The Law and Economics of Reverse Engineering

Pamela Samuelson[†] **and Suzanne Scotchmer**[††]

CONTENTS

† Professor of Law and Information Management, University of California at Berkeley.
†† Professor of Economics and Public Policy, University of California at Berkeley.

Research support for this Article was provided by NSF Grants Nos. 98 18689 and 99 79852. We wish to thank Kirsten Roe, Eddan Katz, and Christine Duh for their excellent research assistance in connection with this Article. We are also grateful for insightful comments on earlier drafts by Rochelle Cooper Dreyfuss, Joseph Farrell, Neil Gandal, Robert J. Glushko, Wendy J. Gordon, Mark A. Lemley, Ejan MacKaay, Stephen Maurer, David McGowan, Michael Moradzadeh, Maureen O'Rourke, Eva Oglieska, Jerry Reichman, Dan Rubinfeld, Hal Varian, Fred von Lohmann, and participants in the Boston University Intellectual Property Workshop Series, the University of Washington Law School's Innovation Workshop, and the Yale Legal Theory Workshop.

I. INTRODUCTION

Reverse engineering has a long history as an accepted practice. What it means, broadly speaking, is the process of extracting know-how or knowledge from a human-made artifact.[1] Lawyers and economists have endorsed reverse engineering as an appropriate way to obtain such information, even if the intention is to make a product that will draw customers away from the maker of the reverse-engineered product.[2] Given this acceptance, it may be surprising that reverse engineering has been under siege in the past few decades.

While some encroachments on the right to reverse-engineer have been explicit in legal rulemaking, others seem implicit in new legal rules that are altogether silent on reverse engineering, including the Agreement on Trade-Related Aspects of Intellectual Property Rights (TRIPS)[3] and the Economic Espionage Act of 1996 (EEA).[4] TRIPS is an international treaty that, among other things, obligates member states of the World Trade Organization to protect trade secrets, yet it neither requires nor sanctions a reverse engineering privilege.[5] The EEA created the first federal cause of action for

1. This is a broader definition than has previously been used by courts and commentators, but it captures how the term is used in this Article. "Human-made artifacts" are objects that embody knowledge or know-how previously discovered by other people. Hence, the engineering required to uncover the knowledge is "reverse" engineering. As we shall see, extraction of this knowledge can be costly or cheap, time-consuming or fast, depending on the artifact, and these notions govern the consequences of allowing it to be extracted. The standard legal definition, from *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470 (1974), is "starting with the known product and working backward to divine the process which aided in its development or manufacture." *Id.* at 476. Professor Reichman conceives of this knowledge as applied scientific or industrial know-how. J.H. Reichman, *Computer Programs as Applied Scientific Know-How: Implications of Copyright Protection for Commercialized University Research*, 42 VAND. L. REV. 639, 656-62 (1989). Treatise author James Pooley has emphasized that the "fundamental purpose of reverse engineering is discovery, albeit of a path already taken." JAMES POOLEY, TRADE SECRET LAW § 5.02, at 5-19 (1997). All of these formulations fit within our simple notion of extracting knowledge from a human artifact.

2. *E.g.*, POOLEY, *supra* note 1, § 5.02[1], at 5-16; David Friedman et al., *Some Economics of Trade Secret Law*, 5 J. ECON. PERSP., Winter 1991, at 61, 71; sources cited *infra* notes 26-36, 162.

3. Final Act Embodying the Results of the Uruguay Round of Multilateral Trade Negotiations, Apr. 15, 1994, LEGAL INSTRUMENTS—RESULTS OF THE URUGUAY ROUND vol. 1, 33 I.L.M. 1125 (1994); Agreement on Trade-Related Aspects of Intellectual Property Rights, Apr. 15, 1994, Marrakesh Agreement Establishing the World Trade Organization, Annex 1C, LEGAL INSTRUMENTS—RESULTS OF THE URUGUAY ROUND vol. 31, 33 I.L.M. 81 (1994) [hereinafter TRIPS Agreement]. The trade secrecy provision of the TRIPS Agreement is Article 39, 33 I.L.M. at 98. For congressional approval of the TRIPS and WTO Agreements, see Uruguay Round Agreements Act, Pub. L. No. 103-465, §§ 101-103, 108 Stat. 4809, 4814-19 (1994) (codified in scattered sections of 15, 17, 19, and 35 U.S.C.).

4. Economic Espionage Act of 1996, Pub. L. No. 104-294, 110 Stat. 3488 (codified at 18 U.S.C. §§ 1831-1839 (Supp. V 1999)).

5. *See* TRIPS Agreement, *supra* note 3, art. 39, 33 I.L.M. at 98. *But see* Charles R. McManis, *Taking TRIPS on the Information Superhighway: International Intellectual Property Protection*

trade secrecy misappropriation. Its lack of a reverse engineering defense has troubled some commentators because rights granted under the EEA arguably implicate certain reverse engineering activities previously thought to be lawful.[6]

Among the explicit legal challenges to reverse engineering are these: In the 1970s and 1980s some states forbade the use of a direct molding process to reverse-engineer boat hulls.[7] In the late 1970s and early 1980s, the semiconductor industry sought and obtained legislation to protect chip layouts from reverse engineering to make clone chips.[8] In the mid-1980s and early 1990s, a controversy broke out about whether decompilation, a common form of reverse engineering of computer programs, was legal as a matter of copyright law.[9] Even after U.S. courts ruled that decompilation was acceptable for purposes such as achieving interoperability,[10] a related controversy broke out over the enforceability of licenses forbidding reverse engineering of software and other digital information.[11] More recently, questions have arisen about whether the decompilation of computer programs infringes upon patent rights in software components.[12] In 1998, Congress outlawed the reverse engineering of technical protections for digital versions of copyrighted works and prohibited both the creation and distribution of tools for such reverse engineering (except in very limited circumstances) as well as the disclosure of information obtained in the course of lawful reverse engineering.[13]

---

and Emerging Computer Technology, 41 VILL. L. REV. 207 (1996) (arguing that reverse engineering of software is acceptable within the TRIPS framework).

6. The "troubling" absence of a specific reverse engineering privilege in the EEA was noted in James H.A. Pooley et al., *Understanding the Economic Espionage Act of 1996*, 5 TEX. INTELL. PROP. L.J. 177, 195 (1997). *See also* Rochelle Cooper 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) (discussing how observers may infer a prohibition against reverse engineering from the EEA). 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 that forbid duplicating trade secrets. Pooley et al., *supra*, at 195-96; *see also* Craig L. Uhrich, *The Economic Espionage Act—Reverse Engineering and the Intellectual Property Public Policy*, 7 MICH. TELECOMM. & TECH. L. REV. 147, 186-87 (2001) (recommending amendments to the EEA to privilege legitimate reverse engineering activities).

7. 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.

8. *See infra* Part III.

9. Decompilation transforms machine-readable electronic impulses of object code into human-readable form. *See infra* Section IV.A.

10. *E.g.*, Sega Enters. v. Accolade, Inc., 977 F.2d 1510 (9th Cir. 1992); *see also infra* Section IV.A (discussing this case).

11. *See infra* Section IV.C.

12. *See* Julie E. Cohen & Mark A. Lemley, *Patent Scope and Innovation in the Software Industry*, 89 CAL. L. REV. 1, 6 (2001); *see also infra* note 175 (discussing this article).

13. Digital Millennium Copyright Act, Pub. L. No. 105-304, 112 Stat. 2860 (1998) (codified in scattered sections of 5, 17, 28, and 35 U.S.C.). There is a limited exception to enable bypassing technical controls and making tools to enable this when necessary to achieve interoperability among programs. 17 U.S.C. § 1201(f) (Supp. V 1999). This law is discussed *infra* Part V.

Our objectives in this Article are, first, to review legal developments regarding the right to reverse-engineer, and second, to understand their economic consequences.

We start in Part II with a discussion of the well-established legal right to reverse-engineer manufactured goods. In our view, the legal rule favoring reverse engineering in the traditional manufacturing economy has been economically sound because reverse engineering is generally costly, time-consuming, or both. Either costliness or delay can protect the first comer enough to recoup his initial research and development (R&D) expenditures.[14] If reverse engineering (and importantly, the consequent reimplementation) of manufactured goods becomes too cheap or easy, as with plug-molding of boat hulls, it may be economically sound to restrict this activity to some degree.

In Parts III, IV, and V, we 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. We think it is no coincidence that proposals to restrict reverse engineering have been so common in information-based industries. Products of the information economy differ from traditional manufactured products in the cost and time imposed on a reverse engineer. With manufactured goods, much of the know-how required to make the goods remains within the factory when the products go to market, so that reverse engineering can capture only some of the know-how required to make the product. The information-rich products of the digital economy, in contrast, bear a higher quantum of applied know-how within the product distributed in the market.[15]

For so-called digital content (movies, sound recordings, and the like), the relevant knowledge is entirely on the surface of the product, at least in

14. 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 secrecy law in providing lead time to innovators. Costliness itself will also suffice even without lead time, as discussed *infra* Part II.

15. We build here on prior work distinguishing the accessibility of know-how in information-based industries as compared with traditional manufacturing industries. *E.g.*, Reichman, *supra* note 1, at 660 ("[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."); Reichman, *supra* note 14, at 2511-18 (giving a historical perspective of the challenges for legal systems of providing protection for applied know-how); Pamela Samuelson et al., *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 because of the applied industrial know-how borne on or near the surface of software products).

the absence of technical protections such as encryption. Technical protections create costs for reverse engineers. When computer programs are distributed in object code form, a difficult analytical process is required to ascertain information embedded in the program, but it is there for the taking if a reverse engineer is willing to spend the time to study it.[16] For computer chips, the relevant knowledge is circuit design, which is not only embodied within the chip, but also readily accessible using technologies discussed below.[17] The challenge is to design legal rules that protect information-rich products against market-destructive cloning while providing enough breathing room for reverse engineering to enable new entrants to compete and innovate in a competitively healthy way.

Part III focuses on the semiconductor chip industry. When the competitive reverse engineering and copying of semiconductor chip designs became too easy and too rapid to enable innovators to recoup their R&D costs, Congress responded by enacting the Semiconductor Chip Protection Act of 1984 (SCPA) to protect chip makers from market-destructive cloning while affirming a limited right to reverse-engineer chips.[18] The SCPA allows reverse engineers to copy circuit design to study it as well as to reuse information learned thereby in a new chip, but it imposes a forward engineering requirement that inevitably increases a second comer's development time and increases its costs.[19] In the context of the chip industry, we think this restriction on reverse engineering is economically sound.

Part IV focuses on the software industry. Reverse engineering is undertaken in the software industry for reasons different from those in other industrial contexts. The most economically significant reason to reverse-engineer software, as reflected in the case law, is to learn information necessary to make a compatible program. The legal controversy over whether copies made of a program during the decompilation process infringe copyrights has been resolved in favor of reverse engineers. But as Part IV explains, the economics of interoperability are more complex than legal commentators have acknowledged. On balance, however, we think that a legal rule in favor of reverse-engineering computer programs for purposes of interoperability is economically sound.

Part V discusses the emerging market for technically protected digital content. Because technical protection measures may be defeated by countermeasures, copyright industry groups persuaded Congress to enact

16. *See infra* notes 182-184 and accompanying text.

17. *See infra* note 140 and accompanying text.

18. Semiconductor Chip Protection Act of 1984, Pub. L. No. 98-620, 98 Stat. 3347 (codified at 17 U.S.C. §§ 901-914 (1994)).

19. *See* 17 U.S.C. § 906(a); *see also infra* notes 94-96 and accompanying text (discussing the SCPA).

the Digital Millennium Copyright Act (DMCA), which creates new legal rules reinforcing technical measures used by copyright owners to protect their works.[20] It protects them against most acts of circumvention, against the manufacture and distribution of circumvention technologies, and against dissemination of information resulting from privileged acts of circumvention.[21] In our view, these new rules overly restrict reverse engineering, although the core idea of regulating trafficking in circumvention technologies may be justifiable.

Part VI steps back from particular industrial contexts and considers reverse engineering as one of the important policy levers of intellectual property law, along with rules governing the term and scope of protection. The most obvious settings for the reverse engineering policy lever are "on" (reverse engineering is permissible) and "off" (reverse engineering is impermissible). However, our study reveals five additional strategies for regulating reverse engineering in the four industrial contexts studied: regulating a particular means of reverse engineering, adopting a "breadth" requirement for subsequent products, permitting reverse engineering for some purposes but not others, regulating tools used for reverse engineering, and restricting the dissemination of information discerned from reverse engineering. In this discussion, we distinguish between regulations affecting the act of reverse engineering and those affecting what the reverse engineer can do with the resulting information. Some restrictions on reverse engineering and on post-reverse-engineering activities may be economically sound, although we caution against overuse of restrictions on reverse engineering because such restrictions implicate competition and innovation in important ways. Part VI also considers policy responses when innovators seek to thwart reverse engineering rights by contract or by technical obfuscation.

Intellectual property law in the United States has an important economic purpose of creating incentives to innovate as a means of advancing consumer welfare.[22] The design of intellectual property rules, including those affecting reverse engineering, should be tailored to achieve these utilitarian goals and should extend no further than necessary to protect incentives to innovate. Intellectual property rights, if made too strong, may impede innovation and conflict with other economic and policy objectives.

---

20. Digital Millennium Copyright Act, Pub. L. No. 105-304, 112 Stat. 2860 (1998) (codified in scattered sections of 5, 17, 28, and 35 U.S.C.).

21. The anticircumvention rules of the DMCA are now codified at 17 U.S.C. § 1201 (Supp. V 1999).

22. *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.").

II. REVERSE ENGINEERING IN TRADITIONAL MANUFACTURING INDUSTRIES

Reverse engineering is generally a lawful way to acquire know-how about manufactured products. Reverse engineering may be undertaken for many purposes.[23] We concentrate in this Part on reverse engineering undertaken for the purpose of making a competing product because this is the most common and most economically significant reason to reverse-engineer in this industrial context.[24] We argue that legal rules favoring the reverse engineering of manufactured products have been economically sound because an innovator is nevertheless protected in two ways: by the costliness of reverse engineering and by lead time due to difficulties of reverse engineering.[25] If technological advances transform reverse engineering so that it becomes a very cheap and rapid way to make a competing product, innovators may not be able to recoup their R&D expenses, and hence some regulation may be justified. An example discussed below is the plug-molding of boat hulls.

A. *A Legal Perspective on Reverse Engineering*

Reverse engineering has always been a lawful way to acquire a trade secret, as long as "acquisition of the known product . . . [is] by a fair and honest means, such as purchase of the item on the open market."[26] As the Restatement of Unfair Competition points out, "The 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,"[27] as when the use or disclosure breaches an implicit or explicit agreement between the parties or when

23. Pooley identifies six reasons for engaging in reverse engineering: learning, changing or repairing a product, providing a related service, developing a compatible product, creating a clone of the product, and improving the product. *See* POOLEY, *supra* note 1, § 5.02[2], at 5-18 to -19.

24. 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 is discussed *infra* Section IV.B.

25. This Part focuses on incentives to invest in innovation in manufacturing industries when patent rights are not available (e.g., because the innovation is too modest an advance to meet the nonobviousness standard) or when firms choose trade secrecy over patents (e.g., because they do not want to disclose the innovation to the public as would be necessary to get a patent). Patents play an important role in creating incentives to invest in innovation, but innovators must recoup R&D expenses regardless of whether patent rights are available.

26. UNIF. TRADE SECRETS ACT § 1 cmt. 2, 14 U.L.A. 437, 438 (1990).

27. RESTATEMENT (THIRD) OF UNFAIR COMPETITION § 43 cmt. a (1995); *see, e.g.*, Tabor v. Hoffman, 23 N.E. 12 (N.Y. 1889) (finding misappropriation of trade secrets where the defendant exceeded authorized access by measuring and copying the plaintiff's patterns in order to make competing pumps).

improper means, such as trespass or deceit, are used to obtain the secret.[28] 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.[29]

The legal right to reverse-engineer a trade secret is so well-established that courts and commentators have rarely 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 "may lead to significant advances in the field."[30] Moreover, "the competitive reality of reverse engineering may act as a spur to the inventor" to develop patentable ideas.[31] 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.[32]

Further justification for the law's recognition of a right to reverse-engineer likely derives from the fact that the product is purchased in the open market, which confers on its owner personal property rights, including the right to take the purchased product apart, measure it, subject it to testing, and the like. The time, money, and energy that reverse engineers invest in analyzing products may also be a way of "earning" rights to the information they learn thereby. Still another justification stems from treating the sale of a product in the open market as a kind of publication of innovations it embodies. This publication dedicates these innovations to the public domain unless the creator has obtained patent protection for them.[33]

Courts have also treated reverse engineering as an important factor in maintaining balance in intellectual property law. Federal patent law allows innovators up to twenty years of exclusive rights to make, use, and sell an invention,[34] but only in exchange for disclosure of significant details about

28. RESTATEMENT OF TORTS § 757 (1939); UNIF. TRADE SECRETS ACT § 1, 14 U.L.A. at 437-38.

29. *See, e.g.*, Heald, *supra* note 7, at 975.

30. 489 U.S. 141, 160 (1989); *see also infra* Section II.C (discussing this case); *cf.* MATTHEW JOSEPHSON, EDISON 91 (1959) ("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*.").

31. *Bonito Boats*, 489 U.S. at 160.

32. *See* Heald, *supra* note 7, 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.

33. *See* Tabor v. Hoffman, 23 N.E. 12 (N.Y. 1889) (discussing the "publication" theory).

34. 35 U.S.C. § 154(a)(2) (1994).

their invention to the public.[35] 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."[36] Reverse engineering, then, is an important part of the balance implicit in trade secret law.

No reverse engineering right, as such, exists in patent law.[37] 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.[38] 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.[39] In addition, a person who tries to make a patented invention to satisfy

35. *Id.* § 112 (setting forth disclosure requirements). The Supreme Court in *Kewanee Oil Co. v. Bicron Corp.* spoke of patent law's disclosure requirement as "the quid pro quo of the right to exclude." 416 U.S. 470, 484 (1974); *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).

36. Chi. Lock Co. v. Fanberg, 676 F.2d 400, 405 (9th Cir. 1981). *Fanberg* relied on the Supreme Court's decision in *Kewanee* 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 than and different from patent law. Reverse engineering was one of the features of trade secrecy law that made it so. *See Kewanee*, 416 U.S. at 489-90; *see also* Rockwell Graphic Sys., Inc. v. Dev. Indus., Inc., 925 F.2d 174, 178-80 (7th Cir. 1991) (discussing reverse engineering as a limitation on trade secret protection); 1 MELVIN F. JAGER, JAGER ON TRADE SECRETS § 5.04[3][a][i], at 5-39 (2001) ("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."); POOLEY, *supra* note 1, § 5.02[1], at 5-16 (explaining that because reverse engineering makes trade secret law weaker than patent law, trade secret law is not preempted by patent law).

37. *See* Cohen & Lemley, *supra* note 12, 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. §§ 2321-2583 (1994), 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 chapter." *Id.* § 2544.

38. 35 U.S.C. § 112.

39. *See* Cohen & Lemley, *supra* note 12, 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 174-175 for a discussion of the 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 enforced contractual restrictions on reverse engineering. *See* K&G Oil Tool & Serv. Co. v. G&G Fishing Tool Serv., 314 S.W.2d 782, 785-86 (Tex. 1958) (enforcing a negotiated agreement not to

scientific curiosity may assert an experimental use defense to patent infringement.[40]

Until quite recently, copyright law neither had nor had need for a reverse engineering privilege. The artistic and literary works this law traditionally protected did not need to be reverse-engineered to be understood.[41] 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 can simply read or analyze the work. Moreover, at least until the admission of computer programs to its domain, copyright law did not protect industrial products of the sort that firms typically reverse-engineer.[42]

B. *An Economic Perspective on Reverse Engineering*

The economic effects of reverse engineering depend on a number of factors, including the purpose for which it is undertaken, the industrial context within which it occurs, how much it costs, how long it takes, whether licensing is a viable alternative, and how the reverse engineer uses information learned in the reverse engineering process.[43] In this Section, we

---

disassemble K&G's magnetic fishing tool against a 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.2d (BNA) 1797 (S.D. Iowa 1999) (enforcing a "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.C.

40. 1 ROGER M. MILGRIM, MILGRIM ON TRADE SECRETS § 1.05[5], at 1-250 (2000). In U.S. patent law, the experimental use defense is quite narrow, not encompassing, for example, scientific or research uses that may lead to development of a patentable invention or a commercial product. *See* 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* 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).

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

42. Pictorial, sculptural, or graphic works can be protected by U.S. copyright law unless they have usefulness beyond conveying information or displaying an appearance. *See* 17 U.S.C. § 101 (1994) (defining "pictorial, sculptural and graphic works" and "useful article"). Many industrial products (e.g., chairs, automobiles, and toasters) have an aesthetic appearance, yet they are not copyrightable in the United States because their aesthetic design is not separable from their utilitarian functions. *See, e.g.*, Brandir Int'l, Inc. v. Cascade Pac. Lumber Co., 834 F.2d 1142 (2d Cir. 1987) (holding that the aesthetic design for a bicycle rack was uncopyrightable because of the inseparability of functional considerations in the final design).

43. Reverse engineering does not itself render the trade secret valueless because reverse engineers do not generally publish their discoveries, instead maintaining the discovered information as their own trade secret. *See* POOLEY, *supra* note 1, § 5.02[2], 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 innovation will no longer be secret.

concentrate on the economics of reverse engineering undertaken for the purpose of developing a competing product.[44]

We argue that a legal right to reverse-engineer does not typically threaten an innovative manufacturer because the manufacturer generally has two forms of protection against competitors who reverse-engineer: lead time before reverse engineers can enter[45] and 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 innovator licenses others as a strategy for preventing unlicensed entry. Provided that 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 other potential entrants.

Our economic assessment of reverse engineering recognizes that this activity is only one step in what is typically a four-stage development process. The first stage of a second comer's development process is an awareness stage.[46] 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 happens 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 goodwill with its customer base.[47]

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.[48] The reverse engineering stage

44. Some economic effects arising from reverse engineering for purposes of developing complementary products are explored *infra* Section IV.B.

45. Empirical studies of manufacturing firms over a long period demonstrate that such firms typically rely more on lead time than on patents as the principal source of protection for their intellectual assets. *See, e.g.*, WESLEY M. COHEN ET AL., PROTECTING THEIR INTELLECTUAL ASSETS: APPROPRIABILITY CONDITIONS AND WHY U.S. MANUFACTURING FIRMS PATENT (OR NOT) (Nat'l Bureau of Econ. Research, Working Paper No. 7552, 2000); *see also* Reichman, *supra* note 14, at 2439-41 (explaining the importance of lead time in trade secrecy law).

46. 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* GEOFFREY A. MOORE, CROSSING THE CHASM: MARKETING AND SELLING HIGH-TECH 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.

47. 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* Mark A. Lemley & David McGowan, *Legal Implications of Network Economic Effects*, 86 CAL. L. REV. 479 (1998).

48. 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.

may be costly, time-consuming, and difficult,[49] although this varies considerably, depending mainly on how readily the innovator's product will yield the know-how required to make it when confronted by a determined and skilled reverse engineer.[50] 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 do not work,[51] and in part because advances in technology typically reduce the costs of rediscovery over time.

Third is the implementation stage.[52] After reverse-engineering the innovator's product, a second comer must take the know-how obtained

---

49. 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* POOLEY, *supra* note 1, § 4.04[4], at 4-42; *see also* RESTATEMENT (THIRD) OF UNFAIR COMPETITION, *supra* note 27, § 39 cmt. f (emphasizing that continued protection as a trade secret depends on the difficulty and the expense of reverse engineering). Firms can sometimes make reverse engineering more difficult, and this may be an economically sensible thing to do if the secret is valuable. Pooley notes:

> It may also be possible to build products that are 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.

POOLEY, *supra* note 1, § 5.02[5], at 5-25. Friedman, Landes, and Posner regard the expenditures required to make a product more difficult to reverse-engineer as costs of not prohibiting reverse engineering. Friedman et al., *supra* note 2, 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, 711-13 (1980); *see also* Steven N.S. Cheung, *Property Rights in Trade Secrets*, 20 ECON. INQUIRY 40, 47 (1982) (discussing the economics of trade secrecy law and various means by which trade secret rents may be dissipated).

