2012

Scaling the Patent System

Christina Mulligan
Brooklyn Law School, christina.mulligan@brooklaw.edu

Timothy B. Lee

Recommended Citation
68 N.Y.U. Ann. Surv. Am. L. 289 (2012-2013)

SCALING THE PATENT SYSTEM

*CHRISTINA MULLIGAN & TIMOTHY B. LEE**

ABSTRACT

Why do firms in some industries ignore patents when developing new products? This paper posits a simple answer to this long-puzzling question: firms ignore patents because they are unable to discover the patents their activities might infringe. The costs of finding relevant patents, which we call "discovery costs," are prohibitively high.

Not all industries face high patent discovery costs. Chemical patents are "indexable," meaning that relevant patents can be efficiently retrieved by chemical formula. As a result, discovery costs in the chemical and pharmaceutical industries are low, and inadvertent infringement by firms in these industries is rare. But many other patent categories are not indexable, and in some cases that makes avoiding infringement practically impossible. In software, for example, patent clearance by all firms would require many times more hours of legal research than all patent lawyers in the United States can bill in a year. The result has been an explosion of patent litigation.

This paper attacks two core premises of patent law—that parties are always able to respect each other's patent rights, and that firms should be punished for infringement even if they could not have avoided it. It concludes with several suggestions for how to change the patent system to alleviate the problems created by non-indexable patents.

* Christina Mulligan is a Postdoctoral Associate and Lecturer in Law at the Information Society Project at Yale Law School, and beginning August 2013, an Assistant Professor of Law at the University of Georgia School of Law. Timothy B. Lee is a computer scientist and journalist. He is an adjunct scholar at the Cato Institute and covers tech policy for *Ars Technica*. The authors wish to thank Bryan Choi, James Grimmelmann, Daniel Hemel, Ben Klemens, Tom Lowenthal, Evan McClanahan, Kevin E. Park, Amanda Rohn, Wendy Seltzer, Samson Vermont, Carly Weinreb, Aaron Williamson, and the attendants of Patent Conference 2 and IP Scholars Conference 2012 for their helpful comments on earlier drafts of this paper.

INTRODUCTION

In 1945, Friedrich Hayek wrote a famous essay called "The Use of Knowledge in Society."[1] Responding to advocates for greater central planning of economic activity, Hayek pointed out that their theories assumed that knowledge about economic circumstances could be taken as "given" to economic decision-makers, an assumption he argued was unreasonable.[2] Knowledge about the state of the economy—about what resources exist and what goods and services consumers demand—is dispersed among millions of people. Gathering the information together in one place would be impractical, and even if it were done, no economic decision-maker could possibly absorb it all.[3]

The tendency to implicitly assume that economic actors are omniscient is a common pitfall of theoretical social science. By definition, the theorist knows everything there is to be known about the stylized model he has invented. Theorists often implicitly assume that economic actors automatically have the information they need to make decisions.[4] Indeed, such assumptions may be essential to building a tractable model of the world. But the failure to ponder the feasibility of acquiring and using information can lead to flawed conclusions.

1. Friedrich A. Hayek, *The Use of Knowledge in Society*, 35 AM. ECON. REV. 519 (1945).
2. *Id.* at 519.
3. *Id.* at 524.
4. *See id.* at 519.

The contemporary patent debate suffers from this blind spot. Each patent is a demand that the world refrain from practicing a claimed art without the patent holder's permission. Potential infringers can only comply with this demand if they are aware of the patent's existence. On a blackboard or in the pages of a law review article, it is easy to assume that everyone knows about every patent.

But the real world is not so simple. To avoid infringement, a firm must expend resources to learn about potentially relevant patents. Typically this means hiring patent lawyers to conduct patent searches, which may or may not be affordable or effective.[5] In this paper, we'll call the costs of such information-gathering activities the patent system's "discovery costs."[6] One criterion for a well-functioning patent system—or any system of property-like rights—is that discovery costs be low enough to make it economically feasible for firms to obtain the information they need to comply with the law.[7]

5. For example, a recent Supreme Court case presumed that "simply asking an attorney to examine a product and compare it to the data base of existing patents is not a dependable way to see if a product is likely to infringe a patent." Stephen M. McJohn, *Top Tens in 2011: Patent & Trademark Cases*, 10 Nw. J. TECH. & INTELL. PROP. 313, 317 (2012) (discussing *Global-Tech Appliances, Inc. v. SEB S.A.*, 131 S.Ct. 2060 (2011)). *See also* Global-Tech Appliances, 131 S.Ct. at 2064 (describing how a patent attorney failed to locate a deep fryer patent that his client's invention infringed).

6. Discovery costs are one of the types of transaction costs identified by Ronald Coase in his seminal essay, *The Problem of Social Cost.* Coase explained:

> In order to carry out a market transaction *it is necessary to discover who it is that one wishes to deal with*, to inform people that one wishes to deal and on what terms, to conduct negotiations leading up to a bargain, to draw up the contract, to undertake the inspection needed to make sure that the terms of the contract are being observed, and so on. These operations are often extremely costly, sufficiently costly at any rate to prevent many transactions that would be carried out in a world in which the pricing system worked without cost.

R. H. Coase, *The Problem of Social Cost*, 3 J.L. & ECON. 1, 15 (1960) (emphasis added); *see also* Paul J. Heald, *Optimal Remedies for Patent Infringement: A Transactional Model*, 45 HOUS. L. REV. 1165, 1167–68 (2008) (explaining Coase's description of transaction costs "in the context of contracting for inventions"). Discovery costs as used in this article are notably not costs associated with the discovery period before a trial.

7. Clarisa Long has argued that the patent system does not "raise the information costs of searching and avoiding [most patented goods] unduly." Clarisa Long, *Information Costs in Patent and Copyright*, 90 VA. L. REV. 465, 524 (2004). *See id.* at 532-33 ("[Specialized and knowledgeable practitioners] will be able to draw on their preexisting knowledge of goods and technologies in the relevant field and as a result search costs will be lower than if they were not knowledgeable. . . . Requiring a small set of people to search exhaustively is not as socially expensive [in patent law as searches by many are in copyright]."); *see also id.* at 503 ("We would expect legal rules to force disclosure . . . , and . . . increase duties of avoidance, when the class of goods is small . . . , when the goods affect fewer observers, when

Thinking explicitly about discovery costs is a powerful tool for understanding the dysfunctions of the patent system. As we will see, discovery costs are relatively low in pharmaceuticals and other chemical industries.[8] As a consequence, the patent system serves these industries relatively well. In contrast, discovery costs in the software industry are so high that most firms do not even try to avoid infringement.[9] Unsurprisingly, software is a major contributor to the recent spike in patent litigation.

We will argue that this disparity can be explained by the fact that pharmaceutical inventions can be organized by chemical formula, while no analogous organizational scheme exists for software inventions.

I.
SCALE AND BIG-O NOTATION

Our subject is what software engineers call "scalability," the ability of a system to perform well as the "problem size" increases.[10] A common experience for companies that build online services is to have a system that worked flawlessly with a limited number of test users grind to a halt when it is released to the public and used by millions of people.[11] Often, a system's bottlenecks only become apparent when it is used at its full capacity.

He didn't put it in these terms, but Hayek was essentially arguing that central planning doesn't scale. Centralized economic decision-making can work in a small tribe whose chief knows every tribe member personally. But in a modern economy with millions of households, a single, central decision-maker would become a bottleneck, causing the entire system to grind to a halt. So modern societies have developed other mechanisms, such as the price sys-

those observers have greater tolerance for incurring costs of understanding the good, and when the disclosed information is objective and readily verifiable. This is indeed what we see with the patent form."). This paper disagrees with Long's view insofar as this paper argues that the costs of discovering whether an independently-created invention has already been patented are actually unduly high.

8. *See infra* Part III.C.

9. *See infra* Part III.D.

10. *See generally* CAL HENDERSON, BUILDING SCALABLE WEB SITES: BUILDING, SCALING, AND OPTIMIZING THE NEXT GENERATION OF WEB APPLICATIONS, ch. 9 (2006).

11. For example, Twitter has experienced repeated service outages as its user base has grown. *See, e.g.*, Joe Tacopino, *World Cup causes Twitter outages, more 'fail whales' to come*, N.Y. DAILY NEWS, June 17, 2010, http://www.nydailynews.com/news/money/world-cup-twitter-outages-fail-whales-article-1.180774.

tem, to coordinate economic decisions in a decentralized, scalable fashion.

Scalability is also an important issue for the patent system. In an island nation of, say, 100,000 people, the patent system's discovery costs would not be a cause for concern. There might be a few patents granted each week, and it would be reasonable to simply expect every firm to read every patent. In contrast, the U.S. Patent & Trademark Office issues several *thousand* patents per week.[12] Clearly, it is not possible to expect every American firm to read and understand every issued patent. So a scalable patent system needs to offer firms efficient mechanisms to sort through those thousands of patents to find the ones that are relevant to them.

