LESS THAN PERFECT
-- Saving Experimental Use Exception for Software Patents

Sheng Wu*

“While we protect real and personal property to protect the owner from harm and give the owner an incentive, we protect intellectual property only to ensure that we create a sufficient incentive to produce it. ‘Sufficient incentive,’ however, is something less than ‘perfect control.’”
-- Lawrence Lessig[1]

INTRODUCTION

The Constitution mandates protection of intellectual creations with the goal of promoting “Science and useful Arts.”[2] Copyright and patent laws are promulgated to implement those mandates and each carries different scope and length of protection, in their respective effort to promote different types of intellectual activities and resulting social progresses. Copyright is generally addressed to artistic or aesthetic works, while patent law generally addresses industrial or technological inventions. Technological advances, however, challenges the traditional notion that these two laws protect different and distinct aspects of intellectual creations, because “novel expressive technology” like software defies clear definition under either copyright or patent.[3] To fit software into the Procrustean bed of existing intellectual property regime, courts often have to stretch both copyright and patent laws to deal with issues as they emerge. The resulting ad hoc convergence of copyright and patent protection on software is nothing but problematic, because these two laws embody different policy preferences and harbor different latitude toward the right to access to the underlying knowledge. Although what is protected in copyright is usually not protected in patent, and vice versa, the past two decades witnessed software owner’s repeated efforts to invoke both copyright and patent protection to stop a competitor’s studying of application programming interfaces (APIs)[4] for reverse-engineering purposes.[5] Courts were quick to dismiss copyright claims by holding that reverse engineering led to flourishing compatible programs and was considered fair use under copyright law.[6] With the Patent and Trademark Office (PTO) opening up doors for software patents, software owners are now pursuing patent suits against competitors for the exactly same activities.
1999, Sony charged Connectix for copyright infringement because Connectix reverse engineered Sony’s PlayStation program and developed a competing program that allows Sony PlayStation game owners to play games on computers (Sony’s PlayStation, on the other hand, is played on television). Immediately after losing its copyright infringement case in the Ninth Circuit, Sony filed for patent infringement suit.

Software owners’ manifested dexterity in invoking different intellectual property laws to suppress the same competing activity raises an interesting question: if copyright law embraces the promoted social progress in expression through reverse engineering, should patent law be modified to accommodate such public interests, namely, to what extent does the law recognize an activity as a permissible fair use under copyright law, yet still as an infringing use under patent law?

This Note will examine the appropriateness, or inappropriateness in this matter, of a stringent patent protection over software against reverse engineering to achieve interoperability. Part I sets the stage by introducing the legal status of reverse engineering under copyright and patent laws, and concludes that as convergence of copyright and patent protection leads to a perfect control over software, a dogmatic application of existing patent law thwarts the very social progress copyright and patent laws are designed to promote. Among the few limited defenses in patent law, experimental use exception is a perfect candidate to relax this perfect control, as it functions as a close counterpart to fair use in copyright law by excusing certain activities that promotes social progress in technology. Part II examines the policy considerations underlying the rise and fall of experimental use exception and demonstrates why experimental use exception should be preserved, or revived (depending on one’s view toward the current status of experimental use), for the software industry in light of those policy considerations. Part III aims to clear obstacles for the statutory recognition of experimental use exception by looking into the statutory creation of experimental use exception for pharmaceutical industry in the 1980s, and briefly addressing the constraints Congress may face in complying with our international treaty obligations.
I. ASYMMETRICAL PROTECTION OF EXPRESSION AND FUNCTION

A. The Case against a Perfect Control

Both the United States Constitution and judicial decisions profess that intellectual property is fundamentally about incentives to invent and create. The Constitution expressly conditions the grant of rights in the Patent and Copyright Clause on the end to “promote the Progress of Science and Useful Arts.”[19] Similarly, the Supreme Court held that copyright and patent laws “[make] reward to the owner a secondary consideration,” because the Patent and Copyright Clause was premised on the “conviction that [granting these rights] is the best way advance public welfare through the talents of authors and inventors in ‘Science and useful Arts.’”[10]

Under this incentive theory, intellectual property laws are created to provide incentives to creators because information is what economists call a public good: it may be consumed by many people without depletion, and it is difficult for the creator to identify those who will not pay and prevent them from using the information. Other prominent examples of public goods are lighthouses and national defenses. For lighthouses and national defenses, the government prevents the underproduction that would result from leaving it to the market by paying for them itself. For information, the government chooses another path: by granting intellectual property rights, it hopes that the control over the use and distribution of the ideas may encourage authors and inventors to invest sufficiently in intellectual creations. A social cost comes with this approach is the limitation on the diffusion of those intellectual creations. The grant of exclusive rights gives intellectual property owners the ability to raise the price of the work above the marginal cost of producing it, which in turn means fewer people will buy the work than if it were distributed on a competitive basis. Since the primary goal is to promote social progress, intellectual property laws should encourage creation and dissemination of new works that offset, but not perpetuate, those costs. Consequently, both patents and copyrights do not give rightholders perfect control over their works. Both copyright and patent are limited in duration and in scope. After a patent or copyright expires, improvers/competitors can utilize freely
materials previously covered by the patent or copyright. A copyright owner can prevent copying, adaptation, distribution, and performance, but cannot prevent private use of work or using the ideas or facts in the work by others. The free access to ideas and facts preserves a viable public domain for the social advance in expressions. Similarly, patent law grants the patentee the right to exclude others from making, using, or selling the claimed invention, but limits the scope of that right to what is set forth in the patent claims and equivalents thereof. Such limitations encourage improvement on the patented technology with the promise that sufficient improvement on the existing invention can remove the accused infringer from the periphery of infringement.

In summary, intellectual property laws, as they are designed, permit others to skirt the edges of the granted rights; hence they offer sufficient control to provide incentives to invent, but not a perfect control to stifle social progress.

B. The Advent of a Perfect Control on Software

Courts seemingly have gone through a nauseous time while navigating through the sea change in technology in the past few decades. Software technology differs from traditional technology in that it is protected by both copyright and patent laws, and is “characterized by rapid sequential innovation, reuse and re-combination of components, and strong network effects that privilege interoperable components and products.”[11] Technological progress that is driven and defined by new forces entails the necessity to re-assess the proper balancing between incentive and progress. However, because our case law system is backward-looking and precedent-bound, courts solve problems as they come along under established legal framework, but do not take their own initiative to learn about social ramifications of those dated rules on new technologies, particularly when those ramifications are not manifest in the case before them. As courts’ reaction to the convergence of copyright and patent protections over software suggests, a perfect control over software is looming as a result of this ad hoc approach to accommodate new challenges under existing rules.
Software owners brought the first flurry of cases against reverse engineering of a software product under copyright law. To reverse engineer, a software engineer has to run the program to either observe the program in operation or perform some examination of the individual computer instructions the program gives while running. A legally recognized “copy” of the original program is made when the program is read into the computer’s random access memory or RAM. Depending on the number of times the examination to be performed, additional “copies” are made every time the computer is booted. Consequently, the original flurry of cases brought under copyright law succeeded to the extent that the prima facie case of infringement was successfully established.

However, it is not difficult for courts to perceive the over-stretched legal concept of copyright infringement in such cases: unlike copying a book or a movie for sale, copying the program into the RAM for the purpose of making a compatible program is not the normal “commercial use” of the copyrighted material, because the copyrighted program is not sold nor is there any infringing final derivative work. More importantly, as the Ninth Circuit saw it, there is certain unfairness in denying such copying, because it “led to an increase in the number of independently designed [programs]. It is precisely this growth in creative expression … that Copyright Act was intended to promote.”

