United States District Court
Northern District of California

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

ANCORA TECHNOLOGIES, INC., **Plaintiff,** **v.** **APPLE INC.,** **Defendant.** **AND RELATED COUNTER-CLAIM**	**Case No.: 11-CV-06357 YGR** **CLAIM CONSTRUCTION ORDER**

Ancora Technologies, Inc. ("Ancora") alleges that devices that run Apple Inc.'s ("Apple") iOS operating system infringe on U.S. Patent No. 6,411,941 (the " '941 Patent"). Apple has counterclaimed for declaratory judgments of non-infringement and invalidity.

The parties have requested the Court construe seven claim terms/phrases from the '941 Patent: (1) "volatile memory"; (2) "non-volatile memory"; (3) "BIOS"; (4) "program"; (5) "license record"; (6) "verifying the program using at least the verification structure"; and (7) whether the steps in the asserted claims must be performed in a specific order. On June 29, 2012, the parties provided a technology tutorial and on July 11, 2012, the Court held a claim construction hearing.

Based upon the papers submitted, the argument of counsel, for the reasons set forth below, the Court provides the following claim construction.

I. BACKGROUND

The patent in suit relates to software anti-piracy technology. At issue here is technology directed at preventing computer users from copying software and then running that software without a license. Ancora is the owner of the '941 Patent, which claims a method of restricting software

operation within a license limitation, *i.e.* it teaches a system for ensuring that only the authorized user of software can operate the software at issue. Apple's iOS operating system also restricts software operation within a license limitation. Ancora alleges that the Apple products that run the iOS operating system infringe on the '941 Patent.

The '941 Patent uses the memory of a computer's "BIOS" to store a "license record" to confirm whether a "program" is licensed to run on that computer. Every computer has a unique identifier embedded at the time of manufacture. Under the teachings of the '941 Patent, when a licensed program first launches it generates a license record using the computer's unique identifier, which license record is stored in the BIOS area of a computer. This license record is unique to that particular computer. When a licensed program is loaded, it can verify whether the software is licensed to run on that computer by referencing the license record stored in the BIOS with the license record from the program. If they match, the program continues to run. If the program has been copied, the license information does not match and the program will not run.

A. BACKGROUND OF THE PATENT

Plaintiffs provide the following background: In 1997, when Miki Mullor and Julian Valiko, the co-inventors of the '941 Patent ("Patentees"), began developing the technology that would become the '941 Patent, there were two approaches to combating software piracy, a hardware approach and a software approach. The hardware approach was costly, inconvenient and not suitable for software downloaded from the internet required as it required users of software to use a piece of hardware called a "dongel" in order to access the software. The software based products were too easily hacked by skilled programmers.

Patentees developed a third approach that had the advantages of both the hardware approach and software approach without the disadvantages of either. Patentees identified available memory space in hardware stored on the computer's motherboard, the BIOS, which they repurposed to store software licensing technology. The inventive aspect of the '941 Patent is that the writable, non-volatile memory of the BIOS is not ordinarily considered to be a storage medium for software licensing technology. The advantage of using the BIOS for this purpose is that the level of programming expertise required to tamper with data stored in the BIOS is substantially greater than

the expertise needed to tamper with data residing in volatile memory, and unsuccessful tampering comes with higher risk as it could render the computer inoperable.

Patentees applied for an Israeli patent in 1998. On October 1, 1998, Patentees applied for the '941 Patent, with a priority date of May 21, 1998 based upon the Israeli patent. The '941 Patent issued in 2002.

B. CLAIM TERMS/PHRASES TO BE CONSTRUCTED

Sixteen claims from the '941 Patent are asserted: independent Claim 1, and dependent Claims 2, 3, and 5-17, which refer to it.

Claim 1, which is the only independent claim asserted, recites the following (the language the parties have identified for construction is in bold and italics):

> 1. A method of restricting software operation within a license for use with a computer including an erasable, ***non-volatile memory*** area of a ***BIOS*** of the computer, and a ***volatile memory*** area; the method comprising the steps of:
>
> selecting a ***program*** residing in the ***volatile memory***, using an agent to set up a verification structure in the erasable, ***non-volatile memory*** of the ***BIOS***, the verification structure accommodating data that includes at least one ***license record***,
>
> ***verifying the program using at least the verification structure*** from the erasable ***non-volatile memory*** of the ***BIOS***, and acting on the ***program*** according to the verification.

('941 Patent, claim 1).

The parties request the Court construe seven claim terms/phrases: (1) "volatile memory"; (2) "non-volatile memory"; (3) "BIOS"; (4) "program"; (5) "license record"; (6) "verifying the program using at least the verification structure"; and (7) All Asserted Claims.[1]

II. PRINCIPLES OF CLAIM CONSTRUCTION

Claim construction is a matter of law, to be decided by the Court. *Markman v. Westview Instruments, Inc.*, 517 U.S. 370, 387 (1996) (determination of infringement is a two-step analysis: First, the Court determines the scope and meaning of the claims; second, the properly construed claims are compared to the accused device.). "[T]he role of a district court in construing claims is … to give meaning to the limitations actually contained in the claims, informed by the written

[1] In addition, the parties have identified one term on which they have agreed on a construction ("verification structure accommodating data that includes at least one license record").