To think rigorously about whether patent searching is scalable, we're going to borrow the standard notation computer scientists use to talk about scalability, called "Big-O" notation.[13] Big-O notation is a way to succinctly summarize how quickly a function grows relative to its input. To illustrate this concept, we'll use the example of a hypothetical chess tournament. Imagine you are planning a chess tournament with n players. You have only one chess set, so games have to be played in sequence. You are deciding between two tournament styles: a round-robin tournament in which every player plays one game with every other player, or a single-elimination tournament in which players are paired off and the loser of each game is dropped from the tournament.

If we assume that each game lasts one hour, and there are no breaks, the round-robin tournament will take $n(n-1)/2$ hours for n players.[14] If n is large, the tournament will be intolerably long. For example, if n=100 the tournament will take more than six months!

Now consider the single-elimination tournament. If we again assume that each game lasts one hour, then the entire tournament will take $n-1$ hours. This is much more manageable; with n=100 players the tournament will take about four days.

12. U.S. Patent Statistics, Calendar Years 1963–2010, UNITED STATES PATENT & TRADEMARK OFFICE, http://www.uspto.gov/web/offices/ac/ido/oeip/taf/us_stat.pdf (noting that 219,614 utility patents were granted in 2010, with an average of 4223 utility patents granted per week).

13. *See* Paul E. Black, *big-O notation*, *in* DICTIONARY OF ALGORITHMS AND DATA STRUCTURES [ONLINE] (Paul E. Black, ed. 2010), http://www.nist.gov/dads/HTML/bigOnotation.html (last updated August 31, 2012); JON KLEINBERG & EVA TARDOS, ALGORITHM DESIGN 36–37, 47–56 (1st ed. 2006).

14. If each of n players played every other contestant, the tournament would take $n(n-1)$ hours. However, this would result in each pair playing each other twice, because after player A challenged player B, player B would then challenge player A. So we divide by two to prevent duplicate games.

To find the "Big-O" class of a function like $n(n\text{-}1)/2$, one expands the function to $^1/_2n^2\text{-}^1/_2n$ and drops everything other than the fastest-growing exponent, leaving n^2.[15] We can then say that the round robin tournament takes $O(n^2)$ time to complete, given n participants. In contrast, the single-elimination tournament takes just $O(n)$ time to complete.

This means that round-robin tournaments scale poorly compared to single-elimination tournaments. Doubling the number of players roughly doubles the length of the single-elimination tournament, while it increases the length of the round-robin tournament by a factor of four.[16] For large tournaments, this effect will dwarf other considerations in choosing between the two options.

II.
ANALYZING THE PATENT SYSTEM'S SCALABILITY WITH BIG-O NOTATION

A. *Simple Model of the Patent System*

Now we use Big-O notation to evaluate the scalability of the patent system. We start by constructing a stylized model of the patent-eligible widget industry. Assume there are n firms producing widgets and each firm produces just one type of widget and holds just one patent. We will relax these assumptions later, but the simplicity of this model makes it a good starting point.

Each firm's widget may infringe multiple competitors' patents, and firms spend resources to learn which patents they must license or invent around. How large are these discovery costs? Suppose that the only known way to find the patents related to a particular widget is to examine all widget patents, and that lawyers can always correctly determine whether a widget violates a patent simply by

15. Factors besides the fastest-growing exponent are dropped because their effect on the length of time it takes to solve a problem becomes mathematically insignificant as the problem size increases.

16. We can demonstrate this by plugging p players and $2p$ players into the equations for determining the length of a single elimination and round robin tournament: $n\text{-}1$ and $^1/_2n^2\text{-}^1/_2n$, respectively. P players could finish a single elimination tournament in $p\text{-}1$ hours and could finish a round robin tournament in $^1/_2p^2\text{-}^1/_2p$ hours. To determine how long it would take $2p$ players to finish a tournament, we replace "n" in the equation with "$2p$". Twice as many players would take almost twice as long to finish a single elimination tournament as p players—$2p\text{-}1$ hours compared to $p\text{-}1$ hours. In contrast, a round robin tournament of $2p$ players would take $^1/_2(2p)^2\text{-}^1/_2(2p)$ hours, which multiplies out to $2p^2\text{-}p$ hours. Whereas the round robin tournament of p players took almost about $^1/_2p^2$ hours, the tournament of $2p$ players takes almost $2p^2$ hours—quadrupling the time even though the players only doubled.

reading the patent claims.[17] Suppose also that it takes a patent lawyer one hour to determine whether a given widget-related product infringes a given widget patent. Then each of the n widget firms will need to pay a patent lawyer to examine the patents held by each of the n-1 other widget firms. That's n-1 hours of work for each firm, leading to an industry-wide discovery cost of $n(n\text{-}1)$, or $n^2 - n$, billable hours. In other words, the discovery costs of the patent system are $O(n)$ for each firm, and $O(n^2)$ across the entire widget industry.

This means that the patent system scales poorly for widgets.[18] If the number of widget firms (and, with it, the number of patents) doubles, the industry's total legal bills increase by a factor of four. In an industry with many firms, the patent system's discovery costs would be a large burden; they could even dwarf some firms' revenues altogether. For example, in a widget industry in which 30,000 firms had one patent apiece and could review one patent per hour, each firm would need to hire around fifteen full-time patent attorneys, resulting an industry-wide total discovery cost of almost a billion billable hours.[19] That's a lot, even for deep-pocketed widget companies.

B. *Indexing Lowers Discovery Costs*

The above analysis assumes that every firm must examine every patent in some detail, but this is not practical in a world where there are hundreds of thousands of valid patents. Ideally, firms in real industries would have ways to quickly find the small number of

17. In the real world, lawyers frequently cannot state with certainty whether a given activity actually infringes a particular patent. In this paper, we will largely set this issue to the side and assume counterfactually that lawyers can always determine whether a particular activity infringes a particular patent in a reasonable amount of time. For further reading on the challenges of claim construction and determining the scope of patents, see Christopher A. Cotropia, *Patent Claim Interpretation and Information Costs,* 9 Lewis & Clark L. Rev. 57 (2005); Christopher A. Cotropia, *Patent Claim Interpretation Methodologies and their Claim Scope Paradigms,* 47 Wm. & Mary L. Rev 49 (2005);,Jeanne Fromer, *Claiming Intellectual Property,* 76 U. Chi. L. Rev. 719 (2009); Michael Risch, *The Failure of Public Notice in Patent Prosecution,* 21 Harv. J. L. & Tech. 179 (2007).

18. Running time can get much worse for computer scientists, who sometimes must solve problems that can only be solved in non-polynomial time, such as $O(k^n)$ for some constant k and number of inputs n.

19. Each of the 30,000 firms would need to hire a patent attorney to examine 29,999 other patents, which takes 1 hour per patent, so attorneys would spend a total of 30,000 * 29,999 * 1 hour = 899,970,000 hours. Assuming a typical attorney bills 2000 hours of work per year, each firm would need just shy of 15 attorneys to examine 29,999 patents.

patents that relate to its own products and ignore the rest. The ability to do this depends on a good system of organization.

We can see the power of organization in the system of real property. Counties do not store real property records in a random order. Rather, they place them in a predictable order based on their geographic location. Filing them in a predictable order allows rapid record retrieval in the same way that alphabetization allows rapid lookup of words in a dictionary. In Big-O terms, retrieving one item from a well-organized collection is roughly an *O(1)*, or constant-time, operation.[20] If a clerk needed to retrieve each record once during the year, he would only spend *O(n)* time sorting through filing cabinets, where *n* is the total number of records.

By way of contrast, we can imagine a county with an incompetent records clerk who placed property records in filing cabinets at random. In this county, the property system would have the same scaling problem as widget patents: as the number of parcels increased, looking up who owned any given parcel would become more and more time-consuming. The only way to find a particular record would be to examine every record, one at a time. That means retrieving all records related to a particular parcel would be an *O(n)* operation. If a clerk needed to retrieve each record once during the year, he would spend $O(n^2)$ time sorting through filing cabinets. In a county with many parcels, the system would be completely unmanageable.

The ability to organize claims to real property such that they can be quickly retrieved depends on the existence of a standardized and predictable representation such as geographic coordinates. If a group of items has such a representation, we call that group "indexable," because the representation makes it possible to build an efficient index of the items. Whether a set is indexable or not depends on its inherent properties. Dictionary words are indexable because they can be organized alphabetically. Real property claims are indexable because they can be organized by their geographic location.

20. Constant-time operations, or operations that take *O(1)* time, take the same amount of time to complete, regardless of the problem size. Depending on the details of the filing method, lookup times might be a slow-growing function like $O(\log_2 n)$. The difference between *O(1)* and $O(\log_2 n)$ is not large, and we're going to pretend that they are the same to simplify the presentation.

