March 1995

The Soul of the Virtual Machine: In re Alappat

W. Wayt King Jr.

Follow this and additional works at: https://digitalcommons.law.uga.edu/jipl

Part of the Intellectual Property Law Commons

Recommended Citation

Available at: https://digitalcommons.law.uga.edu/jipl/vol2/iss2/6
THE SOUL OF THE VIRTUAL MACHINE: IN RE ALAPPAT

I. INTRODUCTION

I am not an advocate for frequent changes in laws and constitutions. But laws and institutions must go hand and hand with the progress of the human mind. As that becomes more developed, more enlightened, as new discoveries are made, new truths discovered and manners and opinions change, with the change of circumstances, institutions must advance also to keep pace with the times.

Thomas Jefferson

The Patent Clause of the United States Constitution has witnessed profound changes in our society and in the "useful arts" that society has relied upon for improvements in the quality of life. The Patent Clause, although born in an 18th century world with an economy based largely on agriculture, contained enough flexibility to accommodate the mechanization of production that characterized the Industrial Revolution. As modern computer technology of the late 20th century bends our conception of "the useful arts," the Patent Clause must now accommodate the next wave of economic development: the Information Age.

Instant access to virtually limitless raw information is at the heart of today's service economy. Economic value resides increasingly in the creation, distribution, interpretation and transforma-

---


2 U.S. CONST. art. I, § 8, cl. 8 states: "The Congress shall have Power To . . . promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."

3 See ALVIN TOFFLER, THE THIRD WAVE (1980) (predicting gradual development of Information Age to supplant Industrial Age of last two centuries).
tion of information. This information may take many forms, including text, sound, images, and video. Representation of information by ones and zeros, the binary language of the computer, permits the development of standardized formats for storage, distribution, manipulation, and display. This standardization lowers costs so that the ideal of universal access to information by all members of society is within reach, spawning the digitally-driven Information Age.

One of the key tools of the Information Age is computer applications software used to manipulate information and perform other useful tasks. Applications software can take a number of forms, including computer operating instructions residing permanently in the read-only memory (ROM) of a personal computer, application-specific integrated circuitry (ASIC) installed in special-purpose test equipment, or the familiar word processor program loaded temporarily into the random-access memory (RAM) of a personal computer.

Applications software tools do not fit comfortably into our more familiar paradigms of work, tools, and “useful arts.” We can neither pick them up nor see them, and their propensity to take many forms gives them a disturbingly ephemeral quality. But tools they are, no less so than the cotton gin, the telegraph, or the intermittent windshield wiper.

Courts and the U.S. Patent and Trademark Office (PTO) have had a difficult and confusing time dealing with patents for computer software, in spite of, or perhaps because of, a doctrinal formalism purporting to provide clear rules. While the Patent Act

4 During consideration of software patent applications in the PTO, the mathematical algorithms or quantitative operations underlying the software are scrutinized to ensure that a patent is not granted for laws of nature, natural phenomena or abstract ideas in mathematical form. Diamond v. Diehr, 450 U.S. 175, 185, 209 U.S.P.Q. (BNA) 1 (1981). For example, the mathematical formula relating the circumference of a circle to its diameter \( C = \pi \times d \) clearly represents unpatentable subject matter. The so-called Freeman-Walter two-part test for such algorithms was first enunciated in In re Freeman, 573 F.2d 1237, 197 U.S.P.Q. (BNA) 464 (C.C.P.A. 1978) and later modified in In re Walter, 618 F.2d 758, 205 U.S.P.Q. (BNA) 397 (C.C.P.A. 1980). The test is as follows:

First, the claim is analyzed to determine whether a mathematical algorithm is directly or indirectly recited. Next, if a mathematical algorithm is found, the claim as a whole is further analyzed to determine whether the algorithm is “applied in any manner to physical elements or process steps,” and, if it is, it “passes muster under § 101.”
uses broad language to describe what sorts of inventions are patentable, courts have excluded abstract ideas, natural phenomena and laws of nature from patentability. Some commentators have argued that the nature of computer software is such that neither patent law nor copyright law provides appropriate protection, and that special software protection legislation should be adopted instead. The software development community, the patent bar, and the PTO have been without clear judicial guidance on this question of software patentability. This Note will examine In re Alappat, a recent case decided by the Court of Appeals for the Federal Circuit (CAFC), sitting en banc, which clears away some of the doctrinal haze obscuring the statutory requirements for the patentability of applications software.

II. BACKGROUND

In In re Alappat, the Court of Appeals for the Federal Circuit determined that a mathematical algorithm used to enhance electrical signals and thereby improve their display on a digital storage oscilloscope constituted patentable subject matter, even though that algorithm could easily be implemented via computer software. The decision was significant because the court expressly

---


5 "Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title." 35 U.S.C. § 101 (1988).


