firms have two principal incentives to exercise tight control over interfaces and to choose non-interoperability: first, restricting access to interfaces protects the platform against being “commoditized,” and second, it gives platform developers considerable power over applications for their platforms. Whether applications are developed in-house or by independent software developers (ISVs) who can be induced to write for the platform, the platform developer will prefer that applications be available only for its platform.¹⁷⁸ Choosing non-interoperability as a strategy is risky because if applications developers and consumers are not attracted to the system, losses can be considerable.¹⁷⁹

Network effects explain the upside potential of non-interoperable systems.¹⁸⁰ If a non-interoperable system becomes a de facto standard in the market—perhaps because of

¹⁷⁶ Id. at 31 (“By the early 1990’s, IBM sold only 23% of the IBM compatible PCs worldwide....”). IBM’s larger problem in controlling the PC market was that “in essence [it] ceded control of the microprocessor architecture to Intel and the operating system architecture to Microsoft.” Id. at 30. As IBM’s fortunes waned, Microsoft’s soared. From 1982 to 1993, “Microsoft’s annual revenues went from \$24 million to \$4.1 billion and its profits from \$3.5 million to \$794 million.” Id. at 31.

¹⁷⁷ Thomas Piraino, *Identifying Monopolists’ Illegal Conduct Under the Sherman Act*, 75 N.Y.U. L. Rev. 809, 888-89 (2000)(quoting a Microsoft manager’s internal email: “to control the APIs is to control the industry”). See also JERRY KAPLAN, *STARTUP: A SILICON VALLEY ADVENTURE* 49-50 (1994) (“our value is in the APIs” and “the real wars [in the computer industry] are over control of the APIs”).

¹⁷⁸ Since the platform developer knows its own APIs, it can easily supply them to applications programmers within the firm. Although the platform developer may also seek to attract independent applications developers to its platform, it may provide ISVs with less complete interface information and perhaps delayed access as compared with that provided within the firm. Microsoft’s practices in this regard were an important reason why the Dept. of Justice recommended breaking Microsoft into two firms, one an operating systems company and the other an applications development firm. Piraino recommends addressing this problem by ordering Microsoft to give applications programmers open access to Windows APIs. Piraino, *supra* note 177, at 888.

¹⁷⁹ See, e.g., Kelly Zito, *New Path For Sega*, S.F. Chronicle, p. E1, Aug. 12, 2001. Sega recently exited the game system market due to \$420 million in losses in 2000 on the Dreamcast system it introduced into the market in 1999. Sega’s new system met with resistance from applications developers who decided not to tailor games for it. See James Surowiecki, *Games People Play*, New Yorker, May 7, 2001, at 36.

¹⁸⁰ See, e.g., Michael L. Katz and Carl Shapiro, *Systems Competition and Network Effects*, 8 J. Econ. Persp. 93 (1994)(discussing network effects). Entrepreneur Jerry Kaplan offers this down-to-earth explanation of the phenomenon:

Creating an API is like trying to start a city on a tract of land you own. First you try to persuade applications programmers to come and build their businesses on it. This attracts users, who want to live there because of all the wonderful services and shops the programmers have built. This in turn causes more programmers to want to rent space for their businesses, to be near the customers. When this process gathers momentum, it’s impossible to stop.

Once your city is established, owning the API is like being king of the city. The king gets to make the rules: collecting tolls for entering the city, setting the taxes that the programmers and users have to pay, and taking first dibs on any prime locations (by keeping some APIs confidential for personal use.

Kaplan, *supra* note 177, at 50.

a “killer app” available only on that platform—other applications developers will have powerful incentives to develop applications for the platform on license terms favorable to the platform provider. Consumers will be increasingly eager to buy or stick with the platform as more applications become available for it. At the same time, entry of competitors is deterred because an entrant would have to enter at two levels: platform development and software development. Apple Computer and Sega are among the firms that hoped to achieve substantial market penetration with non-interoperable systems.

Even if initially successful, a non-interoperable system may lose out over time if other firms are willing to invest in development of a new system to wrest away the incumbent’s market share. Sega, for example, was a second comer to the game system market, entering after Nintendo’s Entertainment System (NES) had achieved substantial market success.¹⁸¹ Sega’s Genesis system offered some features the NES lacked, as well as certain new programs (notably one featuring a sonic hedgehog) that drew customers to the Genesis system. Later, Sega dropped out of the game system market, opting instead to develop games for other systems.¹⁸² The current market leader in the game system market is Sony’s PlayStation,¹⁸³ whose lead is about to be challenged by new entrant Microsoft’s Xbox system.¹⁸⁴ In the game system market, platform developers typically lose money on sales of consoles, making up losses on sales of games and peripherals.¹⁸⁵

In contrast to the game system market, which has been characterized by serial monopolies, Microsoft’s operating system program has become a de facto standard platform in the market for software running on personal computers, a monopoly has been durable over many years.¹⁸⁶ Over this period, Microsoft’s operating systems interfaces have become more complex and obscure, and its licensing practices as to its interfaces more restrictive. One reason the Microsoft interfaces have become more complicated is because Microsoft has often responded to innovations in the applications market by integrating them into its operating system program (by a strategy sometimes known as

¹⁸¹ See, e.g., DAVID SHEFF, *GAME OVER: HOW NINTENDO ZAPPED AN AMERICAN INDUSTRY, CAPTURED YOUR DOLLARS, AND ENSLAVED YOUR CHILDREN* 352-53 (1993)(discussing Sega’s entry into the game system market and its effort to gain market share against Nintendo’s Entertainment System).

¹⁸² See, e.g., Zitto, *supra* note 179, at E1. Sega will now concentrate on the sale of games for other platforms because this is a more profitable line of business. *Id.* at E4.

¹⁸³ *Id.* Sony has an installed base of 85 million PlayStations. *Id.*

¹⁸⁴ See, e.g., Chris Gaither, *Microsoft Delays Release of Xbox Game System by a Week*, *New York Times*, p. C15, Sept. 22, 2001. Microsoft hopes to ship 1.5 million consoles by the end of the 2001 holiday season. *Id.*

¹⁸⁵ *Id.* See also Surowiecki, *supra* note 179, at 36 (“Sony loses money on every PlayStation 2 it makes”). Game consoles are expensive because of the many hardware components (semiconductor chips, graphics cards, memory, and the like). *Id.* In addition, consumers are sufficiently sensitive to the costs of the consoles so that it makes commercial sense to take losses on sales of consoles that can then be made up on sales of applications. A large installed base is helpful to achieving this objective. See, e.g., William E. Cohen, *Competition and Foreclosure in the Context of Installed Base and Compatibility Effects*, 64 *Antitrust L.J.* 535 (1996).

¹⁸⁶ *U.S. v. Microsoft*, 243 F.3d 34 (D.C. Cir. 2001)(finding Microsoft had monopoly power in the market for operating systems for Intel-compatible PCs). See also Franklin M. Fisher & Daniel L. Rubinfeld, *U.S. v. Microsoft: An Economic Analysis* in *DID MICROSOFT HARM CONSUMERS? TWO OPPOSING VIEWS* 9-14 (2000) (discussing Microsoft’s monopoly).

“embrace and extend”).¹⁸⁷ This has effectively destroyed the market for some applications whose developers had been counting on revenues from mass-marketing of their applications. In addition, Microsoft has responded to some competition in the applications market by providing suites of popular applications (e.g., Microsoft Office) at an attractive price to obviate the need for users of the Windows operating system to acquire individual products from other vendors. Microsoft has also been aggressive—many would say overly so—in responding to innovations with potential to become alternative platforms to Windows, such as the Netscape’s browser software and the Java programming system.¹⁸⁸ Even if much is disputed about Microsoft’s conduct in preserving its operating systems monopoly, no one would dispute that Microsoft’s control over the APIs for developing applications for the Windows platform is an important source of its enduring power in this market.¹⁸⁹

Into this strategic environment, we now introduce reverse engineering. Platform developers may have copyrights on operating system programs and patents on some components of their systems, but APIs are typically maintained as trade secrets.¹⁹⁰ If reverse engineering is unlawful or if the platform is otherwise immune from reverse engineering (e.g., because the interfaces are too complicated or change rapidly),¹⁹¹ trade secrets can be a very effective form of intellectual property protection for platform APIs.¹⁹² If reverse engineering is lawful, effective protection for platforms is at greater risk. Reverse engineering clearly threatens to upset a platform developer’s non-interoperability strategy, whether unlicensed entry occurs at the applications level or at the platform level. From the standpoint of an unlicensed applications developer, reverse engineering offers a means of achieving compatibility between its products and the large

¹⁸⁷ The Dept. of Justice charged that Microsoft’s decision to integrate its Internet Explorer browser into the Windows operating systems was intended was done to harm the market for Netscape’s competing browser. See, e.g., *U.S. v. Microsoft*, 243 F.3d 34 (D.C. Cir. 2001)(discussing theory but remanding case to trial court for further findings). See also John Heilemann, *The Truth, The Whole Truth and Nothing But the Truth*, <http://www.wired.com/wired/archives/8.11/microsoft.html>.

¹⁸⁸ The World Wide Web opened up new opportunities for evolution of new platforms, such as browser software, for which applications could be written. See, e.g., Fisher & Rubinfeld, *supra* note 186, at 14-24. See also Mark A. Lemley & David McGowan, *Could Java Change Everything? The Competitive Propriety of a Proprietary Standard*, 43 *Antitrust Bull.* 715 (1998).

¹⁸⁹ See, e.g., *U.S. v. Microsoft*, 243 F.3d 34 (D.C. Cir. 2001)(“the applications barrier to entry protects a dominant operating system irrespective of quality”).

¹⁹⁰ Courts have held that copyright protection does not extend to interfaces of computer programs. See, e.g., *Computer Associates Int’l v. Altai, Inc.*, 983 F.2d 693 (2d Cir. 1992). Patents may sometimes protect program interfaces. See, e.g., *Atari Games Corp. v. Nintendo of Am., Inc.*, 30 U.S.P.Q. 2d (BNA) 1420 (N.D. Cal. 1993)(granting partial summary judgment to Nintendo on patent infringement claims as to interface components).

¹⁹¹ See *infra* Section IV-B(2).

¹⁹² Economists Joseph Farrell and Michael Katz consider intellectual property as determining whether a rival network will be compatible. They do not distinguish platforms from applications, but argue that intellectual property in the interface increases the incentive for quality improvements in a system as a whole. See Joseph Farrell & Michael L. Katz, *The Effects of Antitrust and Intellectual Property Law on Compatibility and Innovation*, 43 *Antitrust Bull.* 609 (1998). We caution, however, that intellectual property in the interface may be unnecessary if platforms and applications are themselves protected. With intellectual property in platforms and applications, intellectual property in the interfaces may serve no beneficial purpose and only allow developers to leverage market power in a way that was unintended as a matter of intellectual property law.

installed base of a successful system.¹⁹³ Although it would have been easier and quicker to license the Sega Genesis interface, Accolade would have had to stop writing for other platforms, due to Sega's insistence on exclusivity in its licenses with applications developers for the Genesis platform.¹⁹⁴ Reverse engineering gave Accolade an alternative way to get access to the Sega interfaces.

Table 3 represents the principal economic effects likely to flow from a decision about the legality of reverse engineering as a means for achieving interoperability. Although we use similar criteria as for traditional manufacturing and semiconductor chips, the welfare effects are more complicated and ambiguous. The reasons for this are explained below.

Table 3
Social Calculus of Reverse Engineering of Software for Purposes of Interoperability

Social welfare criterion	RE legal	RE illegal
Incentives to develop platform	lower (but adequate?)	higher (too high?)
Incentives to develop applications	high	high
Price	higher (short run)? lower (long run)?	lower (short run)? higher (long run)?
Duplicated costs	lower?	higher?

The conclusion about which we have the greatest confidence is that incentives to invest in platform development will be lower if reverse engineering is lawful. If outsiders can legally reverse engineer to get access to software interfaces, this erodes the market power of a non-interoperable platform developer.¹⁹⁵ In this respect, reverse engineering poses the same threat in the software industry as in traditional manufactured industries: it erodes market power by facilitating unlicensed entry or by inducing licensing on terms more favorable to the licensee than if reverse engineering was prohibited.¹⁹⁶ Of course, this does not necessarily mean that reverse engineering should be made illegal in order to protect platform developers. This depends on the cost and time required for reverse engineering. Because decompilation and disassembly are time-

¹⁹³ The unlicensed entrant who reverse engineers the APIs and then sells system components may benefit by substantial expenditures made by the platform provider to promote the platform in the market. Microsoft's Xbox system will be launched with a \$500 million marketing campaign. Gaither, *supra* note 184, at C15.

¹⁹⁴ *Sega v. Accolade*, 977 F.2d at 1514.

¹⁹⁵ Graham and Zerbe emphasize this factor in their economic analysis of reverse engineering in the software industry. See Graham & Zerbe, *supra* note 151, at 122.

¹⁹⁶ See *supra* Section II-B.

consuming and resource-intensive, these forms of reverse engineering do not seem to significantly undermine incentives to invest in platforms.¹⁹⁷

As for applications, there are strong incentives to develop them whether interfaces are open or closed. If interfaces can be reverse engineered (and hence are potentially open), a wide array of applications may be developed from ISVs. Open interfaces allow third-party developers to enter the market at one level (e.g., software applications) rather than having to enter at two levels (e.g., software plus platform). A small applications developer, such as Accolade, would never have been able to launch a whole system to compete with the Sega Genesis system on the strength of its Mike Ditka football program. However, it could afford to reverse engineer the Sega system and tailor the program to be compatible with the Genesis platform. As this example shows, open interfaces not only encourage third parties to develop applications, but also to adapt applications to multiple platforms. This has the social benefit of reducing duplicated costs of software development.

However, there are also strong incentives to develop applications when interfaces are proprietary and cannot be reverse engineered. Platform developers have incentives to develop applications for their platforms and to offer attractive terms to ISVs (including subsidies) because consumers mainly buy platforms because of applications available for them. If some or all of the applications for a proprietary platform are available exclusively for that platform, the proprietary platform will have a competitive advantage over rival platforms insofar as its stable of applications is more desired by consumers. The platform developer's ability to attract ISVs to develop applications for the platform and to recoup subsidies incurred to attract applications developers may be negatively affected by a rule favoring reverse engineering.¹⁹⁸ Of course, incentives to develop applications also depend on the extent of intellectual property protection available to them. If such protection is weak and competitors can imitate design elements of a

¹⁹⁷ It may be worth noting that reverse engineering in the software industry rarely involves development of a competing platform, but more often involves entry at the applications level. In the two cases involving products competing with a platform, *Sony v. Connectix* and *Sony v. Bleem*, discussed supra note 169 and accompanying text, the platform developer was actually losing money on the sale of each platform. See supra note xx. One might, therefore, have expected Sony to welcome new entrants that would expand their installed base without causing the firm additional losses, but this was not Sony's response. Sony complained of reputational damage because games played less well on the emulator platforms. See, e.g., *Connectix*, 203 F.3d at 608-09.