C. *Example: Chemical Patents are Indexable*

Are patents indexable? This is a difficult question to answer in general, since the answer varies by technology class. But at least one category of patents—chemical patents—is already being indexed.

The Food and Drug Administration produces a publication called *Approved Drug Products with Therapeutic Equivalence Evaluations*.[21] This publication, colloquially known as the "Orange Book,"[22] allows people to look up pharmaceutical patents based on the chemical formula of the active ingredient. A German organization, FIZ Karlsruhe, offers an electronic database called STN, which allows researchers to pull up all patents and other literature on particular molecules.[23]

Chemical formulas allow efficient retrieval of chemical patents, just as geographic coordinates allow efficient retrieval of real estate records. In other words, chemical patents are indexable. That means that finding a patent based on its chemical formula is approximately an *O(1)* operation, just as it is for real property records.

Recall that doubling the number of widget patents doubled every widget firm's discovery costs, since each firm was forced to look at twice as many patents to weed out the irrelevant ones. In contrast, doubling the number of chemical patents does not increase chemical firms' discovery costs because a database like STN can quickly filter out irrelevant patents, no matter how many of them are in the database. So the patent system scales well for chemicals.

D. *Example: Software Patents Are Disorganized*

Unfortunately, few, if any, non-chemical patents seem to be indexable. To be sure, there are searchable databases that include non-chemical patents.[24] But few, if any, non-chemical categories of patentable inventions have a standardized and comprehensive scheme for classifying patentable subject matter. That means that there may not be a faster way to find all patents relevant to a partic-

---

21. U.S. DEPT. OF HEALTH & HUMAN SERVS. ET AL., APPROVED DRUG PRODUCTS WITH THERAPEUTIC EQUIVALENCE EVALUATIONS (32d ed. 2012) [hereinafter "Orange Book"], *available at* http://www.fda.gov/downloads/Drugs/DevelopmentApprovalProcess/UCM071436.pdf.

22. *Id.* at iv (describing the origins of the nickname "Orange Book").

23. *See* STN INTERNATIONAL, http://www.stn-international.de/index.php?id=123 (last visited Dec. 10, 2011).

24. *See, e.g.*, DELPHION, http://www.delphion.com (last visited Jan. 18, 2012) (*cited in* Jeanne C. Fromer, *Patent Disclosure*, 94 IOWA L. REV. 539, 585 n.208 (2009)); PATBASE, http://www.patbase.com (last visited Jan. 18, 2012).

ular product than to simply examine all of the patents in a particular technology class, one at a time.

We will focus on software patents, which provide a particularly good illustration of the problem.

As we have already noted, it would be completely impracticable for a firm to read every patent. So patent lawyers use a variety of methods to guess at what patents they should look at. These methods include searching by keyword, patent classification,[25] inventor or patent assignee, and searching for patents that cite to and are cited in similar patent applications.[26]

The keyword search is crucial, but searching by keyword hardly approaches the speed and certainty of searching an indexable system. A firm producing a new word processor might search for patents containing phrases like "word processor," "page layout," "printing," and so forth. But in the absence of a precise, standardized scheme for classifying software inventions, patent applicants are free to use any terms they like—or even make up new ones—to describe their software inventions. The scope of a patent's claims will not always be obvious from a patent's title or abstract.[27] And a single software patent can claim multiple applications that are only loosely connected to each other.[28]

One particularly illustrative example of the limits of keyword searching is U.S. Patent No. 4,528,643—a "System for Reproducing Information in Material Objects at a Point of Sale Location."[29] The

25. The patent classification system tries to categorize patented inventions as particular types. The Patent and Trademark Office assigns classification numbers based only on the patent's claims, rather than the entire application. *See* U.S. PATENT & TRADEMARK OFFICE, U.S. DEPARTMENT OF COMMERCE HANDBOOK OF CLASSIFICATION 9 (2005), *available at* http://www.uspto.gov/web/offices/opc/documents/handbook.pdf (*cited in* Fromer, *supra* note 24, at 586 n.212). John R. Allison and Mark A. Lemley have discovered "numerous instances of what seem to be wrong or arbitrary classification decisions." John R. Allison & Mark A. Lemley, *The Growing Complexity of the United States Patent System*, 82 B.U. L. REV. 77, 92 (2002) (cited in Fromer, *supra*, at 586). They note, "This problem is compounded by the fact that some patents involve more than one type of technology, so that classifying them into only a single category will necessarily mischaracterize them to some extent." *Id.* at 92 n.49.

26. In addition to the USPTO website, numerous private companies provide searchable databases of patents and related information. *See, e.g.*, DELPHION, *supra* note 24; PATBASE, *supra* note 24.

27. The purpose of a patent abstract is to merely "enable the United States Patent and Trademark Office and the public generally to determine quickly from a cursory inspection the nature and gist of the technical disclosure." 37 C.F.R. § 1.72(b) (2012).

28. *See* 37 C.F.R. § 1.75(b) (2012).

29. U.S. Patent No. 4,528,643, at [54] (filed Jan. 10, 1983).

invention that gave rise to the '643 patent was "a kiosk or vending machine to be used in retail locations for producing digital music tapes or other digital reproductions."[30] The patent owner attempted to enforce the patent against several software and publishing companies, claiming that the sale and transfer of software and documents over the Internet infringed the patent.[31] The '643 patent was filed in 1983, long before "e-commerce" existed; indeed, the web page and browser were not created until 1990.[32] Nonetheless, litigation over the meaning of the '643 patent took over seven years to complete.[33]

The '643 patent illustrates the difficulty with relying on keywords to search for patents. The "Background of the Invention" section of the patent discusses "retail outlets (point of sale locations)" and their difficulty deciding which recordings to stock.[34] The patent makes no mention of the Internet or personal computers. An attorney trying to determine *ex ante* whether the process of selling and transferring software over the Internet had been patented would be unlikely to discover the '643 patent by conducting a keyword search.[35]

---

30. *See* JAMES BESSEN & MICHAEL J. MEURER, PATENT FAILURE 194 (2008). Readers who find themselves confused as to what invention this patent describes are in good company. The exact scope of the '643 patent is not clear, even after years of litigation. The original application likely referred to a machine that could burn custom music recordings so the store would not have to keep them in stock.

31. *See* Interactive Gift Exp., Inc., v. Compuserve Inc., 256 F.3d 1323, 1329 (Fed. Cir. 2001). *Catalina Marketing International, Inc. v. Coolsavings.com, Inc.*, 289 F.3d 801, 806–07 (Fed. Cir. 2002), presented a similar fact pattern.

32. *See* TIM BERNERS-LEE, WEAVING THE WEB 28–30 (1999).

33. *See* Order of Dismissal, Interactive Gift Exp., Inc. v. Compuserve Inc., No. 95-6871 (S.D.N.Y. Jan. 21, 2003), ECF No. 224; Complaint, Interactive Gift Exp. Inc., v. Compuserve Inc., No. 95-6871 (S.D.N.Y. July 25, 1995), ECF No. 1. The Federal Circuit had held the term "point of sale location" could include one's home. *See* Interactive Gift Exp., 256 F.3d at 1335. However, it also held that the "material objects" referenced in the patent did not include a buyer's personal computer because the objects had to be offered for sale, removed from the device that wrote information to them after purchase, and intended for use on a device other than that which wrote information to them. *Id.* at 1338. Nonetheless, litigation following the Federal Circuit decision dragged on for one and a half years, concluding with voluntary dismissals and numerous settlement agreements. *See* Docket, Interactive Gift Exp. Inc., v. Compuserve Inc., No. 95-6871 (S.D.N.Y. Nov. 7, 2001), ECF Nos. 184–86, 189, 201, 220–24 (dismissing parties, often after having reached a settlement).

34. U.S. Patent No. 4,528,643 col. 1 ll 6–8; col. 2 ll 13–15 (filed Jan. 10, 1983).

35. At first blush, one might suppose this problem could sometimes be averted by the reverse doctrine of equivalents. "The reverse doctrine of equivalents is an equitable doctrine designed to prevent unwarranted extension of the claims beyond a fair scope of the patentee's invention." Roche Palo Alto LLC v. Apotex,

This problem is exacerbated by the complexity of software products. Real software products often contain thousands—sometimes even millions—of lines of code.[36] Given that a handful of lines of code can constitute a patent-eligible invention, the number of potentially patentable inventions in a 100,000-line computer program can be very large. For example, the patent on raising a pop-up browser window when one attempts to leave a webpage[37] can be infringed by writing a mere three lines.[38]

Hence, it is extremely difficult to anticipate all of the different aspects of a particular computer program that might be regarded as patent-eligible subject matter. It is even more difficult to anticipate all of the terms patent applicants could use to describe those various patentable concepts. The effectiveness of keyword searching is further undermined by the doctrine of equivalents, which holds that a patent can cover subject matter "equivalent" to its claims even if it does not fall within their literal scope.[39] This means that in

Inc., 531 F.3d 1372, 1377 (Fed. Cir. 2008) (internal quotation marks omitted). "[W]here a device is 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 the claim, the [reverse] doctrine of equivalents may be used to restrict the claim and defeat the patentee's action for infringement." Graver Tank & Mfg. Co. v. Linde Air Prods. Co., 339 U.S. 605, 608–09 (1950) (citation omitted). However, the reverse doctrine of equivalents is "all but defunct." *See* Long, *supra* note 7, at 519 (describing how the United States Court of Appeals for the Federal Circuit "proudly declare[d] that it has struck down every successful assertion of the reverse doctrine of equivalents" (citing Tate Access Floors, Inc. v. Interface Architectural Res., 279 F.3d 1357, 1368 (Fed. Cir. 2002)).

