Software Terminology: How to Describe a Software Invention in a United States Patent Application

Christopher E. Everett*
SOFTWARE TERMINOLOGY: HOW TO DESCRIBE A SOFTWARE INVENTION IN A UNITED STATES PATENT APPLICATION

CHRISTOPHER E. EVERETT

I. INTRODUCTION

Software patent drafting and prosecution is still an emerging field because of the constant changes and developments in the evolution of software patents. Patent practitioners that draft software patents have to be concerned with meeting the patentability requirements of the United States Code along with the restrictions in the case law. In addition, there are other challenges associated with the proper drafting of a software patent application including the natural ambiguity of language, the ever-occurring developments in the software field, and the maximization of the protective lifespan of the invention.

* J.D. Candidate 2006, Nova Southeastern University, Shepard Broad Law Center; B.S. in Computer Science and M.S. in Computer Science, Mississippi State University. The author is a Registered Patent Agent specializing in software patents. The author wishes to thank: his wife, Dr. Christine R. Rutter, for all of her love and support throughout this process; his family, especially his little sister Kaylei Koon, for encouraging him to never stop learning; his colleagues at Nova Law Review, especially Sabrina Tirabassi and Carlos Carmona, for their hard work and dedication during the editing of this article; and the faculty of the Law Center, especially Professor Linda F. Harrison, for their support and guidance.
tion. Accordingly, it is imperative that patent practitioners who draft software patents understand the limitations of the terminology that can be used to describe software.

The importance of drafting patent applications that properly protect the invention along with any variations is the amount of money that can be involved in patent litigation lawsuits. One such lawsuit involved Pitney Bowes suing Hewlett-Packard for allegedly infringing on its patented technology that "uses beams of light to make variable-sized dots which help to smooth the edges of characters, making the letters and numbers less blurry."1 "Pitney Bowes was seeking damages of more than $1 billion," but settled for $400 million in cash.2 Another such lawsuit involves Stac Electronics, who won a lawsuit against Microsoft for patent violations involving data compression patents.3 This lawsuit was settled by Microsoft paying Stac approximately $83 million.4 Both of these settlements are examples of why using the correct terminology in a software patent is paramount because it may mean the difference between winning or losing a patent lawsuit which could involve millions of dollars.

A. Overview

Part I of this article will discuss the different areas of concern for patent practitioners including the different sections of the United States Code that have to be satisfied for the successful prosecution, offensive use, and defense of a software patent. Part II will review the different terminology that can be used to describe software technology. This terminology includes method, apparatus, computer, system, means-plus-function, computer-readable medium, computer program product, and article of manufacture. In addition, Part II provides claim examples of the different terminology along with any court decisions that affect the interruption of the terminology. Part III gives a conclusion of the terminology that can be used to describe a software invention. In addition, Part III will review the terminology along with suggestions on how to draft sample claims.

2. Id.
4. Id.
B. Challenges of Software Patents

One challenge in drafting software patents is that the "ambiguity of claim language necessarily results in uncertainty in the scope of protection." Part of this ambiguity stems from the disagreement among experts about the definitions of software-related terms. An example of a word that has caused ambiguity is algorithm. In *Diamond v. Diehr*, the practitioner defined algorithm to have a general meaning, while the court held algorithm to have a narrower definition. Accordingly, all words have some form of ambiguity, especially when the allowance of the patent is concerned or a lawsuit involving millions of dollars. Thus, it is imperative that patent practitioners have a solid foundation of the software terminology that is used in patent applications and how the United States Patent and Trademark Office (USPTO) and the courts have construed the terminology.

Another challenge in drafting software patents is that "words do not exist to describe" the invention. This is because "[t]hings are not made for the sake of words, but words for things." Thus, there may not be ambiguity in the terminology used in the patent application, but the terminology may be undefined in the field of art and thus needs to be properly defined by the patent practitioner drafting the patent application. This challenge is especially important in software patents because software in itself is an emerging field and first received patent protection less than twenty-five years ago. Accordingly, the terminology is still developing and patent practitioners should pay special heed to the challenge of defining new words.

Another challenge is that patent practitioners must be meticulous when drafting the patent specification and claims to ensure that the claims can be useful during the entire "enforcement duration" of twenty years. Since the software field is in constant flux, software patents that are drafted with only

7. Id.
9. Id. at 186 n.9.
10. See Szepesi, supra note 6, at 177.
12. Id. (quoting Autogiro Co. of Am., 384 F.2d at 397).
13. See *Diamond*, 450 U.S. at 175.
today's implementation of the invention in mind may not protect the owner from future changes in technology. One example is in the field of high performance computing. Twenty years ago, high performance computers were single system machines such as Cray supercomputers. Today, high performance computing is being moved to clusters of machines such as a Beowulf cluster. This type of innovative change would dramatically degrade the protection of software that stated its use was only on Cray supercomputers without giving the option of different types of high performance computing systems. Accordingly, the terminology used in a software patent is important to ensure the maximum effective life of the patent is not diminished because of poor word choices.

Accordingly, understanding the perils and requirements of the United States Code and case law will enable patent practitioners who draft software patents to develop fully the invention into a form that will provide the maximum protection under the law. This understanding will benefit both patent practitioners and software inventors by ensuring that the patent practitioners can draft the patent application in a form that will benefit the inventor and the inventor will have the invention protected to the fullest extent allowed by the law.

