No. 13-298

IN THE

# Supreme Court of the United States

ALICE CORPORATION PTY. LTD.,

*Petitioner,*

*v.*

CLS BANK INTERNATIONAL, *et al.*,

*Respondents.*

ON WRIT OF CERTIORARI TO THE UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT

## BRIEF OF *AMICUS CURIAE* ELECTRONIC FRONTIER FOUNDATION IN SUPPORT OF RESPONDENTS


PAMELA SAMUELSON
*Richard M. Sherman Distinguished Professor of Law*
UNIVERSITY OF CALIFORNIA, BERKELEY, SCHOOL OF LAW
434 Boalt Hall
(North Addition)
Berkeley, California 94720
(510) 642-6775

JULIE P. SAMUELS
*Counsel of Record*
MICHAEL BARCLAY
DANIEL K. NAZER
ELECTRONIC FRONTIER FOUNDATION
815 Eddy Street
San Francisco, CA 94109
(415) 436-9333
julie@eff.org

*Attorneys for Amicus Curiae Electronic Frontier Foundation*


February 27, 2014


251687

COUNSEL PRESS

(800) 274-3321 • (800) 359-6859

## TABLE OF CONTENTS

*Page*

TABLE OF CONTENTS. . . . . i

TABLE OF CITED AUTHORITIES . . . . . iv

INTEREST OF AMICUS CURIAE . . . . . 1

INTRODUCTION AND SUMMARY OF ARGUMENT . . . . . 1

ARGUMENT . . . . . 3

I. Limits on the Patentability of Software Innovations Will More Likely Help than Harm the U.S. Software Industry . . . . . 3

A. The U.S. Software Industry Experienced Phenomenal Growth In Its First Four Decades Before the Advent of Software Patents . . . . . 5

1. The Software Industry Was Highly Innovative Well Before 1994 . . . . . 5

2. Software Innovations—and the Software Industry—Continued Apace Without Patent Protection . . . . . 7

*Table of Contents*

*Page*

B. The Modern U.S. Software Industry Largely Does Not Rely on Protection from Software Patents to Grow its Core Business .....11

1. Empirical Evidence Shows that Software Patents Do Not Play an Important Role for Most Software Firms .....11

2. Other Legal and Economic Schemes Adequately Protect Computer Software .....12

3. The Historical Legal Background Illustrates How Software Patents Have Not Increased Growth or Innovation in the Software Industry.....15

a. Legal Background Before 1994.....15

b. 1994-1998: Alappat and State Street .....19

C. If Anything, Evidence Shows that the U.S. Software Industry is Harmed by the Exponential Growth of Vague Software Patents.....22

*Table of Contents*

*Page*

1. Software Patent Litigation Has Dramatically Increased. . . . . .22

2. The High Costs of Software Patent Litigation Fall Largely on Small Innovative Companies . . . . . .24

3. Unclear Software Patents Make the Costs of Litigation Even Worse . . . . . .28

II. A Robust Section 101 Will Help Avoid Widespread Preemption of Abstract Ideas that Inherently Belong in the Public Domain . . . . . .30

A. Section 101's Concern with Preemption Is Intended to Serve the Public Interest . . . . . .30

B. At a Minimum, Performance of an Otherwise Abstract Idea "on a Computer" or "on the Internet" Should Not Make that Idea Non-Abstract . . . . . .32

III. Section 101, When Properly Applied, Can Be a Powerful Tool to Curb Litigation Abuse. . . . .37

CONCLUSION . . . . . .39

## TABLE OF CITED AUTHORITIES

*Page*

**FEDERAL CASES**

*Ass'n for Molecular Pathology v. Myriad Genetics, Inc.*, 133 S. Ct. 2107 (2013) . . . . . .2

*Bancorp Servs., L.L.C. v. Sun Life Assurance Co.*, 687 F.3d 1266 (Fed. Cir. 2012) . . . . . .38

*Bilski v. Kappos*, 130 S. Ct. 3218 (2010) . . . . . *passim*

*CLS Bank Int'l v. Alice Corp.*, 717 F.3d 1269 (Fed Cir. 2013) (*en banc*) . . . . . .33

*Computer Associates Int'l v. Altai, Inc.*, 982 F.2d 693 (2d Cir. 1992) . . . . . .12

*Cyberfone Systems, LLC v. Cellco P'ship, et al.*, No. 11-827-SLR through 11-835-SLR (D. Del. Aug. 16, 2012) . . . . . .38

*Dealertrack, Inc. v. Huber*, 674 F.3d 1315 (Fed. Cir. 2012) . . . . . 32, 33, 34

*Diamond v. Diehr*, 450 U.S. 175 (1981) . . . . . .18

*eBay Inc. v. MercExchange, L.L.C.*, 547 U.S. 388 (2006) . . . . . .1

*Cited Authorities*

*Page*

*Glory Licensing, L.L.C. v. Toys "R" Us, Inc.*,
No. 09-4252 FSH, 2011 WL 1870591
(D. N.J. May 16, 2011) . . . . . .38

*Gottschalk v. Benson*,
409 U.S. 63 (1972). . . . . *passim*

*In re Alappat*,
33 F.3d 1526 (Fed. Cir. 1994) . . . . . .6, 19, 20, 34

*Kewanee Oil Co. v. Bicron Corp.*,
416 U.S. 470 (1974) . . . . . .13

*KSR Int'l Co. v. Teleflex Inc.*,
550 U.S. 398 (2007) . . . . . .1

*Mayo Collaborative Services v.*
*Prometheus Labs., Inc.*,
132 S. Ct. 1289 (2012). . . . . 2, 32, 34

*Medtronic, Inc. v. Mirowski Family Ventures, Inc.*,
134 S. Ct. 843 (2014). . . . . .31

*Microsoft Corp. v.*
*Harmony Computers & Electronics*,
846 F. Supp. 208 (E.D.N.Y. 1994) . . . . . .13

*Microsoft Corp. v. i4i Ltd. P'ship*,
131 S. Ct. 2238 (2011). . . . . .1

*MPHJ Tech. Invs., LLC v. FTC*,
No. 6:14-cv-00011 WSS (W.D. Tex. 2014) . . . . . .27

*Cited Authorities*

*Page*

*MySpace, Inc. v. GraphOn Corp.*,
672 F.3d 1250 (2012). . . . . .36

*Nautilus, Inc. v. Biosig Instruments, Inc.*,
715 F.3d 891, *cert. granted*, 82 U.S.L.W. 3195
(U.S. Jan. 10, 2014) (No. 13-369). . . . . .29

*OIP Techs., Inc. v. Amazon.com, Inc.*,
No. C-12-1233 EMC, 2012 WL 3985118
(N.D. Cal. Sept. 11, 2012). . . . . .38

*Parker v. Flook*,
437 U.S. 584 (1978). . . . . .18, 20

*Pfaff v. Wells Electronics, Inc.*,
525 U.S. 55 (1998). . . . . .30

*Quanta Computer, Inc. v. LG Electronics Inc.*,
553 U.S. 617 (2008). . . . . .1

*Rivendell Forest Prods. v. Georgia Pacific Corp.*,
28 F.3d 1042 (10th Cir. 1994). . . . . .12

*State Street Bank & Trust v.*
*Signature Financial Group, Inc.*,
149 F.3d 1368 (Fed. Cir. 1998). . . . . .19-20

*Ultramercial, LLC v. Hulu, LLC*,
657 F.3d 1323 (Fed. Cir. 2011). . . . . .35

*Uniloc USA, Inc. v. Rackspace Hosting, Inc.*,
No. 12-CV-375 (E.D. Tex. March 27, 2013). . . . . .38

*Page*

*United States v. Microsoft Corp.*,
No. 94-1564 SS (D.D.C. 1994) . . . . . .14

*United States v. Microsoft Corp.*,
No. 98-1232 CKK (D.D.C. 1998) . . . . . .14

*WildTangent, Inc. v. Ultramercial, LLC.*,
722 F.3d 1335 (Fed. Cir. 2013) *petition for cert. filed*, (U.S. Aug. 23, 2013) . . . . . .35

**CONSTITUTIONAL PROVISIONS**

Article I, § 8, cl. 8. . . . . .30

**STATUTES**

35 U.S.C. § 101 . . . . . *passim*

35 U.S.C. § 102 . . . . . .32

35 U.S.C. § 103 . . . . . .32

35 U.S.C. § 112 . . . . . .32

**OTHER AUTHORITIES**

*25 years of AOL: A timeline*, Washington Post (May 23, 2010) . . . . . .10

Adam Lashinsky, *Remembering Netscape: The Birth Of The Web*, CNN Money (July 25, 2005) . . . . . .10