The uneasiness of fitting such an unconventional “intermediate copying” into traditional copyright infringement analysis thus prompts the court to look beyond copyright law. By characterizing reverse engineering as an effort to ascertain the ideas and functional aspects of the program, which are not protected by copyright, the Ninth Circuit found, at least it thought it did, a novel solution to exempt intermediate copying: disassembly for the purpose of studying or examining unprotected aspects of a copyrighted program, if necessary, constitutes fair use. As in the court’s own words, “[t]o the extent that a work is functional or factual, it may be copied, as may those expressive elements of the work that must necessarily be used as incident to expression of the underlying ideas, functional concepts, or facts.”
Still, to find such a copying as fair use, the court needed to leap over the major hurdle of statutory factors of fair use. The foremost important factors for fair use analysis are the nature of copying and the effects on copyright owner’s market. Notwithstanding the presumption of unfairness for copying with commercial purposes, the court expressed a willingness to look into the specific characteristics of a particular commercial use to overcome that presumption. Notably, it emphasized that “[c]ourts are free to consider public benefit resulting from particular use notwithstanding fact that alleged copyright infringer may gain commercially… Public benefit resulting from particular use of copyrighted work for finding of fair use need not be direct or tangible, but may arise because challenged use serves a public interest.”[19] Since reverse-engineering led to the creation of more expressive works and accrued public benefits, the presumption of unfairness could be overcome. Furthermore, the fundamental purpose of the Copyright Act is, as the court explained, to “encourage production of original works by protecting expressive elements of those works while leaving the ideas, facts and functional aspects in the public domain for others to build on.”[20]

What is unique about the Ninth Circuit’s approach is that it did not excuse intermediate copying because it promotes the exact social progress in expression that copyright law is designed to promote; rather it held that intermediating copying accompanying the learning of unprotected functional aspects or ideas is fair use because the copyright owner cannot lock in the functional aspects by copyright. A corollary of this reasoning is that if software owners can obtain patent over some of the functional aspects, intermediate copying is no longer fair.

It is not long before software owners picked up this point and launched the second battle against intermediate copying.[21] After Sony lost its copyright infringement case, it filed a patent infringement against Connectix for the exact same activity on the same program, namely reverse engineering.[22] Compared with copyright, rights granted to a patentee are extremely broad: a patentee can exclude others from making, using, selling, offering to sell, or importing a patented product.[23] Decompiling software programs constitutes a patent infringement because the running of the program amounts to a “use”, at least when the program is claimed as an apparatus.
In addition, decompiling also involves the “making” a copy of the patented software when the program is read into the RAM, despite the temporary nature of the instantiation of the program.\textsuperscript{[24]} A commentator argues that since patent law lacks a fixation requirement, even near-instantaneous duplication of patented software is a prohibited “making” of the software.\textsuperscript{[25]} The prima facie case of patent infringement for decompilation, or intermediate copying in copyright terms, is easily established.

Patent protection, like copyright, is not absolute either. Since knowledge is cumulative,\textsuperscript{[26]} from economic efficiency point of view, patent law discourages repetitive investment on the same technology by granting some exclusive rights to the one that first invents and diligently discloses his invention to the public. At the same time patent law fulfills its constitutional obligation to promote social progress by protecting or exempting certain uses of the patented invention if it is for research purposes or the resulting improvement is significant or radical, generally known as experimental use exemption, “blocking patents”\textsuperscript{[27]} and “the reverse doctrine of equivalents.”\textsuperscript{[28]} An accused infringer who are doing research/experiments or made significant or radical improvements is thus in a position recognized by law to bargain with the original patent owner to use the invention or be free to use the invention without payment.\textsuperscript{[29]}

One plus one usually equals two. However, the convergence of copyright and patent protection on software produces an effect more than the mere combination of protection of expression and functions. It expands the protection of expression and function in a way that its owner can assert a perfect control over the intellectual creation at the cost of a shrinking public domain.

First of all, there is no longer a “fair use” defense under copyright law for intermediating copying or decompilation. Notably, the Ninth Circuit did not say in Sega that intermediate copying can be considered “fair use” because it promotes social progress in expression, although that is a fact unequivocally recognized by the court.\textsuperscript{[30]} Rather, it held intermediate copying a “fair use” because such copying is necessary in ascertaining the unprotected functional aspects of
the program. Once the functions or ideas are protected by patent law, there is no “fairness” in copying the expressions, at least under the Ninth Circuit’s reasoning.

Second, competitors successfully made a significant or radical improvement on a software patent no longer enjoy the bargaining power or immunity guaranteed to them by patent law. A software owner can resort to copyright to hold them liable. Copyright law offers little or no protection to improvers, as demonstrated by the Ninth Circuit’s reluctance to excuse intermediate copying on the basis that it promotes expression. In fact, the exclusive rights granted to a copyright owner in section 106 extend to cover any “copy” or adaptation or alteration of the original that is nonetheless “substantially similar” to the original work. An improver cannot claim copyright to his own original contribution to the underlying copyrighted work, although he can get a patent over his improvement on the original invention. As a result, a competitor who significantly or radically improves a software patent, which is exactly protected and encouraged under patent law, cannot escape liability under copyright law.

Finally, on a more delicate level, patenting software implicates restrictions on free speech. Patent law as currently constituted contains no analogy to the abstraction/ filtration test, which is frequently employed to separate function from expression in copyright cases. In the past, PTO did work hard to distinguish patentable subject maters from copyrightable ones, but ultimately failed. In fact, such separation is impossible in software because the expression is the function. Case law indicates that patent law treats software as a densely programmed artifact: every time a computer is run by a program, it turns into a new machine. Software codes, which are expressive instructions, command the arrangement of voltages in computational registers; they will be executed by the machine automatically and involuntarily once compiled into proper format; they are part of the machine that executes the task. The expressions or instructions to a computer by software are therefore the machine itself, a machine built of text. Since patent law considers what it protects is a machine, it does not care about the “printed lines or characters, useful and intelligible only to the human mind” that are essential to treating software as a literary work under copyright. As a result, if a copyright owner cannot
prevent copying the ideas because of First Amendment constraints, patent will come in handy to prevent others from “making the machine.”

The advent of perfect control over software echoes an apocalyptic warning made two decades ago by a philosopher, “[t]he danger is not of an electronic nightmare, but of human error. It is not computers but policy that threatens freedom.”[38]

II. RELAXING THE PERFECT CONTROL: EXPERIMENTAL USE DEFENSE

Unless patent law is going to recognize certain privileges of unlicensed use without harming the patentee’s incentives, a perfect control over software inevitably thwart the exact social progress intellectual property law is designed to promote. Because compatible software programs usually improves on expression, rather on function, blocking patents and reverse doctrine of equivalents are not helpful in saving the proper balance between private and public rights over intellectual creation. This leaves experimental use exception as the perfect candidate to fulfill this task.[39] As articulated by Justice Story, this doctrine predicates the finding of infringement upon a showing that the making is “with an intent to use for profit and not for the mere purpose of philosophical experiment, or to ascertain the verity and exactness of the specification.”[40] The “no intent to use for profit” prong mirrors the “commercial nature of use” factor under fair use in copyright, evincing some tolerance from courts toward certain uses that do not deprive the owner of lawful reward on his discovery. The “experiment” prong also contains fairness considerations. As the experimental use defense was first formulated in early 19th century, patent applications were filed with the so-called “Commissioners for the promotion of Useful Arts,” composed of the Secretary of State, the Secretary of the Department of War, and the Attorney General, any of two of whom could grant a patent, but none of whom must be technology-savvy.[41] Issued patents suffered from systematic flaws because they did not go through a stringent examination process as they should have or as other patents do nowadays. Experimenting for the purpose of ascertaining the verity and exactness of a patent functions as a self-policing mechanism to weed out systematic flaws in the examination proceedings.
Furthermore, mere research can also be fairly exempted because such activities promote progress in “useful Arts,” especially where such activities are not conducted with an intent to profit.