C. Meeting the 35 U.S.C. § 101 Requirements

Patent practitioners drafting software patents must first be concerned with the requirements of 35 U.S.C. § 101 because the line between patentable software and non-patentable software is sometimes still in flux. Section 101 provides that "[w]hoever 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." The concern for software is not "which of the four categories of subject matter a claim is directed to—process, machine, manufacture, or composition of matter—but rather on the essential characteristics of the subject matter, in particular, its practical util-

16. Id.
18. Id.
ity." Thus, the invention must have "some type of practical application, i.e., 'a useful, concrete and tangible result.'" \(^{19}\)

For example, a claim directed to a word processing file stored on a disk may satisfy the utility requirement of 35 U.S.C. [§] 101 since the information stored may have some "real world" value. However, the mere fact that the claim may satisfy the utility requirement of 35 U.S.C. [§] 101 does not mean that a useful result is achieved under the practical application requirement. The claimed invention as a whole must produce a "useful, concrete and tangible" result to have a practical application. \(^{21}\)

Accordingly, patent practitioners drafting software patents must draft the application in such a way to ensure that the described invention produces "a useful, concrete, and tangible result." \(^{22}\) This requirement will be further analyzed through the review of the terminology.

D. \(\textit{Pitfalls of 35 U.S.C. § 112}\)

Section 112 of the Patent Act has six paragraphs of which three present pitfalls to the successful allowance and court challenge of a software patent. \(^{23}\)
The first paragraph has three separate and distinct requirements. The first requirement is that the specification must contain a written description of the invention that is "a full and clear disclosure of the invention." The second requirement is that there must be a description on how "to make and use the invention." This "enablement requirement" is required so "that one skilled in the art can make and use the claimed invention ... to ensure that the invention is communicated to the interested public in a meaningful way." The third requirement of the first paragraph is the disclosure of the best mode of the invention. "The best mode requirement is a safeguard against the desire on the part of some people to obtain patent protection without making a full disclosure as required by the statute." The three requirements of the first paragraph of § 112 can all create pitfalls for the patent practitioner unaccustomed to drafting software patent applications especially viewed in light of who is skilled in the software arts.

The second paragraph of § 112 contains two separate requirements to satisfy the statute. The first requirement is that "the claims must set forth the subject matter that applicants regard as their invention." This requirement places the burden on the patent practitioner to ensure that the claimed invention is the invention that the inventor or assignee desires and needs to be protected. The second requirement is that "the claims must particularly point out and distinctly define the metes and bounds of the subject matter that will be protected by the patent grant." Patent practitioners should work to avoid the pitfall of this requirement because failure to fulfill the requirement could lead to claims that are ambiguous and thus do not adequately protect the invention. This requirement is examined by "whether the scope

_id_.

24. MPEP, supra note 21, at 2100-163.
25. _Id._
26. MPEP, supra note 21, at 2100-163; accord § 112.
27. MPEP, supra note 21, at 2100-184.
28. § 112.
29. MPEP, supra note 21, at 2100-200.
30. _Id._ at 2100-203 (citing § 112).
31. _Id._
32. _Id._
of the claim is clear to a hypothetical person possessing the ordinary level of skill in the pertinent art."\textsuperscript{33}

The sixth paragraph of § 112 allows for means-plus-function claims to be used.\textsuperscript{34} The use of the sixth paragraph allows for means-plus-function claims, but the claims must meet a three-part test:

A claim limitation will be interpreted to invoke 35 U.S.C. [§] 112, sixth paragraph, if it meets the following 3-prong analysis:

(A) the claim limitations must use the phrase "means for" or "step for";

(B) the "means for" or "step for" must be modified by functional language; and

(C) the phrase "means for" or "step for" must not be modified by sufficient structure, material or acts for achieving the specified function.\textsuperscript{35}

Beyond meeting the three-part test, the claims must also meet the definiteness requirement in "that the corresponding structure ... of a means (or step)-plus-function limitation must be disclosed in the specification itself in a way that one skilled in the art will understand what structure ... will perform the recited function."\textsuperscript{36} Thus, for a patent practitioner to properly utilize the sixth paragraph of § 112, the three-part test must be satisfied along with the other requirements to avoid rejections.

Another challenge when viewed in light of the requirements of § 112 is the United States Supreme Court's holding in \textit{Festo Corp. v. Shoketsu Kinzoku Kogyo Kabushiki Co.}\textsuperscript{37} In \textit{Festo Corp.}, the Court held that "if a § 112 amendment is necessary and narrows the patent's scope—even if only for the purpose of better description—estoppel may apply."\textsuperscript{38} Thus, patent practitioners should be concerned that any modifications to the claims to satisfy § 112 may limit the scope of the invention. Although the presumption is against the narrowing amendment, the presumption can be rebutted but "[t]he patentee must show that at the time of the amendment one skilled in the art

\textsuperscript{33} \textit{Id.}
\textsuperscript{34} § 112.
\textsuperscript{35} MPEP, \textit{supra} note 21, at 2100-221.
\textsuperscript{36} \textit{Id.} at 2100-224 (citing Atmel Corp. v. Info. Storage Devices, Inc., 198 F.3d 1374, 1381 (Fed. Cir. 1999)).
\textsuperscript{37} 535 U.S. 722 (2002).
\textsuperscript{38} \textit{Id.} at 737.
could not reasonably be expected to have drafted a claim that would have literally encompassed the alleged equivalent. Accordingly, the Supreme Court's holding may create problems for patent applications that are not properly described in the specification and claimed in the patent application. Patent practitioners that draft software patents should take extra precautions to ensure that the requirements of § 112 are met and that no amendments that may narrow the patent's scope are needed.