36. *See, e.g.*, MICROSOFT, *A History of Windows*, http://windows.microsoft.com/en-US/windows/history (last visited July 1, 2012) (noting that Windows XP was compiled from 45 million lines of code); Stewart Brand, *The Physicist*, WIRED MAG. (Sept. 1995), *available at* http://www.wired.com/wired/archive/3.09/myhrvold.html (interview with Nathan Myhrvold, estimating Microsoft Word consisted of two million lines of code in 1995).

37. U.S. Patent No. 6,389,458 (filed Oct. 30, 1998).

38. *See* BEN KLEMENS, MATH YOU CAN'T USE 1–2 (2006). The three lines of code necessary to create a pop up window when one attempts to leave a webpage in JavaScript are:

```
function onExit() {
    popup = window.open ("pop.html", "Don't go!") ;
    popup.focus();}
```

Id. at 2.

39. A product or process may infringe a patent under the doctrine of equivalents if it performs "substantially the same function in substantially the same way to obtain the same result" as the patented invention. Graver Tank Mfg. Co., 339 U.S. at 608 (quoting Sanitary Refrigerator Co. v. Winters, 280 U.S. 30, 42 (1929)); *see also* Warner Jenkinson Co. v. Hilton Davis Chem. Co., 520 U.S. 17, 40

practice, a keyword-based patent search will either only find a fraction of the relevant patents, or produce so many results that it would be of little help to the searcher.[40]

Other search strategies, such as searching by inventor, assignee, or citations in related patents, are no more promising. If a firm knows of an existing patent similar to its product, these methods may be useful for finding closely-related patents. But as we have seen, many different aspects of a software product may be patent-eligible, and there is no reason to think that all the patents relevant to a particular product will be linked together by citations, common inventors, or other similarities.

It is theoretically possible that future improvements in artificial intelligence will allow the creation of a search engine for software patents as powerful as conventional chemical patent databases. This search engine would have to be sophisticated enough to analyze a real-world machine or process, make a comprehensive list of characteristics that could constitute patent-eligible subject matter, produce a list of all possible terms that could be used to describe this subject matter, and find all patents that use these terms in a way that indicates possible infringement. But that technology doesn't

(1997) (explaining that the essential inquiry under the doctrine of equivalents is whether "the accused product or process contain[s] elements identical or equivalent to each claimed element of the patented invention"). John R. Allison and Mark A. Lemley argue, "[T]he doctrine of equivalents was . . . near death by the late 1990s . . . [and] district courts are more likely to reject doctrine of equivalents claims today than ever before." John R. Allison & Mark. A. Lemley, *The (Unnoticed) Demise of the Doctrine of Equivalents*, 59 STAN. L. REV. 955, 958 (2007); *see also* Lee Petherbridge, *On the Decline of the Doctrine of Equivalents*, 31 CARDOZO L. REV. 1371 (2010); David L. Schwartz, *Explaining the Demise of the Doctrine of Equivalents*, 26 BERKELEY TECH. L.J. 1157 (2011). Nonetheless, Samson Vermont points out that "one of every four or five cases in which a patentee wins a judgment of infringement is . . . a judgment of infringement under the doctrine of equivalents . . . . [Doctrine of equivalents] scope is litigated frequently. One of every two decisions on infringement is a decision on [doctrine of equivalents] infringement." Samson Vermont, *Taming the Doctrine of Equivalents in Light of Patent Failure*, 16 J. INTELL. PROP. L. 83, 85 (2008).

40. Although keyword searches will find many relevant patents, finding only some relevant patents will not insulate an inventor from lawsuits or create incentives to license or design around found patents. If you can only find 50% of the patents on which your invention might infringe, there is little value in licensing or designing around those patents because you can still be sued by the owners of the other 50% of patents you did not find. Patent searching is not necessarily like searching for legal cases where the cases are similar and related to each other, and where, after a point, finding each new case produces diminishing returns. The first and last patent you find are equally likely to bring an accidental infringer economic ruin.

exist right now, and we are skeptical it will exist any time soon. That means the only reliable way to find all—not merely some—patents infringed by a particular software product is to have a human being look at all patents in software-related technology classes. So the discovery costs of software patents for a single firm is roughly *O(n)* in the number of software patents, not *O(1)* as with chemical patents.

The fact that the average firm has patent discovery costs that are *O(n)* in the number of patents is not a problem if the number of firms and patents are both small. But the more firms and patents there are, the larger the discovery costs will be.

Once again, the software industry is a good example of an industry where the patent system works poorly. The number of firms producing patentable software is massive—much larger than the number of firms in the software industry as it is conventionally defined.[41] Almost every medium and large American firm has an information technology ("IT") department that performs backups, runs file and mail servers, runs the firm's website, and so forth.[42] IT professionals routinely create software to automate such tasks, and this software is potentially patent-eligible. Many firms also develop custom software to automate common business processes, and some of it is quite complex.[43] Hence, most medium and large American firms (as well as many non-profits, universities, and other organizations) are in the software industry as far as patent law is concerned.[44]

And as a consequence, many kinds of firms are the targets of software patent lawsuits.[45] One complaint charged the Green Bay Packers, Caterpillar, Peapod, OfficeMax, and Kraft Foods with in-

41. *See* KLEMENS, *supra* note 38, at 92.

42. *Id.* at 93.

43. *Id.*

44. *Id.* at 4–5 ("[A] patent on a piece of code is a restriction . . . on the information technology department of every company in America, not to mention every person who writes macros to facilitate his or her work . . . .").

45. Even frivolous allegations of patent infringement can be very expensive to dispel. In 2005, the average cost of an opinion letter assessing the validity of a patent and whether an accused party infringed was about $24,000. AM. INTELL. PROP. LAW ASS'N, REPORT OF THE ECONOMIC SURVEY 102 (2005) (*cited in* Matthew Sag & Kurt Rohde, *Patent Reform & Differential Impact*, 8 MINN. J.L. SCI. & TECH. 1, 10 n.41 (2007)). Getting a patent invalidated in court costs on average $650,000. *Id.* at 108 (*cited in* Sag & Rohde, *supra*, at 10 n.44). It is thus often economically rational to pay high licensing fees for invalid patents that one did not even infringe rather than have to participate in a lawsuit. "This is the real perversity of the current patent system: rational actors will pay licensing fees for patents they strongly suspect are either invalid, or simply do not apply to them, because each of the alternatives is worse." Sag & Rohde, *supra*, at 11.

fringing a patent for a "Remote Query Communications System"[46]—specifically for having JPEG images on their websites.[47] Another plaintiff filed lawsuits against firms such as J. Crew and Linens 'N Things for infringing its "Information Processing Methodology" patents[48] by transmitting data that customers entered on the defendants' websites.[49] Other firms facing allegations of software patent infringement include McDonalds, Barnes & Noble, Jamba Juice, Aeropostale, 7-Eleven, and Oprah Winfrey's Harpo Productions.[50]

Not only do firms outside of the conventional software industry frequently produce potentially infringing software, they are also granted the lion's share of software patents. James Bessen has

46. U.S. Patent No. 5,253,341 (filed Apr. 11, 1991).

47. Amended Complaint at 3, Global Patent Holdings, LLC v. Green Bay Packers, Inc., No. 00-4623 (N.D. Ill. Oct. 23, 2007), ECF No. 50. An attempt to enforce the '341 patent in 2000 led to the invalidation of all sixteen claims, as well as to the addition of a seventeenth claim. *See* Motion to Reinstate at 1, Techsearch LLC v. Internet Entm't Grp., No. 00-4623 (N.D. Ill. Sept. 10, 2007), ECF No. 41; *see also id.* at Ex. A (*Ex parte* Global Patent Holdings, LLC, Appeal No. 2006-0698, Reexamination Control No. 90/005,742 (Dec. 26, 2006)). The amended complaint, *supra,* charged the Green Bay Packers and other parties with violating the new seventeenth claim, however an initial re-examination of the seventeenth claim found it invalid on nineteen grounds. *See* Initial Office Action in Ex Parte Reexamination of U.S. Patent No. 5,253, 341, at 3-5 (July 22, 2008), *available at* http://www.scribd.com/doc/4328073/jpg-patent-reexam. The case was dismissed without prejudice pending the reexamination on March 4, 2009. *See* Minute Entry, Global Patent Holdings LLC v. Green Bay Packers, No. 00-4623 (N.D. 2009), ECF No. 154; Agreed Motion to Dismiss, Global Patent Holdings, LLC v. Green Bay Packers, Inc., No. 00-4623, 2009 WL 3059752 (N.D. Ill. 2009), E.C.F. No. 153.