description, the prosecution history if in evidence, and any relevant extrinsic evidence." *American Piledriving Equipment, Inc. v. Geoquip, Inc.*, 637 F.3d 1324, 1331 (Fed. Cir. 2011). "Claim construction is a matter of resolution of disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the claims, for use in the determination of infringement." *U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997). Thus, claim terms need only be construed "to the extent necessary to resolve the controversy." *Wellman, Inc. v. Eastman Chemical Co.*, 642 F.3d 1355, 1361 (Fed. Cir. 2011) (citing *Vivid Technologies, Inc. v. American Science & Engineering, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999).[2]

The starting point in a claims construction analysis is the language of the claims themselves. These define the invention that the patentee may exclude others from practicing. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312-13 (Fed. Cir. 2005). The general rule is to construe a claim term in a manner consistent with its "ordinary and customary meaning," which is "the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention." *Id.* at 1312.

Claims must be read in view of the specification, of which they are a part and in a manner consistent with the patent's specification. *See Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 979 (Fed. Cir. 1995), *aff'd*, 517 U.S. 370 (1996). The specification may act as a sort of dictionary, which explains the invention and may define terms used in the claims. *Id.* The Court also should consider the patent's prosecution history, if it is in evidence. *Id.* at 980. The prosecution history may "inform the meaning of the claim language by demonstrating how the inventor understood the invention and whether the inventor limited the invention in the course of prosecution, making the claim scope narrower than it would otherwise be." *Phillips, supra*, 415 F.3d at 1317 (citing *Vitronics*, 90 F.3d at 1582-83); *see also Chimie v. PPG Indus., Inc.*, 402 F.3d 1371, 1384 (Fed. Cir.

[2] Once the meaning of a term used in a claim has been determined, the same meaning applies to that term for all claims in which the same term appears. *Inverness Med. Switzerland GmbH v. Princeton Biomeditech Corp.*, 309 F.3d 1365, 1371 (Fed. Cir. 2002). After a term is construed, the Court's construction becomes the legally operative meaning of the disputed terms that governs further proceedings in the case. *See Chimie v. PPG Indus., Inc.*, 402 F.3d 1371, 1377 (Fed. Cir. 2005). However, "district courts may engage in a rolling claim construction, in which the court revisits and alters its interpretation of the claim terms as its understanding of the technology evolves." *Pressure Products Medical Supplies, Inc. v. Greatbatch Ltd.*, 599 F.3d 1308, 1316 (Fed. Cir. 2010).

2005) ("The purpose of consulting the prosecution history in construing a claim is to exclude any interpretation that was disclaimed during prosecution.") (internal quotations omitted). The Court may, in its discretion, consider extrinsic evidence[3] if such sources will aid the Court in determining "the true meaning of language used in the patent claims." *Phillips, supra*, 415 F.3d at 1318.

Further, and as relevant here, whether a patent claim complies with the definiteness requirement of 35 U.S.C. § 112, ¶ 2 is also a matter of claim construction. *See Wellman, Inc. v. Eastman Chemical Co.*, 642 F.3d 1355, 1365-66 (Fed. Cir. 2011), *cert. denied*, 132 S.Ct. 1541 (2012). Section 112, paragraph 2 of the Patent Act provides in pertinent part: "[t]he specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention." 35 U.S.C. § 112, ¶ 2. This section contains two requirements: "first, [the claim] must set forth what 'the applicant regards as his invention,' and second, it must do so with sufficient particularity and distinctness, *i.e.*, the claim must be sufficiently 'definite.'" *Allen Eng'g Corp. v. Bartell Indus., Inc.*, 299 F.3d 1336, 1348 (Fed. Cir. 2002); *see also, Phillips, supra*, 415 F.3d at 1316. In determining whether a claim is sufficiently definite, the Court must consider whether "one skilled in the art would understand the bounds of the claim when read in light of the specification." *Allen Eng'g Corp., supra*, 299 F.3d at 1348 (citing *Personalized Media Comm'ns, LLC v. Int'l Trade Comm'n*, 161 F.3d 696, 705 (Fed. Cir. 1998). "Only claims 'not amenable to construction' or 'insolubly ambiguous' are indefinite." *Halliburton Energy Services, Inc. v. M-I LLC*, 514 F.3d 1244, 1250 (Fed. Cir. 2008) (quoting *Datamize, LLC v. Plumtree Software, Inc.*, 417 F.3d 1342, 1347 (Fed. Cir. 2005).

III. DISCUSSION

A. THE FIRST AND SECOND DISPUTED CLAIM TERMS – "VOLATILE MEMORY" & "NON-VOLATILE MEMORY" (CLAIMS 1-3, 5-17)

The parties' dispute focuses on the impact of whether examples provided in the specification render the claim indefinite.

[3] Although the use of extrinsic evidence is discretionary, the court may always consult technical treatises and dictionaries to understand the technology and to construe the claims, so long as no definition in the intrinsic evidence is contradicted.