Patent practitioners also have to draft to avoid 35 U.S.C. § 102 and 35 U.S.C. § 103 rejection. Software patent applications have been found to

39. Id. at 741.
40. 35 U.S.C. § 102 (2000). Section 102 provides in part:

A person shall be entitled to a patent unless—

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for patent, or

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of the application for patent in the United States, or

(c) he has abandoned the invention, or

(d) the invention was first patented or caused to be patented, or was the subject of an inventor's certificate, by the applicant or his legal representatives or assigns in a foreign country prior to the date of the application for patent in the United States, or

(e) The invention was described in—

(1) an application for patent, published under section 122(b), . . . ; or

(2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, . . . ; or

(f) he did not himself invent the subject matter sought to be patented, or

(g)(1) during the course of an interference conducted under section 135 or section 291, another inventor involved therein establishes, to the extent permitted in section 104, that before such person's invention thereof the invention was made by such other inventor and not abandoned, suppressed, or concealed, or (2) before such person's invention thereof, the invention was made in this country by another inventor who had not abandoned, suppressed, or concealed it. In determining priority of invention under this subsection, there shall be considered not only the respective dates of conception and reduction to practice of the invention, but also the reasonable diligence of one who was first to conceive and last to reduce to practice, from a time prior to conception by the other.

Id.

41. 35 U.S.C. § 103 (2000). Section 103 provides in part:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.
be more susceptible to certain rejections. 42 These rejections are also commonplace in non-software patent applications, but because of the relatively new entry of software patents into the patent realm and the ever-changing terminology used in the patent application, software patents seem to be especially susceptible to these rejections. 43

One common rejection is "that differences between a claimed invention and a prior art reference represent 'mere design alternatives.'" 44 The second common rejection is that "it would have been obvious to one skilled in the art at the time of the invention to combine the teaching of" prior art with another idea to give the claimed invention. 45 Besides these two rejections, patent practitioners have to be concerned with all of the other rejections that may stem from §§ 102 and 103. These rejections include publications, prior sales, public use, and other similar rejections stemming from the statutes. Accordingly, patent practitioners should be aware of the common software rejections and draft the software patent application in view of these rejections.

II. TERMINOLOGY

A. Method

The use of methods or processes in software patents is widely used, because most software inventions are implemented in the computer by a method or algorithm. Section 100(b) of the Patent Act states that "'process' means process, art or method, and includes a new use of a known process, machine, manufacture, composition of matter, or material." 46 "Software process claims can be found in roughly 85% of all issued U.S. software patents." 47 One reason that methods are widely used is that algorithms are easily transferred into a description and claim utilizing a method approach. Thus, the widespread use of the method and process claims make it imperative that patent practitioners understand the limitations and properly utilize the claims to fully describe the invention.

43. *Id.*
44. *Id.*
45. *Id.*
46. 35 U.S.C. § 100(b).
47. Witek, supra note 14, at 385.
A method claim must involve "the transformation or conversion of subject matter representative of or constituting physical activity or objects." It is important that the method not just do acts within the computer without some sort of significant output to show for the method. In addition, the claims cannot "consist solely of mathematical operations without some claimed practical application" or "simply manipulate abstract ideas ... without some claimed practical application." Patent practitioners need to pay special heed to ensure that the method specifically accomplishes some sort of "practical application."

A process is statutory if it requires physical acts to be performed outside the computer independent of and following the steps to be performed by a programmed computer, where those acts involve the manipulation of tangible physical objects and result in the object having a different physical attribute or structure.

This "manipulation of tangible physical objects" can include several categories of different acts. One type of act is "physical acts to be performed outside the computer." Examples of this type of act are:

- A method of curing rubber in a mold which relies upon updating process parameters, using a computer processor to determine a time period for curing the rubber, using the computer processor to determine when the time period has been reached in the curing process and then opening the mold at that stage.

- A method of controlling a mechanical robot which relies upon storing data in a computer that represents various types of mechanical movements of the robot, using a computer processor to calculate positioning of the robot in relation to given tasks to be performed by the robot, and controlling the robot's movement and position based on the calculated position.

Both of these acts are physical acts that are accomplished outside of the computer, but the computer is used to control the parameters of the physical acts. These types of physical acts are allowable patent claims because the

49. MPEP, supra note 21, at 2100-12.
50. Id.
51. Id. at 2100-15 (citing Diamond v. Diehr, 450 U.S. 175, 187 (1981)).
52. Id.
53. Id.
54. MPEP, supra note 21, at 2100-15 to -16.
objects have “a different physical attribute or structure” upon completion of the act.\textsuperscript{55} Thus, one way to ensure compliance is to ensure that acts outside of the computer are being accomplished by the method. Physical acts outside of the computer are sometimes difficult—if not impossible—to integrate into software patent claims, but where the computer is interacting without outside objects, this type of claim reduces the risk that the claim will be rejected by the USPTO.

Another type of act that satisfies the statutory requirements is an act that produces “a useful, concrete, and tangible result.”\textsuperscript{56} This type of act allows methods that consist solely in the computer without touching physical items outside of the computer. However, the act must have a result that accomplishes a real goal instead of just pushing around ones and zeros and giving no real result. This distinction may sometimes be difficult to achieve because simply applying an algorithm in a software claim will not suffice to protect the invention while applying an algorithm to achieve “a useful, concrete, and tangible result” has been held to meet the statutory requirements.\textsuperscript{57} The issue then becomes what exactly is “a useful, concrete, and tangible result.”\textsuperscript{58} The following examples are acts that produce “a useful, concrete, and tangible result”:\textsuperscript{59}

- A computerized method of optimally controlling transfer, storage and retrieval of data between cache and hard disk storage devices such that the most frequently used data is readily available.

