THE NEED FOR SOFTWARE INNOVATION POLICY

CHRISTOPHER RILEY*

This paper examines the current legal treatment of software innovation. It argues that recent judicial standards for the regulation of software innovation do not adequately protect innovation. It presents an original standard for the regulation of software innovation, one intended to guide judicial decisions in contributory copyright liability, in interpretations of the Digital Millennium Copyright Act, and in every courtroom where a developer is on trial for the mere creation and distribution of software. The standard presented in this paper separates the questions of liability and remedy in order to produce an optimal dynamic balance of interests.

* Ph.D., Computer Science, Johns Hopkins University, 2004; J.D. expected, Yale Law School, 2007. I would like to thank Professor Jack Balkin, James Grimmelmann, Eddan Katz, and the attendees of the Yale/Harvard Cyberscholar Working Group for their helpful comments.

I. INTRODUCTION 591
II. WHY MUST WE PROTECT SOFTWARE INNOVATION AND HOW IS IT AT RISK? 596
A. Why is Software Innovation Different from Other Forms of Innovation? 596
B. Why is Software Innovation Valuable? 598
C. What Will We Lose if We Do Not Protect Software Innovation Adequately? 599
D. The Legal Climate for Innovation 601
E. Innovations Under Attack 603
III. WHAT IS PROPER SOFTWARE INNOVATION POLICY? 610
A. Grokster, or: What is Improper Software Innovation Policy? 610
B. Separating Liability from Remedy; Separating the Technology from the Developer 612
C. The Difference Between a Liability/Remedy Test and Grokster 615
D. Real-World Applications of the Liability/Remedy Standard 619
IV. CRITICISMS AND ALTERNATIVES 623
A. Workability 623
B. Other Solutions 624
VI. CONCLUSION 627

I. INTRODUCTION

Peer-to-peer filesharing networks like Napster and Grokster are considered a blight on society by the media and copyright holders. They have enabled millions of people to acquire music for free, without paying any royalties or license fees. The users of these programs have broken the law; few would dispute that.[1] The Supreme Court and other courts have held that the producers of the network software also violated the law, under the doctrine of secondary liability for copyright infringement.[2] As a result, these software innovations have been restrained – the developers have stopped distributing their systems, or have converted them into industry-sanctioned subscription services.[3] And the industry continues to fight, to challenge the distribution and use of new generations of filesharing systems.[4]

Let us suppose for a minute that all of this could have been avoided, that before the very first peer-to-peer filesharing network had been released to the public, the copyright industry could have taken its developer into court. Determining that these programs could be used to exchange music files in violation of copyright law, and that this possibility was known to (perhaps even intended by) the developers, the court would have enjoined the distribution of the software, threatening the developers with damages should the systems be used to exchange copyrighted files without permission. The public would never have seen the network, and would not have realized that such forms of communication were possible. Without seeing first-hand the efficiency, portability, and audio quality of MP3-encoded music files, society might not have developed the necessary demand to make the (very expensive) portable MP3 player a market success. We would not have online music stores, such as iTunes, which were developed as legal alternatives to

---

1. At least one person has tried this argument in a court of law. BMG Music v. Gonzalez, 430 F.3d 888, 891 (7th Cir. 2005), *cert denied*, 126 S. Ct. 2032 (2006) (upholding a district court verdict that as a matter of law filesharing did not constitute fair use).

2. Secondary liability is a common law doctrine that penalizes the distributor of a device used by others to infringe copyright. It is often used when punishment of the direct infringers is not feasible. *See, e.g.*, MGM Studios Inc. v. Grokster, Ltd., 545 U.S. 913, 929-30 (2005).

3. The original filesharing service, Napster, has converted into a monthly subscription service, in which users can pay a flat monthly fee and stream music from Napster's catalog. Napster, http://www.napster.com/ (last visited Feb. 13, 2007).

4. One of the major companies offering BitTorrent files, LokiTorrent, attempted to collect donations to fund a legal team to fight the Motion Picture Association of America. *See* Robert Lemos, *LokiTorrent Fights MPAA Legal Attack*, CNET NEWS.COM, Dec. 30, 2004, http://news.com.com/2100-1025_3-5508073.html. They raised $40,000, but the site administrator still agreed to comply, paying a small legal fine and shutting down the service. *See* Michael Ingram, *LokiTorrent Caves to MPAA*, SLYCK NEWS, Feb. 10, 2005, http://www.slyck.com/news.php?story=661.

peer-to-peer networks. Eventually, of course, the major market players might have figured out the strong potential market of online music distribution. But the pace of innovation would have been greatly slowed without competition.

The potential harm to the computer science industry would have been even worse than the harm to the consumer market. Computer scientists and engineers took the ad-hoc, highly distributed model of peer-to-peer networking and adapted it in many ways, creating systems such as SETI@Home for distributed computation or IRIS for distributed storage.[5] Peer-to-peer systems have many technical advantages over traditional client-server systems, including: increased scalability (the capacity of the system to increase the number of participants with low overhead), fault tolerance (the ability of the system to continue functioning even if many individual participants fail), and flexibility (the ability of the system to adapt to serve multiple functions).[6] Again, perhaps these advantages would have been realized eventually. But peer-to-peer filesharing networks brought them to society more quickly, more widely, and at less cost.[7]

This is the story of one innovation, and of what would have been lost if the legal system had cut it off in its incipiency. Peer-to-peer networks and other technological innovations produce transformative effects on our society, both good and bad. Many everyday technologies were themselves once radical technological innovations, such as the

---

5. SETI@Home (SETI stands for "Search for ExtraTerrestrial Intelligence") uses volunteer contributions of idle computing cycles from home personal computers to analyze satellite data. *See* SETI@home, http://setiathome.ssl.berkeley.edu/ (last visited Feb. 11, 2007). The Infrastructure for Resilient Internet Systems (IRIS) project is a collaborative effort of academic computer scientists from five universities to build distributed systems based on Distributed Hash Tables, or DHTs, a structure fundamentally based on the peer-to-peer communications model. *See* IRIS: Infrastructure for Resilient Internet Systems, http://project-iris.net/index.html (last visited Feb. 11, 2007). Many IRIS papers were published at the annual International Workshop on Peer-to-Peer Systems (IPTPS). The first academic Distributed Hash Table, the Chord system of Stoica et al., was published in 2001. Ion Stoica et al., *Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications*, PROC. OF THE 2001 CONF. ON APPLICATIONS, TECHS., ARCHITECTURES, & PROTOCOLS FOR COMPUTER COMM. 149, *available at* http://pdos.csail.mit.edu/papers/chord:sigcomm01/chord_sigcomm.pdf. Contrast this with the Napster peer-to-peer filesharing network, which by 2000 had already reached a federal court. *See* A&M Records, Inc. v. Napster, Inc., 114 F. Supp. 2d 896 (N.D. Cal. 2000), *aff'd in part*, *rev'd in part*, 239 F.3d 1004 (9th Cir. 2001).

6. *See, e.g.*, Rodrigo Rodrigues, Barbara Liskov & Liuba Shrira, *The Design of a Robust Peer-to-Peer System*, 2002 PROC. OF THE 10TH ACM SIGOPS EUR. WORKSHOP: BEYOND THE PC 117, *available at* http://pdos.csail.mit.edu/papers/chord:sigcomm01/chord_sigcomm.pdf.

7. The distributed development of peer-to-peer networks by amateurs is of lower cost to society than academic research, which is funded largely through taxpayer money in the form of grants. Many scholars have praised the collaborative development environments through which these programs are created. *See, e.g.*, YOCHAI BENKLER, THE WEALTH OF NETWORKS: HOW SOCIAL PRODUCTION TRANSFORMS MARKETS AND FREEDOM (2006).

home VCR, with both good and bad effects. These fundamental innovations might have been permanently barred from the world if past courts had not been so open-minded.[8] Similarly, the amateur programmers and would-be inventors, like the graduate students who founded Google,[9] might never have built their world-changing systems if they feared multimillion dollar legal judgments against them based on unanticipated and undesired secondary uses of their products. No one can imagine what the world would look like today had these innovations and others been prohibited. Nor can anyone accurately imagine future technologies or how the courts will react to them, to know what lies on the horizon of the regulation of innovation. But hopefully I have created a suspicion that the risks of overly restricting technological innovation are great.

In the modern era, software innovations – innovations that take the form of new, original computer software programs, or new uses or combinations of existing computer programs – exaggerate the transformative effects of general technological innovations because of their potential for rapid, low-cost development and fast, widespread deployment. Innovation in the computing industry is not a story of patent law and the research and development divisions of multimillion dollar corporations. The real history of Silicon Valley is not a story of the modern-day IBMs and Microsofts, armed with advertising executives and teams of lawyers, but of garage inventors and students with great ideas who were given the freedom to pursue them without fear of legal reprisal.[10] These entrepreneurs operated under only the constraints of technology and the bounds of human imagination.[11] Their innovations broke new ground in the technology industry. Low barriers to entry and a tradition of commercial success engendered a world of small "startup" companies and of individual hobbyists and tinkers. These small

8. The U.S. Supreme Court was faced with this issue in the landmark 1984 case concerning the legal status of the Sony Betamax video recorder, and it chose to interpret existing secondary liability laws in copyright to protect the innovation against established legal interests. Sony Corp. of Am. v. Universal City Studios, Inc., 464 U.S. 417 (1984).

9. Google Corporate Information: Management, Larry Page, http://www.google.com/corporate/execs.html#larry (last visited Mar. 26, 2007); Google Corporate Information: Management, Sergei Brin, http://www.google.com/corporate/execs.html#sergey (last visited Mar. 26, 2007).

10. Hewlett-Packard is one of the original garage companies, started by Dave Packard and Bill Hewlett in the late 1930s. *See* HP Company Information, HP History: HP Timeline – 1930s, http://www.hp.com/hpinfo/abouthp/histnfacts/timeline/hist_30s.html (last visited Feb. 12, 2007). More recently, search engines Yahoo and Google were both created by graduate students as little more than hobbies. *See* Rank for Sales, How Yahoo Was Founded, http://www.rankforsales.com/n-ay/719-seo-aug-18-04.html (last visited Mar. 26, 2007); Wikipedia, Google, http://en.wikipedia.org/wiki/Google (last visited Feb. 12, 2007).

11. Professor Lessig has analyzed the distinctive role of technological constraints on innovation. *See, e.g.*, LAWRENCE LESSIG, CODE AND OTHER LAWS OF CYBERSPACE (1999).

innovations, many of which may have seemed unimportant at their conception, have led to amazing social benefits. Yet these innovations in particular are threatened by the current, heavily risk-averse and pro-corporation legal climate for technology law.

Many software innovations create enormous benefits and enormous harms. They interfere with existing corporate and government interests, and are challenged through the legal system as a result. Legislatures create new laws and courts extend existing laws to contain this interference. Some of the legislative actions, such as restrictions on the sending of unsolicited commercial e-mail,[12] correct clear, widespread social problems. Others, such as the Digital Millennium Copyright Act of 1998, serve narrower corporate interests, and place undesirable restrictions on legitimate activity.[13] Rarely does freedom win in the battle against legal incumbency.[14] The courts have followed a similar pattern. In *Grokster*, the most recent major judicial statement on the regulation of innovation, the Supreme Court introduced a new theory of copyright liability, inducement, to restrict the activity of software developers.[15] The courts occasionally but rarely introduce exceptions.[16] As a result of this tightening, innovators face strict, yet vague controls over the functionality of their developments, and they fear that they may face injunctions or even massive statutory damages.