The parties' proposed constructions are shown below:

TERM	APPLE'S PROPOSED CONSTRUCTION	ANCORA'S PROPOSED CONSTRUCTION
"volatile memory"	This term is indefinite under 35 U.S.C. § 112, ¶ 2.	memory that is not maintained when the power is removed
"non-volatile memory"	This term is indefinite under 35 U.S.C. § 112, ¶ 2.	memory that is maintained when the power is removed

Claim 1 calls out two different types of memory—volatile and non-volatile. The plain and ordinary meaning of the terms is not in dispute. To one of ordinary skill in the art, a volatile memory is memory whose data is not maintained when the power is removed and a non-volatile memory is memory whose data is maintained when the power is removed.

However, the '941 Patent specifically indicates that "volatile memory" can take the form of a hard disk. Apple takes issue with this pronouncement. A "hard disk" is generally known by those of ordinary skill in the art to be *non*-volatile memory because the memory is maintained when the power is removed. Apple argues that the Patentees' express identification of a hard disk as "volatile memory" runs counter to the understanding of one of ordinary skill in the art, and therefore, renders the terms indefinite under 35 U.S.C. § 112, ¶ 2. Next, it argues that a definition of "volatile memory" as "memory that is not maintained when the power is removed" excludes the very same hard disk that the specification identifies as volatile. Furthermore, Apple argues that this ambiguity affects the meaning of "non-volatile memory" by implication, which makes it impossible to determine the scope of both terms as those terms are used in the '941 Patent. According to Apple, this creates an insoluble ambiguity as to what characteristics render memory volatile or non-volatile. Thus, Apple argues that although the specification identifies only "hard disk" as a form of volatile memory, this insoluble ambiguity applies to other types of memories as well because there is no explanation of what characteristics render memory volatile or non-volatile. According to Apple, "[i]t is difficult to imagine a more compelling example of a situation in which '[c]ompetitors trying to practice the invention or to design around it would be unable to discern the bounds of the invention.'" (Opp'n 9 (quoting *Honeywell Int'l, Inc. v ITC*, 341 F.3d 1332, 1341 (Fed. Cir. 2008).)

Ancora argues that the claim terms should not be constructed at all because the plain and ordinary meaning of the terms is not in dispute and the structure disclosed in the specification is irrelevant to the claim construction analysis.[4] Ancora urges the Court to perform a claim construction divorced from the context of the specification and prosecution history. *Old Town Canoe Co. v. Confluence Holdings Corp.*, 448 F.3d 1309, 1318 (Fed. Cir. 2006).

Claims are not indefinite merely because they present a difficult task of claim construction. *See Liebel-Flarsheim Co. v. Medrad, Inc.*, 358 F.3d 898, 911 (Fed. Cir. 2004) ("unless the court concludes, after applying all the available tools of claim construction, that the claim is still ambiguous, the axiom regarding the construction to preserve the validity of the claim does not apply."). Instead, "[i]f the meaning of the claim is discernible, even though the task may be formidable and the conclusion may be one over which reasonable persons will disagree, we have held the claim sufficiently clear to avoid invalidity on indefiniteness grounds." *Halliburton Energy Services, Inc. v. M-I LLC*, 514 F.3d 1244, 1249 (Fed. Cir. 2008) (citing *Exxon Research & Eng'g Co. v. United States*, 265 F.3d 1371, 1375 (Fed. Cir. 2001) (citations omitted).

Ancora explains that volatile memory is routinely stored on the "hard disk" when two programs attempt to write data to the same location in RAM (random access memory), which it refers to as "virtual data." Thus, Ancora argues that it is entirely appropriate to consider a "hard disk" as a form of volatile memory to someone of ordinary skill in the art. Apple disagrees. Apple's view may be appealing to a layman; the same may not be true for someone of ordinary skill in the art.

Based on the foregoing analysis, the Court declines to find the term indefinite as a matter of law and finds that the claim terms should be given their ordinary meaning, specifically:

Volatile memory is memory whose data is not maintained when the power is removed; and non-volatile memory is memory whose data is maintained when the power is removed.

---

[4] Ancora also argues that construction of this term is unnecessary for the infringement analysis because the term hard disk does not appear in Claim 1 and Apple's accused devices do not store the license record in a hard-drive. However, Apple has raised a counterclaim of invalidity, and indefiniteness is a basis to find a patent invalid.

B. THIRD DISPUTED CLAIM TERM/ PHRASE – "BIOS" (CLAIMS 1-3, 5-17)

No party disputes that "BIOS" is an acronym for Basic Input/Output System. Rather, the parties dispute whether the term "BIOS" applies to all computers or only to IBM PC-compatible computers, as Apple proposes.[5] The parties' proposed constructions are shown below:

APPLE'S PROPOSED CONSTRUCTION	ANCORA'S PROPOSED CONSTRUCTION
software routines on IBM PC compatible computers that handle startup operations and support the transfer of data among peripheral devices	software routines that handle startup operations

The claims of the '941 Patent refers to the "non-volatile memory area of the BIOS" of a computer fourteen times (and refers to the "non-volatile memory area of *a* BIOS" once). Neither the claim nor the specification defines "BIOS." In the prosecution history the Patent Examiner gave the following definition of BIOS based upon "The Microsoft Computer Dictionary, 5th Edition, 2002":

> the set of essential software routines that test hardware at startup, start the operating system, and support the transfer of data among hardware devices.