- A method of controlling parallel processors to accomplish multi-tasking of several computing tasks to maximize computing efficiency . . . .

- A method of making a word processor by storing an executable word processing application program in a general purpose digital computer’s memory, and executing the stored program to impart word processing functionality to the general purpose digital computer by changing the state of the computer’s arithmetic logic unit when program instructions of the word processing program are executed.

\textsuperscript{55} Id. at 2100-15 (citing \textit{Diamond}, 450 U.S. at 187).
\textsuperscript{56} In re \textit{Alappat}, 33 F.3d 1526, 1544 (Fed. Cir. 1994).
\textsuperscript{57} Id.
\textsuperscript{58} Id.
\textsuperscript{59} Id.
A digital filtering process for removing noise from a digital signal comprises the steps of calculating a mathematical algorithm to produce a correction signal and subtracting the correction signal from the digital signal to remove the noise.  

Although some of these acts may not seem to produce "a useful, concrete, and tangible result," the optimization of data for increased speed or efficiency is a result that has been held to be useful, concrete, and tangible. Although a process "simply calculating a mathematical algorithm that models noise is nonstatutory." In addition, the imparting of increased functionality to the computer by introducing program instructions to the computer has also been held statutory. Thus, it is imperative that method claims give a result that is not just a mathematical algorithm, but also give a result that is actually accomplishing a result that can be quantified as accomplishing a goal that can be utilized by humans instead of just a machine calculation.

The following example is claim one of U.S. Patent No. 5,333,184:

1. A method for use in a telecommunications system in which interexchange calls initiated by each subscriber are automatically routed over the facilities of a particular one of a plurality of interexchange carriers associated with that subscriber, said method comprising the steps of:
   
   generating a message record for an interexchange call between an originating subscriber and a terminating subscriber, and including, in said message record, a primary interexchange carrier (PIC) indicator having a value which is a function of whether or not the interexchange carrier associated with said terminating subscriber is a predetermined one of said interexchange carriers.

Claim one of U.S. Patent No. 5,333,184 was challenged in AT&T Corp. v. Excel Communications, Inc. as being outside the scope of § 101. The important aspect of claim one is that the method generates a primary interex-

60. MPEP, supra note 21, at 2100-18 (citing In re Bernhart, 417 F.2d 1395, 1400 (C.C.P.A. 1969)).
61. Alappat, 33 F.3d at 1544.
62. MPEP, supra note 21, at 2100-18.
63. Id.
64. See id.
66. 172 F.3d 1352 (Fed. Cir. 1999).
67. Id. at 1358.
change carrier indicator which is "a useful, non-abstract result that facilitates differential billing of long-distance calls made by an IXC's subscriber." The court held that "[b]ecause the claimed process applies the Boolean principle to produce a useful, concrete, tangible result without pre-empting other uses of the mathematical principle, on its face the claimed process comfortably falls within the scope of § 101."69

In AT&T Corp., the important distinction that allowed this claim to be statutory was that the result had "a useful, concrete, and tangible result."70 The result was useful because it facilitated the collection of long distance billing data.71 The result was concrete because it generated and included the primary interexchange carrier indicator that was used for a certain purpose.72 The result was tangible because it was used to bill customers and therefore could be used to produce tangible bills for the customers.73 Accordingly, this claim was held to be statutory and is an example of how the specification interlinks with the claim language to ensure compliance with the requirements of software patents.74

The following example is claim one of U.S. Patent No. 6,763,397:

1. A method for verifying instructions in a module of a computer program, the method comprising:
   ascertaining whether checking an instruction in a first module that is loaded requires information in a referenced module different than the first module;
   if the information is required, determining whether the referenced module is already loaded;
   if the referenced module is determined to be not already loaded, writing a constraint for the referenced module without loading the referenced module; and
   verifying the instruction in the first module, the verifying comprising placing a list including a referenced type defined in a not-yet-loaded module and a different type at a fixed position in a merged snapshot.75

68. Id.
69. Id. (citing Arrhythmia Research Tech., Inc. v. Corazonix Corp., 958 F.2d 1053, 1060 (Fed. Cir. 1992)).
70. Alappat, 33 F.3d at 1544.
71. AT&T Corp., 172 F.3d at 1358.
72. Id.
73. Id.
74. Id.
The question then becomes whether claim one has "a useful, concrete, and tangible result."\(^{76}\) The method claim is checking about whether instructions have loaded into a module and verifying those results.\(^{77}\) This verification is important to ensure that information is correctly loaded. The method claim is also reducing the loading of modules by writing a constraint in lieu of loading the full instruction set. This type of optimization is similar to the "computerized method of optimally controlling transfer, storage and retrieval of data between cache and hard disk storage devices such that the most frequently used data is readily available."\(^{78}\) Since this method is optimizing the access of data for increased access speed, this claim satisfies the statutory requirements of § 101.\(^{79}\)

After meeting the requirements of § 101, patent practitioners also have to ensure that the method claim meets the requirements of § 112.\(^{80}\) This includes satisfying the first paragraph of § 112, which requires that the invention is fully and clearly disclosed, that the description is enabling, and that the best mode is disclosed.\(^{81}\) In addition, the second paragraph of § 112 must also be satisfied. This includes claiming the applicant's invention and specifying the proper bounds of the invention.\(^{82}\) In addition, patent practitioners should be aware of the common rejections under § 102\(^{83}\) and § 103.\(^{84}\) Accordingly, patent practitioners drafting software patents have many concerns to ensure that the patent application will not be rejected and that the inventor will receive the broadest application of the claims.