The balances of interests drawn by cases such as *Grokster* are far from optimal, because they are *static* balances. Courts consider only the current benefits and harms of software, and do not take into account long term and external costs of regulation to the innovator and to other innovators. These errors of judgment result in a balance that, generally, overvalues damage to legal interests and undervalues damage to innovation.[17] Fixing the squeaky wheel in this case greatly reduces

12. Federal and state laws restrict the sending of spam. The federal law is the CAN-SPAM Act. *See* CAN-SPAM Act of 2003, Pub. L. No. 108-187, 117 Stat. 2699 (codified at 15 U.S.C. §§ 7701-7713). California, among many other states, also has a thorough anti-spam law. CAL. BUS. & PROF. CODE § 17529 (West 2004), *available at* http://www.spamlaws.com/state/ca.shtml.

13. *See generally*, Jeffrey D. Sullivan & Thomas M. Morrow, *Practicing Reverse Engineering in an Era of Growing Constraints Under the Digital Millennium Copyright Act and Other Provisions*, 14 ALB. L.J. SCI. & TECH. 1 (2003).

14. One well-publicized example is the Family Movie Act of 2005, part of the Family Entertainment and Copyright Act of 2005, which amended federal copyright law to allow technological blocking of non-family-friendly portions of movies. *See* Family Entertainment and Copyright Act of 2005, Pub. L. No. 109-9, 119 Stat. 218 (codified as amended in scattered sections of 2, 17, 18, 28, and 36 U.S.C.).

15. *Grokster*, 545 U.S. at 936-37.

16. The most significant of these is the *Sony* safe harbor, an exemption from contributory copyright liability for the distributors of devices that possess "significant non-infringing use." *Sony*, 464 U.S. at 442.

17. One might argue that this approach is justified because the future benefits of

future social value.

Regulation must be made under the guidance of a broad innovation policy, one that considers not only the observable infringing and non-infringing uses of software, but also the effects of constraints on future innovation. The concerns of software innovation policy overlap with those of intellectual property, the First Amendment, contract law, and antitrust. But it is distinct from these, as it serves different goals and is concerned with different risks, and it deserves independent consideration. Software innovation policy must protect the benefits of new software innovations while limiting the harms of those innovations, and it must preserve an open and unconstrained environment for innovation, free from undue chilling effects and other powerful disincentives.

I propose that the legal system resolve these conflicting interests through a two-part standard. First, the benefits and harms of the innovation itself, and the benefits and harms in the repercussions of the decision to prohibit or to permit the technology, are balanced, in order to decide whether society is better off, now and in the long term, with or without the innovation. This is the *liability rule*, intended to determine, as an initial matter, whether or not the innovation should be permitted or controlled. The liability rule is dynamic – it looks not just at the current uses of the innovation (the static considerations emphasized by current law), but also at foreseeable future uses, and at the external costs and benefits of regulating or permitting the innovation. Second, and only if the answer to the liability rule is to prohibit the innovation, the intent of the developer is examined to determine whether the appropriate remedy is to enjoin continued development and distribution of the innovation, or to hold the developer responsible for damages. This is the *remedy rule*, designed to structure the legal response to liability in a manner that is neither under- nor over-broad. By separating the legal standard into liability and remedy, and by using proper rules at both levels, the courts can make a clear and correct decision at to whether the technology should be permitted (without using the developer's motive as a proxy for proper decision, as the Court in *Grokster* does), and can structure the remedy in a manner that does not create excessive chilling effects by making other well-intentioned developers fear massive damages.

In this paper, I develop these issues further. In section II, I explain why software innovation is at risk and why it must be protected. In

innovation are speculative and therefore not appropriate for judicial decision-making. There are two compelling reasons not to follow this theory here. First, simply ignoring the prospect of future innovation is absurd, and the consequences would be severe. Second, some types of non-specific long-term harm to innovation, such as the imposition of chilling effects on future developers, can be avoided easily through proper policy, such as the proposal presented in this paper.

section III, I give more detail on the current legal system's approach as established by *Grokster*. I argue that *Grokster* is both over- and under-protective, and I propose a two-part liability-remedy standard that accurately protects innovation. In section IV, I discuss and criticize a number of alternative proposals for the protection of innovation, including expansive readings of copyright's fair use exception and of the First Amendment, and I address potential challenges to my standard.

II. WHY MUST WE PROTECT SOFTWARE INNOVATION AND HOW IS IT AT RISK?

A. Why is Software Innovation Different from Other Forms of Innovation?

Software innovation stands apart from other forms of innovation in many ways. The first of these is discussed in almost every work dealing with the new digital era: the marginal cost of additional copies of the technology is negligible. This is, of course, one of the primary reasons for the creation of intellectual property rights in the first place – the creator cannot internalize the benefits of the technology if the creation of additional copies cannot be controlled and formed into a market, and thus the creator has a greatly reduced incentive to innovate.[18] Redistributing software products is fundamentally different from redistributing physical property, such as a piece of furniture, or many other goods protected by intellectual property, such as textbooks. While a textbook can be reproduced by a photocopier, the labor requirements of this process make mass redistribution impractical, unlike the negligible cost of uploading and downloading a digital file.

There are other major differences as well. As mentioned earlier, the scale of effort and time required to create most software programs is nowhere near the scale required to create other types of innovations. Consider pharmaceuticals – laboratories spend years and millions of dollars on development and testing, and still many of their creations end up being unusable or unmarketable. The industry relies on the blockbuster drug in order to survive. Software development, in contrast, happens in large part by individuals, even hobbyists.[19] Sure, there are some notable larger products, such as Microsoft's Windows operating system. But even large software programs such as operating systems can

18. *E.g.*, WILLIAM M. LANDES & RICHARD A. POSNER, THE ECONOMIC STRUCTURE OF INTELLECTUAL PROPERTY LAW 11 (2003) ("Because intellectual property is often copiable by competitors who have not borne any of the cost of creating the property, there is fear that without legal protection against copying the incentive to create intellectual property will be undermined.").

19. *See, e.g.*, How Yahoo Was Founded, *supra* note 10.

be (and are) created by amateurs, because the development process can be distributed across widely dispersed individuals.[20] Furthermore, software innovation is often a highly collective phenomenon, in which the freely exchanged code and ideas help others with their innovations.[21] These structural features of the process of innovation in software render it prone to crippling regulation – for many innovations there are no companies with teams of lawyers and large capital investments worth defending. Instead, software innovators are often amateurs with many other things going on in their lives, who lack the necessary cohesion and economic motivation to lobby legislators and administrators to shape the law in their favor.

Perhaps the biggest and the most salient difference between computer software and other forms of innovation lies in its adaptability to both socially beneficial and socially harmful uses. Software programs may be created for a single purpose, or for no purpose at all, merely to express some creative impulse of the programmer. But others can later adapt these same programs, either through additional programming or simply through unintended usage, to perform functions beyond those imagined by their programmers. In other words, the original intention of the programmer and the original uses of the program are not enough to form a complete evaluation of the program's overall social utility, complicating further the ability of a primarily backward-looking legal system to resolve equity questions concerning software programs.

Finally, the law treats software innovations differently than other forms of innovation. With most technological innovations, patent law serves as the primary legal control. In software development, on the other hand, copyright law, patent law, and focused statutes such as the Digital Millennium Copyright Act all play major governing roles. Copyright law's fair use provisions and the First Amendment have also had significant impact on software development and use. Even beyond these formal legal systems, software programs come equipped with End-User License Agreements, which use contract law to place additional restrictions on the use of a product. This quagmire of assorted laws places a variety of substantively different limitations on the development

20. The Linux operating system is the classic example of this. *See, e.g.*, Yochai Benkler, *Coase's Penguin, or Linux and the Nature of the Firm*, 112 YALE L.J. 369, 406 (2002).

21. Isaac Newton famously wrote, "If I have seen further, it is by standing on the shoulders of giants." Letter from Sir Isaac Newtwon, Trinity College, to Dr. Robert Hooke (Feb. 5, 1675) *in* SIR DAVID BREWSTER, MEMOIRS OF THE LIFE, WRITINGS, AND DISCOVERIES OF SIR ISAAC NEWTON 142 (1855). Many modern scholars have written on the role of the commons in modern information production. *See generally* LAWRENCE LESSIG, THE FUTURE OF IDEAS: THE FATE OF THE COMMONS IN A CONNECTED WORLD (2001); Benkler, *supra* note 20, *passim*.

of software, and leaves little room for the protection of software in and of itself, as a matter of policy. A separate, independent examination of the nature of software innovation, and of what must be done to protect it, must be conducted outside the doctrinal boundaries of any individual source of regulation.

B. Why is Software Innovation Valuable?

The value of the computing industry as a whole should not need to be argued. The value of innovation, on the other hand, deserves some elaboration. Continuing software innovation confers a number of benefits on the computing industry. Some innovations create new ways in which computing better organizes and makes available information from the outside world.[22] Some improve on existing functionality, either by reducing inefficiency[23] or by improving correctness.[24] Many add new features to existing essential products.[25] These benefits enable the management of ever more data from the outside world, leading to faster and more reliable communications, more powerful computations for scientific applications, and improved efficiency in all operations from hospitals to warehouses to personal computers. To continue creating new social benefits, the computing industry requires a sustained high level of innovation, to keep up with the increasing sources, uses, and amounts of data that must be processed.

Promoting small innovators, in particular, supports a number of other related social values. For example, many legal scholars are studying peer production, a less hierarchical, more fluid and collaborative form of production of knowledge goods.[26] Peer production

---

22. Google Book Search, for example, adds new functionality to the industry. Google Book Search, http://books.google.com/ (last visited Mar. 26, 2007). While the concept of scanning a book is not new, I contend that creating a searchable database of the text of many books is a new and valuable innovation. *See infra* Part II.E.4.

23. Consider the development of the MP3 audio encoding, which permits far more compact storage of high quality audio music. *See, e.g.*, Mp3licensing.com, About mp3, http://mp3licensing.com/mp3/index.html (last visited Mar. 26, 2007).

24. Ongoing improvements in speech recognition software, for example, provide continually more accurate transcriptions. *See, e.g.*, Posting of Amit Agarwal to Digital Inspiration Blog, http://labnol.blogspot.com/2007/01/dragon-naturallyspeaking-9-speech.html (Jan. 22, 2007).

25. Consider journaling file systems such as Redhat's ext3, which serve the same purpose as ordinary file systems, yet implement this purpose in a way which adds new logging to increase reliability. Michael K. Johnson, Whitepaper: Redhat's New Journaling Filesystem: Ext3, http://www.redhat.com/support/wpapers/redhat/ext3/ (last visited Mar. 26, 2007); *see generally* Wikipedia, Journaling File System, http://en.wikipedia.org/wiki/Journaling_file_system (last visited Mar. 26, 2007).

26. *See, e.g.*, Benkler, *supra* note 20, at 375-378 (describing in detail the ability of peer production to organize and produce effectively despite its decentralization and lack of formal incentives relative to the traditional Coasean model of the firm).

improves the quality and speed of software development, increases the diversity of viewpoints in the media landscape, and promotes a cultural democracy.[27] Many digital innovations are peer produced, most notably the Linux operating system.[28] If the legal system does not protect innovation, peer production will lose the tools and the freedom it requires, and many valuable innovations will be lost.