A point needs mentioning upfront is that case law that established this doctrine does not require that the activity to be both non-for-profit and experiment. Sawin established that infringement must be (1) the making with an intent to use for profit and (2) not for experimental purposes, or to ascertain the verity and exactness of specifications. Even if one experiments with an intent to profit, the experimental nature should suffice to remove the accused infringer from liabilities. Similarly, no intent to profit also cuts against a finding of infringement. In fact, the accused infringer in Sawin sold the machine to satisfy a judgment debt: it is neither philosophical nor was it in any way investigation into the patent disclosure. Later courts, however, flipped the test around: to be excused as an experimental use, an act must be conducted with no intent to profit and only for the purposes of experimenting. Experimental use exception thus has significantly been limited in its scope to the extent that the accused infringer has to show both experimental nature of the use as well as not-for-profit-intent of the use.

A. Reverse Engineering as a Permitted Experiment

A compelling reason to create experimental use defense in the first place is that systematic flaws abounded in the patent system at that time. Without some rights to test on the validity of the patented inventions, competitors would be coerced to pay a license fee for something that did not deserve patent protection in the first place. Indeed, absent some systematic flaws, reverse-engineer a patented invention would not have occurred because a patentee is required to disclose the best mode to make the machine in an enabling manner. After all, this is what the patent system is all about: a patentee can get a limited monopoly over a novel and nonobvious invention if he teaches the public how to make it. Modern patent prosecution, where the PTO examines a would-be patentee’s application for the satisfaction of different requirements of novelty, utility, non-obvious and disclosure, makes sure that what is patented deserves the limited monopoly and public can access the invention through the enabling disclosure.
The prosecution of software patents, however, is quite a different story. Software patents have a convoluted history and went through a roller-coaster ride before being recognized as a patentable subject matter.\cite{47} The prolonged debate on eligibility obscured the real pressing issues, i.e., the actual patenting of software programs and how the patent system should treat them to avoid overreaching.\cite{48} For example, as software innovations exist in the source code of commercial products and services, which are not available to customers and hard to catalog or search for ideas, PTO had a hard time in identifying, cataloging, and searching for software prior art, which, in turn, made the determination of novelty and nonobviousness very questionable. In addition, traditional protection of software by copyright obfuscated the needs to have competent examiners specialized on this technology in the Patent Office.\cite{49} As a result, practitioners find software patent prosecutions relatively easy and office actions for software patent prosecution significantly less than those for traditional technologies.\cite{50}

PTO may have let in some undeserving programs because of the difficulty in cataloging prior arts, the judiciary, on the other hand, exacerbated the situation by relaxing the required level of disclosure. A series of recent Federal Circuit opinion indeed indicate that it does not require patent applicants to disclose the invention at all: the applicant need not disclose source or object code, flowcharts, or detailed descriptions of the patented program;\cite{51} rather, high-level functional description is sufficient to satisfy both the enablement and best mode doctrines.\cite{52} The relaxed enablement requirement provides no guidance whatsoever on how the software should be written; it is simply unrealistic to think that one of ordinary skill in the programming field can necessarily reconstruct a computer program given no more than the purpose the program is to perform.\cite{53} Indeed, one may fairly conclude that this peculiar direction in the software enablement cases has effectively nullified the disclosure obligation in software cases.

Therefore, a software engineer who reverse engineered a patent program to ascertain the verity and exactness of the specifications to sort out unpatentable elements and to develop methods to design around the patented elements should be immunized from infringing liabilities,
because such activities promote the fundamental patent policies of disclosure and enablement, and secure to the public what a patentee should have disclosed in the first place. Moreover, the application of experimental use defense to such activities preserves the integrity of the patent system by ferreting out inoperable patents and ensures that patents will not be leveraged to protect unprotectable components.

B. Not Intent to Use for Profit

Justice Story’s original formulation of experimental use exception envisioned an independently permissible use that does not deprive the owner of lawful rewards of his discovery, i.e., a use not intended for profit. Later courts, however, turned this “not-intended-for-profit” test into the centerpiece of experimental use analysis and proscribed an experimental use that is conducted commercially. Later court further expanded on the concept of commercial use and found that any use that affects the pecuniary interest of the patentee is considered an infringing use. As the doctrine stands as of today, practically any use of a patented invention that may be of value to the experimenter or researcher is considered impermissible use. Since reverse engineering involves the use of a patented invention to develop a competing product that serves the competitor’s interest as well public’s interests in having a viable public domain, it is not difficult to see that such activities will fall under this prong of the current experimental use test.

1. Evolution of No-Intent-to-Profit Test

As originally construed, “intent to use for profit” simply means the intent to commercially exploit the patented invention. Later commentators, however, found this interpretation troublesome: if the intermediate use of the underlying patented invention can be excused as experimental use, a competitor can come up with a noninfringing final product without investing in the discovery of the underlying technology or paying the patentee a license fee for using it. The competitor can thus price his own product at lower price and divert the patentee’s consumers. Inventors, as a result, are discouraged from investing in future invention, as it is highly likely he cannot recapture his investment. This is exactly the “public good”
problem intellectual property laws are designed to address. Beginning in early 20th century, commentators started to articulate a different theory about non-infringing use. Under this theory, a patentee’s interests are represented by the compensations he does or might receive from the practice of the invention by himself or others. Infringing acts are those acts attacking the patentee to those compensations, or turning those compensations into other channels or preventing them from accruing in favor of the patentee. Where the products of the experiments are sold, used for the convenience of the experimenter, or if the experiments are conducted with a view to adapt the invention to the experimenter’s business, the acts of making or using antagonize the interests of the patentee and hence constitute infringement.\textsuperscript{[58]} As this theory focuses on the impact of a defendant’s use on the pecuniary interest of the patentee, and one can always pay a licensee fee to use a patented invention, any unlicensed use constitutes infringement because it “antagonizes” some pecuniary interests of the patentee by avoiding the payment.\textsuperscript{[59]}

Courts were quick to adopt the antagonizing theory. Although the Court of Claims had found that the “use for profit” element was typically absent in government use cases and consistently exempted many of the U.S. government’s unlicensed uses of patented inventions in the early part of the 20th century, it started to align with “antagonizing” school and significantly narrowed the scope of permissible experimental use in the 1970s.\textsuperscript{[60]} In two groundbreaking decisions, the court applied the “antagonizing” theory, refusing to find existence of experimental use, because the government’s tests and experiment were in keeping with its own “legitimate business.”\textsuperscript{[61]} Even though no profit motive was attached to the government’s use of the invention, the activity constituted an infringement because it worked against the patentee’s interest. To find non-infringing use, the court asked whether the accused infringer’s use was of a nature that was intended by the patent and whether it was conducted for his legitimate interests. Underlying this inquiry was the court’s conviction that if the accused infringer were to use the invention to further his own legitimate interests, he should have paid a license fee to the patentee. By shifting the inquiry to the link between the use and the legitimate interests of a business or organization, experimental use exception practically received its death toll, because even pure
research or experiments conducted by research institutions for non profit purposes are within the legitimate business of such institutions.\textsuperscript{[62]} In Court of Claims’ own words, experimental uses that do not antagonize a patentee’s interest are the uses for “amusement, to satisfy idle curiosity, or for strictly philosophical inquiry.”\textsuperscript{[63]}