B. **Apparatus**

After *State Street Bank & Trust Co. v. Signature Financial Group, Inc.*,\(^{85}\) apparatus claims in software patents are not as limited by the type of software that is being implemented on the apparatus, but whether the apparatus accomplishes "a useful, concrete, and tangible result."\(^{86}\) Apparatus claims also have to be "directed to a specific apparatus of practical utility and

\(^{76}\) *Alappat*, 33 F.3d at 1544.
\(^{77}\) Patent 6,763,397, *supra* note 75.
\(^{78}\) MPEP, *supra* note 21, at 2100-18.
\(^{81}\) MPEP, *supra* note 21, at 2100-163.
\(^{82}\) *Id.* at 2100-203.
\(^{85}\) 149 F.3d 1368 (Fed. Cir. 1998).
\(^{86}\) *Id.*, at 1373 (quoting *Alappat*, 33 F.3d at 1544).
specified application." Thus, apparatus claims have the same type of limitation as method claims. One benefit to an apparatus claim is that the burden of proving utility may be a little lower because the hardware aspect of the apparatus would in itself lend a type of machine that may qualify under § 101. Section 101 of the Patent Act provides that "[w]hoever 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." Thus, patent practitioners need to ensure that the apparatus claims meet the requirement of "a useful, concrete, and tangible result."

One way the requirement for "a useful, concrete, and tangible result" is met is to ensure a relationship between the hardware devices and software implementation. "If it appears that the mathematical algorithm is implemented in a specific manner to define structural relationships between the physical elements of the claim (in apparatus claims) or to refine . . . , the claim being otherwise statutory, the claim passes muster under § 101." The following example is claim fifteen of U.S. Patent No. 6,763,397:

15. A verification apparatus comprising:
   a computer readable storage medium for storing a module of a computer program;
   a memory into which a module is loaded;
   a processor configured to ascertain whether checking an instruction in a first module that is loaded requires information in a referenced module different than the first module, to determine whether the referenced module is already loaded if the information is required, to write a constraint for the referenced module without loading the referenced module if the referenced module is determined to be not already loaded, and to verify the instruction in the first module, wherein verifying comprises placing a list including a referenced type defined in a not-yet-loaded module and a different type at a fixed position in a merged snapshot.

88. § 101.
89. Id.
90. Alappat, 33 F.3d at 1544.
91. Id.
93. Patent 6,763,397, supra note 75.
One issue for claim fifteen of U.S. Patent No. 6,763,397 is whether the result is “useful, concrete, and tangible.”\(^{94}\) This analysis is similar to the analysis for method claims. In this regard, the claim is accomplishing something besides pushing ones and zeros around.\(^{95}\) The claim is loading and verifying modules in the apparatus for verification purposes and thus seems to accomplish a result that is not just an algorithm.

In addition, the verification apparatus of claim fifteen of U.S. Patent No. 6,763,397 comprises the hardware devices which then have instructions that are loaded into the hardware.\(^{96}\) Then verification is done on the module.\(^{97}\) The hardware devices are the computer-readable storage medium, the memory, and the processor.\(^{98}\) All of these hardware devices are linked to the software because the software is just contained on the hardware devices.\(^{99}\) The software is simply telling the hardware what to do and is thus acting as a machine.

The following example is claim one of U.S. Patent No. 5,249,290:

1. A server apparatus for accessing one or more common resources using a plurality of server processes to which client service requests are assigned, said server apparatus comprising means for receiving an unassigned client service request . . . and means . . . for assigning said unassigned received client service request to a server process having a workload indication which is less than the workload indication of all other server processes.\(^{100}\)

The first issue for claim one of U.S. Patent No. 5,249,290 is whether the result is “useful, concrete, and tangible.”\(^{101}\) In this regard, claim one falls within the method claim examples in that the server apparatus is rearranging the workload between the servers to optimize the efficiency between servers.\(^{102}\) Accordingly, apparatus claims that are similar in result to method claims fall within the same categories and thus meet the statutory requirements in the same way as method claims. In addition, other types of appara-

\(^{94}\) Alappat, 33 F.3d at 1544.
\(^{95}\) See Patent 6,763,397, supra note 75
\(^{96}\) Id.
\(^{97}\) Id.
\(^{98}\) Id.
\(^{99}\) Id.
\(^{101}\) Alappat, 33 F.3d at 1544.
\(^{102}\) See Patent 5,249,290, supra note 100; see MPEP, supra note 21, at 2100-18.
SOFTWARE TERMINOLOGY

103. See MPEP, supra note 21, at 2100-14.
104. See id.
105. Patent 5,249,290, supra note 100.
106. Alappat, 33 F.3d at 1544.
107. Patent 5,249,290, supra note 100.
108. See MPEP, supra note 21, at 2100-18.
109. Alappat, 33 F.3d at 1544.
A second subtype of apparatus claims is system claims. System claims are a subtype of apparatus claims because the system is simply a type of specialized apparatus. The following example is claim forty-two of United States Patent No. 5,878,434:

42. A system for clash handling comprising:

a first computer;
a second computer connected to the first computer by a network link, the first computer and the second computer each containing a replica of a distributed database;
means for merging out a representation of operations performed on the first computer and applying at least a portion of the operations to the second replica;
means for merging in a representation of operations performed on the second computer and applying at least a portion of the operations to the first computer replica;
means for detecting persistent clashes during at least one of the merging steps; and
means for recovering from at least a portion of the detected persistent clashes.  