¹⁹⁸ If reverse engineering is lawful, licensed ISVs may be impaired in their recoupment of R&D expenses if unlicensed entrants can now offer competing applications for the platform—and can do so without paying royalties to the platform developer for the right to make applications for the platform. However, there are counterbalancing factors: first, licensed ISVs will have significant first-mover advantages in the applications market as compared with reverse engineers because decompilation and disassembly are so difficult and time-consuming, and second, over time, licensed ISVs may be in a better position to negotiate with platform developers for terms more favorable to them if reverse engineering is a legal option. Especially if the applications developer has had a “hit” in the applications market for a non-interoperable system, it may be able to negotiate more favorable terms, such as a right to develop its applications for more than one platform. See, e.g., Surowiecki, supra note 179, at 36.

proprietary application, this may erode the market advantage the platform owner had hoped to garner through its investment.¹⁹⁹

Incentives to develop platforms and applications are, of course, tied up with pricing, which is the third social welfare criterion in the table. The economics literature on systems competition delivers a counterintuitive conclusion about prices in respect of interoperable and non-interoperable systems. This literature suggests that prices will be higher when systems are interoperable than when they are not.²⁰⁰ Insofar as this is true, and insofar as reverse engineering enables interoperability and undermines a non-interoperable strategy, pricing considerations might suggest that a rule *against* reverse engineering would be beneficial to consumer welfare.²⁰¹

The argument about systems competition and price has two prongs: First, competition between incompatible systems will be fiercer than between compatible systems, leading to lower prices.²⁰² This is because system developers have an incentive to hold the prices of system components somewhat in check in order to compete successfully against other non-interoperable systems. Interoperability also mutes the fear of losing market share in platforms because a platform owner that makes software for its own and for other platforms will realize that it can still profit from selling software for other platforms and hence, has less at stake in promoting its own platform.²⁰³ The second argument follows from the observations of Augustin Cournot,²⁰⁴ who observed that if a single firm sells complementary pieces of a whole, it will do so at a lower total price than two firms selling the components separately. Interestingly, the total price offered by the

¹⁹⁹ This helps to explain the “look and feel” lawsuits of the late 1980’s and early 1990’s. See, e.g., *Apple Computer Inc. v. Microsoft Corp.*, 35 F.3d 1435 (9th Cir. 1994)(rejecting Apple’s claim that the look and feel of Microsoft’s graphic user interface (GUI) infringed Apple’s copyright in the Macintosh GUI); *Lotus Development Corp. v. Borland Int’l*, 49 F.3d 807 (1st Cir. 1995)(rejecting Lotus’ claim that the emulation interface of Borland’s Quattro Pro spreadsheet program infringed Lotus 1-2-3). See also *Data East USA, Inc. v. Data East Int’l, Inc.*, 204 F.3d 1008 (9th Cir. 1998)(no infringement where similarities between two independently developed karate programs lay in standard features to be expected of such games).

²⁰⁰ By contrast, prices will generally be lower in equilibrium when reverse engineering is permissible in a traditional manufacturing context. See *supra* Section II-B.

²⁰¹ Some commentators have argued the merits of reverse engineering in the software industry based on price considerations. See, e.g., Douglas Lichtman, *Property Rights in Emerging Platform Technologies*, 29 J. Legal Stud. 615 (2000) (proposing that systems developers be given monopoly control over complementary products on the theory that this will lead to lower prices). Cohen and Lemley question Lichtman’s conclusion, see Cohen & Lemley, *supra* note 7 at 22 n. 77, but say that if he was correct, “his argument would be a reason to oppose reverse engineering in one specific class of cases: complementary goods to strong network markets.” *Id.* We believe that price is not the only social welfare criterion that must be considered in judging whether to permit reverse engineering in software system markets and that the economic models on pricing and interoperability may not fully explain software industry dynamics.

²⁰² See, e.g., Carmen Matutes and Pierre Regibeau, *Mix and Match: Product Compatibility Without Network Externalities*, 19 RAND J. Econ. 221 (1988).

²⁰³ See, e.g., Jeffrey Church and Neil Gandal, *Systems Competition, Vertical Merger and Foreclosure*, 9 J. Econ. & Mgmt. Strategy 25 (2000).

²⁰⁴ See Augustin Cournot, *Researches into the Mathematical Principles of the Theory of Wealth* (1838) (in French, English translation by Nathaniel Bacon, published by Oxford Press in 1927). Cournot focused on monopolists rather than oligopolists, but the intuition from his work is nevertheless informative. See Nicholas Economides and Steven C. Salop, *Competition and Integration among Complements and Network Market Structures*, 40 J. Indus. Econ. 105 (1992).

integrated firm will also yield more profit than the (higher) joint price charged by separate firms. Hence, integrated ownership benefits both firms and consumers.²⁰⁵ These arguments have different implications for platform owners. In the first (regarding competition between systems), the lower prices due to integrated ownership are deleterious to platform owners. In the second (regarding monopoly ownership), the lower prices due to integrated ownership lead to higher profit.

We admit that it is difficult to reconcile this analysis about the effects of interoperability on competition and price with the behavior of Sega and Nintendo. These firms sold competing incompatible systems, and tried with every legal means to keep their interfaces proprietary. Both forbade licensees from making games for other platforms,²⁰⁶ and both initiated lawsuits to stop unlicensed entrants, such as Accolade, from making games for their proprietary platforms or adapting games made for other platforms (recall that Accolade made games for IBM PCs). If the argument about the effects of systems competition on price were applicable,²⁰⁷ both Sega and Nintendo would have been better off agreeing to make their systems interoperable. Their opposition to unlicensed entrants such as Accolade and Atari Games certainly did not stem from concern about excessive prices these upstarts might charge for applications for the Sega and Nintendo systems. This discrepancy between theory and practice highlights that the subject is not yet entirely understood by economists, and in particular, that there are aspects to the market that are not captured in the models referred to above.

Another complication about pricing is that systems competition may not be stable. Incompatibility may lead to a standards war, leaving only a single platform in the market.²⁰⁸ If so, the surviving platform can appropriate most of the benefits by charging

²⁰⁵ When one of the sellers unilaterally raises its price, both firms lose customers. The firm with higher price finds this profitable, but the other firm loses out. The price war thus engendered will end up with prices above the joint price that maximizes profit. The important ingredient in Cournot's conclusion is that the two products of the monopolist are complements rather than substitutes. If the two products are substitutes, then breaking up the monopolist will typically lower prices. Indeed, this is a key premise of antitrust policy. In the case of systems competition, each product sold in the market is complementary with some products, and is a substitute for others, leading to ambiguities. Which arguments dominate depend, among other things, on how many firms provide each component. See Joseph Farrell, Hunter Monroe, and Garth Saloner, *The Vertical Organization of Industry: Systems Competition versus Component Competition* 7 J. Econ. & Mgmt Strategy 143 (Summer 1988).

²⁰⁶ A platform developer has two related incentives to develop platform-specific software that is incompatible with other platforms. First, software makes the platform more valuable to its customers. Second, a greater variety of software can attract customers from the rival platforms. See Jeffrey Church and Neil Gandal, *Integration, Complementary Products, and Variety*, 1 J. Econ. & Mgmt. Strategy 651 (1992). There are at least two ways to achieve this objective: The platform owner can either develop the complementary software in-house or it can license independent software developers on condition that they not serve other platforms as well. Independent software developers will have incentives to take such licenses where the installed base of the licensor (platform) is large, and the licensor will offer terms that compensate them for restricting their options elsewhere. Both Sega and Nintendo used licensing as well as in-house software development for their systems.

²⁰⁷ We are less certain about the significance of Cournot's observations as applied to software systems. His insights apply most readily where there is a fixed proportion of complements rather than a decoupled proportion, as in the platform/applications systems market in the software industry context.

²⁰⁸ Katz and Shapiro argue that due to the demand feedback effects in creating proprietary software, the market could easily be tipped in favor one standard over another. For markets with direct network

high prices to captive customers. Hence, with proprietary interfaces immune from reverse engineering, consumers may face lower prices in the short run, but higher prices in the long run, after a single system has won the standards war and no longer feels the pressure to hold prices in check.

Duplicated or wasted costs is the fourth social welfare criterion, and it, too, gives somewhat ambiguous policy prescriptions. Costs can be duplicated or wasted in at least three activities: in the act of reverse engineering itself; in devising technical protections so that interfaces cannot be reverse engineered;²⁰⁹ and in writing different applications for different interfaces rather than the same applications for all interfaces. A prohibition on reverse engineering would avoid the first two, but may well encourage the third. A platform provider can, of course, avoid the first cost by licensing, and as in other industrial contexts, a legal rule in favor of reverse engineering may provide powerful incentives for firms to license to avoid having their products reverse engineered.

It is difficult to integrate all of these considerations into one unassailable conclusion about the welfare effects of reverse engineering in the software industry. On balance, we believe that consumers benefit from interoperability because it encourages the development and wide dissemination of software. While the economics literature has not precisely addressed this question, related research suggests that more software will be developed when software developers are not integrated with or controlled by makers of incompatible platforms.²¹⁰ *A fortiori*, we would expect the incentives to develop software to be even stronger if the systems *were* compatible, as may be achieved by reverse engineering. On the other hand, there is also a powerful incentive to develop applications when systems are incompatible, namely, to capture the whole market through network externalities. Once the market is captured, the incumbent's strategy regarding both prices and further software development may be less beneficial. Interoperability, achieved by reverse engineering or otherwise, reduces this threat.²¹¹ Hence, we conclude that a legal rule permitting reverse engineering to achieve interoperability in the software industry is, on balance, economically sound.

C. Reverse Engineering of Software And Patent Law

externalities to consumers, they argue, under a variety of reasonable assumptions about the formation of expectations as to which system will “win,” such tipping is almost inevitable. Katz & Shapiro, *supra* note 180. In a model more specifically focused on systems competition, Church and Gandal also concluded that a firm might choose incompatibility in order to drive a rival out of business. This can happen even when the number of software products is fixed in advance. Church & Gandal, *supra* note 203. Tipping has the benefit of maximizing network externalities, but may have the economic disadvantage of proprietary prices.

²⁰⁹ See, e.g., Cohen, *supra* note 151, at 1094. See also *infra* Section VI-B(2) for a discussion of policy

²¹⁰ Church & Gandal, *supra* note 206.

²¹¹ Market domination by single standard has two sides. It promotes consumer welfare by capitalizing on network externalities, but if the standard is proprietary, also promotes proprietary prices. Lemley and McGowan, *supra* note 41 at 525, argue that a strong reason to permit reverse engineering is to promote competition within an industry standard. Such competition will not occur if the platform embodying the standard is itself proprietary, e.g., the game platforms of Nintendo and Sega. Here we draw a distinction between intellectual property on the platform, so that entrants cannot duplicate it, and intellectual property on the standard, which would mean that third-party providers cannot write software to it.

Although the decompilation-for-interoperability issue has been resolved as a matter of copyright law, it has not been resolved as a matter of patent law.²¹² To illustrate how decompilation might run afoul of patent law, consider this variant on the *Sega v. Accolade* dispute: Assume that Sega had a patent on an algorithm used in all of its game programs. By disassembling Sega programs, Accolade would arguably “make” or “use” this patented aspect of Sega’s programs, even if it did so unconsciously and inadvertently. Because patent law has no fair use defense, Accolade could not raise such a defense to a patent infringement claim.²¹³ We concur with the views of other commentators who recommend a limited reverse engineering privilege in patent as well as copyright law.²¹⁴

Professors Cohen and Lemley have cogently analyzed decompilation of computer programs as potential patent infringements.²¹⁵ They point out that “because patent law contains no fair use or reverse engineering exemption, patentees could use the grant of rights covering a single component of a complex program to prevent any ‘making’ or ‘using’ of the program as a whole, including those temporary uses required for reverse engineering.”²¹⁶ They propose “a limited right to reverse engineer patented computer programs to permit study of those programs and duplication of their unprotected elements.”²¹⁷ Their argument rests in part on the general point that “reverse engineering is an important means of preserving competition between products and of preserving compatibility between products. In markets characterized by network effects, such as software, this latter objective is particularly important.”²¹⁸ They give several additional reasons for establishing a limited reverse engineering privilege in patent law: “Reverse engineering promotes the fundamental [patent] policies of disclosure and enablement,²¹⁹ ensures that patents will not be leveraged to protect unprotectable components of software,²²⁰ preserves the balance sought by the intellectual property system as a whole,²²¹ and also helps patentees to enforce their rights.”²²²

²¹² See, e.g., Cohen & Lemley, *supra* note 7, at 19-37. There may soon be an opportunity for U.S. courts to rule on the patent reverse engineering issue because Sony Entertainment has sued the makers of emulation programs that allow owners of Sony games to play them on non-Sony platforms, the charge being that these firms infringed Sony’s patents during the decompilation process. See *id.* at 21. Sony brought suit after it lost a similar challenge to the same conduct predicated on copyright law. See *Sony Computer Entertainment, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000).

²¹³ U.S. patent law gives inventors exclusive rights to control the making, using and selling of their protected inventions. 35 U.S.C. sec. 271.

²¹⁴ Cohen & Lemley, *supra* note 7, at 19-37.

²¹⁵ *Id.* at 18-37.

²¹⁶ *Id.* at 6. But see Maureen O’Rourke, *Towards a Fair Use Defense in Patent Law*, 100 Colum. L. Rev. 1177 (2000) (arguing for such a defense).

²¹⁷ Cohen & Lemley, *supra* note 7, at 6.

²¹⁸ *Id.* at 21.

²¹⁹ “[S]oftware patentees generally do not disclose much, if any, detail about their programs, and therefore, there is no easy way to figure out what a software patent owner has build except to reverse engineer the program.” *Id.* at 25.

²²⁰ “If reverse engineering is illegal, then patenting even a small part of one computer program can give the patentee effective control over all the ideas contained in the program.” *Id.* at 26.

²²¹ “Copyright and trade secret law both have strongly articulated policies of permitting reverse engineering where it is undertaken for a legitimate social purpose. For patent law to ban reverse engineering of software would undermine the goals of both copyright and trade secret law.” *Id.* at 27.

Cohen and Lemley consider various doctrines under which such a reverse engineering privilege might be established, including patent law's experimental use defense, exhaustion of rights defense, implied license, and misuse.²²³ They conclude that the policies underlying the exhaustion of rights and implied license doctrines of patent law should suffice to permit reverse engineering of programs.²²⁴ If courts decide otherwise, Cohen and Lemley argue for legislation to permit it.²²⁵ We agree that the limited reverse engineering rule they propose is legally and economically sound.

D. Reverse Engineering of Software And Contract Law

Another strategy for prohibiting decompilation and other forms of reverse engineering of programs has been through contractual restrictions, often by licenses inserted in boxes of packaged software.²²⁶ The enforceability of such restrictions has been a highly contentious legal issue both in the U.S. and abroad.²²⁷ The caselaw in the U.S. is in conflict on the enforceability of anti-reverse engineering clauses in software contracts.²²⁸ Irresolution in the caselaw might suggest the need for a legislative

²²² Id. at 22-23. They point out that “[a] patent owner who suspects a rival of infringing [its] software patent may have no choice but to reverse engineer the rival’s software in order to gain the evidence it needs to file suit.” Id. at 28.

²²³ Id. at 29-36.

²²⁴ Id. at 32. Cohen and Lemley regard efforts to restrict decompilation through software licenses as misuse of intellectual property rights. Id. at 35-36. This issue will be considered in the next section.