Amateur participation in software development also helps to correct the digital divide.[29] Hobbyists, from the United States and from abroad, need only a computer and an Internet connection in order to produce and distribute their own software. A software business can be started without taking out loans to acquire capital, establishing real estate, and hiring employees. Software innovation also helps and is helped by the Access to Knowledge movement.[30] The A2K movement, in part, works to ensure that the information and tools needed to innovate are widely available;[31] but, also, the protection of software innovation preserves freedom to acquire and share knowledge (because amateurs feel free to develop and distribute their own software) and enables the development of communications and management tools necessary to share and organize information, advancing the A2K movement in return.

C. What Will We Lose if We Do Not Protect Software Innovation Adequately?

Prohibiting innovation steals from society any beneficial value of that innovation. Many challenged (or challenge-able) software innovations provide considerable social benefits. For example, the Tor network provides anonymity, which can be used to disguise the identities of copyright infringers, but can also be used to preserve privacy and the freedom of speech.[32] As another example, the creators of the BnetD server may have violated the terms of a license agreement, but they created a program that encourages competition by offering a valuable

---

27. For more on democratic culture and the Internet, s*ee, e.g.*, Jack M. Balkin, *Digital Speech and Democratic Culture: A Theory of Freedom of Expression for the Information Society*, 79 N.Y.U. L. REV. 1 (2004).

28. While in this sense innovation benefits from peer production, peer production also depends on good innovation policy. Full exploration of the synergy between these movements is beyond the scope of this paper.

29. The "digital divide" is the social rift between those who can use and benefit from high technology and those who cannot. *E.g.*, Digital Divide.org, Digital Divide: What It Is and Why It Matters, http://www.digitaldivide.org/dd/digitaldivide.html (last visited Apr. 4, 2007).

30. *See, e.g.*, Posting of Jack Balkin to Balkinization, http://balkin.blogspot.com/2006/04/what-is-access-to-knowledge.html (Apr. 21, 2006).

31. *Id.* (noting that components 2 and 4 of the typology of "access to knowledge" are "Information" and "Tools for the production of knowledge-embedded goods").

32. The Tor system is discussed in more detail in Part II.E. *See infra* notes 76-77.

alternative to Blizzard's official video game servers.[33] If these technologies are too constrained, society will lose their benefits as a consequence of avoiding their harms.

Some technologies that support both benefits and harms should be prohibited, and some should be permitted. Society is not best served by turning a blind eye to the harms of technology, to preserve its benefits regardless of the consequences. The inability to predict the future combined with the harm of letting the technology go through a "trial period" makes infeasible any policy that never prohibits innovation.[34] But if the perceivable beneficial use of a technology is sufficient, then society will lose a great deal of value if the technology and the developer are not protected against legal challenges.

Beyond depriving society of the benefits of the individual innovation, broad regulation produces more peripheral (but no less severe) damage in the form of chilling effects. A chilling effect in the First Amendment context occurs whenever a vague regulation on activity, enforced by criminal sanction, provides too strong an incentive for a legitimate speaker to remain silent out of fear of prosecution.[35] In the context of technology law, one can imagine an analogous chilling effect, in which software developers fear production and distribution of their software because it may trigger liability under copyright law or the Digital Millennium Copyright Act.[36] This concern is made especially ominous by copyright law's severe financial penalties for infringement.[37] The fear of large damage awards empowers rights holders to threaten enforcement of existing laws beyond their actual scope through the use of "cease and desist" letters.[38] As applied to innovation, chilling effects

33. The BnetD system is discussed in more detail in Part II.E. *See infra* note 64.

34. Consider, for example, a software virus. In theory, it is possible that a software virus may lead to future social benefits, such as an increased investment in security or an increased awareness of computer security. But this is too long-term and too speculative, and certainly insufficient to justify permitting a virus to cause harm for a while, just to see if it eventually produces beneficial use.

35. *See, e.g.*, Reno v. ACLU, 521 U.S. 844 (1997).

36. To settle multiple lawsuits against them, Niklas Zennstrom and Janus Friis, the developers of the Kazaa file sharing system, agreed to pay $125 million in damages. Jeremy W. Peters, *Kazaa's Creators Do Latest Venture by the Book*, N.Y. TIMES, Feb. 27, 2007, *available at* http://www.nytimes.com/2007/02/27/technology/27joost.html.

37. *See, e.g.*, Fred von Lohmann, Electronic Frontier Found., Remedying Grokster, July 25, 2005, http://www.eff.org/deeplinks/archives/003833.php ("much of the copyright chill felt by innovators and technology investors can be traced to the prospect of apocalyptic statutory damages that can reach beyond the corporate grave into the personal assets of officers, directors and investors."). Statutory damages in copyright law range from $750 to $30,000 per infringement. 17 U.S.C. § 504(c)(1) (2000). The chilling effect is also amplified by the prevalence of amateur innovators, who would not have the resources to pay attorneys to defend a legal challenge, much less survive a losing decision.

38. The Chilling Effects Clearinghouse, http://www.chillingeffects.org (last visited Feb. 13, 2007) (project is collecting and publishing these letters to increase public notice of First

are generated whenever an innovator is held liable solely for the functional features of the innovation. The best example of this in case law is *A&M Records, Inc. v. Napster, Inc.*, 239 F.3d 1004 (9th Cir. 2001). Nothing like the Napster service had existed before; while the developers might have suspected their service was illegal, there was no way for them to know. In the future, an innovator who fears retribution may refrain from creating and distributing software that is actually legal and valuable for society.[39] What society loses from overregulation, then, is the social value of these foregone innovations.

D. The Legal Climate for Innovation

While software patents exist, the greatest restrictions on software innovation come from copyright law and the Digital Millennium Copyright Act.[40] Since it was passed in 1998, the Digital Millennium Copyright Act has served as one of the most popular legal tools to stifle innovation and competition in the technology industry. The DMCA prohibits the circumvention of a technological protection measure used to protect copyright.[41] The DMCA creates a legal obstacle to technological arms races – sequences of maneuvers where security mechanisms broken by third parties are replaced by stronger mechanisms which are themselves broken. But many private parties have tried to use the law to stifle legitimate competition. It has been used (not always successfully) to challenge generic ink cartridges,[42] video game servers,[43] and garage door openers.[44] These attempts demonstrate the risks that the DMCA poses to innovation, risks that were only briefly acknowledged during the bill's passage.[45] And the legislators' minor nods towards the

---

Amendment and intellectual property rights).

39. Note that, in contrast to First Amendment chilling effects, this conception of chilling effects has considerable utilitarian value. While the direct effect is on the innovators who fear legal retribution, the ultimate loser is society, which is deprived of the benefits of the innovations that would otherwise have been created.

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

41. 17 U.S.C. § 1201(a) (2000).

42. *See, e.g.*, Lexmark Int'l, Inc. v. Static Control Components, Inc., 387 F.3d 522 (6th Cir. 2004).

43. *See, e.g.*, Davidson & Assoc., Inc. v. Internet Gateway (Internet Gateway II), 422 F.3d 630 (8th Cir. 2005).

44. *See, e.g.*, Chamberlain Group, Inc. v. Skylink Techs., Inc., 381 F.3d 1178 (Fed. Cir. 2004).

45. In comments on what would become 17 U.S.C. § 1201(f), an exception for activities constituting reverse engineering for the purpose of creating interoperable products, Senator Orrin Hatch stated that "[t]he purpose of this section is to foster competition and innovation in the computer and software industry." S. REP. NO. 105-190, at 13 (1998). Then-Senator John Ashcroft appeared concerned that the statute might be interpreted to mandate technology design, "which would have a dampening effect on innovation." 144 CONG. REC. S4890 (daily

value of innovation have been overshadowed by the practical applications of the bill and by other legislative action, such as the oft-attempted Broadcast Flag bill.[46]

Copyright law prohibits direct infringement in software development (e.g. by copying and using source code from one program to another without permission). Common law (based on copyright law principles) also prohibits secondary infringement, such as the development of a software tool that is used by others to infringe copyright. Historically, secondary infringement doctrine had two separate grounds for liability, contributory and vicarious. Contributory liability requires that a software developer "knowingly" and "materially" provide assistance to a direct infringer.[47] Vicarious liability requires a developer to have a "financial interest" in the infringement and have "the right and ability to supervise" the infringing activity.[48] In *Metro-Goldwyn-Mayer Studios, Inc. v. Grokster, Ltd.*, 545 U.S. 913 (2005), the Supreme Court added a third basis for liability, inducement, under which software developers could be held liable for secondary infringement if they "induced" the use of their software to commit copyright infringement.[49]

The Court created one important exception for secondary copyright liability. In *Sony Corporation v. Universal City Studios*, 464 U.S. 417 (1984), the Court held that contributory liability for copyright infringement did not apply to the makers of a device if that device had "substantial non-infringing use[.]"[50] The Court protected Sony from liability for producing and selling the Betamax video recording device, which permitted both time-shifting of television programs and the assembly of home libraries of television shows. This, of course, amounted to a decision not to prohibit the video recorder, because it was more beneficial than harmful for society. We are all fortunate that the Court was as open-minded as it was.

---

ed. May 14, 1998) (statement of Sen. Ashcroft). Ashcroft pushed for an amendment to ensure that the statute did not require technology to be designed in compliance with any protection measures. *Id.*

46. The Broadcast bill directly realizes Ashcroft's fear of mandating technology design to enforce compliance. *See generally* Electronic Frontier Foundation, Broadcast Flag, http://www.eff.org/IP/broadcastflag (last visited Feb. 9, 2007); Public Knowledge, Broadcast Flag, http://www.publicknowledge.org/issues/broadcastflag (last visited Feb. 9, 2007).

47. *See, e.g.*, *Sony*, 464 U.S. at 487 (citing Gershwin Publ'g Corp. v. Columbia Artists Mgmt., Inc., 443 F.2d 1159, 1162 (2d Cir. 1971)).

48. *See, e.g.*, *Napster*, 239 F.3d at 1022 (9th Cir. 2001) (citing *Gershwin Publ'g Corp.*, 443 F.2d at 1162).

49. *Grokster*, 545 U.S. at 936-38.

50. *Sony*, 464 U.S. at 442.

E. Innovations Under Attack

1. Peer-to-Peer Filesharing

One of the most controversial innovations in recent years has been software for peer-to-peer filesharing, often known as P2P. P2P networks allow individuals to exchange digital files with other computers connected to the Internet. Users can download copies of files offered by others, and can upload their own files to the network. Most networks enable users to search for files that match a user-entered description. These networks are commonly used to exchange copyrighted digital media files, such as music and movies. The copyright holders have used the legal system to challenge both the distribution and the use of P2P software programs.[51]

The first major peer-to-peer network was Napster. The Napster system consisted of a central index that linked to files offered by users; this central index enabled participants in the system to quickly locate desired content.[52] Multimedia content producers quickly brought suit against the software developers to prevent the continued operation of the network and distribution of the software. The Napster creators lost these suits, the Napster server was shut down, and the software distributors stopped development and distribution. Grokster, a peer-to-peer file sharing network that operates without a central index, succeeded Napster.[53] Content producers brought suit again, to hold the developers liable for the copyright violations of the users of the software.[54] Future peer-to-peer filesharing networks, more technologically advanced and more difficult to shut down than Grokster, will no doubt lead to more lawsuits.[55] In addition to suing the innovators, the content producers are

---

51. *See BMG Music*, 430 F.3d at 888; *Napster*, 284 F.3d at 1091.