50. *See, e.g.*, POOLEY, *supra* note 1, § 4.04[4], 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 COOPER DREYFUSS & ROBERTA ROSENTHAL KWALL, INTELLECTUAL PROPERTY 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 2, at 70; Kitch, *supra* note 49, at 690. However, it may sometimes be possible to persuade courts that independent invention of the same trade secret was unlikely. *See, e.g.*, Pioneer Hi-Bred Int'l v. Holden Found. Seeds, Inc., 35 F.3d 1226, 1237 (8th Cir. 1994).

51. Friedman et al., *supra* note 2, at 63; *see also* JARED DIAMOND, GUNS, GERMS, AND STEEL 224-25, 244-45, 256 (1999) (giving examples of technologies whose reinvention occurred rapidly once it became known that the technology was possible).

52. 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

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 reiterating 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 innovation that will be investigated in the implementation stage.[53] 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.[54]

In the chart and discussion below, we use four criteria to assess the social welfare effects of the law's recognition of a right to reverse-engineer. The criteria are the effects on the following: incentives to innovate, incentives to engage in follow-on innovation, prices, and socially wasteful expenditures of resources. At first glance, these considerations seem to cut in opposite directions in the manufacturing industry context. On the negative side, the right to reverse-engineer seems to decrease incentives for first comers to introduce new products and to encourage wasteful expenditures on reverse engineering.[55] On the positive side, a right to reverse-engineer can increase competition in the marketplace, lead to lower prices, and spur follow-on innovations by second comers.

However, the argument against reverse engineering based on wasted costs is misleading because the cost of reverse engineering can be avoided

---

take a license may decline as its 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.

53. Richard C. Levin et al., *Appropriating the Returns from Industrial Research and Development*, 1987 BROOKINGS PAPERS ON ECON. ACTIVITY 783, 806 (noting the improvements that are likely to result from reverse engineering).

54. 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.

55. 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 49.

by licensing.[56] Licensing should be in the interest of both the innovator and potential reverse engineers as they can share the saved costs.

The key question, however, is how the threat of reverse engineering affects incentives to innovate. If reverse engineering actually occurs, it will erode market power and reduce the innovator's profit to an extent determined by the costliness and time required for reverse engineering. With licensing, the threat of reverse engineering will reduce the innovator's profit to a similar extent. In order to avoid reverse engineering by unlicensed entrants, the licensor must make sure that reverse engineering by unlicensed entrants is unprofitable. He can do this by allowing some measure of competition from licensees (e.g., by licensing with low royalties).[57] How much competition he authorizes will depend on the costs that unlicensed entrants would have to bear in reverse engineering and how long it would take them.[58] The profit earned by the innovator will depend on the relative costs of the innovator and potential reverse engineers, and on the time required for reverse engineering, but not very much on whether reverse engineering is avoidable by licensing.[59]

---

56. *See* Reichman, *supra* note 14, at 2530-34 (discussing licensing as an alternative to reverse engineering); *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).

57. This argument follows an argument in Stephen M. Maurer & Suzanne Scotchmer, *The Independent-Invention Defense in Intellectual Property*, 69 ECONOMICA (forthcoming 2002). That article considers the consequences of allowing entry by independent inventors in markets with patented products. The authors argue that the threat to a rightsholder's market depends on the cost of entry by rivals, in particular the cost of independent invention or inventing around a patent. Reverse engineering is just another costly way to enter the market. Reverse engineering differs from independent invention or inventing around a patent in that the product is typically not patented, and reverse engineering may be less costly than inventing around. Nevertheless, the effect of entry depends only on cost, and the same argument applies in all three contexts. The argument differs from previous ones, *see, e.g.*, Adelman, *supra* note 55, in that unlicensed entry is assumed not to occur. Instead, the threat of entry affects the terms of license, which will be used by the rightsholder for two purposes: to collect profit from authorized entrants, and to control the price of the product. The price will be just low enough to deter further (unauthorized) entry, but not lower.

58. *See supra* notes 46-50 and accompanying text. Dreyfuss and Kwall put the point succinctly:

> 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 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.

DREYFUSS & KWALL, *supra* note 50, at 818.

59. *See* 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 Restitutionary 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, for

Table 1 illustrates the social welfare effects of two possible reverse engineering rules in the context of traditional manufacturing industries: one allowing it and one disallowing it. As to each criterion, the effects of permitting reverse engineering are compared with the effects of forbidding it.

TABLE 1. SOCIAL CALCULUS OF REVERSE ENGINEERING IN THE MANUFACTURING SECTOR

Social Welfare Criterion	Reverse Engineering Legal	Reverse Engineering Illegal
Incentives to innovate	Worse (but generally adequate)	Better (but may be excessive)
Incentives for follow-on innovation	Better	Worse
Prices	Lower	Higher
Wasted costs	Worse (but avoidable by licensing)	Better

On balance, we conclude that a legal rule favoring reverse engineering of traditional manufactured products is economically sound. A prohibition on reverse engineering would, in effect, give firms perpetual exclusive rights in unpatented innovations.[60] Given that the costs and time required for reverse engineering already protect most innovators, a ban on reverse engineering is unnecessary. 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.

example, use of plug molds. *See* Douglas Gary Lichtman, *The Economics of Innovation: Protecting Unpatentable Goods*, 81 MINN. L. REV. 693, 721-23 (1997). Maurer & Scotchmer, *supra* note 57, argues from the other direction: Incentives to innovate will survive a rival's independent innovation whenever its costs are not too high relative to the innovator's development cost.

One reason that the cost of reverse engineering can be very cheap relative to the innovator's cost is that the reverse engineer avoids "dry holes." This is particularly important in some industries. By some counts, only one in five attempts to develop a drug succeeds. The reverse engineer can work on those known to be viable and avoid the others. Fortunately, drugs are generally protected by patents and are hence immune to market-destructive reverse engineering and reimplementation. Where that has not been true, as in India prior to the TRIPS Agreement, drugs were very cheap due to the ease of reverse engineering their chemical structure. *See* JEAN O. LANJOUW, THE INTRODUCTION OF PHARMACEUTICAL PRODUCT PATENTS IN INDIA: "HEARTLESS EXPLOITATION OF THE POOR AND SUFFERING"? 9-10 (Nat'l Bureau of Econ. Research, Working Paper No. 6366, 1998).

60. Friedman et al., *supra* note 2, at 70-71.

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

In the late 1970s through the 1980s, twelve states adopted laws to prohibit plug-molding of manufactured products.[61] These laws typically forbade use of a manufactured item, such as a boat hull, as a "plug" for a direct molding process that yielded 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.[62] California passed a more general anti-plug-mold law.

In *Interpart Corp. v. Imos Italia, Vitaloni, S.p.A.*,[63] a firm charged with violating California's anti-plug-mold law defended against the claim in part by challenging the consistency of this California statute 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 in a direct molding process).[64] California perceived no conflict with federal patent law because its state law did not confer a right to exclude others from making, using, or selling the product.[65] Anyone could reverse-engineer and copy a manufactured product by conventional means; they just couldn't do so by plug-molding.[66]

Four years later the U.S. Supreme Court overruled *Interpart* in *Bonito Boats, Inc. v. Thunder Craft Boats, Inc.*[67] One reason the Court gave for striking down Florida's anti-plug-mold law was that it "prohibit[ed] the entire public from engaging in a form of reverse engineering of a product in the public domain."[68] The Court said that it was "difficult to conceive of a

61. Heald, *supra* note 7, at 962. In some countries, parasitic copying such as that conducted by a plug-mold process is illegal as a matter of unfair competition law. *See* Reichman, *supra* note 14, at 2472-74.

62. Bonito Boats, Inc. v. Thunder Craft Boats, Inc., 489 U.S. 141, 158 (1989); *see also* Lichtman, *supra* note 59, at 719-20. The direct molding process was itself a relatively new technological innovation that had been patented in 1968. *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 provided.'" *Id.* at 164 (quoting from the patent).

63. 777 F.2d 678 (Fed. Cir. 1985).

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

65. *See Interpart*, 777 F.2d at 684-85.

66. *Id.* at 685.

67. 489 U.S. 141.

68. *Id.* at 160. *Bonito Boats* seems to elevate the principle of reverse engineering to a constitutionally protected interest. *See* Chi. Lock Co. v. Fanberg, 676 F.2d 400, 405 (9th Cir. 1982) (opining that for a 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, *supra* note 14, at

more effective method of creating substantial property rights in an intellectual creation than to eliminate the most efficient method for its exploitation."[69] Drawing upon earlier preemption rulings, the Court said they protected "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."[70] It went on to say that "[w]here an item in general circulation is unprotected by patent, '[r]eproduction of a functional attribute is legitimate competitive activity.'"[71]

The economic consequences of plug-molding deserved more serious consideration.[72] The plug-mold process dramatically reduces the costs of, and time required to engage in, reverse engineering and reimplementation of an innovation. If plug-molding undermines incentives to invest in innovative boat hulls or other manufactured goods, a ban on the use of the plug-mold process might be economically sound, at least for some period of time.[73] The germ of an argument that plug-molding might have market-

2473 (interpreting *Bonito Boats* as "endow[ing] the competitor's right to reverse engineer with constitutional underpinnings").

69. *Bonito Boats*, 489 U.S. at 164.

70. *Id.* The cases upon which the Court principally drew were the companion cases *Sears, Roebuck & Co. v. Stiffel Co.*, 376 U.S. 225 (1964), and *Compco Corp. 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. *E.g.*, Sunbeam Prods., 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 trade dress claimants to show that the claimed configuration or design is nonfunctional if it was claimed in an expired patent. *See* TrafFix Devices, Inc. v. Mktg. Displays, Inc., 532 U.S. 23 (2001).

71. *Bonito Boats*, 489 U.S. at 164 (quoting Inwood Labs., Inc. v. Ives Labs., Inc., 456 U.S. 844, 863 (1982) (White, J., concurring)) (second alteration in original). It should be noted that in 1998 Congress enacted a new form of intellectual property protection for vessel hulls. Vessel Hull Design Protection Act, Pub. L. No. 105-304, tit. V, 112 Stat. 2905 (1998) (codified at 17 U.S.C. §§ 1301-1332 (Supp. V 1999)). Now they can be neither plug-molded nor copied by any other method.

72. 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 Article I, Section 8, Clause 8 in the Constitution. *See* THE FEDERALIST NO. 43, at 239-40 (James Madison) (Clinton Rossiter ed., 1961). The nonuniformity problem was present in the *Bonito Boats* case because Thunder Craft Boats was a Tennessee-based company and Tennessee had no anti-plug-mold statute. *See Bonito Boats*, 489 U.S. at 145.

73. It should not be enough for boat designers to testify that they need 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 and Michael Remington articulated a multipart test to determine when legislation of this sort would be warranted. 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).

destructive effects can be found in *Bonito Boats*. The Supreme Court 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,[74] 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.'"[75] 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."[76] Professor Heald has questioned this assertion, pointing out that the Florida law "primarily discriminates against those interested in reproduction rather than innovation"[77] and implying that plug-molding might well "result in less innovation."[78] Heald's is the more economically sound view of the effects of plug-molding on follow-on innovation.[79]

Of course, this does not mean that the laws enacted in Florida or California were adopted 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.[80] Moreover, it did not

74. *Bonito Boats*, 489 U.S. at 144.

75. *Id.* at 164 (quoting from the patent).

76. *Id.* at 160.

77. Heald, *supra* note 7, at 985; *see also* Reichman, *supra* note 14, at 2473 (arguing that plug-molders merely duplicate the originator's product).

78. *See* Heald, *supra* note 7, at 985. The Court insisted that enactment of laws to give incentives to invest in innovation is reserved to the federal government, not to states. *See 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 7, at 969 (citation omitted). 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* John S. Wiley, Jr., Bonito Boats*: Uninformed but Mandatory Innovation Policy*, 1989 SUP. CT. REV. 283, 290-94 (discussing state intellectual property laws threatened by the preemption analysis in *Bonito Boats*).

79. 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 Section II.B has shown, reverse engineering and competitive copying of a product are different activities, even if courts, as in *Bonito Boats*, sometimes conflate them. *Bonito Boats*, 489 U.S. at 160 (stating that Florida law "prohibits the entire public from engaging in a form of reverse engineering of a product in the public domain"); *see also* TrafFix Devices, Inc. v. Mktg. 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 with the costs of initial development, or if 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.

80. Retroactive application of the law cannot encourage 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.

require any showing of originality, novelty, or improvement as a criterion for the grant of protection.[81] Nor was there any durational limit to the protection.[82] It is difficult to believe that perpetual rights are necessary to enable boat-hull designers to recoup their R&D expenses.[83] 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 innovators to recoup their investments, but not more than that.[84] In 1998, Congress enacted a sui generis form of intellectual property protection to protect boat hulls from unauthorized copying, not just from plug-molding.[85]

From an economic perspective, anti-plug-mold laws illustrate that even in the context of traditional manufacturing industries, a form of reverse engineering and reimplementation that produces cheap, rapid, identical copies has the potential to have market-destructive consequences. "[Q]uick imitation robs innovation of value."[86] Insofar as market-destructive effects can be demonstrated, it may be economically sound for the law to restrict a market-destructive means of reverse engineering and reimplementation for a period of time sufficient to enable the innovator to recoup its R&D expenses. Plug-molding is only one example of technological advances that have changed the economic calculus of reverse engineering rules, as subsequent Parts show.

---

81. *See id.* Heald was critical of the Florida plug-mold law for the lack of a creativity requirement. Heald, *supra* note 7, at 987.

82. *See Bonito Boats*, 489 U.S. at 144-45.

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

84. The new form of intellectual property right Congress enacted in 1998 to protect boat hulls does have an originality requirement and a durational limitation. 17 U.S.C. § 1302 (Supp. V 1999) (originality requirement); *id.* § 1305 (durational limitation).

85. Vessel Hull Design Protection Act, Pub. L. No. 105-304, tit. V, 112 Stat. 2905 (1998) (codified at 17 U.S.C. §§ 1301-1332 (Supp. V 1999)). In protecting the configuration of boat hulls, the Act most closely resembles utility model laws adopted in some countries. Reichman, *supra* note 14, at 2455-59 (discussing utility model laws). For the moment, the Act covers only vessel hulls, but some commentators suggest that 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 twentieth century because of concerns that it would unduly impede competition in product markets. *See* Richard G. Frenkel, Comment, *Intellectual Property in the Balance: Proposals for Improving Industrial Design Protection in the Post-TRIPs Era*, 32 LOY. L.A. L. REV. 531, 575-81 (1999). For a discussion of industrial design protection more generally and why it has been controversial over the years, see J.H. Reichman, *Design Protection and the New Technologies: The United States Experience in a Transnational Perspective*, 19 U. BALT. L. REV. 6 (1989). The expansion of state and federal trade dress protection for product configurations, however, has had much the same effect as an industrial design protection law would have in the United States. *Id.* The functionality limitation on trade dress protection limits the utility of this law as a surrogate for a European-style utility model law.

86. E-mail from Michael Moradzadeh, former Executive, Intel Corp., to Pamela Samuelson, Professor of Law and Information Management, University of California at Berkeley (Apr. 26, 2001) (on file with authors).

III. REVERSE ENGINEERING IN THE SEMICONDUCTOR INDUSTRY

The semiconductor industry is in many respects a traditional manufacturing industry. However, we give it separate treatment here for two reasons. First, semiconductors are information technology products that bear a high quantum of the know-how required to make them on their face.[87] This made them vulnerable to rapid, cheap, competitive cloning that industry leaders asserted undermined their ability to recoup the very high costs of R&D necessary to produce new chips.[88] Second, Congress responded to these industry concerns about "chip piracy"[89] by creating a new form of intellectual property protection for semiconductor chip designs.[90]

The Semiconductor Chip Protection Act (SCPA)[91] is noteworthy for a number of reasons.[92] First, it is one of the few intellectual property laws[93]

87. *See* Morton D. Goldberg, *Semiconductor Chip Protection as a Case Study*, *in* GLOBAL DIMENSIONS OF INTELLECTUAL PROPERTY RIGHTS IN SCIENCE AND TECHNOLOGY 329, 333 (Mitchel B. Wallerstein et al. eds., 1993) ("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."); Reichman, *supra* note 14, at 2479-80 (noting that the semiconductor industry is "an industry where know-how is easily appropriated by technological means"); Samuelson et al., *supra* note 15, at 2338 (discussing the vulnerability of information technology products to market-destructive appropriations because of the high quantum of know-how they bear on their face when sold in the marketplace).

88. *See, e.g.*, *Copyright Protection for Semiconductor Chips: Hearing on H.R. 1028 Before the Subcomm. on Courts, Civil Liberties, and the Admin. of Justice of the House Comm. on the Judiciary*, 98th Cong. 21-28 (1983) [hereinafter *House Hearings*] (statement of F. Thomas Dunlap, Jr., Corporate Counsel and Secretary, Intel Corp.) (explaining the industry's need for this legislation).

89. *See id.* at 3 (statement of Sen. Mathias) ("[Chip] innovators are being ripped-off by onshore and offshore 'chip pirates,' who, for a fraction of the developers' cost, can now legally appropriate and use these chip designs as their own."). Of particular concern was the loss to American 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. Steven P. Kasch, *The Semiconductor Chip Protection Act: Past, Present, and Future*, 7 HIGH TECH. L.J. 71, 79 (1992).

90. Commentators have suggested that the semiconductor industry "greatly overstated the severity of the chip piracy problem" in testimony before Congress. Robert L. Risberg, Jr., Comment, *Five Years Without Infringement Litigation Under the Semiconductor Chip Protection Act: Unmasking the Spectre of Chip Piracy in an Era of Diverse and Incompatible Process Technologies*, 1990 WIS. L. REV. 241, 244-45; *see also* Kasch, *supra* note 89, at 92-96 (questioning the evidence of chip piracy at the legislative hearings).

91. Semiconductor Chip Protection Act, Pub. L. No. 98-620, 98 Stat. 3347 (1984) (codified at 17 U.S.C. §§ 901-914 (1994)).

92. The SCPA has been the subject of much commentary. *See, e.g.*, Kasch, *supra* note 89; 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); Linda B. Samuels & Jeffrey M. Samuels, *Semiconductor Chip Protection Act of 1984: An Analytical Commentary*, 23 AM. BUS. L.J. 601 (1986); Terril G. Lewis, Comment, *Semiconductor Chip Process Protection*, 32 HOUS. L. REV. 555 (1995); Risberg, *supra* note 90; *see also* ANDREW CHRISTIE, INTEGRATED CIRCUITS AND THEIR CONTENTS (1995); RICHARD H. STERN, SEMICONDUCTOR CHIP PROTECTION (1986); Symposium, *The Semiconductor Chip Protection Act of 1984 and Its Lessons*, 70 MINN. L. REV. 263 (1985).

with an express reverse engineering privilege.[94] Second, the privilege permits the copying of protected chip designs in order to study the layouts of circuits, and also the incorporation of know-how discerned from reverse engineering in a new chip.[95] Third, the SCPA requires reverse engineers to engage in enough "forward engineering" to develop an original chip design that itself qualifies for SCPA protection.[96] This is in contrast to the predominant legal rule for manufacturing industries that permits reverse engineers to make and sell products identical or nearly identical to those they have reverse-engineered.[97] The economic rationale for the forward engineering requirement was not articulated with precision during the SCPA debate, but we think it is fundamentally sound as applied to this industry.

A. *Perturbations in Product Life Cycles in the Chip Industry*

The typical product life cycle in the semiconductor industry was relatively constant in the 1970s and 1980s.[98] 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 manufacturer [became] more efficient it [would cut] prices to expand its market and discourage competition. Nonetheless, second-source

93. Although trade secrecy is sometimes characterized as a form of intellectual property protection, *e.g.*, Stanley M. Besen & Leo J. Raskind, *An Introduction to the Law and Economics of Intellectual Property*, J. ECON. PERSP., Winter 1991, at 3, 3, it is more appropriately understood as a branch of unfair competition law. *See* RESTATEMENT (THIRD) OF UNFAIR COMPETITION, *supra* note 27, §§ 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.

94. Professor Raskind has spoken of the reverse engineering privilege as the "capstone" of the SCPA. Leo J. Raskind, *Reverse Engineering, Unfair Competition, and Fair Use*, 70 MINN. L. REV. 385, 385 (1985). The reverse engineering privilege of the SCPA also received attention in other commentary, including Lee Hsu, *Reverse Engineering Under the Semiconductor Chip Protection Act: Complications for Standard of Infringement*, 5 ALB. L.J. SCI. & TECH. 249 (1996), and Harold R. Brown, Note, *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 SYRACUSE L. REV. 985 (1990).

95. 17 U.S.C. § 906(a). Indeed, a congressional explanatory memorandum about the SCPA states that chip designs produced by this sort of reverse engineering would be noninfringing unless they were substantially identical to the reverse-engineered chip. *See* 130 CONG. REC. 28,960 (1984) (reprinting the explanatory memorandum to the Mathias-Leahy Amendment to Senate Bill 1201). Section 906 differs from patent rules in two significant respects: First, it creates a right in unlicensed firms to engage in intermediate copying of the protected innovation, which patent law does not, and second, it allows the new product resulting from reverse engineering to be free from blocking intellectual property rights, as would generally be the case with patents as to subsequent inventions substantially incorporating the innovator's invention.

96. *See* Kasch, *supra* note 89, at 85 (discussing forward engineering in the context of SCPA legal analysis); Elliot J. Chikofsky & James H. Cross II, *Reverse Engineering and Design Recovery: A Taxonomy*, IEEE SOFTWARE, Jan. 1990, at 13, 14-15 (defining forward engineering).

97. *See supra* Section II.A.

98. Kasch, *supra* note 89, at 78 (discussing the life cycle in the industry).

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."[99] Toward the end of this life cycle, the pioneer's profit margins would trail off, and it would have to hope that the next round of innovation would allow it to regain market share and profits.

Semiconductor firms have historically relied on lead time and secrecy far more than on patents to protect their intellectual assets.[100] 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.[101] 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 obviously could not protect the layout of chips sold in the marketplace, as this information was readily ascertainable from examination of the marketed product (that is, it could be readily reverse-engineered).[102]

Several factors contributed to the fact that patents did not play a crucial role in the early and mid-development phases of this industry.[103] For one thing, semiconductors are a cumulative system technology in which the interrelatedness of inventions requires extensive cross-licensing of patents in order for industry participants to make advanced chips.[104] Second, some major customers of this industry, notably the U.S. government, insisted on "second-sourcing," that is, attracting competitive suppliers of compatible

99. *Id.*

100. COHEN ET AL., *supra* note 45, tbls.1-2 (showing that semiconductor firms regard trade secrecy and lead time as far more effective than patents in protecting firm intellectual assets from market-destructive appropriations); *see also* Levin et al., *supra* note 53 (discussing the reliance of many manufacturing industries on lead time protection instead of patents).

101. Early stages of a chip production process generally result in a lower yield of salable chips than later stages, when fine-tuning of the production process yields a higher quantity of salable chips.

102. *See* Reichman, *supra* note 14, at 2478-80.

103. *See* Bronwyn H. Hall & Rosemarie Ham Ziedonis, *The Patent Paradox Revisited: An Empirical Study of Patenting in the U.S. Semiconductor Industry, 1979-1995*, 32 RAND J. ECON. 101, 119 tbl.2 (2000) (showing the pattern of patenting in the semiconductor industry over this period).

