No. 13-298

IN THE
Supreme Court of the United States

ALICE CORPORATION PTY., LTD.,
Petitioner,
v.
CLS BANK INTERNATIONAL AND CLS SERVICES LTD.,
Respondents.

On Writ of Certiorari to the United States Court of Appeals for the Federal Circuit

BRIEF OF MICROSOFT CORPORATION, ADOBE SYSTEMS INC., AND HEWLETT-PACKARD COMPANY AS *AMICI CURIAE* IN SUPPORT OF AFFIRMANCE

HORACIO E. GUTIÉRREZ
JULE L. SIGALL
DAVID W. JONES
MICROSOFT CORPORATION
One Microsoft Way
Redmond, WA 98052
(425) 882-8080
Counsel for Microsoft Corporation

JEFFREY A. LAMKEN
Counsel of Record
MICHAEL G. PATTILLO, JR.
MOLOLAMKEN LLP
The Watergate, Suite 660
600 New Hampshire Ave., NW
Washington, D.C. 20037
(202) 556-2000
jlamken@mololamken.com
JOHN M. WHEALAN
4613 Merivale Road
Chevy Chase, MD 20815
(202) 994-2195
Counsel for Amici Curiae

[List of Counsel continued on Inside Cover]

WILSON-EPES PRINTING CO., INC. – (202) 789-0096 – WASHINGTON, D.C. 20002

ANTHONY BACA
CYNTHIA BRIGHT
CURTIS ROSE
ROBERT WASSON
HEWLETT-PACKARD COMPANY
OFFICE OF THE GENERAL
COUNSEL
3000 Hanover Street
Palo Alto, CA 94304
(650) 857-1501

Counsel for Hewlett-Packard Company

DANA S. RAO
ADOBE SYSTEMS INC.
345 Park Avenue
San Jose, CA 95110
(408) 536-3298

Counsel for Adobe Systems Inc.

QUESTION PRESENTED

Whether the courts below correctly concluded that none of the asserted claims are patent eligible under 35 U.S.C. § 101.

TABLE OF CONTENTS

TABLE OF CONTENTS—Continued

TABLE OF AUTHORITIES

TABLE OF AUTHORITIES—Continued

Page(s)

OTHER AUTHORITIES

TABLE OF AUTHORITIES—Continued

Page(s)

TABLE OF AUTHORITIES—Continued

TABLE OF AUTHORITIES—Continued

IN THE

Supreme Court of the United States

NO. 13-298

ALICE CORPORATION PTY., LTD.,

Petitioner,

v.

CLS BANK INTERNATIONAL AND CLS SERVICES LTD.,

Respondents.

On Writ of Certiorari to the United States Court of Appeals for the Federal Circuit

---

BRIEF OF MICROSOFT CORPORATION, ADOBE SYSTEMS INC., AND HEWLETT-PACKARD COMPANY AS *AMICI CURIAE* IN SUPPORT OF AFFIRMANCE

INTEREST OF *AMICI CURIAE*[1]

Microsoft Corporation is a leader in the technology industry and an active and longstanding player in the patent system. Since its founding in 1975, it has devel-

---

[1] Pursuant to Supreme Court Rule 37.6, counsel for *amici curiae* states that no counsel for a party authored this brief in whole or in part, and no party or counsel for a party made a monetary contribution intended to fund the preparation or submission of this brief. No person other than *amici* or their counsel made a monetary contribution to this brief's preparation or submission. Counsel of record for both petitioner and respondents have consented to the filing of *amicus* briefs in letters that have been filed with the Clerk.

oped and marketed a wide range of software, services, and hardware products, including the flagship Windows operating system, the Office suite of productivity applications, the Surface tablet computer, and the Xbox gaming system.

Adobe Systems Inc. is devoted to changing the world through digital experiences. With its over 30-year history in the software and services fields, and flagship products like Photoshop, Acrobat, Flash, and Adobe Marketing Cloud, Adobe provides the tools the world uses to create groundbreaking digital content, deploy it across all screens, measure and optimize it over time, and achieve greater business success.

Hewlett-Packard Company is the world's largest information technology company. It offers personal computers, enterprise storage and servers, networking devices, IT management software, IT services, and imaging- and printing-related products.

The question presented in this case purports to concern the patent eligibility of computer-implemented inventions. The patents at issue, however, do not claim true computer-implemented inventions, but instead claim business methods for mitigating risk in financial transactions. *Amici* have a profound interest in ensuring that courts understand the differences between the two. As leading technology companies, *amici* currently spend $14 billion annually on research and development and hold nearly 63,000 U.S. patents, most of which relate to computer-implemented inventions. *Amici* are also frequently sued for allegedly infringing business-method patents, some of which, like the claims in this case, masquerade as computer-implemented inventions. *Amici* thus have unique and balanced insights on, and experience with, the issues presented by this case.

INTRODUCTION AND SUMMARY OF ARGUMENT

Petitioner frames the question presented as "[w]hether claims to computer-implemented inventions * * * are directed to patent-eligible subject matter within the meaning of 35 U.S.C. §101" (rather than being categorically excluded as abstract ideas). Pet. Br. i. That question has an easy, but superficial, answer: yes. But the actual question is whether petitioner's patent claims, which are directed to abstract business methods, qualify as "computer-implemented inventions" eligible for patent protection. The answer to that question is no.

Petitioner's patents claim a method of performing escrows—an otherwise unpatentable business method—to which petitioner has essentially bolted on the directive "perform that method on a computer." But merely saying, "perform an otherwise unpatentable idea on a general-purpose computer" does not make the idea patentable. In *Bilski* v. *Kappos*, 130 S. Ct. 3218 (2010), the Court held that the abstract idea of hedging was not patentable. Adding the directive to do the hedging "on a computer" would not have changed that result. The same is true here.

I. Precisely because this case does not involve a true computer-implemented invention (but rather a business method posing as one), the Court has no reason to address—or to risk the potential uncertainty and unforeseen impacts that can be created by addressing—the patentability of computer-implemented inventions and software generally. The importance of software to the progress of innovation and the economy cannot be overstated. Software provides the infrastructure for innovation in today's information age. Computers of all types rely for their operation on the same physical activity—turning

millions of tiny transistors, embedded in silicon chips, on and off in the desired sequences. It is software that defines the operation of a computer's transistors. By controlling those transistors, software can enable a single general-purpose computer to function as a camera, a telephone, a video player, a gaming device, and more.

Previously, most patentable inventions took the form of new, single-purpose mechanical devices, or improvements to existing physical mechanisms. Today, many technological inventions—both pioneering discoveries and incremental improvements—rely not on advances in mechanical design, but on advances in the software that controls computer hardware. These digital advances are the heirs to their mechanical forebears—they are no less innovative, and they serve the same practical function in advancing the state of technology. The fact that they take the form of a process implemented through software's reconfiguration of transistors, rather than modifying the physical structure of a machine, does not make them any less patent eligible.