52. *Napster*, 239 F.3d at 1011-12. Since its legal challenge, the Napster name and logo have been assigned to a legal music download-for-pay service. *See* Napster, http://www.napster.com (last visited Feb. 9, 2007).

53. *See Grokster*, 545 U.S. at 919-27 (describing Grokster's architecture).

54. The suit reached the Supreme Court in *Grokster*, in which the Supreme Court held that secondary liability for copyright could be triggered by "inducing" direct infringements of copyright; the Court then remanded the case for further proceedings considering liability under this theory. *Id.* at 936-38.

55. The Freenet and BitTorrent represent different technological advancements beyond the Grokster network. Freenet adds redundancy and anonymity to strengthen peer-to-peer networks against disruptions. Freenet Project, http://freenetproject.org (last visited Feb. 9, 2007). BitTorrent breaks large files into small pieces and uses multiple sources for each download. BitTorrent, http://www.bittorrent.com (last visited Feb. 9, 2007). This makes BitTorrent the technology of choice for downloading pirated movies, along with legitimate large digital. The MPAA has been very resistant to BitTorrent, and many popular "trackers" (sites providing pointers to file downloads), including the once-popular LokiTorrent, have settled. Ashlee Vance, *MPAA closes Loki*, REGISTER, Feb. 10, 2005, http://www.theregister.co.uk/2005/02/10/loki_down_mpaa/.

also suing the users of the networks for direct copyright infringement.[56] Because the targets of such suits cannot afford to risk full liability (where a damage award could amount to $150,000 per song),[57] the vast majority have settled out of court.[58]

Peer-to-peer filesharing may well have considerable negative effects, beginning with the narrow economic effects focused on reduced sales by music producers. It is quite rational to assume that many consumers will not purchase music that they can acquire in a nearly identical form for free. As the corporation loses more money, it receives less in return for its investments in the artists and in production, marketing, and distribution. This may discourage some individuals from starting recording companies, and may lessen expansion efforts of existing companies, possibly depressing the entire industry. Many skeptics have responded by claiming that the actual losses caused by peer-to-peer file sharing are nowhere near as large as alleged[59]; some note that music sharing exposes users to many new bands, supplementing the marketing efforts of the recording industry, and thus results in increased sales.[60] But we cannot determine with any degree of certainty how much money recording companies are losing because of file sharing; we certainly cannot tell how much they would lose if the legal system were to condone file sharing. It is possible that a great many more music fans would acquire their music for free.

However, punishing the file sharer and punishing the software developer are not the same, even if they are intended to address the same problem. Punishing the software developer harms society in other ways which must be taken into account. Peer-to-peer networks, especially modern ones such as BitTorrent, are efficient means of distribution for digital content.[61] They greatly reduce unnecessary overhead in production and distribution, not just for music files, but also for large software packages and other files.[62] BitTorrent technology is currently used to transfer many legal files.[63] If the BitTorrent software were

56. *See BMG Music*, 430 F.3d at 888.

57. 17 U.S.C. § 504(c)(2) (2000).

58. *See, e.g.*, *Court Rules Against Song Swappers*, BBC NEWS, Jan. 27, 2006, http://news.bbc.co.uk/2/hi/entertainment/4653662.stm; *but see BMG Music*, 430 F.3d at 888.

59. *See, e.g.*, Michael Geist, *Piercing the Peer-to-Peer Myths: An Examination of the Canadian Experience*, FIRST MONDAY, Apr. 2005, http://www.firstmonday.org/issues/issue10_4/geist/.

60. *See, e.g.*, Owen Gibson, *Online File Sharers 'Buy More Music'*, GUARDIAN UNLIMITED, July 27, 2005, http://www.guardian.co.uk/arts/news/story/0,11711,1536886,00.html.

61. *See, e.g.*, John Borland, *File Swapping Shifts Up a Gear*, CNET NEWS.COM, May 27, 2003, http://news.com.com/2100-1026_3-1009742.html.

62. *Id.*

63. New versions of the Linux operating system are routinely distributed through BitTorrent, as they are downloaded by many users in parallel shortly after their release. *See,*

prohibited, society would lose the benefit of using the network for these transfers. And this loss is insignificant compared to the chilling effects that would follow from punishing the developers of the networks. Punishing the developers might scare away the programmers who would otherwise have developed the software behind the next revolution.

2. Blizzard v. BnetD

The recent 8th Circuit Case *Davidson & Associates, Inc. v. Internet Gateway*, 422 F.3d 630 (8th Cir. 2005), also known as "Blizzard v. BnetD," concerns the video game company Blizzard's "Battle.net" online service, which enables users of multiple Blizzard video games to play each other over the Internet.[64] As part of its functionality, the Battle.net service prevented pirated copies of the video games from being played online.[65] Out of frustration over problems with the service, a group of users of Blizzard games developed their own server software, "BnetD," to replace Blizzard's official servers.[66] The BnetD designers could not enable their server to block illegal games, as Blizzard did not make available its detection process for illegal games.[67]

Blizzard brought suit against the BnetD designers in order to enjoin the operation of their service, alleging violations of the Digital Millennium Copyright Act and of the license agreements for use of the software.[68] The programmers of BnetD in response claimed that their actions in creating the BnetD service constituted reverse engineering to produce an interoperable product, and thus were covered by explicit protections for reverse engineering in the DMCA.[69] But because BnetD-

e.g., The Linux Mirror Project, http://www.tlm-project.org (last visited Feb. 9, 2007).

64. Full details on the case, including links to all court documents, are available through the Electronic Frontier Foundation, who served as co-counsel for the case. Electronic Frontier Foundation, Blizzard v. BNETD, http://www.eff.org/IP/Emulation/Blizzard_v_bnetd (last visited Feb. 9, 2007). The district court decision found for the video game manufacturers. *See, e.g.*, Davidson & Assoc., Inc. v. Internet Gateway, Inc. (Internet Gateway I), 334 F. Supp. 2d 1164 (E.D. Mo. 2004).

65. Brief of Defendants-Appellants at 17, Internet Gateway II, 422 F.3d 630 (8th Cir. 2005) (No. 04-3654).

66. *Id.* at 8.

67. Given the weakness of the authentication mechanism, widely publishing this information would have made it easy for users of unauthorized copies of the games to disguise their games as legitimate. This is known in the computer science community as "security through obscurity," and is considered unacceptably weak. *See, e.g.*, S. Forrest et al., *Building Diverse Computer Systems*, 1997 PROC. OF THE 6TH WORKSHOP ON HOT TOPICS IN OPERATING SYS. 71 (1997) ("Within computer security there is widespread distrust of 'security through obscurity' . . . .").

68. *Internet Gateway I*, 334 F.Supp. 2d at 1167.

69. *Id.* at 1183-84. The DMCA's protections for reverse engineering are codified at 17 U.S.C. § 1201(f) (2000). The parties' argument was based on a recent case upholding this exception in the context of reverse engineering printer ink cartridges. *See Lexmark Int'l*, 387 F.3d at 522.

based servers permitted illegal copies of games to be played online, the district court found that the actions of the BnetD developers went beyond the scope of the exception for production of interoperable products and constituted copyright infringement.[70] Additionally, the district court found that the BnetD program constituted an anti-circumvention device in the language of the DMCA.[71] The Eighth Circuit affirmed the judgments of the district court.[72]

Permitting the BnetD server to operate bears little risk of significant social harm. There are two categories of possible damages: competition between BnetD and Battle.net, and the marginal increase in the value of illegal copies of Blizzard games (coupled with a greater incentive to make such copies) through online play enabled by the use of BnetD-based servers. As for the former, if the BnetD server is good enough to take users away from the (free) Battle.net service, then it possesses inherent social value which exceeds the minor loss in Blizzard's motivation to invest in the service resulting from the lost advertising revenue associated with the service. Additionally, if Blizzard improves their Battle.net service to win customers back, society benefits from the competition.

As for the marginal increase in value of illegal games, it is possible that Blizzard may lose some sales revenue. Some who would otherwise have bought a legal copy of a Blizzard game may decide to acquire an illegal copy because the BnetD server permits the illegal copy to be played online. But this is a small portion of the value of the video games – even without the Battle.net server, illegal copies of games can be played offline, and even over Local Area Networks (LANs). As a method for discouraging piracy, reducing the value of the games by this small a margin is likely to prove ineffective.

Prohibiting the BnetD server, on the other hand, carries a great risk of significant social harm. It grants Blizzard the power to eliminate any competition with their Battle.net service. While the court did not grant a damage award to the plaintiffs, as that issue was settled out of court,[73] an award of damages in a similar case would have the same chilling effects discussed in the context of peer-to-peer networks. Additionally, the 8th Circuit upheld in full the software license agreement governing the Blizzard software, despite its conflict with the reverse engineering protections of the DMCA.[74] This decision ignores a Congressional balance governing technological protection measures, and it may have

70. *Internet Gateway I*, 334 F.Supp. 2d at 1184-85.
71. *Id.* at 1186-87.
72. *See Internet Gateway II*, 422 F.3d at 630.
73. *Internet Gateway I*, 334 F. Supp. 2d at 1167.
74. *Internet Gateway II*, 422 F.3d at 641-42.

repercussions which extend far beyond this case and which cause great detriment to society.[75]

3. Tor

The Tor communications system is an implementation of a technology known as "onion routing."[76] Onion routing protects the anonymity of an Internet user by routing messages through multiple intermediate nodes.[77] Each intermediate node hides the origin of messages in such a way that a reply message can reach the original source node, and yet no node knows more of the path of the message than the nodes immediately before and after it on the message path.[78]

Providing anonymity for Internet traffic has significant positive social benefits. The anonymity and encryption provided by the service make it far more difficult for ISPs and nations to censor the speech of Internet users, and make it impossible to monitor Internet traffic to collect personal information. But anonymizers, like Tor, enable undesirable activities as well. Users of the Tor network can transfer copyrighted files or child pornography through the network. Anonymity makes it more difficult for law enforcement officials to determine the identity of the illegal actors.

The legal status of Tor is far from clear. Because Tor can be used to facilitate the transfer of copyrighted files without detection, the governing legal doctrine is secondary copyright infringement. The tests of *Sony* and *Grokster* apply. The rule of *Sony* is that contributory liability for copyright infringement cannot be assigned to the makers of a device if that device had "substantial non-infringing use[.]"[79] Tor clearly has some non-copyright-infringing uses, through its protections of free speech and privacy. Whether this is "substantial" is a decision for the courts to make. The *Grokster* opinion holds that the makers of a device can be held liable for secondary infringement if they "induced" the use of the device in an infringing manner.[80] This opinion has not been widely tested, and it is unclear what will constitute inducement, and unclear whether or not this doctrine could be used to regulate Tor.

As with other innovations, the positive social value of the Tor network is significant, and must be considered even if the system

75. This was one of the primary arguments of the counsel for the defendants. Brief of Defendants-Appellants, *supra* note 65, at 39.

76. Tor Homepage, http://tor.eff.org/ (containing a basic description of the Tor system and onion routing technology) (last visited Feb. 10, 2007).

77. Tor, Overview, http://tor.eff.org/overview (last visited, Feb. 10, 2007).

78. *Id.*

79. *Sony*, 464 U.S. at 442.

80. *Grokster*, 545 U.S. at 936-38.

facilitates illegal activity.