8 In re Alappat, 33 F.3d 1526, 31 U.S.P.Q.2d (BNA) 1545 (Fed. Cir. 1994) (en banc).

9 Id. at 1545.
recognized that computer applications software by itself is within the subject matter realm of 35 U.S.C. § 101. The decision requires that the PTO revise its treatment of the patentability of computer software.

Kuriappan P. Alappat and his fellow engineers claimed to have invented a solution to a problem endemic to the digital storage oscilloscope, an electronic test instrument used by engineers and technicians to measure and display electrical signals. Without the use of Alappat’s invention, the signal waveforms displayed on a digital oscilloscope screen often exhibit jagged edges, known as aliasing, when the oscilloscope is measuring and displaying signals with rapid rise or fall times. This aliasing is a result of the limited resolution of the oscilloscope’s digital sampling scheme. Alappat developed a technique to smooth out these jagged edges by illuminating display pixels (spots of light on the display screen) near the waveform being displayed with intensities proportional to the distance from the center of each pixel to the desired smooth display trace. This new “rasterizer” used a mathematical algorithm, or sequence of calculation steps, to calculate the intensity for the nearby pixels. While Alappat’s claims recited the use of electronic hardware to implement his

10 Id. (“[A] computer operating pursuant to software may represent patentable subject matter, provided, of course, that the claimed subject matter meets all of the other requirements of Title 35.”).
13 Alappat, Edward E. Averill, and James G. Larsen were employees of Tektronix, Inc. of Wilsonville, Oregon, a manufacturer of electronic test equipment and computer-related equipment. Their patent application was designated in the PTO as application Serial No. 07/149,792. In re Alappat, 33 F.3d 1526, 1530 (Fed. Cir. 1994).
14 Id. at 1537.
15 Id. at 1537-38.
16 Id. A raster is a particular array of pixels in a digital oscilloscope display, which consists of pixels arranged in rows and columns. A time-varying electrical signal is represented on a simple digital oscilloscope by turning on one pixel in each column. Each successive column represents an incremental increase in time, and each row represents measured signal magnitude. A “rasterizer” effects a transformation of this simple measured signal map into a new, more complex map. Id. at 1537.
17 Id. at 1538.
algorithm,\textsuperscript{18} it could have easily been implemented in computer software or firmware.\textsuperscript{19}

Alappat's claims\textsuperscript{20} incorporating the rasterizer algorithm were rejected under 35 U.S.C. § 101 by the PTO Examiner as being directed to non-statutory subject matter.\textsuperscript{21} Alappat appealed to the Board of Patent Appeals and Interferences (the Board),\textsuperscript{22} and a three-member panel reversed the Examiner's claims rejection.\textsuperscript{23} The Examiner requested reconsideration of that decision, and an expanded eight-member Board affirmed the Examiner's original rejection on the prior subject matter grounds.\textsuperscript{24}

Alappat appealed the expanded Board's rejection to the Court of Appeals for the Federal Circuit (CAFC). Before reaching the merits of the dispute, the en banc CAFC raised sua sponte\textsuperscript{25} a question of the validity of the Patent Commissioner's empaneling of an expanded Board to reconsider the three-member panel's decision. The CAFC concluded that the expanded Board's decision was proper and that the court therefore had jurisdiction to consider the

---

\textsuperscript{18} In re Alappat, 33 F.3d 1526, 1539 (Fed. Cir. 1994).
\textsuperscript{19} Id. at 1527.
\textsuperscript{20} Claim 15 of Alappat's patent application read: A rasterizer for converting vector list data representing sample magnitudes of an input waveform into anti-aliased pixel illumination intensity data to be displayed on a display means comprising:
(a) means for determining the vertical distance between the endpoints of each of the vectors in the data list;
(b) means for determining the elevation of a row of pixels that is spanned by the vector;
(c) means for normalizing the vertical distance and elevation; and
(d) means for outputting illumination intensity data as a predetermined function of the normalized vertical distance and elevation.

Id. at 1538-39.
\textsuperscript{21} Id. at 1531.
\textsuperscript{22} See Simson L. Garfinkel, Patently Absurd, 2.07 WIRED, July 1994, at 104 (lamenting lack of access to prior art within PTO); Elinor Mills, Annulling Compton's Patent Helps PTO Earn Industry's Trust, INFOWORLD, Nov. 14, 1994, at 38 (observing PTO's recent attempts to build prior art database); Briefs, COMPUTERWORLD, Aug. 22, 1994, at 67 (noting addition of nine computer scientists to PTO staff).
\textsuperscript{23} In re Alappat, 33 F.3d 1526, 1531 (Fed. Cir. 1994).
\textsuperscript{24} Ex parte Alappat, 23 U.S.P.Q.2d (BNA) 1340 (B.P.A.I. 1992). The expanded Board characterized its reversal as a "modification" of the original three-member Board's decision, but stated that its decision was a "new decision" for the purposes of reconsideration or court review. In re Alappat, 33 F.3d 1526, 1531 (Fed. Cir. 1994).
\textsuperscript{25} See Coastal Corp. v. United States, 713 F.2d 728 (Fed. Cir. 1983) (holding that jurisdiction cannot be conferred on CAFC by implied or express waiver).
appeal.\(^{26}\)