Software-based innovation pervades every sector of the economy. In the aerospace and automobile industries, innovations like fly-by-wire controls and anti-lock brakes are not so much the product of mechanical evolution as of software advances. Such computer-implemented innovations have been considered, and should remain, just as patent eligible as their mechanical predecessors.

II. Petitioner's patents in this case cover a "business method" of the sort this Court held to be an unpatentable abstract idea in *Bilski*. While some of petitioner's claims do refer to a computer or computer implementation, none relates to an advance in computing, software, or any other technology. Nor do the claims represent an innova-

tion in how a computer may be employed to produce the desired result. Instead, the claims recite an abstract idea for conducting a business transaction and then add what amounts to the directive, "do it on a computer." Petitioner's patents are no more directed to a computer-implemented innovation for electronic transactions than a generic claim for a "molecule configured to cure cancer" describes a novel pharmaceutical for treating oncology patients.

"[T]o transform an unpatentable law of nature into a patent-eligible *application* of such a law, one must do more than simply state the law of nature while adding the words 'apply it.'" *Mayo Collaborative Servs.* v. *Prometheus Labs., Inc.*, 132 S. Ct. 1289, 1294 (2011). Likewise, to transform an otherwise unpatentable business method into a patentable computer-implemented invention, one must do more than say "do it on a computer." See *Parker* v. *Flook*, 437 U.S. 584, 593 (1978). To resolve this case, the Court's decision need not, and should not, go further.

III. This Court's § 101 cases provide well-suited threshold tests for filtering out unpatentable business-method patents, including those disguised as computer-implemented inventions. But applying them to true computer-implemented inventions may prove harder in future cases. Assessing the eligibility of software-related claims often presents difficult issues and unique concerns. The Court should not purport to issue guidelines for such claims in the absence of a record that will help the Court evaluate those complexities.

In the meantime, courts can continue to address the patent eligibility of computer-implemented inventions with reference to established § 101 principles. Those principles require courts to consider the actual patent claims themselves, taken as a whole. And they establish

that a claim invoking abstract concepts is patent eligible only if it also contains additional limitations that direct the claim to a specific, practical application of the idea—an invention that advances computer functionality or represents a new way to use a computer to achieve a result—rather than attempting to monopolize the idea itself.

ARGUMENT

I. SOFTWARE-IMPLEMENTED TECHNOLOGICAL INNOVATIONS ARE PATENT ELIGIBLE UNDER § 101

A. Software Makes Computing Technology Work

It is natural to envision innovation in computing technology in terms of hardware, as each generation of smartphones, tablets, and laptops is faster, more powerful, and more feature-rich than predecessors. And advances in computer hardware—exemplified by increases in processing power pursuant to Moore's law[2]—abound. Ultimately, however, software underlies the varied functionality and practical utility that computers provide. Without software, computers can do nothing useful. As a result, software is an indispensable aspect of innovation in every technological field and sector of the economy.

Despite the countless types of computing devices, they all operate using the same simple physical activity: the routing and re-routing of electrical signals by means of on-off switches, typically transistors on silicon chips. See generally David A. Patterson & John L. Hennessy, *Computer Organization and Design* (4th ed. 2009); Ron White, *How Computers Work* (9th ed. 2008). While diffi-

---

[2] "Moore's law" is the prediction that "[t]he number of transistors incorporated in a [semiconductor] chip"—and, roughly, the processing power of chips—will "double every 24 months." *Moore's Law and Intel Innovation*, http://www.intel.com/content/www/us/en/history/museum-gordon-moore-law.html.

cult to visualize, the complex functions that computers perform come down to the rapid manipulation of millions of transistors in particular configurations.

Software defines the operations of those transistors. Software developers conceive of the functionality they want a computer to perform, and then write instruction code using a "programming language" that allows operations to be defined through syntax and semantics that humans can readily understand. But the functions developed in programming language must then be reduced to "machine language," which provides instructions at the rudimentary level of turning the computer's transistors on and off in the desired sequences. See Patterson & Hennessy, *supra*, at 11-13, 20-21, 76-80.

Different software programs enable a general-purpose computer to perform new and different tasks. "Such programming creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software." *In re Alappat*, 33 F.3d 1526, 1545 (Fed. Cir. 1994) (en banc). Simply put, it is software that transforms the single set of circuits in a general-purpose computer into a word processor, camera, telephone, video player, or gaming device. Where formerly inventors had to build new devices, software can endow a single computing device with new features, capabilities, and advances in performance by reconfiguring the device's existing transistors.

B. Software-Enabled Inventions Are the Modern-Day Heirs to Mechanical Inventions

Software innovations are the modern-day heirs to unquestionably patent-eligible mechanical inventions—they are precisely the sort of "new and useful advances in

technology" that the patent system was designed to encourage and protect. *Pfaff* v. *Wells Elecs., Inc.*, 525 U.S. 55, 63 (1998). The fact that technological advances may take the form of software reconfiguring a computer's transistors, rather than an improvement in the mechanical structure of a machine, does not make them any less patent eligible.

The technological evolution from mechanical typewriter to today's word-processing software illustrates that point. Christopher Latham Sholes did not invent the typewriter, but he developed the first commercially viable one. In 1868, he received a patent for his "Improvement in Type-Writing Machines." At that time, typewriters were purely mechanical devices. Sholes' innovations were likewise mechanical: They included "a better way of working the type-bars," "of moving and regulating the movement of the carriage," and "of holding, applying, and moving the inking-ribbon." U.S. Patent No. 79,265, at 1 (issued June 23, 1868).

Other innovations followed—they, too, were patent-eligible "new and useful improvement[s]" in a typewriter's mechanics. 35 U.S.C. §101. The capital-shift function allowed users to toggle between upper- and lower-case characters on a single type bar. Several patents were awarded in the late 19th century for implementations of a capital-shift mechanism. *E.g.*, U.S. Patent Nos. 202,923 (issued Apr. 30, 1878), 551,842 (issued Dec. 24, 1895), 564,699 (issued July 28, 1896). And the invention of a mechanism for proportional spacing, which adjusts the distance between letters in proportion to their varying widths, produced more aesthetically pleasing typesetting. See U.S. Patent No. 2,111,410 (issued Apr. 15, 1935).

In the 1930s, incorporation of electric motors increased typewriters' speed. Several patents for that technology were issued to Russell Thompson of Electromatic Typewriters, Inc. See, *e.g.*, U.S. Patent Nos. 1,940,155 (issued Dec. 19, 1933), 2,063,530 (issued Dec. 8, 1936). IBM acquired Electromatic, using its patents to produce the first commercially successful electric typewriter. See *The History of IBM Electric Typewriters*, http://www.03.ibm.com/ibm/history/exhibits/modelb/modelb_history.html. IBM further advanced the art in the 1960s with its Selectric typewriter, replacing the movable carriage and individual typestrikers with a golf-ball-shaped revolving printing head. See U.S. Patent No. 2,895,584 (issued July 21, 1959).