The Patent Examiner further commented:

> This definition is consistent with the specification of the '941 patent. Since a BIOS is therefore defined by the functional descriptive material contained within it, one skilled in the art would consider any non-functional descriptive material, such as tables, to be part of the BIOS only if it is made and used by the functions of the BIOS itself. This does not preclude such material being also used or modified by programs located outside of the BIOS, such as applications running in an operative system. The fact that a program or tables resides in non-volatile memory does not necessarily mean that it is part of the BIOS. It is therefore the case that a reasonable examiner would only consider a table to be in a BIOS if it were, at a minimum, created by a function residing in the BIOS.

(Rondini Dec. ¶ 4, Ex. 3, Order Granting Request for Reexamination, at 8-9, ANCA 2568-69.)

---

[5] Apple also seeks to limit the scope of the definition so that BIOS handles transfer of data among "peripheral" devices. Ancora argues that the Claim does not mention "peripheral" devices and that the Examiner used the broader "hardware" language not the narrower term "peripheral." In a footnote, Apple states that its construction is consistent with the definitions in the technical dictionaries, but Apple's proposed construction is consistent with only one of the five the technical dictionaries quoted in Apple's *Markman* Brief. Apple has not otherwise explained why it believes that BIOS should be construed to interact with peripheral devices only. The Court has not found any basis to support this interpretation.

Apple points out that the definition supplied by the Patent Examiner is incomplete, referring again to the Microsoft dictionary:

> BIOS n. Acronym for basic input/output system. On PC-compatible computers, the set of essential software routines that tests hardware at startup, starts the operating system, and supports the transfer of data among hardware devices, including the date and time. The operating system date is initialized from the BIOS or Real Time Clock date when the machine is booted. Many older PCs, particularly those dating before 1997, have BIOSs that store only 2-digit years and thus may have suffered from Year 2000 problems. The BIOS is stored in read-only memory (ROM) so that it can be executed when the computer is turned on. Although critical to performance, the BIOS is usually invisible to computer users.

(*Id*. Ex. I (Microsoft Press Computer User's Dictionary (5th ed. 2002)).)[6]

---

[6] In addition to the dictionary definition used by the Patent Examiner, Apple offers four other dictionary definitions of the term BIOS:

> BIOS n. Acronym for basic input/output system. On PC-compatible computers, the set of essential software routines that test hardware at startup, start the operating system, and support the transfer of data among hardware devices. The BIOS is stored in ROM so that it can be executed when the computer is turned on. Although critical to performance, the BIOS is usually invisible to computer users.

(Rahebi Dec., Ex. K (Microsoft Press Computer User's Dictionary (3d ed. 1998)).)

> A set of programs encoded in read-only memory (ROM) on IBM PC-compatible computers. These programs handle startup operations such as the power-on self-test (POST) and low-level control for hardware, such as disk drives, keyboards, and monitor. The BIOS programs of IBM personal computers are copyrighted, so manufacturers of IBM PC-compatible computers must create BIOSs that emulate the IBM BIOS or buy an emulation from companies, such as Phoenix Technologies and American Megatrends, Inc. ….

(*Id*. Ex. F (Webster's New World Dictionary of Computer Terms (6th ed. 1997)); *id*. Ex. G (Que's Computer & Internet Dictionary (6th ed. 1995)).)

> [A] set of procedures stored on a ROM chip inside IBM PC compatible computers. These routines handle all input-output functions, including screen graphics, so that programs do not have to manipulate the hardware directly ….

(*Id*. Ex. H (Barron's Dictionary of Computer and Internet Terms (5th ed. 1996)).)

> On PC-compatible computers, the set of essential software routines that test hardware at startup, start the operating system, and support the transfer of data among hardware devices. The BIOS is stored in read-only memory (ROM) so that it can be executed when the computer is turned on. Although critical to performance, the BIOS is usually invisible to computer users.

(*Id*. Ex. E (Microsoft Press Computer Dictionary (3d ed. 1997)).)

> The part of the system software of the IBM PC and compatibles that provides the lowest level interface to peripheral devices and controls the first stage of the bootstrap process, including installing the operating system. The BIOS is stored in ROM, or equivalent, in every PC. Its main task is to load and execute the operating system which is usually stored on the computer's hard disk, but may be loaded from CD-ROM or floppy disk at install time.

(*Id*. Ex. J (Free On-Line Dictionary of Computing (June 6, 1999)).)

Apple proposes that BIOS be construed to operate only on IBM PC-compatible computers. According to Apple, the plain and ordinary meaning of the term "BIOS" at the time of the invention, as demonstrated by dictionary definitions is that BIOS is specific to the IBM PC-compatible computer platform.

Ancora urges the Court to avoid using a dictionary to construe BIOS by arguing that the Federal Circuit in *Phillips* stated that using a dictionary to alter the claim term violates the public notice function of the patent. The problem identified in *Phillips* is that a dictionary definition oftentimes will be overly broad and when taken out of the context of the patent at issue can lead to an "unduly expansive" construction of the claim term. 415 F.3d at 1321. The opposite problem is presented here. Apple is resorting to the dictionary definition to narrow the construction of the term because it believes that the Patent Examiner gave BIOS an unduly expansive construction.