²²⁵ Id. at 36-37.

²²⁶ See, e.g., Mark A. Lemley, *Beyond Preemption: The Law and Policy of Intellectual Property Licensing*, 87 Calif. L. Rev. 111, 129 (1999).

²²⁷ The European Union has declared that anti-decompilation clauses in software contracts are null and void. See European Software Directive, *supra* note 163, Art. 9(1). The principal reason EU chose to make anti-decompilation clauses unenforceable was to create incentives for firms to license interface information on a reasonable basis so that the second comers would not resort to reverse engineering to get this information. See Official Commentary, reproduced in Czarnota & Hart, *supra* note 163, at 76-80. A few other countries, notably Australia, have followed suit. See, e.g., Jonathan Band, *Software Reverse Engineering Amendments in Singapore and Australia*, J. Internet L. 17, 20 (Jan. 2000).

²²⁸ Courts have sometimes rejected reverse engineering defenses in trade secrecy cases because this activity exceeded the scope of licensed uses of the software. See, e.g., *Technicon Data Systems Corp. v. Curtis 1000, Inc.*, 224 U.S.P.Q. 286 (Del. Ch. 1984) (consultant to hospital used improper means to obtain trade secret interface information by wiretapping the hospital’s licensed software system to study the manner in which the server software exchanged data with the client software because it had not been authorized by the hospital, and even if it had been, the action would have breached restrictive terms in the license). See also *DSC Communications Corp. v. Pulse Communications, Inc.*, 170 F.3d 1354 (Fed. Cir. 1999) (triable issue of fact as to whether Pulsecom’s use of “snooper board” at telephone company to get access to interface information about DSC’s software was misappropriation of trade secret in view of restrictions in the telephone company’s license to use DSC’s software).

A few cases have ruled against defendants because of anti-reverse engineering clauses in software licenses. See *DVD-CCA v. McLaughlin*, 2000 WL 48512 (Cal. Sup. Ct. 2000) (posting program to bypass technical protection used by movie companies to protect DVDs held to be a misappropriation of trade secrets because program was developed in violation of shrinkwrap license restriction on reverse engineering). For a non-software case in which an anti-reverse engineering clause was enforced, see *K&G Oil & Tool Service Co. v. G&G Fishing Tool Service*, 158 Tex. 594, 314 S.W.2d 782 (1958).

In other cases, courts have declined to enforce shrinkwrap license restrictions against reverse engineering, sometimes because of a conflict between the clause and federal intellectual property policy.

resolution. However, legislative approaches have also been contentious, as witnessed by the controversy over the model law now known as the Uniform Computer Information Transactions Act (UCITA).²²⁹

UCITA aims to resolve the decades' long controversy about shrinkwrap and other mass market licenses for software.²³⁰ As long as a user has had a reasonable opportunity to review the terms of a license, merely using the software may constitute the user's assent to the license terms.²³¹ Endorsing favoring freedom of contract as its policy,²³² UCITA generally presumes license terms to be enforceable unless unconscionable.²³³ Yet, owing to lingering concerns about imbalance in UCITA,²³⁴ this model law now provides that if "a term of a contract violates a fundamental public policy, the court may enforce the remainder of the contract without the impermissible term, or so limit the application of the impermissible term as to avoid any result contrary to public"²³⁵ UCITA also recognizes that if federal law preempts one of its provisions, that provision is "unenforceable to the extent of the preemption."²³⁶

The principal case is *Vault Corp. v. Quaid Software Ltd.*, 847 F.2d 255 (5th Cir. 1988) in which the maker of a copy-protection program sought to enforce an anti-reverse engineering clause of a shrinkwrap license under Louisiana law against a firm that had reverse engineered the copy-protection scheme. The Court of Appeals held that: "[t]he provision in Louisiana's License Act, which permits a software producer to prohibit the adaptation of its licensed computer program by decompilation or disassembly, conflicts with the rights of computer program owners under [the copyright law] and clearly 'touches upon an area' of federal copyright law. For this reason...we hold that this provision of Louisiana's License Act is preempted by federal law, and thus that the restriction in Vault's license agreement against decompilation or disassembly is unenforceable." *Id.* at 270. See also *Symantec Corp. v. McAfee Associates*, 1998 WL 740798 (N.D. Cal. 1998) (state unfair business practice claim based on reverse engineering of another firm's program in violation of license agreement held preempted by copyright law).

Some cases have also ruled against enforcing shrinkwrap licenses as a matter of contract law, either as contracts of adhesion or as lacking mutuality of consent, although the caselaw is mixed on this issue as well. Compare *Step Saver Data Systems v. Wyse Technology*, 939 F.2d 91 (3d Cir. 1991) (shrinkwrap license not enforceable as a matter of contract law) and *ProCD Inc. v. Zeidenberg*, 86 F.3d 1447 (7th Cir. 1996) (enforcing shrinkwrap license restriction). See also Band & Katoh, *supra* note 151, at 221; L. RAY PATTERSON & STANLEY W. LINDBERG, *THE NATURE OF COPYRIGHT—A LAW OF USER'S RIGHTS* 220 (1991).

²²⁹ Uniform Computer Information Transactions Act. With some consumer protection modifications, UCITA was enacted and is in force in Maryland. The Virginia also enacted it with a two year moratorium.

²³⁰ See, e.g., Robert W. Gomulkiewicz, *The License Is the Product: Comments on the Promise of Article 2B for Software and Information Licensing*, 13 *Berkeley Tech. L.J.* 891 (1998) (reviewing history of model law project and issues).

²³¹ See UCITA, *supra* note 229, secs. 112, 210-11.

²³² See Reporter's Notes to UCITA sec. 104 (UCITA conforms to "fundamental policy of the United States which holds that freedom of contract governs").

²³³ UCITA, *supra* note 229, sec. 111. UCITA does limit licensor freedom to some degree, for example, as to choice of law clauses in consumer contracts. *Id.*, sec. 109. To the extent UCITA might conflict with an applicable consumer protection law, the latter will govern. *Id.*, sec. 105(c). Some commentators have pointed out that most consumer protection laws apply to sales of goods and not to licenses of goods, and hence sec. 105 may supply less protection to consumers than might be apparent. See, e.g., Jean Braucher, Memorandum, *The Uniform Computer Information Transactions Act (UCITA): Objections From The Consumer Perspective*, Aug. 15, 2000 (on file with the authors).

²³⁴ See, e.g., Charles R. McManis, *The Privatization (or "Shrinkwrapping") of American Copyright Law*, 87 *Calif. L. Rev.* 173, 187-90 (1999) (discussing compromise).

²³⁵ *Id.*, sec. 105(b).

²³⁶ *Id.*, sec. 105(a).

The implications of these UCITA provisions for anti-reverse engineering clauses have been the subject of considerable debate.²³⁷ Some commentators believe that anti-reverse engineering clauses in mass market licenses should be unenforceable on copyright preemption grounds.²³⁸ Others have asserted that such clauses should be considered a misuse of intellectual property rights.²³⁹ Still others have suggested enforcing such license terms in negotiated licenses, but not in non-negotiated standard form contracts.²⁴⁰ Another suggestion is to enforce them unless the firm imposing the license term has monopoly power.²⁴¹ A new doctrine of public interest unconscionability has also been proposed under which anti-reverse engineering clauses would be unenforceable.²⁴²

Counterarguments abound as well.²⁴³ Critics point out that copyright preemption of contract terms is rare.²⁴⁴ Misuse of intellectual property rights is a doctrine of uncertain scope and application, and some have opined that it should extend no farther than antitrust law would.²⁴⁵ Because most consumers do not want to reverse engineer the software they buy, it may be difficult to challenge anti-reverse engineering clauses on unconscionability grounds.²⁴⁶ While antitrust and competition law may regulate anti-reverse engineering clauses in an appropriate case or context, no such claim has as yet been brought, let alone sustained.

²³⁷ See, e.g., Lemley, *supra* note 226; David McGowan, *Free Contracting, Fair Competition, and Article 2B: Some Reflections on Federal Competition Policy, Information Transactions, and "Aggressive Neutrality,"* 13 Berkeley Tech. L.J. 1173 (1998); Reporter's Notes to UCITA sec. 105.

²³⁸ See, e.g., McManis, *supra* note 234; David A. Rice, *Public Goods, Private Contract and Public Policy: Federal Preemption of Software License Prohibitions Against Reverse Engineering,* 53 U. Pitt. L. Rev. 543 (1992).

²³⁹ See, e.g., Lemley, *supra* note 226, at 151-58. But see Marshall Leaffer, *Engineering Competitive Policy and Copyright Misuse,* 19 U. Dayton L. Rev. 1087, 1106 (1994).

²⁴⁰ See, e.g., David Nimmer, Elliot Brown, & Gary N. Frischling, *The Metamorphosis of Contract Into Expand,* 87 Calif. L. Rev. 17, 68 (1999). The Reporter's Notes to sec. 105(b) opined that anti-reverse engineering terms would likely be enforced as to negotiated contracts, but acknowledged as an open question whether they would be enforced in mass market licenses. See, e.g., McGowan, *supra* note 237, at 1195-98 (1998) (reviewing various iterations of the Reporter's Notes).

²⁴¹ See, e.g., Maureen O'Rourke, *Drawing the Boundary Between Copyright and Contract: Copyright Preemption of License Terms,* 45 Duke L.J. 479, 551 (1995). See also McGowan, *supra* note 237, at 1176-77 (raising question about enforcing such terms in concentrated markets).

²⁴² See, e.g., J.H. Reichman & Jonathan A. Franklin, *Privately Legislated Intellectual Property Rights: Reconciling Freedom of Contract With Public Good Uses of Information,* 147 U. Pa. L. Rev. 875, 939 (1999).

²⁴³ See, e.g., Raymond T. Nimmer, *Breaking Barriers: The Relation Between Contract and Intellectual Property Law,* 13 Berkeley Tech. L.J. 827, 861-88 (1998) (responding to arguments based on preemption, misuse, and other doctrines).

²⁴⁴ See, e.g., Reporter's Notes to UCITA sec. 105(a). See also Lemley, *supra* note 226, at 144-50 (discussing limits of preemption doctrine as applied to licensing). In general, state contract claims are different enough in kind from copyright claims as to be beyond preemption. See, e.g., *National Car Rental System, Inc. v. Computer Associates Int'l, Inc.*, 991 F.2d 426 (8th Cir. 1993); *ProCD Inc. v. Zeidenberg*, 86 F.3d 1447 (7th Cir. 1996). But see Nimmer et al., *supra* note 240, at 42-57 (critical of preemption analysis in *ProCD*); Niva Elkin-Koren, *Copyright Policy and the Limits of Freedom of Contract,* 12 Berkeley Tech. L.J. 93 (1997).

²⁴⁵ See, e.g., Lemley, *supra* note 226, at 152, n. 188.

²⁴⁶ See, e.g., McGowan, *supra* note 237, at 1204-14.

Some legal commentators have pointed to collective action problems and negative externalities as impediments to achieving the appropriate market outcomes via contract law that UCITA's freedom of contract policy assumes.²⁴⁷ In respect of anti-reverse engineering clauses, Professor McGowan points out:

On average, consumers would probably assent to limitations relating to reverse engineering, their assent would be rational, and requiring evidence of deliberative assent therefore would increase transactions costs without yielding corresponding benefits that are relevant to federal policy concerns....The collective product of such atomistic acts of assent, however, would pose the same risks for social welfare that advocates of legal rules facilitating reverse engineering...would like to ameliorate—lethargic transition among standard products and diminished production of works building upon ideas embedded in object code.²⁴⁸

There is, in addition, a wider public interest in the availability of competitive products in the future that might be thwarted if anti-reverse engineering clauses were enforced. Third party effects of enforcing anti-reverse engineering clauses might, therefore, be harmful to consumer welfare. McGowan concludes that if “reverse engineering furthers copyright’s goal of promoting the dissemination and improvement of intellectual property [and] reverse engineering does not deprive authors of returns necessary to induce investment..., then competition policy would favor reverse engineering as a device to lower the cost of transition among standard products (thereby increasing allocative efficiency) without infringing on copyright goals or methodology.”²⁴⁹

As explained above, we believe that the welfare effects of reverse engineering in the software industry context are more complex than this. However, on balance, reverse engineering and interoperability are important because they promote development of a wider range of software from a wider array of developers than a market in which platform developers were insulated from reverse engineering. To the extent that enforcement of anti-reverse engineering clauses would have a detrimental effect on competitive development and innovation, legal decisionmakers may be justified in not enforcing them.²⁵⁰

V. Reverse Engineering of Technically Protected Digital Content

²⁴⁷ See, e.g., Julie E. Cohen, Lochner in Cyberspace: *The New Economic Orthodoxy of “Rights Management,”* 97 Mich. L. Rev. 462, 536-38, 547 (1998).

²⁴⁸ McGowan, supra note 237, at 1213-14.

²⁴⁹ Id. at 1205-06.

²⁵⁰ We agree with other commentators that the argument for non-enforcement of anti-reverse engineering clauses is strongest as to mass-market software and weakest as to negotiated agreements between sophisticated firms. See, e.g., O’Rourke, supra note 241; Reichman & Franklin, supra note 242. See also infra Section VI-B(1).

In October 1998, the U.S. Congress enacted the Digital Millennium Copyright Act (DMCA).²⁵¹ Among other things, it substantially limits reverse engineering of technical measures used to protect copyrighted digital content, such as motion pictures, sound recordings, videogames, computer software, and “e-books.” One rule forbids circumvention of technical measures used to control access to such content.²⁵² A second outlaws technologies primarily designed to circumvent technical measures.²⁵³ A third outlaws alteration or removal of copyright management information (CMI), such as digital watermarks.²⁵⁴

Although the DMCA rules are not explicitly cast as restrictions on reverse engineering, that is their essential nature. Just as it is impossible to reverse engineer object code without decompiling or disassembling it, it is impossible to reverse engineer a technical protection measure without circumventing it and impossible to reverse engineer a digital watermark without removing or altering information that it contains. Someone who reverse engineers a technical protection measure will also generally need a tool in order to perform such reverse engineering activities, so by outlawing the making of circumvention technologies, the law indirectly forbids reverse engineering.

The DMCA’s restrictions on reverse engineering represent a complete inversion of the rules that apply in other industrial contexts. Under the DMCA, reverse engineering would seem to be illegal except when a specific statutory or rule-making exception applies.²⁵⁵ Even when allowed, the DMCA strictly regulates what can be done with the resulting information.²⁵⁶ Even tools for reverse engineering are, for the most part, banned. The range of these restrictions is unprecedented in American law.

The DMCA anti-circumvention rules respond to copyright industry fears of uncontrolled infringements as to digital versions of their content (movies, music, and the like). Digital content is very cheap and easy to copy and distribute via digital networked environments, and hence, it is vulnerable to market-destructive appropriations.²⁵⁷ As the well-known cryptographer Bruce Schneier has wittily observed, “[d]igital files cannot be

²⁵¹ Pub. L. No. 105-304, 112 Stat. 2860 (1998).

²⁵² 17 U.S.C. sec. 1201(a)(1)(A). Enforcement of this provision was suspended for the first two years after enactment of the DMCA. *Id.*

²⁵³ 17 U.S.C. sec. 1201(a)(2), (b)(1). The first of these provisions pertains to technologies to bypass access controls and the other to technologies to bypass other technical protection measures used by copyright owners to protect their works. These rules also apply if there is no commercially significant use of the technology except for circumvention and if the technology is marketed as a circumvention device.