The first issue for claim forty-two of U.S. Patent No. 5,878,434 is whether the result is "useful, concrete, and tangible." Claim forty-two is similar to method claims that increase the efficiency between computers except that this claim increases the efficiency by fixing the conflicts between the databases. Accordingly, this claim was allowed by the USPTO and seems to meet the statutory requirements for software patents.

3. Means-Plus-Function

A third subtype of apparatus claims is a means-plus-function claim. A means-plus-function claim is a subtype of apparatus claims because it is a type of machine, but the difference is that the machine is described by its function instead of a given name. Since the means-plus-function claim gets meaning from how it is described, the description of means-plus-function

111. Alappat, 33 F.3d at 1544.
112. Patent 5,878,434, supra note 110; see MPEP, supra note 21, at 2100-18.
113. See Patent 5,878,434, supra note 110; see MPEP, supra note 21, at 2100-18.
claims in the specification is important because without an adequate description, the means-plus-function claims may be rejected.\textsuperscript{114} In this regard, paragraph six of §112 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.\textsuperscript{115}

In \textit{B. Braun Medical, Inc. v. Abbott Laboratories},\textsuperscript{116} the court held that a "structure disclosed in the specification is ‘corresponding’ structure only if the specification or prosecution history clearly links or associates that structure to the function recited in the claim."\textsuperscript{117} In addition, the court stated:

[I]f one employs means-plus-function language in a claim, one must set forth in the specification an adequate disclosure showing what is meant by that language. If an applicant fails to set forth an adequate disclosure, the applicant has in effect failed to particularly point out and distinctly claim the invention as required by the second paragraph of section 112.\textsuperscript{118}

Patent practitioners need to be concerned that the claim and specification are linked in such a manner that "particularly point[s] out and distinctly claim[s] that particular means."\textsuperscript{119} Thus, it is vital to the software patent application that the patent practitioner ensures all means-plus-function claims are properly supported in the specification of the application. Another aspect of this requirement is that the "interpretation of what is disclosed must be made in light of the knowledge of one skilled in the art."\textsuperscript{120} This interpretation is limited to "whether one of skill in the art would understand the specification itself to disclose the structure, not simply whether that person would be capable of implementing that structure."\textsuperscript{121}

\textsuperscript{114} MPEP, \textit{supra} note 21, at 2100-221.
\textsuperscript{116} 124 F.3d 1419 (Fed. Cir. 1997).
\textsuperscript{117} \textit{Id.} at 1424.
\textsuperscript{118} \textit{Id.} at 1425 (quoting \textit{In re Donaldson Co.}, 16 F.3d 1189, 1195 (Fed. Cir. 1994)).
\textsuperscript{119} \textit{Id.}
\textsuperscript{120} \textit{Atmel Corp. v. Info. Storage Devices, Inc.}, 198 F.3d 1374, 1380 (Fed. Cir. 1999).
\textsuperscript{121} \textit{Med. Instrumentation & Diagnostics Corp. v. Elekta AB}, 344 F.3d 1205, 1212 (Fed. Cir. 2003) (citing \textit{Atmel Corp.}, 198 F.3d at 1382).
To reiterate, the following example is claim forty-two of United States Patent No. 5,878,434:

42. A system for clash handling comprising:
   a first computer;
   a second computer connected to the first computer by a network link, the first computer and the second computer each containing a replica of a distributed database;
   means for merging out a representation of operations performed on the first computer and applying at least a portion of the operations to the second replica;
   means for merging in a representation of operations performed on the second computer and applying at least a portion of the operations to the first computer replica;
   means for detecting persistent clashes during at least one of the merging steps; and
   means for recovering from at least a portion of the detected persistent clashes. 122

The means-plus-function terminology is further defined in the specification of U.S. Patent No. 5,878,434:

Synchronization of the database replicas is performed after the computers are reconnected. Synchronization includes a “merging out” step, a “merging in” step, and one or more clash handling steps. During the merging out step, operations performed on the first computer are transmitted to the second computer and applied to the second replica. During the merging in step, operations performed on the second computer are transmitted to the first computer and applied to the first replica. 123

The first issue is to ensure that the means-plus-function claim is properly supported in the specification. 124 Claim forty-two of United States Patent No. 5,878,434 discusses a means for merging operations on computers. 125 The specification of United States Patent No. 5,878,434 then discusses how this merging cooperation is accomplished between the computers. 126 Thus, the specification is supporting the means-plus-function part of claim forty-two and satisfies the requirements of statutory and case law.

123. Id.
124. See B. Braun Med., Inc., 124 F.3d at 1425 (quoting Donaldson Co., 16 F.3d at 1195).
126. Id.
The second issue is "whether one of skill in the art would understand the specification itself to disclose the structure, not simply whether that person would be capable of implementing that structure." This issue is important practically because one skilled in the art depends on the exact field of art that is being described. Claim forty-two of United States Patent No. 5,878,434 is about database operations. Thus, the knowledge of one skilled in the art of databases would need to be examined to ensure the specification fully describes the invention. In this case, the specification describes in detail the process of the merging operation while not describing the communication between the two computers. Failing to state how the two computers communicate is not critical because one skilled in the art of computers—and especially databases—would understand the specification without describing in exact detail the communication used between the two computers.

The following example is claim one of United States Patent No. 5,249,290:

1. A server apparatus for accessing one or more common resources using a plurality of server processes to which client service requests are assigned, said server apparatus comprising means for receiving an unassigned client service request requesting access to one of said common resources and means, responsive to a workload indication from each server process, each workload indication being less than a maximum workload for that server process, for assigning said unassigned received client service request to a server process having a workload indication which is less than the workload indication of all other server processes.