The current court overseeing development in patent law, the Federal Circuit, acknowledged that it is bound by the Court of Claims precedents.\textsuperscript{[64]} In Roche Products, Inc. v. Bolar Pharmaceutical Co.,\textsuperscript{[65]} it asseverated that experimental use was “truly narrow” and only covered situations where the use was “for amusement, to satisfy idle curiosity, or for strictly philosophical inquiry.”\textsuperscript{[66]} There, Bolar wanted to market a generic version of Roche’s patented anti-anxiety drug flurzaepam upon the expiration of the patent term. In anticipating the coming expiration date, Bolar imported quantities of the drug for use in carrying out bioequivalency and stability studies that ultimately would be required for FDA marketing approval. This “experimental” use is found to be “solely for business reasons and not for amusement, to satisfy idle curiosity, or for strictly philosophical inquiry.” Where the experiment is conducted “with a view to the adaptation of the patented invention to the experimenter’s business,” it is a violation of the rights of the patentee to exclude others from using his patented invention.\textsuperscript{[67]}

This decision, however, is quickly overturned by Congress. Drug Price Competition and Patent Term Restoration Act of 1984 (Hatch-Waxman Act) provides that it is not an act of infringement to practice a patented invention “solely for uses reasonably related to the development … of information” and its submission to the Food and Drug Administration.\textsuperscript{[68]}

Unfortunately, a blanket research exemption is not statutorily recognized. In the late-1980s, Congress started contemplating about carving out a broad exemption for research.\textsuperscript{[69]} The administration, however, opposed the codification of research exemption. In a letter to Congress, the Commissioner of Patents and Trademarks stated that “the Administration would not favor legislation creating a research exemption or codifying the experimental purpose doctrines, because it could diminish the strong incentive provided by the patent system.”\textsuperscript{[70]} There is even less enthusiasm on the part of the judiciary. The Federal Circuit reiterated its position in 2002
that a research or experimental activity that was in keeping with the alleged infringer’s legitimate business, regardless of commercial implications, is not exempted as experimental use.\textsuperscript{[71]} In Madey v. Duke University, the patent holder implemented certain of his patented inventions in a research lab for the university, but was later removed from his position as lab director.\textsuperscript{[72]} The Federal Circuit found that experimental use exception did not apply because research projects furthered a research institution’s legitimate business objectives, including educating and enlightening students and faculty participating in these projects.\textsuperscript{[73]} These projects also served to increase the status of the institution and lure lucrative research grants, students and faculty.\textsuperscript{[74]}

It is not this Note’s prerogative to question the wisdom of Federal Circuit’s attitude toward experimental use defense in general; many commentators have expressed renewed concern for Madey’s untoward consequences and argued for the preservation of a viable public domain.\textsuperscript{[75]} Proponents of experimental use have argued that near-death of this exception is inconsistent with the disclosure requirement of the patent statute. The exclusive rights are granted in exchange for the disclosure of the patent; if the public has absolutely no right to use the disclosure without the patentee’s consent up till patent expiration, there is no reason for imposing an “enabling disclosure” requirement on the patentee at the outset of the patent term.\textsuperscript{[76]} The “enablement” requirement secures to the public the means to design around and build upon the disclosed invention, and any design-around or build-upon improvement can be interpreted as “antagonistic” to the interest of the patentee, either because the public has something else to choose or because the patentee could have got license fees from the experimenter. If courts treat the disclosure requirement seriously, a general experimental use exception should be preserved.

What this Note attempts to achieve is less ambitious. It focuses on the economics of software patent in the general incentive-creating scheme and advocates for the preservation of experimental use defense for a specific type of experiment: reverse engineering to achieve compatibility in software programs.
2. Incentive-Building Forces Other than Exclusive Patent Rights

In software industry, there are a lot of alternative ways to provide incentives to invent and innovate, other than granting exclusive patent rights. As previously discussed in Part II A, the relaxed disclosure requirement forces competitors to reverse engineer a program to make his own program compatible. Reverse engineering of a software can be costly and time consuming, as a reverse engineer has to decompile machine-readable only object code into source code and then develop compatible products. This prevents the public goods problem and assures to the original owner sufficient time to recoup his investment.

Moreover, the fact that software is protected by copyright and patent hints that these two laws may be providing repetitive incentives to create the same subject matter. As originally designed, copyright and patent laws protect different subject matters and provide different incentives: copyright provides incentives to create expressions while patent encourages technological inventions. The emergence of software technology is erasing the line between patentable and copyrightable works, because copyright and patent protect precisely the same material in a software program: expressive instructions commanding the function of a machine. Accordingly, both laws are providing identical stimulus to same type of intellectual creation in software. Copyright exists automatically upon completion of the work. It has been the traditional, and universal, form of protection and has been supplying sufficient incentives to the prosperity of this industry in the past. Despite the cloud over the patent eligibility of software for the past two decades, growth of this industry showed no sign of losing its momentum due to lack of impetus. Even after software patentability became certain, systematic flaws in software patent prosecution leave patent owners insecure about the validity of their patent rights. They usually will not risk litigation for patent infringement, if copyright claims can be substantiated, because patent validity is always the first defense to be raised. A telling fact is that Sony simply sued Connectix for copyright infringement, despite the fact that it held patent rights over the program. The truth of matter is, with copyright as backup, software owners have enough incentive to create, innovate and enforce rights. The convergence of copyright and
patent does not provide additional incentives to create; it merely provides opportunities for software owners to exploit the system, obtaining protection in one sphere that nullifies competitor’s legitimate use in another sphere. This added layer of patent protection thus has lower marginal returns in creating additional incentives while at the same time threatens the exact progress intellectual property law intends to promote.

Technical constraints of interoperability requirement in a virtual network market also affect incentive-building analysis in intellectual property law making. Software industry is a virtual network industry because the inherent value of the goods to the consumers increases with the climbing number of additional users of identical and/or interoperable goods. Take an operating system as an example, as the number of applications running on it increases, the value of this system goes up, which leads to increased number of developers writing programs for this system and, in turn, more consumers adopting this system for the better enjoyment of available programs. Compatibility with existing popular or dominant systems, called interoperability, functions as a technical constraint on a newcomer’s ability to compete. If an operating system wants to compete with Windows, it must be “horizontally compatible” with Windows, implementing all of its APIs so that current Windows users can continue to use programs they already have when they switch to the new system. In the same vein, for an application program to be competitive in the software market, it must be “vertically compatible” with Windows by running successfully on it. These technical constraints of interoperability assure to inventors or investors that their investment can be recouped with lead time and additional users from compatible networks, hence providing them with incentives to invent and innovate.