Ancora argues that the Examiner defined BIOS broadly, not limited to IBM PC-compatible computers, and that the claim itself is not limited to a brand of computer but the "BIOS of a computer."[7] Ancora argues that virtually all computers have BIOS, including the accused devices (iPhone, Apple laptops and Apple desktops).[8]

As set forth above, "BIOS" stands for Basic Input/Output System; it is software code. No one disputes that a person of ordinary skill in the art reading the Claim in the context of the specification and prosecution history would understand the "BIOS" to be the location in the computer where the software code was stored. The inventive aspect of the '941 Patent was to write information onto unused memory in the BIOS area of the computer. The limiting aspect of the invention is to store information in the BIOS, not the type of computer that runs BIOS. The technical dictionaries that Apple has offered into evidence do not convince the Court otherwise.

---

[7] Although Ancora argues that the Patent Examiner gave BIOS a specific definition, "the set of essential software routines that test hardware at startup, start the operating system, and support the transfer of data among hardware devices," Ancora argues for an entirely different construction of BIOS tethered only to Apple's proposed construction. Other than to argue that its patent applies to non-IBM PC-compatible computers, Ancora does not base its interpretation on any intrinsic evidence.

[8] Ancora also offers evidence of a 1983 Commodore 64 computer that ran BIOS, and that during the mid-1990's Apple's Macintosh brand of computers, when operated in a PC-compatible mode, had BIOS. This extrinsic evidence both before and after the relevant time period does not reflect the usage or meaning in the art in 1998.

These dictionaries indicate that PC-compatible computers run BIOS. However, exclusionary language is not found in the proffered dictionaries, *i.e.* the dictionaries do not indicate that the non-IBM PC-compatible computers do not or cannot run BIOS, nor that a person of ordinary skill in the art would have understood as much. Accordingly, the Court will not define BIOS by the hardware architecture of the computer on which it runs.

The Court rejects both parties' proffers and provides the following construction:

"BIOS" is an acronym for Basic Input/Output System. It is the set of essential startup operations that run when a computer is turned on, which tests hardware, starts the operating system, and supports the transfer of data among hardware devices.

C. FOURTH DISPUTED CLAIM TERM/ PHRASE – "PROGRAM" (CLAIMS 1-3, 5-17)

The parties dispute whether an "operating system" is a type of "program" as that term is used in the '941 Patent. The parties' proposed constructions are shown below:

APPLE'S PROPOSED CONSTRUCTION	ANCORA'S PROPOSED CONSTRUCTION
Software application that interacts with and relies on the operating system	A set of instructions that can be executed by a computer.

Apple argues that (i) using the term program in the specification to refer only to application programs and then (ii) emphasizing the distinction during prosecution demonstrates a narrow use of the term. Ancora argues that (i) the specification does not clearly set forth a narrower definition of the term; (ii) the prior art reference over which the Examiner allowed the '941 Patent to issue described an operating system as a type of program; and (iii) there is no disavowal of the plain and ordinary meaning of the term in the prosecution history.

According to Ancora, "[e]very person of skill in the computer field knows that a 'program' is: 'a set of instructions that can be executed by a computer.'" (Ancora's Br. 12.) The construction that Ancora advances is the definition in the Microsoft Computer Dictionary, 5th Edition, 2002. By contrast, Apple argues that to a person of skill in the field, the term "program" as used in the '941 Patent means a "software application that interacts with and relies on the operating system." As explained below, Ancora's definition is too broad, while Apple's may be misunderstood by a jury.

The first step is to look at the language of the use of the term in the claim itself: the '941 Patent teaches a method of "… selecting a program residing in the volatile memory, … verifying the program …, and acting on the program."

Ancora argues that no construction is necessary because the term "program" is a commonly understood term. However, the term program has a different meaning depending on the context. *See REC Software USA, Inc. v. Bamboo Solutions Corp.*, 2012 WL 1533965, at *5 (W.D. Wash. April 30, 2012) ("the word has numerous meanings, *e.g.*, a computer program versus a baseball game program versus a television program."). A person attending a play would consider a program to be the list of the performers or performances; a person watching a television show would consider the show itself to be the program; to a computer programmer, a program is a "set of instructions." *See id.* The '941 Patent is this latter type of program. Due to the technical nature of the term, the Court believes that construction is necessary to resolve the dispute over the meaning of the term. *See U.S. Surgical Corp.*, *supra*, 103 F.3d at 1568 (claim construction may be necessary to resolve disputes over the meaning or scope of technical terms and words of art). The issue then is whether the invention applies to all types of computer programs, as Ancora argues, or application programs only, as Apple argues.

1. Specification

A technical term in a patent claim is construed in accordance with its description in the specification. *Phillips*, *supra*, 415 F.3d at 1315. Apple argues that the summary of the invention refers to "application program" ('941 Patent 1:53-54), and the specification refers to a "license verifier application" (*id.* 2:13-14), which can only run if the operating system is already running and thus could not verify the operating system. Apple also notes that the only "program" expressly identified in the specification is "Lotus 123," an application program. Ancora counters that the term "program" is used broadly within the specification to include "software" and "application," which demonstrates the breadth of the term rather than a limitation on the definition.