4. Google Book Search

The Google Book Search project allows users to search for keywords and phrases in digitized versions of books.[81] The service acquires books from two sources – publishers provide books directly to Google, and libraries loan books to Google to be scanned (and then returned).[82] Google makes this information available to three different extents. If a book is out of copyright, Google permits the user to scan the entire book. With permission from the publisher or author, Google allows a few sample pages of the book to be seen.[83] Otherwise, Google displays card catalog information about the book, and a few sentences around the search term.

The structure of this system provides the most benefit to users while causing the least harm to the interests of the copyright holders.[84] As with many of its products, Google has deliberately chosen not to internalize many of the benefits of the service.[85] This service is an enormous public good and does little harm to publishers. It may in fact benefit them extraordinarily, as it makes it easier for consumers to find books they may want to purchase. Despite all of this, many otherwise innovation-friendly thinkers have spoken out against the project.[86] Two lawsuits have already been filed against Google by groups of publishers.[87] Their suits are not unfounded – Google's actions include making an

81. Google, About Google Book Search, http://books.google.com/intl/en/googlebooks/about.html (last visited Feb. 10, 2007).

82. *Id.*

83. *Id.* Note that the Google site says "publisher or author", but depending on the author's agreement, it is likely that a published book would require the publisher to agree to the display.

84. *See* Eric Schmidt, Op-Ed, *Books of Revelation*, WALL ST. J., Oct. 18, 2005, at A18; Posting of Susan Wojcicki to Official Google Blog, http://googleblog.blogspot.com/2005/09/google-print-and-authors-guild.html (Sept. 20, 2005).

85. Schmidt, *supra* note 84 ("[W]e don't make a penny on referrals. We also don't place ads on Google Print pages for books from our Library Project, and we do so for books in our Publishing Program only with the permission of publishers. . . .").

86. Posting of Siva Vaidhyanathan to Sivacracy.net, http://www.nyu.edu/classes/siva/archives/001841.html (Aug. 12, 2005) (saying that Google's actions may lead to a "copyright meltdown", in which publishers will request and receive Congressional support in further tightening their copyrights). *But see* Posting of Derek Slater to A Copyfighter's Musings Blog, http://blogs.law.harvard.edu/cmusings/2005/10/24#a1449 (Oct. 24, 2005).

87. One suit pits the Author's Guild against Google. Complaint, Author's Guild v. Google Inc., No. 05-CV-8136 (S.D.N.Y. Sep. 20, 2005), *available at* http://news.findlaw.com/hdocs/docs/google/aggoog92005cmp.pdf. The other suit pits McGraw-Hill and other publishers against Google. Complaint, McGraw-Hill Cos. v. Google Inc., No. 05-CV-8881 (S.D.N.Y. Oct. 19, 2005), *available at* http://www.publishers.org/press/pdf/40%20McGraw-Hill%20v.%20Google.pdf.

unauthorized (digital) copy of the published works, though Google has a credible fair use defense.[88] While it would be better for Google to obtain permission from publishers before digitizing their works, this is simply not feasible given the transactional (and actual) costs of negotiating with every publisher over every work. As James DeLong puts it, "[t]o insist that Google get permission means that the post-1923 literature cannot be included."[89]

Google Book Search is different from the preceding examples in many ways. For one, it is the innovation of a major (and wealthy) American corporation. This means that Google is not judgment-proof – it could be held liable for immense damages. At the same time, Google's history of valuable innovations and of being "good"[90] have not gone unnoticed by the public. The risk of losing Google's innovations is far more cognizable than the risk of losing the innovations of an unknown amateur programmer.[91] For another, this is not an innovation in the same sense as others. This is not a single new software program, such as a file sharing client or a network routing tool. But Google Book Search is very much a new software innovation, in part because it represents a new combination and use of existing software tools, and in part because it creates new beneficial and harmful activities that need to be balanced to determine the overall social equity of the service. The Google Book Search example highlights the tradeoff that innovation policy is intended to resolve – it is a software innovation that creates massive social benefits, yet it violates the law as it is constructed. The primary question, then, is whether the violation is so egregious as to require the service to be stopped, or whether the social benefits outweigh the harms.

---

88. Google's claim to fair use may rest in its efforts to transform (by digitizing) the copyrighted work, that it does not overly harm the market for the work, and that it results in significant social value. *See* Kelly v. Arriba Soft Corp., 77 F. Supp. 2d 1116, 1118-23 (C.D. Cal. 1999), *aff'd in part, rev'd in part*, 336 F.3d 811 (9th Cir. 2003); Siva Vaidhyanathan, *A Risky Gamble With Google*, CHRON. OF HIGHER EDUC., Dec. 2, 2005, at B7, *available at* http://chronicle.com/weekly/v52/i15/15b00701.htm; Posting of C.E. Petit to Scrivener's Error, http://scrivenerserror.blogspot.com/2005/10/authors-guild-v-google-5-fair-use.html (Oct. 4, 2005).

89. Posting of James DeLong to IPCentral Weblog, http://weblog.ipcentral.info/archives/2005/10/the_google_prin_1.html (Oct. 20, 2005).

90. Google, Our Philosophy, http://www.google.com/intl/en/corporate/tenthings.html (last visited Feb. 10, 2007) (referring to Rule #6, "[y]ou can make money without doing evil").

91. As Derek Slater puts it, this may be "a chance for a legitimate defendant to take a real shot at making some good law." Slater, *supra* note 86.

III. WHAT IS PROPER SOFTWARE INNOVATION POLICY?

A. Grokster*, or: What is Improper Software Innovation Policy?*

The Supreme Court in *MGM v. Grokster* delivered the most recent statement on software regulation.[92] Before the court were many strong arguments supporting the Grokster software. Respondents' brief notes many values of the technology developed by Grokster. It improves reliability and efficiency over related programs.[93] Businesses have developed around use of the technology.[94] Many music artists have supported the technology, recognizing that it improves their name recognition and increases their fanbase.[95] Respondents also note that, given their originality, the technical innovations may lead to unforeseen future value.[96] Furthermore, the respondents note that any decision to regulate the innovation may lead to complex and expensive future litigation to determine the limits of valid technologies.[97] All of these factors are significant in determining whether as a matter of policy a technology innovation should be regulated.

Justice Breyer's concurrence addresses some issues of the benefits and harms of innovation. Breyer emphasizes *Sony*'s explicit balance of interests,[98] enumerates many positive values of digital technologies,[99] and even considers the respondents' concerns that updating the technology to add a filtering mechanism may be prohibitively difficult[100] and that the technology has led to many new valuable, legitimate businesses.[101]

The majority opinion, in contrast, did little to protect the benefits of innovation. It acknowledged the technical benefits of the innovation and the value of non-infringing uses of the technology.[102] It also expressed a concern that the wrong legal standard may have negative repercussions on legitimate innovation.[103] The Court left *Sony* intact (though still unclear), and it adopted an "inducement" theory of liability, to separate

---

92. *See,* MGM Studios Inc. v. Grokster, Ltd., 545 U.S. 913 (2005).

93. Brief for Respondents at 6-8, *Grokster*, 545 U.S. 913 (No. 04-480).

94. *Id.* at 21.

95. Jonathan Krim, *Artists Break with Industry on File Sharing*, WASH. POST, Mar. 1, 2005, at E5.

96. Brief for Respondents, *supra* note 93, at 25.

97. *Id.* at 30-31.

98. *Grokster*, 545 U.S. at 949-50 (Breyer, J., concurring).

99. *Id.* at 950-56 (Breyer, J., concurring).

100. *Id.* at 957-59 (Breyer, J., concurring).

101. *Id.* at 963-65 (Breyer, J., concurring).

102. *Id.* at 919-20.

103. *Id.* at 936-37. ("We are, of course, mindful of the need to keep from trenching on regular commerce or discouraging the development of technologies with lawful and unlawful potential.").

out and hold liable those developers who acted to induce others to directly infringe copyright, independent of any other grounds for liability.[104] This move further protects the interests of copyright holders, and further chills innovation, to the detriment of society.

Inducement and the existing doctrines of contributory and vicarious liability form a three-part test for liability (with one important exemption). The three parts of *Grokster*'s liability test are all directed primarily to the software's developer.[105] As a proxy for determining whether the software itself is worth protecting, secondary liability investigates the motive, knowledge, and ongoing activity of the software's developer. The Court modified the secondary liability doctrine through *Sony*, creating a technology-specific exemption for devices which have "substantial non-infringing use."[106] This safe harbor restores some of the balance by protecting devices which already have beneficial social value. But even in its original conception, its attachment to static, demonstrable positive uses limits it, given the dynamic nature of the equity of uses of software discussed earlier. And the Supreme Court in *Grokster* emphasized that the exception applies only to contributory copyright liability, and in particular, does not provide an exception to liability for inducement. Even beyond these limitations, the flaws with *Grokster* run deeper than questions of scope. Determining liability for software development on the basis of the activities and motives of developers can produce only an approximation of correct policy because it looks solely at peripheral factors that often bear little relation to the actual social value and harm of the software.

So why persist in the illusion? In part, because it is far easier to create bright-line rules judging human conduct than to create clear rules for the proper social balance of the benefits and harms of technology. One of the foremost concerns of innovation policy is for chilling effects, and establishing bright-line rules (regardless of their correctness) helps developers know how they can avoid liability. Even if an ad-hoc standard based on the value of the technology made more correct decisions, it might be worse for innovation if every developer feared facing and losing a judgment.

---

104. *Grokster*, 545 U.S. at 936-37.

105. Inducement liability examines only the conduct of the actor – whether the actor has promoted the use of the software for infringing purposes. *Id.* Vicarious liability questions the relationship of the developer to the software, in particular whether the developer has the ability and duty to police uses of the software for infringing purposes. *Id.* at 930 n.9. Contributory liability considers in part the technology in requiring "material contribution" to the infringement. *Napster*, 239 F.3d at 1022. But an equal part of the test is the question of whether the actor has knowledge of the infringing activity. *Id.* at 1019 (citing *Gershwin Publ'g*, 443 F.2d at 1162).

106. *Sony*, 464 U.S. at 442.

In theory, this is a strong argument. But comments on the *Grokster* decision have emphasized that it is highly ambiguous.[107] Not only did the Court fail to resolve existing ambiguities in the interpretation of the *Sony* standard,[108] but it also created additional ambiguity by adding a theory of liability based on the intent of the developer.[109] Considerations of intent can be valuable for proper innovation policy, but in the inducement theory as introduced by the *Grokster* court, they both worsen the law's clarity and more strongly attach liability to the actor (and not the software itself). Additionally, the evidentiary requirements for determining the developer's intent will require many cases to survive summary judgment, increasing the risk of expensive litigation and increasing chilling effects imposed on other developers.

B. Separating Liability from Remedy; Separating the Technology from the Developer

Software innovation policy must balance the benefits of individual software innovations, the legal entitlements they harm, and the repercussions of assigning or not assigning liability. It must not prohibit software too readily, or too many social benefits will be shut off. It also must not construct remedies in a manner that places excessive chilling effects on other software developers. Proper policy separates the question of liability for the development and distribution of software into two questions, one of (pure) liability and one of remedy. The liability question focuses on the technology itself, on its benefits and harms to society. The remedy question, asked only if liability is found, focuses on the developer and on the incentives created by assigning various forms of punishment. Current law conflates and misdirects these questions, and as a result, delivers incorrect results. By separating these questions, courts can make optimal dynamic balances while avoiding unnecessary litigation expenses, preserving as much clarity of law as is feasible, and minimizing chilling effects imposed on other developers.