In the 1970s, typewriters began shifting to digital technology, as companies developed single-purpose word-processing computers. Lexitron, for example, patented an "Electronic Text Display and Processing System" that included a display, a memory, a processor, and a keyboard. U.S. Patent No. 3,810,107, col.1, ll.50-57 (issued May 7, 1974) ("'107 Patent"). Functionality previously achieved through gears and levers—like proportional font spacing—was re-developed using software that provided instructions to computing hardware. Those digital improvements served the same function as, and were no less technologically innovative than, their mechanical forebears. Accordingly, the inventions were patentable. See, *e.g.*, U.S. Patent No. 3,654,609 (issued Apr. 4, 1972).

Digital word processors also introduced entirely new features. Early word-processing devices allowed operators to edit text on a display before committing it to paper. See, *e.g.*, '107 Patent, col.1-2. Later inventions utilized software to identify misspellings and recommend

corrections. See U.S. Patent No. 4,859,091, col.1 (issued Aug. 22, 1989).

Today, word-processing software runs on general-purpose computers. Manufacturers no longer need to design customized devices for different uses. Instead, software applications tailored to different specialties—like Microsoft Word for business documents, Adobe Acrobat for printed brochures, and Sublime Text for software code—transform the same general-purpose computer into a specialized word-processing device. See *Alappat*, 33 F.3d at 1545. In a mechanical typewriter, an etched piece of metal stamped a letter on paper. Now, software renders that letter so it will appear on a pixel-oriented LCD display or print on a printer. See U.S. Patent No. 5,740,456, col.1 (issued Apr. 14, 1998).

Other software-implemented advances in word processing have no analog in the mechanical typewriter. Today's word-processing software can spell-check documents without human input; automatically correct text; call up a graphical interface integrating e-mail functionality within the word-processing application; and even create web pages. Our system has rightly encouraged and rewarded these software-implemented inventions through patents. See U.S. Patent Nos. 5,787,451 (issued July 28, 1998), 5,761,689 (issued June 2, 1998), 6,405,225 (issued June 11, 2002), 7,340,675 (issued Mar. 4, 2008).

C. Software Advances Pervade Today's Technological Landscape

As the story of the typewriter demonstrates, many types of technological advances that once were achieved by changing the device's physical structure are now implemented through advances in software. Software drives more than general-purpose computers like PCs and tablets. Programmable microchips—and the soft-

ware that runs them—are now standard components in everything from mobile phones to washing machines. Software enables improvements and new capabilities in countless devices, without requiring changes to the machinery the software operates.

The reach of software-based technological advances thus extends far beyond the "software" industry. The aerospace industry has played a leading role. "Often expensive new uses of computing would be developed" in the aerospace sector, "and then later transferred to other industries." James W. Cortada, *The Digital Hand: How Computers Changed the Work of American Manufacturing, Transportation, and Retail Industries* 154 (2003). For example, an "American fighter of World War II could be built on the basis of roughly 8,000 blueprints." *Id.* at 155. By the 1960s, "a military plane required between 50,000 and 100,000 drawings." *Ibid.* The Industry developed "automated drafting tools" to keep pace. *Ibid.* Later, the development of computer-assisted design software programs ("CAD") enabled engineers to "simulat[e] design, stress, [and] performance of materials" before ever building prototypes. *Ibid.*

Software transformed not only aerospace design, but the vehicles themselves. One example is "advanced vehicle control software." Cortada, *supra*, at 154. Aircraft once were operated through mechanical controls. When the pilot made a steering adjustment by moving the yoke, a series of cables and connectors would physically relay the pilot's movements to airplane control surfaces (the ailerons, elevator, and rudder). James E. Tomayko, *Computers Take Flight: A History of NASA's Pioneering Digital Fly-By-Wire Project* 11-12 (2000), http://www.nasa.gov/centers/dryden/pdf/182985main_DFBW_rev1.pdf. Later, hydraulic systems were used.

They amplified the force applied, but they still utilized physical links connecting the yoke to the control surfaces. Those have now given way to computer-controlled "fly-by-wire" systems. Dating to the computerized systems used in the Apollo spacecraft, fly-by-wire technology first appeared in an airplane in 1972 in NASA's F-8 Crusader. *Id.* at viii, 16-19.

When the pilot moves the yoke in a fly-by-wire system, it feeds data into a computer that then adjusts the control surfaces. Tomayko, *supra*, at 11-12. The computer can execute faster, finer, and more frequent adjustments than a human pilot ever could. *Ibid.* Fly-by-wire control yields enormous benefits, including lower airframe weight, increased fuel efficiency, improved handling and responsiveness, and greater safety. *Id.* at viii. Fly-by-wire is now used in everything from commercial airliners, to military fighter jets, to the Space Shuttle Orbiter. *Id.* at vii, 127-128.

Software-driven innovations originating in the aerospace industry have rapidly spread to the automotive industry. Cortada, *supra*, at 154. Everything from the initial design, to production, to the end-product cars themselves has been improved by innovations in software.

Until the 1960s, engineers designed cars by making hand-drawn sketches. Today, engineers use CAD programs. Michael Tovey *et al.*, *Sketching and Direct CAD Modelling in Automotive Design*, 21 Design Studies 569, 570-571 (2000). Three-dimensional computer modeling helps designers hit their targets for styling, energy efficiency, cost, and safety. John Brandon, *How Computers Have Revolutionised Our Cars*, TechRadar (July 4, 2013), http://www.techradar.com/us/news/car-tech/how-computers-have-revolutionised-car-design-1160141; Alan

Cane, *Car Manufacturing: Vehicle Makers Appreciate the Virtues of Virtual Design*, Financial Times (May 28, 2008), http://www.ft.com/cms/s/0/76265d22-2af8-11dd-a7fc-000077b07658.html#axzz2tL1PQkhN.

Automobile assembly was originally purely manual and mechanical. When Henry Ford introduced the Model T in 1908, workers carried components to a stationary chassis, manually attaching each piece. See Paul Ingrassia, *This Car Changed America*, The Wall Street Journal (Sept. 27, 2008), http://online.wsj.com/news/articles/SB122246777029780525. Even with Ford's introduction of the assembly line, production required an army of workers to physically bolt parts to the car frame as it moved past. That began to change in the 1960s when GM first installed an industrial robot in its Trenton, New Jersey plant. Int'l Fed'n of Robotics, *History of Industrial Robots* (2012), http://www.ifr.org/uploads/media/History_of_Industrial_Robots_online_brochure_by_IFR_2012.pdf. The technology proliferated, and computer-controlled industrial robots now perform welding, painting, and quality-control. See John Markoff, *Skilled Work, Without the Worker*, The New York Times (Aug. 18, 2012), http://www.nytimes.com/2012/08/19/business/new-wave-of-adept-robots-is-changing-global-industry.html?pagewanted=all&_r=0.