²⁵⁴ 17 U.S.C. sec. 1202. Because of differences in the way this provision is formatted in some statutory compilations, there is some ambiguity about whether alteration or removal of CMI is by itself enough to violate section 1202 or whether this must be done for the purpose of facilitating or concealing copyright infringement. We believe the latter interpretation is correct, and if so, this would limit adverse impacts of 1202 on reverse engineering of digital watermarks. In some statutory compilations, however, section 1202 is formatted as though this qualifying language only pertains to 1202(b)(3), not 1202(b)(1).

²⁵⁵ See *infra* notes 275-85 and accompanying text.

²⁵⁶ See *infra* Section VI-A(5).

²⁵⁷ NATIONAL RESEARCH COUNCIL, *THE DIGITAL DILEMMA: INTELLECTUAL PROPERTY IN THE INFORMATION AGE* 28-45 (2000) (cited hereinafter as “Digital Dilemma”).

made uncopyable, any more than water can be made not wet.”²⁵⁸ Although digital content can be scrambled, every known scrambling system has been hacked. According to Schneier, “nothing works against a dedicated and skilled hacker... including unlock codes, encryption, serial numbers, hardware devices, on-line verification, copy protection, file encryption and watermarking.”²⁵⁹ Schneier says that almost any protection will work against the average user, but *no* protection system will work against the power user, hacker, or professional pirate.²⁶⁰

The view articulated by Schneier may or may not be overstated, but we shall take it at face value as it provides the strongest argument for anti-circumvention rules. Even so, we will argue that the DMCA rules are far more restrictive than is necessary to achieve the objectives Congress had in mind when it adopted the DMCA rules.²⁶¹

Section A will recapitulate the legislative history that led to Congress’ enactment of the anti-circumvention rules. In Section B, we give an economic assessment of how the anti-circumvention rules address (or not) the fear of widespread infringement. In Section C we argue that, in its haste to restrain infringements through the DMCA, Congress enacted a broader set of rules than is needed to achieve this purpose. The overbreadth of the DMCA rules has produced serious collateral damage. In Section D we consider several less restrictive alternatives to the current DMCA rules that would more appropriately balance the interests of copyright owners and the public.

A. Legislative History of the DMCA Anti-Circumvention Rules

The Clinton Administration proposed amending copyright law to outlaw circumvention technologies in 1995 in its “White Paper” on Intellectual Property and the National Information Infrastructure.²⁶² Without anti-circumvention legislation, the White Paper expressed concern that copyright owners would not provide content for the NII because their works would be too vulnerable to widespread infringements.²⁶³ To give new assurances to copyright owners, it proposed a ban on making or distributing technologies, the primary purpose or effect of which were to circumvent technical

²⁵⁸ Bruce Schneier, *The Futility of Digital Copy Protection* at 2 (on file with the author). Schneier is the Chief Technology Officer of Counterpane Internet Security, Inc., designer of the popular Blowfish encryption system, and author of six books, including *SECRETS AND LIES: DIGITAL SECURITY IN A NETWORKED WORLD* (2000).

²⁵⁹ Bruce Schneier, *The Natural Laws of Digital Content*, slides of presentation at a conference on “Digital Libraries: Digital Asset Management” held at the Institute of Mathematics and its Applications in Minneapolis, MN, February 12, 2001 (on file with the authors).

²⁶⁰ *Id.*

²⁶¹ Schneier believes that the DMCA rules will, in the end, prove futile because the Internet is an inherently global communications medium. Even if the U.S. and some allies adopt similar anti-circumvention rules, such rules “would never have the global coverage [they] need[] to be successful.” Schneier, *supra* note 258, at 2. Schneier does not believe that the Internet spells the death of copyright, but only that “[w]e need business models that respect the natural laws of digital content instead of fighting them.” *Id.*

²⁶² Bruce Lehman, Report of the Working Group on Intellectual Property, Intellectual Property and the National Information Infrastructure 230-34 (Sept. 1995)(cited hereinafter as “White Paper”).

²⁶³ *Id.* at 230.

protections for copyrighted works.²⁶⁴ The Clinton Administration proposed a similar rule for a draft copyright treaty scheduled for consideration at a 1996 diplomatic conference convened at the World Intellectual Property Organization (WIPO).²⁶⁵ The draft treaty contained an anti-circumvention provision modeled on the White Paper proposal, but this proved controversial once the conference began.²⁶⁶ Diplomats eventually agreed upon a compromise provision directing member states to provide “adequate protection” and “effective remedies” against circumvention of technical protections,²⁶⁷ leaving the details of implementation to national discretion.

In 1997, the U.S. Congress considered legislation to implement the WIPO Copyright Treaty,²⁶⁸ the Clinton Administration supported anti-circumvention rules that were more expansive than the original White Paper proposal,²⁶⁹ although the new bill had an exception allowing circumvention of technical protection measures for legitimate law enforcement and national security purposes.²⁷⁰ Witnesses at hearings on the bill convinced Congress to craft other exceptions as well, and to authorize the Librarian of Congress to create other exemptions in periodic rulemakings.²⁷¹ Although some witnesses expressed concern about the impact of the anti-circumvention rules on fair uses of copyrighted works,²⁷² major copyright industry representatives opposed any exception for fair uses. One publishing industry witness stated: “Fair use doesn’t allow you to break into a locked library in order to make ‘fair use’ copies of books in it, or steal newspapers from a vending machine in order to copy articles and share them with a friend.”²⁷³ Circumvention and tools used for circumvention were analogized to burglary and burglars’ tools.²⁷⁴ Powerful rhetoric of this sort seems to have persuaded Congress that a general ban on circumvention and circumvention tools was necessary to protect copyrighted works in the digitally networked environment. Had Congress instead

²⁶⁴ Id., app. 1 at 6. Legal protection for technical measures was, the White Paper asserted, “not unprecedented,” id. at 232, giving as an example the rule against making and selling decoder boxes to unscramble encrypted cable and satellite television transmissions. See id.; 47 U.S.C. sec. 605(e).

²⁶⁵ See Pamela Samuelson, *The U.S. Digital Agenda at WIPO*, 37 Va. J. Int’l L. 369, 411-12 (1997) (discussing U.S. treaty proposal and the draft treaty provision).

²⁶⁶ Id. at 413-15 (discussing controversy over anti-circumvention rules at the WIPO conference).

²⁶⁷ WIPO Copyright Treaty, adopted Dec. 20, 1996, CRNR/DC/94, art. 11. A similar treaty pertaining to sound recordings was also adopted at the same diplomatic conference, and it has a nearly identical anti-circumvention provision. See WIPO Performances and Phonograms Treaty, adopted Dec. 20, 1996, CRNR/DC/95.

²⁶⁸ See H.R. 2281, 105th Cong., 1st Sess. (1997).

²⁶⁹ See Prepared Statement of Bruce A. Lehman, Hearing Before the Subcomm. On Courts & Intell. Prop. of the Comm. on the Judiciary, on H.R. 2281 (WIPO Copyright Treaties Implementation) and H.R. 2280 (Online Copyright Liability Limitation Act), 105th Cong., 1st Sess., Sept. 16-17, 1997 (cited hereinafter as “Judiciary Hearings”).

²⁷⁰ See H.R. 2281, sec. 3, reproduced in Judiciary Hearings, supra note 269, at 13.

²⁷¹ The evolution of this legislation is recounted in detail in Pamela Samuelson, *Intellectual Property and the Digital Economy: Why the Anti-circumvention Rules Need to Be Revised*, 14 Berkeley Tech. L.J. 519 (1999). See also JESSICA LITMAN, DIGITAL COPYRIGHT 89-150 (2001).

²⁷² See, e.g., Prepared Statement of Douglas Bennett, Judiciary Hearings, supra note 269, at 240-44.

²⁷³ Id. at 208 (prepared statement of Allan Adler of the Association of American Publishers).

²⁷⁴ See, e.g., House Manager’s Report, at 5 (characterizing circumvention tools as “the digital equivalent of burglars’ tools”).

understood the DMCA rules as anti-reverse engineering rules, the legislative debate might have ended with a more balanced result.

In addition to the exception for law enforcement and national security purposes,²⁷⁵ the DMCA now permits circumvention for these purposes:

- 1) to achieve program-to-program interoperability,²⁷⁶
- 2) to engage in “legitimate” encryption research (subject to many conditions that substantially limit its application),²⁷⁷
- 3) to test the security of computer systems (also subject to conditions that substantially limit its application),²⁷⁸
- 4) to allow non-profit libraries, archives, and educational institutions to enable them to make purchasing decisions,²⁷⁹
- 5) to allow parents to control their children’s use of the Internet,²⁸⁰ and
- 6) to protect personal privacy.²⁸¹

Since then, the Librarian of Congress has decided that circumventing access controls should be lawful in two other circumstances:

- 1) when an access control system is broken and the circumventor has a right to access the material, and
- 2) when necessary to assess the effectiveness of a software filtering program to determine what sites it blocks.²⁸²

Neither expressly authorizes the making of a tool to accomplish such privileged circumventions, and indeed it is unclear the Librarian of Congress has the authority to do so.²⁸³ Four of the seven statutory exceptions to the act-of-circumvention rule lack

²⁷⁵ 17 U.S.C. sec. 1201(e).

²⁷⁶ 17 U.S.C. sec. 1201(f). The reverse engineering exception adopts the core holding of *Sega v. Accolade* in legitimating reverse engineering when necessary to achieving interoperability. However, it narrows *Sega v. Accolade* by restricting what can be done with information obtained during the reverse engineering process, id. 1201(f)(3), by designating interoperability as the only legitimate purpose for which reverse engineering may be done, and by restricting the exception to achieving program-to-program interoperability even though circumvention may be needed to achieve hardware-to-program interoperability or program-to-data interoperability.

²⁷⁷ 17 U.S.C. sec. 1201(g). The limitations of this exception are discussed infra Section VI-C.

²⁷⁸ 17 U.S.C. sec. 1201(j).

²⁷⁹ 17 U.S.C. sec. 1201(d). This exception is of very limited utility to nonprofit libraries, archives and educational institutions.

²⁸⁰ 17 U.S.C. sec. 1201(h).

²⁸¹ 17 U.S.C. sec. 1201(i). This provision only applies if the user did not receive advance notice that the technical protection system would be collecting personal data. For a discussion of the implications of digital rights management systems technologies on user privacy, see Julie E. Cohen, *A Right to Read Anonymously: A Closer Look at “Copyright Management” in Cyberspace*, 28 Conn. L. Rev. 981 (1996). For an example of a privacy-intrusive use of technical protection measures that is not covered by this exception, see Samuelson, supra note 271, at 552-54.

²⁸² See Copyright Office, Exemption to Prohibition of Circumvention of Copyright Protection Systems for Access Control Technologies, 65 F.R. 64555, 64574 (2000), codified in 37 C.F.R. sec. 201.40(b) (2000).

²⁸³ The Librarian’s rulemaking authority seems to be limited under 17 U.S.C. sec. 1201(a)(1)(C) to developing exceptions to the act of circumvention rule of 1201(a)(1)(A). Yochai Benkler argues that the DMCA anti-circumvention rules are unconstitutional, in part because the Librarian’s authority is too constricted. See Yochai Benkler, *Free As the Air to Common Use: First Amendment Constraints on*

express authorization to make tools to accomplish circumventions.²⁸⁴ This raises a question whether there is an implied right to make a tool to engage in privileged circumventions or whether Congress created meaningless rights.²⁸⁵

B. An Economic Rationale for the DMCA Rules

Digital content cannot be protected against copying at all without technical protections.²⁸⁶ Copying is essentially costless and undetectable since such copying is likely to occur in nonpublic spaces. The difficulty in detecting infringement is a key distinguishing factor between digital information and manufactured objects. With manufactured objects, large-scale infringements require relatively large-scale manufacturing and distribution facilities. This makes such infringements public, which then makes enforcement possible.

Technical protections generally do not prevent copying, but rather impede the transposition of digital content into consumable form. The fact that it is difficult and costly to reverse engineer technical measures to make content consumable is not insignificant. If an encryption or scrambling system is good,²⁸⁷ the cost of circumvention can be very high. If so, users may find it easier and more sensible to buy the protected content rather than to circumvent the technical protection.²⁸⁸ In this sense, technical protections serve their intended purpose. They may also have the salutary effect of restraining the content owner's price so that users prefer purchase to piracy.²⁸⁹

Enclosure of the Public Domain, 74 N.Y.U. L. Rev. 354 (1999). Many scholars question the constitutionality of the DMCA anti-circumvention rules. See, e.g., Brief Amicus Curiae of Intellectual Property Professors, submitted to the Second Circuit Court of Appeals in *Universal City Studios, Inc. v. Reimerdes* (Jan. 26, 2001), available at

http://www.eff.org/IP/DMCA/MPAA_DVD_cases/20010126_ny_lawprofs_amicus.html.

²⁸⁴ For a discussion of this problem, see Samuelson, *supra* note 271, at 537-46

²⁸⁵ See, e.g., *id.* at 547. See also *Digital Dilemma*, *supra* note 257, at 175 (noting ambiguity in the DMCA as to whether there is an implied right to make a tool to engage in privileged circumventions).

²⁸⁶ For a discussion of technical protection measures that content owners are using or planning to use to protect their works, see, e.g., *id.* at 152-73, Appendix E; Daniel J. Gervais, *Electronic Rights Management and Digital Identifier Systems*, J. Electronic Publishing, March 1999, available at

<http://www.press.umich.edu/jep/04-03/gervais.html>. As Professor Lessig points out, the computer code that serves as a rights management technology is a kind of private governance system. See, e.g., LAWRENCE LESSIG, *CODE AND OTHER LAWS OF CYBERSPACE* (2000).

²⁸⁷ One of our concerns about the DMCA rules is that they provide such strong legal protection to technical protection measures that they may cause content owners to rely on weak encryption. As Professor Peter Swire has observed, "After last week's events [the destruction of the World Trade Center towers by hijacked airplanes], it is less tolerable to have a legal regime that encourages weak computer security and makes it illegal to push companies toward stronger security." Email communication from Peter Swire, Sept. 14, 2001 (on file with the authors).

²⁸⁸ Because circumvention is likely to be costly and difficult, users will tend to do this only when they really need to. For a list of some reasons why users might want to circumvent technical controls other than to infringe copyrights, see *infra* note 319 and accompanying text.

²⁸⁹ Our economic analysis of the DMCA rules does not include, as previous sections have done, a table setting forth the social calculus for the DMCA restrictions vs. no DMCA rules or other alternatives because there are simply too many complexities for one table to reflect all the economic considerations and how they interact. For example, in assessing how the DMCA rules affect incentives to innovate, it would seem necessary to measure the incentives of copyright owners whose works are technically protected, but what

The problem with this scenario—that users will prefer purchase to circumvention as a way to get access to content—is that circumvention only needs to be done once to produce market-destructive consequences. Widespread infringement can occur if a single reverse engineer develops and generally distributes a circumvention tool to unskilled users.²⁹⁰ As Bruce Schneier succinctly puts it, “automation allows attacks to flow backwards from the more skilled to the less skilled.”²⁹¹ The user’s cost of illegal access is then the price of getting it from the reverse engineer, rather than the cost of reverse engineering the technical measure oneself. In our view, this possibility threatens providers of digital products.