---

107. *See, e.g.,* Galen Hancock, *Metro-Goldwyn-Mayer Studios Inc. v. Grokster, Ltd.: Inducing Infringement and Secondary Copyright Liability*, 21 BERKELEY TECH. L.J. 189, 189 (2006) ("At the same time, Grokster may frustrate copyright owners who will have to satisfy a new and ambiguous indirect infringement standard."); Jefferson Graham, *Entertainment Firms Win File-Sharing Duel*, USA TODAY, June 27, 2005, *available at* http://www.usatoday.com/tech/news/techpolicy/2005-06-27-fileshare-cover-usat_x.htm ("Chipmaker Intel, which filed legal documents in support of Grokster, said the ruling was so ambiguous that the company didn't have an immediate reaction.").

108. *See, e.g.*, Evan F. Fitts, Note, *Inducement Liability for Copyright Infringement is Born*, 71 MO. L. REV. 767, 782 (2006) ("The Court's failure to remedy the ambiguous standards set forth in Sony could have negative effects.").

109. On some level, of course, intent is also a factor in other theories of liability. But it is more central to inducement, which asks if the objective of the developer was to promote the use of the software by others for infringing purposes.

1. The Liability Rule

The question of liability for a software program is, at its heart, the question answered by the court in *Sony*. If technology has both benefits and harms, at what point can a court (or legislature, for that matter) say that the benefits exceed the harms and the technology should not be prohibited? The *Sony* court famously answered this question by declaring that a technology developer could not be liable for contributory copyright infringement if the technology has "substantial non-infringing use." This is a good start for proper policy, but it is too limited. It is limited in its legal applicability, as its safe harbor does not protect against other forms of secondary liability. It is also limited in its scope of consideration, as it reaches only a static balance of interests – current uses, both beneficial and harmful.

The proper rule for determining liability begins with *Sony*'s examination of beneficial and harmful current uses of the technology. It then considers foreseeable future uses of the technology – considering not just empirical reports of current usage patterns, but also trends in usage patterns and expert testimony as to future uses of the technology. Most importantly, the rule weighs the costs of avoiding the harms and retaining the benefits, whether these costs are incurred by the innovator or by the incumbent rights holder.[110] If the innovator can cheaply avoid or reduce the harms of the technology, then a court should favor a finding of liability, to provide an incentive for the innovator to incur the expense of the modifications. Conversely, if the harms can be easily mitigated or avoided entirely by the incumbent rights holder, this should go far towards a finding of no liability.

To avoid the harms, the innovator can modify the technology, for example, by adding filters to a filesharing program to block transfer of copyrighted works. This generates two costs: the cost of implementing the modifications, and the damage that the modifications have on the beneficial uses of the technology, such as false positives generated by a filtering technology, or a heavy burden of additional user effort (such as needing to verify legitimate files) that discourages adoption of the technology. The incumbent rights holder can forestall or at least mitigate the harms as well, through a wide variety of mechanisms. Sometimes the rights holder may be able to increase technological protection measures governing the technology.[111] Some measures are more expensive to implement, such as designing an online distribution system like Apple's iTunes to compete with the filesharing systems, but these systems can

110. This is reminiscent of the "cheapest cost avoider" theory of tort law, and for good reason.

111. Though, this can lead to inefficient racing behavior, if the new modifications can be easily compromised.

also result in great increases in revenue for the company and great benefits for society as a whole.[112] These changes incur costs for implementation and for reductions of the benefits of the innovation, as before. As the example of iTunes demonstrates, they also have the potential to result in broader social benefits; while these are highly speculative, to the extent they can be foreseen, they should also be included in the balance.

Critics of my approach will note that it is on some level more restrictive than the balance drawn by *Sony*. While this approach more clearly acknowledges many of the external costs of regulating or permitting an innovation, it is not as permissive of speculative future benefits as the Court's standard in *Sony*. By permitting any technology that has "substantial non-infringing use," many interpret the *Sony* rule as leaving room to protect innovations that may in the future have significant beneficial use, even if that use is not immediately foreseeable. The rule I offer deliberately omits this consideration, for two reasons. First, while innovations do sometimes lead to unpredictable significant benefits, these are highly speculative and unlikely (in particular if they're not at all foreseeable *ex ante*), and it seems unfair for them to outweigh demonstrable, significant harm in the present. Second, it is also possible that the innovations will lead to significant unforeseeable harms – this is, after all, the nature of the unforeseeable. Any policy must make some compromise, and it is just too costly to permit a current harm out of a purely speculative possibility of future benefits.

2. The Remedy Rule

Once an innovation has been found to be against society's best interests, the next question concerns the proper response. The weaker response merely enjoins the continued distribution and development of the software. The stronger response holds the developer liable for damages. The current legal system takes the latter approach, subjecting secondary infringers to considerable damages.[113] These damages serve

112. *See* Raymond Shih Ray Ku, *The Creative Destruction of Copyright: Napster and the New Economics of Digital Technology*, 69 U. CHI. L. REV. 263 (2002) (the approach of "creative destruction" at work).

113. Secondary copyright infringement imposes the same liability as direct infringement, which carries large statutory damage awards. In practice, the parties often settle on a considerably smaller sum of money, along with injunctive relief. *Compare* How the RIAA Litigation Process Works, http://info.riaalawsuits.us/howriaa.htm#set (last visited Mar. 27, 2007) ("settlement is usually for $3750, non-negotiable, and contains numerous one-sided and unusual provisions, such as a representation that peer to peer file sharing of copyrighted music is a copyright infringement"), *with* 17 U.S.C. § 504(c)(1)-(2) (2000) (authorizing "not less than $750 or more than $30,000" for each infringed work, or up to $150,000 if the copyright owner demonstrated that the infringement was willful).

as a considerable incentive to discourage others from infringing in the future. But these incentives can become too severe. Holding an innocent developer – one who did not intend or desire that his product be used for copyright infringement – liable for large damages scares other innocent developers, who will fear that their products will be wrongly used by others, subjecting them to large damages, placing their personal assets at risk. Instead of indiscriminately imposing damages, a court can apply an intent-based standard in awarding punitive damages, thereby limiting the liability of innocent developers that simply distribute and develop the software.

The Supreme Court has laid the groundwork for such a distinction in its inducement theory in *Grokster*.[114] The Court did not specify the amount of encouragement necessary to trigger liability; many cases will likely be decided to resolve the right threshold. The bar must be set high to avoid assigning large damages to innocent actors. One appropriate standard would be to assign damages only if no reasonable person could interpret the actions of the innovator to be in good faith and without redeeming social value; this standard, resembling that of criminal law, would go far towards eliminating the worst chilling effects imposed on software developers.

C. The Difference Between a Liability/Remedy Test and Grokster

As we have seen, elements of the existing standards of *Sony* and *Grokster* can be key components to a proper standard. But as it is currently constructed, the standard of *Grokster* both over- and under-protects innovation. Because it places so much emphasis for its liability on the intent of the developer, it will find many harmless (and even beneficial) programs liable. Less obvious is the standard's risk of underprotection. It is possible for a software program developed by an innocent developer to fail a balance of interests test, even if the developer has no intention of enabling illegal use. Both of these errors are not simply problems of execution where the standards of *Grokster* are valid but simply applied too tightly. They are deep, fundamental problems with the approach of all existing cases concerning innovation regulation.

1. Overprotection – The NES Emulator

As mentioned above, *Grokster* overprotects when it assigns an inappropriate remedy – assigning damages in circumstances where they serve perverse incentives. But *Grokster* also overprotects at the liability

114. Though the court in *Grokster* introduced inducement as a theory of liability, its principles apply to this context as well.

stage, as it prohibits innovations that were intended to break the law but are, on balance, beneficial for society. This includes, for example, technologies which were created for nefarious purposes and can be readily thwarted. An example would be an easily filtered communications program, designed to exchange illegal files, that also allows for the efficient transfer of other types of files. More importantly, it includes technologies where the harm is significant from a legal perspective but negligible from a practical perspective.

Consider a Nintendo Entertainment System ("NES") emulator.[115] This emulator assists individuals who copy digital files of NES cartridges by enabling them to play the games. It also creates considerable value by enabling additional uses of the video games, uses not contemplated by the games creators. How would the *Grokster* standard treat the NES emulator? Suppose the developer openly intended and encouraged the use of the emulator with unlicensed copies of NES games. It seems certain that a court would find the developer liable for contributory infringement – the software enables the play of copyrighted games, clearly contributing to infringement, and the developer's demonstrable intent is certainly enough to constitute knowledge. The *Sony* safe harbor may or may not protect the developer – it would depend on the court's analysis of the legality of an owner of the game's use of the emulator to play a digitized version of the game, and on the court's empirical determination about the percentage of these uses. Vicarious liability likely would not apply, as the developer released the software without any retained control. Ultimately, though, a court would find the developer liable under inducement theory, as the stated intent of the developer was to enable and encourage infringing use.

Innovation policy would answer the question of liability in the reverse. The NES emulator passes both static and dynamic components of a balance of interests test. Its harm to current interests is miniscule. Aged systems such as the original NES have insignificant markets – the video game industry is characterized by a particularly short shelf life, and a decade after a system is released, it is worthless. Furthermore, any additional sales do not contribute anything to the copyright holders, as all transactions take place in used video game stores and through online sales by private individuals. Beyond these limitations, the benefits of the software are considerable. Players of the games no longer need to struggle with old, malfunctioning hardware; players who value the games at less than their purchase price can play and enjoy the games; and all

115. An emulator is a software program that mimics the functionality of a physical console. It can execute the digital code of the original video game file, translating keyboard keys into joystick commands and translating television output into screen output. Wikipedia, Console Emulator, http://en.wikipedia.org/wiki/Console_emulator (last visited Mar. 27, 2007).

players can install the emulators and games on laptops to play while traveling, an activity not possible using the original consoles.[116] The only static harm caused by the emulators is to those who would resell their physical games – but there is significant nostalgic value in the physical games and systems, value which far exceeds the value of the games themselves.[117]

The dynamic balance of interests reinforces this. Potential buyers of systems may be discouraged from making a purchase, knowing that they will eventually be able to play the games through emulators; as a result, sellers may be harmed by the lost volume. But we are considering here only emulators of systems which are long past their prime – say, at least a decade. Few video game fans who contemplate spending $300 or more on a system and $50 on a new game will choose to wait 10 years in order to play the games for free. And overshadowing this marginal harm is the value of letting future developers play with the emulator and the games, creating new levels and modifications and brand new games with ease.[118]

2. Underprotection – PeerProduce

The standard of *Grokster* may be underprotective as applied to developers whose innovations have unintended or undesired harmful uses. There are many general-purpose innovations which have both legitimate and illegitimate potential uses, including software based on encryption, the protection of anonymity (such as Tor), file exchange (such as peer-to-peer file sharing), and DRM circumvention (permitted for reverse engineering for interoperability).[119] Some of these may be created by a developer who has no desire or even suspicion that the device can be used for illegal purposes.

Consider a hypothetical development tool, PeerProduce. PeerProduce is a tool for collaborative, distributed, peer-to-peer software development. It allows amateur programmers to share their repositories of written source code with others, and it enables others to search the network to find pieces matching the description of the software they are looking for. The search is based primarily on programmer-supplied

116. From the perspective of copyright infringement, it can be argued that none of these benefits constitute "non-infringing use" in the sense of Sony. Nevertheless, they are considerable benefits, especially measured against the limitations on the practical harm of the violations.