On the merits of Alappat's patent application, the court, by a six to two margin,\(^{27}\) reversed the expanded Board's rejection and held that the use of a mathematical algorithm in software does not bar a machine from patentability on Section 101 subject matter grounds. The court noted, however, that the claimed invention as a whole must not merely be an abstract mathematical concept, but must have practical application.\(^{28}\)

III. THE FEDERAL CIRCUIT'S ANALYSIS OF THE MERITS

A. JUDGE RICH'S MAJORITY OPINION

The court was presented with two issues regarding the merits of the case:

A) whether 35 U.S.C. Section 112 Paragraph 6 requires that a claim for an algorithm be construed as including the structure described in the specification portion of the application;\(^{29}\) and
B) whether Alappat's algorithm falls within the judicially-created "mathematical algorithm" exception to the subject matter provisions of Section 101.\(^{30}\)

Judge Rich wrote the court's opinion and was joined by or explicitly concurred with by five of the other eleven judges, with

---

\(^{26}\) In re Alappat, 33 F.3d 1526, 1530-36 (Fed. Cir. 1994). The issue of the authority of the PTO to empanel an expanded Board for reconsideration was significant, and deserving of substantial comment, but is beyond the scope of this Note. Id. at 1530.

\(^{27}\) Judges Newman, Lourie, Plager and Rader joined on the merits in Judge Rich's majority opinion. Chief Judge Archer and Judge Nies dissented on the merits, while Judges Mayer, Clevenger and Schall would hold that the Court lacked jurisdiction and therefore provided no opinion on the merits. Id. at 1530.

\(^{28}\) Id. at 1544.

\(^{29}\) Id. at 1540. 35 U.S.C. § 112 (1988) states:
An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

\(^{30}\) See infra note 35.
three of the judges taking no position. The majority quickly disposed of the § 112 issue, holding that the plain meaning and prior CAFC interpretation of the statute rendered Alappat’s algorithm to be directed toward a machine made up of a combination of electronic circuits disclosed in the patent application specification. The majority said that the expanded Board had “erred as a matter of law in refusing to apply [Sec.] 112, [Para.] 6

After determining that the claim was indeed for a “machine,” the majority addressed the second issue of whether the invention nonetheless fell within the judicially-created “mathematical algorithm” exception to the § 101 litany of patentable subject matter. Significantly, Judge Rich abandoned the two-part Free- man-Walter-Abele test developed during the late 1970s and early 1980s for algorithm patentability.

Instead of the formalism of the two-part test, the majority read Supreme Court precedent to require that, for subject matter analysis, mathematics-based claims must be considered within the context of the invention as a whole. Judge Rich concluded that “the dispositive inquiry is whether the the claim as a whole is directed to statutory subject matter,” and not “whether a claim

---

31 See supra note 27.
32 Judge Archer’s dissent described this issue as a “red herring.” In re Alappat, 33 F.3d 1526, 1561 (Fed. Cir. 1994).
33 Id. at 1541. The specification of Alappat’s invention provided in the patent application disclosed a hardware implementation of the rasterizer using several common discrete digital circuit devices, including shift registers, arithmetic logic units, barrel shifters, counters and read-only memory (ROM). Id. at 1558-59.
34 Id. at 1540.
35 Abstract mathematical algorithms standing alone have been considered by courts to be unpatentable subject matter because they are considered fundamental properties of nature not created by mankind. Gottschalk v. Benson, 409 U.S. 63, 71 (1972) (holding unpatentable algorithm to convert binary coded decimal numbers into decimal numbers without recital of machine structure); Parker v. Flook, 437 U.S. 584, 589-90 (1978) (holding algorithm to update chemical process alarm limits to be unpatentable despite “post-solution” physical activity); but cf. Diamond v. Diehr, 450 U.S. 175, 192-93 (1981) (finding mathematical algorithm to be patentable when included as part of physical process for curing rubber).
36 See supra note 4.
37 In re Alappat, 33 F.3d 1526, 1543 (Fed. Cir. 1994) (quoting Diamond v. Diehr, 450 U.S. 175, 192 (1981) (“When a claim containing a mathematical formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect . . . , then the claim satisfies the requirements of Sec. 101.”)).
J. INTELL. PROP. L.