*Page*

Brian T. Yeh, *An Overview of the "Patent Trolls" Debate,* Cong. Research Serv., R42668 (2012) ..... 24, 26, 37

Business Software Alliance, *Software Industry Facts and Figures* .....11

Chris Barry et al., *2013 Patent Litigation Survey*, PricewaterhouseCoopers, 6 (2013)..... 22, 23, 24

Colleen Chien, *Patent Assertion and Startup Innovation*, New America Foundation (Sept. 2013) ..... 26, 28

Colleen Chien, *Startups and Patent Trolls* (Santa Clara Univ. School of Law, Legal Studies Research Paper Series, Accepted Paper No. 09-12, 2012) .....26, 27

*Computer and Internet Use in the United States*, U.S. Census Bureau (May 2013).....36

Executive Office of the President, *Patent Assertion and U.S. Innovation*, 5 (2013) .....24

Iain M. Cockburn & Megan J. MacGarvie, *Entry and Patenting in the Software Industry* (NBER Working Paper No. 12563, Table 3 (2006)) .....11

IBM Annual Report 25-26 (2010) .....14

*Page*

James Bessen & Michael Meurer, *Patent Failure: How Judges, Bureaucrats, and Lawyers Put Innovators at Risk* 191 (2008). . . . . .23

James Bessen & Michael Meurer, *The Direct Costs from NPE Disputes*, Boston Univ. School of Law, Law and Economics Research Paper No. 12-34 (2012) . . . . . .24, 27, 28

James Bessen & Robert M. Hunt, *An Empirical Look at Software Patents*, 16 J. Econ. & Mgmt Strategy 157 (2007). . . . . .11

James Bessen, *A Generation of Software Patents*, Boston University School of Law Working Paper No. 11-31 (June 21, 2011) . . . . . *passim*

James Bessen, Jennifer Ford and Michael Meurer, *The Private and Social Costs of Patent Trolls*, Boston Univ. School of Law, Working Paper No. 11-45 (2011) . . . . . .25, 29

Joe Mullin, *Patent Trolls Want $1,000—For Using Scanners*, Ars Technica (Jan. 2, 2013). . . . . .27

John Allison, Mark A. Lemley and Joshua Walker, *Patent Quality and Settlement Among Repeat Patent Litigants*, 99 Geo. L. J. 677 (2010). . . . . 25, 28, 37

*Page*

John E. Jankowski, *Business Use of Intellectual Property Protection Documented in NSF Survey*, National Science Foundation InfoBrief (February 2012) .....12

Mark A. Lemley & David McGowan, *Legal Implications of Network Effects*, 86 Cal. L. Rev. 479 (1998) .....13, 14

Mark A. Lemley, *Software Patents and the Return of Functional Claiming*, 2013 Wis. L. Rev. 905 (2013) .....29

Martin Campbell-Kelly, *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry* (I. Bernard Cohen & William Aspray eds., 2003) ..... *passim*

Maureen O'Rourke, *The Story of Diamond v. Diehr: Toward Patenting Software*, in Intellectual Property Stories 194-219 (Jane C. Ginsburg & Rochelle Cooper Dreyfuss eds. 2006) .....18

Michael Cusamano, *The Business of Software: What Every Manager, Programmer, and Entrepreneur Must Know to Thrive in Good Times and Bad* 273-74 (2004) .....14

Michael Katz & Carl Shapiro, *Antitrust in Software Markets*, in Competition, Innovation, and the Microsoft Monopoly 32-34 (Jeffrey Eisenach & Thomas M. Lenard, eds. 1999) .....13

*Page*

National Commission on New Technological Uses of Copyrighted Works, Final Report 16-17 (1979) ..... 17-18

Office of Technology Assessment, Computer Software and Intellectual Property: Background Paper 8 (1990).....17

Pamela Samuelson, *Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions*, 39 Emory L.J. 1025 (1990).....18

Pamela Samuelson, et al. *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 Colum. L. Rev. 2308 (1994).....15

Report of the President's Commission on the Patent System, *"To Promote the Progress of Useful Arts" in an Age of Exploding Technology* 13 (1966) .....16

RFC 1945, Hypertext Transfer Protocol/ HTTP/1.0 (May 1996) .....10

RFC 791, *Internet Protocol*, DARPA Internet Program Protocol Specification (September 1981) .....10

Robert M. Hunt & James Bessen, *The Software Patent Experiment*, Fed. Reserve Bank of Phila. Bus. Rev., Q3 2004 .....11

*Page*

Rochelle C. Dreyfuss & James P. Evans, *From Bilski Back to Benson: Preemption, Inventing Around, and the Case of Genetic Diagnostics*, 63 Stan. L. Rev. 1349 (2011). . . . . .32

Ronald J. Mann, *Do Patents Facilitate Financing in the Software Industry?*, 83 Tex. L. Rev. 961 (2005). . . . . .11

Stephen Breyer, *The Uneasy Case for Copyright: A Study of Copyright for Books, Photocopies, and Computer Programs*, 84 Harv. L. Rev. 281 (1970). . . . . .16

Stuart J.H. Graham, et al., *High Technology Entrepreneurs and the Patent System: Results of the 2008 Berkeley Patent Survey*, 24 Berkeley Tech. L.J. 1255 (2009). . . . . .11, 13, 14

United States Government Accountability Office, *Assessing Factors That Affect Patent Infringement Litigation Could Help Improve Patent Quality*, 22 (2013). . . . . 23, 26

## INTEREST OF AMICUS CURIAE[1]

The Electronic Frontier Foundation ("EFF") is a nonprofit civil liberties organization that has worked for more than 20 years to protect consumer interests, innovation, and free expression in the digital world. EFF and its more than 29,000 dues-paying members have a strong interest in helping the courts and policy-makers in striking the appropriate balance between intellectual property and the public interest. As part of its mission, EFF has often served as amicus in key patent cases, including *Microsoft Corp. v. i4i Ltd. P'ship,* 131 S. Ct. 2238 (2011); *Bilski v. Kappos*, 130 S. Ct. 3218 (2010); *Quanta Computer, Inc. v. LG Electronics Inc.*, 553 U.S. 617 (2008); *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398 (2007); and *eBay Inc. v. MercExchange, L.L.C.*, 547 U.S. 388 (2006).

## INTRODUCTION AND SUMMARY OF ARGUMENT

This Court has repeatedly made clear that 35 U.S.C. § 101 should serve to define what inventions make up patentable subject matter; in so doing, it has corrected the Federal Circuit's erroneously expansive interpretation of that section. Despite the Court's clear guidance in recent cases, the Federal Circuit has failed to implement

---

1. No counsel for a party authored this brief in whole or in part, and no such counsel or party made a monetary contribution intended to fund the preparation or submission of this brief. No person other than the amicus curiae, or its counsel, made a monetary contribution intended to fund its preparation or submission. Letters of the Parties' general consent to the filing of amicus briefs are on file with the Court. Web sites cited in this brief were last visited on February 21, 2014.

a workable standard—or, frankly, any standard at all—as to what computer- and Internet-implemented inventions are patentable. The resulting legal instability has driven up the already-ballooning costs of patent litigation and has discouraged district courts from using §101 to slow that trend.

The Court should use this opportunity to do what it has already done in other technological areas[2]—clearly state that § 101 serves as a meaningful tool to reign in overbroad software patents. These patents must be dealt with: they threaten the public interest by throttling the freedom to create, they discourage young innovators from increased innovation, and, consequently, they tie up important downstream inventions. Software now forms one of the largest industries in the United States, yet no proof exists that the advent of software patents in the mid-1990s had anything to do with its growth. In fact, the data show the opposite: the industry maintained a growth rate before 1994 that it continued to see after the Federal Circuit devised the notion of broad software patents that year. Software patents, if anything, have hurt the industry.

The growth of this industry has brought about many life-changing innovations, some of which are summarized below. One of particular importance is that it has endowed anyone with access to a computer the power to code, and, thus, create. It has lowered the barrier to entry in new and fundamentally important ways, opening doors for individuals from all walks of life to create and disseminate

2. See *Ass'n for Molecular Pathology v. Myriad Genetics, Inc.*, 133 S. Ct. 2107 (2013); *Mayo Collaborative Services v. Prometheus Labs., Inc.*, 132 S. Ct. 1289 (2012).

software-type inventions. These innovators should have the freedom to write software code unobstructed by patents on abstract methods and systems, such as those claimed by Alice Corp. here.