More recently, manufacturers "have realized that with the right software and safety controls," robots can "work in close proximity to humans." Will Knight, *Smart Robots Can Now Work Right Next to Auto Workers*, MIT Technology Review (Sept. 17, 2013), http://www.technologyreview.com/news/518661/smart-robots-can-now-work-right-next-to-auto-workers/. Thus, the "collaborative robot"—or "cobot"—now works alongside humans, maximizing efficiency and quality. See Eric Rog-

ell, *Car-Building Robots with Laser Eyes Can "See" Exactly Where to Place Parts for Custom-Like Fit*, Discovery News (Apr. 12, 2012), http://news.discovery.com/autos/drive/national-robotics-week-car-building-robots-with-laser-eyes-can-see-exactly-where-to-place-parts-for-custom-like-fit.htm. As a result, the same factory that required 5,000 workers in the 1960s now requires a workforce of only 1,500.

Today, cars themselves are essentially "rolling computers." Johannes Winterhagen, *Road to a Digital Future*, Pictures of the Future Magazine 64 (Fall 2012), http://www.siemens.com/innovation/apps/pof_microsite/_pof-fall-2012/_html_en/automotive-software.html. "Similar to computers and smartphones, electronic parts like sensors and microprocessors comprise the 'backbone' of today's cars." Auto Alliance, *Technologies Working Together: Autos, Computers and Electronics*, http://www.autoalliance.org/index.cfm?objectid=1D64C240-7EE5-11E3-9303000C296BA163. "Engines, transmissions, brakes, airbags, and even power windows are equipped with intelligent electronics." Winterhagen, *supra*, at 64. Computer-controlled, software-implemented "[d]river assist systems include lane departure and blind spot warnings, adaptive cruise control, automatic braking, telematics control systems and more." Auto Alliance, *supra*.

Onboard GPS receivers employ software to determine the car's location based on satellite signals. See National Geographic Education, *GPS*, http://education.nationalgeographic.com/education/encyclopedia/gps/?ar_a=1. Software also allows GPS devices to determine the best route, anticipate delays, and predict arrival time.

Software has also enabled new safety features. Antilock braking systems ("ABS"), for example, sense the rate of rotation of each wheel and use software to

determine when a computer should apply the brakes to prevent wheel lock and ensure the driver maintains control of the car. AAA Foundation for Traffic Safety, *FAQs: Anti-Lock Braking System (ABS)*, (Feb. 15, 2005), https://www.aaafoundation.org/faqs-anti-lock-braking-system-abs. Stability-control systems similarly use software to monitor each wheel and make adjustments to maximize traction. Jamie Page Deaton, *Ask the Editor: What Is Stability Control?*, U.S. News & World Report (July 22, 2011), http://usnews.rankingsandreviews.com/cars-trucks/best-cars-blog/2011/07/Ask_the_Editor_Whats_Stability_Control. And computer-controlled crash-avoidance systems can "help drivers stay in their lanes" and even "brake automatically in critical situations." Winterhagen, *supra*, at 64. Those advances were primarily the result of software innovations, rather than advances in mechanical components. And the next level of software-driven advances—self-driving cars in which software can "read road signs and traffic signals" and "navigate roadways, traffic and pedestrian hazards"—is just around the corner. Auto Alliance, *supra*.

To fuel that innovation, the automotive industry spends $18 billion annually in the U.S. alone on research and development. Ctr. for Auto. Research, *Just How High-Tech is the Automotive Industry?* 1 (2014), http://www.autoalliance.org/auto-innovation/2014-car-report. Much of that research and development is directed toward software-enabled innovation. And auto makers rely on patents to protect those investments: "[A]pproximately 5,000 new patents per year" are awarded to companies in the automotive industry. *Id.* at 2.

The aerospace and automotive industries are but two examples of industries where software, as much as new mechanical components, drives innovation. Virtually

every industry—from medicine[3] to law enforcement[4]—has been transformed. The vast majority of software-related patents today go not to traditional software companies, but to manufacturing companies that integrate software with their products. See James E. Bessen & Robert M. Hunt, *An Empirical Look at Software Patents*, 16 J. Econ. & Mgmt. Strategy 157, 171 (2007).

* * * * *

Software provides the "infrastructure" of innovation in today's "information age." President's Info. Tech. Advisory Comm., Nat'l Coordination Office for Computing, Info. & Commc'ns, *Information Technology Research: Investing in Our Future* 23 (1999), available at http://research.microsoft.com/en-us/um/people/gray/papers/pitac_report_99_2_24.pdf. Where, as in the examples above, software plays an integral role in achieving a practical, technological effect, that computer-implemented invention should be just as patent eligible under § 101 as its mechanical forebears. The fact that a software-based technological invention achieves its result through instructions that manipulate transistors on a silicon chip does not make the invention any more "abstract" than if it were a new mechanical structure.

Precisely because software innovations play such a critical role in this Nation's economy and progress, the Court should take care not to "create uncertainty as to

---

[3] See, *e.g.*, Kunio Doi, *Current Status and Future Potential of Computer-Aided Diagnosis in Medical Imaging*, 78 Brit. J. of Radiology S3, S3-S19 (2005); Univ. of Ariz. Dep't of Surgery, *Robotic-Assisted Surgery*, http://surgery.arizona.edu/unit/program/robotic-assisted-surgery.

[4] See, *e.g.*, FocusMagic, *Forensics Examples*, http://www.focusmagic.com/forensics-examples.htm; Ocean Systems, *ClearID Examples*, http://www.oceansystems.com/forensic/forensic-Photoshop-Plugins/examples.php.

the patentability of" true computer-implemented inventions, like "software * * * and inventions based on linear programming, data compression, and the manipulation of digital signals." *Bilski* v. *Kappos*, 130 S. Ct. 3218, 3227 (2010). Fortunately, this case does not require the Court to make pronouncements that could create such a destabilizing effect. As explained below, this case does not present the patentability of a genuine computer-implemented innovation. Instead, it presents the more pedestrian question whether an otherwise-unpatentable abstract business method can be rendered patentable simply by reciting a directive that it should be performed on a computer. Because the answer to that question is no, the Court should go no further than to so hold, avoiding "categorical rules that might have wide-ranging and unforeseen impacts" on types of patents not currently before the Court. *Id.* at 3229.

II. THE PATENTS AT ISSUE IN THIS CASE ARE NOT DIRECTED TO COMPUTER-IMPLEMENTED INVENTIONS AND ARE INELIGIBLE UNDER § 101

This case does not concern a computer-implemented invention. Rather, the patents at issue here are directed to an unpatentable business method combined with an equally abstract directive to perform that method using a computer. Petitioner thus misframes the matter before the Court when it seeks a broad referendum on whether "claims to computer-implemented inventions * * * are directed to patent-eligible subject matter within the meaning of 35 U.S.C. § 101" (rather than being categorically excluded under the abstract-ideas exception). Pet. Br. i. That begs the antecedent question whether petitioner's business-method-based claims truly are "computer-implemented" inventions. They are not.