48. U.S. Patent No. 7,184,162 (filed Apr. 15, 2005); U.S. Patent No. 7,075,673 (filed Nov. 6, 2003); U.S. Patent No. 6,683,697 (filed Dec. 9, 1999).

49. *See* Complaint at ¶¶ 14, 17, 20, Eon-Net L.P. v. J. Crew Inc., No. 07-10488 (S.D.N.Y. Nov. 20, 2007), ECF No. 1; Amended Complaint at ¶¶ 12, 15, Eon-Net, L.P. v. Linens 'N Things, Inc., No. 06-315 (D.N.J. Aug. 2, 2006), ECF No. 15. The cases were dismissed or settled before the court rendered a final judgment. *See* Stipulation and Order of Dismissal with Prejudice, Eon-Net, L.P. v. J. Crew Inc., No. 07-10488 (S.D.N.Y. Aug. 4, 2008), ECF No. 11; Stipulation of Dismissal, Eon-Net, L.P. v. Linens 'N Things, No. 06-315 (D.N.J. Jan. 5, 2007), ECF No. 23; Stipulation of Dismissal, Eon-Net, L.P. v. Linens 'N Things, No. 06-315 (D.N.J. Dec. 22, 2006), ECF No. 22.

50. *See* Complaint, Card Activation Techs., Inc. v. 7-Eleven Inc., No. 10-4984 (N.D. Ill. Aug. 9, 2010), ECF No. 1; Complaint, Illinois Computer Research, LLC v. Harpo Productions, Inc., No. 08-7322 (N.D. Ill. Dec. 23, 2008), ECF No. 1; Complaint, Card Activation Techs., Inc. v. Barnes & Noble, Inc., No. 07-1230 (N.D. Ill. Mar. 2, 2007), ECF No. 1 (also naming Jamba Juice Co. and Aeropostale Inc. as defendants); Complaint, Card Activation Techs., Inc. v. McDonald's Corp., No. 06-5578 (N.D. Ill. Oct. 13, 2006), ECF No. 1.

found that 83% of software patents granted to public firms in 2006 went to firms outside of the conventional software industry.[51]

In our discussion of the widget industry, we stipulated that each firm would hold exactly one patent and concluded that the industry-wide discovery costs of widget patents are $O(n^2)$. Obviously, real industries aren't like that. Some firms have many patents and others have none at all. So it's more precise to say that the patent system's discovery costs in non-indexable industries are $O(n_f n_p)$, where n_f is the number of firms and n_p is the number of patents. The widget industry in our example is a special case where $n=n_f=n_p$, so that total discovery costs are $O(n^2)$.

In the software industry, $n_f \neq n_p$, but n_f and n_p are both large. As we have seen, most medium and large firms produce patent-eligible software. There are roughly 635,000 firms in the United States with twenty or more employees.[52] While not all of these firms produce software, many of the 1.7 million firms with five to nineteen employees do, so we'll estimate the number of firms that create software, n_f, to be 600,000 firms. And n_p, the number of software patents issued, is around 40,000 in a typical year (and growing).[53] That means that there are around twenty-four billion new patent-firm pairs each year that could produce accidental infringement. Even if a patent lawyer only needed to look at a patent for ten minutes, on average, to determine whether *any* part of a particular firm's software infringed it,[54] it would require roughly two million patent attorneys, working full-time, to compare every firm's prod-

51. James Bessen, *A Generation of Software Patents*, 18 B.U. J. SCI. TECH. L. 241, 256 (2012) (showing only 17.2% of software patents granted to public firms were granted to firms in the computer services and software industries).

52. According to the U.S. Census Bureau, there were 635,000 businesses with twenty or more employees in 2008, and 1.7 million firms with five to ninteen employees. *See Statistics About Business Size (including Small Business) from the Census Bureau*, CENSUS BUREAU HOMEPAGE, http://www.census.gov/econ/smallbus.html (last visited Dec. 9, 2011).

53. *See* Bessen, *supra* note 51, at 253.

54. Ten minutes is an unrealistically low amount of time. Patentable software can be written in only a few lines, *see supra* note 27, and many software programs consist of millions of lines of code. *See, e.g.*, MICROSOFT *supra* note 36 (noting that Windows XP was compiled from 45 million lines of code); Brand, *supra* note 36 (interview with Nathan Myhrvold, estimating Microsoft Word consisted of two million lines of code in 1995). It is plainly beyond human capacity for an attorney to be able to hold in his or her mind everything that a large software program does, let alone to compare it to the content of a patent in a matter of minutes.

ucts with every patent issued in a given year.[55] At a rate of $100 per hour, that would cost $400 billion. For comparison, the software industry was valued at $225.5 billion in 2010.[56]

Obviously, $400 billion is a highly speculative figure. But the exact number doesn't matter because there are only around 40,000 registered patent attorneys and agents in the United States.[57] Even if the entire patent bar worked full-time on patent clearance for software firms, there wouldn't be nearly enough lawyers to go around.

E. *Example: Discovery Costs for Corkscrews Are Low Because There Are Few Corkscrew Patents*

We have argued that discovery costs are low for the pharmaceutical industry because chemical patents are indexable by chemical formula. But even non-indexable patent classes can have modest discovery costs if n_p, the number of potentially-relevant patents, is small enough.

For example, consider corkscrews. A search of corkscrew-related technology classes[58] reveals that 301 utility patents were awarded between 1992 and 2011. Just five of these were issued in 2011.

We have argued that it would be impossible for anyone to read and understand the hundreds of thousands of existing software patents, or even to keep up with the hundreds of software patents the patent office issues each week. But it would only take a few weeks to read and understand the 301 utility patents related to corkscrews.[59] Given that only about fifteen patents are issued in corkscrew-related technology classes in a typical year, it would be fairly easy for an

---

55. The math behind this is straightforward: 40,000 patents*600,000 firms*(10 minutes per patent-firm pair)/(2,000 hours of work per attorney *60 minutes per hour)=2 million attorneys.

56. *Software: Global Industry Guide 2010*, DATA MONITOR RESEARCH STORE (2012), *available at* http://www.datamonitor.com/store/Product/software_global_industry_guide_2010?productid=4F026C5C-EBCC-4193-AD27-77260196E7F5.

57. *See Patent Attorney/Agent Search*, UNITED STATES PATENT AND TRADEMARK OFFICE, https://oedci.uspto.gov/OEDCI/query.jsp (search for "US" in the country field) (last visited Dec. 11, 2011).

58. We used patent classes 81/3.2, 81/3.4, 81/3.7, 81/3.9, 81/3.29, 81/3.36, 81/3.37, 81/3.45, 81/3.48. *See e.g.*, Full-Text and Image Database, UNITED STATES PATENT & TRADEMARK OFFICE (current through Oct. 16, 2012), http://patft1.uspto.gov/netahtml/PTO/search-bool.html.

59. For example, if we assume that it takes an hour, on average, to understand a corkscrew-related patent, then it would take approximately eight forty-hour workweeks to familiarize oneself with all 301 corkscrew-related utility patents.

attorney who specialized in corkscrew patents to keep abreast of new patents.

In March 2012, Amazon.com listed 4,551 corkscrews for sale in its "home and kitchen" section, made by 737 manufacturers.[60] Not every firm manufacturing corkscrews is listed on Amazon.com, but it's reasonable to assume a majority of the commercially significant ones are. So we'll estimate that n_f, the number of firms in the corkscrew industry, is no more than 1,500.

We have already estimated that n_p=301 for corkscrews. If we again assume that each firm-patent comparison takes ten minutes, then it would take approximately 75,250 hours to conduct patent clearance for all corkscrews currently on the market.[61] That would require the services of approximately forty patent attorneys working full-time for a year.

And of course, this process wouldn't need to be repeated every year. In a typical year, firms would only need to clear their new products and compare their existing products with newly-issued patents. So in a typical year, the corkscrew industry would require the services of significantly fewer than forty full-time patent attorneys.

Moreover, the fact that there are few enough patents that a single person could read and understand all of them means that the vetting process is likely to be considerably more efficient for corkscrews than for software. A patent is much harder to understand the first time it's read than on the second, fifth, or twentieth encounter. There are so many software patents that no one could possibly read more than a small fraction of them, so attorneys doing software patent clearance spend most of their time reading patents for the first time. In contrast, attorneys that specialize in corkscrews would be looking at the same 301 patents over and over again. Their familiarity with the corkscrew patents would allow them to quickly identify which were relevant to a particular client's products.