This Court recognized the inherent challenges that come with this shift in *Bilski v. Kappos* nearly four years ago: "With ever more people trying to innovate and thus seeking patent protection for their inventions, the patent law faces a great challenge in striking the balance between protecting inventors and not granting monopolies over procedures that others would discover by independent, creative applications of general principles." 130 S. Ct. 3218, 3228 (2010). At that time, the Court declined to definitively resolve that challenge. Respectfully, the time has come. The current failure of § 101 to reign in overbroad software patents hurts the software industry, stifles innovation, and harms the public interest.

## ARGUMENT

### I. Limits on the Patentability of Software Innovations Will More Likely Help than Harm the U.S. Software Industry

Alice and its amici suggest that the software industry will suffer enormous harm if the Court does not broadly uphold the patent eligibility of computer software. *See* Alice's Opening Brief at 55 (stating that a ruling adverse to Alice "will decimate entire sectors of the economy and stifle innovation"). The Federal Circuit's former Chief Judge advises that software patents are "essential" and that weakening software patents will supposedly "cripple, if not destroy, computer-related industries . . . which are

vital to the future of the country." Brief of Paul R. Michel at 1, 9. We are told that the "consequences of a failure to preserve the patent-eligibility of software-implemented solutions are numerous and harmful to our country." Brief of IEEE-USA at 26-28. The Court is warned "not to erect barriers to the patentability" of computer-implemented technologies (Brief of Intellectual Property Owners Association at 2), since doing so would cause "vital and important inventions to be lost" (Brief of Conejo Valley Bar Ass'n at 4).

Alice and its amici are wrong. The Court need not be concerned that a decision expanding the effect of § 101 might adversely affect the computer software industry. Software patents do not promote innovation in the computer software industry—in fact, the recent flood of such patents impedes innovation.

Significantly, Alice and its amici do not analyze or discuss whether there is any causal connection between widespread software patenting, on the one hand, and innovation and growth of the software industry, on the other. If there was any causal connection, one would expect minimal software growth before software patents became available, followed by a booming industry when patents became freely available. But no such thing happened: software patents simply don't cause more software innovation, since this field doesn't rely on patents to spur innovation. A decision that broadly and clearly enforces § 101, and that has the effect of limiting the number of software patents, will thus be greatly beneficial to the industry, its innovators, and its users.

### A. The U.S. Software Industry Experienced Phenomenal Growth In Its First Four Decades Before the Advent of Software Patents

#### 1. The Software Industry Was Highly Innovative Well Before 1994

Assuming that there is some actual causal relationship between the grant of large numbers of software patents, on the one hand, and software innovation, on the other—as Alice and its amici argue—then one would expect that the software market would show a similarly large increase after the 1994 time frame, when the PTO started granting software patents in earnest. *Infra* at Section I.B.3.b. But no such causation exists, because the software industry was highly innovative before that time. *See* Martin Campbell-Kelly, *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry* Fig. 1.2 at 16 (I. Bernard Cohen & William Aspray eds., 2003) ("Campbell-Kelly"), which shows how the software market rapidly increased well before 1994:

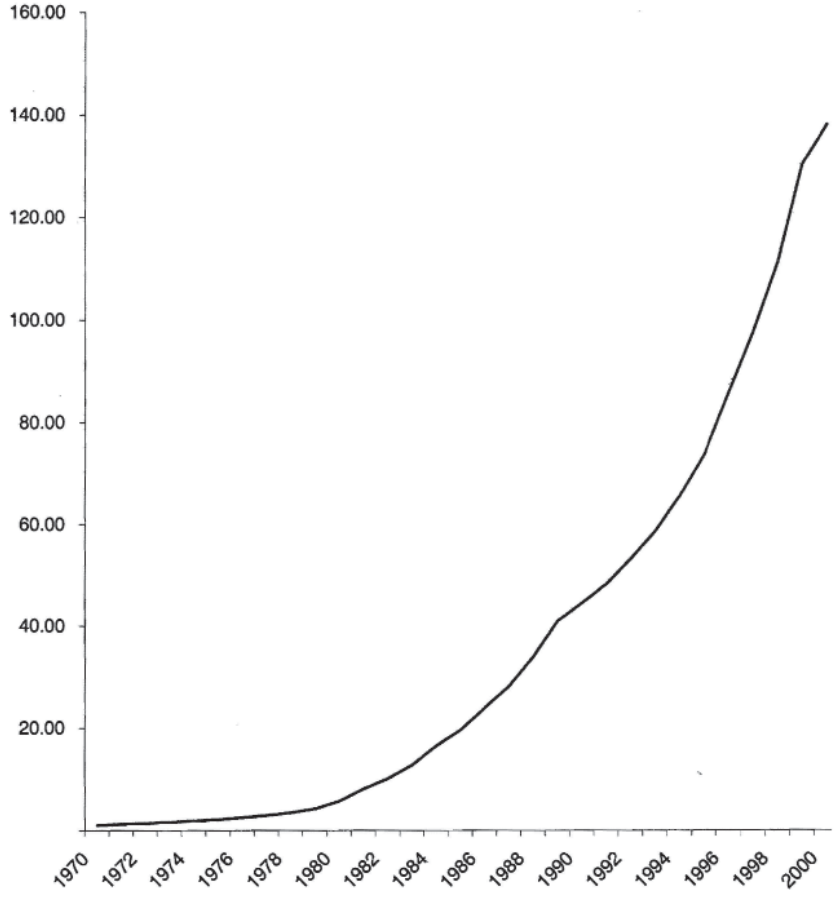



*Figure 1.2*
The total US software market (user expenditures in billions of dollars), 1970–2000. Courtesy of INPUT.

The software market began its rapid increase in the early 1980s— around the time personal computers became widespread—more a decade *before* the Federal Circuit concocted widespread software patents in 1994. *See In re Alappat*, 33 F.3d 1526 (Fed. Cir. 1994); *infra* at Section

I.B.3.(b). Obviously, no patents were needed for software to become a $60 billion/year industry by 1994. Moreover, if software patenting caused software innovation, one would have expected a sharp increase in the market after 1994—an increase that tracked the corresponding increase in software patents[3]—but that, too, didn't happen. The industry merely continued its steady climb that began well before any significant software patenting was available.

### 2. Software Innovations—and the Software Industry—Continued Apace Without Patent Protection

From the 1950s through the mid-1990s, the U.S. software industry grew from a nascent data processing and services business to a broad-based industry with annual revenues in the many tens of billions. Campbell-Kelly, *supra*, at 16-19, 29. The industry did so without the benefit of software patents, as shown by the following examples.

- *Mainframe computers*. The computer industry started in the 1950s with mainframe computers (largely made by IBM) and accompanying software. This included fundamental programming languages such as FORTRAN and COBOL, and key applications such as the IBM-American Airlines SABRE airline reservation system. *Id.* at 29-54.

3. As discussed more fully, *infra* at Section I.B.3.(b)., the annual grants of software patents began increasing in 1994; that increase began a more sustained spike beginning in 1997.

- *Expansion in the 1960s and 1970s*. Much of the software produced in the 1950s and 1960s came bundled with computer hardware, although a services industry emerged (e.g., to provide system integration and customization). Also, ambitious application programs in the banking, airline, aerospace, and military contracting industries became economically significant. *Id*. at 57-87. The 1970s saw a substantial expansion of the software industry after the U.S. Antitrust Division persuaded IBM Corp. to "unbundle" software it had provided to its popular mainframe computers. *Id*. at 114-18. Product lines expanded to include communications, engineering, financial, retail, and transportation industries. *Id*. at 136-41. The UNIX operating system began in the early 1970s. *Id*. at 143-45.

- *Database and office automation software*. IBM and others developed database systems starting around 1964, while Wang Laboratories and others introduced office automation software in the early 1970s. Database developer Oracle Corp. grew from $13 million in revenue in 1984 to $2 billion in revenue by 1994. *Id*. at 145-49, 159-61, 185-91.

- *Personal Computers*. Even more significant in the rise of the U.S. software industry in the 1980s was the introduction of the IBM PC and the development of "killer apps" such as Lotus 1-2-3 which drove demand for PCs. The era developed innovative products such as the MS-DOS and CP/M operating systems, the BASIC programming language, and applications such

as the Lotus and other spreadsheets, word processors, and database programs. *Id.* at 201-28, 252-57. While the industry as a whole grew rapidly between 1980 and 1995, *id.* at 16 (Fig. 1.2), some firms experienced phenomenal revenue growth. For example:

- Computer Associates' annual revenues were $18 million in 1980, then grew to over $2 billion by 1994 and $3.5 billion by 1996. *Id.* at 178-85.