Words of a claim are to be given the meaning they would have to a person of ordinary skill in the art, who read the claims in the context of the description of the invention in the specification and the prosecution history. *See Phillips*, *supra*, 415 F.3d at 1313. The ordinary and customary

meaning of a claim term is not the dictionary definition[9] as Ancora would like, but the meaning of that claim term to a person of ordinary skill in the art "after reading the entire patent." *Id.* at 1320 ("Properly viewed, the 'ordinary meaning' of a claim term is its meaning to the ordinary artisan after reading the entire patent."). In reviewing the intrinsic record, this Court must also "strive to capture the scope of the actual invention, rather than strictly limit[ing] the scope of the claims to disclosed embodiments or allow[ing] the claim language to become divorced from what the specification conveys is the invention." *Retractable Techs., Inc. v. Becton*, 653 F.3d 1296, 1305 (Fed. Cir. 2011). There is a fine line between construing claims in light of their specification and improperly importing a limitation into a claim. *See Phillips*, *supra*, 415 F.3d at 1323. The distinction between proper claim construction and improper limitation turns on "whether a person of skill in the art would understand the embodiments to define the outer limits of the claim term or merely to be exemplary in nature." *Id*.

Ancora's use of the word program throughout the '941 Patent only refers to application programs. The '941 Patent teaches only a method of restricting the operation of unauthorized application programs. While the '941 Patent does not repeatedly use of the phrase "application program," it does not need the modifier for it to be clear that the '941 Patent does not teach a method of restricting the operation of all types of programs. From the outset of the specification of the '941 Patent, Ancora limits the nature of a "program." Thus:

> FIELD OF THE INVENTION
>
> This invention relates to a method and system of identifying and restricting an unauthorized software program's operation.

---

[9] The Federal Circuit in *Philips* instructed courts to avoid an over-expansive definition of claim terms: courts should "focus[ ] at the outset on how the patentee used the claim term in the claims, specification, and prosecution history, rather than starting with a broad definition and whittling it down." *Philips*, *supra*, at 1321. The *Phillips* Court's concern with dictionaries is that it may lead judges to construe terms in an overbroad manner:

> if the district court starts with the broad dictionary definition in every case and fails to fully appreciate how the specification implicitly limits that definition, the error will systematically cause the construction of the claim to be unduly expansive.

Id.

BACKGROUND OF THE INVENTION

> Numerous methods have been devised for the identifying and restricting of an unauthorized software program's operation.

('941 Patent 1:5-14.)

As described in the '941 Patent specification, the Patentees' purpose focused solely on the prevention of "illegal copying" of software. Ancora's use of the word "application" to modify the word "program" emphasizes the nature of the program at issue. While at times redundant, the language does not undermine the fundamental teachings of the patent. To assert that the patent teaches anything with respect to non-application based programs, such as an operating system, finds no basis in the language of the patent itself.

Based on the foregoing analysis, the Court concludes that the specification does not support Ancora's position that "program" is defined broadly in the '941 Patent.

2. Prosecution history

The prosecution history "provides evidence of how the PTO and the inventor understood the patent" and "can often inform the meaning of the claim language by demonstrating how the inventor understood the invention and whether the inventor limited the invention in the course of prosecution, making the claim scope narrower than it would otherwise be." *Phillips, supra*, 415 F.3d at 1317 (citing *Vitronics, supra*, 90 F.3d at 1582-83).

Apple argues that the prosecution history confirms that the claims are directed to confirming the license of programs running at the operating system level and not confirming the operating system itself, because, according to Apple, Ancora disavowed a broad definition of program in the patent prosecution.

"Under the doctrine of prosecution disclaimer, a patentee may limit the meaning of a claim term by making a clear and unmistakable disavowal of scope during prosecution." *Purdue Pharma L.P. v. Endo Pharmaceuticals Inc.*, 438 F.3d 1123, 1136 (Fed. Cir. 2006) (citing *Seachange Int'l, Inc. v. C-COR Inc.*, 413 F.3d 1361, 1372-73 (Fed. Cir. 2005); *Omega Eng'g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1323-26 (Fed. Cir. 2003)). A patentee may disavow the scope of the meaning of a claim term by, for example, clearly characterizing the invention in a way to try to overcome rejections based on prior art. *See, e.g., Microsoft Corp. v. Multi-Tech Sys., Inc.*, 357 F.3d 1340,

1349 (Fed. Cir. 2004) (limiting the term "transmitting" to require direct transmission over telephone line because the patentee stated during prosecution that the invention transmits over a standard telephone line, thus disclaiming transmission over a packet-switched network); *Alloc v. Int'l Trade Comm'n*, 342 F.3d 1361, 1372 (Fed. Cir. 2003) (finding the patentee expressly disavowed floor paneling systems without "play" because the applicant cited the feature during prosecution to overcome prior art); *Bell Atl. Network Servs. v. Covad Commc'ns Group, Inc.*, 262 F.3d 1258, 1273 (Fed. Cir. 2001) (limiting operation of the "transceiver" to the three stated modes because of clearly limiting statements made by the patentee to try to overcome a prior art rejection). Where the patentee has unequivocally disavowed a certain meaning to obtain the patent, the doctrine of prosecution disclaimer attaches and narrows the ordinary meaning of the claim congruent with the scope of the surrender. *Cordis Corp. v. Medtronic Ave, Inc.*, 511 F.3d 1157, 1177 *supplemented sub nom. Cordis Corp. v. Boston Scientific Corp.*, 275 F. App'x 966 (Fed. Cir. 2008) ("an applicant can make a binding disavowal of claim scope in the course of prosecuting the patent, through arguments made to distinguish prior art references.") (citing *Conoco, Inc. v. Energy & Envtl. Int'l, L.C.*, 460 F.3d 1349, 1364 (Fed. Cir. 2006); *Pharmacia & Upjohn Co. v. Mylan Pharms., Inc.*, 170 F.3d 1373, 1376 (Fed. Cir. 1999); *Litton Sys., Inc.*, 140 F.3d at 1458).