104. *See* DEEPAK SOMAYA & DAVID J. TEECE, COMBINING INVENTIONS IN MULTI-INVENTION PRODUCTS: ORGANIZATIONAL CHOICES, PATENTS, AND PUBLIC POLICY (SSRN Elec. Library, Working Paper No. 259,889, 2000), http://papers.ssrn.com/paper.taf?abstract_id=259889 (discussing the implications for patent policy of products that incorporate large numbers of technologies); Hall & Ziedonis, *supra* note 103, at 102 (characterizing the semiconductor industry as involving cumulative system technology and emphasizing the importance of cross-licensing in this industry); Robert P. Merges & Richard R. Nelson, *On the Complex Economics of Patent Scope*, 90 COLUM. L. REV. 839 (1990) (same); Suzanne Scotchmer, *Standing on the Shoulders of Giants: Cumulative Research and the Patent Law*, J. ECON. PERSP., Winter 1991, at 29, 29 (discussing cumulative system technologies); *see also* Risberg, *supra* note 90, at 249 (noting widespread licensing in the semiconductor industry).

chips to reduce the risk of unforeseen supply problems.[105] 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.[106] Fourth, during the 1970s, 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.[107] A fifth limitation of patents, much emphasized in the legislative history of the SCPA, was that under then-prevailing standards, the overall layout of chip circuits was rarely if ever patentable.[108]

While the U.S. semiconductor industry thrived for years under these conditions, the life-cycle pattern of chip products was so disrupted during the late 1970s and early 1980s that leading chip producers sought legislative help. Several factors contributed to this disturbance. First, there was a steep rise in the cost of developing and marketing new chips.[109] Second, advances in chip manufacturing technologies dramatically reduced the cost 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.[110] 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.[111]

B. *Copyright or Sui Generis Protection for Chip Designs?*

Intel Corp. initially sought to combat "chip piracy" with copyright law. It obtained copyright registration certificates for drawings of chip circuitry,[112] and then it sought to register masks (that is, stencils used in manufacturing chips) and chips themselves as derivative works of the drawings. This would have provided a basis for claiming that manufacturers of identical or near-identical chips were infringing copyrights in protected

105. *See* Kasch, *supra* note 89, at 96-98 (discussing second-sourcing); Risberg, *supra* note 90, at 247 n.29 (discussing licensing as a way to accomplish second-sourcing).

106. Hall & Ziedonis, *supra* note 103, at 102; *see also* Goldberg, *supra* note 87, at 330 (discussing the manner in which innovation outpaced patent effectiveness).

107. Risberg, *supra* note 90, at 266-67. As patents grew progressively stronger in the 1980s, chip firms increased the rate of their patenting. Hall & Ziedonis, *supra* note 103, at 104; Risberg, *supra* note 90, at 267-77.

108. *See* H.R. REP. NO. 98-781, at 3-4 (1984), *reprinted in* 1984 U.S.C.C.A.N. 5750, 5752.

109. *See* Kasch, *supra* note 89, at 78-79 (estimating the costs of new chip development at $40-$50 million by 1983).

110. *See id.* (estimating the costs of chip cloning, a three- to six-month process, at $50,000-$100,000).

111. *See id.* at 79; *see also* Raskind, *supra* note 94, at 413-15 (describing the decline of the domestic semiconductor industry and the increasing Japanese market share).

112. *See* Kasch, *supra* note 89, at 80.

drawings, masks, or chips. Intel's strategy was derailed when the U.S. Copyright Office rejected its application to register chips because of their utilitarian function.[113] Although Intel sued the Register of Copyrights to compel registration,[114] it soon dropped the litigation and turned to Congress for legislative relief.[115]

Intel's second strategy was also based on copyright. It asked Congress to amend the copyright law to add "mask works" to the subject matter of copyright.[116] 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.[117] A nearly identical argument was made by the congressional Commission on New Technological Uses of Copyrighted Works (CONTU), which supported the use of copyright law to protect computer programs.[118] Because programs and chips are both utilitarian information technology products that are expensive to develop and cheap and easy to copy, one might have thought that copyright should be used for both or for neither. Yet, the copyright argument was successful as to programs,[119] but not as to chips.

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 could continue.[120] An explicit

113. *Id.*; *see also House Hearings*, *supra* note 88, at 87-88 (statement of Dorothy Schrader, Associate Register of Copyrights for Legal Affairs, Copyright Office, Library of Congress) (questioning the registrability of masks because of their role in the process of manufacturing chips).

114. *House Hearings*, *supra* note 88, at 88 n.10 (statement of Dorothy Schrader, Associate Register of Copyrights for Legal Affairs, Copyright Office, Library of Congress).

115. Stern gives a chronology of the legislative activity on the chip bills. STERN, *supra* note 92, app. B, at 493-95. He reports that the first bill was introduced in Congress in 1978 to protect chip designs through copyright law. 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 to move forward and pass. *Id.*

116. Kasch, *supra* note 89, at 80.

117. *See House Hearings*, *supra* note 88, at 21-29 (statement of F. Thomas Dunlap, Jr., Corporate Counsel and Secretary, Intel Corp.).

118. NAT'L COMM. ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, FINAL REPORT 12-13 (1978) [hereinafter CONTU REPORT]. *But see* Pamela Samuelson, *Creating a New Kind of Intellectual Property: 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 be protected by the same type of law).

119. *See* An Act To Amend the Patent and Trademark Laws, Pub. L. No. 96-517, § 10, 94 Stat. 3015, 3028 (1980) (codified at 17 U.S.C. § 117 (1994)) (implementing CONTU's recommendations for amending copyright law to protect programs).

120. *See* Kasch, *supra* note 89, at 81 (reporting the sharp industry divide in the first hearing on chip legislation).

reverse engineering privilege was added to a later bill. However, it allowed reproducing a chip design for study and analysis without expressly allowing reverse engineers to use the results in designing a new chip.[121] 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 necessarily be quite similar to the chips being reverse-engineered, although not necessarily in a competitively harmful way.[122]

Lack of industry consensus stalled movement on 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.[123] Yet those compromises so deviated from traditional copyright rules that a new and different kind of opposition arose.[124] 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 infringers and the extension of compulsory licensing; and, most notably, the limitation imposed on the duration of protection of this particular class, and the distortion of the fair use doctrine to accommodate reverse engineering.[125]

It would be better, he argued, to develop sui generis legislation[126] to protect semiconductor chip designs[127]—which is what Congress ultimately did in 1984.

The SCPA regime resembles copyright in significant respects.[128] One conceptual holdover from Intel's copyright strategy was the subject matter

121. *Id.* at 82.

122. *See* Brown, *supra* note 94, at 997-99.

123. *See* Kasch, *supra* note 89, at 82.

124. *See, e.g.*, S. 1201, 98th Cong. (1983), *discussed in House Hearings*, *supra* note 88, at 128-33 (statement of Dorothy Schrader, Associate Register of Copyrights for Legal Affairs, Copyright Office, Library of Congress) (comparing the main features of the sui generis and copyright bills).

125. *House Hearings*, *supra* note 88, at 11-12 (statement of Jon A. Baumgarten, Copyright Counsel, Association of American Publishers); *see also id.* at 12 n.2 (expressing doubt that reverse engineering would be fair use under traditional principles of copyright law).

126. *See* Reichman, *supra* note 14, at 2453-504 (discussing various sui generis regimes).

127. *House Hearings*, *supra* note 88, at 11 (statement of Jon A. Baumgarten, Copyright Counsel, Association of American Publishers).

chosen for SCPA protection, namely, mask works.[129] As with copyright, mask works must be "original" to qualify for protection.[130] Rights attach automatically by operation of law, but registration with the Copyright Office brings benefits unavailable to nonregistrants.[131] The legislative history demonstrates that copyright-like concepts of substantial similarity and substantial identity were to be used in judging infringement of SCPA rights.[132] And the SCPA relies, as copyright does, on a grant of exclusive rights to control reproductions and distributions of products embodying the protected work.[133]

A notably sui generis feature of the SCPA[134] is its reverse engineering provision:

> [I]t is not an infringement of the exclusive rights of the owner of a mask work for—

128. Reichman, *supra* note 14, at 2478-79 (discussing similarities between the SCPA and copyright law).

129. 17 U.S.C. § 902 (1994). 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 87/54/EEC on the Legal Protection of Topographies of Semiconductor Products, 1987 O.J. (L 24) 36 [hereinafter Council Directive]. A serious disadvantage of mask works as the protected subject matter under the SCPA is that its technology-specific nature meant that the SCPA would become obsolete if chip production moved beyond the use of masks in the manufacturing process—as indeed has occurred. Goldberg, *supra* note 87, at 333.

130. 17 U.S.C. § 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, combined in a way that, considered as a whole, is not original." *Id.* § 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 before subsequent chips would be deemed noninfringing. *See* Brown, *supra* note 94, at 991-92; Risberg, *supra* note 90, at 262.

131. 17 U.S.C. § 908 (stating, inter alia, that rights under the SCPA terminate unless the chip design is registered within two years); *see also id.* § 412 (stating that the right to statutory damage awards and to recovery of attorney's fees depends on prompt registration of copyright claims with the Copyright Office).

132. *See, e.g.*, H.R. REP. NO. 98-781, at 20-23 (1984), *reprinted in* 1984 U.S.C.C.A.N. 5750, 5769-72 (anticipating the use of copyright-like concepts of substantial similarity and substantial identity in infringement decisions). Second comers, however, cannot hope to make a workable compatible chip merely by making minor variations on an innovative chip design in order to avoid infringement. As one commentator has noted, "very subtle variations in logic flow, or in certain arrangement configurations, may make interchangeability impossible." Brown, *supra* note 94, at 998.

133. *Compare* 17 U.S.C. § 905 (laying out the SCPA's exclusive rights provision), *with id.* § 106 (giving copyright law'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.

134. The SCPA contains a number of novel and specially tailored provisions apart from the reverse engineering privilege. Samuelson, *supra* note 118, at 492-501 (discussing other sui generis features of the SCPA).

> (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.[135]

Industry witnesses distinguished "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 lay out a different pattern. This pattern could be used to fabricate a version of the semiconductor chip 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 cost of the chip.[136]

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.[137]

The House Report on the 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 creation of the first chip.

---

135. 17 U.S.C. § 906(a). Similar provisions exist in other laws protecting chip designs. *See, e.g.*, Council Directive, *supra* note 129, arts. 5(2)-(4), 1987 O.J. (L 24) at 38.

136. *House Hearings*, *supra* note 88, at 27-28 (statement of F. Thomas Dunlap, Jr., Corporate Counsel and Secretary, Intel Corp.).

137. 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 House report gives no weight to the first set of factors but some weight to the latter. *See* H.R. REP. NO. 98-781, at 21, *reprinted in* 1984 U.S.C.C.A.N. at 5770.

> 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 was the product of substantial study and analysis, and not the mere result of plagiarism accomplished without such study or analysis.[138]

One commentator characterized the SCPA as "accept[ing] 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."[139]

C. *An Economic Rationale for the SCPA Rules*

Part II argued that reverse engineering does not unduly undermine incentives to invest in innovation as long as it is costly, time-consuming, or both. During the time that the SCPA and predecessor bills were pending in Congress, reverse engineering of chips could be done very cheaply and quickly by peeling away layers of a purchased chip, one at a time, photographing each layer, making a mask from these photographs, and then using these masks to manufacture identical chips.[140] The SCPA rules made this cheap and rapid route to competitive entry illegal and required reverse engineers to design original chips in order to avert infringement liability. The forward engineering requirement lengthened second comers' development time and increased their costs, thereby giving the innovator more lead time to recoup its R&D expenses and more protection against clone-based pricing. The forward engineering requirement also increased the likelihood that second comers would advance the state of the art in semiconductor design.[141] As long as second comers had to make their chips different, they might as well make them better.

---

138. H.R. REP. NO. 98-781, at 22, *reprinted in* 1984 U.S.C.C.A.N. at 5771.

139. Raskind, *supra* note 94, at 391. Shortly after the enactment of the SCPA, Professor Raskind predicted:

> When 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 shows, this prediction has proved accurate.

140. *See* H.R. REP. NO. 98-781, at 2-3, *reprinted in* 1984 U.S.C.C.A.N. at 5751-52.

141. *See generally* Ted O'Donoghue et al., *Patent Breadth, Patent Life, and the Pace of Technological Progress*, 7 J. ECON. & MGMT. STRATEGY 1 (1998) (discussing the effective life of intellectual property protection in rapidly evolving sequential technologies and how the breadth of protection interacts with this phenomenon).

Table 2 uses the same social welfare criteria as Table 1 to illustrate our assessment of the economic effects of pre-SCPA rules as compared with post-SCPA rules.

TABLE 2. SOCIAL CALCULUS OF REVERSE ENGINEERING IN THE CHIP INDUSTRY PRE- AND POST-SCPA

Social Welfare Criterion	Pre-SCPA	Post-SCPA
Incentives to innovate	Worse (too little)	Better
Incentives to improve	Worse (too little)	Better
Prices	Lower (but too low)	Higher
Wasted costs	Better	Worse (but avoidable by licensing)

Incentives to invest in innovative chip designs were too low before enactment of the SCPA because cloners rapidly eroded lead time advantages for innovators. In the short run, this may have brought low prices and few wasted costs, but prices were too low to allow innovators to recoup R&D expenses as long as cloning was legal. Incentives to innovate were restored once cloning was no longer an option. Incentives to invest in follow-on innovation were also very low in the pre-SCPA era because firms capable of investing in improved chips chose instead to clone while it was still legal. When chip cloning became illegal, firms had strong incentives to invest in improvements. Although consumers may have initially benefited from lower prices in the pre-SCPA era, prices were so low that innovators could not recoup their costs. The SCPA may result in more socially wasteful costs because some second comers may spend resources making chip circuitry different to satisfy the originality requirements. However, some of these wasted costs are avoidable by licensing.

From an economic standpoint, the anticloning rules of the SCPA are designed to achieve much the same result as the anti-plug-mold rules discussed in Part II, although they do so by a different technique. Chip cloners were no more engaged in innovation-enhancing discovery of applied industrial know-how than were plug-molders. The SCPA rule inducing second comers to join the ranks of innovation-enhancing firms is similar to the anti-plug-mold rule that induced second comers to engage in more conventional forms of reverse engineering likely to advance the state of the art of boat-hull design. The SCPA achieves this result by establishing a kind of "breadth" requirement for subsequent products in contrast to the

anti-plug-mold laws that instead outlawed a particular means for making a competing product.[142]

D. *Post-SCPA Developments*

There has been very little litigation under the SCPA rules. Yet the one reported judicial decision under the SCPA is instructive because it involved a failed reverse engineering defense. In *Brooktree Corp. v. Advanced Micro Devices, Inc.*,[143] 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.[144] It also emphasized many differences between the layout of its chip circuitry and Brooktree's.[145] However, under pressure from an impending deadline, AMD's principal designer revisited the Brooktree chip layout and thereafter abandoned his plans for a six or eight transistor core cell design in favor of the same ten transistor design arrangement in Brooktree's chip.[146] The court of appeals concluded that "[a] reasonable jury could have inferred 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."[147] While AMD surely made a far greater investment in engineering than the cloning firms at which the SCPA was principally aimed, AMD did not, as the 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 the SCPA is as a sign that the law successfully deterred chip piracy. However, most legal

142. The notion of "breadth" has no formal meaning in law. However, the economics literature has interpreted breadth as measuring how much a product must be improved to avoid infringing a prior patent. *See* Jerry R. Green & Suzanne Scotchmer, *On the Division of Profit in Sequential Innovation*, 26 RAND J. ECON. 20, 21, 23 (1995); O'Donoghue et al., *supra* note 141, at 2-3. In patent law, the requirements of nonobviousness and novelty jointly govern both the "breadth" of a patent and the advance over prior art required for patentability. These requirements are joined in the SCPA, so that any improvement either escapes infringement and receives protection as a joint package or does neither. In general the requirements necessary to escape infringement and receive protection are different. See Mark A. Lemley, *The Economics of Improvement in Intellectual Property Law*, 75 TEX. L. REV. 989 (1997), for a discussion of copyright law and patent law in this regard. We emphasize breadth in our discussion of the SCPA because the SCPA solved the problem of cloning by providing that clones infringe, and it solved the problem of encouraging improvement by allowing the reverse engineer to escape infringement by improving the chip. *See infra* Subsection VI.A.2 (discussing breadth requirements for products of reverse engineering).

143. 977 F.2d 1555 (Fed. Cir. 1992).

144. *Id.* at 1566-67.

145. *Id.*

146. *Id.* at 1567-68.

147. *Id.* at 1569.

commentators have inferred from this that the SCPA is unimportant.[148] Some put the blame on bad drafting, claiming that the SCPA is technologically obsolete or provides too thin a scope of legal protection.[149] Others assert that the SCPA became unimportant because of subsequent legal developments, such as the renewed importance of patents in the aftermath of the creation of the Federal Circuit or the rise of second-source licensing agreements between pioneers and follow-on innovators.[150] Still others assert that technological changes, 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 alternative layouts, rendered infeasible the kind of copying that gave rise to the SCPA.[151]

One indication of a continuing interest in the SCPA among chip designers can be found in the number of chip designs registered with the U.S. Copyright Office and counterpart agencies elsewhere.[152] Legal protection for the layout of integrated circuits was also deemed important enough to warrant its inclusion in the TRIPS Agreement.[153] TRIPS

148. *See, e.g.*, Kasch, *supra* note 89, at 72 (arguing that the SCPA is of "largely academic interest"); Risberg, *supra* note 90, at 245 (describing the SCPA as "a largely untested, if not impotent, piece of legislation").

149. Goldberg, *supra* note 87, at 332-35 (making both complaints).

150. Hall & Ziedonis, *supra* note 103, at 104 (attributing a substantial increase in patenting in the semiconductor industry to a strong "pro-patent" shift in the U.S. legal environment).

151. Kasch, *supra* note 89, at 73, 103; Risberg, *supra* note 90, at 273-76. Kasch predicted that further changes in technology might cause the SCPA's anticloning protection to have renewed importance in the future. Kasch, *supra* note 89, at 103-04.

152. Risberg, *supra* note 90, at 243 n.16 (reporting that the Copyright Office accepted 4291 mask work registrations in the first four years of administering the SCPA); *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 United States. 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. E-mail from Rosemarie Ziedonis, Assistant Professor of Management, University of Pennsylvania, to Pamela Samuelson, Professor of Law and Information Management, University of California at Berkeley (May 18, 2001) (on file with authors).

153. *See* TRIPS Agreement, *supra* note 3, arts. 35-38, 33 I.L.M. at 97. 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, the 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. The SCPA established a process under which U.S. officials could judge whether other nations had adopted sufficient laws. *See* 17 U.S.C. § 914 (1994). Although the United States was able to persuade many other nations to adopt chip protection laws, *see, e.g.*, Council Directive, *supra* note 129; STERN, *supra* note 92, §§ 10.1-10.3, at 379-444, there has been some resentment among intellectual property professionals in other countries about the U.S. reciprocity approach. This approach also came back to haunt the United States when the European Commission decided to adopt a new form of legal protection for the

incorporates by reference a number of provisions of an earlier treaty on legal protection for the layout of integrated circuits, including a reverse engineering privilege closely modeled on the SCPA rule.[154] The semiconductor chip industry, as a consequence, is the only industry whose reverse engineering activities are expressly protected in an international intellectual property treaty.

In the years since the SCPA's enactment, the semiconductor industry has enjoyed very considerable growth, and U.S. firms have dominated a larger global chip market.[155] Interestingly, in the post-SCPA era, there has been a partial bifurcation of the design and fabrication components of the chip industry.[156] 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 more aggressive enforcement of patent rights, especially by the design firms.[157] From an economic perspective, if the 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 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 it is in the traditional manufacturing and semiconductor industries.[158] For much of the past two decades, however, the legality of two common forms of software reverse engineering, namely,

---

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* J.H. Reichman & Pamela Samuelson, *Intellectual Property Rights in Data?*, 50 VAND. L. REV. 51, 96 & n.198 (1997) (discussing the European database directive and the initial U.S. objections to its reciprocity provision).

154. *See* TRIPS Agreement, *supra* note 3, art. 35, 33 I.L.M. at 97 (incorporating various provisions of the Treaty on Intellectual Property in Respect of Integrated Circuits, May 26, 1989, 28 I.L.M. 1477, commonly known as the "Washington Treaty," including Article 6(2)'s reverse engineering privilege). The United States and other developed chip-producing countries objected to some provisions of the Washington Treaty and refused to sign it. *See* Goldberg, *supra* note 87, at 335-36 (discussing various complaints about 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 Agreement, *supra* note 3, arts. 36-37, 33 I.L.M. at 97.

155. *See* Hall & Ziedonis, *supra* note 103. *See generally* U.S. DEP'T OF COMMERCE, THE EMERGING DIGITAL ECONOMY app. 1 (1998), http://www.ecommerce.gov/emerging.htm (reporting on the high growth of information technology industries, including the semiconductor industry, in the United States).

156. *See* Hall & Ziedonis, *supra* note 103, at 104-05.

157. *Id.* One would expect design firms to rely not only on patents (as they apparently do, *id.* at 104), but also on the legal protection the SCPA provides against copying of chip layouts, although the latter has not been documented.

158. *See* 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 . . . .").

decompilation and disassembly of object code,[159] has been challenged on trade secret, copyright, patent, and contract law theories. This Part first reviews the legal debate about reverse engineering of computer software as a matter of intellectual property law and explains why courts and legal commentators have overwhelmingly supported the legality of such reverse engineering. It then goes on to assess the economic effects of decompilation and disassembly of program code, particularly when done for purposes of developing a program capable of interoperating with another program. The economic case for allowing reverse engineering to achieve interoperability is not as open and shut as some legal commentators have suggested.[160] We believe, however, that interoperability has, on balance, more beneficial than harmful economic consequences. Hence, a legal rule permitting reverse engineering of programs to achieve interoperability is economically sound. This Part concludes with a discussion of the legal debate over enforceability of contractual restrictions on reverse engineering of computer software and economic reasons for not enforcing them.

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 the functionality that object code forms of programs provide and do not want to read the program's text; and second, because the developers want to maintain source code forms of their products and other human-readable documentation as trade secrets.[161] Decompilation or disassembly of object code provides a way for reverse engineers to "work[] backwards from object code to produce a simulacrum of the original source code."[162] From this approximation of source code,

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

160. *See infra* note 188.

161. *See* Jessica Litman, *Copyright and Information Policy*, LAW & CONTEMP. PROBS., Spring 1992, at 185, 196-201 (discussing strategies of software industry lawyers for maintaining the internal aspects of programs as trade secrets); *see also* Reichman, *supra* note 1, at 701 (describing the nondisclosure of internal program information as a business imperative, although concluding that second comers ought to be able to reverse-engineer object code).

162. Cohen & Lemley, *supra* note 12, at 16 n.52. Litman explains:

> 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 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 product of such reverse

reverse engineers can discern or deduce internal design details of the program, such as information necessary to develop a program that will interoperate with the decompiled or disassembled program. Lawyers for some major software producers argue that decompilation and disassembly should be illegal as a matter of copyright and trade secrecy law. They argue that the unauthorized copies of programs made in the process of decompiling or disassembling them infringe the program copyright, and this infringement makes the decompilation or disassembly an improper means of obtaining program trade secrets.[163]

---

engineering will include only the parts of the program that were compiled into object code in the first instance; the English language 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.

Litman, *supra* note 161, at 197-98.

163. *See, e.g.*, Allen R. Grogan, *Decompilation and Disassembly: Undoing Software Protection*, COMPUTER LAW., Feb. 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) 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.C for further discussion of the shrinkwrap license issues pertaining to reverse engineering of software. For similar arguments, see Anthony L. Clapes, *Confessions of an Amicus Curiae: Technophobia, Law, and Creativity in the Digital Arts*, 19 U. DAYTON L. REV. 903 (1994); Duncan M. Davidson, *Common Law, Uncommon Software*, 47 U. PITT. L. REV. 1037 (1986); and Arthur R. Miller, *Copyright Protection for Computer Programs, Databases and Computer-Generated Works: Is Anything New Since CONTU?*, 106 HARV. L. REV. 977 (1993).