Economics of network effects thus indicates that relatively weak intellectual property protection works best to promote efficiency at a societal level. In a network market, a consumer’s decision to join has a positive externality because it increases the network’s value to existing members; a decision not to enter or to join another network, on other hand, creates a negative externality. Strong intellectual property rights provide incentive to invent and innovate because they confer to the right holder the ability to price over his marginal cost. However,
the ability of the right holder to charge a price over marginal cost brings about negative externalities because high price drives away consumers, and a competitor’s network may increase in value if consumers switch to that system. Conversely, with weaker intellectual property protection, competitors may make its product compatible with the right holder’s system or product, thus offer an alternative for its own consumers to join the patent holder’s network, which in return increases the value of the network.\textsuperscript{188}

However, strong intellectual property protection in a network market is prone to create distorted incentives. Network effect expands intellectual property protection both temporally and horizontally.\textsuperscript{189} For example, Bell successfully leveraged a 17-year patent into long-term control over the telephone market. Similarly, Microsoft leveraged its monopoly in operating system, obtained partly through copyright ownership over the system, into web browser market.\textsuperscript{190} Horizontally, even if a patent only explicitly covers a particular product, users in the network market are “locked in” because they are reluctant to incur the cost to learn about or convert an incompatible alternate product, even though it is not inherently inferior.\textsuperscript{191} A strong patent protection combined with network effects thus promises a large reward: if one can set the standard, he may control the market, even different markets, beyond the term of his intellectual property rights. This induces socially excessive investment in R&D during the race to set the industry standard,\textsuperscript{192} and declining innovation after the standard is set: even though the standard-setting monopolist will innovate, it may not be as responsive to incentives as firms in competition would be.\textsuperscript{193} In fact, since network markets are heavily path-dependent, if the “winning” standard is both inferior and durable, it locks consumers into a sub-optimal system regardless of the legal regime. The classical example is the QWERTY keyboard.\textsuperscript{194} In rejecting Microsoft’s contention that it integrated its Internet Explorer into the Windows operating system with the intent to benefit consumers, the district court there noted that IE “is not demonstrably the current ‘best of breed’ Web browser, nor is it likely to be at any time in the immediate future.”\textsuperscript{195} Finally, consumers fall prey to a distorted network market. They have to predict
which product will win the standards competition; otherwise, they may be stranded on the losing, incompatible product and incur the costs of switching to the winner.\footnote{961}

Although a strong intellectual property protection in a network market is less than optimal, there are still many compelling reasons to grant patent protection in a network market to preserve incentives. First of all, network markets are risky. At the time an inventor seeks protection, the nature of the market and the strength of network effects, if any, are unlikely to be reliably known.\footnote{971} Investors are thus more likely to demand some measure of exclusivity in the form of intellectual property rights before they invest. Moreover, investors fund innovations in network markets not only in anticipation of protection of new innovations, but also those parts that are traditionally protected by intellectual property law.\footnote{981} A categorical rule denying protection thus may deter incentives for initial innovation.

The proper solution to ensure optimal efficiency as well as sufficient incentives, the patent protection should be tailored to the unique nature of the software industry: protection should be dynamically assessed, beginning with a strong protection regime, and changing to compatibility when more information about market conditions becomes available.\footnote{991} Under this new paradigm, software is patent eligible, to provide incentives for inventors to develop more useful interfaces. Where network effects create cycle of consumer preferences and developer incentives that locks consumer into a dominant product, if the dominant firm were to engage in conduct directed toward controlling other product markets or the innovation market, a dynamic approach requires courts to open the standard to competition.\footnote{1001}

Applying this new paradigm to Sony’s patent infringement case, a court may find, without too much a stretch, that the code to PlayStation should be opened up to allow competition. The fact that Connectix had to reverse engineer PlayStation game console is indicative of the existing “locked in” consumer preference and Sony’s unwillingness to “open up” the system.\footnote{1011} In addition, Sony’s anti-competitive conduct is an attempt to leverage its patent protection to unprotectable expressive elements in its competitor’s product.\footnote{1021} Drawing on the Ninth Circuit’s reasoning in \textit{Sega} that intermediate copying for the purpose of
ascertaining aspects that were not covered by copyright is not considered commercial use under copyright law, a court may conclude that an accused defendant’s use of the program to ascertain expressive elements not covered by the patent claims, though having commercial implications, is not considered as a commercial use that undermines the lawful reward to the right holder. As Sony’s patent suit is directed toward controlling the expressive market, as the procedural history clearly shows that Sony considered this as a copyright infringement in the first place, there are ample equitable elements to persuade a court to invoke experimental use to prevent the abuse of the system and to preserve the social progress intellectual property laws are designed to promote.

III. CLEARING THE WAY FOR STATUTORY RECOGNITION OF EXPERIMENTAL USE

A. Domestic Legislative Process

As noted earlier, Congress tipped its toe into the hot water of creating a broad research or experiment exception in 1990. The Judiciary Committee reported a proposed bill to the House, under the name of Title IV, Research, Experimentation and Competitiveness, of the Patent Competitiveness and Technological Innovation Act of 1990s. This proposed bill intended to encourage research and experiment by providing that the making and using of a patented invention solely for research or experimentation would not constitute an act of infringement unless the patented invention is primarily used for research or experimentation. If the patented invention is primarily used for research or experimentation, such as a microscope, then it would not be an act of infringement to (1) study, characterize, or evaluate the invention; or (2) use the invention to create a product outside the scope of the patent.

To justify this exemption, the Committee referred back to the incentive/progress balance in intellectual property law making, as evidenced by the requirement that the resulting product be outside the scope of patent. It emphasized the central tenet of patent law that scientific information may be used to create new and better inventions “in competition with the patented invention,” as manifested by case law’s endorsement of “design around.” Notably, Sawin is construed as standing for the proposition that if the particular complained-of use is not for profit, then the use is protected, even if carried out by a for-profit institution. It even stated that “(i)t
is also possible to argue that the law of experimental use is broader and applies to testing a patented invention for adaptation to the experimenter’s business provided that such is not for profit. Following this reasoning, Congress included in the list of exempted activities such use as experimenting on a patented invention for the purpose of improving on it or developing a further patentable invention, experimenting for designing around a patented invention; testing to determine whether the invention meets the tester’s purposes in anticipation of requesting a license.

The abortion of this Act evinced Congress’s reluctance to grant a blanket exemption. For advocates of software exemption, hope is still floating because an exemption specifically targeted at software industry is not precluded, as demonstrated by the successful creation of experimental use exception for the pharmaceutical industry.

In *Eli Lilly and Co. v Medtronic, Inc.*, Justice Scalia carefully examined the legislative history of Hatch-Waxman Act, which statutorily recognized experimental use for drug companies applying for regulatory approval, and expounded on the rationale of creating such an exception. He noted that the Act was designed to respond to two unintended distortions of the 17-year patent term produced by the requirement that certain products must receive premarket regulatory approval. The first market distortion occurs when the patent holder cannot as a practical matter reap any financial rewards during the early years of the term, because the “clock” on his patent term runs independently of regulatory approval, while it usually takes several years after the grant of patent for regulatory approval to go through. The second distortion occurs at the other end of the patent term. After the Federal Circuit decided that even uses for the sole purpose of conducting tests and developing information necessary to apply for regulatory approval constitutes patent infringement, those who planned to compete with the patentee cannot start testing on a competing product until the expiration of the entire patent term. A patentee’s de facto monopoly would continue for an often substantial period until regulatory approval for the competing product was obtained. The Hatch-Waxman Act eliminated distortion at each ends of the patent term: Section 201 established a patent-term extension for patents
relating to certain products that were subject to lengthy regulatory delays and could not be marketed prior to regulatory approval, Section 271(e)(1), allows competitors, prior to the expiration of a patent, to engage in otherwise infringing activities necessary to obtain regulatory approval.\footnote{\textsuperscript{111}}