Here, the Patent Examiner rejected all Claims, including Claim 1, as being unpatentable as obvious based on the prior art, Misra *et al.*, U.S. Patent No. 6,189,146, Goldman *et al*., U.S. Patent No. 5,684,951, and Ewertz *et al*., U.S. Patent No. 5,479,639. Patentees distinguished that prior art by stating that Misra *et al*. teaches a method for enforcing software licenses that describes the step of generating a license IDs for, *inter alia*, an Operating System program but fails to teach using the BIOS of a computer to store that license ID. In distinguishing that prior art, Patentees emphasized that "[s]oftware license management applications, such as the one of the present invention [the '941 Patent], are operating system (OS) level programs," which "do not ordinarily interact or communicate" with the BIOS. (Ex. 2, ANCA 713.) Subsequently the Patent Examiner allowed the Claims and emphasized that the prior art "do not teach licensed programs running at the OS level interacting with a program verification structure stored in the BIOS to verify the program using the

verification structure and having a user act on the program according to the verification." (*Id.*, ANCA 722.) Both parties believe that this supports their construction.

Statements in the prosecution history, similar to the specification, describe the invention as an application program. This demonstrates what Patentees considered to be their invention and consequently, what the Patent Examiner would have understood Patentees' invention to be. Moreover, the Patentees specifically described their invention as an application program in order to overcome rejection based upon prior art. This rises to the level of prosecution disclaimer, and clearly, it demonstrates that Patentees and the Patent Examiner understood the invention to apply to application programs. By emphasizing that the '941 Patent is a "[s]oftware license management application[ ]," Patentees themselves narrowed the ordinary meaning of the claim term.

Further, Ancora's argument makes no logical sense; nor does the record support the broad construction it proposes. The '941 Patent teaches verification of programs which cannot function unless the operating system is already running. While those programs may then operate at the "same level" as the operating system, a "[s]oftware license management applications, such as the ['941 Patent]," cannot function if the operating system is not functional.

Based on the foregoing analysis, the Court finds that the term "program" as used in the '941 Patent means:

A set of instructions for software applications that can be executed by a computer.

D. FIFTH DISPUTED CLAIM TERM/ PHRASE – "LICENSE RECORD" (CLAIMS 1-3, 5-17)

The parties disagree as to whether the term "license record" is a record that identifies the licensed program and the number of licensed users, as Apple urges, or more broadly, information for verifying a licensed program, as Ancora contends. The parties' proposed constructions are shown below:

APPLE'S PROPOSED CONSTRUCTION	ANCORA'S PROPOSED CONSTRUCTION
record from the licensed program that identifies the licensed program and the number of licensed users	information for verifying a licensed program

The '941 Patent stores a "license record" in the BIOS area of a computer. Neither the claim nor the specification defines "license record."

Apple argues that a license record must provide data about the nature of the license for it to be a record, and therefore, the license record "identifies the licensed program and the number of licensed users." ('941 Patent 6:9-16.) Apple bases its construction on two non-limiting examples provided in the specification. The first non-limiting example identifies the following information as part of a license record: "author name, program name, and number of licensed users (for network)." (*Id.* 1:55-57.) Additionally, the "Detailed Description of a Preferred Embodiment" provides as an example of what the license record includes: "Application names [*sic*] (e.g. Lotus 123), Vendor name (Lotus inc. [*sic*]), and number of licensed copies (1 for stand alone [*sic*] usage, >1 for number of licensed users for a network application)." (*Id.* 5:29-33.) Ancora argues that these non-limiting "examples" cannot be used to limit the claim term.

The non-limiting examples provide examples of information that a license record *may* contain, but not what information a license record *must* contain. In the specification's first example, "the number of licensed users" is information included in the license record "for network" computers, which suggests that the number of licensed users is not a required part of the license record for non-network computers. Moreover, the specification provides that the contents of a license record "may include terms, identifications, specifications, or limitations related to the manufacturer of a software product, the distributor of a software product, the purchaser of a software product, a licensor, a licensee, items of computer hardware or components thereof, or to other terms and conditions related to the aforesaid." (*Id.* 6:11-17.) Nothing in the non-limiting examples contained in the specification requires that a license record identify the number of licensed users, as Apple urges.

Apple argues that Ancora's proposed construction would render claim language meaningless because Ancora's construction fails to distinguish between "license record" and "license information" by construing license record to mean *any* information verifying a licensed program without any requirement that the information indicate anything about the nature of the license. In so doing, Apple attempts to use dependent claim 15, as well as independent claim 18 and claim 19,

which is dependent on claim 18 (all three referring to "license information") to limit independent claim 1 (referring to "license record"). While Apple is correct that the licensed record must contain information about the license for it to be a record, the specification does not support Apple's conclusion regarding the specific nature of the contents of a license record.