The predominant view among legal commentators, however, supports a right to reverse-engineer software under copyright law. *See, e.g.*, Brief of Amici Curiae of Eleven Copyright Law Professors, Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510 (9th Cir. 1992) (No. 92-15655), *reprinted in* 33 JURIMETRICS J. 147 (1992); JONATHAN BAND & MASANOBU KATOH, INTERFACES ON TRIAL: INTELLECTUAL PROPERTY AND INTEROPERABILITY IN THE GLOBAL SOFTWARE INDUSTRY 167-225 (1995); Julie 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 D. Graham & Richard O. Zerbe, Jr., *Economically Efficient Treatment of Computer Software: Reverse Engineering, Protection, and Disclosure*, 22 RUTGERS COMPUTER & TECH. L.J. 61 (1996); Dennis S. Karjala, *Copyright Protection of Computer Documents, 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 of Computer Software*, 30 JURIMETRICS J. 15 (1989); Lemley & McGowan, *supra* note 47; Litman, *supra* note 161, at 196-201; 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, *supra* note 1; 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); Samuelson et al., *supra* note 15; 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); 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.

The principal decision testing this legal theory was *Sega Enterprises Ltd. v. Accolade, Inc.*[164] 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.[165] 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."[166] Reverse engineering was, moreover, the only way that Accolade could gain access to this information.[167] Although Accolade had copied the whole of Sega's programs in the course of its reverse analysis, the court discounted this conduct 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, they did not do so in a way about which copyright law is concerned.[168] 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."[169] An important policy

164. 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 be fair use in appropriate circumstances. *Atari Games Corp. v. Nintendo of America, Inc.*, 975 F.2d 832 (Fed. Cir. 1992), was decided shortly before the Ninth Circuit decision. The *Atari Games* 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's 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 836. The Federal Circuit ruled that the initial decompilation copying was fair use. *Id.* at 843.

165. 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. 17 U.S.C. § 107 (1994). It is interesting to note that *Sega* relied in part on the legislative history of the SCPA, 977 F.2d at 1521, in which some witnesses had expressed doubt that reverse engineering could be fair use as a matter of copyright law, *House Hearings*, *supra* note 88, at 11-12 (statement of Jon A. Baumgarten, Copyright Counsel, Association of American Publishers, Inc.).

166. *Sega*, 977 F.2d at 1522.

167. *Id.* The court stated:

> The unprotected aspects of most copyrighted works 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 contained in object code without disassembling that code . . . .

Id. at 1525.

168. *Id.* at 1524. Copyright law is concerned with infringing copies that compete with the author's works, not with competition on the merits among noninfringing works.

169. *Id.* at 1523.

consideration was 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]."[170] To get a monopoly on such ideas and functional concepts, a creator needs to seek patent protection.[171]

Still, the court did not give a green light to all reverse engineering of program code, but only to that undertaken for a "legitimate reason," 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."[172]

The Ninth Circuit recently reaffirmed the *Sega v. Accolade* ruling in *Sony Computer Entertainment, Inc. v. Connectix Corp.*[173] The main difference between it and *Sega v. Accolade* was that Connectix disassembled Sony programs in order to develop emulation software to allow owners of Apple iMac computers to play Sony PlayStation games. That is, Connectix reverse-engineered in order to make a competing platform, not to make compatible games. The appellate court perceived no legal difference between the decompilation-for-interoperability considerations pertinent to development of competing platforms and those pertinent to games. In the wake of this loss, Sony has charged makers of emulation programs with patent infringement based on decompilation of its programs.[174] It will be interesting to see if the courts will be equally receptive to a decompilation-for-interoperability defense as a matter of patent law.[175]

---

170. *Id.* at 1527.

171. *Id.* at 1526.

172. *Id.* at 1518.

173. 203 F.3d 596 (9th Cir. 2000) (reaffirming and extending *Sega v. Accolade* to a defendant who reverse-engineered Sony games in order to develop software to enable users to play Sony games on Apple computers).

174. *See* Cohen & Lemley, *supra* note 12, at 21. 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.

175. Cohen and Lemley have cogently argued for a limited reverse engineering privilege in patent law to allow decompilation of computer programs. *Id.* at 18-37. 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." *Id.* at 6. They argue that "reverse engineering is an important means of preserving competition between different products and of preserving compatibility between products. In markets characterized by network effects, such as software, this latter objective is particularly important." *Id.* at 21. They also point out that "[r]everse engineering promotes the fundamental patent policies of disclosure and enablement, ensures that patents will not be leveraged to protect unprotectable components of software, preserves the balance sought by the intellectual property system as a whole, and also helps patentees enforce their rights." *Id.* at 22.

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

Sega v. Accolade has been followed in virtually all subsequent cases.[176] It has been widely praised by legal commentators.[177] It is also consistent with the rules of other nations.[178] Those who argued that decompilation was

---

rights defense, implied license, and misuse. *Id.* at 29-36. 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. *Id.* at 32. If courts decide otherwise, Cohen and Lemley argue for legislation to permit it. *Id.* at 36-37. We agree that the limited reverse engineering rule they propose is legally and economically sound. *See also* Maureen A. O'Rourke, *Toward a Doctrine of Fair Use in Patent Law*, 100 COLUM. L. REV. 1177 (2000) (arguing for a fair use defense in patent law in part to enable decompilation for interoperability). It is worth pointing out that even before the Ninth Circuit *Sega v. Accolade* decision, the Court of Appeals for the Federal Circuit had ruled that decompilation for purposes of interoperability could be a fair and noninfringing use of copyrighted programs. *See* Atari Games Corp. v. Nintendo of Am., Inc., 975 F.2d 832 (Fed. Cir. 1992). Perhaps this augurs well for the recognition of a similar limited privilege as a matter of patent law, albeit on grounds other than fair use.

176. *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).

177. *See, e.g.*, sources cited *supra* note 163.

178. 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 the 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. (C 91) 4. United States trade negotiators and representatives of some U.S. computer companies argued that this was as it should be. *See, e.g.*, Victor Siber, Letter to the Editor, *Interpreting Reverse Engineering Law*, IEEE SOFTWARE, July 1990, at 4 (explaining IBM's position against reverse engineering of software); *see also* BAND & KATOH, *supra* note 163, at 228-41 (discussing U.S. industry lobbying and government officials' positions on the software directive); Thomas C. Vinje, *The Legislative History of the EC Software Directive*, *in* A HANDBOOK OF EUROPEAN SOFTWARE LAW 39 (Michael Lehmann & Colin Tapper eds., 1993) (discussing the evolution of the European software directive as to interoperability provisions). The Commission's competition directorate, however, 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* Pamela Samuelson, *Comparing U.S. and EC Copyright Protection for Computer Programs: Are They More Different than They Seem?*, 13 J.L. & COM. 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, art. 6(1), 1991 O.J. (L 122) 42, 45 [hereinafter European Software Directive]; *see also* BRIDGET CZARNOTA & ROBERT J. HART, LEGAL PROTECTION OF COMPUTER PROGRAMS IN EUROPE: A GUIDE TO THE EC DIRECTIVE 73-86 (1991) (providing the Official Commentary on this provision of the Directive); A HANDBOOK OF EUROPEAN SOFTWARE LAW, *supra* (offering other commentary on the Directive). 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, although it does not say what they might be. Error correction and detecting infringement are two other legitimate reasons to decompile programs. *See* E.F. Johnson Co. v. Uniden Corp. of Am., 623 F. Supp. 1485 (D. Minn. 1985) (considering decompilation to detect infringement); Samuelson, *supra*, at 289 n.59 (arguing that decompilation to detect infringement should be lawful, even though the European Directive seems not to permit it).

The European Software Directive also limits the follow-on uses that can be made of information obtained in the course of decompilation. *See* European Software Directive, *supra*, art. 6(2), 1991 O.J. (L 122) at 45. One cannot, for example, publish information learned during reverse engineering. This puts at risk authors of books such as ANDREW SCHULMAN ET AL., UNDOCUMENTED WINDOWS: A PROGRAMMER'S GUIDE TO RESERVED MICROSOFT WINDOWS

and should be illegal predicted grievous harm to the software industry if this form of reverse engineering was deemed lawful. These predictions have not been borne out. The American software industry has done well since 1992 when the *Sega v. Accolade* decision came down.[179]

B. *The Economics of Interoperability and Software Reverse Engineering*

Sega v. Accolade and its progeny show that reverse engineering is undertaken in the software industry for reasons different from those in other industrial contexts studied thus far. In manufacturing industries, reverse engineering is mainly undertaken in order to make directly competing stand-alone products.[180] Copyright law protects programs from the cheapest and most rapid way to make a directly competing identical product, namely, copying program code exactly.[181] However, reverse engineering of object code is generally so difficult, time-consuming, and resource-intensive that it is not an efficient way to develop competing but nonidentical programs.[182] As one technologist has explained:

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 they learn in the course of their reverse engineering efforts. The official commentary to the European 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 allow others access to the information which he had then made public." CZARNOTA & HART, *supra*, at 81. The European Software Directive, in essence, converts copyright law into trade secrecy law with regard to internal elements of programs.

Europe's adoption of a decompilation-for-interoperability privilege and the *Sega v. Accolade* decision in the United States did not end the international debate about decompilation. U.S. officials continued to insist that decompilation should be unlawful. In the mid-1990s, for example, Japan considered a proposal to amend its copyright law to allow reverse engineering of software, but it dropped the proposal under intense pressure from U.S. officials. *See* T.R. Reid & Peter Behr, *A Software Fight's Blurred Battle Lines: U.S. Computer 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* BAND & KATOH, *supra* note 163, 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 decompilation-for-interoperability exceptions similar to those of the European Software Directive. *See* BAND & KATOH, *supra* note 163, at 271-82.

179. *See* PRICEWATERHOUSECOOPERS, CONTRIBUTIONS OF THE PACKAGED SOFTWARE INDUSTRY TO THE GLOBAL ECONOMY (1999), http://www.bsa.org/usa/globallib/econ/pwc1999.pdf.

180. *See supra* Section II.B.

181. *See* Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir. 1983) (holding that exact copying of Apple operating system programs infringed Apple's copyright).

182. *See* Andrew Johnson-Laird, *Software Reverse-Engineering in the Real World*, 19 U. DAYTON L. REV. 843, 843 (1994). That is not to say that reverse engineering to make a directly competing product is unknown in the software industry, but it is uncommon. *See, e.g.*, Alcatel USA, Inc. v. DGI Techs., Inc., 166 F.3d 772 (5th Cir. 1999) (ruling on the reverse engineering of telecommunications switching software to make a competing product); Secure Servs. Tech., Inc. v. Time & Space Processing, Inc., 722 F. Supp. 1354 (E.D. Va. 1989) (considering reverse engineering of embedded software in secure facsimile machines for the purpose of making

> [Software] [r]everse 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.[183]

A software reverse engineer must do considerable intellectual work to extract higher level abstractions and information from the text of the decompiled program, and still more work to incorporate what he or she has learned from this analysis into a new program.[184] In this respect, software resembles traditional manufacturing products. The reverse engineering of both types of products involves high costs and other difficulties, and this insulates producers from market-destructive reverse engineering and reimplementation.[185]

Given the high costs and difficulties of software reverse engineering, it may seem surprising that it is such a standard industry practice. Software engineers reverse-analyze programs for a variety of reasons, including to fix bugs, to customize the program for the user's needs (e.g., to add some firm-specific features), to detect infringement, and to learn what others have done.[186] We focus our economic assessment of reverse engineering in the

---

competing, compatible facsimile machine). Notice that both of these examples involve embedded software in a traditional manufactured product.

183. Johnson-Laird, *supra* note 182, at 896.

184. It is worth noting that the nature of reverse engineering activities in the software industry is different from their nature in manufacturing industries. Reverse engineering of manufactured products involves manipulation of physical objects. Reverse engineering of computer software involves analysis of program texts. Samuelson et al., *supra* note 15, at 2320.

185. This has caused some 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." *Id.* at 2392. Because the present state of decompilation technology does not permit trivial acquisition of equivalence, the *Manifesto* authors concluded that there is currently no economically sound reason to regulate decompilation. *Id.* If technological change shifted the balance and enabled rapid, inexpensive copying that would be market-destructive, it might become necessary to regulate decompilation to some degree. *Id.* at 2392-93. *But see* COMPUTER SCI. & TELECOMM. BD., NAT'L RESEARCH COUNCIL, INTELLECTUAL PROPERTY ISSUES IN SOFTWARE 78 (1991) (quoting an IBM executive expressing concern that reverse analysis of programs could allow illegal copying of the internal information of programs that would escape easy detection).

186. *See, e.g.*, OFFICE OF TECH. ASSESSMENT, *supra* note 159, at 148-50 (giving various reasons for decompiling or disassembling programs). Concerning bug-fixing and adaptations, see 17 U.S.C. § 117(a), (c) (1994); and Pamela Samuelson, *Modifying Copyrighted Software: Adjusting Copyright Doctrine To Accommodate a Technology*, 28 JURIMETRICS J. 179, 215-20 (1988). An example of reverse engineering to detect infringement can be found in *E.F. Johnson Co. v. Uniden Corp. of America*, 623 F. Supp. 1485 (D. Minn. 1985). Reverse engineering of software for purposes such as those identified above may be less onerous than reverse engineering for the purpose of making a directly competing nonidentical clone because the reverse engineer may not have to analyze the whole program, but only the parts where the bug is located or where necessary to add a particular feature.

software industry on interoperability for two reasons: first, because this has been the most economically significant reason for software reverse engineering; and second, because most of the litigation about software reverse engineering has involved interoperability issues.[187] As will become apparent, the economics of interoperability are more complicated than some previous commentators have suggested.[188]

1. *Incentives for Interoperable or Noninteroperable Strategies*

Before considering the role that reverse engineering plays in the interoperability debate, we discuss the incentives for firms to design their systems to be interoperable or noninteroperable. 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) and applications designed to run on it (e.g., Sega's Sonic the Hedgehog game or Lotus 1-2-3).[189] In the software industry, platforms and applications are not just complementary products; they are complementary parts of a system by virtue of their conformity to interfaces necessary for achieving interoperability. Platforms are typically designed first. If an application developer wants to make a program that will fully interoperate with a particular platform, he or she must have access to very precise details about how the platform receives and sends information.[190] Collectively, these

187. *See, e.g.*, Sony Computer Entm't, Inc. v. Connectix Corp., 203 F.3d 596 (9th Cir. 2000); Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510 (9th Cir. 1992); *see also* Atari Games Corp. v. Nintendo of Am., Inc., 975 F.2d 832 (Fed. Cir. 1992) (concerning reverse engineering to develop games that could be played on Nintendo consoles); Vault Corp. v. Quaid Software Ltd., 847 F.2d 255 (5th Cir. 1988) (concerning reverse engineering of copy-protection software to make software to bypass the copy-protection function). Reverse engineering of software has sometimes been done to develop a complementary service. *See, e.g.*, Allen-Myland, Inc. v. IBM Corp., 746 F. Supp. 520 (E.D. Pa. 1990) (concerning an engineering service that reverse-engineered IBM software to aid in the reconfiguration of leased computers for subsequent lease customers); Hubco Data Prods. Corp. v. Mgmt. Assistance, Inc., 219 U.S.P.Q. (BNA) 450 (D. Idaho 1983) (concerning reverse engineering to discover code blocking access to advanced features so a reverse engineer could remove the blocking code and provide cheaper access to the features); *see also* Bateman v. Mnemonics, Inc., 79 F.3d 1532, 1545-47 (11th Cir. 1996) (noting that compatibility considerations limit the scope of copyright protection in programs).

188. Other legal commentators have concluded that the economic consequences of reverse engineering in the software industry are relatively benign. *See, e.g.*, Graham & Zerbe, *supra* note 163, at 132-34; Lemley & McGowan, *supra* note 47, at 525. However, we believe that these analyses of the economic effects of reverse engineering are incomplete.

189. Application programs can sometimes serve as platforms for applications that interoperate with them. *See, e.g.*, Lewis Galoob Toys, Inc. v. Nintendo of Am., Inc., 964 F.2d 965 (9th Cir. 1992) (considering a Game Genie program designed to interoperate with Nintendo games and change certain aspects of the game displays).

190. One important role for platforms is to provide certain commonly needed services to applications. It is typically more efficient for the platform to do this, rather than requiring all developers of applications to write redundant code to do the same thing. Application developers, however, need to know how to invoke needed platform functionality. This requires knowing how the platform expects to receive instructions from applications for a function to be successfully

details are known as application programming interfaces (APIs). Some platform developers publish interfaces, some license them freely, and others maintain their APIs as closely held trade secrets.[191]

The developer of a new platform might decide to publish its interfaces or make them available under open license terms—an act that makes reverse engineering unnecessary—in order to make it easy for application developers to adapt existing applications or make new applications for the platform. An important reason to open interfaces is to drive demand for the new platform.[192] Only if desirable applications are available for the platform will consumer demand for the platform skyrocket. In the 1980s, for example, IBM, then a new entrant into the personal computer (PC) market, published technical specifications for the PC and required Microsoft to license the APIs to its operating system broadly to enable application developers to write programs for the IBM PC.[193] This resulted in "[a] large library of off-the-shelf IBM PC compatible application software (particularly Lotus 1-2-3) [that] made the IBM PC an attractive platform."[194] This allowed the IBM PC to achieve substantial market success rapidly.[195]

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 that took advantage of IBM's decision to embrace open architectures in the PC market. Consumers benefited from competition among IBM-compatible PCs and from a wide array of applications for this standard system. IBM,

---

invoked and how it sends information pertinent to that functionality. *See* BAND & KATOH, *supra* note 163, at 7.

191. *See* Cohen, *supra* note 163, at 1094; Samuelson et al., *supra* note 15, at 2402-03.

192. Another reason to open interfaces is to aid development of an open source platform capable of supporting a range of applications. The Linux/GNU operating system is the most widely known open source platform. *See* PETER WAYNER, FREE FOR ALL: HOW LINUX AND THE FREE SOFTWARE MOVEMENT UNDERCUT THE HIGH-TECH TITANS (2000). The popularity of some open source platforms, such as the Apache web server, has caused commercial firms such as IBM to include it in their systems. *Id.* at 181-83.

193. Ferguson and Morris write:

> 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 Raton [where IBM developed the PC] recognized that the best way to make the PC the industry standard 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 took Apple years to understand.

CHARLES H. FERGUSON & CHARLES R. MORRIS, COMPUTER WARS: HOW THE WEST CAN WIN IN A POST-IBM WORLD 29 (1993). Band and Katoh emphasize IBM's insistence on requiring Microsoft to license APIs broadly for its operating system for the PC. BAND & KATOH, *supra* note 163, at 30. A more recent example of a firm that freely publishes interface specifications for its platform is the maker of the popular Palm Pilot system. *See* Douglas Lichtman, *Property Rights in Emerging Platform Technologies*, 29 J. LEGAL STUD. 615, 616 (2000).

194. BAND & KATOH, *supra* note 163, at 30.

195. *Id.* (stating that "in 1984 alone, IBM's PC revenues were $4 billion").

however, lost market share in part because the openness of its PC architecture enabled the PC to be "commoditized" or cloned.[196]

Alternatively, firms may choose to keep their interfaces closed, not only as a defensive measure against the platform being commoditized, but as an offensive measure to capture the market.[197] Proprietary interfaces give the platform developer considerable control over applications available for the platform, in particular, the ability to insist that applications not be available for rival platforms.[198] The platform owner can ensure exclusivity either by developing the applications in-house or by making exclusivity a condition of licensing. Firms that tried to keep their interfaces proprietary included Sega and Nintendo. Both forbade licensees from making games for other platforms, 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).[199] The focus here is not on their attempts to stop software development for their platforms, but on their insistence that such development occur under license. Licensing would allow them to impose exclusivity.

By keeping its interface proprietary and by providing an exclusive set of applications, a platform owner has some hope of exploiting "network effects"[200] to become a de facto standard in the market. In fact, a single

196. *Id.* at 31 ("By the early 1990s, IBM sold only 23% of the IBM compatible PCs worldwide . . . ."). IBM also had difficulty controlling the PC market because "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.

197. *See* Thomas A. 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 e-mail, which stated: "[T]o control the APIs is to control the industry"); *see also* JERRY KAPLAN, STARTUP 49-50 (1995) (stating "'our value is the APIs'" and "[t]he real wars [in the computer industry] are over control of APIs" (quoting an industry remark)).

198. 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 application developers to its platform, it may provide independent software vendors 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 Department 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 197, at 888.

199. Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510 (9th Cir. 1992); Atari Games Corp. v. Nintendo of Am., Inc., 975 F.2d 832 (Fed. Cir. 1992).

200. *See generally* Michael L. Katz & Carl Shapiro, *Systems Competition and Network Effects*, J. ECON. PERSP., Spring 1994, at 93 (discussing network effects); Lemley & McGowan, *supra* note 47 (same). 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 that 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

"killer app" may suffice.[201] The more successful a proprietary platform becomes, the easier it is to attract software developers, and the easier it is to attract consumers; both factors reinforce the system's market dominance. At the same time, rivals may be forced out of the market and entry deterred. If the dominant firm has a proprietary interface, an entrant faces the difficulty of entering at two levels: platform development and software development. Apple Computer and Sega are among the platform developers that hoped to achieve substantial market penetration with noninteroperable systems.

But just as publishing interfaces can be risky, so can the strategy of keeping them closed. If application developers and consumers are not attracted to the system, losses can be considerable.[202] Even if initially successful, a noninteroperable system may lose out over time if other firms develop new systems to wrest away the incumbent's market share. Sega, for example, was a second comer to the game system market, entering after the Nintendo Entertainment System (NES) had achieved substantial market success.[203] Sega's Genesis system offered some features the NES lacked, as well as certain new programs (notably one featuring Sonic the 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.[204] The current market leader in the console game system market is Sony's PlayStation,[205] whose lead is about to be challenged by new entrant Microsoft's Xbox system.[206] In the game system market, platform

---

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 the 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 197, at 50.

201. *See* BAND & KATOH, *supra* note 163, at 30 (emphasizing the importance of Lotus 1-2-3 in contributing to the success of the IBM PC).

202. *See, e.g.*, Kelly Zito, *New Path for Sega: Company Decides Profit Lies in Video Games, Not the Consoles*, S.F. CHRON., Aug. 12, 2001, at E1. Sega recently exited the game-system market due to $420 million in losses in 2000 on the Dreamcast system it introduced in 1999. Sega's new system met with resistance from application developers who decided not to tailor games for it. *See* James Surowiecki, *Games People Play*, NEW YORKER, May 7, 2001, at 36, 36.

203. 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).

204. Zito, *supra* note 202 (reporting that Sega will now concentrate on the sale of games for other platforms because this is a more profitable line of business).

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

206. *See* Chris Gaither, *Microsoft Delays Release of Xbox Game System by a Week*, N.Y. TIMES, Sept. 22, 2001, at C15. Microsoft hoped to ship 1.5 million consoles by the end of the 2001 holiday season. *Id.*

developers typically lose money on sales of consoles, making up losses on sales of games and peripherals.[207]

In contrast to the game system market, which has been characterized by serial monopolies, Microsoft's operating system has become a de facto standard platform for applications running on personal computers, a monopoly that has been durable over many years.[208] Over this period, Microsoft's operating system interfaces have become more complex, and its licensing practices as to interfaces more restrictive. One explanation for the increasing complexity of Windows interfaces is that Microsoft has responded to some innovative applications by integrating them into the Windows operating system (a strategy sometimes known as "embrace and extend").[209] This has undermined the market of some competing applications, such as Netscape's Navigator browser, and threatened their viability. Microsoft has also responded to competition in the applications market by providing suites of popular applications (e.g., Microsoft Office) at attractive prices so that consumers will buy the suites instead of separate products from competing vendors. In addition, Microsoft has responded aggressively to innovations with potential to become alternative platforms to Windows, such as the Java programming system.[210] Even if much is disputed about Microsoft's conduct in preserving its operating system monopoly, no one would dispute that Microsoft's control over the APIs for

207. *Id.*; *see also* Surowiecki, *supra* note 202, at 36 (stating that "Sony loses money on every PlayStation 2 it makes"). Game consoles are expensive because of their many hardware components (semiconductor chips, graphics cards, memory, and the like). *Id.* Consumers are sufficiently sensitive to the cost of the consoles 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* William E. Cohen, *Competition and Foreclosure in the Context of Installed Base and Compatibility Effects*, 64 ANTITRUST L.J. 535 (1996).