- IBM's software revenues grew from $800 million in 1980 to almost $13 billion by 1995. *Id.* at 175.

- Microsoft grew from $8 million in revenue in 1980 to more than $4 billion in 1994. *Id.* at 233. (Indeed, by 1994 Microsoft was able to attain monopoly power in the market for operating systems. *See* section I.B.2. below.)

- *Graphical User Interfaces.* A major, pre-1994 innovative development (still in universal use today) was the graphical user interface for computers. This work started in the 1970s at Xerox's Palo Alto Research Center, followed by Apple Computer's Lisa and Macintosh products in 1983-84, and then Microsoft Windows in 1985. *Id.* at 246-251.

- *Video games and recreational software*. By 1994, a multi-billion-dollar industry had been developed by companies such as Atari, Activision, Tandy, Broderbund, Nintendo, Sega and others, for video games such as Pong, Space Invaders, Mario Bros., Tetris, the unforgettable Sonic the Hedgehog, and others. *Id*. at 269-288.

- *Computer networks and the World Wide Web*. The Internet Protocol, used by the modern Internet, was invented around 1981.[4] Tim Berners-Lee invented the HTTP protocol used for the World Wide Web in 1990;[5] the Web became popularized after Netscape released the first commercial browser in 1994.[6] AOL and other companies popularized the use of home computers online starting in 1985.[7]

---

4. RFC 791, *Internet Protocol*, DARPA Internet Program Protocol Specification (September 1981), available at: http://www.ietf.org/rfc/rfc791.txt.

5. RFC 1945, Hypertext Transfer Protocol/HTTP/1.0 (May 1996), available at: http://www.isi.edu/in-notes/rfc1945.txt.

6. Adam Lashinsky, *Remembering Netscape: The Birth Of The Web*, CNN Money (July 25, 2005), available at: http://money.cnn.com/magazines/fortune/fortune_archive/2005/07/25/8266639/.

7. *25 years of AOL: A timeline*, Washington Post (May 23, 2010), available at: http://www.washingtonpost.com/wp-dyn/content/article/2010/05/23/AR2010052303551.html.

### B. The Modern U.S. Software Industry Largely Does Not Rely on Protection from Software Patents to Grow its Core Business

#### 1. Empirical Evidence Shows that Software Patents Do Not Play an Important Role for Most Software Firms

According to the Business Software Alliance, the U.S. software industry today "adds more than $260 billion in value to the U.S. economy." Business Software Alliance, *Software Industry Facts and Figures*.[8] The software industry has thus grown to be an even more significant contributor to the gross national product and to U.S. export markets than in the 1990s. It is, therefore, very telling that several empirical studies report that only a small minority of software development firms seek patent protection for their innovations. Robert M. Hunt & James Bessen, *The Software Patent Experiment*, Fed. Reserve Bank of Phila. Bus. Rev., Q3 2004 at 24;[9] James Bessen & Robert M. Hunt, *An Empirical Look at Software Patents*, 16 J. Econ. & Mgmt Strategy 157, 171 (2007); Ronald J. Mann, *Do Patents Facilitate Financing in the Software Industry?*, 83 Tex. L. Rev. 961, 964 (2005); Iain M. Cockburn & Megan J. MacGarvie, *Entry and Patenting in the Software Industry* (NBER Working Paper No. 12563, Table 3 (2006)); Stuart J.H. Graham, et al., *High Technology Entrepreneurs and the Patent System: Results of the 2008 Berkeley Patent Survey*, 24 Berkeley Tech. L.J. 1255, 1277 (2009).

---

8. Available at: http://www.bsa.org/country/public%20policy/~/media/files/policy/security/general/sw_factsfigures.ashx.

9. Available at: http://www.phil.frb.org/files/br/brq304rh.pdf.

A recent study by the National Science Foundation is informative. John E. Jankowski, *Business Use of Intellectual Property Protection Documented in NSF Survey*, National Science Foundation InfoBrief (February 2012).[10] In the information sector (which includes software, Internet, and Data processing) only 10 percent of companies found utility patents either "very" or even "somewhat" important. *Id.* at 3. Those companies rely instead on the copyright, trademark and trade secret protection discussed in the next section. *Id.* During the period 1994-2004, only 20 percent of software startup companies even applied for a patent. James Bessen, *A Generation of Software Patents*, Boston University School of Law Working Paper No. 11-31 (June 21, 2011) ("Bessen Generation") at 6.[11]

### 2. Other Legal and Economic Schemes Adequately Protect Computer Software

An important reason why most U.S. software developers still rarely patent their innovations is because copyright, trade secrecy, trademarks, and licensing are all available and widely used to protect programs and program innovations. *See, e.g.*, *Computer Associates Int'l v. Altai, Inc.*, 982 F.2d 693, 702 (2d Cir. 1992) (recognizing that copyright protects not only program code but also some of the structure, sequence, and organization of programs); *Rivendell Forest Prods. v. Georgia Pacific Corp.*, 28 F.3d 1042 (10th Cir. 1994) (applying trade secrecy

10. Available at: http://www.nsf.gov/statistics/infbrief/nsf12307/.

11. Available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1868979.

law to computer programs); *Microsoft Corp. v. Harmony Computers & Electronics*, 846 F. Supp. 208 (E.D.N.Y. 1994) (recognizing trademark, as well as copyright, and breach of licensing claims in software). *See also Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 482-83 (1974) ("trade secret law protects items which would not be proper subjects for consideration for patent protection under 35 U.S.C. § 101.").

A recent empirical study of executives from software startups shows that copyrights, trademarks, and secrecy were more important than patents in providing software developers with competitive advantage in the marketplace. *See* Graham, *supra* at 1290. Patents were, in fact, perceived to be the least important means of attaining competitive advantage for software innovations. *Id.* This survey showed that software entrepreneurs regarded patents as providing relatively weak incentives to invest in software development. *Id.* at 1285.

Far more important than intellectual property rights to these software executives to gaining a competitive edge in the marketplace were first mover advantages and complementary assets. *Id.* at 1290. First movers in the software industry are often able to benefit from the existence of network effects to drive growth. *See, e.g.*, Michael Katz & Carl Shapiro, *Antitrust in Software Markets*, in Competition, Innovation, and the Microsoft Monopoly 32-34 (Jeffrey Eisenach & Thomas M. Lenard, eds. 1999); Mark A. Lemley & David McGowan, *Legal Implications of Network Effects*, 86 Cal. L. Rev. 479 (1998). Indeed, network effects were so powerful a driver of competitive advantage for Microsoft that the U.S. Department of Justice in the mid- and late 1990s attacked

the firm as a monopoly for misusing its dominant position in the marketplace for operating systems. *Id.* at 500-06. *See United States v. Microsoft Corp.*, No. 94-1564 SS (D.D.C. 1994);[12] *United States v. Microsoft Corp.*, No. 98-1232 CKK (D.D.C. 1998).[13]

Among the complementary assets that can provide software developers with considerable revenues are integration, customization, and other types of services. For example, 57 percent of IBM's $99.87 billion in revenues in 2010 came from providing services to its customers. IBM Annual Report 25-26 (2010). This was more than double its revenues from sales of software and more than three times its revenues attributable to sale of computer systems. *Id. See generally* Michael Cusamano, *The Business of Software: What Every Manager, Programmer, and Entrepreneur Must Know to Thrive in Good Times and Bad* 273-74 (2004) (emphasizing the importance of services as a supplement to software sales).

There are several other reasons why patents play a modest role in the software industry. For one thing, patents are very costly to obtain and to enforce. Graham, *supra*, at 1313. Software entrepreneurs do not want to disclose their innovations if they can be kept secret. *Id.* They worry also that it will be easy to invent around any patent they might get. *Id.* Many also regard their software innovations to be unpatentable. *Id.*

---

12. Information available at: http://www.justice.gov/atr/cases/f0000/0045.htm.

13. Information available at: http://www.justice.gov/atr/cases/ms_index.htm.

There is, moreover, a significant mismatch between the pace of evolution in the software industry and the processes of the patent system. Concerns about lack of expertise in the Patent Office and lack of awareness about the state of the art in software have continued to be pervasive and contribute to the problem of "bad" patents. *See, e.g.*, Pamela Samuelson, et al. *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 Colum. L. Rev. 2308, 2361-64 (1994).