Claim one of United States Patent No. 5,249,290 is another example of a means-plus-function claim. The means-plus-function aspect of this claim is part of the server apparatus. This claim is simply an example of a means-plus-function claim and no relation to the requirements of means-plus-function claims is reviewed.

129. Id.
130. Patent 5,249,290, supra note 100.
131. Id.
C. Computer-Readable Medium

Computer-readable medium claims are important because they are part of the holistic protection of software. Without computer-readable medium claims, a competitor could copy software and avoid infringement until the program was executed. With computer-readable medium claims, the patent is infringed when the program is copied onto the medium. Accordingly, computer-readable medium claims are very important to ensure the holistic protection of software. In addition, there are two subtypes of computer-readable medium claims. These two subtypes are computer program product and articles of manufacture.

"[C]omputer programs embodied in a tangible medium, such as floppy diskettes, are patentable subject matter under 35 U.S.C. § 101 and must be examined under 35 U.S.C. §§ 102 and 103."132 Computer-readable medium claims must relate to a functional matter that has "a useful, concrete, and tangible result."133 One claim that produced "a useful, concrete, and tangible result"134 was a "claim to data structure stored on a computer-readable medium that increases computer efficiency held statutory."135 "When nonfunctional descriptive material is recorded on some computer-readable medium, it is not statutory since no requisite functionality is present to satisfy the practical application requirement. Merely claiming nonfunctional descriptive material stored in a computer-readable medium does not make it statutory."136

In addition, patent practitioners should carefully draft software patent claims that utilize data structures and computer programs. "Data structures not claimed as embodied in computer-readable media are descriptive material per se and are not statutory because they are not capable of causing functional change in the computer."137 "Similarly, computer programs claimed as computer listings per se . . . are not physical 'things.'"138

The following example is claim thirty-one of U.S. Patent No. 5,878,434:

31. A computer-readable storage medium having a configuration that represents data and instructions which cause a first computer

132. In re Beauregard, 53 F.3d 1583, 1584 (Fed. Cir. 1995).
133. See Alappat, 33 F.3d 1526, 1544 (Fed. Cir. 1994).
134. Id.
135. MPEP, supra note 21, at 2100-12 (citing In re Lowry, 32 F.3d 1579, 1583-84 (Fed. Cir. 1994)).
136. Id.
137. Id. at 2100-13 (citing In re Warmerdam, 33 F.3d 1354, 1361 (Fed. Cir. 1994)).
138. Id.
and a second computer connected by a network link to perform method steps for handling clashes, the first computer and the second computer each containing a replica of a distributed database, the method comprising the steps of:

merging out a representation of operations performed on the first computer by applying at least a portion of the operations to the second replica;
merging in a representation of operations performed on the second computer by applying at least a portion of the operations to the first computer replica;
detecting persistent clashes during at least one of the merging steps; and
recovering from at least a portion of the detected persistent clashes. 139

The first issue is whether the claim produces "a useful, concrete, and tangible result."140 The computer-readable storage medium embodies a method of preventing database clashes.141 Thus, the method of preventing database clashes is increasing the efficiency of database operations by detecting and attempting to recover from clashes.142

The second issue is whether the claim contains data structures or computer programs not embodied into a computer-readable storage medium. Claim thirty-one contains a computer program that is embodied in the computer-readable storage medium.143 Thus, the computer program is not standing on its own without a hardware implementation and satisfies the requirements of patent law.

1. Computer Program Product

One subtype of a computer-readable storage medium is a computer program product. A computer program product is a medium similar to a storage medium. The following example is claim nine of United States Patent No. 6,763,397:

9. A computer program product for verifying a module of a computer program, the product comprising:
   a computer readable storage medium;

139. Patent 5,878,434, supra note 110.
140. Alappat, 33 F.3d at 1544.
141. Patent 5,878,434, supra note 110.
142. Id.
143. Id.
computer controlling commands, stored on the computer readable storage medium, for ascertaining whether checking an instruction in a first module that is loaded requires information in a referenced module different than the first module, for determining whether the referenced module is already loaded if the information is required, for writing a constraint for the referenced module without loading the referenced module if the referenced module is not already loaded, and for verifying the instruction in the first module, the verifying comprising placing a list including a referenced type defined in a not-yet-loaded module and a different type at a fixed position in a merged snapshot.144

The same issues that were addressed in a computer-readable storage medium have to be addressed in a computer program product. The first issue is whether the claim produces "a useful, concrete, and tangible result."145 The claim checks, verifies, and obtains information for a module.146 This produces a result that meets the requirements because the result increases the efficiency of the machine. The second issue is whether the claim contains data structures or computer programs not embodied into a computer program product. This claim embodies the computer program into the hardware element and thus meets the requirements under the controlling case law.147 Accordingly, computer program products are simply another way to write computer-readable medium claims.