208. United States v. Microsoft Corp., 253 F.3d 34, 54-58 (D.C. Cir. 2001) (finding that 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*, 46 ANTITRUST BULL. 1, 13-19 (2001) (discussing Microsoft's monopoly). It is worth pointing out that operating systems with open interfaces can be supplied by competing firms. *See, e.g.*, WAYNER, *supra* note 192, at 41-52 (discussing the successful struggle to open the Unix operating system).

209. The Department of Justice charged that Microsoft's decision to integrate its Internet Explorer browser into the Windows operating systems was intended to harm the market for Netscape's competing browser. *See Microsoft*, 253 F.3d at 84-97 (discussing the theory but remanding the case to the trial court for further findings); *see also* John Heilemann, *The Truth, the Whole Truth and Nothing but the Truth*, WIRED, Nov. 2000, http://www.wired.com/wired/archive/8.11/microsoft.html.

210. The World Wide Web opened up new opportunities for evolution of new platforms, such as browser software, for which applications could be written. *See* Fisher & Rubinfeld, *supra* note 208, at 20-23; Mark A. Lemley & David McGowan, *Could Java Change Everything? The Competitive Propriety of a Proprietary Standard*, 43 ANTITRUST BULL. 715 (1998).

developing applications for the Windows platform is an important source of its enduring power in this market.[211]

Into this strategic environment we now introduce reverse engineering. Platform developers typically copyright operating system programs, and they may also patent some components of their systems, but APIs are typically maintained as trade secrets.[212] 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),[213] trade secrets can be a very effective form of intellectual property protection for platform APIs.[214] If reverse engineering is both lawful and feasible, trade secrecy protection for platform APIs is at risk. Reverse engineering clearly threatens to upset a platform developer's noninteroperability strategy, whether unlicensed entry occurs at the applications level or at the platform level. From the standpoint of an unlicensed application developer, reverse engineering offers a means of achieving compatibility between its products and the large installed base of a successful system.[215] 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.[216] Reverse engineering gave Accolade an alternative way to access the Sega interfaces and enter the market with competing applications.

211. *See* Piraino, *supra* note 197, at 888-89 (quoting a Microsoft manager on the importance of APIs); *see also Microsoft*, 253 F.3d at 55-56 (discussing the applications barrier to entry that protects a dominant operating system irrespective of quality).

212. Courts have held that copyright protection does not extend to interfaces of computer programs. *See, e.g.*, Computer Assocs. Int'l v. Altai, Inc., 982 F.2d 693 (2d Cir. 1992). Patents may sometimes protect aspects of program interfaces. *See* 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).

213. *See infra* Subsection VI.B.2.

214. Economists Joseph Farrell and Michael Katz assume that intellectual property law determines the extent of network externalities between rival networks. They do not distinguish platforms from applications, but argue that intellectual property rights in the interface increase the incentive for quality improvements in a system as a whole. 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 rights in the interface may be unnecessary if platforms and applications are themselves protected. With intellectual property rights in platforms and applications, intellectual property rights in the interfaces may serve no beneficial purpose and may only allow developers to leverage market power in a way that was unintended as a matter of intellectual property law.

215. The unlicensed entrant who reverse-engineers the APIs and then sells system components may benefit from 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 206.

216. Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1514 (9th Cir. 1992).

2. *Welfare Effects of Reverse Engineering To Achieve Interoperability*

Table 3 compares the principal economic effects of allowing or disallowing reverse engineering to achieve interoperability in the software industry. Although we use criteria similar to those for the traditional manufacturing and semiconductor chip industries,[217] the welfare effects of reverse engineering rules in the software industry are more complicated and ambiguous. We explain the reasons for this below.

TABLE 3. SOCIAL CALCULUS OF REVERSE ENGINEERING OF SOFTWARE FOR PURPOSES OF INTEROPERABILITY

Social Welfare Criterion	Reverse Engineering Legal	Reverse Engineering Illegal
Incentives to develop platforms	Worse (Adequate?)	Better (Too high?)
Incentives to develop applications	Good (Better?)	Good
System prices • Short-run • Long-run	 Ambiguous Lower	 Ambiguous Higher
Wasted costs	Better?	Worse?

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 third parties can legally reverse-engineer program interfaces, this erodes the market power of a noninteroperable platform

217. The price and wasted costs criteria are identical to the earlier charts, although price is now a more complicated phenomenon because we must consider the effects of pricing of both the platform and applications. Incentives to innovate must, however, be split into two components, one focusing on incentives to develop platforms and one focusing on incentives to develop applications. Because platforms are typically developed before applications, applications are an important category of follow-on innovation. We could have broken down incentives for follow-on innovation further into incentives to improve platforms and incentives to improve applications, but this would needlessly complicate the main points we seek to make in this Subsection about systems competition issues.

developer.[218] In this respect, reverse engineering has the same effect in the software industry as in traditional manufacturing industries: It erodes market power by facilitating unlicensed entry or by inducing licensing on terms more favorable to the licensee than if reverse engineering were prohibited.[219] Of course, this does not necessarily mean that reverse engineering should be made illegal in order to protect platform developers. That would depend on the cost and time required for reverse engineering. Because decompilation and disassembly are time-consuming and resource-intensive, these forms of reverse engineering do not, we believe, significantly undermine incentives to invest in platforms.[220]

As for applications, there are strong incentives to develop them whether interfaces are open or closed. If interfaces can lawfully be reverse-engineered and hence are potentially open, any software developer will be able to develop applications for the platform, not just the developers licensed by the platform developer. Accolade, for example, adapted its Mike Ditka football game program to run on the Sega Genesis system, increasing the number of applications available for that platform. As this example shows, open interfaces facilitate not only third-party development of applications, but also the adaptation of applications to multiple platforms, which saves software development costs.[221]

There are also strong incentives, however, to develop applications when interfaces are proprietary and cannot be reverse-engineered. The developer of a noninteroperable platform wants a large installed base of customers. It can attract customers by providing a large number of attractive applications,

218. Graham and Zerbe emphasize this factor in their economic analysis of reverse engineering in the software industry. Graham & Zerbe, *supra* note 163, at 122.

219. *See supra* Section II.B.

220. 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 *Sony Computer Entertainment, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000), the platform developer was actually losing money on the sale of each platform. *See supra* notes 173, 207 and accompanying text. One might have expected Sony to welcome new entrants to expand its installed base without causing the firm additional losses, but this was not Sony's response. Sony complained of reputational damage to its system because PlayStation games operated less well on the emulated platforms. *See Connectix*, 203 F.3d at 608-09.

221. Church and Gandal address the question of software development under open and closed interfaces, although they do not assume that independent software vendors write software for all platforms simultaneously, even when interfaces are open. Jeffrey Church & Neil Gandal, *Integration, Complementary Products, and Variety*, 1 J. ECON. & MGMT. STRATEGY 651 (1992). There is no opportunity in their model to avoid software development costs by making each application compatible with all platforms, even when interfaces are open. Church and Gandal argue that despite the social benefits of open interfaces, firms have an incentive to choose proprietary interfaces. It seems that changing the model such that software vendors can write for all platforms simultaneously under a system of open interfaces would reinforce the conclusion that the firms' incentives to "go proprietary" are detrimental not only to the firms, but also to consumers.

especially those that are exclusive to that platform.[222] Independent software developers may easily be drawn to developing applications if they think the platform will emerge as the dominant one. If the platform is struggling to gain a toehold, its creator may have an even larger incentive to develop applications, perhaps doing so in-house or subsidizing independent developers who might otherwise be reluctant.[223]

Incentives to develop platforms and applications are naturally tied up with equilibrium prices. Two key market ingredients that affect pricing are, first, whether systems are compatible or incompatible, and, second, whether platform owners supply their own applications. We refer to the latter as "integrated" systems, in contrast to "unintegrated" systems in which independent firms supply applications for separately owned platforms. We think the most natural stylization of the pricing problem is that closed interfaces lead to incompatible and integrated systems,[224] while open interfaces lead to compatible and unintegrated systems.

We have not found an economic model with which to compare prices or incentives to develop platforms and applications in these two market structures. The economics literature has mainly compared two types of integrated ownership, namely, ownership with interoperable applications

222. 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 the design elements of a proprietary application, this may erode the market advantage the platform owner had hoped to garner through its investment. This helps to explain the "look and feel" lawsuits of the late 1980s and early 1990s. *See, e.g.*, Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807 (1st Cir. 1995) (rejecting Lotus's claim that the emulation interface of Borland's Quattro Pro spreadsheet program infringed Lotus 1-2-3), *aff'd by an equally divided Court*, 516 U.S. 233 (1996); 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 graphical user interface (GUI) infringed Apple's copyright in the Macintosh GUI); *see also* Data E. USA, Inc. v. Epyx, Inc., 862 F.2d 204 (9th Cir. 1988) (finding no infringement where the similarities between two independently developed karate programs lay in standard features to be expected of such games).

223. The platform developer's ability to attract developers to develop applications for the platform and to recoup subsidies incurred to attract application developers may be negatively affected to some degree by a rule favoring reverse engineering. If reverse engineering is lawful, licensed developers may worry about 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 application developers 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. Second, over time, licensed independent software vendors 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 application developer has had a hit in the applications market for a noninteroperable system, it may be able to negotiate more favorable terms, such as a right to develop its applications for more than one platform. *See* Surowiecki, *supra* note 202.

224. Platform owners with closed interfaces may contract with independent software vendors for development of applications, but we assume that they do so on license terms that capture much of the value for the platform owner and under terms of exclusivity. Thus, prices should not depend very much on whether platform owners contract with independent software vendors or develop their applications in-house. For our purposes, the key distinction is whether systems are integrated and incompatible, or unintegrated and compatible.

and ownership with noninteroperable applications. That literature yields inconclusive results.[225] In any case, it seems that integrated ownership of compatible systems would likely be unstable. With open interfaces, achieved by reverse engineering or otherwise, independent application developers will enter with compatible applications, and platform providers will enter with compatible platforms. Both undermine the integrated market structure.

It is difficult to compare prices between the two market structures. In an integrated system, platforms and applications may be sold as a unit, but they may also be sold separately with cross-subsidies between system components. In an unintegrated system, platforms and applications are priced and sold separately, at prices that are governed by the degree of competition in both markets, and possibly by intellectual property law. Our entries in Table 3 are inconclusive about pricing, but they indicate that when reverse engineering is illegal, so that systems may be integrated and incompatible, prices may be higher in the long run than in the short run due to the threat of tipping.

"Tipping" means that a single interface succeeds in becoming the standard in the market, creating a monopoly. Such tipping may be detrimental to consumers, but it is beneficial to the winning platform owner. By buying up talented independent application developers, entering into exclusive licensing agreements with them, or simply attracting them

225. *E.g.*, Carmen Matutes & Pierre Regibeau, *"Mix and Match": Product Compatibility Without Network Externalities*, 19 RAND J. ECON. 221 (1988). These authors argue that with two firms and demand conditions such that each consumer uses only one application, system prices will be higher when the integrated systems are incompatible than when they are compatible. *Id.* The same result recurs in a different model, Joseph Farrell et al., *The Vertical Organization of Industry: Systems Competition Versus Component Competition*, 7 J. ECON. & MGMT. STRATEGY 143 (1998), but the latter also shows that with more than two firms, the result on prices can be reversed—systems prices can be higher when systems are incompatible. *Id.* Using a model with two systems, Church & Gandal, *supra* note 221, at 663, concludes that incompatibility leads to lower systems prices than an unintegrated system.

An intuitive reason for higher prices with compatible, but integrated, systems is that platform owners will compete less fiercely because a seller's loss in platform sales can be mitigated by increased sales of his application to purchasers of the other platform. A second intuitive reason follows the observations of Cournot, 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. AUGUSTIN COURNOT, RESEARCHES INTO THE MATHEMATICAL PRINCIPLES OF THE THEORY OF WEALTH 103 (Nathaniel T. Bacon trans., MacMillan Co. 1927) (1838). The total price offered by the integrated firm will also yield more profit than the (higher) joint price charged by separate firms. Nicholas Economides, *Quality Choice and Vertical Integration*, 17 INT'L J. INDUS. ORG. 903, 913 (1999). Some commentators have relied on Cournot's insights to argue that consumers would be better off if platform developers controlled the applications market through licensing of interfaces. *See, e.g.*, Lichtman, *supra* note 193, at 624. We question the applicability of Cournot's analysis to software system markets, as both sides of those markets are subject to competitive forces.

with its large installed base, a platform owner may create sufficient network externalities to drive out rivals and remain the sole platform provider.[226]

A right to reverse-engineer may neutralize this threat of tipping. If the interface becomes open through reverse engineering or otherwise, other firms can develop platforms to compete with the proprietary platform and thereby undermine the latter's monopoly pricing strategy. Insofar as this interface becomes a de facto standard, consumers will benefit because more applications will be available for the platform and application developers will be in a better position to negotiate with firms competing in the platform market for better access to interface information.

Minimizing wasted costs is the fourth social welfare criterion. It too yields somewhat mixed policy prescriptions. Duplicated or wasted costs may arise in the software industry from at least three activities: the act of reverse engineering itself (costs wasted by the reverse engineer); the process of devising ways (e.g., technical protection measures) to make interfaces difficult or impossible to reverse-engineer (costs wasted by the platform developer);[227] and the development of different applications for different interfaces rather than the same applications for all interfaces (costs wasted by application developers generally). 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 these disparate welfare effects into an unassailable view as to whether reverse engineering for interoperability purposes should be legal. On balance, we believe that consumers benefit from interoperability because it encourages the development of a larger variety of software applications from a wider array of software developers with fewer wasted application development costs. Incentives to develop platforms are generally adequate owing to the high costs and difficulties of reverse-engineering software. Furthermore, interoperability lessens the potential for tipping into monopoly. Reverse engineering to achieve

226. Several commentators have argued against intellectual property rights in interfaces on these grounds. *E.g.*, Jeffrey Church & Roger Ware, *Network Industries, Intellectual Property Rights and Competition Policy*, *in* COMPETITION POLICY AND INTELLECTUAL PROPERTY RIGHTS IN THE KNOWLEDGE-BASED ECONOMY 227 (Robert D. Anderson & Nancy T. Gallini eds., 1998); Lemley & McGowan, *supra* note 47, at 525. *But see* Farrell & Katz, *supra* note 214 (arguing that intellectual property in interfaces can give firms incentives to improve their platforms). We caution, however, that if platforms and applications are themselves protected by appropriate intellectual property rights, then providing rights to interfaces might only give platform owners a means to leverage their market power beyond that intended by Congress.

227. *See* Cohen, *supra* note 163, at 1094; *see also infra* Subsection VI.B.2 (discussing the policy implications of efforts to thwart reverse engineering by making one's product difficult to reverse-engineer).

interoperability may also lessen a monopoly platform provider's market power by providing application developers with an alternative means of entry if the monopolist's licensing terms are unacceptable.

C. *Reverse Engineering of Software and Contract Law*

Another strategy for prohibiting decompilation and other forms of reverse engineering of programs has been the use of contractual restrictions, often by licenses inserted in boxes of packaged software.[228] The enforceability of such restrictions has been a highly contentious legal issue both in the United States and abroad.[229] The case law in the United States is in conflict on the enforceability of anti-reverse-engineering clauses in software contracts.[230] Uncertainty in the case law might suggest the need for

---

228. Mark A. Lemley, *Beyond Preemption: The Law and Policy of Intellectual Property Licensing*, 87 CAL. L. REV. 111, 129 (1999).

229. The European Union has declared that antidecompilation clauses in software contracts are null and void. *See* European Software Directive, *supra* note 178, art. 9(1), 1991 O.J. (L 122) at 45. The principal reason the EU chose to make antidecompilation clauses unenforceable was to create incentives for firms to license interface information on a reasonable basis so that second comers would not resort to reverse engineering to get this information. *See* CZARNOTA & HART, *supra* note 178, at 76-80 (reproducing the Directive's official commentary). A few other countries, notably Australia, have followed suit. Jonathan Band, *Software Reverse Engineering Amendments in Singapore and Australia*, J. INTERNET L., Jan. 2000, at 17, 20, *available at* http://www.gcwf.com/articles/journal/jil_jan00_1.html.

230. Courts have sometimes rejected reverse engineering defenses in trade secrecy cases because this activity exceeded the scope of licensed uses of the software. *E.g.*, Technicon Data Sys. Corp. v. Curtis 1000, Inc., 224 U.S.P.Q. (BNA) 286 (Del. Ch. 1984) (holding that a consultant to a 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 this use had not been authorized by the hospital; stating further that even if the use had been authorized, 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) (holding that there was a triable issue of fact as to whether Pulsecom's use of a "snooper board" at a telephone company to get access to interface information about DSC's software resulted in a misappropriation of a trade secret in view of restrictions in the telephone company's license to use DSC's software). For a nonsoftware case in which an anti-reverse-engineering clause was enforced, see K&G Oil Tool & Serv. Co. v. G&G Fishing Tool Serv., 314 S.W.2d 782 (Tex. 1958).

In some 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. The leading 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 in a shrinkwrap license under Louisiana law against a firm that had reverse-engineered the copy-protection scheme. The court of appeals held:

> The 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 at least 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 Assocs., Nos. 96, 112, 142, 1998 WL 740798 (N.D. Cal. June 9, 1998) (holding that a state unfair business practice claim based on the reverse

a legislative resolution. Legislative approaches, however, have also been contentious, as shown by the controversy over the model law now known as the Uniform Computer Information Transactions Act (UCITA).[231]

UCITA aims to resolve the decades-long controversy about shrinkwrap and other mass-market licenses for software.[232] 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.[233] Endorsing freedom of contract as a core value,[234] UCITA generally presumes license terms to be enforceable unless unconscionable.[235] Yet, owing to lingering concerns about imbalance in UCITA,[236] this model law now provides that if "a term of a contract violates a fundamental public policy, the court may refuse to enforce the contract, enforce the remainder of the contract without the impermissible term, or so limit the application of the impermissible term so as to avoid any result contrary to public policy."[237] UCITA also recognizes that if federal law preempts one of its provisions, that provision is "unenforceable to the extent of the preemption."[238]

---

engineering of another firm's program in violation of a license agreement was preempted by copyright law).

Some courts have also ruled against enforcing shrinkwrap licenses as a matter of contract law, either as contracts of adhesion or as contracts lacking mutuality of consent, although the case law is mixed on this issue as well. *Compare* Step-Saver Data Sys. v. Wyse Tech., 939 F.2d 91 (3d Cir. 1991) (holding that a shrinkwrap license is not enforceable as a matter of contract law), *with* ProCD, Inc. v. Zeidenberg, 86 F.3d 1447 (7th Cir. 1996) (enforcing a shrinkwrap license restriction). *See also* BAND & KATOH, *supra* note 163, at 221-22 (arguing that software shrinkwrap license restrictions on reverse engineering ought to be unenforceable); L. RAY PATTERSON & STANLEY W. LINDBERG, THE NATURE OF COPYRIGHT—A LAW OF USER'S RIGHTS 220 (1991) (criticizing shrinkwrap licenses "as unilateral attempts to override public law").

231. UNIF. COMPUTER INFO. TRANSACTIONS ACT, 7 U.L.A. pt. II, at 9 (Supp. 2001), http://www.ucitaonline.com/ucita.html [hereinafter UCITA]. With some consumer protection modifications, UCITA was enacted and is in force in Maryland. Virginia also enacted it with a two year moratorium. For a status report on state enactments of UCITA, see Status of the UCITA in the States, *at* http://www.ucitaonline.com/slhpsus.html (last modified Apr. 6, 2001).

232. *See* 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 the history of model-law projects and issues).

233. UCITA §§ 112, 210-211, 7 U.L.A. pt. II, at 49-50, 72-75.

234. *E.g.*, Gomulkiewicz, *supra* note 232, at 904-08 (invoking freedom of contract as an important principle of this model law).

235. UCITA § 111, 7 U.L.A. pt. II, at 48. UCITA does limit licensor freedom to some degree, for example, as to choice-of-law clauses in consumer contracts. *Id.* § 109, 7 U.L.A. pt. II, at 45. To the extent UCITA might conflict with an applicable consumer protection law, the latter will govern. *Id.* § 105(c), 7 U.L.A. pt. II, at 38. Some commentators have pointed out that most consumer protection laws apply to sales of goods and not to licenses of goods, and hence section 105 may supply less protection to consumers than might be apparent. *See, e.g.*, Jean Braucher, The Uniform Computer Information Transactions Act (UCITA): Objections from the Consumer Perspective (Aug. 15, 2000) (unpublished memorandum, on file with authors).

236. *E.g.*, Charles R. McManis, *The Privatization (or "Shrink-Wrapping") of American Copyright Law*, 87 CAL. L. REV. 173, 187-90 (1999) (discussing concerns about imbalance in the model law vis-à-vis other public policies, and compromise provisions to rectify the imbalance).

237. UCITA § 105(b), 7 U.L.A. pt. II, at 37-38.

238. *Id.* § 105(a), 7 U.L.A. pt. II, at 37.

The implications of these UCITA provisions for anti-reverse-engineering clauses have been the subject of considerable debate.[239] Some commentators believe that anti-reverse-engineering clauses in mass-market licenses should be unenforceable on copyright preemption grounds.[240] Others have asserted that such clauses should be considered a misuse of intellectual property rights.[241] Still others have suggested enforcing such license terms in negotiated licenses, but not in nonnegotiated standard form contracts.[242] Another suggestion is to enforce them unless the firm imposing the license term has monopoly power.[243] A new doctrine of public interest unconscionability has also been proposed under which anti-reverse-engineering clauses in mass-market licenses would be unenforceable.[244]

Counterarguments abound as well.[245] Critics point out that copyright preemption of contract terms is rare.[246] Misuse of intellectual property rights is a doctrine of uncertain scope and application, and some have opined that it should extend no further than antitrust law would.[247] Because

239. *E.g.*, UCITA § 105 cmt. 1, 7 U.L.A. pt. II, at 38 (explaining how courts might use section 105 to balance competing interests as to reverse engineering); Lemley, *supra* note 228; 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).

240. *E.g.*, McManis, *supra* note 236; 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).

241. *E.g.*, Lemley, *supra* note 228, at 151-58. *But cf.* Marshall Leaffer, *Engineering Competitive Policy and Copyright Misuse*, 19 U. DAYTON L. REV. 1087, 1106-08 (1994) (expressing concern about interference with legitimate interests of trade secret owners if copyright misuse doctrine forbids anti-reverse-engineering clauses).

242. *E.g.*, David Nimmer et al., *The Metamorphosis of Contract into Expand*, 87 CAL. L. REV. 17, 68 (1999). The Official Comment to section 105 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. UCITA § 105 cmt. 3, 7 U.L.A. pt. II, at 40; McGowan, *supra* note 239, at 1195-98 (reviewing various iterations of the Official Comment).

243. Maureen A. O'Rourke, *Drawing the Boundary Between Copyright and Contract: Copyright Preemption of Software License Terms*, 45 DUKE L.J. 479, 551 (1995); *see also* McGowan, *supra* note 239, at 1175-77 (raising questions about the enforcement of such terms in concentrated markets).

244. 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).

245. *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).

246. *E.g.*, UCITA § 105 cmt. 2, 7 U.L.A. pt. II, at 39; Lemley, *supra* note 228, at 144-50 (discussing the 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.*, ProCD, Inc. v. Zeidenberg, 86 F.3d 1447 (7th Cir. 1996); Nat'l Car Rental Sys., Inc. v. Computer Assocs. Int'l, Inc., 991 F.2d 426 (8th Cir. 1993). *But see* Niva Elkin-Koren, *Copyright Policy and the Limits of Freedom of Contract*, 12 BERKELEY TECH. L.J. 93, 106-13 (1997) (criticizing the *ProCD* decision because of conflicts with copyright public policy); Nimmer et al., *supra* note 242, at 42-63 (criticizing the preemption analysis in *ProCD*).

247. *E.g.*, Lemley, *supra* note 228, at 152 & n.188.

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.[248] While antitrust and competition law may regulate anti-reverse-engineering clauses in an appropriate case or context, no such claim has yet been brought, let alone sustained.