### 3. The Historical Legal Background Illustrates How Software Patents Have Not Increased Growth or Innovation in the Software Industry

#### a. Legal Background Before 1994

To understand why software patents have played little to no positive role in the sustained and continued growth of the software industry, it is helpful to understand the relevant history in light of the legal background. Initially, there were both doctrinal and practical reasons for doubting the soundness of patenting software innovations, as a Presidential Commission reported in 1966:

> Uncertainty now exists as to whether the statute permits a valid patent to be granted on programs. Direct attempts to patent programs have been rejected on the ground of nonstatutory subject matter. Indirect attempts to obtain patents and avoid the rejection, by drafting claims as a process, or a machine or components thereof programmed in a given manner, rather than as a program itself, have confused the issue further and should not be permitted.

> The Patent Office now cannot examine applications for programs because of a lack of a classification technique and the requisite search files. Even if these were available, reliable searches would not be feasible or economic because of the tremendous volume of prior art being generated. Without this search, the patenting of programs would be tantamount to mere registration and the presumption of validity would be all but nonexistent.
>
> It is noted that the creation of programs has undergone substantial and satisfactory growth in the absence of patent protection and that copyright protection for programs is presently available.

Report of the President's Commission on the Patent System, *"To Promote the Progress of Useful Arts" in an Age of Exploding Technology* 13 (1966).[14]

Six years later, the U.S. Supreme Court concurred in the Commission's judgment when it encountered the first test case on the patentability of software innovations. *Gottschalk v. Benson*, 409 U.S. 63, 72 (1972) (fully quoting the passage above). In line with the Presidential

---

14. Available at: http://digitalcommons.law.scu.edu/cgi/viewcontent.cgi?article=1010&context=historical.

In his tenure article, then Professor (now Justice) Stephen Breyer expressed reservations about the patenting of computer software innovations. *See* Stephen Breyer, *The Uneasy Case for Copyright: A Study of Copyright for Books, Photocopies, and Computer Programs*, 84 Harv. L. Rev. 281, 347-49 (1970).

Commission's recommendations, the Patent Office had rejected Benson's application for a patent on a method of transforming numerical information from one form of representation to another (that is, a way to convert binary coded decimals into pure binary form). *Id.* at 64. One claim called for carrying out this method with the aid of a computer, while a second claim would have covered all ways of carrying out the method. *Id.* at 73-74.

The Court rejected patent protection for this method for several reasons. For one thing, it was influenced by an amicus curiae brief submitted by IBM Corp. which spoke of Benson's method as a "mathematical algorithm," the patenting of which would have "the general effect of extending patent protection to mathematical techniques, scientific principles, and other abstract concepts or rules." Brief Amicus Curiae of International Business Machines at 7, 9. This brief gave several examples of important mathematical discoveries that could only be effectively carried out by computer. *Id.* at 13-18. IBM warned of the "stifling effect" that patents on program methods would have on discourse in mathematical and scientific fields. *Id.* at 19. This resonated with the Court in Benson which spoke of "[p]henomena of nature, though just discovered, mental processes, and abstract intellectual concepts" as unpatentable because "they are the basic tools of scientific and technical work." *Benson*, 409 U.S. at 67.

In the aftermath of the Court's ruling in *Benson*, it was widely believed that patents were unavailable for software innovations. *See, e.g.*, Office of Technology Assessment, Computer Software and Intellectual Property: Background Paper 8 (1990); National Commission on New Technological Uses of Copyrighted

Works, Final Report 16-17 (1979). The Court reinforced this conclusion when it ruled against another program-related invention in *Parker v. Flook*, 437 U.S. 584 (1978). Flook sought a patent for an improved method of analyzing data and updating alarm limits for catalytic conversion plants, the only novel element of which was an equation in the second step of the process. *Id.* at 585-88. The Court agreed with the Patent Office that this method was unpatentable under *Benson*, opining that merely adding some conventional or insignificant post-solution activities to a claim would not suffice to qualify the method for a patent. *Id*. at 590.

Three years later, the Supreme Court once again reviewed a patent claim for a computer-implemented method in *Diamond v. Diehr*, 450 U.S. 175 (1981). This time the Court upheld the patentability of a claim for an improved method of curing synthetic rubber that utilized a computer program as one component of the process. Because the Supreme Court was deeply divided (5-4 split) over the patentable subject matter issue in this case, and because rubber curing is a conventional technological process, it seemed for some time as though the door to patenting of software innovations had opened only slightly.[15]

---

15. For a fuller recounting of the history of software patenting decisions, *see, e.g.*, Maureen O'Rourke, *The Story of Diamond v. Diehr: Toward Patenting Software*, in Intellectual Property Stories 194-219 (Jane C. Ginsburg & Rochelle Cooper Dreyfuss eds. 2006); Pamela Samuelson, *Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions*, 39 Emory L.J. 1025 (1990).

### b. 1994-1998: Alappat and State Street

Two significant Federal Circuit decisions caused a huge rise in software patenting. The first was the en banc ruling, *In re Alappat*, 33 F.3d 1526 (Fed. Cir. 1994), in which the Federal Circuit concocted the notion of broad software patents. *Alappat* upheld a claim for a computerized apparatus for creating a smooth waveform display for oscilloscopes using an anti-aliasing technique. The Patent Office had rejected the claim because it was for a mathematical method for calculating numerical values. *Id.* at 1539-40. A plurality opinion questioned whether the *Benson* exclusion of mathematical methods applied to apparatus claims. *Id.* at 1542. In the plurality's view, Alappat's claim was "not [for] a disembodied mathematical concept which may be characterized as an 'abstract idea,' but rather a specific machine to produce a useful, concrete and tangible result." *Id.* at 1544. The plurality regarded general purpose computers as new machines whenever loaded with programs that made them into special purpose machines. *Id.* at 1545. Notably, in his dissent, Chief Judge Archer warned that allowing for this type of broad § 101 interpretation, a compact disc player would be a different machine each time a different compact disc was played inside. *Id.* at 1553-54 (Archer, C.J. dissenting).

The *Alappat* decision "opened the way for a large-scale increase in the patenting of software." Bessen Generation, *supra*, at 2. *See also* Brief of Dale R. Cook at 2 (*Alappat* "changed everything"); Brief of IEEE-USA at 6 (noting industry "reliance" on *Alappat*).

A second contributor to the current surge in software patents was *State Street Bank & Trust v. Signature*

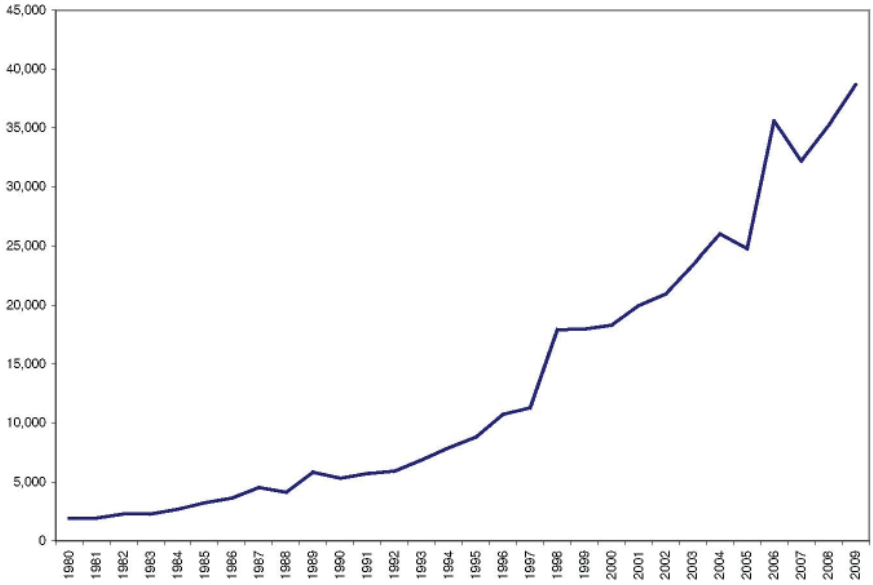
*Financial Group, Inc.*, 149 F.3d 1368 (Fed. Cir. 1998). State Street had obtained a patent on a data processing system for configuring financial services with a hub and spoke model. After it sued Signature Financial for patent infringement, Signature Financial challenged the validity of State Street's patent on subject matter grounds. The Federal Circuit upheld the validity of the patent, invoking the standard that the *Alappat* plurality had endorsed:

> Today we hold that the transformation of data, representing discrete dollar amounts, by a machine through a series of mathematical calculations into a final share price, constitutes a practical application of a mathematical algorithm because it produces a 'useful, concrete and tangible result"—a final share price momentarily fixed for recording and reporting purposes and even accepted and relied upon by regulatory authorities and in subsequent trades.

*Id.* at 1373.