Hence, the fact that a patent class is non-indexable does not necessarily mean that discovery costs will be prohibitively high. If

60. AMAZON.COM, http://www.amazon.com/ (last visited March 2012) (search for "corkscrews" in the category "Home and Kitchen"; click on "see more" hyperlink under "brands.").

61. 301*1,500=451,500 comparisons. If each takes ten minutes, this will take 451,500/6=75,250 hours of work. Obviously, 10 minutes per comparison is a rough estimate, but one that suffices to illustrate the point that corkscrew patent discovery costs are several orders of magnitude smaller than software patent discovery costs.

n_p, the total number of patents in that technology class, is small, then the "brute force" approach of examining every patent may be feasible.

III.
DISORGANIZED PATENTS HAVE LED TO A LITIGATION EXPLOSION

In practice, firms don't—and can't—spend whatever it takes to avoid infringement. Rather, they spend only as much money on patent searches as they believe will "pay off" in lower future litigation and licensing costs.[62] In industries with low discovery costs, a rational firm is likely to spend enough money to find all patents relevant to its products. Inadvertent infringement in these industries is rare. On the other hand, in industries where discovery costs are high, the rational firm might not even try to avoid infringement, because a dollar spent on patent searches will produce much less than a dollar in savings due to reduced litigation.

Unsurprisingly, the software industry—and, indeed, the larger IT industry of which it is a part—is in the latter category. In a widely-cited paper, Mark Lemley documents the widespread IT industry practice of ignoring patents and tries to explain why IT firms behave as they do.[63] He suggests several explanations: patent negotiations take a long time, patent holders may not be willing to offer reasonable terms, many patents turn out to be invalid, and the number of patents a given firm must license would be large.[64] These are all plausible explanations, but there is a more important and fundamental one: firms have no cost-effective way of obtaining a complete list of relevant patents in the first place. Licensing the few they know about provides no protection against the many others they have not yet discovered.

Lemley notes that the pharmaceutical industry is one of the few industries that does not ignore patents, and he attributes the difference to the fact that the FDA forces patent holders to disclose relevant patents.[65] But such disclosures can only be compiled into a useful form because chemical patents are indexable. Without chemical formula as an organizational scheme, it would not be possible

62. *See* Rebecca S. Eisenberg, *Patent Costs and Unlicensed Use of Patented Inventions*, 78 U. Chi. L. Rev. 53, 55 (2011) ("Information costs and transaction costs may dwarf potential gains to users from identifying and clearing rights . . . .").

63. Mark A. Lemley, *Ignoring Patents*, 2008 Mich. St. L. Rev. 19, 21 (2008).

64. *Id.* at 25–29.

65. *Id.* at 29–30.

to create a publication like the Orange Book that allows rapid retrieval of patents relevant to a particular real-world product.

The empirical evidence on litigation costs is consistent with our hypothesis about the importance of discovery costs. We should expect industries with high discovery costs to have high rates of inadvertent infringement and, as a consequence, high rates of litigation. In their 2008 book *Patent Failure,* James Bessen and Mike Meurer used stock market event studies to estimate the total costs of patent litigation for various industries during the 1980s and 1990s. They found that litigation costs for chemical patents are much lower than the profits from these patents.[66] This happy state of affairs can be explained by the low discovery costs of chemical patents—litigation is rare because infringement is rare.

Bessen and Meurer found litigation costs were much higher for non-chemical patents. From 1984 until 1994, the costs of litigation over non-chemical patents were roughly equal to the profits from those patents.[67] And from 1994 to until the end of their study period in 1999, the costs of litigation over non-chemical patents increased dramatically.[68]

Why was there a spike starting in the mid-1990s? Between 1989 and 1998, courts made it dramatically easier to obtain patents on software and business methods.[69] Bessen and Meurer found that these patents contributed a disproportionate share of patent litigation. Software patents were more than twice as likely to be involved in patent litigation as other kinds of patents.[70] The closely-related category of business method patents was nearly seven times as likely to be involved in litigation.[71] This is not surprising. We have already seen that software patents have particularly high discovery costs, and business method patents have high discovery costs for similar reasons.

66. *See* BESSEN & MEURER, *supra* note 30, at 139 fig.6.5.

67. *Id.* at 138–39.

68. *Id.*

69. The Federal Circuit began upholding patents on software in *In re Iwahashi*, 888 F.2d 1370, 1374 (Fed. Cir. 1989), and *In re Alappat*, 33 F.3d 1526, 1545 (Fed. Cir. 1994) (en banc). In 1998, the Federal Circuit eliminated the common-law ban on business method patents in *State Street Bank & Trust Co. v. Signature Fin. Grp.*, Inc., 149 F.3d 1368, 1375 (Fed. Cir. 1998). For a more general discussion of the expansion of patentable subject matter to cover software, see ROBERT PATRICK MERGES & JOHN FITZGERALD DUFFY, PATENT LAW & POLICY 151–55 (4th ed. 2007).

70. BESSEN & MEURER, *supra* note 30, at 22, 153 fig.7.2 .

71. *Id.*

IV.
POLICY SUGGESTIONS: BEYOND PATENT QUALITY

We have shown that for technology classes with high discovery costs, the patent system is unlikely to ever work like a traditional property system. Rampant infringement is inevitable because firms have no way to discover which patents they are infringing. Because of this, firms have little incentive to even attempt to clear patent rights before introducing new products into the market.

No one would claim this state of affairs is ideal, but is this system the best we can do?

We don't think so. The patent system is supposed to be a mechanism for promoting the progress of the useful arts by transferring resources from the users of technologies to their inventors. But in industries where discovery costs are high, it does so in an erratic, wasteful, and unjust fashion. The system resembles a lottery more than a system of property rights—an unlucky minority of infringers is the target of ruinous lawsuits, and only a minority of patent owners "win" by catching infringers who pay up.

Not only is this unfair to the targets of these lawsuits, but it creates a generalized disincentive to innovate. Developing new products comes with the risk of incurring crippling liability and having one's business enjoined—precisely the opposite of the effect patents are supposed to have. And, of course, many of the resources consumed by the patent system flow not to inventors, but to pay patent attorneys and cover the patent system's other deadweight costs.

Many observers have argued that patent law reforms should focus on increasing patent "quality."[72] They usually mean that more patents should be invalidated for obviousness or non-novelty and that patents should have narrower and clearer claims. These are worthwhile goals to be sure. But they do not directly address the discovery cost problem.

There is no reason to think obvious or non-novel patents cost more to discover than "high quality" patents. So reforms that invalidate "low quality" patents reduce discovery costs only because they

72. *See, e.g.*, U.S. PATENT & TRADEMARK OFFICE, 2010–2015 STRATEGIC PLAN 9–25 (2010), *available at* http://www.uspto.gov/about/stratplan/USPTO_2010-2015_Strategic_Plan.pdf (describing initiatives to improve patent examination timelines and patent quality); ADAM B. JAFFE & JOSH LERNER, INNOVATION AND ITS DISCONTENTS 171, 178–81 (2004) (discussing ways to improve patent quality); John R. Allison & Starling D. Hunter, *On the Feasibility of Improving Patent Quality One Technology at a Time: The Case of Business Methods*, 21 BERKELEY TECH. L.J. 729 (2006).

reduce the total number of patents being issued. That is valuable, of course, but really dramatic changes would be needed to invalidate enough patents to bring discovery costs under control in large, non-indexable industries like software.

Nor are high discovery costs primarily due to unclear patent boundaries. Unclear boundaries do raise discovery costs, since they require patent attorneys to spend more time examining each patent. But recall that we began our analysis with the counterfactual assumption that widget patents have perfectly clear boundaries. So discovery cost problems will crop up even where patent boundaries are crystal clear. Problems related to unclear patent boundaries exist in addition to the basic discovery cost problems that are the focus of this article.

There are several strategies the government could take to reduce discovery costs, ranging from making small changes to radically restructuring the patent system. Here we look at a few categories of policy changes that would lessen discovery costs to various degrees.

A. *Subject Matter Restriction*

The preceding analysis suggests that the patent system will work well when there is a clear and comprehensive way to index patents—reducing the entire industry's patent discovery time to $O(n_f)$ rather than $O(n_f n_p)$. And the patent system will also work better when n_f or n_p is very small—i.e., in industries with a small number of firms or inventions.

Together, these criteria arrange technologies on a spectrum. At one end of the spectrum are pharmaceuticals, an industry that is highly concentrated, has relatively few inventions, and can use chemical formulas to organize its patents. At the other end of the spectrum is software, an industry that is highly decentralized, produces many patents, and has no standardized classification system.

Notice that the factors that make software patents work poorly are characteristics of software itself and the software *industry*, not the patent system. This suggests that it is probably not possible to make the patent system work well for software. The basic problem is that there is a massive number of firms producing potentially-infringing software, a massive number of software patents, and no systematic way to organize them all. That is, discovery costs grow as $O(n_f\ n_p)$, and n_f and n_p are both large numbers. Changes to patent law probably can not make software inventions indexable, and policymakers certainly should not try to reduce the number of firms.