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.[249] With respect to anti-reverse-engineering clauses in software licenses, 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 transaction 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.[250]

There is 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:

> 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 enhancing allocative efficiency) without infringing on copyright goals or methodology.[251]

As explained above, we believe that the welfare effects of reverse engineering in the software industry context are somewhat more complex than this. However, on balance, reverse engineering and interoperability are

248. McGowan, *supra* note 239, at 1204-14.

249. *E.g.*, Julie E. Cohen, Lochner *in Cyberspace: The New Economic Orthodoxy of "Rights Management,"* 97 MICH. L. REV. 462, 536-38, 547 (1998).

250. McGowan, *supra* note 239, at 1213-14 (citations omitted).

251. *Id.* at 1205-06 (citations omitted).

important because they likely promote development of a wider range of software from a broader array of developers than a market in which platform developers are 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.[252]

V. REVERSE ENGINEERING OF TECHNICALLY PROTECTED DIGITAL CONTENT

The market for copyrighted works seems to be in a transitional period. For many years, copyright industries have derived the bulk of their revenues from the sale of physical products, such as books and videocassettes, in the mass market. Advances in digital technology have opened up the possibility of a future in which a substantial portion of copyright industry revenues may come from mass-marketing of technically protected digital content.[253] Copyright industry groups persuaded Congress to provide legal reinforcements to these technical protections so that it would become illegal to circumvent technical measures used by copyright industries to protect their works and to develop or distribute circumvention technologies. The result was the Digital Millennium Copyright Act (DMCA) of 1998.[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. 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 restricts reverse engineering.

The DMCA's restrictions on reverse engineering represent an inversion of the rules that apply in other industrial contexts. Under the DMCA, reverse engineering of technical measures may be illegal except when authorized by a specific statutory or rulemaking exception.[255] Even when

252. We agree with other commentators that the argument for nonenforcement of anti-reverse-engineering clauses is strongest as to mass-market software and weakest as to negotiated agreements between sophisticated firms. *E.g.*, O'Rourke, *supra* note 243, at 482; Reichman & Franklin, *supra* note 244, at 940-41; *see also infra* Subsection VI.B.1.

253. For a discussion of technical protection measures for digital content generally, see COMPUTER SCI. & TELECOMM. BD., NAT'L RESEARCH COUNCIL, THE DIGITAL DILEMMA: INTELLECTUAL PROPERTY IN THE INFORMATION AGE 153-76 (2000).

254. Digital Millennium Copyright Act, Pub. L. No. 105-304, 112 Stat. 2860 (1998) (codified in scattered sections of 5, 17, 28, and 35 U.S.C.).

255. See *infra* notes 288-295 and accompanying text for a description of exceptions.

reverse engineering is 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.[257] The range of these restrictions is unprecedented in American law.

Section V.A provides an overview of a future market in technically protected digital works that copyright industries envision. Section V.B discusses the law pertaining to circumvention and circumvention tools in the pre-DMCA era. It goes on to consider the circumstances leading up to the DMCA and the complex architecture of the DMCA rules. Section V.C explores the economics of the DMCA rules. It explains why those rules are overbroad and how the rules might be reformed to be more economically sound.

A. *Emerging Markets in Technically Protected Works*

The idea of technically protecting digital forms of copyrighted works is not a wholly new one. In the 1980s some computer software developers used copy-protection technologies when mass-marketing their products. Two factors led to the abandonment of copy-protection measures for software: First, copy-protection measures were displeasing to major customers because they interfered with some legitimate uses of software products, such as making backup copies; and second, makers of some competing software decided to make their products available without copy-protection to give them a competitive advantage.[258] This strategy worked well enough that copy-protection schemes for mass-marketed software died out in the marketplace. In the early 1990s digital audio tape (DAT) machines were first sold into the consumer market with a built-in technical protection measure. The Audio Home Recording Act required that all consumer-grade DAT machines include a serial copy management system chip that allowed users to make individual personal use copies of DAT sound recordings, but ensured that perfect digital copies could not be made from those personal use copies.[259] DAT technologies met with little success in the marketplace.[260] However, cable and satellite television programming are examples of technically protected content that have met with commercial success.

256. These restrictions are discussed *infra* Subsection VI.A.5.

257. 17 U.S.C. § 1201(a)(2), (b)(1) (Supp. V 1999).

258. *See* Cohen, *supra* note 249, at 521 n.221 (giving Borland International as an example of a new entrant willing to sell unprotected software to acquire market share from the market leader, Lotus 1-2-3, which was selling copy-protected software).

259. 17 U.S.C. § 1002 (1994); *see* ROBERT A. GORMAN & JANE C. GINSBURG, COPYRIGHT 508-09 (6th ed. 2002) (explaining the serial copy management system required by the Audio Home Recording Act).

260. *See* Cohen, *supra* note 249, at 525-26.

Despite the mixed market results of technically protected content, interest in technical protection measures as a way of controlling access to and uses of digital forms of copyrighted works has grown considerably since the mid-1990s. The motion picture industry is the first copyright industry to mass-market technically protected copies of digital content successfully. DVD movies are protected by a technology known as the "Content Scrambling System" that uses an authentication protocol to enforce country or region coding embedded in discs and players, as well as an anticopying mechanism.[261] The motion picture industry has persuaded manufacturers of equipment to make players conforming to the Content Scrambling System so that the technical controls built into DVDs will be enforced.[262] The sound recording industry has been working on a Secure Digital Music Initiative (SDMI) to embed technical controls in digital sound recordings that would be read and enforced by players.[263] The publishing industry is hoping to develop secure e-books.[264] Some technically protected content is already being delivered to consumers without the distribution of copies, such as by the "streaming" of audio or video files over the Internet.[265] More elaborate plans to build a "celestial jukebox" through which consumers could order a wide range of technically protected digital content are also underway.[266] One scholar believes that a fundamental transition is underway: from owning copies to "experiencing works."[267]

Technical protection systems provide new opportunities for content owners to protect commercially distributed copyrighted works against

261. *See* Universal City Studios, Inc. v. Reimerdes, 111 F. Supp. 2d 294, 346 (S.D.N.Y. 2000), *aff'd sub nom.* Universal City Studios, Inc. v. Corley, 273 F.3d 429 (2d Cir. 2001).

262. The DVD Copy Control Association (DVD-CCA) licenses the Content Scrambling System, certain patent rights necessary to make DVD players, and other know-how to equipment manufacturers. *See* DVD Copy Control Ass'n v. McLaughlin, No. CV 786804, 2000 WL 48512 (Cal. Super. Ct. Jan. 21, 2000), *rev'd sub nom.* DVD-CCA v. Bunner, No. CV 786804 (Cal. App. Dep't Super. Ct. Nov. 1, 2001), http://www.eff.org/cases/DVDCCA_case/20011101_bunner_appellate_decision.pdf.

263. *See* SDMI Challenge FAQ, *at* http://www.cs.princeton.edu/sip/sdmi/faq.html (last visited Mar. 7, 2002).

264. *See* Charles Clark, *The Answer to the Machine Is in the Machine*, *in* THE FUTURE OF COPYRIGHT IN A DIGITAL ENVIRONMENT 139 (P. Bernt Hugenholtz ed., 1996) (discussing the interest among publishers of scientific, technical, and medical books in electronic copyright management systems); Mark Stefik, *Shifting the Possible: How Trusted Systems and Digital Property Rights Challenge Us To Rethink Digital Publishing*, 12 BERKELEY TECH. L.J. 137 (1997).

265. *See* RealNetworks, Inc. v. Streambox, Inc., No. C99-2070P, 2000 U.S. Dist. LEXIS 1889 (W.D. Wash. Jan. 18, 2000) (discussing streaming technology).

266. PAUL GOLDSTEIN, COPYRIGHT'S HIGHWAY: FROM GUTENBERG TO THE CELESTIAL JUKEBOX (1994).

267. JANE C. GINSBURG, FROM HAVING COPIES TO EXPERIENCING WORKS: THE DEVELOPMENT OF AN ACCESS RIGHT IN U.S. COPYRIGHT LAW (Public Law & Legal Theory Working Paper Group, Columbia Law Sch., Paper No. 8, 2000), http://papers.ssrn.com/paper.taf?abstract_id=222493.

unauthorized uses. They enable new business models, and importantly, they reduce the need to rely on the law of copyright to regulate uses of digital content in the hands of consumers.[268] Technical protection systems are not, in themselves, fail-safe measures. What technology can do, another technology can undo. Some hackers regard technical measures as a challenge to be surmounted.[269] Some computer scientists view them as suitable subjects for research.[270] Those intent on infringing copyrights may also be motivated to break technical protections that rightsholders use to protect their works.[271] Reverse engineering is a necessary step in the undoing of any technical protection measure.

B. *Circumstances Leading Up to the DMCA Rules*

Prior to enactment of the DMCA, circumvention of technical measures used to protect copyrighted works had received little attention from the law. One exception was a provision in the Audio Home Recording Act forbidding the manufacture of technologies whose primary purpose or effect was to circumvent the serial copy management system chip in DAT machines.[272] Also outlawed was the sale of so-called black boxes for decoding encrypted satellite cable television programming.[273] Only one copyright case had considered the legality of making and selling a program that "undid" another vendor's copy-protection system.[274] Vault made a copy-protection program, PROLOK, that it marketed to commercial software developers for use in protecting mass-market copies of their programs. Quaid reverse-engineered PROLOK to figure out how it worked and developed a program called RAMKEY that circumvented the PROLOK system. Vault sued Quaid for contributory copyright infringement, alleging that purchasers of RAMKEY would use it to infringe the copyrights of Vault's customers' programs and thus harm the market for

268. *See* COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 79-95.

269. *E.g.*, Universal City Studios, Inc. v. Reimerdes, 111 F. Supp. 2d 294, 320 (S.D.N.Y. 2000).

270. *See* Amy Harmon, *Group Says It Beat Music Security but Can't Reveal How*, N.Y. TIMES, Jan. 15, 2001, at C2.

271. *See WIPO Copyright Treaties Implementation Act and Online Copyright Liability Limitation Act: Hearing Before the Subcomm. on Courts & Intellectual Property of the Comm. on the Judiciary*, 105th Cong. 215-16 (1997) [hereinafter *Judiciary Hearings*] (statement of Gail Markels, General Counsel and Senior Vice President, Interactive Digital Services Software Association) (discussing circumvention technologies used to enable "piracy" of copyrighted works).

272. 17 U.S.C. § 1002(c) (1994).

273. 47 U.S.C. § 605(e)(4) (1994).

274. Vault Corp. v. Quaid Software Ltd., 847 F.2d 255 (5th Cir. 1988). The court did not regard copies made in the reverse engineering process to be infringing. *Id.* at 261, 270.

Vault's software.[275] The court ruled against Vault because Quaid's product had a substantial noninfringing use, namely, enabling users to make backup copies of programs, as copyright law authorized them to do.[276]

In 1995, as part of its National Information Infrastructure Initiative, the Clinton Administration proposed amending copyright law to outlaw circumvention technologies in its White Paper, *Intellectual Property and the National Information Infrastructure*.[277] The White Paper expressed concern that without anticircumvention legislation, copyright owners would not provide content for this infrastructure because their works would be too vulnerable to widespread infringement.[278] To give new assurances to copyright owners, it proposed a ban on making or distributing technologies whose primary purpose or effect was to circumvent technical protections for copyrighted works.[279] No longer would the existence of a substantial noninfringing use shield a technology from the control of copyright owners.

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.[280] The draft treaty's anticircumvention provision, modeled on the White Paper proposal, proved controversial once the conference began.[281] Diplomats eventually agreed upon a compromise provision directing member states to provide "adequate protection" and "effective remedies" against circumvention of technical protections,[282] leaving the details of implementation to national discretion.

In 1997, the Clinton Administration announced its support for anticircumvention rules that were more expansive than the original White Paper proposal.[283] Under this new legislation, it would be illegal to circumvent a technical measure used by copyright owners to protect access

275. *Id.* at 258. One of the interesting questions in *Vault* was whether the copy-protection firm had standing to complain about infringement of software protected by PROLOK in view of the fact that it was not the holder of copyrights in that software. The appellate court ruled that Vault did have standing because "RAMKEY destroys the commercial value of PROLOK diskettes." *Id.* at 263.

276. *Id.* at 263-67 (discussing 17 U.S.C. § 117(2), which permits creating archival copies).

277. BRUCE A. LEHMAN, WORKING GROUP ON INTELLECTUAL PROP. RIGHTS, INTELLECTUAL PROPERTY AND THE NATIONAL INFORMATION INFRASTRUCTURE 230-34 (1995).

278. *Id.* at 230.

279. *Id.*

280. Pamela Samuelson, *The U.S. Digital Agenda at WIPO*, 37 VA. J. INT'L L. 369, 411-13 (1997).

281. *Id.* at 413-15 (discussing the controversy over anticircumvention rules at the World Intellectual Property Organization conference).

282. World Intellectual Property Organization Copyright Treaty, *adopted* Dec. 20, 1996, art. 11, 36 I.L.M. 65, 71. A similar treaty pertaining to sound recordings was also adopted at the same diplomatic conference, and it has a nearly identical anticircumvention provision. *See* World Intellectual Property Organization Performances and Phonograms Treaty, *adopted* Dec. 20, 1996, art. 18, 36 I.L.M. 76, 86.

283. *See Judiciary Hearings*, *supra* note 271, at 35-43 (statement of Bruce A. Lehman, Assistant Secretary of Commerce and Commissioner of Patents and Trademarks).

to their works. This provision was widely criticized as too broad. In response to some of these concerns, Congress crafted several specific exceptions to the anticircumvention rules and authorized the Librarian of Congress to create other exemptions in periodic rulemakings.[284] Much of the contention was about the impact the anticircumvention rules would have on fair uses of copyrighted works.[285] 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."[286] Circumvention and tools used for circumvention were analogized to burglary and burglars' tools.[287] 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 understood the DMCA rules as anti-reverse-engineering rules, the legislative debate might have ended with a more balanced result.

The DMCA now permits circumvention for seven purposes: legitimate law enforcement and national security purposes,[288] achieving program-to-program interoperability,[289] engaging in "legitimate" encryption research,[290] testing the security of computer systems,[291] enabling nonprofit libraries, archives, and educational institutions to make purchasing

284. The evolution of this legislation is recounted in detail in Pamela Samuelson, *Intellectual Property and the Digital Economy: Why the Anti-Circumvention Regulations Need To Be Revised*, 14 BERKELEY TECH. L.J. 519 (1999). *See also* JESSICA LITMAN, DIGITAL COPYRIGHT 89-150 (2001) (discussing the DMCA legislative debate).

285. *E.g.*, *Judiciary Hearings*, *supra* note 271, at 240-44 (statement of Douglas Bennett, President, Earlham College, and Vice President, American Council of Learned Society, on behalf of the Digital Future Coalition).

286. *Id.* at 208 (statement of Allan R. Adler, Vice President for Legal and Governmental Affairs, Association of American Publishers).

287. *E.g.*, STAFF OF HOUSE COMM. ON THE JUDICIARY, 105TH CONG., SECTION-BY-SECTION ANALYSIS OF H.R. 2281 AS PASSED BY THE UNITED STATES HOUSE OF REPRESENTATIVES ON AUG. 4, 1998, at 5 (Comm. Print 1998) (characterizing circumvention tools as "the digital equivalent of burglars' tools").

288. 17 U.S.C. § 1201(e) (Supp. V 1999).

289. *Id.* § 1201(f). The reverse engineering exception adopts the core holding of *Sega v. Accolade* in legitimating reverse engineering when necessary to achieve 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.

290. *Id.* § 1201(g). Conditions that substantially limit the application of this exception are discussed *infra* Section V.C.

291. 17 U.S.C. § 1201(j). This is subject to many conditions that substantially limit its application.

decisions,[292] allowing parents to control their children's use of the Internet,[293] and protecting personal privacy.[294] Since then, the Librarian of Congress has decided that circumventing access controls should be lawful in two other circumstances: when an access control system is broken and the circumventor has a right to access the material, and when circumvention is necessary to assess the effectiveness of a software filtering program to determine which sites it blocks.[295] Neither expressly authorizes the making of a tool to accomplish such privileged circumventions, and indeed it is unclear whether the Librarian of Congress has the authority to do so.[296] Four of the seven statutory exceptions to the act-of-circumvention rule lack express authorization to make tools to accomplish circumventions.[297] 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.[298]

The DMCA anticircumvention rules respond to copyright industry fears of uncontrolled infringement of 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.[299] As cryptographer Bruce Schneier has observed, "Digital files cannot be made uncopyable, any more than water can be made not wet."[300] Although digital content can be scrambled, every

292. *Id.* § 1201(d). This exception is of very limited utility to nonprofit libraries, archives, and educational institutions. It applies only to circumvention for purposes of deciding whether to purchase the technically protected content and not for such purposes as backup copying, preserving information, or making fair uses.

293. *Id.* § 1201(h).

294. *Id.* § 1201(i). This provision only applies if the user did not receive advance notice that the technical protection system would be collecting personal data. *Id.* § 1201(i)(1)(B). For a discussion of the implications of digital rights-management technologies for 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 an intrusive use of technical protection measures not covered by this exception, see Samuelson, *supra* note 284, at 552-54.

295. 37 C.F.R. § 201.40(b) (2001).

296. The Librarian's rulemaking authority seems to be limited under 17 U.S.C. § 1201(a)(1)(C) to developing exceptions to the act of circumvention rule of § 1201(a)(1)(A). Yochai Benkler argues that the DMCA anticircumvention 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 Enclosure of the Public Domain*, 74 N.Y.U. L. REV. 354, 427-29 (1999). Many scholars question the constitutionality of the DMCA anticircumvention rules. *E.g.*, Brief of Amici Curiae of Intellectual Property Law Professors, Universal City Studios, Inc. v. Corley, 273 F.3d 429 (2d Cir. 2001) (No. 00-9185), http://www.eff.org/IP/DMCA/MPAA_DVD_cases/20010126_ny_lawprofs_amicus.html.

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

298. *See id.* at 547; *see also* COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 175 (noting an ambiguity in the DMCA as to whether there is an implied right to make a tool to engage in privileged circumventions).

299. *See* COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 28-45.

300. Bruce Schneier, *The Futility of Digital Copy Prevention*, CRYPTO-GRAM, May 15, 2001, *at* http://www.counterpane.com/crypto-gram-0105.html#3. Schneier is the Chief Technology Officer of Counterpane Internet Security, Inc., designer of the popular Blowfish encryption

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."[301] Schneier says that almost any protection system will work against the average user, but *no* protection system will work against the power user, hacker, or professional pirate.[302]

The view articulated by Schneier may or may not be overstated, but we take it at face value as it provides the strongest argument for anticircumvention rules. Even so, we argue that the DMCA rules are more restrictive than is necessary to achieve the objectives Congress had in mind when it adopted them.[303]

C. *An Economic Analysis of the DMCA Rules*

Broadly speaking, the anticircumvention rules have consequences for protection of, access to, and uses of digital content, and competition in creating and marketing technical protection systems. Protection of digital works was, of course, the principal motivation for the DMCA anticircumvention rules. We argue, however, that the anticircumvention rules go further than necessary to accomplish the goal of protecting digital content, causing collateral harm that could be avoided. In particular, the rules may unduly impinge on fair and other noninfringing uses of digital content, on competition within the content industry, on competition in the market for technical measures, and on encryption and computer security research.

From an economic standpoint, we believe that it would be desirable to maintain the DMCA's prohibition on public distribution of tools designed to circumvent technical protection measures that protect against copyright infringement. We recommend, however, exempting individual acts of circumvention and private tool-making incidental to such circumventions.[304]

---

system, and author of six books, including BRUCE SCHNEIER, SECRETS AND LIES: DIGITAL SECURITY IN A NETWORKED WORLD (2000).

301. Bruce Schneier, The Natural Laws of Digital Content, Presentation at the Institute of Mathematics and Its Applications in Minneapolis, Minn. (Feb. 12, 2001) (slides, on file with authors).

302. *Id.*

303. Schneier believes that the DMCA rules will, in the end, prove futile because the Internet is an inherently global communications medium. Even if the United States and some allies adopt similar anticircumvention rules, such rules "would never have the global coverage [they] need[] to be successful." Schneier, *supra* note 300. 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 the digital world instead of fighting them." *Id.*

304. Of course, this does not mean that an individual act of circumvention should exempt the circumventor from liability if it results in copyright infringement. Circumvention of access controls may also sometimes violate the Computer Fraud and Abuse Act, 18 U.S.C. § 1030

Hence, we propose narrowing the DMCA rules in accord with our economic arguments. This is consistent with the original White Paper proposal, which did not recommend legislation to outlaw acts of circumvention, but only to outlaw the manufacture and distribution of circumvention tools.[305] While our proposed anti-tool rule is narrower than the White Paper's proposal,[306] it nevertheless focuses on the same risk for copyright owners. As reflected in Table 4, the essence of our argument is that the narrower rule would achieve the intended benefits for copyright owners while reducing harms to fair uses and improving incentives to develop, improve, and use technical protection measures.

(1994). We reserve judgment on whether an even better rule would be tort liability for distribution of circumvention tools rather than criminal liability.

305. *See* LEHMAN, *supra* note 277, app. 1, at 6.

306. In particular, the White Paper would have outlawed technologies whose primary purpose or effect was circumvention. *Id.* Computer industry groups objected to the primary effect language in the White Paper's proposal because it put firms at risk if customers used products to circumvent, even if they were not designed to do so. A better rule is one that focuses on what the technology was designed to do, as our proposal does. Our proposal adds a qualification about technologies that pose a high risk of facilitating infringement, as we believe that the DMCA anti-tool rules have sometimes been invoked where there is no danger of copyright infringement. *See, e.g.*, Sony Computer Entm't Am., Inc. v. GameMasters, 87 F. Supp. 2d 976 (N.D. Cal. 1999) (enjoining as a DMCA violation the sale of a product complementary to a technically protected game).

TABLE 4. SOCIAL CALCULUS OF REVERSE ENGINEERING OF TECHNICALLY PROTECTED CONTENT

Social Welfare Criterion	Pre-DMCA	DMCA	Narrower Rule (i.e., Ban on Tool Distribution)
Incentives to develop content	Worse	Better	Good
Opportunity for fair use of content	Better	Worse	Good
Incentives to develop and improve technical protection measures	Good	Worse	Better
Price of content	Low	High	Moderate
Expenditures on technical protection measures for content providers	Worse	Better	Good
Wasted costs	Worse	Better	Good

1. *Protecting Copyrighted Works*

As is apparent from the legislative history, Congress's concern in enacting the DMCA was to protect copyrights in digital content. Without technical protections, digital content is vulnerable to uncontrolled copying. Technical protections generally do not prevent copying, but only make the digital content uninterpretable without authorized use of a key or detection of a watermark.[307] An alternative to authorized use of a key is unauthorized

307. For a discussion of technical protection measures that content owners are using or planning to use to protect their works, see, for example, COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 152-73; *id.* app. E, at 282-303; and Daniel J. Gervais, *Electronic Rights Management and Digital Identifier Systems*, 4 J. ELECTRONIC PUBLISHING (1999), *at* http://www.press.umich.edu/jep/04-03/gervais.html. As Professor Lessig points out, the computer

decryption or circumvention, which involves reverse engineering. Decryption and circumvention are costly and difficult, and this is a significant check on the threat to copyright owners.

Most users have neither the inclination nor the ability to circumvent a technical protection measure.[308] A potential infringer will only infringe rather than buy a legitimate copy if the cost of circumventing the technical measure is less than the price of the copy. Content providers will take account of the potential for circumvention in setting their prices. As compared to the DMCA, content providers have an incentive to moderate their prices under the narrower rule and also to employ effective technical measures.

The DMCA gives no incentive for the content providers to moderate their prices, and it gives little incentive to employ effective technical measures. The DMCA allows criminal penalties in cases of individual acts of willful circumvention and infringement.[309] A circumventor would seem to be in jeopardy of criminal penalties even if the circumvention is trivial. Fear of such penalties is more likely than technical measures to deter infringement. Under the DMCA, any trivial technical measure may suffice because circumventing a technical measure raises the specter of criminal prosecution. Thus, the stringent penalties under the DMCA for individual acts of circumvention could have the odd consequence of *reducing* reliance on technical protection measures, as compared to the situation before the DMCA was enacted and as compared to the narrower rule we propose. By reducing the market for effective technical measures, the DMCA also reduces the incentive to develop them and improve them, as we discuss below.