An abstract idea is not patentable subject matter under § 101, but an "*application*" of an abstract idea "may well be deserving of patent protection." *Diamond* v. *Diehr*, 450 U.S. 175, 187 (1981); see *Mackay Radio & Tel. Co.* v. *Radio Corp. of Am.*, 306 U.S. 86, 94 (1939) ("While a scientific truth, or the mathematical expression of it, is not a patentable invention, a novel and useful structure created with the aid of knowledge of scientific truth may be."). There is a broad spectrum of patents that invoke use of a computer, some of which will be unpatentable "abstract ideas," while others are patentable applications. At one end, as discussed above, are computer-implemented inventions that involve legitimate innovations in computer science, computerized systems, or other technological or industrial fields. Such software innovations are no different than other types of technological inventions that have long been patent eligible under § 101. There should be no doubt that they, too, are patent eligible. See pp. 7-15, *supra*.

At the other end of the spectrum are many so-called "business method" patent claims. Such claims primarily involve processes in non-technological fields, like business or finance. Because such claims are concerned with intangible concepts about organizing human behavior and transactions, they often will implicate the "abstract ideas" exception to § 101. *Bilski*, 130 S. Ct. at 3229-3230. The fact that a business-method claim may also instruct that the concept be implemented using a computer is not itself sufficient "to transform an unpatentable [abstract idea] into a patent-eligible *application* of such [an abstract idea]." *Mayo Collaborative Servs.* v. *Prometheus Labs., Inc.*, 132 S. Ct. 1289, 1294 (2011).

Petitioner's patents fall into the latter category. The claims all focus on a method for reducing settlement risk

in financial transactions. Some claims may mention computers, but there is no connection between the method described and technological innovation; the references to computers are incidental or pretextual. The putative invention is the method, and that cannot be changed by saying, "and do it on a computer." Far from being directed to patentable computer-implemented inventions, the patents concern unpatentable "abstract ideas" for organizing human obligations. See *Bilski*, 130 S. Ct. at 3225. It is thus not true that, "if all these claims * * * are not patent eligible, this case is the death of * * * all * * * software patents." Pet. App. 85a-86a (Moore, J., dissenting in part). Confirming the invalidity of petitioner's business-method patents would not, under an appropriate standard, imperil the patent eligibility of computer-implemented inventions that represent true technological advances.

A. Adding "a Computer" to the Steps of an Abstract Business Method Does Not Make a Claim Patent Eligible

This case involves so-called "business method" patents. Business-method patents are directed primarily to "methods of organizing human activity." U.S. Br. 8, *Bilski* v. *Kappos*, No. 08-964 (U.S. Sept. 2009) ("U.S. *Bilski* Br."). The purported innovation typically concerns the practice or management of business or financial concepts. While the patent may make token references to using a computer when reciting the steps of the business method, the patent, on its face, is not technological in nature. The mention of a computer is not integral to performing the method, and the claims do not disclose any contribution to computer technology.[5]

---

[5] In *Bilski*, the United States urged that claims directed to "technological and industrial processes" should be patent eligible,

Patent claims directed to business methods are different in kind from claims directed to technological or industrial computer-implemented innovations. Indeed, "[f]or centuries, it was considered well established that a series of steps for conducting business was not, in itself, patentable." *Bilski*, 130 S. Ct. at 3232 (Stevens, J., concurring in the judgment). While the Court has declined to hold that § 101 "categorically excludes" such business-method patents, *id.* at 3228 (majority opinion), it has cautioned that they often "raise special problems in terms of vagueness and suspect validity," *id.* at 3229 (plurality opinion); see also *eBay Inc.* v. *MercExchange, L.L.C.*, 547 U.S. 388, 397 (2006) (Kennedy, J., concurring). Chief among those is the concern that such patents may not be patent eligible because they seek to monopolize "abstract ideas" about economic endeavors. See *Bilski*, 130 S. Ct. at 3229-3230.

Where a business-method-based patent would otherwise cover only an "abstract idea," the mere fact that it also recites a computer is not enough to make it a patent-eligible, computer-implemented invention. "[T]o transform an unpatentable [abstract idea] into a patent-eligible *application* of such [an abstract idea], one must do more than simply state the [abstract idea] while adding the words 'apply it.'" *Mayo*, 132 S. Ct. at 1294. Likewise, a claim that recites an abstract business-method idea does not become a patent-eligible application merely by adding the words "do it on a computer"

---

but claims directed to "methods of organizing human activity that are untethered to technology" are not. U.S. *Bilski* Br., *supra*, at 8. While the Court did not adopt that as a test for patent eligibility, it is useful for distinguishing computer-implemented inventions from business methods.

(or by adding some similarly token computer-based limitation).

Rather, patent eligibility must turn on whether the use of a computer is integral to the claimed invention itself, or whether the recited use of a computer is merely "insignificant post-solution activity" tacked on to the steps of the business method. *Diehr*, 450 U.S. at 191-192. In *Mayo*, this Court addressed the patent eligibility of claims that "purport to apply natural laws describing the relationships between the concentration in the blood of certain thiopurine metabolites and the likelihood that the drug dosage will be ineffective or induce harmful side effects." 132 S. Ct. at 1294. That correlation itself was an unpatentable law of nature. *Id.* at 1296. The Court held that the other aspects of the claims, such as the process for administering the test, did not "add *enough* to their statements of the correlations to allow the processes they describe to qualify as patent-eligible processes that *apply* natural laws," as opposed to merely seeking to claim the natural law itself. *Id.* at 1297.

Business-method claims likewise will often recite an unpatentable abstract idea. See *Bilski*, 130 S. Ct. at 3229. Generally reciting the use of a computer to implement an otherwise unpatentable method does not "add *enough*" to make it a patent-eligible computer implementation of the idea. *Mayo*, 132 S. Ct. at 1297. Bolting on a requirement that a computer be used in connection with a business method at most "limit[s] the use of the [idea] to a particular technological environment." *Diehr*, 450 U.S. at 191. But that, too, fails to make the abstract method patent eligible. *Id.* at 192.

True computer-implemented inventions—those that are inherently connected to computer technology, enable new functionality, or otherwise produce a technical effect

contributing to the utility of the computer—should not run afoul of that prohibition. See pp. 7-15, *supra*. But allowing the token recitation of a computer to establish the patent eligibility of an otherwise unpatentable, non-technological business method would "exalt[ ] form over substance." *Parker* v. *Flook*, 437 U.S. 584, 590 (1978). It is precisely the type of approach this Court has warned "would make the determination of patentable subject matter depend simply on the draftsman's art and would ill serve the principles underlying the prohibition against patents for [abstract] 'ideas' * * *." *Id.* at 593; see also *Diehr*, 450 U.S. at 192. As explained below, this is such a case: Petitioner's invocation of a computer in its patents is a fig leaf meant to disguise an unpatentable business method as a patent-eligible computer-implemented invention.