Like pharmaceutical products, software industry is the right, even propitious, candidate for this exception. First of all, the incentive to disclose arguably does not exist, or is not required, under the current regime, and greater needs exist to extract the verity of the invention. An invention not sufficiently disclosed is not disclosed at all and secrecy effectively allows a patentee to leverage the patent into a term longer than it should have enjoyed, and even worse in software industry, to suppress competition in expressive fields that are not covered by patents. Second, network effect further gives a patentee the capacity to expand the scope of rights temporally and horizontally, and even creates a distorted incentive to suppress social progress. Finally, both technologies do not fit very comfortably into the traditional patent protection framework and courts have done lots of stretching and twisting to provide technology-specific protection to both.\footnote{\textsuperscript{112}} As “a patentee’s de facto monopoly would continue for an often substantial period”,\footnote{\textsuperscript{113}} even expanding into horizontally related markets, while at the same time the judicial treatment of the new technology “are not expressly informed by the economics of the industries, but by an ad hoc combination of judicial anthropology and stare decisis”, it is high time for Congress to use the more informed legislative process to recognize a technology-specific experimental use exception to fix specific market distortions within the software industry, just like it did two decades ago for the pharmaceutical industry.\footnote{\textsuperscript{114}}

\textbf{B. International Obligations under TRIPS}

Domestic law making in the United States is now being shaped by international law development.\footnote{\textsuperscript{115}} Treaties, of course, are supreme laws of the land and its obligations should be complied with. The United State is a member of the World Trade Organization and is bound by the Trade Related Aspects of Intellectual Property Rights ("TRIPS").
The TRIPS Agreement recognizes the “principle that patents should not impair the advance of technology, and, consequently, that the effects of patents should not cover certain acts.” The final version of Article 30 is rather vague:

“Members may provide limited exceptions to the exclusive rights conferred by a patent, provided that such exceptions do not unreasonably conflict with a normal exploitation of the patent and do not unreasonably prejudice the legitimate interests of the patent owner, taking account of the legitimate interests of third parties.”

Ambiguities in international treaties are usually construed as affording members states some autonomy on specific issues that implicates national authority and country-specific policies. One commentator examined the legislative debate and other proposals of TRIPS, and concluded that Article 30 is intended to allow exceptions for infringement conducted for (i) private, non-commercial purposes; (ii) research; (iii) experimentation for testing or improvement; and (iv) educational purposes. The statutory recognition of experimental use in general thus fits in the ambit of Article 30.

Ironically, a technology-specific exemption is seemingly inconsistent with a recent WTO panel interpretation of Article 27(1). In Canada-Patent Protection of Pharmaceutical Products, a WTO panel rejected the European Union’s challenge to the regulatory review exemption created by Canada, but only on Canada’s assurance that the exemption was technology-neutral, namely, legally available to every product that was subject to marketing approval requirements. There, Canada’s regulatory review exemption mirrors the experimental use exception in U.S. law, allowing competitors to experiment on a patented invention before the expiration of patent term, with the sole purpose to get regulatory approval. The European Union challenged this law on the basis that it created a de facto discrimination against pharmaceutical products, because only pharmaceutical products need to go through regulatory approval. The panel agreed with the European Union on this point, though it eventually dismissed the case because there was not enough evidence showing disparate impacts
on other industries. This interpretation, however, creates an extra hurdle for member states to avail themselves of the fine-tuning abilities under Article 30.

An obvious flaw in the panel’s attempt to reign Article 30 exception-making analysis into Article 27’s technology-neutrality analysis is that these two Articles deal with different issues. Article 27 deals with “patentable subject matter”, while Article 30 deal with tailoring patent rights to specific problems within a member state. As the dynamic intellectual property paradigm, discussed supra in Part II B 2, indicates, strong intellectual property protection in the first place creates incentives to invent and innovate. A technology-neutrality requirement ensures that new technological advances may be eligible for patent protection, and hence preserves incentives. The dynamic approach also requires courts or other competent branches of member states to weaken intellectual property protections where granted rights are improperly exploited due to the misfit between the technology and established, yet ill-suited, legal rules. Article 30 thus reflects a totally different policy objective and relies on the technology-specific approach. As two commentators noted, because disparate impact is almost impossible to avoid, member states would not have committed themselves to the TRIPS Agreement had they known that they did not have the right to fine-tune their laws to different problems posed by different technologies.\[121\] Rather, the non-discriminatory treatment under Article 27 is intended to restrict a de facto discrimination against a specific industry without a legitimate purpose.\[122\] After all, the panel did acknowledge in Canada-Pharmaceuticals that “Article 27 does not prohibit bona fide exemptions to deal with problems that may exist only in certain product areas.”\[123\]

Therefore, the statutory recognition of experimental use in software industry is consistent with both domestic policy prerogatives and international lawmaking norms.

**CONCLUSION**

Since its inception, software technology has been significantly altered, even warped, the course of intellectual property law making. The convergence of copyright and patent on the same subject matter, namely software, came to fruition through a series of ad hoc analysis of unrelated cases, and many of the policy implications, especially the broader constitutional balancing of
incentive to create and promotion of social progress, are not carefully examined. As exclusive rights are granted expansively without concomitant concern for the shrinking public domain, the new wave of software patents portends the advent of a perfect control over software. As a perfect control over an intellectual creation deprives the public opportunities to use freely elements traditionally not protected by copyright law or to build upon existing technology to promote social progress, the constitutional balance between incentives and progress has been direly tilted to one end. Experimental use exception in patent law should thus be resurrected and, together with fair use in copyright law, preserve the rights of competitors to gain access to certain aspects of software to develop compatible programs. The resulting increase in the number of independently designed programs is precisely what intellectual property law is designed to achieve. Both the administrative and the judiciary branch showed little enthusiasm in re-assessing the constitutional balance. Congress thus needs to use the more informed legislative process to save experimental use exception to restore the market distortions created by the convergence of copyright and patent as well as the network effects in this particular industry.

[4] APIs are the connectivity components of operating systems that specify how a particular operating system and its applications communicate. As parts of a computer program, they are protected by copyright.
[5] Software engineers designing a product that must be compatible with a copyrighted product frequently must “reverse engineer” the copyrighted product to gain access to the functional elements of the copyrighted product. This process is usually referred to as “decompilation,” as it involves working backwards from object code to construct a simulacrum of the source code. See Andrew Johnson-Laird, Software Reverse Engineering in the Real World, 19 U. DAYTON L. REV. 843, 845-46 (1994).
[6] See Sony Computer Entertainment, Inc. v. Connectix, 203 F.3d 596 (9th Cir., 2000) (“Sony I”) (holding that it is lawful to reverse engineer a video game system as an intermediate step to creating a computer program that would allow games designed for that system to run on a PC); Bateman v. Mnemonics, Inc., 79 F.3d 1532, 1539-49 n.18 (11th Cir., 1996) (endorsing the use of reverse engineering to gain access to the unprotected ideas in a program); Sega Enterprises v. Accolade Inc., 977 F.2d 1510 (9th Cir., 1993) (reverse engineering for the purpose of developing compatible video game programs is considered fair use under copyright law).
Despite the initial reluctance of the PTO and the courts to recognize the patentability of software, by the year 2000, roughly 100,000 software or software-related patents are in force in the U.S., and several thousands more are being issued every year. Cohen and Lemley, supra note 11, at 11.