Table 4 reflects these arguments. The price of copyrighted content susceptible to technical protection is likely to be highest under the DMCA and lowest without any such legislation. The narrower anti-tool rule helps enforce copyrights, but the price of content under this narrower rule is constrained by the threat of circumvention and infringement in a way that can be modified by the copyright holder in his choice of technical measure.

---

code that serves as a rights-management technology is a kind of private governance system. *See* LAWRENCE LESSIG, CODE AND OTHER LAWS OF CYBERSPACE 135-38 (1999).

308. *See* James Raymond Davis, *On Self-Enforcing Contracts, the Right To Hack, and Willfully Ignorant Agents*, 13 BERKELEY TECH. L.J. 1145, 1147 (1998) (pointing out the high cost and difficulties of hacking technical protection measures); *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).

309. 17 U.S.C. § 1204 (Supp. V 1999). In addition to being willful, an act of circumvention must also have a commercial purpose or be done for private financial gain. *Id.* We note that all infringements that displace a purchase will involve commercial harm to the copyright holder. If "commercial purpose" or "financial gain" is interpreted by courts to exclude infringement for personal use, then criminal enforcement of the DMCA would be less worrisome.

Table 4 also shows that content providers' expenditures on technical measures will be higher under the narrower rule than under the DMCA and probably highest with no legislation at all. Under a pre-DMCA regime of no prohibitions on circumvention, costs will likely be wasted on a measures-and-countermeasures war. Anticircumvention rules may curb this war, but as explained above, the DMCA goes too far. It protects content owners without encouraging them to use really effective technical measures.[310] A narrower anti-tool rule could both curb the measures-and-countermeasures war and also encourage content providers to use effective technical measures for protection. There is, of course, a sense in which all expenditures on technical measures are "wasted," at least by comparison to an idealized world in which intellectual property is automatically respected. But in Table 4, we have separated "expenditures on technical measures" from "wasted costs." The latter reflect the cost of a measures-and-countermeasures war that can be avoided by appropriate circumvention rules.

Content providers would likely spend more on technical measures under the narrower rule than under the DMCA, but we do not view this as a reason to prefer the DMCA. As we have explained, the DMCA protects rightsholders by increasing the penalties for copyright infringement, not by encouraging the use of technical measures. We contend that if Congress wants to strengthen criminal penalties for copyright infringement, then it should do it straightforwardly, rather than through the back door of the DMCA. While the narrower rule we propose is likely to increase the sums that content providers spend on technical measures, it avoids unnecessary criminalization of copyright infringement.

Our proposal for a narrower rule still maintains that the *public distribution* of circumvention tools should be prohibited. Otherwise, a single reverse engineer could induce widespread infringement by distributing the tool.[311] As Bruce Schneier puts it, "[A]utomation allows

310. As Professor Peter Swire has observed, "After [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 . . . ." E-mail from Peter Swire, Visiting Professor of Law, George Washington Law School, to Pamela Samuelson, Professor of Law and Information Management, University of California at Berkeley (Sept. 14, 2001) (on file with authors).

311. The premise of this argument is that, although *creating* a circumvention tool is time-consuming and costly, the *use* of the tool is not. We notice, however, that notwithstanding the widespread availability of DeCSS, a program capable of bypassing the Content Scrambling System that protects DVD movies, sales of DVD movies remain very strong and the motion picture plaintiffs in *Universal City Studios, Inc. v. Reimerdes*, 111 F. Supp. 2d 294 (S.D.N.Y. 2000), were unable to identify a single act of infringement of their movies attributable to the use of DeCSS. *Id.* at 314. Thus, at least in the short run, and possibly in the long run, there may be impediments to *using* circumvention tools. Such impediments would also protect content owners and thus render the extensive DMCA prohibitions unnecessary.

attacks to flow backwards from the more skilled to the less skilled."[312] In our view, that is the real threat that undermines the efficacy of technical measures, and it should be kept in check. It is also worth noting that a rule against distribution would be easier to enforce than a rule against individual circumvention because distribution is easier to detect.[313]

2. *Casualties of the DMCA: Fair Use and Competition*

The main premise underlying the DMCA act-of-circumvention rule is that circumvention will overwhelmingly be undertaken for purposes of infringement. We dispute that premise. As the nine exceptions to this rule demonstrate, there are many reasons to circumvent technical protections that have nothing to do with copyright infringement. We note that three of the nine exceptions—those permitting reverse engineering to achieve interoperability among programs, encryption research, and computer security testing—are principally aimed at promoting follow-on innovation, either by permitting development of new products or by improving products that already exist. The other six recognize that reverse engineering of technically protected digital content, such as reverse analysis of filtering software to discern what sites it blocks and decryption incidental to law enforcement and national security activities, may be reasonable and do not undermine copyright protection.

There are, however, many other reasons for reverse analysis of technical protections that promote follow-on innovation.[314] These include: locating, assessing, and fixing bugs in software; analyzing software to understand how to add additional features; understanding the internal design of a technical protection measure for research purposes; understanding its internal design to develop a competing product; understanding its internal design in order to make a compatible product, such as an alternative nonsoftware platform; analyzing a technical measure to enable interoperability with data; and enabling critical commentary on a technically protected movie by taking fair use clips from it.[315]

---

312. Schneier, *supra* note 301.

313. Gilbert & Katz, *supra* note 308, at 982-83.

314. Most of these examples and those in the following paragraphs occurred to us as we discussed the DMCA anticircumvention rules; a few were suggested by others. Professor Samuelson had previously identified some in an earlier article on the DMCA. *See* Samuelson, *supra* note 284, at 537-46.

315. Even Judge Kaplan has admitted that the fair uses excluded by technical protections are "remarkably varied." *Reimerdes*, 111 F. Supp. 2d at 337-38 (giving examples). This 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 use of DVD movies even if most people could not. *Id.* at 337-38. 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 § 1201(a)(2). It seems unreasonable to require a fair user to buy two copies of a

There are also many reasons to reverse-engineer technical protection measures to enable other reasonable follow-on uses of technically protected digital content: analyzing technical measures used to hide infringing copies of copyrighted works, analyzing technical measures used to hide stolen trade secrets or other confidential information, analyzing a virus program wrapped in a technical measure, creating backup copies of software or data, restoring a rightful copy after the crash of one's hard drive, preserving information (e.g., evidence of some illegal activity), preventing surveillance of a licensee's business activities, preventing technical "self-help" measures from being wrongfully invoked, 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, bypassing controls that prevent users from fast-forwarding through a movie, and making other fair uses, such as excerpting clips from technically protected movies to demonstrate that a particular word (e.g., "redskins") has been used in a derogatory fashion.[316]

It is also worth pointing out that although circumvention of copy-control measures is not illegal under the DMCA, courts have, in essence, made it illegal by interpreting copy controls as "access controls," circumvention of which is banned under the DMCA. One court, for example, has declared that the Content Scrambling System used to protect movies on DVDs is an access control.[317] As a consequence, purchasers of DVD movies can only play them on devices licensed by the DVD Copy Control Association, even if they would prefer to watch them on a Linux player. DVD movies can only be played on a device with the same country code as the movie. Another court characterized a country-coding scheme embedded in a mass-marketed videogame as an access control, thereby making it illegal to use lawfully purchased games on players with different country codes.[318] Without taking a position on the legitimacy or economic

---

movie to make fair use instead of one 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 seem to violate § 1201(b) to make a tool to engage in fair use of analog versions of movies. However, the Second Circuit affirmed Judge Kaplan's rulings on Corley's fair use defenses in *Corley*, 273 F.3d 429.

316. Commentators differ in their views about the effects of the DMCA on fair uses. Some assert that the DMCA rules preclude fair uses. *See, e.g.*, David Nimmer, *A Riff on Fair Use in the Digital Millennium Copyright Act*, 148 U. PA. 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 284, 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.*, GINSBURG, *supra* note 267; Benkler, *supra* note 296; Neil Weinstock Netanel, *Locating Copyright Within the First Amendment Skein*, 54 STAN. L. REV. 1 (2001); *see also* LESSIG, *supra* note 307, at 132-38 (questioning whether the U.S. Constitution requires limitations on copyright, such as fair use).

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

318. Sony Computer Entm't Am., Inc. v. GameMasters, 87 F. Supp. 2d 976, 987 (N.D. Cal. 1999).

soundness of country-coding,[319] we wish to point out that these applications of the DMCA dramatically alter buyers' rights, and we think further debate on these issues is warranted.

Some of the restrictions on use imposed by the DMCA, and overcome by the narrower rule, are not fair uses in the classical, copyright sense. All these uses, however, may lead to follow-on innovation. The DMCA inhibits many fair and other reasonable uses. Our proposed narrower anti-tool rule gives some opportunity to make fair and reasonable uses that the copyright owner might want to prevent. It is therefore more likely to support follow-on innovations and reasonable uses, as reflected in Table 4.

A narrower anti-tool rule might also prevent anticompetitive uses of the DMCA by content providers. In the past three years, plaintiffs have asserted violations of the DMCA rules in order to exclude competitors from the marketplace,[320] to control the market for complementary products,[321] and to facilitate their preferred market allocation and pricing strategies.[322] One

319. There is a substantial international debate about whether the sale of intellectual property products in one nation should "exhaust" the rightsholders' exclusive distribution rights throughout the world or whether rights should only be exhausted in the nation or region in which they were sold. Country codes embedded in software, games, or DVDs are designed to enforce national or regional exhaustion preferences of the rightsholder. The economics of international versus national or regional exhaustion are complex and as yet unresolved. For a discussion of the issues, see, for example, 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).

320. *See, e.g.*, *GameMasters*, 87 F. Supp. 2d at 982 (upholding a § 1201 claim against Game Enhancer software that competed with Sony's Game Shark software). Sony also asserted anticircumvention claims against Connectix, Inc. and Bleem, Inc., because both firms make emulator programs that did not read the anticopying technology in Sony games. These emulator programs compete with PlayStation in the platform market. *See* Samuelson, *supra* note 284, at 556-57 (discussing Sony's anticircumvention claim against Connectix); *see also Hearing on Exemption to Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies*, U.S. Copyright Office, Docket No. Rm9907, at 221, 224-32 (May 19, 2000), http://www.loc.gov/copyright/1201/hearings/1201-519.rtf (statement of Jonathan Hangartner, Attorney, Bleem, Inc.) (discussing Sony's anticircumvention claim against Bleem and the implications of § 1201(a)(1)(A) going into effect for future Sony anticircumvention claims against Bleem).

321. *See, e.g.*, RealNetworks, Inc. v. Streambox, Inc., No. C99-2070P, 2000 U.S. Dist. LEXIS 1889 (W.D. Wash. Jan. 18, 2000) (holding that Streambox VCR software, designed to interoperate with RealNetworks software, violated the anticircumvention rules); *GameMasters*, 87 F. Supp. 2d at 987-88 (holding that Game Enhancer software that interoperated with Sony PlayStation games violated § 1201 because it bypassed Sony's country coding); *see also Reimerdes*, 111 F. Supp. 2d at 320 (giving no weight to the claim that DeCSS was intended to enable development of a Linux platform for playing DVD movies).

322. Michael Owen-Brown, *Regulator Challenges DVD Zones*, AUSTL. TIMES, May 24, 2001, http://www.news.com.au/common/story_page/0,4057,2032464%255E421,00.html (stating that Australian competition and consumer protection authorities were investigating DVD country coding because of market allocation and discriminatory pricing impacts); *see also European Union Probes DVD Pricing*, SAN JOSE MERCURY NEWS, June 12, 2001, at 3C (describing EU competition concerns about country coding); *Video Store Raided for Selling Imported DVDs*, 2600 HACKER Q., June 28, 2001, *at* http://www.2600.com/news/display.shtml?id=541 (reporting on a raid of video stores in Gothenburg and Stockholm that were selling imported DVDs with the "wrong" country code).

commentator, disturbed by this trend, recommends development of a concept of misuse of DMCA rights akin to the misuse doctrines of patent and copyright law to thwart competitively harmful activities.[323]

Joint ownership of a proprietary technical protection system by major content providers may conceivably allow them to leverage their market power as to content into the market for equipment. For example, the motion picture industry controls the DVD player industry by its joint ownership of patent rights necessary to make DVD players; one of the conditions of this license is installation of the Content Scrambling System.[324] More recently, the recording industry has sought to leverage its market power over digital music into the market for players, through the Secure Digital Music Initiative (SDMI). The goal of the SDMI is to develop standard digital watermarks for digital music. The watermark must be detected by software in the player before the music can be heard.[325] In both examples, players and content become a "system" much like the operating systems and applications software discussed in Part IV. In the digital entertainment systems, entry into the player market is foreclosed, in part because of the DMCA rules, which essentially make the interface proprietary.[326] In the absence of legislation mandating installation of technical controls,[327] the

323. Professor Dan Burk of the University of Minnesota Law School has a work in progress on misuse of DMCA anticircumvention rights. E-mail from Dan Burk, Julius E. Davis Professor of Law, University of Minnesota Law School, to Pamela Samuelson, Professor of Law and Information Management, University of California at Berkeley (Sept. 19, 2001) (on file with authors).

324. *See Reimerdes*, 111 F. Supp. 2d at 310. As a close reading of *Reimerdes* reveals, any firm that wants to make a DVD player needs to get a license from DVD-CCA. *Id.* at 337 n.243. Although the court asserted that such licenses are "available to anyone on a royalty-free basis and at modest cost," *id.* at 337, such licenses, in fact, are only available "subject to strict security requirements," *id.* at 310. This precludes an open source Linux player. Any effort to develop an unlicensed platform would require reverse engineering of the Content Scrambling System (as well as a tool to do so). The motion picture industry would almost certainly claim that this is illegal under the DMCA.

Jointly established royalties also have the potential to facilitate price collusion, although there is no evidence that this has yet happened.

325. For a description of the SDMI watermarks and their intended uses, see SDMI Challenge FAQ, *supra* note 263.

326. In the software context, the market power constrained by reverse engineering lies in the platform provider because of its control over APIs. In the digital entertainment context, the market power is chiefly wielded by those who are rightsholders in the applications market. The economics of interoperability are 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 applies only to program-to-program interoperability. In one decision, a judge interpreted this provision as inapplicable because technically protected DVD movies are not "programs" but rather data. *See Reimerdes*, 82 F. Supp. 2d at 217-18. Because there are programs on DVDs as well as data, we question this ruling.

327. Senators Hollings and Stevens have announced their intent to introduce legislation to mandate installation of technical protections in future digital technologies. Declan McCullagh, *New Copyright Bill Heading to DC*, WIRED NEWS, Sept. 7, 2001, http://www.wired.com/news/politics/0,1283,46655,00.html (discussing the Security Systems Standards and Certification Act, which would mandate installation of standard technical protection measures in all interactive digital devices). Enactment of this legislation would foreclose the possibility of marketplace

market power that is implicitly facilitated may be the only way to ensure that highly protected products will enjoy success in the marketplace.[328]

3. *Competition in the Market for Technical Protection Measures*

The incentive to develop and improve technical measures depends on the market for them, which in turn depends on whether copyright owners need them to enforce copyrights. We argue that our proposed narrower anticircumvention rule will increase the demand for effective technical measures, which will increase the incentive to develop and improve them.

The super-strong protection of the DMCA not only erodes incentives to use technical measures, it also erects barriers to entering the market to supply them. The DMCA creates an extremely strong form of trade-secret-like protection for technical protection measures, far beyond that provided by any other law. Ordinarily, an unpatented product such as a technical measure would be subject to reverse engineering and competition. As in the traditional manufacturing context, the vulnerability of unpatented products to reverse engineering limits market power in a competitively healthy way. The DMCA rules effectively insulate makers of technical protection measures from competitive reverse analysis. This result could be avoided by the narrower rule we propose.

The narrower anti-tool rule would also enhance the ability of researchers to learn from each other. The DMCA inhibits research and hence follow-on innovation in technical measures because it limits the ability of researchers to learn from their predecessors. A reverse engineer who discovers a problem with another firm's technical measure and offers suggestions about how to improve it is at risk of getting indicted on criminal DMCA charges, rather than being offered a commercial or academic opportunity to improve the product.

Reverse engineering lies at the very heart of encryption and computer security research:

---

competition between protected and unprotected devices (except perhaps as regards used computers and other digital technologies).

328. 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 use. *See, e.g.*, COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 87-88, 154; *see also* Anna Wilde Mathews, *Antipiracy Tools in CDs Can Interfere with Playback*, WALL ST. J., Nov. 29, 2001, at B1. As mentioned earlier, *supra* note 258 and accompanying text, marketplace competition among software developers led to the abandonment of copy-protection systems for 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 R. VARIAN, INFORMATION RULES 102 (1999).

> 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 each others' work, cryptographers can improve the technologies they work with, discard those that fail, and gain confidence in technologies that have withstood repeated testing.[329]

A recent report of the National Academy of Sciences observes that "[r]egulating circumvention must be done very carefully lest we hobble the very process that enables the development of effective protection technology."[330] The report identifies some key ambiguities in the DMCA's anticircumvention rules that put encryption and computer security researchers at risk.[331] These assertions apply as much to commercial research as to academic research.

In the academic arena, the chilling effects of the DMCA on encryption and computer 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 software,[332] and threats of litigation against Princeton computer scientist Edward Felten and his colleagues after they wrote a paper about flaws they discovered in digital watermarks that the recording industry planned to use to protect digital music.[333] Although the DMCA provides some room for encryption

329. Brief of Amici Curiae Dr. Steven Bellovin et al., Universal City Studios, Inc. v. Corley, 273 F.3d 429 (2d Cir. 2001) (No. 00-9185), http://eon.law.harvard.edu/openlaw/DVD/NY/appeal/000126-cryptographers-amicus.html [hereinafter Bellovin Amici Brief].

330. COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 173.

331. *See, e.g.*, *id.* app. G, at 318-20 (discussing ambiguities and other problems with the DMCA anticircumvention provisions).

332. *See* Robert Lemos, *FBI Nabs Russian Expert at Def Con*, ZDNET UK NEWS, July 18, 2001, *at* http://news.zdnet.co.uk/story/0,,t269-s2091458,00.html. Instead of fixing the flaw in its software, Adobe asked the Justice Department 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 United States at a conference. In December 2001, the Justice Department decided to drop prosecution of Sklyarov. Jennifer 8. Lee, *In Digital Copyright Case, Programmer Can Go Home*, N.Y. TIMES, Dec. 14, 2001, at C4.

333. In September 2000, the Secure Digital Music Initiative issued a public challenge inviting skilled technologists to defeat digital watermarking technologies that SDMI had selected as candidate standards for protecting digital music. *See* Press Release, Secure Digital Music Initiative, An Open Letter to the Digital Community (Sept. 6, 2000), http://diddl.firehead.org/censor/hacksdmi.org/letter.asp. 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 titled *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, *supra* note 263.

and computer security research, the exceptions for these activities are so narrowly drawn that neither seems to apply to Sklyarov or Felten.[334] Consider, for example, that the encryption research exception does not apply to Felten because his research focused on digital watermarks that do not use encryption.[335]

This and other restrictions have caused prominent cryptographers to characterize the encryption research and computer security exceptions as "so parsimonious as to be of little practical value" as well as being based on a "fundamentally mistaken conception of cryptographic science."[336] The encryption research exception only applies, for example, if the researcher is employed or has been trained as a cryptographer,[337] even though some

---

An executive from Verance, the developer of one of the candidate technologies, and the Recording Industry Association of America (RIAA) 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 was that the presentation of the paper constituted distribution of a circumvention tool in violation of § 1201(b)(1). *See* Letter from Matthew J. Oppenheim, Recording Industry Association of America, to Professor Edward Felten, Department of Computer Science, Princeton University (Apr. 9, 2001), http://cryptome.org/sdmi-attack.htm (asserting that presentation or publication of the researchers' paper would violate the DMCA).

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* Edward Felten, Statement at the Fourth International Information Hiding Workshop (Apr. 26, 2001), *at* http://cryptome.org/sdmi-attack.htm. Felten's decision was widely reported in the press. *See, e.g.*, Sam Costello, *SDMI Attempts To Quash Researcher's Findings*, LINUXWORLD.COM, Apr. 25, 2001, *at* http://www.linuxworld.com/english/crd_sdmi_521727.html; David P. Hamilton, *Professor Savors Being in Thick of Internet Rows*, WALL ST. J., June 14, 2001, at B1. The Electronic Frontier Foundation agreed to represent Felten and his coauthors in an affirmative challenge to the RIAA and SDMI that sought a judicial declaration that the paper did not violate the DMCA so that Felten could present the paper at a conference in August 2001. *See* Complaint, Felten v. RIAA, No. CV-01-2669 (D.N.J. filed June 6, 2001), http://www.eff.org/Legal/Cases/Felten_v_RIAA/20010606_eff_complaint.html. This complaint was recently dismissed because SDMI, RIAA, and Verance withdrew their objections to publication of the paper. *See* John Schwartz, *2 Copyright Cases Decided in Favor of Entertainment Industry*, N.Y. TIMES, Nov. 29, 2001, at C4.

334. 17 U.S.C. § 1201(g), (j) (Supp. V 1999). If the SDMI watermarks are not access controls, the computer security testing exception would be 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 claims under 17 U.S.C. § 1202, a provision that protects copyright management information from alteration or removal. Verance had claimed that Felten violated § 1202 as well as § 1201. 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. Also, Sklyarov did not get Adobe's permission before testing its e-book security, as § 1201(j) requires.

335. *Id.* § 1201(g).

336. Bellovin Amici Brief, *supra* note 329, pt. III. Problems with the overly narrow and ambiguous encryption and computer security exceptions to the DMCA are discussed in COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 174-75.

337. 17 U.S.C. § 1201(g)(3)(B).

brilliant breakthroughs have come from persons without such training.[338] The exception is also only available if the researchers have sought permission from affected rightsholders before trying to reverse-engineer an encryption technology.[339] Researchers must, moreover, prove the necessity of their acts.[340] And the exception may be unavailable if the researcher publishes his or her results on the Internet where they may be accessible to potential pirates.[341]

Encryption and computer security may be crippled if researchers are at risk of liability under the DMCA in the ordinary course of their research.[342] As we argued in the case of the SCPA, reverse engineering can facilitate competition for improvements. The right balance between facilitating improvements and protecting earlier innovators can be achieved by granting a kind of "leading breadth" to each innovation,[343] but not by prohibiting researchers from access to knowledge, as the DMCA does.

VI. REVERSE ENGINEERING AS A POLICY LEVER

All intellectual property rights regimes—utility patent, plant variety protection, copyright, and the 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 case law 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 to some degree.[344] A rationale for this difference lies in the disclosure obligations that patent law imposes on

338. *See* SIMON SINGH, THE CODE BOOK: THE SCIENCE OF SECRECY FROM ANCIENT EGYPT TO QUANTUM CRYPTOGRAPHY (1999) (discussing the history of cryptography, and, in particular, the contributions of Whitfield Diffie).

339. 17 U.S.C. § 1201(g)(2). The computer security exception requires that the researcher actually get, and not just ask for, permission to defeat the technical protection measure. *Id.* § 1201(j)(1).

340. *Id.* § 1201(g)(1), (2)(B).

341. *Id.* § 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. *Id.* § 1201(g)(3)(C).

342. *See, e.g.*, Bellovin Amici Brief, *supra* note 329; Andrew W. Appel & Edward W. Felten, *Technological Access Control Interferes with Noninfringing Scholarship*, 43 COMM. ACM 21 (2000); Pamela Samuelson, *Anti-Circumvention Rules: Threat to Science*, 293 SCIENCE 2028 (2001).

343. *See* O'Donoghue et al., *supra* note 141, at 4.

344. *See supra* notes 30-40 and accompanying text.

innovators that trade secret owners avoid. For the traditional subject matters of copyright law, namely, artistic and literary works, reverse engineering has not been an issue because viewers and readers do 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.[345]

The optimal setting for any given policy lever depends in part on how the other levers are deployed.[346] 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 programs.[347] If decompilation and disassembly were illegal, 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 has generally been 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, constrains market power, and induces licensing that enables innovators to recoup R&D costs.