Following the Federal Circuit's decisions in *Alappat* and *State Street*, the "useful, concrete and tangible result" standard for patentable subject matter contributed to a very substantial surge in the patenting of software. It seemed as though *Benson* and *Flook* had, in effect, been overruled. The biggest spike in applications and in issuance of software patents began in 1994. Bessen Generation, *supra*, at 26, Fig. 1:

Figure 1. Annual grants of US software patents.




Note the increase in 1994, and the sharp increase starting in 1997—corresponding to the approximate three-year window of time for the Patent Office to examine and allow patent applications. *See also* Brief of IEEE-USA at 30, Fig. 1 (showing a similar graph as Bessen Generation, *supra*, Fig. 1). However, as noted above, that shift in the patenting of software provided no additional stimulus to the already growing software industry.

### C. If Anything, Evidence Shows that the U.S. Software Industry is Harmed by the Exponential Growth of Vague Software Patents

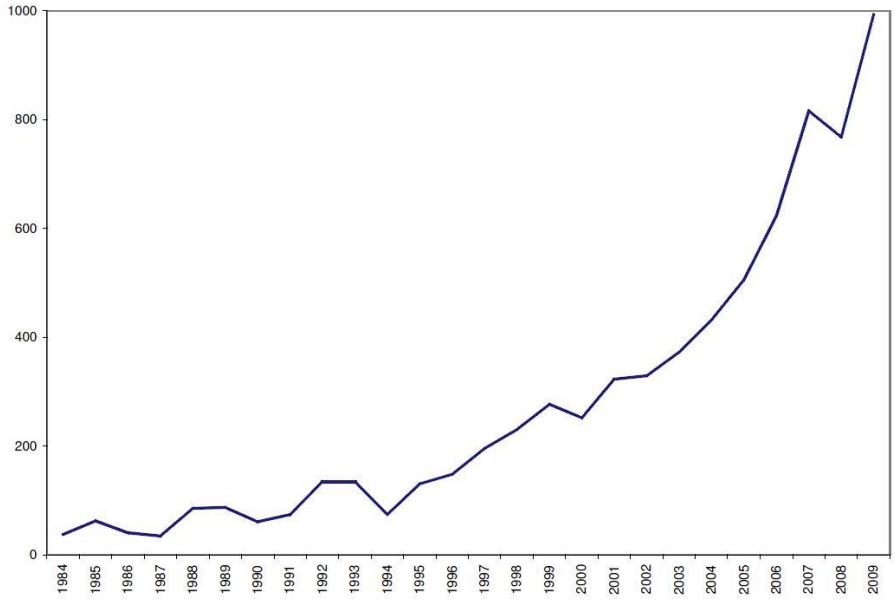
#### 1. Software Patent Litigation Has Dramatically Increased

This increase in software patents has been quite costly, not just in fees to the Patent Office, but in harm to innovative companies who increasingly face those patents in an explosion of software patent litigation. In fact, contrary to what Alice and its amici argue, there is significant empirical evidence, particularly in recent years, that software patents actually harm the industry. To understand, one must look at recent trends in patent litigation. There were 5,189 patent actions filed in 2012, compared to fewer than 3,000 such actions filed in 2009. Chris Barry et al., *2013 Patent Litigation Survey*, PricewaterhouseCoopers, 6 (2013) ("PWC 2013").[16]

Litigation involving software patents has also rapidly increased, at a rate far higher than in other technological areas, Bessen Generation, *supra*, Fig. 3 at 19:

16. Available at: http://www.pwc.com/en_US/us/forensic-services/publications/assets/2013-patent-litigation-study.pdf.

Figure 3. Number of patent lawsuit filings involving software patents




One study has found that between 2007 and 2011, 46 percent of patent lawsuits involved software patents, accounting for 89 percent of the increase in the number of patent defendants during this timeframe. United States Government Accountability Office, *Assessing Factors That Affect Patent Infringement Litigation Could Help Improve Patent Quality*, 22 (2013) ("GAO Report").[17] *See also* PWC 2013, *supra*, at 14 (software and Internet/online services industries "experienced significant increases in identified decisions from 2007 through 2012."); James Bessen & Michael Meurer, *Patent Failure: How Judges, Bureaucrats, and Lawyers Put Innovators at Risk* 191, Table 9.1 (2008) (Software patents are more than twice as

17. Available at: http://www.gao.gov/assets/660/657103.pdf.

likely to be litigated as are other patents; business-method patents are nearly seven times more likely to be litigated); Bessen Generation, *supra*, at 259.

Much of this litigation involves patent cases brought by non-practicing entities (NPEs), also known as patent assertion entities (PAEs), patent monetizers, or colloquially, "patent trolls," which have significantly increased in recent years. PWC 2013, *supra*, at 7; *see also* Executive Office of the President, *Patent Assertion and U.S. Innovation*, 5 (2013) (finding a "dramatic increase" in PAE activity in recent years).[18]

### 2. The High Costs of Software Patent Litigation Fall Largely on Small Innovative Companies

This explosion of litigation has been costly. According to a congressional study, NPEs activity cost defendants and licensees $29 billion in 2011, a 400 percent increase over $7 billion in 2005, and the losses are mostly deadweight, with less than 25 percent flowing to innovation and at least that much going towards legal fees. Brian T. Yeh, *An Overview of the "Patent Trolls" Debate,* Cong. Research Serv., R42668, at Summary and 2 (2012) ("Yeh")[19] (citing James Bessen & Michael Meurer, *The Direct Costs from NPE Disputes*, Boston Univ. School of Law, Law and Economics Research Paper No. 12-34 (2012) ("Bessen

18. Available at: http://www.whitehouse.gov/sites/default/files/docs/patent_report.pdf.

19. Available at: https://www.eff.org/sites/default/files/R42668_0.pdf.

2012") at 1, 17-18.).[20] The research shows that that "NPE lawsuits are associated with half a trillion dollars of lost wealth to defendants from 1990 through 2010. During the last four years the lost wealth has averaged over $80 billion per year." James Bessen, Jennifer Ford and Michael Meurer, *The Private and Social Costs of Patent Trolls*, Boston Univ. School of Law, Working Paper No. 11-45 (2011) ("Bessen 2011") at 2.[21]

One study found that software patents account for 74-93 percent of NPE lawsuits. John Allison, Mark A. Lemley and Joshua Walker, *Patent Quality and Settlement Among Repeat Patent Litigants*, 99 Geo. L. J. 677, 695-96 (2010).[22] Thus, the software industry bears a large percentage of NPE-related costs. As the congressional study noted:

> Experts attribute the proliferation of PAEs over the past 10 to 15 years to the explosion of the information technology (IT) industry and patent law's struggle to adapt to the unique issues presented by this new frontier of innovation. They indicate that the PAE business model is not about licensing patents generally but *high-tech* patents in particular, including those on software and business methods or processes related to software, as well as computers and electronics.

---

20. Available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2091210.

21. Available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1930272.

22. Available at: http://georgetownlawjournal.org/files/pdf/99-3/AllisonLemleyWalker%2520677-712.PDF.

Yeh, *supra*, at 8 (footnotes omitted). Several technology companies have publicly reported that they have increasingly become the subject of patent litigation lawsuits by NPEs in the last ten years. *See, e.g.,* GAO Report, *supra*, at 16.

The litigation explosion particularly burdens small companies, which often find themselves the targets of these suits. One study has found that nearly 75 percent of venture capitalists have had their portfolios impacted by litigation from a patent troll. Colleen Chien, *Patent Assertion and Startup Innovation*, New America Foundation (Sept. 2013) at 10.[23] More than half of the defendants involved in litigation brought by patent NPEs are companies with annual revenues of $10 million or less. *Id.* at 11. Litigation-based legal expenses can kill small startups entirely, and the mere threat of those expenses can chill innovation. In a small company, key management and engineers must deal with an NPE claim. Colleen Chien, *Startups and Patent Trolls* (Santa Clara Univ. School of Law, Legal Studies Research Paper Series, Accepted Paper No. 09-12, 2012) at 10-13.[24] Professor Chien noted:

> Although large companies tend to dominate patent headlines, most unique defendants to patent troll suits are small. Companies with less than $100M annual revenue represent at least 66% of unique defendants and the majority of

23. Available at: http://www.newamerica.net/sites/newamerica.net/files/policydocs/Patent%20Assertion%20and%20Startup%20Innovation_updated.pdf.

24. Available at: http://ssrn.com/abstract=2146251.

> them make much less than that: at least 55% of unique defendants in PAE suits make under $10M per year. Suing small companies appears to distinguish PAEs from operating companies, who sued companies with less than $10M of annual revenue only 16% of the time, based on unique defendants.