2. Article of Manufacture

Another subtype of computer-readable medium claims is article of manufacture claims. This type of claim has to meet the same requirements as a computer-readable medium claim although the burden may be a little less when the article of manufacture falls under the definition of machine in § 101.148 “If a claim defines a useful machine or manufacture by identifying the physical structure of the machine or manufacture in terms of its hardware or hardware and software combination, it defines a statutory product.”149

144. Patent 6,763,397, supra note 75.
145. Alappat, 33 F.3d at 1544.
146. Patent 6,763,397, supra note 75.
147. See MPEP, supra note 21, at 2100-12.
149. MPEP, supra note 21, at 2100-14.
This seemingly lower burden may help in claim allowance, but the claim still has to produce "a useful, concrete, and tangible result."\textsuperscript{150} The following example is claim three of United States Patent No. 6,760,799:

3. An article of manufacture comprising a computing-device readable medium having encoded thereon instructions to direct a processor to perform the operations of:
   - queuing a first network traffic unit having an associated origin;
   - receiving a second network traffic unit having an associated origin, and receiving subsequent network traffic units each having an associated origins;
   - comparing said origins of said first and second network traffic units; and
   - if said origins differ, then queuing said second unit, else interrupting a host processor for the first unit and the second unit, and interrupting the host processor for each one of the subsequent network traffic units consecutively received after the second network traffic unit if said subsequent network traffic units are determined to be substantially received from said origin of said second network traffic unit.\textsuperscript{151}

Claim three is similar to computer-readable medium claims because it is comprised of "a computing-device readable medium."\textsuperscript{152} In addition, it is similar to a method claim because it is accomplishing a method on the article of manufacture.\textsuperscript{153} This claim compares received information and retransmits if the buffers do not match.\textsuperscript{154} In this regard, claim three is increasing efficiency of the transfer operation and thus produces "a useful, concrete, and tangible result."\textsuperscript{155} Accordingly, article of manufacture claims are another type of claim that can be utilized for software patents.

\section*{III. CONCLUSION}

This article sets forth the reasons why patent practitioners who draft software patents should be knowledgeable about the different terminology that can be utilized in describing software technology. The three main types

\begin{verbatim}
150. Alappat, 33 F.3d at 1544.
152. Id.
153. Id.
154. Id.
155. Alappat, 33 F.3d at 1544.
\end{verbatim}
of software claims are method claims, apparatus claims, and computer-readable medium claims. Each of these claim types has subtypes that can be utilized in properly claiming the software invention. Accordingly, there are many choices for describing the software invention, but each type of claim has its own advantages and disadvantages.

The optimum protection of software inventions comes from the use of the three main types of software claims. The use of method, apparatus, and computer-readable medium claims each have their own advantages in light of meeting the statutory and case law requirements and when used in the offensive or defensive infringement stance. Method claims are useful since most software inventions are implemented in the computer via an algorithm. Method claims describe these algorithms in an efficient way. In addition, since software inventors are familiar with algorithms, they are more likely to understand the specification describing the claims, the method claims, and the respective diagrams. Accordingly, method claims are an excellent tool in describing software inventions.

Apparatus claims are also well suited to describe software inventions since software is implemented on computer hardware devices. Since the computer hardware is an apparatus, there is a direct correlation between the apparatus that is stated and the hardware that is utilized in a software invention. Two of the subtypes of apparatus claims are types of computer hardware described in computer terminology. Means-plus-function claims are useful in software inventions because some types of means do not have a definition that is well known in the field of art, and thus the function of the means can give a better description of the component than an artificially defined word. Accordingly, apparatus claims have the benefit of being closely related to the hardware that most software inventions are implemented.

Computer-readable medium claims provide protection to software inventors to decrease the copying of patented software. Without computer-readable medium claims, it is possible for the patented software to be copied and sold without infringement. The only infringement with only method claims and possibly apparatus claims would occur when the software is executed, and thus the method or apparatus is infringed. With computer-readable medium claims, the copying of the claimed software will infringe the patent when the software is copied onto the computer-readable medium. Thus, computer-readable medium claims have an important protection aspect to software patents.

156. MPEP, supra note 21, at 2100-12.
157. See id.
Patent practitioners who draft software patents should utilize each of the main types of claims when drafting a software patent application. In addition, patent practitioners should pay special attention to the claim types to ensure that each of the special requirements for each type are satisfied in the specification of the patent application. For example, means-plus-function claims have to contain a disclosure of the structure in the specification.\textsuperscript{158} Thus, a "structure disclosed in the specification is 'corresponding' structure only if the specification or prosecution history clearly links or associates that structure to the function recited in the claim."\textsuperscript{159} Accordingly, the use of the claim types necessitates an understanding of the requirements.

One commonality between all three of the main types of claims used in software patents is that the claim must have "some type of practical application, i.e., 'a useful, concrete and tangible result.'"\textsuperscript{160} In making this determination, the concern for software is not "which of the four categories of subject matter a claim is directed to—process, machine, manufacture, or composition of matter—but rather on the essential characteristics of the subject matter, in particular, its practical utility."\textsuperscript{161} Thus, patent practitioners do not have to be concerned about fitting into the box of the four categories but must be concerned with the overall result of the claim.

In conclusion, all three types of claims are important for the protection of software patents. The use of all three together helps obtain a holistic protection mechanism for the software. This article gives an overview of the pitfalls that patent practitioners should be aware of when drafting the different types of claims. Patent practitioners who draft software patents have many issues that are of concern when drafting the patent application, but the use of all three types of claims will maximize the protection of the software.

\textsuperscript{158} B. Braun Med., Inc. v. Abbott Labs., 124 F.3d 1419, 1425 (Fed. Cir. 1997).
\textsuperscript{159} Id. at 1424.
\textsuperscript{160} State St. Bank & Trust Co. v. Signature Fin. Group, Inc., 149 F.3d 1368, 1373 (Fed. Cir. 1998) (quoting In re Alappat, 33 F.3d 1526, 1544 (Fed. Cir. 1994)).
\textsuperscript{161} Id. at 1375.