What, then, does the work of protecting digital content? Surely it is not the DMCA’s rule against individual acts of circumvention. Most individuals have neither the capability nor inclination, and, in any case, circumvention will be too costly and is likely to happen in nonpublic arenas, making enforcement difficult or impossible.²⁹² What does the work is the rule against *widespread distribution* of circumvention tools.²⁹³ Such a prohibition will not only keep circumvention tools from the masses, thereby preventing many infringements of copyrighted works that would otherwise take place in private, but the more public nature of commercial-scale manufacture and distribution makes this activity more susceptible to meaningful regulation.²⁹⁴ It is worth noting that the original White Paper proposal did not recommend legislation to outlaw acts of circumvention, but only to stop the manufacture and distribution of circumvention tools.²⁹⁵

about incentives to develop technical protection measures, and if so, do we only consider the incentives of those who have already developed such measures or those who might develop similar but stronger measures if allowed to reverse engineer their competitors’ products? Similar complexities attend assessment of the impact of the DMCA on follow-on innovation, price and wasted costs. Hence, while the criteria for assessing social welfare still apply, the analysis of these factors vis a vis the DMCA rules does not lend itself to a tabular format. In this and the following subsection, the text discusses some of the economic impacts of the DMCA rules with these criteria in mind.

²⁹⁰ One of us (Samuelson) wishes to point out that even when circumvention technologies are available, it may be costly in time and energy to use them for infringing purposes, even if one is inclined to do so, whereas the other of us thinks the economics of the DMCA must be assessed as though the time and energy required to use a circumvention tool was costless. It is somewhat instructive, though, that notwithstanding the widespread availability of DeCSS, sales of DVD movies remain very strong and the motion picture plaintiffs in the *Reimerdes* case were unable to identify a single act of infringement of their movies attributable to the use of DeCSS.

²⁹¹ Schneier, *supra* note 259.

²⁹² See, e.g., James R. Davis, *On Self-Enforcing Contracts, the Right to Hack, and Willfully Ignorant Agents*, 13 Berkeley Tech. L.J. 1145, 1147 (1998) (pointing out that most users won’t have sufficient skill to circumvent to make fair uses). See also Richard J. Gilbert & Michael L. Katz, *When Good Value Chains Go Bad: The Economics of Indirect Liability for Copyright Infringement*, 52 Hastings L.J. 961, 982 (2001) (noting difficulties with regulating acts of circumvention).

²⁹³ In the next subsection, we explain why development of a tool to engage in circumventions that do not have market-destructive activities should be permissible. From an economic standpoint, the ability of users to circumvent for legitimate purposes would not only permit these legitimate activities, but would also provide greater incentives for content owners to use stronger encryption. The DMCA rules now so insulate content owners from circumvention that they have little incentive to adopt better technical measures since any technical protection system they adopt, however weak, is protected by the DMCA.

²⁹⁴ Gilbert & Katz, *supra* note 292, at 982-83.

²⁹⁵ See White Paper, *supra* note 262, appendix 1 at 6.

C. Collateral Damage of Overbroad DMCA Rules

The DMCA does far more than establish a rule against trafficking in circumvention tools. Other aspects of the DMCA rules are not only inessential, but harmful. We consider here three types of collateral damage that flow from overbroad DMCA rules. First, the DMCA's anti-circumvention rules impede encryption and computer security research that are essential to the long-term societal interest in development of more secure systems. Second, the DMCA rules have been misused to harm competition and innovation and facilitate collusion. Third, purchasers of digital content are harmed because the DMCA rules preclude fair uses of copyrighted works and inhibit other activities that do not harm content providers.

The chilling effects of the DMCA on encryption and security research have already surfaced after the arrest of Russian programmer Dmitri Sklyarov who wrote a program capable of bypassing a technical protection measure in Adobe's e-book software²⁹⁶ and threats of litigation against Princeton computer scientist Edward Felten and his colleagues after he and his colleagues wrote a paper about flaws they discovered in digital watermarks that the recording industry planned to use to protect digital music.²⁹⁷ Although the DMCA has exceptions for encryption and computer security

²⁹⁶ See, e.g., Robert Lemos, FBI nabs Russian expert at Def Con, ZDNet News, July 17, 2001, <http://www.zdnet.com/zdnn/stories/news/0,4586,5094266,00.html>. Instead of fixing the flaw in its software, Adobe complained to the Justice Dept. and asked them to prosecute Dmitri Sklyarov, a Russian citizen who wrote the software in Russia (where development of such software is apparently legal) while he was in the U.S. at a conference.

²⁹⁷ In September 2000 the Secure Digital Music Initiative (SDMI) issued a public challenge inviting skilled technologists to defeat digital watermarking technologies that SDMI had selected as candidate standards for protecting digital music. See "An Open Letter to the Digital Community" available at http://www.sdmi.org/pr/OL_Sept_6_2000.htm. SDMI offered to pay successful hackers \$10,000 per broken watermark. Princeton computer scientist Edward Felten and his colleagues decided to accept this challenge, although not to seek the prize money because SDMI was only willing to award the money to those who agreed not to reveal how they defeated the watermarks to anyone but SDMI. Felten and his colleagues instead wrote a paper for a scientific workshop on the results of their research about the SDMI watermarks. The paper was entitled "Reading Between the Lines: Lessons From the SDMI Challenge" and was scheduled for presentation at the Fourth International Information Hiding Workshop in Pittsburgh, Pennsylvania, on April 26, 2001. For further details, see SDMI challenge FAQ at <http://www.cs.princeton.edu/sip/sdmi/faq.html>. An executive from Verance, the developer of one of the candidate technologies, and the Recording Industry Association of America found out about the paper and asked Felten to omit certain details about the weaknesses of the SDMI technologies. Felten and his coauthors decided that these details were necessary to support their scientific conclusions. SDMI and RIAA asserted that presentation of the paper at the conference or its subsequent publication in the conference proceedings would subject Felten, his coauthors, members of the program committee, and their institutions to liability under the DMCA, and made clear their intent to take action against the researchers unless they withdrew the paper. RIAA's theory is that the presentation of the paper constitutes distribution of a circumvention tool in violation of 1201(b)(1). A copy of the RIAA letter to Professor Felten asserting that presentation or publication of the researchers' paper would violate the DMCA is available at <http://cryptome.org/sdmi-attack.htm>. Although convinced that they would be vindicated if the matter went to court, Felten and his coauthors reluctantly withdrew the paper from the April conference out of concern about the high costs of litigation. See announcement at <http://cryptome.org/sdmi-attack.htm>. Felten's decision was widely reported in the press. See, e.g., David P. Hamilton, Professor Savors Being in the Thick of Internet Rows, Wall St. J., p. B1, June 14, 2001; Charles C. Mann, *Secure-Music Group Threatens*

research, neither seem to apply to Sklyarov or Felten.²⁹⁸ Prominent cryptographers have characterized these exceptions as “so parsimonious as to be of little practical value” as well as being based on a “fundamentally mistaken conception of cryptographic science.”²⁹⁹ The exception only applies, for example, if the researcher is employed or has been trained as a cryptographer, even though some brilliant breakthroughs in this field have come from amateurs.³⁰⁰ The researcher must also seek permission from affected rightsholders before trying to reverse engineer encryption technology.³⁰¹ Moreover, it requires that the researcher prove necessity.³⁰² And the exception may be unavailable if the researcher publishes his or her results on the Internet, because this will make them accessible to potential pirates.³⁰³

The more fundamental point is that “the science of cryptography depends on cryptographers’ ability to exchange ideas in code, to test and refine those ideas, and to challenge them with their own code. By communicating with other researchers and testing one another’s work, cryptographers can improve the technologies they work with, discard those that fail, and gain confidence in technologies that have withstood repeated testing.”³⁰⁴ Further evidence that the DMCA does not provide researchers enough latitude can be found in a recent report of the National Academy of Sciences which observes that “[r]egulating circumvention must be done very carefully, lest we hobble the very process that enables the development of effective protection technology.”³⁰⁵ This

Researchers Who Plan to Publish on Hacking Success, Inside Mag., 4/22/01, available at <http://www.inside.com>. The Electronic Frontier Foundation agreed to represent Felten and his coauthors in an affirmative challenge to the RIAA and SDMI claim which seeks a judicial declaration that the paper does not violate the DMCA so that Felten can present the paper at a conference in August 2001. See Felten v. RIAA Complaint, available at http://www.eff.org/Legal/Cases/Felten_v_RIAA/20010606_eff_complaint.html.

²⁹⁸ 17 U.S.C. sec. 1201(g), 1201(j). Felten and his coauthors seem ineligible for the encryption research exception because the SDMI watermarks do not use encryption. The watermarks are also not access controls, and so the computer security testing exception is inapplicable because it only permits making a tool to bypass an access control under 1201(a)(2), not making a tool to bypass other controls under 1201(b). Neither privilege applies to 1202 claims. Sklyarov does not qualify for either exception, even though he is a trained cryptographer, because the firm for which he works has sold copies of the bypassing software over the Internet, thereby distributing the tool beyond the scope of the exception.

²⁹⁹ Brief of Amici Curiae of Dr. Steven Bellovin, Dr. Matt Blaze, Dr. Dan Boneh, Mr. Dave Del Torto, Dr. Ian Goldberg, Dr. Bruce Schneier, Mr. Frank Andrew Stevenson, & Dr. David Wagner, in *Universal City Studios, Inc. v. Reimerdes*, to the Second Circuit Court of Appeals, Jan. 26, 2001, available at <http://eon.law.harvard.edu/openlaw/DVD/NY/appeal/000126-cryptographers-amicus.html> (cited hereinafter as “Bellovin Amicus”). Problems with the overly narrow and ambiguous encryption and computer security exceptions to the DMCA are discussed in *Digital Dilemma*, supra note 257, at 174-75, Appendix G (2000).

³⁰⁰ 17 U.S.C. sec. 1201(g)(3)(B).

³⁰¹ 17 U.S.C. sec. 1201(g)(2)(C). The computer security exception requires that the researcher actually get, and not just ask for, permission to defeat the technical protection measure. 17 U.S.C. sec. 1201(j)(1).

³⁰² 17 U.S.C. sec. 1201(g)(1), (g)(2)(B).

³⁰³ 17 U.S.C. sec. 1201(g)(3)(A). The encryption researcher must also provide affected copyright owners with the results of his or her research in a timely manner. 17 U.S.C. sec. 1201(g)(3)(D).

³⁰⁴ Bellovin Amicus., supra note 299.

³⁰⁵ *Digital Dilemma*, supra note 257, at 173.

report identifies some key ambiguities in the DMCA's anti-circumvention rules that put encryption and computer security researchers at risk.³⁰⁶

Encryption and computer security cannot get stronger if researchers are at risk of liability under the DMCA in the ordinary course of their research.³⁰⁷ As we argued for SCPA, reverse engineering facilitates competition for improvements. The right balance between facilitating improvements and protecting innovators can be achieved by granting a kind of "leading breadth" to each innovation,³⁰⁸ but not by prohibiting researchers from sharing prior knowledge, as the DMCA does.

In addition to its chilling effects on innovation in the encryption and computer security research fields, the DMCA may have anticompetitive effects in both the market for technical protection systems and the markets for digital products. The DMCA creates trade-secret-like protection for technical protection systems far beyond that provided by any other law. Ordinarily, an unpatented product would be vulnerable to reverse engineering and competition. As we argued for traditional manufacturing, the vulnerability of unpatented products to reverse engineering limits market power in a competitively healthy way. In contrast, the DMCA's rule against reverse engineering deprives consumers of the benefits of competition in creating and marketing technical protections. The DMCA rules effectively insulate makers of technical protection measures from competitive reverse analysis.

As regards the market for digital information products, the DMCA extends the market power of content providers in ways that are harmful to competition. In the past three years, plaintiffs have asserted violations of the DMCA rules in order to exclude competitors from the marketplace,³⁰⁹ to control the market for complementary products,³¹⁰ and to facilitate their preferred market allocation and pricing strategies.³¹¹

³⁰⁶ See, e.g., *id.*, App. G (discussing ambiguities and other problems with the DMCA anti-circumvention provisions).

³⁰⁷ See, e.g., Andrew W. Appel & Edward W. Felten, *Technological Access Control Interferes with Bellovin Noninfringing Scholarship*, 43 Comm. ACM 21 (Sept. 2000); Bellovin Amicus, *supra* note 299; Pamela Samuelson, *Anti-Circumvention Rules Threaten Science*, 293 Science 2028 (Sept. 2001).

³⁰⁸ See, e.g., O'Donoghue, Scotchmer, & Thiesse, *supra* note 131.

³⁰⁹ See, e.g., *Sony Computer Entertainment America Inc. v. Gamemasters*, 87 F. Supp.2d 976, 982 (N.D. Cal. 1999)(successful 1201 claim against Game Enhancer software that competed with Sony's Game Shark software). Sony Computer Entertainment also asserted anti-circumvention claims against Connectix, Inc. and Bleem, Inc. because both firms make emulator programs that did not read the anti-copying technology in Sony games. These emulator programs compete with PlayStation in the platform market. See Samuelson, *supra* note 271, at 556-57 (discussing Sony's anti-circumvention claim against Connectix). See also Testimony of Jonathan Hangartner, attorney for Bleem, Inc., at Copyright Office Hearings on Anti-Circumvention Rules, at 224-32, held at Stanford University, May 19, 2000, available at <http://www.loc.gov/copyright/1201/hearings/1201-519.rtf> (discussing Sony's anti-circumvention claim against Bleem and implications of 1201(a)(1)(A) going into effect for future Sony anti-circumvention claims against Bleem).

³¹⁰ *Gamemasters*, 87 F. Supp.2d at 987-88 (Game Enhancer software that interoperated with Sony PlayStation games held to violate 1201 because it bypassed Sony country coding); *RealNetworks, Inc. v. Streambox, Inc.*, 2000 U.S. Dist. LEXIS 1889 (W.D. Wash. 2000) (Streambox VCR software, designed to interoperate with RealNetworks software, held to violate the anti-circumvention rules). See also

One commentator, disturbed by this trend, recommends that development of a concept of misuse of DMCA rights akin to the misuse doctrines of patent and copyright law doctrines to thwart competitively harmful activities.³¹²

Joint ownership or adoption of a technical protection system by major content providers may also facilitate collusion and the ability of content providers to leverage their market power as to content to control equipment and player software markets.³¹³ The motion picture industry, for example, was able to achieve significant control over the DVD player industry by its joint ownership of patent rights in components necessary to manufacture of DVD players, a license for which requires, among other things, the installation of CSS in all players.³¹⁴ This mode of achieving standardization allows the motion picture industry to coordinate on pricing and other terms for their products as well as controlling a significant aspect of the market for equipment. More recently, the recording industry has sought to leverage its market power over digital music into the market for playing technologies through the Secure Digital Music Initiative (SDMI). The goal of SDMI is to develop standard digital watermarks for digital music, the presence of which must be detected before the music can be played, and standard technical measures for players.³¹⁵ In both examples, players and content become a “system” much like the operating systems and applications software discussed in Section IV, except that in the player-movie/music systems, non-collusive entry into the player market is foreclosed, in part because of the DMCA rules.³¹⁶ In the absence of legislation mandating installation

Reimerdes, 111 F. Supp.2d at 320 (giving no weight to claim that DeCSS was intended to enable development of Linux platform for playing DVD movies).