*Id.* at 1-2. This results in small cash-poor companies becoming vulnerable targets that lack leverage to deal with an NPE claim, leaving them stuck paying nuisance settlements regardless of the merits of the underlying claim. *Id.* at 3. With small- and medium-sized companies making up 90 percent of the defendants in NPE suits, Bessen 2012, *supra*, at 13, such nuisance settlements are widespread.

In another troubling trend, small companies increasingly find themselves targeted by NPEs based on their use of basic technologies, for actions such as using a scanner or wireless Internet. Joe Mullin, *Patent Trolls Want $1,000—For Using Scanners*, Ars Technica (Jan. 2, 2013) (stating "2012 may go down as the year of the user").[25] This problem is widespread: one NPE alone, who claims to own a patent covering the technology behind scanning documents to email, has sent demand letters to at least 16,465 companies across the country accusing those companies of patent infringement. *See MPHJ Tech. Invs., LLC v. FTC*, No. 6:14-cv-00011 WSS (W.D. Tex. 2014).[26]

---

25. Available at http://arstechnica.com/tech-policy/2013/01/patent-trolls-want-1000-for-using-scanners.

26. Information available at: https://ia600804.us.archive.org/28/items/gov.uscourts.txwd.669787/gov.uscourts.txwd.669787.1.0.pdf

Moreover, one analysis has found that the top ten patent litigation campaigns over the past three years (as determined by number of named defendants) all involved users and implementers of a technology. Chien, *Patent Assertion and Startup Innovation*, *supra*, at 12. Small companies are particularly vulnerable to such lawsuits, as they are unlikely to have been able to negotiate indemnity protection. *Id.* at 13.

### 3. Unclear Software Patents Make the Costs of Litigation Even Worse

Finally, NPEs have found success with their "business model" by asserting hard-to-understand patents with overbroad claims. The type of software patents that NPEs use to litigate[27] are "notoriously difficult to interpret." Bessen 2012, *supra*, at 8. As Professor Lemley notes:

> A related problem is the uncertainty associated with the meaning and scope of a software patent. Unlike chemistry and biotechnology, where we have a clear scientific language for delineating what a patent claim does and doesn't cover, there is no standard language for software patents. Accordingly, no one can really know what a software patent covers until the court has construed the language of the patent claims.

27. As noted previously, software patents account for 74-93 percent of NPE lawsuits. Allison et al., *supra*, at 695-96.

Mark A. Lemley, *Software Patents and the Return of Functional Claiming*, 2013 Wis. L. Rev. 905, 930 (2013) (footnote omitted) ("Lemley"). (This claiming problem is one of the subjects of *Nautilus, Inc. v. Biosig Instruments, Inc.*, 715 F.3d 891, *cert. granted*, 82 U.S.L.W. 3195 (U.S. Jan. 10, 2014) (No. 13-369).)

In other words, "software patents have 'fuzzy boundaries': they have unpredictable claim interpretation and unclear scope . . . and the huge number of software patents granted makes thorough search to clear rights infeasible, especially when the patent applicants hide claims for many years by filing continuations. This gives rise to many situations where technology firms inadvertently infringe." Bessen 2011, *supra*, at 24. This lack of clarity directly feeds into the NPE business model and, consequently, the recent increase in both NPE and software patent litigation. Specifically, "there is a business opportunity based on acquiring patents that can be arguably read to cover existing technologies and asserting those patents, litigating if necessary in order to obtain a licensing agreement. . . . the patent troll business model only makes economic sense when there is such inadvertent infringement." *Id.*

All this means is that there has been "a substantial increase in litigation risk and hence a disincentive to invest in innovation." Bessen Generation, *supra*, at 20.

## II. A Robust Section 101 Will Help Avoid Widespread Preemption of Abstract Ideas that Inherently Belong in the Public Domain

### A. Section 101's Concern with Preemption Is Intended to Serve the Public Interest

As the Court has noted, one of § 101's core tenets is to eliminate the threat of unnecessarily preempting further innovation. In *Benson,* for instance, this Court declined to grant patent protection when the "mathematical formula involved …[had] no substantial practical application except in connection with a digital computer … [for fear that] the patent would wholly pre-empt the mathematical formula and in practical effect would be a patent on the algorithm itself." *Benson,* 409 U.S. at 71-72. And more recently in *Bilski v. Kappos,* this Court highlighted that the threshold question of abstractness serves as an important check on inventions that could "pre-empt use of [an abstract] approach in all fields, [] . . . effectively grant[ing] a monopoly over an abstract idea." *Bilski*, 130 S. Ct. at 3231.

It goes without saying that the patent system is one of incentives. Not only does Article I, § 8, cl. 8 of the Constitution make that clear with its admonition that Congress "promote the Progress of Science and useful Arts," but so does this Court's jurisprudence. For instance, in *Pfaff v. Wells Electronics, Inc.,* 525 U.S. 55, 63 (1998), the Court said:

> the patent system represents a carefully crafted bargain that encourages both the creation and the public disclosure of new and

> useful advances in technology, in return for an exclusive monopoly for a limited period of time. The balance between the interest in motivating innovation and enlightenment by rewarding invention with patent protection on the one hand, and the interest in avoiding monopolies that unnecessarily stifle competition on the other, has been a feature of the federal patent laws since their inception.

It is no wonder, then, that the doctrine of preemption serves as a helpful lens through which to view the oftentimes complicated questions of incentives and competition that naturally arise when the government finds itself in the business of granting monopolies. When the grant of the monopoly is too wide, second movers lose incentive to create. *Medtronic, Inc. v. Mirowski Family Ventures, Inc.,* 134 S. Ct. 843, 851 (2014) ("The public interest, of course, favors the maintenance of a well-functioning patent system. But the 'public' also has a 'paramount interest in seeing that patent monopolies … are kept within their legitimate scope.'") (internal citations omitted). The preemption doctrine helps insulate that public interest from the creep of overbroad monopolies:

> But even though there are other doctrines that can be used to protect competitive development, a preemption doctrine is nonetheless critical. All the other requirements permit patents—they will simply be narrower than might otherwise be claimed, or delayed until a use is identified. Yet because patents—once issued—cover all uses, there will be situations where even very narrow patents block off too much, especially

> in areas (like computer science and genetic diagnostics) where applications flow easily from basic discoveries.

Rochelle C. Dreyfuss & James P. Evans, *From Bilski Back to Benson: Preemption, Inventing Around, and the Case of Genetic Diagnostics*, 63 Stan. L. Rev. 1349, 1359 (2011).

Section 101 is *the* section of the Patent Act situated to do the heavy lifting with regard to preemption. In *Mayo Collaborative Services v. Prometheus Laboratories, Inc.*, this Court unanimously "decline[d] . . . to substitute §§ 102, 103, and 112 inquiries for the better established inquiry under § 101." 132 S. Ct. 1289, 1304 (2012) (rejecting the argument that §§ 102, 103, and 112 could perform § 101's "screening function"). Otherwise, "to shift the patent-eligibility inquiry entirely to these later sections risks creating significantly greater legal uncertainty, while assuming that those sections can do work that they are not equipped to do." *Id.*

### B. At a Minimum, Performance of an Otherwise Abstract Idea "on a Computer" or "on the Internet" Should Not Make that Idea Non-Abstract

In effort to avoid preemption, courts have rightfully found that performing an abstract idea on a general purpose computer, without more, does not make that invention patent-eligible. For example, in *Dealertrack, Inc. v. Huber*, 674 F.3d 1315 (Fed. Cir. 2012), the Federal Circuit found claims impermissibly abstract when they were "silent as to how a computer aids the method, the extent to which a computer aids the method, or the

significance of the computer to the performance of the method," even though the patent at issue limited the claims to "computer-aided." *Id.* at 1333. Following the reasoning first laid out in *Benson,* the Federal Circuit held:

> Because the computer here "can be programmed to perform very different tasks in very different ways," *Aristocrat,* 521 F.3d at 1333, it does not "play a significant part in permitting the claimed method to be performed." *Cybersource,* slip op. at 19 (citing *SiRF Tech.*, 601 F.3d at 1333). Simply adding a "computer aided" limitation to a claim covering an abstract concept, without more, is insufficient to render the claim patent eligible. *See SiRF,* 601 F.3d at 1333 ("In order for the addition of a machine to impose a meaningful limit on the scope of a claim, it must play a significant part in permitting the claimed method to be performed, rather than function solely as an obvious mechanism for permitting a solution to be achieved more quickly, i.e., through the utilization of a computer for performing calculations.").