This principle is established in patent law through a series of litigations over pharmaceutical products. The Federal Circuit held that a patent is infringed when the patented product is generated by metabolization of a different drug within the human body, and that chemical “intermediates” temporarily generated in the course of making a final product infringe a patent covering those intermediates. Hoechst-Roussel Pharm., Inc. v. Lehman, 109 F.3d 756, 759 (Fed. Cir. 1997); Zenith Labs v. Bristol Myers Squibb, 19 F.3d 1418, 1422 (Fed. Cir. 1994).


As Sir Isaac Newton put it, “If I have seen further it is by standing on the shoulders of giants.” The Oxford Dictionary of Quotations 362 (3d ed. 1979).

A blocking patent is a patent on the improvement of a previously patented invention. The accused infringer owns his own patent over the improvement; therefore he can prevent the original patent owner from using or making the improvement. The original owner, of course, can prevent the improver from using or making the underlying basic invention. Unless the parties bargain, no one gets the benefit of the improvement. The law, by granting the improver his own patent rights over the improvement, promotes social progress and gives the improver an important weapon to bargain. See generally, Robert Merges, Intellectual Property Rights and Bargaining Breakdown: The Case of Blocking Patents, 62 TENN. L. REV. 75 (1994).

The reverse doctrine of equivalents protects from infringement a product “so far changed in principle from a patented article that it performs the same or similar function in a substantially different way, but nevertheless falls within the literal words of a claim.” Graver Tank & Mfg. Co. v. Linde Air Products Co., 339 U.S. 605, 609 (1950).

For a more detailed explanation about when an improver can bargain for rights and profits from using the invention and when he can use it for free, see Mark A. Lemley, The Economics of Improvement in Intellectual Property Law, 75 TEX. L. REV. 989 (1997).

In fact, even courts did apply the abstraction/filtration test in copyright cases fail to successfully separate expression from function. As a commentator noted, many courts have attempted to do so by analogizing software to an “instructive” work such as a cookbook or blueprint. They treat source code, which is easily manipulated by humans, as the expression, and the object code, the machine-readable language that is executed by computer, as the instructions or function. However, this attempt conflates two different meanings of the word “instruction.” Instructions to a computer, unlike those to a person, do not inform the computer how to carry out a process; instead, they command the arrangement of voltages in computational registers, a process which humans won’t follow. The code, in essence, is abstract machinery. Burk, supra note 3, at 118-119.

Lemley, supra note 29, at 1084 at n.224.
but true programmers are extremely skilled, although commentators are quick to point out that this assumption was nothing


the Federal Circuit has gone as far as saying that the patentee can satisfy the best mode requirement for inventions

conversion of a complete thought … into a language a machine understands is necessarily a mere clerical function to

Out” Programs Cohen, another article that until recently computer scientists were not even eligible to sit for the patent bar. Julie E. Thur

issued every year.

nature of software patent, the Patent Office has issued numerous software patents since 1980s. By the year 2000,


This classic justification for having a patent system is expounded by the Supreme Court in Kewanee Oil Corp. v. Bicron Corp., 416 U.S. 470, 489 (1974) and later reiterated in Bonito Boats, 489 U.S. 141, 147-154 (1989).

Software was first held as not patentable subject matter because it is based on algorithms, which are fundamental scientific principles. Gottschalk v. Benson, 409 U.S. 63 (1972). After several twists, finally in Diamond v. Diehr, the Court found that if the program led to the production of some physical result, it is patentable. 450 U.S. 175 (1981). See generally, Pamela Samuelson, Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions, 39 EMORY L.J. 1025 (1990).

Cohen and Lemley, supra note 11, at 7-8. They noted that despite the then-questionable, and hotly debated, nature of software patent, the Patent Office has issued numerous software patents since 1980s. By the year 2000, close to 100,000 software or software-related patents are in force in the U.S., and several thousand more are being issued every year.


See Fonar Corp. v. General Electronic Corp., 107 F.3d 1543, 1549 (Fed. Cir. 1997). The reason is that “the conversion of a complete thought … into a language a machine understands is necessarily a mere clerical function to a skilled programmer.” Northern Telecom, Inc. v. Datapoint, Corp., 908 F.2d 931, 941-42 (Fed. Cir. 1990). In fact, the Federal Circuit has gone as far as saying that the patentee can satisfy the best mode requirement for inventions implemented in software even though they do not use the term “computer” or “software” anywhere in the specification. Robotic Vision Sys., Inc. v. View Eng’g, Inc., 42 U.S.P.Q.2d 1619 (Fed. Cir. 1997); In re Dossel, 42 U.S.P.Q.2d 1881 (Fed. Cir. 1997).

A careful reading of such cases reveals that the Federal Circuit did so because it believes that computer programmers are extremely skilled, although commentators are quick to point out that this assumption was nothing but true. Dan L. Burk & Mark A. Lemley, Is Patent Law Technology-Specific, 17 BERKELEY TECH. L. J. 1155,
1185 (noting that the Federal Circuit’s decisions are based on a perception that was set in earlier cases but does not reflect the modern realities of the software industry).

[51] Cohen & Lemley, supra note 11, at 24 n. 87.

[52] Systematic flaws in software patents can be cured by imposing more stringent disclosure standard. However, in light of the fact that function and expression are mixed, and even inseparable, in a program, a stringent disclosure requirement may force software owner to disclose certain part of the program that should properly be protected by copyright and trade secrets. If reverse engineer is allowed, then only those who are interested in ascertaining the verity will engage in decompiling the codes, and they are also motivated not to disclose their discovery to others to keep their own competitive edge. The experimental use defense regime is thus more finely tuned to address the particular challenges imposed by software.

[53] With a very stringent application of commercial use analysis, the only permissible experiments are those conducted for “amusement, to satisfy idle curiosity, or for strictly philosophical inquiry.” Madey, 307 F.3d at 1362.

[54] In Madey, the Federal Circuit found that research conducted in universities is not entitled to an exemption from infringement liability. 307 F.3d at 1362.

[55] In the seminal case that creates this defense, Sawin v. Guild, the court excused the sale of a patented invention to satisfy a judgment. 21 F. Cas. at 555.


[57] The same conclusion has been reached by many courts deciding copyright “fair use” claims. As a dissenting judge noted, by focusing on impact on a copyright holder’s market, this approach will always yield a finding of infringement because fees that the copyright holder might extract from the market will be “lost” if the taking were deemed fair use. Princeton University Press v. Michigan Document Servs., Inc., 99 F.3d 1381, 1407 (6th Cir. 1996) (Ryan, J., dissenting).

[58] See Ordnance Engineering Corp. v. United States, 84 Ct. Cl. 1, 4 (1937) (excluding from accounting 7,425 illuminating shells to be built for “experimental purposes”); Chesterfield v. United States, 159 F. Supp. 371 (1958) (finding no infringement despite the United States admittedly used 3,679 pounds of a patented mental alloy, because a portion of the alloy was used for experimental purposes and there was no evidence that the remainder was used other than experimentally).

[59] In Douglas v. United States, the court found experimental use defense did not apply where the government bought six British made aircraft and replacement engines and used them over a four-year period, because the use was for “legitimate government interest.” 181 U.S.P.Q. (BNA) 170 (1974). In Pitcairn v. United States, the court found that “tests, demonstrations, and experiments” using infringing aircraft were ‘intended uses’ of such aircraft and are “in keeping with the legitimate business of the government.” 547 F.2d 1106, 1125-26 (Cl. Ct. 1976).