contains . . . any mathematical subject matter which standing alone would not be entitled to patent protection."38 The majority reasoned that Alappat's invention as a whole was clearly directed to a machine and accordingly held that the algorithm claim was patentable.39 Moreover, the majority looked to the text of § 101 itself to assert that use of the term "any" in the statute was evidence of Congress' intent that the statute be interpreted broadly, and that a plain meaning of the statute supported the same inference.40

The majority recognized that the Supreme Court had previously carved out three exceptions for subject matter for which patent protection is not available: "laws of nature, natural phenomena, and abstract ideas."41 But the majority limited the exceptions established by prior cases to the rule that mathematics itself is not patentable until it is practically applied.42

In what could be characterized as dicta, the majority expanded its holding to the broader issue of the patentability of computer software. Pointing to case law developed by the Federal Circuit's predecessor court,43 Judge Rich urged that a computer programmed by software to perform particular functions "creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed . . . ."44 Judge Rich closed with the assertion that "[i]n any case, a computer, like [Alappat's] rasterizer, is apparatus not mathematics."45 Although the preamble to Alappat's patent application described discrete electronic hardware circuitry for implementing the algorithm, the majority would hold that the algorithm claims write on a general-purpose computer as well.46

38 In re Alappat, 33 F.3d 1526, 1543 (Fed. Cir. 1994).
39 Id. at 1544-45.
40 Id. at 1542. The court felt that no limitations should be read into 35 U.S.C § 101 unless the legislative history clearly shows the intention of such limitations. Id.
41 Id. at 1542 (quoting Diamond v. Diehr, 450 U.S. 175, 185 (1981)).
42 Id. at 1543.
43 In re Alappat, 33 F.3d 1526, 1545 (Fed. Cir. 1994) (citing In re Freeman, 573 F.2d 1237 (C.C.P.A. 1978); In re Noll, 545 F.2d 141, 191 U.S.P.Q. (BNA) 721 (C.C.P.A. 1976); In re Prater, 415 F.2d 1393, 162 U.S.P.Q. (BNA) 541 (C.C.P.A. 1969)).
44 In re Alappat, 33 F.3d 1526, 1545 (Fed. Cir. 1994).
45 Id.
46 Id.
B. CHIEF JUDGE ARCHER’S DISSENTING OPINION

Judge Archer disagreed with the majority’s application of case precedent and statutory interpretation, and would have held Alappat’s rasterizer to be an unpatentable mathematical equation. First, he pointed to historical treatment of the subject matter inquiry, noting that statutory subject matter has traditionally been found only in new and useful effects or results grounded in the earth’s substances. Second, he urged that terms of art such as “process,” “machine” and “invent” in § 101 must be construed in accordance with judicial precedent and with established meanings, and that novelty considerations should not enter into the subject matter calculus.

The dissent analogized computer software and abstract mathematics to musical creations. Chief Judge Archer compared a song, which may be embodied in a compact disc, record or cassette tape, to computer software or a mathematical algorithm, which may be implemented in a number of ways. Continuing the analogy, Judge Archer held that the mere recital of hardware structure along with an algorithm claim does not render the claim patentable any more than the recital of a compact disc player along with music renders the music itself patentable.

Judge Archer distinguished Supreme Court precedent, arguing that an algorithm previously held patentable was not claimed independently of hardware, while Alappat’s disputed algorithm claim stood alone and recited only abstract mathematics. Finally, the dissent attacked the majority’s reliance on dated, overruled precedent in the Federal Circuit’s predecessor court.

47 Id. at 1551-52.
48 Id. at 1552 (citing G. CURTIS, A TREATISE ON THE LAW OF PATENTS FOR USEFUL INVENTIONS xxiii-xxv (4th ed. 1873) (providing excerpt from 1873 patent treatise discussing transformations in particles of matter and physical forces which are purportedly at heart of patentable subject matter)).
49 In re Alappat, 33 F.3d 1526, 1553 (Fed. Cir. 1994) (arguing that terms should be read in patent law context, requiring inventions to have practical application).
50 Id. (“Specific conditions for patentability follow” 35 U.S.C. § 101 (quoting Diamond v. Diehr, 450 U.S. 175, 189 (1981))).
51 Id. at 1554.
52 Id.
53 Id. at 1557 (citing Diamond v. Diehr, 450 U.S. 175, 192 n.14 (1981)).
Judge Archer contended that such precedent has led to a surfeit of claim structures where drafting artfulness is exalted over claim substance.\(^{54}\)

The dissent predicted that the majority's broader holding regarding the patentability of software for general-purpose computers would have "untold consequences,"\(^{55}\) warning of stifling mathematical patents and the "technological exclusivity" they would engender.\(^{56}\)

C. THE CONCURRING OPINIONS

Judge Newman provided a strong, policy-centered concurring opinion, arguing that patent law has always evolved and must continue to evolve in parallel with technological change.\(^{57}\) She looked to mathematics as merely one of many basic tools of technology, not as "a monster to be struck down,"\(^{58}\) and urged a broad interpretation of the subject matter scope of § 101. She argued that such an interpretation is consistent with a public policy directed toward the fostering of industrial innovation.\(^{59}\) She concurred that Alappat's algorithm invention was patentable subject matter, whether implemented as hardware in discrete digital circuitry or in a custom semiconductor chip, or implemented as software in the permanent read-only memory (ROM) or temporary random-access memory (RAM) of a general-purpose computer.\(^{60}\)

\(^{54}\) In re Alappat, 33 F.3d 1526, 1561 (Fed. Cir. 1994).