*Dealertrack,* 674 F.3d at 1333; *see also CLS Bank Int'l v. Alice Corp.*, 717 F.3d 1269, 1283-84 (Fed Cir. 2013) (en banc) ("bare field-of-use limitations cannot rescue a claim from patent ineligibility where the claim as written still effectively preempts all uses of a fundamental concept within the stated field.") (Lourie, J. concurring).

Unfortunately, as the instant case demonstrates, the Federal Circuit has failed to demonstrate consistency with regard to its own rule that tying an otherwise abstract

idea to a general purpose computer does not make that idea patent-eligible. The claims at issue here, like those in *Dealertrack, Benson,* and *Bilski,* take an abstract idea—using a third-party intermediary to settle risk—and merely tie that idea to a general "data processing system" (in the system claims) and a "computer program … comprising a computer readable storage medium" (in the media claims). CLS Bank Br. at 48-51. Nevertheless, five judges on the Federal Circuit would have found these system claims patent-eligible.

Essentially, if those judges had their way, and the patents were found patentable, Alice Corp. would own practically all implementations of the general idea of using third-party intermediaries to eliminate risk (essentially realizing Chief Judge Archer's warning in *Alappat*, *supra* at 22). Can one even imagine an implementation of that idea in modern times that would not take place on a "data processing system"? As this Court made clear in *Bilski*:

> The Information Age empowers people with new capacities to statistical analyses and mathematical calculations with a speed and sophistication that enable the design of protocols for more efficient performance of a vast number of business tasks. If a high enough bar is not set when considering patent applications of this sort, patent examiners and courts could be flooded with claims that would put a chill on creative endeavor and dynamic change.

130 S.Ct. at 3229.

These dangers appear not just when inventions are tied to general purpose computers, but also to the Internet, as in a patent that merely claims its invention is conducted "online." Another case, *WildTangent, Inc. v. Ultramercial, LLC.*, 722 F.3d 1335 (Fed. Cir. 2013) *petition for cert. filed*, (U.S. Aug. 23, 2013) (No. 13-255), which currently has a petition for certiorari pending before the Court, well illustrates this point. The *Ultramercial* case involves U.S. Patent No. 7,346,545 ("the '545 Patent"), which claims the method of distributing ads "over the Internet." This supposedly allows consumers to view ads online before accessing content protected by intellectual property rights in lieu of paying for that content. *See generally* '545 Patent, col. 8, lines 5-48; col. 9, line 1 to col. 10, line 9 (claims 1 and 8).

When taken together, the claims of the '545 Patent contain nothing more than an abstract process, at best solely tied "to a particular technological environment." *See Bilski*, 130 S. Ct. at 3230. The initial Federal Circuit panel that investigated the '545 Patent held that it was not impermissibly abstract because many of claimed steps "*are likely* to require intricate and complex computer programming" and that "certain of these steps clearly require specific application to the Internet and a cyber-market environment." *Ultramercial, LLC v. Hulu, LLC*, 657 F.3d 1323, 1328 (Fed. Cir. 2011) (emphasis added). But claims that "likely" require complex programming to apply to "the Internet" simply do not lead to the conclusion that that '545 Patent's invention—"a method for monetizing and distributing copyrighted products," essentially using advertising—is anything other than impermissibly abstract.

Even if the claims were "likely" to require programming, in fact they did ***not*** recite any programming steps, and even if they did recite such steps, the '545 patent would still be impermissibly abstract under § 101. This is because, of course, much of the business we conduct on a daily basis now takes place on the Internet. For example, 71 percent of U.S. households used the Internet in 2011. *Computer and Internet Use in the United States*, U.S. Census Bureau (May 2013).[28] Increasingly, the public uses the Internet for everyday commercial activities formerly done on Main Street. Given this reality, merely tying an otherwise abstract business method to that environment cannot be sufficient to make that method patentable, any more than tying such a method to a public road.

"On the Internet," just like a "data processing system" or a "computer readable storage medium," has become a commonplace term in today's world. As such, tying an invention simply to the Internet or a general purpose computer should never alone be enough to make an otherwise abstract idea patent eligible. Indeed, "[g]iven the ubiquity of computers in contemporary life, allowing a process to become patentable simply because it is computer-implemented or invokes the use of the Internet would render the subject-matter eligibility criteria contained in section 101 virtually meaningless." *MySpace, Inc. v. GraphOn Corp.,* 672 F.3d 1250, 1267 (2012) (Mayer, J. dissenting). This Court should enact a clear rule saying as much.[29]

---

28. Available at: http://www.census.gov/prod/2013pubs/p20-569.pdf.

29. This is not to say that the Court should not, too, apply a rule defining what inventions are otherwise abstract or not. It

## III. Section 101, When Properly Applied, Can Be a Powerful Tool to Curb Litigation Abuse

This inability to discern a patent's scope or assess its validity leads to two distinct unfortunate results: (1) it drives parties to litigate cases that might otherwise fairly settle; and/or (2) it encourages parties to accept settlements that do not reflect the real value of the technology at issue (or the merits of the case). Thus, the present state of confusion surrounding § 101 blunts an otherwise powerful incentive to dispose of cases at the summary judgment stage (or earlier), before the need to engage in expensive and lengthy discovery.

Widespread agreement exists that the harm from NPEs outweighs any benefit they provide. Yeh, *supra*, at Summary, 2, 6; *supra* at Section I.C.2. Despite this, there is an apparent lack of consensus as to the best way to fix the problem. One crucial way to stem abuse by NPEs is to create incentives for those facing litigation (or litigation threats) to pursue their meritorious defenses of noninfringement and invalidity. *Id.* at 5 (citing Allison, Lemley & Walker, *supra*, at 694) ("Studies suggest that [non-practicing entities] rarely prevail on the merits. Their win rate in cases decided on the merits is just 8 percent, versus 40 percent for other entities . . . . But they persist with litigation nonetheless, apparently supported by the licensing fees obtained by posing a credible threat of *extended* litigation.") (emphasis added).

---

is merely to say that, at a minimum, application of an otherwise abstract idea to the Internet or a general purpose computer does not somehow make that abstract idea miraculously patentable.

Indeed, the most troubling aspect of the NPEs' business model—the push to deter meritorious litigation in lieu of cheaper licensing deals—is necessarily discouraged by additional opportunities to appropriately dispose of cases at the early stages of litigation. Defendants who might otherwise choose to settle rather than litigate a meritorious case will be incentivized to do so if they might be able to see that case disposed of efficiently. Finally, the ability to address § 101 issues at early stages of litigation will not harm the rights of any non-practicing entity (or of any plaintiff) who attempts to enforce a patent that is non-abstract.

Thus, several cases have properly decided § 101 issues at an early stage, either by summary judgment or on a motion to dismiss. *See, e.g.*, *Bancorp Servs., L.L.C. v. Sun Life Assurance Co.*, 687 F.3d 1266, 1273-74 (Fed. Cir. 2012); *OIP Techs., Inc. v. Amazon.com, Inc.*, No. C-12-1233 EMC, 2012 WL 3985118 (N.D. Cal. Sept. 11, 2012); *Glory Licensing, L.L.C. v. Toys "R" Us, Inc.*, No. 09-4252 FSH, 2011 WL 1870591 (D. N.J. May 16, 2011); *Uniloc USA, Inc. v. Rackspace Hosting, Inc.*, No. 12-CV-375 (E.D. Tex. March 27, 2013); *Cyberfone Systems, LLC v. Cellco P'ship, et al.*, No. 11-827-SLR through 11-835-SLR (D. Del. Aug. 16, 2012). Moreover, in *Mayo,* this Court has already stated that § 101 should play that role: "to shift the patent-eligibility inquiry entirely to these late sections [§§ 102, 103, 112] risks creating significantly greater legal uncertainty, while assuming that those sections can do work that they are not equipped to do." 132 S. Ct. at 1304. As this Court understands, the trend of using § 101 early in litigation should be encouraged, and this case serves as a proper vehicle to do just that.

## CONCLUSION

The Court should affirm the judgment below, with directions to the Federal Circuit that § 101 should strictly limit the scope of patent eligible subject matter.

February 27, 2014

Respectfully submitted,

PAMELA SAMUELSON
*Richard M. Sherman Distinguished Professor of Law*
UNIVERSITY OF CALIFORNIA, BERKELEY, SCHOOL OF LAW
434 Boalt Hall
(North Addition)
Berkeley, California 94720
(510) 642-6775

JULIE P. SAMUELS
*Counsel of Record*
MICHAEL BARCLAY
DANIEL K. NAZER
ELECTRONIC FRONTIER FOUNDATION
815 Eddy Street
San Francisco, CA 94109
(415) 436-9333
julie@eff.org

*Attorneys for Amicus Curiae Electronic Frontier Foundation*