B. Petitioner's Purported Invention Is Not Patent-Eligible Subject Matter, But Rather "*Bilski* on a Computer"

1. *Petitioner's Patents Seek To Cover an Abstract Idea*

In *Bilski*, the patent claimed a method for "how buyers and sellers of commodities in the energy market can * * * hedge[ ] against the risk of price changes." 130 S. Ct. at 3223. Claim 1 recited a process involving (1) "initiating a series of transactions" in which "consumers" purchase a "commodity" from a "commodity provider" "at a fixed rate based upon historical averages" that "correspond[ ] to a risk position of said consumers"; (2) "identifying market participants for said commodity having a counter-risk position to said consumers"; and (3) "initiating a series of transactions" between the "commodity provider" and the "market participants at a second fixed rate such that" the "series of market participant transac-

tions balances the risk position of" the first "series of consumer transactions." *Id.* at 3223-3224. Claim 4 "put[ ] the concept articulated in Claim 1" into a "mathematical formula," and other claims described the application of Claims 1 and 4 to energy markets. *Ibid.*

Those claims, the Court explained, sought to cover "the *concept* of hedging risk." 130 S. Ct. at 3229 (emphasis added). The Court held that the claims "fall[ ] outside of § 101," and "are not patentable processes" because they "are attempts to patent abstract ideas." *Id.* at 3229-3230.[6] Upholding petitioners' claims "would effectively grant a monopoly over" the general concept of "risk hedging," and thus "pre-empt use of this approach in all fields." *Id.* at 3231. The fact that certain claims addressed "the application of that concept to energy markets" did not change the outcome. *Id.* at 3229. "[L]imiting an abstract idea to one field of use * * * d[oes] not make the concept patentable." *Id.* at 3231.

This case involves a similar attempt to patent an abstract idea—the concept of an escrow. See Pet. App. 28a (Lourie, J., concurring); *id.* at 82a (Rader, C.J., concurring in part and dissenting in part). For example, claim 33 of petitioner's '479 Patent recites "[a] method of exchanging obligations as between parties, each party holding a credit record and a debit record with an exchange institution, the credit records and debit records for exchange of predetermined obligations." Pet. App. 26a. It breaks the method into four steps: (1) a third-party "supervisory institution" creates "shadow" records for each stakeholder that reflect the parties' balances in

---

[6] Four Justices would have held that the claims were not patent eligible "because methods of doing business are not, in themselves, covered by the statute." *Bilski*, 130 S. Ct. at 3257 (Stevens, J., concurring in the judgment).

real financial accounts at "exchange institutions"; (2) the supervisory institution obtains a "start-of-day-balance" from the parties' exchange institutions for use in their shadow records; (3) transactions are referred to the supervisory institution, which "adjust[s] each respective party's shadow" records and "permit[s] only those transactions" for which updated records indicate there are sufficient funds; and (4) "at the end-of-day," the supervisory institution instructs the respective exchange institutions to "irrevocabl[y]" "exchange credits or debits" in the parties' financial accounts. *Id.* at 26a-27a.

For purposes of § 101 patent eligibility, that method claim is "indistinguishable from the claim in *Bilski*." Pet. App. 84a (Rader, C.J., concurring in part and dissenting in part); *id.* at 31a (Lourie, J., concurring) (claims "closely resemble those in *Bilski*"). Like the claim in *Bilski*, it covers a "fundamental economic practice" for "protecting against risk," *Bilski*, 130 S. Ct. at 3231—the use of an intermediary to "mitigate 'settlement risk.'" Pet. App. 26a (Lourie, J., concurring). While the claim breaks the process down into steps (again like *Bilski*), those steps simply "explain the basic concept" of an escrow. *Bilski*, 130 S. Ct. at 3231. The claim invokes "extravagant" *sounding* terms (like "shadow records"), Pet. App. 30a (Lourie, J., concurring), but upon analysis, each step is itself "inherent within the concept of an escrow," *id.* at 83a (Rader, C.J., concurring in part and dissenting in part). The Federal Circuit thus had no difficulty concluding, by a 7-3 vote, that the method claim here, like the claim in *Bilski*, seeks to cover an unpatentable "abstract idea"—the idea of an escrow itself—as opposed to a particular application of that idea. See Pet. App. 3a n.1 (Lourie, J., concurring).

2. *Recitation of a Computer Does Not Make the Claims Patent Eligible*

The patents at issue here also *purport* to cover "computer-based" implementations of an escrow. Petitioner's supposedly computer-based claims, however, stand in stark contrast with the software-based innovations described above (at pp. 7-15). The claims at issue here do nothing to advance the state of technology. They do not describe a new computer, enable general-purpose computers to perform new functions related to escrows, or allow computers to execute the steps of an escrow transaction faster, more efficiently, or more reliably than before. Petitioner's own description of its "computer systems, computerized methods, and computer products" claims makes clear that they are not directed to technological innovation, but to steps "to eliminate * * * settlement risk." Pet. Br. 4; see also *id.* at 7, 8-9, 10. In substance, the claims have nothing to do with computers beyond saying that one could use a computer to perform an escrow. The abstract business-method claims in *Bilski* would not have become patent eligible had the inventor simply added "perform the steps of hedging on a computer." This case should be no different.

Petitioner's method claims make that particularly clear. They simply recite the steps inherent in executing an escrow transaction—they do not even mention a computer. See pp. 23-24, *supra*. While the parties stipulated that the steps should "be interpreted to require a computer," Pet. App. 6a (Lourie, J., concurring), the claim does not describe a series of computer operations. It is thus impossible to construe the operation of a computer as being integral to executing the steps of the method, much less to read the claim as encompassing an advance in software or any computer-

related field. As a result, the parties' stipulation that a computer should be used at most stipulates to some "insignificant post-solution activity." *Diehr*, 450 U.S. at 191-192. The stipulation may "limit the use of the [idea] to a particular technological environment," but that cannot make the method patent eligible. *Id.* at 191.

Petitioner's "computer readable medium" and "system" claims fare no better. Both purport to describe tangible things—respectively, a "computer program product comprising a computer readable storage medium," Pet. App. 32a, and a "data processing system" comprised of "a data storage unit" "coupled to" "a computer," *id.* at 35a. But neither set of claims describes a technological innovation in the recited hardware or the software that makes it run. Indeed, the claims have nothing to do with advances in or new applications of computing technology to execute escrow transactions. Instead, the claims simply recite the same steps of the business method recited in the method claim, while sprinkling in references to generic computer hardware. See *id.* at 32a, 35a. Merely mentioning a computer in connection with the method steps does not "add *enough*" to the description of the abstract business method to limit claims to a specific, practical application that is narrower than the idea itself. *Mayo*, 132 S. Ct. at 1297. Simply adding "program code for causing a computer" to execute an escrow transaction, Pet. App. 32a, or generically "configur[ing]" a "computer" to execute an escrow, *id.* at 35a, "add[s] nothing of significance" to the basic idea of an escrow, *Mayo*, 132 S. Ct. at 1302.

Through the "draftsman's art," *Flook*, 437 U.S. at 593, petitioner's computer-readable-medium and system claims purport to describe tangible, technological devices. But one should not "exalt[ ] form over substance."

Id. at 590. The computer references are incidental or pretextual—they amount to nothing more than saying, "one could perform the steps of an escrow using a computer." That is not enough "to transform an unpatentable [abstract idea] into a patent-eligible *application* of [that idea]." *Mayo*, 132 S. Ct. at 1294. To resolve this case, the Court need go no further.