[60] One commentator thus argues that the exception is never justified. Richard E. Bee, Experimental Use as an Act of Patent Infringement, 39 J. PAT. OFF. SOC’Y 357, 371-72 (1957) (suggesting that even use by universities for teaching purposes should not be shielded from infringement, as universities are in the business of educating students and charge a tuition fee for such education).


[62] See South Corp. v. United States, 690 F.2d 1368, 1369 (Fed Cir. 1982) (en banc).

[63] Id. at 863.

[64] Id.


[67] Transgenic Act Hearing at 189.


[69] Id. at 1352-53.

[70] Id.


[73] See supra notes 33-37 and accompanying text.
Part II A of this Note, addressing the issue of systematic flaws in software patents. An example is the multimedia data retrieval patent granted to Compton’s New Media: industry criticism prompted the PTO to reexamine the patent and ultimately to reject it because it did not represent a novel and nonobvious advance over existing technology. Cohen, supra note 49, at 1179.

There is an international consensus that software can be protected by copyright, but no such a consensus exist for software patents. Graeme B. Dinwoodie, LECTURES ON INTERNATIONAL INTELLECTUAL PROPERTY, notes on file with the author.

See supra note 29, at 1036.

Interestingly, Prof. Lemley argued that because improvements on existing technology are better protected under patent law, as patent law is more concerned with technological advancement, copyright owners will use copyright protection to nullify improver’s rights in the patent sphere. Lemley, supra note 29, at 1036. The same can be told about software patent owners.

Concededly, this conclusion is based on legal reasoning rather than hard data. It is not this Note’s goal to prove this point through extensive data collection and analysis. For more detailed economic analysis of the cost to social progress resulting from restriction on reverse engineering, see Pamela Samuelson and Suzanne Scotchmer, THE LAW AND ECONOMICS OF REVERSE ENGINEERING, 111 Yale L. J. 1575 (2002).

Network effects exist when a consumer’s utility associated with a good increases as others also purchase it. In an actual network like the telephone system, the network effect is direct as the product’s entire value inheres in enabling communication among product owners. In contrast, goods constitute virtual networks when the inherent values to consumers increase as more people use identical or interoperable goods. See Michael L. Katz & Carl Shapiro, Network Externalities, Competition, and Compatibility, 75 AMER. ECON. REV. 424, 424 (1985); see also Mark A. Lemley & David McGowan, Legal Implications of Network Economic Effects, 86 CAL. L. REV. 479, 488-89 (1998).

The network effects also characterize the market for application programs, but in a less obvious way. For example, when a user invests time and effort in learning how to use a particular program, say Lotus electronic spreadsheet, he or she is unlikely to adopt a new, improved product that requires different skills and/or that cannot process the user’s files. See Lotus Dev. Corp. v. Borland Int’l, 49 F.3d 807 (1st Cir. 1995), aff’d by an equally divided Court, 516 U.S. 233 (1996).

See Lemley & McGowan, supra note 84, at 524.

See, e.g., Lemley, supra note 29t, at 996 (noting that exclusive rights “prevent competition in the sale of the particular work or invention covered by the intellectual property right, and therefore allow the intellectual property owner to raise the price of that work above the marginal cost of reproducing it”).

See also Lemley & McGowan, supra note 84, at 534 (arguing that the traditional case for intellectual property can be rebutted in network markets by demonstrating that the network effects themselves will ensure an adequate return to the initial creator even absent intellectual property protection.) In fact, one may argue that intellectual property system is a costly way to protect such rights if network effect can protect it for free: intellectual property rights involves administrative costs in setting up and maintaining the system, enforcing rights through courts, costs associated with obtaining, maintaining, and enforcing the rights.


See Lemley & McGowan, supra note 84, at 496.


This point is made clear by the finding of facts in Microsoft antitrust case. The opinion is replete with references to instances of Microsoft using its monopoly power to direct and stifle innovation in various markets with the goal of protecting its profit stream from the Windows operating system and the applications barrier to entry into the market for competitive operating systems. United States v. Microsoft Corp., 87 F.Supp.2d 30 (D.D.C. 2000)(noting that Microsoft sought to deter IBM, Apple, and others form innovating and stating that “Microsoft’s anticompetitive actions trammeled the competitive process through which the computer software industry generally stimulates innovation and conduces to the optimum benefit of consumers.”)


Microsoft, 87 F. Supp. 2d at 40.
A related counter-point is that more often than not investors even want to use the intellectual property protection mechanism to protect what is not protected. For example, when only part of a program is patented, the existence of patent right to exclude making or using of that part effectively prevents the making or use of the whole program because once a program is run, it infringes.


There is no indication that there was a breakdown in negotiating for a license to use Sony’s software by Connectix, probably because the right to reverse engineer has been firmly established. Sony I, 203 F.3d 596. However, the negotiation for a license does not necessarily mean Sony has sure rights over the software. Sometimes the fear of litigation costs persuades firms to obtain licenses.

Note that the anticompetitive effect is felt in the reduced final products that do not infringe either Sony’s copyright and patent rights. The seemingly troublesome obstacle is that intermediating copying has been found to be copyright infringing activities. This problem is created by copyright law’s insensitivity to improvement, which is traditionally not an issue in copyright law. See Lemley, supra note 29, 989 (arguing that copyright law should accommodate public interest in improvement, as patent law does, because of the convergence of copyright and patent protection over software).


Id. at 65-66.

Patent House Report at 41 (quoting State Indus. v. A.O. Smith Corp., 751 F.2d 1226, 1236 (Fed. Cir. 1985) (characterizing designing around a competitor’s product as a “negative incentive” under patent law, but endorsing it because it confers the benefits to bring a steady flow of innovations into the marketplace)).

Id. at 42. As the using of the patented invention during reverse engineering is a use intended to ascertain the functional element, not to profit, it fits in the ambit of this proposed exception.

This proposition, of course, is arguably not supported by cases like Douglas and Pitcarin, see supra note 61.


Id. at 671.

See Burk and Lemley, supra note 52, at 1160-83 (using biotechnology patent and software patent cases to illustrate the technology-specific nature of current patent protection regime, despite a technology-neutral statutory regime).

Eli Lily, 496 U.S. at 671.

Id. at 1205.

An argument to comply with international norms is to keep abreast of major competitors’ lawmaking to retain one’s own competitive edge. Domestic law making also has impacts on international norm building. An even more ambitious approach to initiating timely change in U.S. law is to put heat on competitors and give U.S. Trade Representative more bargaining power in pushing changes in foreign law or international law in the future. Graeme B. Dinwoode, Lecture on International Intellectual Property. Notes on file with author.


Agreement on Trade-Related Aspects of Intellectual Property Rights, Apr. 15, 1994, art. 30 [hereinafter TRIPS Agreement].


Article 27, in its relevant parts, provides:

1. Subject to the provisions of paragraphs 2 and 3, patents shall be available for any inventions, whether products or processes, in all fields of technology, provided that they are new, involve an inventive step and are capable of industrial application. (5) Subject to...
paragraph 4 of Article 65, paragraph 8 of Article 70 and paragraph 3 of this Article, patents shall be available and patent rights enjoyable without discrimination as to the place of invention, the field of technology and whether products are imported or locally produced.


[121] Dinwoodie & Dreyfuss, supra note 75 at pages 16-17 (arguing that tailoring is the centerpiece of intellectual property law, each type of law is geared toward to requirement of particular industries and characteristics of the subject matter those industries produce).

[122] Id. at 16.

[123] Canada-Pharmaceuticals, supra note 120 at pp. 170-71, ¶ 7.92.