\(^{55}\) Id. at 1552.

\(^{56}\) Id. at 1568.

\(^{57}\) Id. at 1568-71.

\(^{58}\) Id. at 1570.

\(^{59}\) In re Alappat, 33 F.3d 1526, 1570-71 (Fed. Cir. 1994).

\(^{60}\) Id. at 1569-70. Embedded in a digital oscilloscope, Alappat's algorithm would most likely be implemented either in a circuit of discrete digital logic chips or in a custom chip such as an application-specific integrated circuit (ASIC). But the algorithm could easily be implemented as software loaded temporarily into the RAM of a general purpose computer or as software loaded more permanently into the ROM of a dedicated computer or controller. See also Brief for Appellant at 47-48, In re Alappat, 33 F.3d 1526 (Fed. Cir. 1994) ("In today's technological environment virtually every machine . . . uses digital processing, either with specific digital circuitry and/or a microprocessor executing a program . . . If the process is new, useful and nonobvious, does it really matter whether the implementation is in the form of analog components, digital components, programs for a computer, or a combination thereof? Isn't such a differentiation actually exalting form over substance? A programmed digital computer becomes a special purpose digital computer to perform the function specified
Judge Newman summarized her arguments in a plea demanding that the fruits of the Information Age be accorded the monopoly protections and disclosure obligations of the patent law system, arguing that the incentives provided by the patent system are necessary to maximize development of new technologies. She concluded by noting with approval that the CAFC had undertaken to "remove the cloud on whether [the inventions of the Information Age] may participate in the benefits and obligations of the patent system."

Judge Rader authored a separate concurring opinion to stress his view that Alappat's algorithm constituted patentable subject matter, whether claimed in "machine" language or in "process" language. He interpreted Supreme Court precedent to permit inventions to be expressed in a wide range of forms, including mathematical algorithms and symbols, and decried the PTO's restrictive application of § 101 to algorithm claims.

IV. EXISTING LAW

The Supreme Court first considered application of 35 U.S.C. § 101 subject matter requirements to the patentability of computer software in a 1972 case concerning claims for converting binary-coded numbers into pure binary form. Because the claims were for a mathematical "algorithm" alone without any physical...
embodiment, the Court held that the claims were not patentable under § 101. Some commentators viewed the decision as precluding all software from patent protection.

In 1978, the Supreme Court considered claims directed toward a computerized monitor of a chemical process which automatically updated alarm limits for the process. Even though the claim included steps beyond the mere solution of the mathematical formula (steps to update the alarm limit), the Court followed precedent and held such “post-solution activity” insufficient to permit patenting a claim which included an algorithm that the Court considered to be no more than a scientific truth.

In 1981, the Supreme Court in Diamond v. Diehr considered an algorithm which incorporated the well-established Arrhenius equation into a computerized process for curing rubber. The Court held that a claim relying on the equation was patentable subject matter if used in a “useful” process. According to one interpretation, the Court limited Benson and Flook to the proposi-
tion that only narrowly-drawn laws of nature, abstract ideas and natural phenomena are excluded from patent protection, and that a mathematical algorithm implemented in computer software was not necessarily excluded from protection. On this view, the Supreme Court in Diehr addressed both the PTO's and lower courts' previous exclusion of algorithms from patent protection, in the absence of any statutory foundation for such exclusion, when it commanded that "courts should not read into the patent laws limitations and conditions which the legislature has not expressed." Instead, the Court looked to the legislative history and purpose of the statute to conclude that such unexpressed limitations were improper. Some commentators and lower courts have suggested that Diehr truly superseded the prior Supreme Court algorithm case law and was not merely an extension of those prior cases.