117. This author, in fact, is proud to own a working original NES system, along with a sizable collection of games.

118. *See, e.g.*, Mega Man vs. Ghosts 'n Goblins, http://www.brokenfunction.com/content/mmvs2/ (last visited Mar. 27, 2007).

119. More discussion is found in Part II.E above; *see also* Electronic Frontier Foundation, *supra* note 64.

descriptions of the source code they provide, but as a fallback, the search program looks at the names and folder paths of files.[120] Based on the strong organizational tendencies of software developers, PeerProduce also includes an auto-indexing feature that can take a folder full of programs and can index the folder and its subfolders to make all of the code available to and easily searchable by others. Unbeknownst to our hypothetical, naïve developer, PeerProduce can be used directly (or with some minor modifications to refine the search process) as a peer-to-peer filesharing program for music and movie files – exactly replicating the functionality of the Grokster system. PeerProduce is released without any filters on the type or contents of files or search requests.

How would the *Grokster* standard respond to PeerProduce? First, consider contributory infringement. PeerProduce certainly contributes materially to direct infringement, as it replicates the functionality of the Grokster software. Whether PeerProduce's developer knows of this assistance is a trickier question, but it is one that does not need to be resolved. This is a classic example of the *Sony* safe harbor, as the technology has substantial noninfringing use. Therefore even if contributory copyright infringement would apply, the developer would be protected by the exemption. Vicarious infringement would also exculpate the developer, who has no ability to control or supervise subsequent use of the software. Inducement liability would also not apply, as the developer had only honest intentions. None of the *Grokster* elements would apply, and continued development and distribution of the technology would be permitted.

Proper innovation policy would decide otherwise. The static balance of interests resembles that of Grokster. PeerProduce enables the exchange of copyrighted music and movie files, which (for the sake of argument) cause considerable harm to the copyright holders' economic interest.[121] It has benefits as well, of course – it greatly lowers costs of collaboration in software development, by making it easy to both offer software to others and to find software offered by others. But there are other options for this which have only marginally higher costs, such as Sourceforge, an enormous repository of open-source software.[122] The

120. For example, a searcher looking for networking software will be able to find a program containing "TCP" located in a subfolder "Net" of a folder "Utils".

121. Of course, there is much debate over this, and one could argue that the static balance of interests is in favor of PeerProduce. But, given that it is essentially identical to Grokster, this is likely not the prevailing attitude. After recognizing that "these fears [may] be offset by the different concern that imposing liability . . . could limit further development of beneficial technologies", the Court said that "[t]he argument for imposing indirect liability in this case is, however, a powerful one, given the number of infringing downloads that occur every day using StreamCast's and Grokster's software." *Grokster*, 545 U.S. at 929.

122. *See* SourceForge.net, http://sourceforge.net/ (last visited Feb. 12, 2007).

existence of these alternatives reduces the value of the software considerably. The dynamic balance of interests is mixed. Prohibiting the software from being distributed in its current form imposes some chilling effects, though far less than the effects of a large damage award.[123] Permitting the software to continue to be distributed, though, leads to greater ongoing harm to protected economic interests. Furthermore, the cost of adding filters to the system (to examine the content of the files to see if it is text/source code, or at the least to prohibit the exchange of files with an MP3 extension) is very minimal – the court can require the developer to add these to the system before it can be legally distributed.

D. Real-World Applications of the Liability/Remedy Standard

1. *Grokster*

The static balance of interests in Grokster is similar to that in PeerProduce. The harms are identical – the software enables (and is in practice used for) the transfer of copyrighted music files. The benefits of Grokster are similar, as it supports a variety of legitimate file transfer operations, including the sharing of music by artists who wish their works to be distributed through peer-to-peer networks, to increase the size of their fan-base or to distribute music that the recording label rejected. This is likely a considerably smaller share of the use of the system than the share of legitimate usage in PeerProduce. Also, as with PeerProduce, there are other options for the legitimate exchange – many artists host websites and make their music available through them – but they are not quite as effective. While the question has not entirely been resolved, it seems likely that Grokster would lose in this balancing.

On the dynamic scale, as with PeerProduce, permitting the continued distribution of the code risks ongoing harm to the copyright interests of music holders. Prohibiting the software carries the same potential chilling effects (though of course the intent test limits these by providing a high, clear standard before assigning large damages). But prohibiting the software has a different practical effect. The intended purpose of PeerProduce (the exchange of program source code) could be realized while avoiding the majority of the harms by adding simple filters for music files. Given that Grokster's primary beneficial purpose is to share music files, effective filters would need to separate authorized from unauthorized transfers, a far more difficult task. Stopping the unauthorized transfers would likely require stopping the authorized as

123. Of course, good innovation policy would not apply damages, as the harm was unintended.

well, a tradeoff that is still likely worth it, though it is a matter of debate.

As for intent and the possible assignment of damages, the Supreme Court noted, in particular, the pieces of evidence indicating that Grokster had tried to absorb as much of the former Napster user base as possible.[124] This might enough to pass some low trigger threshold, but the standard must be stricter than this, given the massive chilling effects of damage awards. Damages are not an appropriate form of remedy, without clear evidence that the developer knowingly designed the software primarily for illegal use.

2. Blizzard

In Blizzard, the harm to the copyright holder is indirect. The BnetD server allows pirated copies of Blizzard games to be played over the Internet. This produces a marginal increase in the value of pirated copies of games, and consequently a greater incentive to copyright games. But this increase is small. Even without BnetD, illegal copies of games can still be played, both offline and with friends over a Local Area Network. Also, BnetD does not share players with Blizzard – the large community of Blizzard players will still be inaccessible to those with pirated copies of games. As another type of harm, the BnetD server will draw game players away from the official Blizzard server, reducing their revenue from advertising. But to this extent, the harm is caused by competition – players with legal copies of games will only switch to BnetD if it represents a better game playing experience.[125] This is not the sort of harm that the legal system wishes to avoid. It is in fact one of the benefits of the BnetD server – it represents a competitor in the market for Blizzard video game servers, and it in fact incorporates a number of improvements.[126] Given the limitations of the harms and the strength of the benefits, a static balance of interests test would come out against regulation of the innovation.

The dynamic balance of interests reinforces this determination. Prohibiting the distribution of the BnetD server would have chilling effects greater than those of Grokster, because the creators of the server likely thought and intended that their work would be protected by the reverse engineering exceptions to the DMCA and to copyright law in general. By interpreting these exceptions narrowly so as to prohibit the

124. *Grokster*, 545 U.S. at 925.

125. Of course, if BnetD is only competitive because source code was taken from Blizzard, then it is the sort of competition that copyright law is designed to shut off. But in the actual case, and for the purposes of this hypothetical situation, questions of actual copyright infringement were not being decided. The legal question is the circumvention of a technological protection measure in violation of the DMCA.

126. Brief of Defendants-Appellants, *supra* note 65, at 4.

server, future developers will be uncertain about the legal status of any future reverse engineering activity, and on some level, uncertain about the scope of other fair use exceptions, such as the exception for educational activities. Permitting the distribution of the server, on the other hand, has considerable beneficial results. Blizzard will be forced to improve the quality of their server in order to retain players. Blizzard also may choose to share the CD-Key checking mechanism with the developers to enable them to add security measures to BnetD to prevent the use of unauthorized games.

The liability balance of interests clearly opposes regulation of the BnetD server; as a result, the question of remedy does not need to be raised.

3. Tor

The effects of Tor are considerable for both harmful and beneficial use. It is hard to weigh the benefits of free speech and privacy against the harms of child pornography and copyright infringement, and the anonymity produced by Tor protects all of these. Consider first whether the designers of Tor can modify their software to reduce the social harms. It is difficult to construct filters that can detect child pornography, but there are ongoing efforts to develop filters that can block simple transfers of copyrighted music files, and Tor does not include any such devices. It is also useful to include a blacklist – computers, identifiable perhaps by their MAC address or some other identifying information, that are not permitted to use the Tor network because they have been determined by some other means to be producers or distributors of illegal material. Given the apparent ease of including such techniques within the software, the burden of proof should lie with the Tor developers to demonstrate that these techniques are technologically unworkable, for example that their inclusion would involve a redesign of the system that would increase its latency or decrease its bandwidth and render it unable to confer its social benefits. In the absence of such a demonstration, proper innovation policy dictates that in its current form it should not be distributed or used.

The remedy rule I offer sets a high threshold for assigning large liability damages to the software's developer. Given the many beneficial uses of the Tor service, the developers must be understood to have had good intentions in producing and distributing their software, and cannot be held liable for damages. To do so would produce too many chilling effects for other software engineers who seek to promote free speech and privacy values through their tools.

4. Google Book Search

The static balance compares the direct and indirect harm to copyright owners to the benefits to consumers of the service. As Google retains a few digital copies of copyrighted works without permission, this is a clear, direct, but bounded (and small) harm. Portions of this digital copy are transmitted to consumers in search results, though Google restricts the display of this digital copy so that the amount transmitted to others is of an amount generally considered fair use.[127] Another harm is the risk that Google may leave the database insufficiently secured, enabling massive copyright violations.[128] Weighted against these harms are the benefits the service offers. For years, services such as LexisNexis have enabled scholars to search through the text of journal articles, making research considerably easier. Extending this capability to entire books will produce enormous additional benefits, sufficient to outweigh the limited and speculative harms of the service.

Many commentators have stated that, despite its size and available cash, requiring Google to gain any form of permission from every copyright owner would be prohibitively difficult.[129] As a result, requiring Google to abate the harms by requesting permission to copy the books for its own purposes would likely cause Google to abandon its efforts.[130] Though the burden of proof would lie with Google to make this demonstration, it is almost certain that it could be met, as the number of copyrighted (and orphaned) works makes this task impossible. This is not like the Tor example above – the harms and the benefits are inextricably linked, and must be taken together. And, given the relatively minor harms, the balance of equities strongly favors permitting the service to operate as is.

Given that the liability balance argues against prohibiting Google Book Search, the remedy rule need not be applied – the developers cannot be held liable for distributing a legal product.

---

127. But, of course, fair use is a multifactor test, and it is unclear whether Google Book Search is fair use. *See supra* note 88.

128. Paul Aiken, Authors Guild, Speaker at the Yale Information Society Project Conference: Regulating Search? A Symposium on Search Engines, Law, and Public Policy (Dec. 3, 2005) (one of the plaintiffs who brought suit against Google, Paul Aiken raised this point while speaking).

129. DeLong, *supra* note 89.

130. This speaks to the static balance – it's a loss of the current benefits. But in some previous examples, the decision to restrict an innovation is less harmful when examining the dynamic balance because there are simple potential modifications to avoid the harms. *See* i*nfra* Part III.D.3.

IV. CRITICISMS AND ALTERNATIVES

A. Workability

Achieving an optimal dynamic balance of interests is difficult, and creating a policy based on more than ad-hoc decision making is even more so. Many might criticize the policy proposal I have offered by saying it does not create a workable standard for courts to follow. And if the only suggestion I offered to the court was that it should look at a dynamic balance of interests instead of a static balance, this would be a legitimate concern. Courts would select a wide variety of factors to consider when crafting a dynamic balance.