III. In Harder Cases, Patent Eligibility Should Turn on Whether the Claim as a Whole Recites a Specific, Practical Application of the Idea Rather Than Merely Reciting Steps Inherent in the Idea Itself

Far from describing a technological innovation in computing technology, the patents at issue disclose an abstract idea and add to it only a bald directive to implement it on a computer. Because § 101 serves as "a 'coarse' gauge of the suitability of broad subject matter categories for patent protection," *Ultramercial, Inc.* v. *Hulu, LLC*, 722 F.3d 1335, 1339 (Fed. Cir. 2013), § 101's "threshold test," *Bilski*, 130 S. Ct. at 3225, is well-suited to the evaluation of such unpatentable business methods. Given their non-technical nature, the substance and preemptive scope of such patents often will be readily apparent to a non-expert district court judge on the face of the claim language. See *id.* at 3231. Efforts to use the "draftsman's art," *Flook*, 437 U.S. at 593, to dress up an abstract business-method claim as something it is not will generally be transparent.

When it comes to true computer-implemented inventions, however, a blunt or simplistic application of the "abstract ideas" exception poses grave risks. Software is heavily dependent upon the use of algorithms. And it is "clear that a process is not unpatentable simply because it contains a * * * mathematical algorithm." *Flook*, 437

U.S. at 590. Where a software patent describes a practical application of an algorithm as part of a process that produces a technological effect or useful result on a computer—*e.g.*, encryption, data compression, higher-speed processing—that should be patent eligible under this Court's precedent. See *Diehr*, 450 U.S. at 187-188. By contrast, where a patent claims use of an algorithm merely to produce "a number," but does not utilize that calculation in service of a specific technological application, that may not be patent eligible under existing precedent, even where used on a computer. *Flook*, 437 U.S. at 585. Regrettably, it may be very difficult for a lay judge, with little technical understanding of how computers actually operate, to discern the difference between the two. In such cases, expert testimony may be necessary to illuminate the technological principles at issue, and claim construction required to determine the precise boundaries (and actual preemptive scope) of the patent. Even then, it still may be difficult to determine whether the patent covers an "abstract idea" or not.

Precisely because that difficulty is not present here, the Court should avoid "adopting categorical rules" and broad pronouncements regarding § 101 patent eligibility "that might have wide-ranging and unforeseen impacts"—indeed, the potential to destabilize multiple industries. *Bilski*, 130 S. Ct. at 3229. To the extent lower courts require guidance for determining the patent eligibility of software claims—as opposed to business-method claims masquerading as software—the Court should await a case with a "fully developed factual and legal record" reflecting the issues. *Adams* v. *Robertson*, 520 U.S. 83, 92 (1997) (per curiam).

In the meantime, courts can continue to address the patent eligibility of computer-implemented inventions

using principles articulated in this Court's prior § 101 decisions. Those principles require courts to focus on the patent claims themselves, taken as a whole. And they make clear that a claim invoking an abstract concept will be patent eligible only if it also contains sufficient limitations to restrict the invention to a specific, practical application of the idea, rather than covering the idea itself. Those principles may prove difficult to apply in some circumstances, but by retaining a claim-specific approach, they help ensure that § 101 remains flexible enough "to encompass new and unforeseen inventions." *J.E.M. Ag Supply, Inc.* v. *Pioneer Hi-Bred Int'l, Inc.*, 534 U.S. 124, 135 (2001).

A. Patent Eligibility Can Be Determined Only by Looking to Each Claim, Taken as a Whole

The Court has made clear that, in determining whether a patent seeks to cover an unpatentable abstract idea or a patent-eligible application of the idea, courts must look to what is actually claimed. The primary concern animating the abstract-idea exception is the "danger" that patents that "tie up" an abstract idea's use "will inhibit future innovation premised upon" that idea, and thus "foreclose[ ] more future innovation than the underlying discovery could reasonably justify." *Mayo*, 132 S. Ct. at 1301. And one can determine how much of an idea the patent "ties up" only by looking to the claims themselves, because it is the claims that "define[ ] the scope of a patent grant." *Markman* v. *Westview Instruments, Inc.*, 517 U.S. 370, 373 (1996) (quotation marks omitted). If a claim recites additional limitations drawn to a sufficiently specific, practical means of applying the underlying abstract idea, it will ensure that the patent does not foreclose others from developing or practicing different implementations of the idea.

All of the claim's elements thus must be "considered as a whole." *Diehr*, 450 U.S. at 188. In *Mayo*, the Court described the process as looking to the "inventive concept" in the claim. 132 S. Ct. at 1294. That does not suggest that courts should isolate what they perceive to be the "heart" or "gist" of the claim and ask whether *that* is an abstract idea. Rather, it means the court should consider the *entire* "combination of elements" recited in the claim in order "to ensure that the patent in practice amounts to significantly more than the [abstract idea] itself." *Ibid.* Indeed, the Court has held that "[i]t is inappropriate to dissect the claims" by isolating the core concept from the limitations that purport to implement that concept. *Diehr*, 450 U.S. at 188; see also *Bilski*, 130 S. Ct. at 3230. Where a court focuses its analysis on the underlying concept divorced from the context of the claim, it improperly skews the analysis, because "all inventions at some level embody, use, reflect, rest upon or apply laws of nature, natural phenomena, or abstract ideas." *Mayo*, 132 S. Ct. at 1293. It likewise runs the risk of trivializing the other limitations and improperly dismissing them as "insignificant post-solution activity." *Diehr*, 450 U.S. at 191-192.

B. This Court's Precedents Guide the Distinction Between Unpatentable Abstract Ideas and Patent-Eligible Applications

This Court has explained that, in distinguishing between an unpatentable abstract idea and a patent-eligible application of that idea, courts must ask whether the limitations in "the patent claims," taken as a whole, "add *enough*" in the way of specific, practical application to differentiate the scope of the claimed invention from the underlying abstract idea itself. *Mayo*, 132 S. Ct. at

1297. This Court's precedents provide several helpful guideposts in drawing that distinction.

O'Reilly v. *Morse*, 56 U.S. (15 How.) 62 (1853), which concerned a patent by Samuel Morse, is particularly instructive because it involved some claims that were patent eligible and others that were not. Claim 8 of Morse's patent purported to cover "the use of the motive power of the electric or galvanic current, which I call electro-magnetism, however developed for marking or printing intelligible characters, signs, or letters, at any distances." *Id.* at 112-13. That claim, the Court found, recited the *idea* of using electric current for printing characters at a distance, but it did not describe any specific "process or machinery [by which] the result is accomplished"—put differently, the claim recited the idea, but no practical *application* of the idea. *Id.* at 113. As a result, the claim sought "a monopoly in [the idea's] use, however developed," thus "claim[ing] an exclusive right to use * * * manner[s] and process[es] which [Morse] has not described and indeed had not invented." *Ibid.* Allowing a patent over the idea itself, the Court noted, would "shut[ ] the door" to "every new discovery and development of the science," no matter what form it took. *Ibid.* Accordingly, "the claim [was] too broad" and not patentable subject matter. *Ibid.*

By contrast, the Court did allow other claims that recited specific, practical applications of the principle Morse had articulated. For example, Claim 3 described in detail a device for using galvanic current to print at distances, which was comprised of "the combination of machinery herein described, consisting of the generation of electricity, the circuit of conductors, the contrivance for closing and breaking the circuit, the electro-magnet, the pen or contrivance for marking, and the machinery

for sustaining and moving the paper." *Morse*, 56 U.S. (15 How.) at 85. And Claim 5 dealt not with machinery, but with the use of Morse code—*i.e.*, "the system of signs, consisting of dots and spaces, and of dots, spaces, and horizontal lines, for numerals, letters, words, or sentences," as described in the patent—"for telegraphic purposes." *Id.* at 86. For those specific, practical applications of Morse's discovery, Morse was "entitled to a patent." *Id.* at 117.