During the development of the Supreme Court's subject matter jurisprudence, the PTO and the lower courts gradually developed a two-part test, first enunciated in 1978, to determine the patentability of mathematical algorithms. The first step of the test was a straightforward inquiry into the existence of a mathemat-

---


54 Arrhythmia Research Technology, Inc. v. Corazonix Corp., 958 F.2d 1053, 1057 n.4 (Fed. Cir. 1992); cf. Jur Strobos, Stalking the Elusive Patentable Software: Are There Still Diehr or Was it Just a Flook?, 6 HARV. J.L. & TECH. 363, 364 (1993) ("In retrospect, the Supreme Court cases say far less about software patentability in general, and more about the specific claims confronted. . . . The opinions of the Supreme Court are consistent with a much broader view of software subject matter patentability than has been implemented by the Federal Circuit."); but see Pamela Samuelson, Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions, 39 EMORY L.J. 1025, 1028-29 (1990) (claiming that Diehr "reaffirmed the Benson ruling on the unpatentability of algorithms").


56 This test was predicated on the interpretation that Diamond v. Diehr merely extended the limitations on algorithm patentability set out in Gottschalk and Flook. See supra note 4.
matical algorithm within the claim.\textsuperscript{87} If an algorithm was found, the second step would render an algorithm statutory if it was used to "define structural relationships between [the claim's] physical elements."\textsuperscript{88} This second step was relaxed in 1982 so that if an algorithm-incorporating claim, considered without the algorithm, was statutory, then the claim including the algorithm was also statutory.\textsuperscript{89}

Development of subject-matter doctrine was hindered by confusion over the meaning of the term "algorithm," and patent applicants learned to avoid naked claims to computer software regardless of the software's utility.\textsuperscript{90} The Federal Circuit avoided the restrictions of the two-step test in 1989,\textsuperscript{91} when it held that § 112 para. 6\textsuperscript{92} operated to import external structural elements to combine with an algorithm, rendering the algorithm itself patentable subject matter.

In 1992, the Federal Circuit considered an appeal\textsuperscript{93} involving a technique to analyze electrocardiogram signals to detect latent heart defects. The court held that the analysis algorithm, when considered as combined with the physical structure of the signal analyzer, did not operate on merely abstract signals and hence constituted statutory subject matter.\textsuperscript{94} Judge Rader's concurring opinion\textsuperscript{95} suggested that \textit{Diehr} required a more direct reliance on the patent statute and urged the abandonment of the judicially-created \textit{Freeman-Walter-Abele} two-step test.\textsuperscript{96}

\textsuperscript{87} \textit{In re Freeman}, 573 F.2d 1237, 1245 (C.C.P.A. 1978).
\textsuperscript{88} \textit{In re Walter}, 618 F.2d 758, 767, 205 U.S.P.Q. (BNA) 397 (C.C.P.A. 1980).
\textsuperscript{89} \textit{In re Abele}, 684 F.2d 902, 907, 214 U.S.P.Q. (BNA) 682 (C.C.P.A. 1982).
\textsuperscript{90} Arrhythmia Research Technology, Inc. v. Corazonix Corp., 958 F.2d 1053, 1062-64 (Fed. Cir. 1992).
\textsuperscript{91} \textit{In re Grams}, 888 F.2d 835 (Fed. Cir. 1989).
\textsuperscript{92} \textit{In re Iwahashi}, 888 F.2d 1370 (Fed. Cir. 1989).
\textsuperscript{93} 35 U.S.C § 112 para. 6 (1988) states that claims presented in means-plus-function terms "shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof." \textit{Id.} at 1375.
\textsuperscript{94} Arrhythmia Research Technology, Inc. v. Corazonix Corp., 958 F.2d 1053 (Fed. Cir. 1992).
\textsuperscript{95} \textit{Id.} at 1059-61.
\textsuperscript{96} \textit{Id.} at 1061-66 (Rader, J., concurring). Note that Judge Rader expressed similar sentiments in his concurring opinion in \textit{Alappat}. See supra note 63 and accompanying text.
\textsuperscript{97} \textit{Id.} at 1064-65.
V. ANALYSIS OF IN RE ALAPPAT

In Alappat, a majority of the en banc CAFC held that an algorithm easily amenable to computer software implementation may nonetheless constitute patentable subject matter within the meaning of 35 U.S.C § 101. This holding firmly establishes that computer applications software is patentable and should eliminate much of the confusion and uncertainty surrounding the issue among the legal and software development communities. The holding is consonant with the policy considerations underlying the patent statute, and it reflects congressional intent. While some argue that patent protection will shackle, rather than promote, further software development,9 the court correctly required that patent law embrace the Information Age by recognizing that applications software installed into the memory of a general-purpose computer creates a virtual machine.