Based on the foregoing analysis, the Court concludes that Apple's construction is too narrow, while Ancora's construction is too broad. The Court provides the following definition for "license record":

A record from a licensed program with information for verifying that licensed program.

E. SIXTH DISPUTED CLAIM TERM/ PHRASE – "VERIFYING THE PROGRAM USING AT LEAST THE VERIFICATION STRUCTURE" (CLAIMS 1-3, 5-17)

The parties dispute whether verifying that a program is licensed requires a comparison between the license record extracted from the program to the license record in the verification structure, or whether it can cover any method of verification, using at least the verification structure. The parties' proposed constructions are shown below:

APPLE'S PROPOSED CONSTRUCTION	ANCORA'S PROPOSED CONSTRUCTION
confirming whether the program is licensed by comparing the license record extracted from the program to the license record in the verification structure	confirming whether a program is licensed using at least the verification structure

This phrase, which appears in Claim 1 of the '941 Patent is not defined in either the Claim or the specification.

For simplicity, the Court will divide this Claim phrase, and the parties' constructions thereof, into three component parts: (1) "verifying the program"; (2) "using at least"; and (3) "the verification structure." The parties' proposed constructions of the first and third parts of the Claim phrase are virtually identical. Apple and Ancora's propose construing "verifying the program" as "confirming whether *the* program is licensed" and "confirming whether *a* program is licensed," respectively. Both parties agree that "the verification structure" means "the verification structure." Thus, the dispute really is over the meaning of "using at least." That is, whether "using at least"

means "using at least," as Ancora urges, or whether "using at least" means "by comparing the license record extracted from the program to the license record in," as Apple urges.

Ancora argues that a comparison between the license record extracted from the program and the license record in the verification structure is not required, while Apple argues verification is not possible without such a comparison. Apple argues that the specification describes the verification process, which includes a comparison between the license record extracted from the program and the license record in the verification structure. Ancora notes that the specification describes multiple examples of techniques to verify a program, including verification by determining whether a particular program is "compatible" with the license record. ('941 Patent, 3:38-41.) According to Ancora, "comparison" is not required for this compatibility verification.

The Court cannot find any limitation in the Claim or specification that verifying a program requires a "comparison." Indeed, Apple's proposed construction renders the claim language "at least" meaningless. Based on the foregoing, the Court will adopt Ancora's proposed construction of the disputed claim phrase:

Confirming whether a program is licensed using at least the verification structure.

F. SEVENTH DISPUTED CLAIM TERM/ PHRASE – ALL ASSERTED CLAIMS

The parties dispute whether the claimed steps must be performed in a specific order to infringe on the '941 Patent. The parties' proposed constructions are shown below:

APPLE'S PROPOSED CONSTRUCTION	ANCORA'S PROPOSED CONSTRUCTION
The steps in each asserted claim must be performed in the order recited.	Unless the steps of a method actually recite an order, the claim is not limited to performance of the steps in the order recited.

"Unless the steps of a method actually recite an order, the steps are not ordinarily construed to require one. However, such a result can ensue when the method steps implicitly require that they be performed in the order written." *Altiris, Inc. v. Symantec Corp.*, 318 F.3d 1363, 1369-70 (Fed. Cir. 2003) (quoting *Interactive Gift Exp., Inc. v. Compuserve Inc.*, 256 F.3d 1323, 1342 (Fed. Cir. 2001)). "First, we look to the claim language to determine if, as a matter of logic or grammar, they must be performed in the order written. … If not, we next look to the rest of the specification to

determine whether it 'directly or implicitly requires such a narrow construction.' If not, the sequence in which such steps are written is not a requirement." *Id.*

Apple argues that there is only one possible sequence to perform the steps in Claim 1, the order recited. "In this case, nothing in the claim or the specification directly or implicitly requires such a narrow construction." *Id.* As Ancora points out, Apple's proposed construction is contrary to the express teachings of the '941 Patent. In the Summary of the Invention, the '941 Patent teaches an embodiment of the invention in which the first step to be performed is setting up the verification structure. ('941 Patent 1:59-65.) In contrast to the express teachings of the '941 Patent, Apple argues that the first step is to select a program and the second step is setting up the verification structure. Nothing in the '941 Patent directly or implicitly requires that the steps be performed in the order recited.

Based on the foregoing analysis, the Court will not construe the claim to require the steps be performed in the order written.

IV. CONCLUSION

For the reasons set forth above, the Court provides the following construction of the disputed claim terms/phrases:

DISPUTED CLAIM TERM/PHRASE	CONSTRUCTION
volatile memory	Memory whose data is not maintained when the power is removed.
non-volatile memory	Memory whose data is maintained when the power is removed.
BIOS	An acronym for Basic Input/Output System. It is the set of essential startup operations that run when a computer is turned on, which tests hardware, starts the operating system, and supports the transfer of data among hardware devices.
program	A set of instructions for software applications that can be executed by a computer.
license record	A record from a licensed program with information for verifying that licensed program.

verifying the program using at least the verification structure	Confirming whether a program is licensed using at least the verification structure.
All Asserted Claims	The steps of the Claim do not need to be performed in the order recited.

IT IS SO ORDERED.

Date: December 31, 2012

YVONNE GONZALEZ ROGERS
UNITED STATES DISTRICT COURT JUDGE

United States District Court
Northern District of California