The fact that some categories of patentable subject matter have much higher discovery costs than others suggests an obvious policy response: exclude subject matter with high discovery costs from patent eligibility.[73] The strongest version of this reform would be to exclude all non-indexable technology classes from patent eligibility.[74] Or policymakers might exclude only those non-indexable categories for which litigation costs are highest.

The problem of high discovery costs provides a strong rationale for courts' traditional prohibition on patents on abstract ideas.[75] The more abstract an invention is, the more different par-

---

73. Excluding inventions with high discovery costs from patentability will not necessarily have deleterious effects. There are many arguments that software patents are not necessary to incentivize software development and that copyright, a sui generis regime, or no protection at all would be sufficient to encourage new innovation in software. These arguments are beyond the scope of this paper, but for more discussion, *see generally* KLEMENS, *supra* note 38, at 17–23; Pamela Samuelson, et al., *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 COLUM. L. REV. 2308, 2404 (arguing for a sui generis intellectual property regime to protect software); Pamela Samuelson, *Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions*, 39 EMORY L.J. 1025, 1148–53 (1990); Timothy B. Lee, *Patently Absurd - Copyright Law Can Meet the Needs of Software Developers*, NAT'L REV. (Oct. 3, 2011), *available at* http://www.cato.org/publications/commentary/patently-absurd-copyright-law-can-meet-needs-software-developers; Timothy B. Lee, *The Case against Literary (and Software) Patents*, TECHKNOWLEDGE (August 28, 2009), *available at* http://www.cato.org/publications/techknowledge/case-against-literary-software-patents; Wendy Seltzer, *Software Patents and/or Software Development* (Har. Univ. – Berman Ctr. for Internet & Soc. TPRC 2011, Sept. 24, 2011), *available at* http://ssrn.com/abstract=1985780 (arguing software patents retard software development).

74. The patent system is often described as a "bargain with the public in which the inventor gives information about the invention in exchange for an exclusive right." Mark A. Lemley, *The Myth of the Sole Inventor*, 110 MICH. L. REV. 709 745 (2012) (citing Eldred v. Ashcroft, 537 U.S. 186, 225 (2003) (referring to a patent as a "quid pro quo" for disclosure)); *see also* Aronson v. Quick Point Pencil Co., 440 U.S. 257, 262 (1979) (identifying the fact that the patent system "promotes disclosure of inventions" as one of its key functions); Fromer, *supra* note 24, at 542; Lisa Larrimore Ouellette, *Do Patents Disclose Useful Information?*, 25 HARV. J.L. & TECH. 531, 532 (2012). If non-indexable patents are, in practice, undiscoverable, then the justification behind this "bargain" is significantly diminished because even fully disclosed patents would be relatively obscured among the many other nonindexable patents.

75. *See* Diamond v. Diehr, 450 U.S. 175, 185 (1981) ("Excluded from such patent protection are laws of nature, natural phenomena, and abstract ideas.") (citations omitted); Diamond v. Chakrabarty, 447 U.S. 303, 309 (1980) ("The laws of nature, physical phenomena, and abstract ideas have been held not patentable.") (citations omitted); Parker v. Flook, 437 U.S. 584, 589 (1978) ("'Phenomena of nature, though just discovered, mental processes, and abstract intellectual concepts are not patentable . . . .'") (quoting Gottschalk v. Benson, 409 U.S. 63, 67 (1972)).

ties are likely to use it for different purposes, and the more flexibility parties will have to describe it.[76] All of these factors mean that more abstract patents will produce particularly high discovery costs and, as a consequence, particularly high rates of inadvertent infringement and litigation.

At one time, software and business method patents were considered too abstract to constitute patentable subject matter, but the United States Court of Appeals for the Federal Circuit took a more permissive stance towards software and business method patents during the 1990s.[77] The result has been an unprecedented explosion of litigation related to software and business method patents.[78] The Federal Circuit's de facto legalization of software and business method patents was a mistake and should be reversed.

---

76. For example, the relationship between energy and mass ($e=mc^2$) has applications for the study of radioactivity, space travel, nuclear energy, and the composition of the universe. *See* Peter Tyson, *The Legacy of E=MC²*, NOVA (Oct. 11, 2005), http://www.pbs.org/wgbh/nova/physics/legacy-of-e-equals-mc2.html. A patent application purporting to cover $e=mc^2$ could be described in terms of any of these applications.

77. Before the 1990s, courts generally held that software was not patentable. In *Gottschalk v. Benson*, 409 U.S. 63 (1972), the Supreme Court invalidated a process patent on software for converting signals from binary-coded decimal into binary, emphasizing that "[p]henomena of nature, . . . mental processes, and abstract intellectual concepts [we]re not patentable." *Id.* at 67. The Court found that granting the patent "in practical effect would [grant] a patent on the [conversion] algorithm itself," *id.* at 72, and indicated that patents on software programs were beyond the scope of the patent statute, absent legislative change. *See id.* at 72–73.

Following *Benson*, patent drafters attempted to redraft abstract process claims into claims for making a new machine, in the hopes of concealing any resemblance of their claims to the process claims at issue in *Benson*. *See* ROBERT PATRICK MERGES & JOHN FITZGERALD DUFFY, PATENT LAW AND POLICY 151–53 (4th ed. 2007). This tactic succeeded when the Federal Circuit upheld machine claims for software in *In re Iwahashi*, 888 F.2d 1370, 1374 (Fed. Cir. 1989), and *In re Alappat*, 33 F.3d 1526 (Fed. Cir. 1994) (*en banc*). The Federal Circuit in *Alappat* acknowledged that "many, or arguably even all, of the means elements recited in [the claim] represent circuitry elements that perform mathematical calculations." 33 F.3d at 1544. Nonetheless, the majority concluded, "This [claim] is not a disembodied mathematical concept . . . but rather a specific machine to produce a useful, concrete, and tangible result." *Id.*

In 1998, the Federal Circuit decided *State Street Bank & Trust Co. v. Signature Fin. Grp., Inc.*, 149 F.3d 1368, 1375 (Fed. Cir. 1998) (eliminating the common-law ban on business method patents).

78. In 2008, software patents were more than twice as likely to be litigated as other patents, and business method patents were nearly seven times more likely to be litigated than other patents. BESSEN & MEURER, *supra* note 30, at 22, 153 fig.7.2. In the late 1990s, software patents accounted for 38% of the cost of patent litigation for public firms. *Id.* at 22.

An enumeration of other technology classes with high discovery costs is beyond the scope of this paper, but would be a worthwhile subject for future study.

One alternative to exclusions based on subject matter would be to vary the application fee for patents based on the estimated discovery costs of each technology class. The patent office would take into account the indexability of the technology classes and the number of firms practicing in it. A party applying for a chemical patent would thus have to pay a very low fee, whereas the fee for patents in software-related technology classes would be drastically increased.[79] This would decrease n_p over time by discouraging applications for patents in areas with high discovery costs.[80]

B. *Independent Invention*

Another powerful reform would be to create an independent invention defense to patent infringement.[81] Ninety to ninety-eight percent of modern patent lawsuits are filed against independent

79. In 2001, Mark A. Lemley estimated that the cost of prosecuting a patent was between $10,000 and $30,000. *See* Mark A. Lemley, *Rational Ignorance at the Patent Office*, 95 Nw. U. L. Rev. 1495, 1498 (2001). A change in patent application fees would have to be significant in relation to the already high costs of hiring attorneys to draft and prosecute the patent.

80. Peter S. Menell and Michael J. Meurer similarly suggest that the cost of evaluating patent applications "should be borne by the applicants and should be tailored to the costs of examining particular applications (or at least classes of applications)." Peter S. Menell & Michael J. Meurer, *Notice Failure and Notice Externalities* 35 (Boston Univ. Sch. of Law, Law and Econ. Research Paper No. 11-58, Boston Univ. Sch. of Law, Pub. Law Research Paper No. 11-58, Stanford Law and Econ. Olin Working Paper No. 418, UC Berkeley Pub. Law Research Paper No. 1973171, Feb. 16, 2012), *available at* http://ssrn.com/abstract=1973171.