The majority's conclusion is correct for several reasons. First, the plain meaning of Section 101 states that "any . . . process, machine, manufacture, or composition of matter" is eligible for protection.99 Use of the modifier "any" implies that a broad interpretation is to be granted.100 Judge-made exceptions to the statute, for laws of nature, natural phenomena, and abstract ideas, must be given a narrow effect in light of the statute's expansive language.101 Further, the legislative history supports an inclusive interpretation of the statute, as Congress intended that § 101 subject matter include "anything under the sun that is made by man."102

The most powerful argument for an inclusive interpretation of

---

9 Garfinkel, supra note 22. Some commentators argue that software development thrives in an open, sharing environment and that monopoly rights provided by patent law will actually hinder future software development. See James W. Morando & Christian H. Nadan, Do Software Patents 'Stac' the Deck Against the Competition?, 11 COMPUTER LAW 1 (April 1994) (noting that software development has grown rapidly over past two decades without significant patent protection, and warning that many small software developers may be forced out of business by threat or reality of patent litigation).


§ 101, however, is grounded in policy. Other than to change the word "art" to "process," the text of § 101 has served to protect the nation's technical community unchanged since 1793.\footnote{In re Alappat, 33 F.3d 1526, 1570 (Fed. Cir. 1994).} Since that time our once agrarian society has been transformed by the technologies of the Industrial Revolution, and the patent law has evolved to serve corresponding new needs.\footnote{Id. at 1570 (quoting Barr v. United States, 324 U.S. 83, 90 (1945) ("If Congress has made a choice of language which fairly brings a given situation within a statute, it is unimportant that the particular application may not have been contemplated by the legislators.").} The dissent in \textit{Alappat} looked to the long history of the application of the patent laws and saw as useful only those inventions based on the industrial age technologies of mechanics and materials.\footnote{Id. at 1551-57 (Archer, C.J., dissenting).} The dissent failed to grasp that modern electronics and computer technology are not only useful, but will increasingly dominate our economy.\footnote{Green Paper, supra note 1. The National Information Infrastructure (NII) is the Clinton Administration's initiative to facilitate the development of a nationwide broadband two-way communications network, popularly referred to as the information superhighway. Computer and communications technologies will continue to converge as the information superhighway is developed; many believe that new information and entertainment services enabled by these new technologies will propel economic growth forward as we enter the new millennium. \textit{See}, e.g., David Kline, \textit{Align and Conquer}, WIRED 2.07, Feb. 1995, at 110 (describing Bell Atlantic's vision of future broadband networks and services); cf. Carole E. Handler & Alex W. Farr, \textit{The Information Superhighway in 1995: Proceeding with Caution}, COMMS. LAW., Winter 1995, at 9 (discussing regulatory impediments to convergence among telecommunications and cable TV companies necessary to build information superhighway infrastructure).} As our nation leads the world into the Information Age via the digital revolution, the law must continue to evolve as well.

The protection of useful innovations is key to our modern technology-driven economy. Such protection is at the heart of the economic incentive that underpins free-market theory and provides motivation to develop the technology from which society benefits. In the words of Judge Newman's concurrence in \textit{Alappat}, "[o]ne must have a powerful reason to exclude technology from the scope of Title 35."\footnote{In re Alappat, 33 F.3d 1526, 1571 (Fed. Cir. 1994) (Newman, J., concurring).} To exclude the virtual machine created by loading applications software into general-purpose computers would be to deny that such machines are at the heart of much of modern work.

The distinction between statutory applications software and non-

\footnote{\textit{In re Alappat}, 33 F.3d 1526, 1570 (Fed. Cir. 1994).}
statutory (but copyrightable) content software is important. The former includes software elements of word processors, database managers, spreadsheets, and many other software tools used to transform information in the modern world. Alappat's rasterizer algorithm, which can be implemented in software, is but another tool, transforming a jagged oscilloscope display trace into a smoother trace. In contrast, content software includes music, literature, documents, images, video and other information, all of which may be stored and used in digital form by a computer, but none of which performs any useful work alone. Indeed, applications software may be applied to perform work on content software, changing it to create new information or to perform some other useful function. The dissent argued that Alappat's algorithm, and by implication all computer software, is analogous to music since it can be recorded in many forms. The dissent thus failed to appreciate the distinction between applications software, which is a tool of the modern world, and content software, which is properly the subject matter of copyright law.

The dissent also argued that mathematical algorithms cannot be rendered statutory merely by an association with digital hardware in a patent claim. However, mathematics includes not only fundamental rules, but also practical applications based on those rules. Mathematics is another tool in the arsenal of the inventor, just as chemistry is such a tool. Just because complex digital electronics circuits, and the software which reconfigures them in the virtual machine, can be represented wholly by mathematical equations should be no bar to patentability.