The Court's decisions in *Parker* v. *Flook*, 437 U.S. 584 (1978), and *Diamond* v. *Diehr*, 450 U.S. 175 (1981), are, together, similarly helpful guideposts, because they are "two cases in which the Court reached opposite conclusions about the patent eligibility of processes that embodied the equivalent of natural laws." *Mayo*, 132 S. Ct. at 1298. *Flook* concerned a method claim for "updating the value" of an "alarm limit" on a "process variable"—*e.g.*, temperature, pressure, flow rates—"involved in a process comprising the catalytic chemical conversion of hydrocarbons." 437 U.S. at 596. Specifically, the recited steps involved (1) "[d]etermining the present value" of the relevant "process variable"; (2) "[d]etermining a new alarm base" using a specified mathematical formula; (3) "[d]etermining an updated alarm limit"; and (4) "[a]djusting said alarm limit to said updated alarm limit value." *Id.* at 597. The Court held that the claim was not patent eligible because it sought to cover an abstract idea—the mathematical formula utilized in the claim—rather than a sufficiently particularized application of that idea. While the claim nominally purported to cover *use* of the equation in connection with a catalytic converter, it did not recite a sufficiently specific method for *how* to do so. The claim did not, for example, "explain how to select the appropriate margin of safety, the weighting factor, or any of the variables. Nor d[id] it

purport to contain any disclosure relating to the chemical processes at work, the monitoring of process variables, or the means of setting off an alarm or adjusting an alarm system." *Id.* at 586. To the contrary, "all that it provide[d] is a formula for computing an updated alarm limit," *ibid.*, which is itself simply "a number," *id.* at 585. Thus, the Court held, the "other steps in the process did not limit the claim to a particular application" of the unpatentable mathematical formula. *Mayo*, 132 S. Ct. at 1299 (discussing *Flook*).

The claim at issue in *Diehr* was for a "method of operating a rubber-molding press for precision molded compounds with the aid of a digital computer." 450 U.S. at 179 n.5. Like *Flook*, the claim involved use of a mathematical formula, the "Arrhenius equation." *Ibid.* The claim recited the following steps: (1) "providing said computer with a data base for said press including at least, natural logarithm conversion data (ln), the activation energy constant (C) unique to each batch of said compound being molded, and a constant (x) dependent upon the geometry of the particular mold in the press"; (2) "initiating an interval timer in said computer upon the closure of the press for monitoring the elapsed time of said closure"; (3) "constantly determining the temperature (Z)"; (4) "repetitively calculating in the computer, at frequent intervals during each cure, the Arrhenius equation for reaction time during the cure, which is ln v=CZ+x where v is the total required cure time"; (5) "repetitively comparing in the computer at said frequent intervals during the cure each said calculation of the total required cure time calculated with the Arrhenius equation and said elapsed time"; and (6) "opening the press automatically when a said comparison indicates equivalence." *Ibid.*

Contrasting the claim at issue with that in *Flook*, the Court found that it did "not seek to patent a mathematical formula" or "pre-empt the use of [the Arrhenius] equation." *Diehr*, 450 U.S. at 187. Rather, it sought "patent protection for a process of curing synthetic rubber," *ibid.*, and the "claims describe in detail a step-by-step method for accomplishing such" a process, *id.* at 184. The Court held that "Arrhenius' equation is not patentable in isolation, but when a process for curing rubber is devised which incorporates in it a more efficient solution of the equation, that process is at the very least not barred at the threshold by §101." *Id.* at 188. "Industrial processes such as this are the types which have historically been eligible to receive the protection of our patent laws." *Id.* at 184.

The different outcomes in *Flook* and *Diehr* thus boiled down to whether the claims recited a practical application of the abstract concept with sufficient specificity. If the claim in *Flook* had not merely recited the mathematical equation for adjusting an alarm limit—which yielded only "a number," 437 U.S. at 585—but instead also recited specific steps for a practical application of the equation in connection with catalytic conversion, the Court may well have found the claim patent eligible. Conversely, if the claim in *Diehr* had generically recited the concept of using the Arrhenius equation to determine how long to let rubber cure in a mold, without detailing the specific steps of *how* to effectuate that through a particular process, that claim may have been found ineligible.

Morse, *Flook*, and *Diehr* thus provide courts with guidance in deciding the patent eligibility of computer-implemented processes. By focusing on whether the claims recite a sufficiently specific application of any abstract ideas on which they depend, courts can ensure

that the patentee's right to exclude "is commensurate with the invention's practical, real-world contribution, rather than asserting coverage over general ideas unmoored to a specific application." Mark A. Lemley *et al.*, *Life After* Bilski, 63 Stan. L. Rev. 1315, 1317 (2011). There may be harder cases that involve how to apply those principles. But this case is not one of them.

CONCLUSION

The judgment of the Federal Circuit should be affirmed.

Respectfully submitted.

HORACIO E. GUTIÉRREZ
JULE L. SIGALL
DAVID W. JONES
MICROSOFT CORPORATION
One Microsoft Way
Redmond, WA 98052
(425) 882-8080

Counsel for Microsoft Corporation

JEFFREY A. LAMKEN
Counsel of Record
MICHAEL G. PATTILLO, JR.
MOLOLAMKEN LLP
The Watergate, Suite 660
600 New Hampshire Ave., NW
Washington, D.C. 20037
(202) 556-2000
jlamken@mololamken.com

ANTHONY BACA
CYNTHIA BRIGHT
CURTIS ROSE
ROBERT WASSON
HEWLETT-PACKARD COMPANY
OFFICE OF THE GENERAL COUNSEL
3000 Hanover Street
Palo Alto, CA 94304
(650) 857-1501

Counsel for Hewlett-Packard Company

JOHN M. WHEALAN
4613 Merivale Road
Chevy Chase, MD 20815
(202) 994-2195

Counsel for Amici Curiae

DANA S. RAO
ADOBE SYSTEMS INC.
345 Park Avenue
San Jose, CA 95110
(408) 536-3298

Counsel for Adobe Systems Inc.

FEBRUARY 2014