But my proposal offers far more structure than that. Separating the question of liability from the question of remedy, and separating an analysis of the value of the technology from the behavior of the developer, enables courts to convert one very difficult question into two questions that are very similar to the questions of copyright law. The second question, the question of remedy, is the easier of the two. It examines the motive of the developer, distinguishing the developer whose intent was to commit infringement from the developer whose intent was innocent. This is essentially the inducement test of *Grokster* and of patent law – it is not an easy determination, but it is familiar to courts. The question of liability is somewhat more difficult, and my proposal does increase the complexity beyond that of current law, but it remains quite manageable. At its core, the balance of interests is derived from *Sony* – if the innovation has substantial beneficial (or non-infringing, in the words of the Court in *Sony*) use, then it should be permitted. This is no less workable than current law, as it is already part of the determination process. My policy proposal adds considerations of specific, reasonably foreseeable repercussions of the decision. These questions place most of the burden on the parties, who must demonstrate the repercussions of an adverse decision, ideally through expert testimony from technology professionals. Resolving such conflicts of expert opinions falls well within the bounds of ordinary judicial processes.

The policy proposal I offer cuts across existing legal systems, most of which are directed to the behavior of an actor and not to the virtues and vices of a device. As a result, it is not possible to simply adopt my approach once and for all. After all, there is no doctrine of innovation law in which to operate. This paper has primarily dealt with secondary liability for copyright infringement because in recent years this has been the active area of law. But software innovation is also heavily regulated by the Digital Millennium Copyright Act and by private contracts (particularly in the form of license agreements), as Blizzard v. BnetD

demonstrates. Software innovation policy applies whenever a software developer is brought into court for the mere creation and distribution of an innovation. The positive and negative uses of the software and the repercussions of prohibiting or permitting the software are still the key factors in the balance of interests, whether the illegal activity is measured by damage to intellectual property interests or by the violation of contract terms or by any other harm. Moreover, the dual separations of liability from remedy, and the technology from the activity of the developer, are still the right policy approach, as they help produce the optimal dynamic balance of interests and avoid peripheral chilling effects on innovation.

Perhaps proper software innovation policy will need to be integrated into existing legal doctrines over time. Or perhaps it will require legislative action, an affirmative Congressional action to protect software innovation intended to cut across other disciplines. But at the very least, judges and legislatures can consider the principles I offer as they craft legal standards across the board. They can be more cognizant of the dangers that some legal systems pose to innovation. They can also adopt separate elements of my proposal to provide some amount of support for innovation. For example, a court could apply its own liability standard, but limit awards of damages to cases where the developer demonstrably intended the innovation to be used to violate the law. In this case, the court's decision would at least avoid creating chilling effects to discourage other well-intentioned innovators.

B. Other Solutions

Many critics will reply that any solution must operate within an existing legal doctrine, and that the language of the existing statutes and broader readings of existing principles must support any policy proposals. Given that existing principles are almost universally based on static balances of interests, and that innovation policy truly cuts across legal boundaries, these limited approaches are simply not sufficient to fully protect innovation.

Much cyberlaw scholarship in recent years has focused on increasingly restrictive interpretations of intellectual property law. The stated objective of patent and copyright law is given in Article 1, Section 8 of the U.S. Constitution, in a line known by heart to many IP scholars: "The Congress shall have the power. . . To promote the progress of science and useful arts, by securing for limited times to authors and inventors the exclusive right to their respective writings and discoveries . . . ." In the current legal climate, intellectual property law and policy are shifting towards "exclusive right" and away from "progress," treating

intellectual and digital property more and more like real property.[131] The rights of copyright holders in particular have been extended in recent years by both courts and legislatures.[132] The law contains many exceptions to these rights, such as the fair use doctrine.[133] But fair use falls far short of converting a property regime into an engine for innovation and progress. By centering the legal discussion around commercial interests, IP law ultimately fails to protect innovation.

Creative destruction is one of few theories that avoids this focus on commercial interests; in fact, it argues that society can sometimes be improved through damage to commercial interests.[134] In particular, proponents of creative destruction in cyberlaw see the traditional methods of production and distribution of cultural materials as outdated and no longer necessary.[135] Many have proposed replacing copyright law (whose purpose is to protect these outdated methods) entirely with alternate compensation methods.[136] While it might, in the long run, be efficient for society to replace copyright law (at least in the context of musical works) with an entirely different system, innovation policy must operate at a more fine-tuned level than complete regime change. Innovation policy must correctly and specifically identify which innovations are on balance beneficial and which are harmful, rather than advocating the total overthrow of existing conceptions of legal harm.

Legal scholarship also uses the First Amendment as a defense

---

131. *See* Dan Hunter, *Culture War*, 83 TEX. L. REV. 1105 (2005); Peter S. Menell, *Envisioning Copyright Law's Digital Future*, 46 N.Y.L. SCH. L. REV. 63 (2002); Brian F. Fitzgerald, *Digital Property: The Ultimate Boundary?*, 7 ROGER WILLIAMS U. L. REV. 47 (2001). For a considerably older (but still accurate) discussion, *see* L. RAY PATTERSON & STANLEY W. LINDBERG, THE NATURE OF COPYRIGHT 213 (1991). Many have studied this transition and have offered explanations and criticisms. *See, e.g.*, Mark A. Lemley, *Property, Intellectual Property, and Free Riding*, 83 TEX. L. REV. 1031, 1031-32 (2005) (interpreting the increasing propertization of copyright as a transition to a state in which copyright owners internalize all of the social value of their intellectual property); Hannibal Travis, *Pirates of the Information Infrastructure: Blackstonian Copyright and the First Amendment*, 15 BERKELEY TECH. L.J. 777 (2000).

132. For a legislative example, consider the Sonny Bono Copyright Term Extension Act, Pub. L. No. 105-298, 112 Stat. 2827 (1998) (codified as amendments to 17 U.S.C. §§ 108, 203, 301-04). Judicial action to increase copyright holders' rights has mostly taken the form of increasing grounds of liability for infringement, such as *Grokster*'s addition of inducement liability.

133. 17 U.S.C. § 107 (2000) (fair use limitation on exclusive rights in copyright).

134. *See* Ku, *supra* note 112, at 268-69 (adapting to cyber law, Schumpeter's notion of "creative destruction," in which capitalism progresses not through minor adjustments in efficiency or variety of production capabilities but through fundamental changes in economic models underlying the production).

135. *Id.* at 269 ("[D]igital technology and the Internet strike at the foundation of copyright and the industries built upon copyright by eliminating the need for firms to distribute copyrighted works and for exclusive property rights to support creation.").

136. *See, e.g.*, *id.* at 311-22; WILLIAM W. FISHER III, PROMISES TO KEEP: TECHNOLOGY, LAW, AND THE FUTURE OF ENTERTAINMENT (2004).

against excessive legal regulation of technology innovation and use.[137] One can rationalize the application of the freedom of speech either to expressive uses of innovations or to the expression inherent in the lines of code of tools.[138] To determine whether or not a restriction on innovation is permissible, a court could apply a variant of First Amendment doctrine to the law.[139] A court might, for example, ask whether the law is narrowly tailored to achieve a legitimate government purpose. It would examine the purpose of the law and the way in which the law was constructed, but it would not ask whether the innovation being restricted is valuable enough to be worth protecting, and it would not attempt to measure the amount of harm caused by the innovation. It would never examine the balance of value against harm. In fact, First Amendment doctrine is specifically constructed so as not to make judgments on the activity being regulated,[140] and therefore cannot serve as a guide towards proper innovation policy.

Another interpretation of the value of the First Amendment is directed less towards the speech produced and more towards the identity of the speaker. In particular, the promotion of individual speech enables the speaker to participate in democratic self-governance,[141] and promotes a democratic culture.[142] Jack Balkin goes so far as to put forth "democratic control in technological design" as one of the core values involved in the freedom of speech in the modern era.[143] This modernized conception of the freedom of speech is necessary to promote "interactivity, mass participation, and the ability to modify and transform

137. *See, e.g.*, 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 (1999); Jed Rubenfeld, *The Freedom of Imagination: Copyright's Constitutionality*, 112 YALE L.J. 1 (2002); *but see* David McGowan, *Why the First Amendment Cannot Dictate Copyright Policy*, 65 U. PITT. L. REV. 281, 284 (2004) ("The First Amendment does not supply a premise a court can use to limit congressional power to give authors rights to exclude others from their works, nor to give others—including other authors—the right to use their works.").

138. *See, e.g.*, Bernstein v. U. S. Dep't of State, 922 F. Supp. 1426, 1436 (N.D. Cal. 1996) (acknowledging that in some circumstances software code is protected speech).

139. Neil Weinstock Netanel, *Locating Copyright within the First Amendment Skein*, 54 STAN. L. REV. 1, 21-23 (2001) (proposing treating all of copyright law as a content-neutral or content-based restriction of speech, and applying First Amendment doctrine appropriately); *but see* McGowan, *supra* note 137.

140. *See* Brandenburg v. Ohio, 395 U.S. 444, 449 (1969) (overturning the clear and present danger test as used in *Whitney*, which permitted the regulation of speech which in its substance advocated violence, and establishing modern First Amendment law as neutral to the substance of speech unless its context indicates that it will result in imminent violence).

141. Outside the context of digital culture, these ideas are associated with Meiklejohn. *See* ALEXANDER MEIKLEJOHN, FREE SPEECH AND ITS RELATION TO SELF-GOVERNMENT (1948).

142. *See* Balkin, *supra* note 27.

143. *Id.* at 52.

culture."[144] In the context of innovation, a democratic culture enforces a balance of power between the production industry and the individual. If users are afraid to fully use and experiment with their technology they become trapped in the role of technology consumer. This is the "passivity thesis" described in the context of copyrighted works by Michael McGowan.[145] These are interesting as cultural theories, but they serve only to offer additional rhetorical support for the statement that courts should generally disfavor assigning liability to amateur software developers. They are not capable of providing innovation policy.

None of these approaches can offer anything resembling the breadth of the proposal given in this paper. They are not comprehensive enough to protect innovation against all legal restrictions, and they are not thorough enough to consider all of the benefits and harms of innovation and the concerns of regulation. Software innovation needs and deserves a stand-alone, comprehensive policy, one that can guide judges and legislators when considering all types of legal harm.

VI. CONCLUSION

The free and open climate of technology innovation which produced the computing industry as we know it is under attack by a legal system too concerned with short-term damage to intellectual property and other corporate interests. The consequences of the actions of courts and legislatures to regulate innovation are harmful to future societies in ways that are not always obvious at first glance. The rhetoric of "piracy" and "property" sometimes drowns out all other voices. Attempts to bolster the defense of innovation by expanding exceptions to intellectual property laws or by applying some other legal regime continue to fall short. Without a clearer understanding of the dangers of restricting innovation, and without a better idea of how to structure the legal system to protect innovation without throwing all existing legal interests out the window, courts and legislators will continue to tighten the bonds on software developers.

But innovation can yet win this war. This paper proposes a stand-alone software innovation policy, a policy that protects innovation and produces the proper incentives for other actors. This balance is not hard to achieve. It can be accomplished by separating the regulation of innovation into two questions, one of liability and one of remedy. Proper policy separates the developer from the innovation, examining only the benefits and harms of the innovation when determining liability, and

144. *Id.* at 6.

145. McGowan, *supra* note 137, at 289, 323-27 (criticizing the use of this thesis to defend against copyright control of activity).

only the intent of the developer when designing the appropriate remedy. By regulating innovation this way, society can reach an optimal dynamic balance of interests, one that respects existing legal interests, discourages true bad actors, and encourages valuable innovation.