The majority opinion's choice of precedent in In re Alappat only tenuously supported its holding. Instead, the court could

106 Id. at 1553-54 (Archer, C.J., dissenting).
107 Id. at 1557 (Archer, C.J., dissenting).
108 Id. at 1569 (Newman, J., concurring).
109 Id.
111 Id. at 1545 (citing early cases listed supra note 43).
112 Not surprisingly, the dissent attacks the majority's reliance on the outdated C.C.P.A. precedent, the most recent of which is a 1978 decision. Id. at 1561-62 (Archer, C.J., dissenting) (arguing that Diehr overruled previous C.C.P.A. cases).
have read Diehr more broadly, explicitly abandoning the two-part test doctrine of Freeman-Walter-Abele and eliminating the overly formalistic requirement that an algorithm-based claim must be statutory when considered excluding the algorithm. 115

Since Congress has made no mention of per se exceptions for the patentability of algorithms, prior decisions concerning algorithms should be confined to the most narrow definition of that term consistent with facts of those cases. 116 Diehr requires that subject matter inquiries focus on the terms of the statute itself, and not on vague judicially created exceptions to the statute. 117 A correct reading of Diehr and 35 U.S.C. § 101 is in harmony with the economic incentive policy rationale supporting the patentability of computer applications software and other modern technologies.

The Alappat decision is likely to result in an acceleration in software patent applications as the software development community 118 gains confidence in the stability of PTO and judicial interpretation of subject matter criteria in the software art. 119 Attempts to protect software through trade secret law should diminish, with the corresponding benefits of full disclosure to the public that the patent law provides. 120 Clear and consistently applied statutory criteria 121 should allay fears from smaller software developers of potentially ruinous software patent litigation.

---

116 See supra note 69.
117 Arrhythmia Research Technology, Inc. v. Corazonix Corp., 958 F.2d 1053, 1064 (Fed. Cir. 1992) ("courts 'should not read into the patent laws limitations and conditions which the legislature has not expressed.' " (quoting Diamond v. Diehr, 450 U.S. 175, 182 (1981))).
118 The software development community is most concerned with novelty and non-obviousness issues, and the PTO's inability to develop and apply a prior art database in the software area to correctly address those issues. Garfinkel, supra note 22.
120 Those benefits include not only the spread of knowledge in the software development community which results from open and full disclosure, but also the improvement in the prior art database available to the PTO. Garfinkel, supra note 22, at 140.
121 Opening up the patent examination process to permit adversarial comment during prosecution would also contribute to consistency in software patent determinations. See Garfinkel, supra note 22, at 142.
The PTO administration must continue to add staff and prior art data for the computer software group in order to keep up with the strong growth promised by software development and software patent activity.

But patent protection typically does not attach until two to three years after a patent application is initially filed with the PTO, and trade secret law is unable to protect the value intrinsic to software in the interim between product release and patent issuance. As a result, additional protection for computer software during this interim period is needed to reward and encourage innovation. As a means to achieve this, commentators have suggested augmenting copyright and patent protection of software via vigorous assertion of a cause of action in tort for misappropriation in order to provide reasonable market lead time to innovative software developers.

122 Leslie Helm, Appeals Court Ruling Opens Door to Further Software Patents, L.A. TIMES, Aug. 10, 1994, at 4; Garfinkel, supra note 22.
123 Garfinkel, supra note 22.
125 Manifesto, supra note 7, at n.134. The behavior of applications software (i.e. its features and functions available to the user) is the primary source of value in software. Since that behavior is readily apparent to potential competitors and can easily be replicated without access to the source code, trade secret law is of no avail in protecting the developer's investment in software creation. Competitors can imitate the software's features and functions (its behavior), without violating present copyright or trade secret law, well before any patent will issue. Id.
126 Id. at 2356 ("Most of the commercially significant innovations in software will be underprotected if patent law adheres to its traditional bounds . . .").
127 Misappropriation is defined as "[t]he unauthorized, improper, or unlawful use of . . . property for [a] purpose other than that for which intended." BLACK'S LAW DICTIONARY 998 (6th ed. 1990). Under this theory, substantial replication of program behavior would be actionable even though no patent has yet been issued. Manifesto, supra note 7, at 2423.
128 Manifesto, supra note 7, at 2423. The incremental nature of software development largely precludes patent protection for most new software because the innovation therein does not rise to the level of "invention" and hence will not meet the novelty and nonobviousness requirements of the patent law. Id. at 2333. Availability of a cause of action in tort for misappropriation would provide the "lead time" necessary to reward and motivate software innovators. Manifesto, supra note 7, at 2423.
VI. CONCLUSION

In holding that computer applications software may be patentable subject matter, the Court of Appeals for the Federal Circuit has correctly interpreted the direction provided by the Supreme Court in *Diamond v. Diehr* and has abandoned the overly formalistic and unwieldy *Freeman-Walter-Abele* test for algorithm patentability. The CAFC has now recognized that the technical tools of the Information Age merit the protections and obligations of the patent laws—the same laws that helped to develop the tools of the Industrial Revolution. The decision should lead to more consistent treatment of software patent applications by both the Patent and Trademark Office and the Federal Circuit, and the resulting stability and predictability should contribute to continued progress within the software development community.

W. Wayt King, Jr.