Second, we have found it useful to distinguish between the act of reverse engineering, which is generally performed to obtain know-how about another's product, and what a reverse engineer does with the know-how thereby obtained (e.g., designing a competing or complementary product).[348] The act of reverse engineering rarely, if ever, has market-

345. *See supra* Section IV.A.

346. There is an extensive economics literature on the interdependence of intellectual property policy levers. Most saliently, economics scholars have addressed the interaction of length and breadth. *See, e.g.*, Nancy T. Gallini, *Patent Policy and Costly Imitation*, 23 RAND J. ECON. 52 (1992); Richard Gilbert & 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). See also Maurer & Scotchmer, *supra* note 57, for the static context; and O'Donoghue et al., *supra* note 141, for the cumulative context.

347. We agree with Graham and Zerbe on this point. *See* Graham & Zerbe, *supra* note 163, at 128-31.

348. The SCPA rules, for example, explicitly permit reverse engineering, but impose a burden on reverse engineers to invest in post-reverse-engineering design work. *See supra* Section III.C. The DMCA rules, in contrast, include restrictions on acts of reverse engineering of technical protection measures. *See supra* Part V. The anti-tool rules of the DMCA go further in regulating preparatory activities for reverse engineering, namely, the making of tools for use in reverse engineering.

destructive effects and has the benefit of transferring knowledge. Harmful effects are far more likely to result from post-reverse-engineering activities (e.g., making a competing product with know-how from an innovator's product). Because of this, it may be more sensible to regulate post-reverse-engineering activities than to regulate reverse engineering as such. This view is reinforced by difficulties of enforcement. Acts of reverse engineering typically take place in private and are more difficult to detect than post-reverse-engineering activities (such as introducing competing or complementary products to the market). They are, as a consequence, less susceptible to effective regulation. In the discussion below, we distinguish between regulatory strategies aimed at acts of reverse engineering and those aimed at 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: regulating a particular means of reverse engineering,[349] establishing a breadth requirement for subsequent products,[350] using purpose- and necessity-based requirements for judging the legitimacy of reverse engineering,[351] regulating reverse engineering tools,[352] and restricting publication of information discovered by a reverse engineer.[353]

We review these options in Section VI.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

349. *See supra* Section II.C.

350. *See supra* Section III.C.

351. *See supra* Section IV.A.

352. *See supra* Section V.B.

353. *See supra* note 178. 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* Samuelson et al., *supra* note 15, at 2341. Second, one decision would have allowed reverse engineering of a program for the purpose of achieving present compatibility with the other firm's software, but not for the purpose of achieving future compatibility. Atari Games Corp. v. Nintendo of Am., Inc., 975 F.2d 832, 843 (Fed. Cir. 1992). Third, the legality of the second comer's reverse engineering efforts has sometimes been undermined by a "taint" in the last stages of the reverse engineering process, as when a firm's lawyers lie to the U.S. Copyright Office in order to get a copy of another company'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 the engineers needed to make compatible games. This inequitable conduct affected the court's view on Atari's fair use defense. *Id.* at 842-43. Fourth, the European Software Directive seems to give weight to establishing a "paper trail" to show the legitimacy of reverse engineering of software. *See* CZARNOTA & HART, *supra* note 178, at 84.

reverse engineering has been regulated in the past and under what conditions restrictions on reverse engineering are justifiable.

In Section VI.B, we observe that 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: by requiring customers to agree not to reverse-engineer their products, or 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*

When a particular means of reverse engineering makes competitive copying too cheap, easy, or 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.C, are an example. Using a competitor's product as a "plug" to make a mold from which to make competing products permits competitive copying that is so cheap and fast that it undermines 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,[354] 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 used to make competing products.

Another controversial act of reverse engineering was decompilation and disassembly of computer programs, discussed in Part IV.[355] Some industry participants feared that reverse engineering would allow second comers to appropriate valuable internal design elements of programs. Decompilation and disassembly were eventually accepted as legal, in part because they

354. Bonito Boats, Inc. v. Thunder Craft Boats, Inc., 489 U.S. 141, 160 (1989).

355. The DMCA rule outlawing circumvention (that is, reverse engineering) of technical measures that control access to copyrighted works does not fall into this category because it does not focus on a particular means. Yet, it too is a direct regulation of acts of reverse engineering. Although this rule has some exceptions (e.g., to enable program-to-program interoperability), Part V explained why that ban is nevertheless too restrictive.

require so much time, money, and energy that the original developer is not significantly threatened. If reverse engineering actually occurs in the face of these costs, it may enable the development of interoperable products and erode the market power of industry leaders in a competitively healthy way.

Our advice to policymakers is this: Before banning a means of reverse engineering, require convincing evidence that this means has market-destructive consequences. Realize that existing market participants may want 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 parasitic activities. For example, it may be sensible to make the restriction prospective rather than retroactive, to require that innovations embody some minimal creativity, or to limit the duration of the ban.[356] 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.[357]

2. *A Breadth Requirement for Products of Reverse Engineering*

Another policy option is to establish a breadth requirement for products developed after reverse engineering.[358] 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 and to take time so that the earlier innovator is still protected. The Semiconductor Chip Protection Act was our principal example.[359] The SCPA permits intermediate copying of chip circuitry for purposes of study and analysis; it also permits reuse of some know-how discerned in the reverse engineering process. This is a useful boost to competitors designing integrated circuits. The SCPA, however, requires reverse engineers to design an "original" chip rather than simply

356. *See supra* notes 80-84 and accompanying text.

357. *See, e.g.*, Eisenberg, *supra* note 40, at 1074-78 (proposing compensation for research uses of patented research tools); Mueller, *supra* note 40, at 54-66 (same); Reichman, *supra* note 56, at 1776-91 (proposing liability rules for subpatentable innovation); Samuelson et al., *supra* note 15, at 2369-71 (proposing a liability rule for reuses of industrial compilations of applied know-how in software).

358. See *supra* note 142 for a discussion of how economists have treated breadth.

359. A similar rule exists in the Plant Variety Protection Act, 7 U.S.C. §§ 2541, 2544 (1994). Use of a protected variety to develop a new variety is noninfringing as long as the subsequent variety itself is eligible 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* Peter J. Goss, *Guiding the Hand That Feeds: Toward Socially Optimal Appropriability in Agricultural Biotechnology Innovation*, 84 CAL. L. REV. 1395 (1996).

making a clone or near-clone of the integrated circuit that was reverse-engineered.[360]

Since the SCPA rules allow 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 advances while still preserving enough market power so that the innovator recoups costs in markets where cloning of the innovator's product would be market-destructive.[361] Again, policymakers should be wary of undocumented claims that reverse engineering is per se destructive.[362] Establishing breadth requirements may be unnecessary to protect the lead time of innovators in many industries because the costliness and difficulties of reverse engineering and reimplementation may provide adequate protection.[363] The SCPA rules responded to specific perturbations in the semiconductor chip market that undermined lead time.

While most legal regimes do not link the legitimacy of reverse engineering with technical advance, the software copyright case law may do so implicitly. 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 intended to bring about.[364] 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.[365] 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

360. The SCPA's reverse engineering privilege may be instructive even if the SCPA itself is flawed or no longer necessary for the reasons discussed *supra* Section III.D.

361. 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.

362. *See supra* Section II.C.

363. In addition, lead time can be governed by breadth. The length of each innovator's dominance in the market is determined in part by how long it takes a rival to find a noninfringing improvement. O'Donoghue, Scotchmer, and Thisse refer to this lead time as the "effective patent life." O'Donoghue et al., *supra* note 141, at 2. It may be shorter than the statutory patent life, and in this way, the possibly excessive reward granted by the twenty-year term of a patent can be modified endogenously by breadth. *See id.* at 2-3.

364. *See supra* Section IV.A.

365. 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) (finding infringement based not on decompilation but on the copying of expressive aspects of internal program information).

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 way to deploy the reverse engineering policy lever is to judge its legitimacy based on its purpose or necessity.[366] As with regulation of particular means, this approach focuses on the act of reverse engineering itself. Purpose-based rules assume that reverse engineering is sometimes socially beneficial and sometimes harmful, and at a deeper level, that society will benefit from a reverse engineer's acquisition of some types of know-how embedded in commercially distributed products but not others. Necessity-based rules assume that societal resources should not be expended on reverse engineering if the information being sought is already available. 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 almost absolute.

Two examples of purpose- and necessity-based privileges from this study are *Sega v. Accolade* and its progeny, which permit reverse engineering of computer software for the purpose of achieving interoperability,[367] and the DMCA anticircumvention rules that permit reverse engineering of access controls for some purposes, including achieving interoperability.[368]

We have mixed reactions to purpose- and necessity-based criteria for regulating reverse engineering. Of course it is true that the economic effects of reverse engineering depend on the reverse engineer's purpose, and purpose-based reasoning is common in intellectual property law. A second comer's purpose often determines whether he or she qualifies for an exception to or limitation on intellectual property rights.[369] Copyright's fair use doctrine, for example, gives considerable weight to the purpose of a fair use claimant's activities.[370]

One positive consequence of purpose-based rules is to induce knowledge-sharing through licensing or voluntary disclosure. European

366. Judging the legitimacy of reverse engineering based on the individual's purpose in engaging in the activity or on the necessity of 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 and in the DMCA, we treat them together in this Subsection.

367. Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1522-23 (9th Cir. 1993).

368. 17 U.S.C. § 1201(d)-(i) (Supp. V 1999).

369. *See supra* notes 37, 40, 165 and accompanying text (discussing the experimental use defense in patent law, the research exception in PVPA, and the fair use defense in copyright law).

370. 17 U.S.C. § 107(1) (1994).

policymakers legalized decompilation of computer programs for purposes of achieving interoperability in order to make the threat of reverse engineering credible enough so that software developers would disclose interface information voluntarily or license it on reasonable terms.[371] This ensured that European software developers could enter software markets with interoperable products. European policymakers did not wish to encourage licensing of other program know-how, but rather sought to encourage second comers to do their own independent design work, and hence, they restricted the privilege to decompilation for interoperability.[372]

A downside of purpose-based regulations is that if reverse engineering is not averted by licensing, wasteful litigation may be the only way to determine the reverse engineer's purposes. 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 its operating system to benefit consumers) or bad purposes (e.g., trying to put Netscape out of business).[373] Moreover, reverse engineers may have multiple purposes, only some of which may be privileged by purpose-based rules.

We believe that purpose-based rules are better than necessity-based rules as a strategy for limiting reverse engineering. Necessity-based rules may be a trap for the unwary. For example, if a software developer offers to license interface information on terms a second comer deems unreasonable, reverse engineering may seem necessary to the second comer, but not to the prospective licensor. Similarly, a software developer may be willing to license a minimal amount of interface information, but not enough to make the program fully interoperable. Once again, whether reverse engineering is necessary is disputable. A further problem arises if the information is available in an obscure place unknown to the reverse engineer who, in ignorance, exposes himself to liability by going ahead with reverse engineering he believed to be necessary. Necessity-based rules would also seem to be largely unnecessary given that rational second comers would almost always prefer to avoid the expenses of reverse engineering if the desired information was available without it.

371. *See* CZARNOTA & HART, *supra* note 178, at 74, 76-80 (reprinting the Official Comment to Article 6 of the European Software Directive).

372. *Id.* at 79-80. The *Sega v. Accolade* decision, by contrast, regards decompilation as legitimate if done to access information that is unprotected by copyright law (e.g., algorithms or mathematical constants). 977 F.2d at 1526 (emphasizing the need to decompile to access unprotected aspects of programs). Purpose-based limits on reverse engineering may also protect developers against difficult-to-detect infringements or avoid wasteful expenditures on reverse engineering undertaken for harmful purposes (e.g., to develop virus programs).

373. *See* United States v. Microsoft Corp., 253 F.3d 34, 84-96 (D.C. Cir. 2001) (reviewing conflicting views on Microsoft's purposes in integrating Internet Explorer into the Windows operating system).

Finally, we observe that enumerating exceptions to a general prohibition has different consequences than enumerating exceptions to a general privilege. The DMCA has a general prohibition of reverse engineering of access controls that is subject to various purpose-based exceptions.[374] This approach implies that reverse engineering of access controls for every other purpose is illegal. Given that the principal objective of the DMCA rule is to prevent copyright infringements, a more straightforward approach would have been to establish a general privilege in favor of reverse engineering, but to disallow it for the purpose of enabling infringement of copyrighted work. Because the DMCA adopts the more restrictive approach, a host of reasonable circumventions must be presumed illegal.[375] Those who reverse-engineer for unenumerated but benign purposes can only hope that their activities will escape the notice of the copyright industries and federal prosecutors.

4. *Regulating Reverse Engineering Tools*

The DMCA anticircumvention 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 Part V, we think the DMCA's anti-tool rules are overbroad. We recognize, however, that the anti-tool rules cannot be judged by the same considerations used in other industrial contexts. Our general assumption about reverse engineering 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 its content from unauthorized access and use. Circumvention undermines this strategy. Since circumvention tools are essential to reverse-engineer 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 circumvention technologies. Without ready access to circumvention tools, both large- and small-scale infringements may be prevented. It is, moreover, easier to detect and police a public market in circumvention technologies than to control private acts

374. *See* 17 U.S.C. § 1201(a)(1)(A) (Supp. V 1999) (setting forth the general prohibition on circumventing technological measures that control access to a work); *id.* § 1201(d)-(k) (listing purpose-based exceptions).

375. *See supra* Section V.C.

of circumvention and copying.[376] Nevertheless, we have argued that the anti-tool rules of the DMCA are defective because they reach many activities that have little 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.[377] 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 is to allow reverse engineering but to forbid publication or other disclosures of information obtained thereby. For the most part, the law has not had to address this issue because reverse engineers have generally had little incentive to publish or otherwise disclose information they learn from reverse engineering. Reverse engineers have typically kept the resulting know-how secret for competitive advantage.

Publishing information learned through lawful reverse engineering has long been legal in the United States. In *Chicago Lock Co. v. Fanberg*,[378] for example, the Ninth Circuit overturned an injunction against publication of a book containing compilations of key codes for tubular locks whose manufacturer claimed the codes as trade secrets. Because the author of the book, himself a locksmith, had gathered the information from fellow locksmiths who had lawfully reverse-engineered the information in the course of helping their customers, there was no misappropriation of Chicago Lock's trade secrets.

In recent years, restrictions on publication and other disclosures of reverse-engineered information have begun to appear. In the early 1990s, for example, the European Union adopted a directive on the legal protection of computer software that forbade publication or licensing of information obtained in the course of lawful decompilation of programs to achieve interoperability.[379] In 1998, the U.S. Congress adopted the DMCA anticircumvention rules, which impose numerous restrictions on disclosure

376. *See* Gilbert & Katz, *supra* note 308, at 982-83.

377. *See supra* Section V.C.

378. 676 F.2d 400 (9th Cir. 1981).

379. European Software Directive, *supra* note 178, art. 6(2), 1991 O.J. (L 122) at 45. The EU rule essentially requires each firm that wants to reverse-engineer to bear 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 software developer licenses interface information to foreclose the decompilation effort. At least one commentator has opined that publishing reverse-engineered information about the internal design elements of computer software should be illegal. *See* Davidson, *supra* note 163, at 1074-75.

of information learned 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 cannot disclose information learned therefrom unless the sole purpose of the disclosure is to accomplish interoperability.[380] One judge has opined that a journalist's publication of such information would violate the DMCA, even if acquisition of the information was lawful under the interoperability exception.[381] The presentation of a scientific paper on flaws in digital watermarking technology has been challenged as a violation of the DMCA anticircumvention rules.[382] Although the DMCA's exceptions permit some dissemination of the results of legitimate encryption research,[383] it puts encryption researchers at risk if they publish their results on the Internet because courts might decide this facilitates infringement.[384]

When it comes to restrictions on publication, it may be that the economic considerations underlying the DMCA rules are in irreconcilable conflict with values embodied in the First Amendment.[385] Moreover, economic considerations themselves may be in conflict. Publication of circumvention information may have the same market-destructive potential as if its author trafficked in circumvention tools for the purpose of facilitating copyright infringement. This destructive potential, however, must be weighed against the right of free speech and against another economic purpose, that of furthering encryption and computer security research.[386]

B. *Policy Options when Innovators Try To Prevent Reverse Engineering*

The very reasons that reverse engineering is socially beneficial—for example, that it erodes a first comer's market power and promotes follow-on innovation—may be why some innovators desire to prevent reverse

380. 17 U.S.C. § 1201(f)(3) (Supp. V 1999).

381. Universal City Studios, Inc. v. Reimerdes, 111 F. Supp. 2d 294, 320 (S.D.N.Y. 2000). This aspect of the *Reimerdes* ruling is difficult to square with the Supreme Court's decision in *Bartnicki v. Vopper*, 532 U.S. 514 (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.

382. *See supra* note 333 and accompanying text; *see also* Julie E. Cohen, *Call It the Digital Millennium Censorship Act: Unfair Use*, NEW REPUBLIC ONLINE, May 23, 2000, *at* http://www.tnr.com/online/cohen052300.html (discussing Microsoft's claims that an online discussion of how to bypass click-through licenses violated the DMCA anticircumvention rules); John Schwartz, *Apple Offers More than an Update to Its System*, N.Y. TIMES, Dec. 1, 2001, at C14 (discussing Apple's claim that online posting of information enabling access to a software upgrade violated the DMCA rules).

383. 17 U.S.C. § 1201(g).

384. *See supra* Section V.C.

385. *See* Bellovin Amici Brief, *supra* note 329, pt. II.

386. *See supra* Section V.C.

engineering altogether or render it moot. When reverse engineering is lawful, firms may seek to thwart it in one of two ways: 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. This Section addresses the policy responses available to deal with attempts to circumvent legal rules permitting reverse engineering.

1. *Avoiding the Threat of Reverse Engineering by Contract*

Software licenses often prohibit reverse engineering, even when (or especially when) reverse engineering is allowed by law.[387] Whether such contracts should be enforceable as a general matter is an unsettled question of law, as Part IV has shown.

We believe that in markets for products heavily dependent on intellectual property rights, such as computer software, there is reason to worry about contractual restrictions of reverse engineering. Some market power is inevitable in such markets, or else the intellectual property right has no purpose. The policy levers that define the intellectual property right 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.[388] Hence, it may be reasonable not to enforce contract terms purporting to override reverse engineering privileges in intellectual-property-dependent markets such as software,[389] as the European Union has done by nullifying license terms forbidding decompilation of computer programs.[390]

387. A parallel policy problem is whether to enforce contractual overrides of fair use and first sale rights of copyright law. *See* COMPUTER SCI. & TELECOMM. BD., *supra* note 253, at 101-02; McManis, *supra* note 236, at 175-76, 184 n.42.

388. *See* Lemley & McGowan, *supra* note 47, at 523-27; *see also* United States v. Microsoft Corp., 253 F.3d 34, 63 (D.C. Cir. 2001) ("The company claims an absolute and unfettered right to use its intellectual property as it wishes. . . . That is no more correct than the proposition that use of one's personal property, such as a baseball bat, cannot give rise to tort liability.").

389. *See supra* Section IV.C. Of course, contracts that prohibit reverse engineering do not render the reverse engineering right entirely moot. If the product is available in the market from another source, a potential reverse engineer may have 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.

390. European Software Directive, *supra* note 178, art. 9(1), 1991 O.J. (L 122) at 45. The nullification extends only to decompilation for purposes of achieving interoperability.

2. *Avoiding the Threat of Reverse Engineering by Technical Obfuscation*

Firms sometimes design their products so that it will be difficult or impossible to reverse-engineer them.[391] Such expenditures would be unnecessary if reverse engineering were unlawful. In the economic calculus of reverse engineering, we must count expenditures to thwart reverse engineering as socially wasteful. Efforts to prevent reverse engineering may, however, be unsuccessful or only partially successful. Determined second comers may 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 do not 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 a reverse engineer finished his work, the next version of the product would be in the marketplace.

One policy option for dealing with such a situation is to force the innovator to disclose certain information about her product.[392] For example, if arguments in favor of open interfaces have merit and interfaces cannot be effectively discerned by reverse engineering, then it may sometimes make sense to require interfaces to be made public. This is essentially what happened some years ago in Europe when antitrust authorities brought suit against IBM for abuse of its dominant position because it had been altering the interfaces to its mainframe computers frequently, thereby disadvantaging European makers of peripheral products. The dispute was eventually resolved by IBM's agreement to announce changes to its interfaces in advance so that peripheral manufacturers could adjust their products accordingly.[393] Some have suggested a similar remedy in *United*

391. *See supra* note 49.

392. It is worth pointing out that in a variety of other circumstances, legal decisionmakers, in the interests of public policy, have forced firms to disclose information not readily discernible from examination of publicly distributed products. 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) (considering a challenge to requirements of the Federal Insecticide, Fungicide, and Rodenticide Act to submit safety test data to the EPA, 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 proposal of Professors Burk and Cohen 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. Dan L. Burk & Julie E. Cohen, *Fair Use Infrastructure for Rights Management Systems*, 15 HARV. J.L. & TECH. 41 (2001).

393. *See, e.g.*, BAND & KATOH, *supra* note 163, at 22 n.30. *But cf.* Berkey Photo, Inc. v. Eastman Kodak Co., 603 F.2d 263 (2d Cir. 1979) (finding that a monopoly firm had no duty under the antitrust laws to predisclose information about a new camera and film format to enable competitors in the film market to prepare compatible products).

States v. Microsoft.[394] 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 far more difficult than, say, reverse engineering interfaces of game platforms, and it may be impracticable. Forcing Microsoft to publish its APIs would certainly erode its market power, but this raises a host of other difficulties.[395]

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 the products of others. 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 means for information to percolate through a technical community than patenting or publication, but it is nonetheless an effective source of information.[396] Reverse engineering leads to dependent creations, but this does not taint them, for in truth, all innovators stand on the shoulders of both giants and midgets.[397] 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 in the past two decades most of the proposals to restrict reverse engineering have arisen in the context of information-based products, such as semiconductors and software. The high quantum of know-how that such products bear on or near their face makes

394. *See, e.g.*, Piraino, *supra* note 197, at 887-89; *see also* R. Craig Romaine & Steven C. Salop, *Slap Their Wrists? Tie Their Hands? Slice Them into Pieces? Alternative Remedies for Monopolization in the* Microsoft *Case*, ANTITRUST, Summer 1999, at 15, 18-19.

395. A key difficulty arises from the fact that program interfaces are not always self-evident or self-defining. *See* BAND & KATOH, *supra* note 163, at 6-7 (illustrating the difficulty of precisely defining "interface"); CZARNOTA & HART, *supra* note 178, at 37-38. Much judicial oversight might be necessary to enforce an obligation by Microsoft to disclose interface information. *See* Romaine & Salop, *supra* note 394, at 19.

396. As Dreyfuss and 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 50, at 818.

397. That is, progress happens through both breakthrough innovations and the accumulation of small steps. Economists have focused on designing standards of patentability and breadth in order to balance the incentives of earlier and later innovators. *See* Nancy T. Gallini & Suzanne Scotchmer, *Intellectual Property: When Is It the Best Incentive System?*, *in* 2 INNOVATION POLICY AND THE ECONOMY (Adam Jaffe et al. eds., forthcoming 2002) (synthesizing the economics literature on cumulative innovation); O'Donoghue et al., *supra* note 141 (focusing on incremental innovation); Scotchmer, *supra* note 104 (focusing on breakthrough inventions). Some legal scholars have proposed supplementary legal regimes to deal with subpatentable innovations. *See, e.g.*, Reichman, *supra* note 14, at 2444-45; Samuelson et al., *supra* note 15, at 2365.

these products more vulnerable than traditional manufactured goods to market-destructive appropriations.[398] 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 network 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 of the industrial contexts we 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, policymakers should 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 backward 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."[399] It is, however, a distinction that has been a foundational principle of intellectual property and unfair competition law, at least until enactment of the DMCA. It is, moreover, a distinction whose abandonment could have detrimental consequences for innovation and competition.

398. *See* Reichman, *supra* note 14, at 2443-44.
399. DREYFUSS & KWALL, *supra* note 50, at 818.

***