³¹¹ See, e.g., Michael Owen-Brown, Regulator challenges DVD zones, Australian Times (May 24, 2001), available at http://www.news.com.au/common/story_page/0,4057,2032464%255E421,00.html (Australian competition and consumer protection authorities investigating DVD country coding because of market allocation and discriminatory pricing impacts). See also <http://www.europa.eu.int/eur-lex/en/dat/2001/ce053/ce05320010220en01570157.pdf> and <http://www.europa.eu.int/eur-lex/en/dat/2001/ce053/ce05320010220en01580159.pdf> (EU competition concerns about country coding). See also <http://www.2600.com/news/display.shtml?id=541> (reporting on a raid of video stores in Gothenburg and Stockholm because they were selling imported DVD's with the “wrong” country code).

³¹² Professor Dan Burk of the University of Minnesota Law School has a work in progress on misuse of DMCA anti-circumvention rights. Personal email communication from Dan Burk to Pamela Samuelson, dated Sept. 19, 2001.

³¹³ As a close reading of *Reimerdes* reveals, any firm that wants to make a DVD player has to get a license to do so. *Reimerdes*, 111 F. Supp.2d at 337. Although the court asserted that such licenses are “available to anyone on a royalty-free basis and at a modest costs,” *id.*, such licenses, in fact, are only available “subject to strict security requirements,” *id.* at 310. This precludes an open source Linux player impossible. Any effort to develop an unlicensed platform would require reverse engineering of CSS (as well as make a tool to do so) which is almost certainly illegal under the DMCA, at least as interpreted in *Reimerdes*.

³¹⁴ *Reimerdes*, 111 F. Supp.2d at 319-20.

³¹⁵ For a description of the SDMI watermarks and their intended uses, see SDMI challenge FAQ at <http://www.cs.princeton.edu/sip/sdmi/faq.html>.

³¹⁶ In the software context, the market power constrained by reverse engineering constrained lies in the platform provider because of its control over APIs. In the digital entertainment context, the market power is chiefly wielded those who are rightsholders in the applications market. The economics of interoperability is the same, although the DMCA rules change the legal analysis significantly at least when firms want to develop alternative platforms to interoperate with digital data. The reverse engineering exception in 1201 only applies to program-to-program interoperability. See *Universal City Studios, Inc. v. Reimerdes*, 82 F. Supp.2d 211,217-18 (S.D.N.Y. 2000)(ruling that the 1201 reverse engineering exception

of technical controls,³¹⁷ collusion may be the only way to ensure that highly protected products will enjoy success in the marketplace.³¹⁸

Of course, entry by a non-colluding firm offering a different protection system would be difficult, even in the absence of DMCA rules, because of network effects. Content providers are unlikely to choose a new system over an established system with an installed base of customers. As in any market governed by network externalities, consumer choice among protection systems will thus be limited. Although the DVD or SDMI system may provide the cartel of system component providers with too much market power, there is at least this offsetting efficiency: Every buyer of a platform (e.g., a CSS-compatible DVD player) will have access to all the content produced rather than having to choose among incompatible alternatives.

Finally, we argue that the DMCA causes collateral harm to users of digital content in part because there are many reasons to circumvent technical protection that have nothing to do with copyright infringement. Nine of such reasons are already acknowledged in the statutory and rule-making exceptions to the DMCA rules, but others include:

- 1) detection of technical measures that may be hiding infringing copies of copyrighted works,
- 2) detection of technical measures that are being used to hide stolen trade secrets or other confidential information,
- 3) analyzing a virus program wrapped in a technical measure,
- 4) locating, assessing, and fixing bugs in software,
- 5) reverse analyzing software to integrate it with other system software,
- 6) creating backup copies of software or data,
- 7) understanding the internal design of a technical protection measure for research purposes,
- 8) preserving information (e.g., evidence of some illegal activity),
- 9) enabling the development of an alternative non-software platform,

does not apply to development of platforms to play DVD movies because DVD movies are not “programs”).

³¹⁷ Senators Hollings and Stevens have announced their intent to introduce legislation to mandate installation of technical protections in future digital technologies. See, e.g., Declan McCulloch, New Copyright Bill Heading to DC, *Wired News*, Sept. 7, 2001, available at <http://www.wired.com/news/politics/0,1283,46655,00.html> (discussing the Security Systems Standards and Certification Act that would mandate installation of standard technical protection measures in all interactive digital devices). Enactment of this legislation would foreclose the possibility of marketplace competition between protected and unprotected devices (except perhaps as regards used computers and other digital technologies).

³¹⁸ Given a choice, consumers generally prefer unprotected products to protected products in part because technical protection measures often make products more difficult and inconvenient to us. See, e.g., *Digital Dilemma*, supra note 257, at 87-88, 154. An example is the marketplace competition among software developers that led to the abandonment of copy-protection systems for software. See, e.g., Cohen, supra note 247, at 523-25 (discussing competitive demise of copy-protected software). Economists Carl Shapiro and Hal Varian assert that “[t]rusted systems, cryptographic envelopes, and other copy protection schemes have their place but are unlikely to play a significant role in mass-market information goods because of standardization problems and competitive pressures.” CARL SHAPIRO & HAL VARIAN, *INFORMATION RULES* 102 (1998).

- 10) enabling interoperability with data,
- 11) restoring a rightful copy after the crash of one's hard drive,
- 12) stopping surveillance of a licensee's business activities,
- 13) preventing technical "self-help" measures from being wrongfully invoked,
- 14) bypassing country codes in a product so one can play a DVD movie for which one has already paid the standard fee on one's DVD player,
- 15) bypassing controls that prevent users from fast-forwarding through a movie, and
- 16) making fair uses of copyrighted works (such as excerpting clips from technically protected movies to demonstrate that a particular word ("redskins") has been used in a derogatory fashion).³¹⁹

One judge in a recent DMCA case has observed that the fair uses excluded by technical protections are "remarkably varied."³²⁰ Although circumvention of copy-control measures is not directly illegal under the DMCA, the DMCA may nevertheless effectively preclude it insofar as this kind of reverse engineering requires use of a tool, the making of which would seem to be illegal under 1201(b)(1).

Contributing to the unreasonable restrictive effects of the DMCA rules is the fact that the term "access controls" for DMCA purposes has been construed so broadly that buyers can be deprived of rights to use content that they have legitimately acquired. For example, a court declared that the Content Scrambling System ("CSS"), used to protect movies on DVDs, is an access control.³²¹ As a consequence, purchasers of DVD movies cannot play them except on any device licensed by DVD-CCA, even if the purchaser of the movies would prefer to watch them on a Linux player, and then only on a device with the same country code as the movie. Another case characterized a country-coding scheme embedded in a mass-marketed videogame as an access control, thereby making illegal software that allowed lawfully purchased games to be used on a player with a

³¹⁹ Commentators differ in their views about the effects of the DMCA on fair uses. Some assert that Congress DMCA rules preclude fair uses. See, e.g., David Nimmer, *A Riff on Fair Use in the Digital Millennium Copyright Act*, 148 U. Penn. L. Rev. 673 (2000). Others find some basis in the DMCA for preserving fair uses as to technically protected works. See, e.g., Samuelson, *supra* note 271, at 540. See also Julie E. Cohen, *Copyright and the Jurisprudence of Self-Help*, 13 Berkeley Tech. L.J. 1089 (1998). Still others believe that the DMCA would be unconstitutional if it foreclosed fair uses. See, e.g., Benkler, *supra* note 283; Jane C. Ginsburg, *From Owning Copies to Experiencing Works*, in UNITED STATES INTELLECTUAL PROPERTY LAW (Hugh Hansen, ed., forthcoming 2001); Neil Netanel, *Locating Copyright Within the First Amendment Skein*, Stan. L. Rev. (forthcoming 2001). See also Lessig, *supra* note 286, at 132-38 (characterizing as latent ambiguity whether the U.S. Constitution requires limitations on copyright, such as fair use)

³²⁰ *Reimerdes*, 111 F. Supp.2d at 337-38 (giving examples). The judge concluded that the impact of the DMCA rules on fair use would be negative but "probably only to a trivial degree," *id.* at 337, because fair uses could be made of analog versions of movies, even if not of DVDs, and because some skilled technologists could make fair uses of DVD movies even if most people could not. An obvious flaw in the latter reason is that the skilled person would have to make a DeCSS equivalent in order to make fair uses of a DVD, which would seem to run afoul of section 1201(a)(2). It seems unreasonable to require a fair user to buy two copies of a movie instead of one to make fair uses or to relegate fair users to an inferior format. In addition, the former rationale ignores that analog VCR movies are also protected by technical measures and it would seemingly violate 1201(b) to make a tool to engage in fair uses of analog versions of movies.

³²¹ *Reimerdes*, 111 F. Supp.2d at 317-18.

different country code.³²² Without taking a position on the legitimacy of country-coding,³²³ we wish to point out that these kinds of access controls dramatically alter buyers' rights, and we think further debate on these issues are warranted.

D. Alternatives to the DMCA Anti-Circumvention Rules

The WIPO Copyright Treaty calls for “adequate” protection against circumvention of technical protection,³²⁴ not the excessive protection provided by the DMCA. In this subsection, we identify a number of alternative ways to address the economic concerns underlying the DMCA rules that are more consistent with other social goals, such as promoting innovation and competition.³²⁵

One proposed less restrictive alternatives to the DMCA has been suggested by two law professors who point out that the DMCA could have been modeled on the Audio Home Recording Act (AHRA) which offers a better balancing of interests.³²⁶ AHRA requires installation of serial copy management system (SCMS) chips in consumer-grade digital audio taping equipment.³²⁷ SCMS allows users to make usable first-generation copies of digital content, although not perfect copies from which perfect copies can be made. This solution liberates consumers from some of the inconveniences of technical protections by allowing them to make fair uses. A similar requirement might have been adopted in the DMCA.

A second less restrictive alternative to the DMCA, also drawn from the AHRA, is the proposal of one commentator to levy taxes on copying technologies and storage media akin to those now levied on consumer-grade DAT machines and tapes that then provides a pool of money with which to compensate content owners for digital copies.³²⁸

³²² Sony Computer Entertainment America Inc. v. Gamemasters, 87 F. Supp.2d 976 (N.D. Cal. 1999).

³²³ There is an international debate about whether the sale of intellectual property products in one nation should “exhaust” the rightsholders’ exclusive distribution rights or whether these rights should be exhausted only in the nation or region in which they were sold. Country codes embedded in software, games or DVDs are designed to be a technical means of preventing an international market in intellectual property goods beyond the market controlled by rightsholders. The economics of international vs. national or regional exhaustion are complex and as yet unresolved. For a discussion of the issues, see, e.g., Vincent Chiappetta, *The Desirability of Agreeing to Disagree: The WTO, TRIPS, International IPR Exhaustion, and A Few Other Things*, 21 Mich. J. Int’l L. 333 (2000).

³²⁴ WIPO Copyright Treaty, supra note 267, art. 11.

³²⁵ Other less restrictive alternatives to the DMCA besides those discussed here are discussed in the Letter Brief By Order of the Court, submitted to the Hon. Jose Cabranes et al., by the Electronic Frontier Foundation, May 30, 2001, at 11-14. While the EFF’s analysis of less restrictive alternatives aims to demonstrate that the DMCA rules are unconstitutional restrictions on speech, less restrictive alternatives are also of significance for an economic analysis of the DMCA rules. If less restrictive rules would fulfill statutory objectives without collateral damage to competition and innovation, it would be more economically sensible to adopt the narrower rules.

³²⁶ See Brief Amicus Curiae of Yochai Benkler and Lawrence Lessig to the Second Circuit Court of Appeals in *Universal City Studios, Inc. v. Reimerdes* (Jan. 26, 2001), available at http://www.eff.org/IP/DVD_cases/200110126_ny_2profs_amicus.html.

³²⁷ 17 U.S.C. sec. 1002.

³²⁸ See Glynn S. Lunney, *The Death of Copyright: Digital Technology, Private Copying and the DMCA*, Va. L. Rev. (forthcoming 2001). The AHRA rules on royalty payments on equipment and tapes can be

A third less restrictive alternative that has been suggested is that the DMCA be revised to require copyright owners who want the benefits of the DMCA anti-circumvention rules to put keys to unlock their technically protected digital content in escrow so that fair users can apply for access to the keys when they wish to exercise fair use rights.³²⁹

Fourth, Congress might ban circumvention for purposes of infringing copyrights or facilitating infringement.³³⁰ A general ban on technologies capable of circumvention, in this view, is unjustifiable if the technologies have substantial lawful uses.

Fifth, Congress might add a general purpose exception to the act of circumvention rules so that courts will be able to limit the reach of the DMCA in appropriate cases and recognize a legal right to make a tool in order to engage in circumventions that do not harm copyright owners.³³¹

The adaptation to the DMCA most consistent with the analysis in this article is that the law should maintain a prohibition on trafficking in circumvention tools, but not necessarily on the development of such technologies for personal or firm-internal use in view of the many reasons for engaging in circumventions that have nothing to do with infringement of copyrights.

We urge further study of less restrictive alternatives to the DMCA anti-circumvention provisions.

VI. Reverse Engineering as a Policy Lever

All intellectual property rights regimes—utility patent, plant variety protection, copyright, and SCPA—have certain policy levers in common, wielded to a greater or lesser extent. All establish, for example, a length of protection, a breadth of protection (sometimes legislated and sometimes evolving through caselaw interpretations), and some fair use or policy-based limitations on the scope of protection. By wielding the available policy levers appropriately, legal regimes can be made sensitive to the technological and industrial contexts they regulate so as to avoid either over-rewarding or under-rewarding innovators.

We conceive of the legal status of reverse engineering as one such policy lever. This policy lever is set differently in different legal contexts. Trade secrecy law, for example, exposes innovators to reverse engineering whereas patent law limits it.³³² A

found at 17 U.S.C. secs. 1002-1007. The Copyright Office collects and administers the funds collected thereby and distributes them to the appropriate rightsholders. *Id.* at secs. 1006-1007.

³²⁹ Dan L. Burk and Julie E. Cohen, *Fair Use Infrastructure for Copyright Management Systems*, Harv. J. L. & Tech. (forthcoming 2001).

³³⁰ See, e.g., H.R. 3048, 105th Cong., sec. 8 (1997).

³³¹ Samuelson, *supra* note 271, at 546-51.

³³² See *supra* notes 31-32 and accompanying text.

rationale for this difference lies in the disclosure obligations that patent law already imposes on innovators that trade secret owners avoid. For the traditional subject matters of copyright law—artistic and literary works—reverse engineering has not been an issue because viewers and readers did not need to reverse engineer these works to understand them. Yet as copyright’s subject matter expanded to include computer software, reverse engineering became a significant policy issue in copyright law as well.³³³

The optimal setting for any given policy lever depends in part on how the other levers are deployed.³³⁴ Consider, for example, the interaction of reverse engineering rules and the length of protection. Outlawing decompilation of computer programs is inadvisable in part because of the long duration of protection that copyright provides to computer programs.³³⁵ If decompilation and disassembly were illegal, computer programs would be immune from an important source of competition for almost a century, which would likely impede innovation in the software industry. Such a rule would provide far more protection than necessary to protect innovative software firms against market-destructive appropriations.