81. Other authors have written about the possibility of creating an independent invention defense to patent infringement. *See generally, e.g.*, John S. Leibovitz, *Inventing a Nonexclusive Patent System*, 111 Yale L.J. 2251 (2002) (suggesting that it would be economically efficient for independent inventors to receive independent patents on the same invention); Mark A. Lemley, *Should Patent Infringement Require Proof of Copying?*, 105 Mich. L. Rev. 1525 (2007); Oskar Liivak, *Rethinking the Concept of Exclusion in Patent Law*, 98 Geo. L.J. 1643 (2010); Stephen M. Maurer & Suzanne Scotchmer, *The Independent Invention Defence in Intellectual Property*, 69 Economica 535 (2002), *available at* http://www.dklevine.com/archive/scotchmer-independent-invention.pdf (arguing that independent creation defense would preserve incentives to invent, but also permit more efficient use of inventions); Elisabetta Ottoz & Franco Cugno, *The Independent Invention Defence in a Cournot Duopoly Model*, Econ. Bull., June 20, 2004, at 1–7, *available at* http://www.economicsbulletin.com/2004/volume12/EB-04L10005A.pdf; Samson Vermont, *Independent Invention as a Defense to Patent Infringement*, 105 Mich. L. Rev. 475, 480 (2007).

inventors, not copiers.[82] Independent inventors develop and commercialize inventions without copying from existing, patented inventions. Patent owners who have patented the same invention and who identify an independent inventor can exact licensing fees or even stop the invention's use by the independently-inventing party entirely.[83] This system hurts independent inventors and the public by forcing independent inventors to negotiate licensing fees to keep their existing products on the market—or by removing otherwise-successful products from the market.

There is an extensive literature on proposed independent invention defenses,[84] and a full consideration of the arguments for and against such a defense is beyond the scope of this paper. Nonetheless, we note that in principle, an independent invention defense could reduce discovery costs to zero. With an independent invention defense on the books, patent holders would still have the security of knowing that a competitor could not *copy* their work, but independent parties who happened to create something that infringed a patent would not be liable to the patent holders. Inventors would also have no obligation to search for patents they potentially were infringing because so long as they were not copying another's work, their inventions would be safe from patent lawsuits.

C. *Limiting Injunctions and Multiplied Damages for Patent Infringement*

A final reform would be to limit patent remedies for infringement of non-indexable patents to actual damages, rather than per-

---

82. Lemley, *supra* note 74, at 713 (citing Christopher A. Cotropia & Mark A. Lemley, *Copying in Patent Law*, 87 N.C. L. REV. 1421, 1424 (2009)). Notably, the prevalence of independent invention acts as evidence that the patent system is frequently unnecessary to spur invention. Independent inventors often develop and commercialize the very same inventions others have patented. The fact that a later party developed and commercialized a patented invention independently indicates that it was not necessary to award the original patentee a patent in order for society to benefit from the invention. *See* Lemley, *supra* note 81, at 1527 (citing Vermont, *supra* note 81).

83. *See* 35 U.S.C. § 283 (2006); *see also* eBay Inc. v. MercExchange, L.L.C., 547 U.S. 388 (2006) (describing test for injunctive relief under the patent act).

84. *See* Leibovitz, *supra* note 81; Lemley, *supra* note 81; Liivak, *supra* note 81; Long, *supra* note 7, at 525–33; Maurer & Scotchmer, *supra* note 81; Roger Milgrim, *An Independent Invention Defense to Patent Infringement: The Academy Talking to Itself: Should Anyone Listen?*, 90 J. PAT. & TRADEMARK OFF. SOC'Y 295 (2008); Ottoz & Cugno, *supra* note 81; Vermont, *supra* note 81.

mitting injunctive relief and multiplied damage awards.[85] In other words, remedies for patent infringement could be limited to those instantiating "liability rules" rather than "property rules."

"Property rules" are designed to prevent parties from using another's asset without permission.[86] In the case of patent law, its property rule system "include[s] injunctions and supercompensatory damages that would make [patent infringement] less attractive than bargaining to a consensual price with the [patent] owner."[87] In contrast, "liability rules" such as lost profits or a royalty, theoretically allow parties to infringe on another's patent "as long as . . . officially determined damages are paid. The level of the damages is set to compensate the owner," rather than punish the infringer.[88]

Generally speaking, property-rule remedies are considered beneficial when transaction costs between property holders and those who want to acquire property are low.[89] As Stewart Sterk explains:

> Because property rules require all potential users of a resource to buy rights from th[e resource] owner, property rules enable the owner to accumulate information about potential bidders and the values those bidders attach to those rights. As a result, property rules enable resource owners to channel those resources to the bidders who value them most—promoting efficient use of resources.[90]

---

85. Currently, patent infringers may be enjoined from future infringement and made to pay damages in the form of a reasonable royalty or lost profits. *See* 35 U.S.C. § 283 ("[C]ourts. . .may grant injunctions in accordance with the principles of equity to prevent the violation of any right secured by patent, on such terms as the court deems reasonable."); 35 U.S.C. § 284 (2006) (Courts may award "damages adequate to compensate for the infringement, but in no event less than a reasonable royalty for the use made of the invention by the infringer, together with interest and costs as fixed by the court. . . . [T]he court may increase the damages up to three times the amount found or assessed.").

86. Guido Calabresi & A. Douglas Melamed, *Property Rules, Liability Rules, and Inalienability: One View of the Cathedral*, 85 HARV. L. REV. 1089, 1092 (1972); Henry Smith, *Property and Property Rules*, 79 N.Y.U. L. REV. 1719, 1720 (2004).

87. Smith, *supra* note 86, at 1720.

88. *Id.*

89. Stewart E. Sterk, *Property Rules, Liability Rules, and Uncertainty About Property Rights*, 106 MICH. L. REV. 1285, 1290 (2008) (citing Guido Calabresi & A. Douglas Melamed, *Property Rules, Liability Rules, and Inalienability: One View of the Cathedral*, 85 HARV. L. REV. 1089, 1106–10, 1118 (1972)) (explaining that Guido Calabresi and A. Douglas Melamed observed that "property rules are efficient in cases of low transaction costs, while liability rules are preferable in cases of high transaction costs").

90. *Id.* at 1295 (footnote omitted).

But property rules fail to promote the efficient use of resources when transaction costs are high. The combination of very high transaction costs (e.g., the costs of locating a patent holder to negotiate a license with) and punitive legal penalties (e.g., an injunction or multiplied damages for infringement) can prevent beneficial uses of property and waste resources by making property use very costly.[91]

Permitting property-rule remedies in patent law is harmful because the discovery costs of locating relevant patents render transaction costs too high for many non-indexable patents.[92] It is not merely costly for potential infringers of non-indexable patents to locate the patents they might infringe—it is completely impracticable. As a result, firms are faced with a *disincentive* to develop new products because of the liability that could result if those products infringed others' patents. Firms cannot determine ex ante what is infringing, but an injunction ex post could be crippling.

> Once a product or process has been designed in a way that incorporates a patented invention, redesigning the product might require shutdown for retooling. In addition, especially when the patented invention is a small component in the design of a complex product or process, a redesign around the patented invention may take substantial effort . . . .[93]

In this case, a patent owner may then exact enormous licensing fees from an accidental infringer that the infringer would not have agreed to if the infringed patent had been identified in the product-development stage. Eliminating property-rule remedies—specifically eliminating injunctions and multiplied damages—would lessen the disincentives to producers created by the high discovery costs of the patent system.[94] This proposal does not constitute a

91. *See id.* at 1290.

92. *See id.* at 1296 ("Only if potential resource users know that use of the resource would intrude on someone else's property right, and can readily identify the owner of that right, will they approach the owner to act as an information clearing-house."); *id.* at 1304 ("[C]ompared to a liability-rule regime, a property-rule regime creates excessive incentives to search [to determine the scope of one's legal rights] even when the search costs are high, [and] the probability of encroachment [on another's right] is relatively low . . . ."); *see also id.* at 1311 ("Because the consequences of using without search are so draconian [in a property-rule regime], the user will often be willing to undertake an expensive search even when the probability of liability is very low.").

93. *Id.* at 1333.

94. For a more detailed discussion in favor of awarding non-punitive royalties, see Brian J. Love, *The Misuse of Reasonable Royalty Damages as a Patent Infringement Deterrent*, 74 Mo. L. Rev. 909 (2009).

complete solution: litigation costs are still very high, and potential infringers and patent holders will still dispute the existence of liability and the size of damages. But it would be a step in the right direction.

A more modest proposal would combine this proposal and the previous one: limit injunctions and multiplied damages to cases where the plaintiff can demonstrate that actual copying took place. Under this approach, independent inventors would still be subject to liability for infringement, but their products could not be enjoined and their damages would be limited to a reasonable royalty rate determined by a judge. In contrast, a party caught copying another's invention would be subject to harsher remedies, including injunctions and heightened damages.

CONCLUSION

The patent system is supposed to promote the progress of science and the useful arts, but in some industries it seems to be doing just the opposite. The sheer number of patents and firms, and the lack of an effective organizational scheme for patents, can mean that patent clearance is practically impossible. In software, for example, patent clearance would require the services of many more patent attorneys than exist in the United States. In short, the patent system doesn't scale.

It's a fundamental problem that inventions in certain industries are not indexable, and incremental changes to the patent rules, such as beefing up the novelty and obviousness requirements, are not going to fix the problem. Only dramatic reforms—such as excluding industries with high discovery costs from patent protection, establishing an independent invention defense, or eliminating injunctions—can return the patent system to its proper role of promoting innovation.