Our study of reverse engineering in various industrial contexts leads us to two general conclusions. The first is that reverse engineering is generally a competitively healthy way for second comers to get access to and discern the know-how embedded in an innovator’s product. If reverse engineering is costly and takes time, as is usually the case, innovators will generally be protected long enough to recoup R&D expenses. More affirmatively, the threat of reverse engineering promotes competition in developing new products and constrains market power as well as inducing licensing that enables innovators to recoup R&D costs.

Second, we have found it useful to draw a distinction between the act of reverse engineering, which is generally performed to obtain know-how about another’s product, and what the reverse engineer does with the know-how thereby obtained (e.g., designing a competing or complementary product). The act of reverse engineering rarely (if ever) has market-destructive effects. Harmful effects are far more likely to result from post-reverse engineering activities (e.g., selling a competing product made with know-how from an innovator’s product). Because of this, it may be more sensible to focus regulatory controls on market-destructive post-reverse engineering activities than to regulate reverse engineering as such, especially given that reverse engineering aims at discovery of know-how that often leads to technical advance. Acts of reverse engineering are, moreover, typically nonpublic and difficult to detect, whereas post-reverse engineering activities (such as introducing competing or complementary products

³³³ See supra Section IV-A.

³³⁴ There is an extensive economics literature on the interdependency of intellectual property policy levers. Most saliently, economics scholars have addressed the interaction of length and breadth. See Richard Gilbert and Carl Shapiro, *Optimal Patent Length and Breadth*, 21 RAND J. Econ. 106 (1990); Paul Klemperer, *How Broad Should the Scope of Patent Protection Be?*, 21 RAND J. Econ. 113 (1990), Nancy T. Gallini, *Patent Length and Breadth with Costly Imitation*, 24 RAND J. Econ. 52 (1992), and Maurer & Scotchmer, supra note 51, for the static context, and O’Donoghue, Scotchmer and Thisse, supra note 131, for the cumulative context.

³³⁵ We agree with Graham and Zerbe on this point. See Graham & Zerbe, supra note 151, at 128-31.

to the market) are more likely to be public and hence more susceptible to regulatory control. In the discussion below, we distinguish regulatory strategies aimed at acts of reverse engineering and those that regulate post-reverse engineering activities.

The bluntest way to deploy the reverse-engineering lever is to switch it “on” (making it legal) or “off” (making it illegal). Our study has revealed five more nuanced ways to deploy this lever: 1) regulating a particular means of reverse engineering, 2) establishing a breadth requirement for subsequent products, 3) purpose- and necessity-based requirements for judging the legitimacy of reverse engineering, 4) regulating reverse engineering tools, and 5) restricting publication of information discovered by a reverse engineer.³³⁶

We review these options in subsection A, for two reasons. First, they have been adopted in some industrial contexts and should be assessed for their economic reasonableness. Second, proposals for additional restrictions on reverse engineering may be made in the future. Legal decisionmakers may be better equipped to respond to such proposals if they understand how reverse engineering has been regulated in the past and in what senses, if any, restrictions on reverse engineering are justifiable.

In subsection B, we observe that the existence of a legal right to reverse engineer may be so threatening to some innovators that they will endeavor to render the legal right moot through one of two strategies: 1) by requiring customers to agree not to reverse engineer their products, or 2) by configuring their products to make reverse engineering extremely difficult or impossible. Legal decisionmakers have the option of responding to such efforts by deciding not to enforce such contractual restrictions or by forcing disclosure of product know-how.

A. Ways to Regulate Reverse Engineering

1. Regulating a Market-Destructive Means of Reverse Engineering

³³⁶ Our study also uncovered four other proposals to regulate reverse engineering in the software industry. One proposal was to allow decompilation or disassembly if done through a “clean room” process, that is, by separating the team assigned to reverse engineer another firm’s program from the team that uses information provided by the first team in developing a new program. See, e.g., Laurie & Everett, *supra* note 151. Second, one decision would have allowed reverse engineering of a program for purposes of achieving present compatibility with the other firm’s software, but not for purposes of achieving future compatibility. See *Atari Games Corp. v. Nintendo of Am., Inc.*, 975 F.2d 832 (Fed. Cir. 1992). Third, the legality of the second comer’s reverse engineering efforts has sometimes undermined by a “taint” in the last stages of the reverse engineering process, as when defendant’s lawyers lied to the U.S. Copyright Office in order to get a copy of plaintiff’s source code that the innovator had filed when registering its claim of copyright. In essence, the Atari Games’ engineers’ efforts to reverse engineer from the code did not yield enough information, so the lawyers were sent to get source code listings on file in the Copyright Office so that they could get the additional information they needed to make compatible games. This inequitable conduct affected the court’s view on Atari’s fair use defense. *Id.* Fourth, the European Software Directive seems to give weight to establishing a “paper trail” to show the legitimacy of reverse engineering of software may be important. See Czarnota & Hart, *supra* note 163, at 84.

When a particular means of reverse engineering makes competitive copying too cheap, easy and rapid, innovators may be unable to recoup R&D expenses. If so, it may be reasonable to regulate that means. Anti-plug mold laws, discussed in section II, illustrate this principle. Using a competitor's product as a "plug" with which to make a mold from which to make competing products permits competitive copying that is so cheap and fast as to undermine the incentives to invest in designing an innovative product. Restrictions on plug molding may restore adequate incentives to make such investments. Notwithstanding the Supreme Court's characterization of plug-molding as an efficient means of reverse engineering,³³⁷ we suggest that plug molding is better understood as an efficient means of reimplementing the original innovation. Plug molding has the potential to undermine an innovator's incentives without any offsetting social benefit of follow-on innovation because a plug-molder does not aim to learn anything that might lead to further innovation. Thus, one of the key benefits of reverse engineering will be lost if plug-molding is utilized to make competing products.

Another type of act of reverse engineering that some industry participants argued should be illegal was decompilation and disassembly of computer programs, discussed in Section IV. This argument was based on the fear that second comers would appropriate valuable internal design elements of programs. Decompilation and disassembly were eventually accepted as legal, in part because they require so much time, money and energy that the original developer is not significantly threatened. If reverse engineering actually occurs in face of these costs, it may enable the development of competitive interoperable products and erode the market power of industry leaders in a competitively healthy way.

A third example of regulating a particular means of reverse engineering discussed in this study was the DMCA anti-circumvention rule that outlaws circumvention (that is, reverse engineering) of technical measures that control access to copyrighted works. This too is a direct regulation of acts of reverse engineering. Although this rule has some exceptions (e.g., to enable program-to-program interoperability), Section V explains why that ban is nevertheless too restrictive.

Our advice to policymakers is this: Before banning a means of reverse engineering, require convincing evidence that this means has market-destructive consequences. Realize also that existing market participants may be seeking a ban mainly because they wish to protect themselves against competitive entry. Any restriction on reverse engineering should be tailored so that it does not reach more than parasitical activities. For example, it may be sensible not to make the restriction retroactive, to require that innovations embody some minimal creativity, and/or to limit the duration of the ban. Another possibility is to outlaw market-destructive reimplementations of innovations, rather than banning reverse engineering as such. Alternatively, reverse engineers could be required to compensate rightsholders for research uses of the innovation aimed at development of follow-on innovation.³³⁸

³³⁷ *Bonito Boats*, 499 U.S. at 160.

³³⁸ See, e.g., Eisenberg, *supra* note 34 (proposing compensation for research uses of patented research tools); Mueller, *supra* note 34 (accord); Manifesto, *supra* note 14 (proposing liability rule for reuses of

2. A Breadth Requirement For Products of Reverse Engineering

Another policy option is to establish a breadth requirement for products of a reverse engineering process that limits how the results of reverse engineering can be used thereafter.³³⁹ If second comers must invest in some forward-engineering and not simply free-ride on the previous innovation by copying it exactly, the second comer's efforts are more likely to advance the state of technology, as well as to extend the second comer's development cycle so that the earlier innovator is still protected. The Semiconductor Chip Protection Act was our principal example.³⁴⁰ SCPA permits the layout of circuits to be copied for purposes of study and analysis, as well as permitting reuse of some of the know-how discerned in the reverse engineering process. This is a useful boost to competitors designing integrated circuits. However, SCPA requires reverse engineers to design an "original" chip rather than simply making a clone or near-clone of the integrated circuit that was reverse engineered.³⁴¹

Since SCPA allows later innovators to learn from earlier ones, while still allowing chip designers to recoup expenses, we think it is competitively healthy. More generally, we find merit in the idea of establishing a breadth requirement to ensure that reverse engineering leads to further advance, while still preserving enough market power so that innovator recoups costs, in markets where cloning the innovator's product will be market-destructive.³⁴² Again, policy makers should be wary of undocumented claims that reverse engineering is *per se* destructive.³⁴³

While most legal regimes do not link the legitimacy of reverse engineering with technical advance, the case law on software copyright may implicitly do so. In *Sega v. Accolade*, for example, the court's perception that Accolade's reverse engineering was legitimate rested in no small part on the defendant's having developed a new, noninfringing program that promoted the very kind of progress that copyright law was

industrial compilations of applied know-how in software); Reichman, *Green Tulips*, supra note 50 (proposing liability rules for subpatentable innovation).

³³⁹ The notion of "breadth" has no formal meaning in law. However, the economics literature has interpreted breadth in the cumulative context as measuring the amount by which a product is improved. See, e.g., O'Donoghue, Scotchmer and Thisse, supra note 131, and Jerry Green and Suzanne Scotchmer, *On the Division of Profit between Sequential Innovators*, 26 RAND J. Econ. 20 (Spring 1995).

³⁴⁰ A similar rule exists in the Plant Variety Protection Act, 7 U.S.C. sec. 2541, 2544. Use of a protected variety to develop a new variety is non-infringing as long as the subsequent variety itself qualifies as a distinct variety that qualifies for PVPA protection. In 1994 Congress limited application of this rule so that if the subsequent variety retains virtually the whole genetic structure of the earlier variety, the subsequent variety may infringe. See, e.g., Peter J. Gross, *Guiding the Hand that Feeds: Toward Socially Optimal Appropriability in Agricultural Biotechnology Innovation*, 84 Calif. L. Rev. 1395 (1996).

³⁴¹ SCPA's reverse engineering privilege may be instructive even if SCPA itself is flawed or no longer necessary for reasons discussed supra Section III-C.

³⁴² There are, of course, important issues about how much progress should be required for the new product to be permissible, but the basic principle is sound: By prohibiting clones, but permitting reverse engineering to make improved products, each innovator is protected for some period against horizontal competition, but must eventually give way to a better product.

³⁴³ See supra Section II-C.

intended to bring about.³⁴⁴ Nevertheless, a linkage between the legitimacy of reverse engineering and a breadth requirement in the software industry may be unnecessary for two reasons: First, decompilation and disassembly of programs are so difficult and time-consuming that second comers generally do not find it profitable to develop market-destructive clones in this way.³⁴⁵ Second, reverse engineering of software does not generally lead to the development of a competing product, but rather to the development of interoperable programs or to the fixing of software “bugs.” Breadth requirements seem most appropriate when the goal is development of a competing product.

3. Purpose- And Necessity-Based Criteria for Determining the Legitimacy of Reverse Engineering

A third policy option for regulating reverse engineering was to judge its legitimacy based on the purpose for which it was conducted and whether such reverse engineering was necessary.³⁴⁶ Under this approach, reverse engineering may be permitted for some purposes, but not for others. Disassembly of computer software, for example is fair use of the program if done for a legitimate purpose such as achieving interoperability,³⁴⁷ although not for other purposes. The same approach recurs in the DMCA where reverse engineering of access controls is permissible for some purposes, but not others.³⁴⁸ It is worth noting that the legitimacy of reverse engineering has traditionally not depended on its purpose or necessity. For traditional manufactured items, the right to reverse engineer has been virtually absolute.

We have mixed reactions to purpose- and necessity-based criteria for regulating reverse engineering. Of course it is true that the economic effects depend on the purpose, and purpose-based reasoning is not unheard of in other areas of law. In intellectual property law, a second comer’s purpose may well determine whether he or she qualifies for an exception to or limitation on intellectual property rights.³⁴⁹ Copyright’s fair use doctrine, for example, gives considerable weight to the purpose of a fair-use claimant’s activities.³⁵⁰

On the positive side, purpose- or necessity-based limits on reverse engineering may sometimes protect developers against infringements that are difficult to detect. They may also avoid wasteful expenditures on reverse engineering undertaken for harmful purposes. Purpose- and necessity-based limits may also induce licensing or voluntary disclosure of know-how (e.g., interface information) in order to obviate the second

³⁴⁴ See supra Section IV-A.

³⁴⁵ To the extent decompilation results in an infringing program, copyright law already provides an adequate remedy. See, e.g., *E.F. Johnson Co. v. Uniden Corp. of Am.*, 623 F. Supp. 1485 (D. Minn. 1985).

³⁴⁶ Judging the legitimacy of reverse engineering based on the person’s purpose for engaging in the activity or on the necessity of the reverse engineering would seem, in theory, distinct mechanisms for regulating reverse engineering. Because these two criteria have been linked in the regulation of reverse engineering in the software industry, we will treat them together in this subsection.

³⁴⁷ *Sega*, 977 F.2d at 1522.

³⁴⁸ 17 U.S.C. sec. 1201 (d)-(i).

³⁴⁹ See supra notes 34, 151, 340 and accompanying texts (discussing the experimental use defense in patent law, the research exception in PVPA, and the fair use defense in copyright law).

³⁵⁰ 17 U.S.C. sec. 107 (preamble, subsec. 1).

comer's need for reverse engineering.³⁵¹ Licensing will help the innovator recoup its R&D expenses and voluntary disclosure may be a reasonable tradeoff because the second comer will still have to make substantial investments to make a commercially viable new product.

However, purpose- and necessity-based limits on reverse engineering pose other problems. A strict necessity requirement, for example, may become a trap for the unwary. A rational second comer would almost always prefer to get interface or other useful information from public sources in order to save itself unnecessary expenses of reverse engineering. If such a person has a reasonable belief that the desired information is not publicly available, he or she should not be punished later merely because the information turned out to be publicly available in some remote place that he or she neglected to check.

In addition, it may be difficult to judge the legitimacy of reverse engineering based on purpose may be problematical for a number of reasons. Purpose may, for example, be hard to discern and much in dispute, leading to wasteful litigation. Antitrust law faces similar difficulties, as when a court must decide whether a certain defendant (say, Microsoft) engaged in certain acts for good purposes (e.g., integrating its browser into the operating system to benefit consumers) or bad purposes (e.g., trying to put Netscape out of business).³⁵² In addition, reverse engineers may have multiple purposes, some of which may be more "legitimate" than others.

In assessing purpose-based approaches to judging the legitimacy of reverse engineering, it is useful to compare the open-ended approach of *Sega v. Accolade* with the closed approach of the DMCA. The latter prohibits reverse engineering except in specifically enumerated circumstances,³⁵³ each of which depends on the purpose for which circumvention was undertaken. The *Sega v. Accolade* approach is more flexible, yet more uncertain. The DMCA is more certain, but inflexible. Given that the DMCA's principal objective is to prevent piracy, it would have been more straightforward to prohibit reverse engineering for purposes of infringing the copyrighted work or for enabling such infringement, but not necessarily otherwise. The DMCA's presumptions run in the opposite direction: reverse engineering of certain technical measures is deemed illegal unless specifically exempted by the statute. Thus a host of reasonable circumventions must be presumed illegal,³⁵⁴ and those who reverse engineers for these purposes can only hope that their activities will escape the notice of the copyright industries and federal prosecutors. A more open-ended criterion for purpose-based regulations of reverse engineering may forestall such unfortunate consequences.

³⁵¹ The principal reason that European policymakers decided to permit decompilation of computer programs for purposes of achieving interoperability was to make the threat of reverse engineering credible enough so that innovators would license interface information voluntarily. See Official Commentary to the European Software Directive, reproduced in Czarnota & Hart, *supra* note 163, at 76-80.

³⁵² See, e.g., *U.S. v. Microsoft Corp.*, 243 F.3d 34 (D.C. Cir. 2001)(review of conflicting views on Microsoft's purposes in integrating Internet Explorer into the Windows operating system).

³⁵³ See 17 U.S.C. sec. 1201 (d)-(k).

³⁵⁴ See *supra* Sections V-C.

4. Regulating Reverse Engineering Tools

The DMCA anti-circumvention rules are unique among the legal regimes we studied in regulating the development and distribution of tools for reverse engineering. This strategy does not regulate the act of reverse engineering or post-reverse engineering activities so much as preparatory activities necessary to engage in reverse engineering. For reasons given in section V, we think the DMCA's anti-tools rules are overbroad, but we recognize that these rules cannot be judged by the same considerations as we used in other industrial contexts. Our general assumption about reverse engineering in other contexts has been that once the proper boundaries of intellectual property are established, the property right will be enforced. The anti-tool rules, in contrast, are directed at the problem of enforcement.

The enforcement problem arises because digital content is very cheap and easy to copy. To overcome this, the entertainment industry is increasingly using technical measures to protect their content from unauthorized access and use. Circumvention undermines this strategy. Since circumvention tools are essential to reverse engineering of these technical measures, the entertainment industry persuaded Congress to outlaw circumvention tools. We agree that there are some good economic arguments for regulating trafficking in anti-circumvention technologies. Without ready access to circumvention tools, both large- and small-scale infringements may be prevented. Anti-trafficking tool rules may avoid wasteful expenditures on an "arms race" of technical measures and countermeasures. It is, moreover, easier to detect and police a public market in circumvention technologies than to control private acts of circumvention and copying.³⁵⁵ Nevertheless, we have argued that the anti-tool rules of the DMCA are defective because they reach many activities that have little marginal value for enforcement purposes. Overbroad anti-tool rules are also harmful because they have provided copyright owners with a potent weapon for excluding competitive or complementary products from the market.³⁵⁶ They also facilitate the ability of copyright owners to leverage their market power in content into the equipment market.

5. Restricting Publication of Information Discovered By a Reverse Engineer

A fifth policy option for regulating reverse engineering is to restrict how the resulting information can be used. For the most part, it has been unnecessary for the law to address this issue because reverse engineers generally do not publish the information they learn. Instead, they typically keep this know-how secret in order to have a competitive advantage over those who have not performed similar reverse engineering activities.

However, publishing information learned through (lawful) reverse engineering is legal in the U.S.,³⁵⁷ although one commentator has opined that publishing reverse-engineered information about the internal design elements of computer software should

³⁵⁵ See, e.g., Gilbert & Katz, *supra* note 292, at 982-83.

³⁵⁶ See *supra* Section V-C.

³⁵⁷ See, e.g., *Chicago Lock Co. v. Fanberg*, 676 F.2d 400 (9th Cir. 1981).

be illegal.³⁵⁸ The European Software Directive takes a similar view, making it illegal to publish (or to license) information obtained in the course of reverse engineering computer software.³⁵⁹

The DMCA anti-circumvention rules deviate from the traditional U.S. rule in their many restrictions on disclosure of information learned even in the course of privileged acts of reverse engineering. A reverse engineer can, for example, bypass technical protections when necessary to achieve program-to-program interoperability, but he or she cannot disclose information learned in that process unless the sole purpose of the disclosure is to achieve interoperability.³⁶⁰ One judge has opined that a journalist's publication of such information would violate the DMCA, even if the reverse engineer lawfully obtained the information under the interoperability exception.³⁶¹ Similarly, the presentation of a scientific paper on flaws of a digital watermark has been said to violate the DMCA anti-circumvention rules.³⁶² Although the DMCA's exception for encryption research permits some dissemination of the results of legitimate encryption research,³⁶³ it puts encryption researchers at risk if they publish their results on the Internet or otherwise in a manner that courts might decide facilitates infringement.³⁶⁴ Leaving aside the serious question as to whether such restrictions are justifiable under the First Amendment to the Constitution,³⁶⁵ we question whether the publication of information learned in the course of reverse engineering of a technical measure should be treated as equivalent to distribution of a circumvention technology, especially in view of the collateral damage that such restrictions would likely have on the progress of the science of encryption and computer security.³⁶⁶

B. Policy Options When Innovators Try to Bypass Reverse Engineering

The very reasons that reverse engineering is socially beneficial—for example, in eroding a first comer's market power and contributing to follow-on innovation—may be why some innovators want to bypass reverse engineering altogether or render it moot. When reverse engineering is lawful, private firms may seek to thwart this activity in one of two ways: either by requiring customers to agree not to reverse engineer the product or by designing the product to make it very difficult or impossible to reverse engineer.

³⁵⁸ See, e.g., Davidson, *supra* note 151, at 1074-75.

³⁵⁹ European Software Directive, *supra* note 163, art. 6(2). The EU rule essentially puts each firm that wants to reverse engineer to the full expense of decompiling the program on its own. This preserves the lead-time of the firm whose program has been decompiled, but leads to more socially wasteful costs unless the decompilee licenses interface information to foreclose the decompilation effort.

³⁶⁰ 17 U.S.C. sec. 1201(f)(3).

³⁶¹ *Universal City Studios, Inc. v. Reimerdes*, 111 F.Supp.2d 294, 320 (S.D.N.Y. 2000). This decision is currently on appeal to the Second Circuit Court of Appeals. This aspect of the *Reimerdes* ruling is difficult to square this with the Supreme Court's decision in *Bartnicki v. Vopper*, 2001 U.S. Lexis 3815 (U.S. 2001) in which the Court held that a journalist could not be held liable for publishing illegally obtained information as long as the journalist did not participate in the illegal interception of the information.

³⁶² See *supra* note 297 and accompanying text.

³⁶³ 17 U.S.C. sec. 1201(g).

³⁶⁴ See *supra* Section V-C.

³⁶⁵ See, e.g., Bellovin Brief, *supra* note 299.

³⁶⁶ See *supra* Section V-C.

This subsection addresses the policy responses available to deal with these strategies for bypassing the public law values embodied in reverse engineering rules.

1. Avoiding the Threat of Reverse Engineering By Contract

Section IV pointed out that software licenses often prohibit reverse engineering, even when (or especially when) reverse engineering is allowed by law.³⁶⁷ Whether such contracts should be enforceable as a general matter is an unsettled question of law.

In general, we see no reason why the law should interfere with negotiated contracts in competitive markets. However, in markets for products heavily dependent on intellectual property rights, such as software, we think there is reason to worry about contractual restrictions in general, and in particular, as they relate to reverse engineering. Market power is inevitable in markets protected by intellectual property, or else the intellectual property right has no purpose. The policy levers that define the intellectual property are devices that both grant market power and limit its boundaries. If the intellectual property regime is well designed in the first place, we see no intrinsic reason why contracting should be allowed to circumvent it, especially in markets with strong network effects.³⁶⁸ Hence, it may be reasonable not to enforce contract terms that override the right to reverse engineer in such markets.³⁶⁹ The European Union has chosen to nullify contract terms that forbid decompilation when necessary to achieving interoperability.³⁷⁰

2. Avoiding the Threat of Reverse Engineering by Technical Obfuscation

Firms sometimes invest in designing their products so that it will be difficult or impossible to reverse engineer them.³⁷¹ These expenditures would be unnecessary, of course, if reverse engineering was unlawful. In the economic calculus about reverse engineering, we must count expenditures to thwart reverse engineering as socially wasteful. Efforts to thwart reverse engineering may, however, be unsuccessful, or only partially successful. Determined second-comers may manage to figure out enough through reverse engineering to make a competitive product, albeit one missing some of the innovator's "secret sauce." Sometimes, however, efforts to circumvent reverse engineering may be successful. In addition, even when firms don't intentionally design their products to make reverse engineering impossible, products may, as a practical matter, be immune from reverse engineering because of the sheer complexity of the product or because details of the product design change so rapidly that by the time

³⁶⁷ A parallel policy problem is whether to enforce contractual overrides of fair use and first sale rights of copyright law. See, e.g., *Digital Dilemma*, supra note 257, at 101-02; *McManis*, supra note 234.

³⁶⁸ See, e.g., *Lemley & McGowan*, supra note 41, at 523-27.

³⁶⁹ See supra Section IV-D. Of course, contracts that prohibit reverse engineering do not entirely render the reverse-engineering right moot. A potential reverse engineer always has the option to decline the license, and reverse engineer instead. This option will have a salutary impact on the contract terms that are offered, which creates some benefits even if the right to reverse engineer is given up.

³⁷⁰ EU Software Directive, supra note 163, art. 9(1).

³⁷¹ See supra note 43.

reverse engineer finished his work, the next version of the product would be in the marketplace.

There is a policy option for dealing with such a situation, namely, forcing the innovator to disclose certain information about its product.³⁷² For example, if the arguments in favor of open interfaces have merit and interfaces cannot be effectively discerned by reverse engineering, then it may sometimes make sense to mandate directly that they be made public. That is essentially what happened when European antitrust authorities brought suit against IBM Corporation some years ago for abuse of dominant position. IBM had been altering the interfaces to its mainframe computers frequently, which disadvantaged European makers of peripheral products. The dispute was eventually resolved by an agreement by which IBM would make advance announcements of changes to its interfaces so that peripheral manufacturers could adjust their products accordingly.³⁷³ Some have suggested a similar remedy in *United States v. Microsoft*.³⁷⁴ Microsoft has maintained its monopoly position in the operating systems market in part through control over the APIs to the Windows platform. Reverse engineering of the Windows APIs is certainly far more difficult than, say, reverse engineering interfaces to game platforms and may be impracticable. Forcing Microsoft to publish its APIs would certainly erode its market power, but this raises a host of other difficulties.³⁷⁵

VII. Conclusion

Reverse engineering is fundamentally directed to discovery and learning. Engineers learn the state of the art not just by reading printed publications, going to technical conferences, and working on projects for their firms, but also by reverse engineering others' products. Learning what has been done before often leads to new products and advances in know-how. Reverse engineering may be a slower and more expensive way for information to percolate through a technical community than patenting

³⁷² It is worth pointing out that in a variety of other circumstances, legal decision-makers have forced firms to disclose information pertaining to publicly distributed products that are not readily discernible from examination of the product when necessary to achieve some important public purpose. While such regulations have sometimes been challenged as unjustified "takings" of private property, for the most part, such challenges have not been successful. See, e.g., *Ruckelshaus v. Monsanto Co.*, 467 U.S. 986 (1984) (challenging requirements of the Federal Insecticide, Fungicide, & Rodenticide Act to submit safety test data to the Environmental Protection Agency which the EPA could consider in connection with a competitor's application for permission to sell the same chemical). The idea of forced disclosure also underlies the Burk & Cohen proposal for a key escrow system to enable prospective fair users to get access to encryption keys so that they can make fair uses of technically protected digital content. See *supra* note 329.

³⁷³ See, e.g., Band & Katoh, *supra* note 151, at 22, n. 30.

³⁷⁴ See, e.g., Piraino, *supra* note 177, at 888-89. See also R. Craig Romaine & Steven C. Salop, *Slap Their Wrists? Tie Their Hands? Slice Them Into Pieces? Remedies for Monopolization in the Microsoft Case*, 13 *Antitrust* 15, 18 (Summer 1999).

³⁷⁵ A key difficulty arises from the fact that program interfaces are not always self-evident or self-defining. See, e.g., Czarnota & Hart, *supra* note 163, at 37-38, and Band & Katoh, *supra* note 151, at 6-7 (discussing difficulties of precisely defining "interface"). Much judicial oversight might be necessary to enforce an obligation by Microsoft to disclose interface information. See Romaine & Salop, *supra* note 374, at 19.

or publication, but it is nonetheless an effective source of information.³⁷⁶ Of necessity, reverse engineering is a form of dependent creation, but this does not taint it, for in truth, all innovators, as the saying goes, “stand on the shoulders of giants,”³⁷⁷ as well as on the shoulder of other incremental innovators.³⁷⁸ Progress in science and the useful arts is advanced by dissemination of know-how, whether by publication, patenting or reverse engineering.

We think it is no coincidence that most of the proposals to restrict reverse engineering in the past two decades have arisen as to information-based products, such as semiconductors and software. The high quantum of know-how that such products bear on or near the face of the product make these products more vulnerable than traditional manufactured goods to market-destructive appropriations.³⁷⁹ This is especially true when the information is in digital form. Copying and distribution of digital products is essentially costless and almost instantaneous in the digital networked environment. The vulnerability of information products to market-destructive appropriations may justify some limitations on reverse engineering or post-reverse engineering activities, but reverse engineering is important to innovation and competition in all industrial contexts studied.

Adapting intellectual property law so that it provides adequate, but not excessive, protection to innovations is a challenging task. In considering future proposals to limit reverse engineering, we hope that policymakers will find it helpful to consider the economic effects of mechanisms that have been employed in the past. Restrictions on reverse engineering ought to be imposed only if justified in terms of the specific characteristics of the industry, a specific threat to that industry, and the economic effects of the restriction.

We worry that the recent DMCA restrictions on reverse engineering may propagate backwards and erode longstanding rules permitting reverse engineering in other legal regimes. As Professors Dreyfuss and Kwall have observed, “the distinction between, say, breaking into a factory (improper) and breaking into the product (proper) may seem artificial.”³⁸⁰ It is, however, a distinction that has been a foundational principle of intellectual property and unfair competition law, at least until enactment of the DCMA. It is, moreover, a distinction whose abandonment could have detrimental consequences for innovation and competition.

³⁷⁶ As Dreyfuss & Kwall observe, “Since there is no time limit to trade secrecy protection, reverse engineering is the principal way in which a trade secret enters the public domain.” Dreyfuss & Kwall, *supra* note 44, at 818.

³⁷⁷ See, e.g., Scotchmer, *supra* note 98.

³⁷⁸ See, e.g., Reichman, *Legal Hybrids*, *supra* note 12, at 2443-44; Manifesto, *supra* note 14, at 2329-33 (emphasizing incremental innovation).

³⁷⁹ See, e.g., Reichman, *Legal Hybrids*, *supra* note 12, 2443-44.

³⁸⁰ Dreyfuss & Kwall, *supra* note 44, at 818.