IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
AUSTIN DIVISION

ANCORA TECHNOLOGIES, INC., Plaintiff, v. LG ELECTRONICS INC. and LG ELECTRONICS U.S.A., INC., Defendants.	CIVIL ACTION NO. 1:20-CV-0034 JURY TRIAL DEMANDED
ANCORA TECHNOLOGIES, INC., Plaintiff, v. SAMSUNG ELECTRONICS CO., LTD., and SAMSUNG ELECTRONICS AMERICA, INC., Defendants.	CIVIL ACTION NO. 1:20-CV-0034 JURY TRIAL DEMANDED

PLAINTIFF'S OPENING CLAIM CONSTRUCTION BRIEF

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

Cases

Statutes

I. INTRODUCTION

This case concerns a single patent: U.S. Patent No. 6,411,941. *See* Ex. 1. The '941 Patent has a lengthy history. It was invented in the late 1990s by Ancora's president, Mr. Miki Mullor and a colleague and was described by the Patent and Trademark Office during prosecution as proceeding against the conventional wisdom in the art to do something "the closest prior art, singly or collectively," never contemplated: "using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS." Ex. 2 (Examiner's Reasons for Allowance) at ANCORA_451.

Moreover, since it issued, the patent has been examined repeatedly. Its claims have been construed by multiple courts, including the Federal Circuit. *Ancora Techs., Inc. v. Apple Inc.*, 2012 WL 6738761 (N.D. Cal. Dec. 31, 2012) ("*Apple I*"); *Ancora Techs., Inc. v. Apple, Inc.*, 744 F.3d 732 (Fed. Cir. 2014) ("*Apple II*"). And its validity has been confirmed repeatedly, including most recently by the Federal Circuit in *Ancora Techs., Inc. v. HTC Am., Inc.* ("*HTC*"), which held that the asserted claims satisfied 35 U.S.C. § 101 as a matter of law. 908 F.3d 1343 (Fed. Cir. 2018).

In asserting infringement in this matter, Ancora has relied on the entirety of these courts' guidance. In fact, with limited exceptions, Ancora simply asks this Court to adopt constructions already reached by prior courts. In contrast, Defendants seek to avoid infringement by offering a raft of new constructions—constructions that conflict not just with the claims and specification, but the prior holdings and constructions already provided by prior courts.

II. BACKGROUND

The '941 Patent relates to a specific technique for "identifying and restricting of an unauthorized software program's operation." '941 Patent at 1:6-8.

Specifically, before the '941 Patent, there were two basic (and sub-optimal) methods of verifying and restricting the operation of a program. One involved "software-based methods" that "require[d] writing a license signature on the computer's hard drive." *HTC*, 908 F.3d at 1344. A key

"flaw in those methods," however, "is that such a signature can be changed by hackers without damaging other aspects of computer functionality." *Id.* (citing '941 Patent at 1:19-26). Hardware-based methods also existed, but "require[d] inserting a dongle into a computer port to authenticate the software authorization." *Id.* (citing '941 Patent at 1:27-32). As a result, those "methods are costly, inconvenient, and not suitable for software sold and downloaded over the internet." *Id.*

The '941 Patent improved over these prior art techniques by "using the memory space associated with the computer's basic input/output system (BIOS),[1] rather than other memory space, to store appropriately encrypted license information to be used in the verification process." *Apple II*, 744 F.3d at 733 (citing '941 Patent at 1:46-2:5, 4:45-48, and 5:19-24). Such BIOS memory space was and "is typically used for storing programs that assist in the start-up of a computer." *HTC*, 908 F.3d at 1345. Prior to the '941 invention, however, it was not contemplated that operating system ("OS") level programs could interact with the BIOS at all—much less "us[e] an agent to setup a verification structure in the erasable non-volatile memory area of the BIOS."[2] Ex. 2 (Reasons for Allowance) at ANCORA_451; *HTC*, 908 F.3d at 1348-49 (stating that "[t]he claimed method here specifically identifies how that functionality improvement is effectuated in an assertedly unexpected way: a structure containing a license record is stored in a particular, modifiable, non-volatile portion of the computer's BIOS, and the structure in that memory location is used for verification").

As the Federal Circuit explained, using the BIOS in this unexpected manner "improves computer security, . . . because successfully hacking BIOS memory (*i.e.*, altering it without rendering the computer inoperable) is much harder than hacking the memory used by the prior art to store license-verification information." *HTC*, 908 F.3d at 1345.

[1] The BIOS "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." *Apple I*, 2012 WL 6738761, at *7.

III. AGREED TERMS

Term	Agreed Construction
"a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area" (Claim 1 Preamble)	The parties agree that this portion of the preamble is limiting.
"non-volatile memory" (Claims 1, 3, 7, 9, 12, 16)	"memory whose data is maintained when the power is removed"
"the verification" (Claim 1)	"verifying the program using at least the verification structure from the erasable nonvolatile memory of the BIOS"
"using the key" (Claim 8)	"using a pseudo-unique key"

IV. DISPUTED TERMS

In this matter, Ancora has asserted that Defendants infringed Claims 1-3, 6-14, and 16 of the '941 Patent. From these claims, 13 terms and phrases have been identified for construction.

For the Court's convenience, Ancora addresses these 13 terms and phrases in the order in which they first appear in the asserted claims. Further, because the majority of these terms and phrases first appear in Claim 1, Ancora includes that claim in full below—underlining the disputed phrases and double-underlining/bolding the disputed terms:

> 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.

[2] All emphases in this brief have been added unless otherwise specified.

1. "license" (Claim 1 - Preamble) / "license record" (Claims 1, 3, 6, 8-9, 14, 16)

Term	Ancora's Construction	Defendants' Construction
"license"	The portion of the preamble reciting "A method of restricting software operation within a license…" is non-limiting, and the term "license" thus does not need to be construed.	The preamble of the claim is limiting, and "license" means "a legal contract between a software provider and a user that specifies the rights of the user to use, distribute or resell the software"
"license record"	"license record" means "a record associated with a licensed program with information for verifying that licensed program"	"license record" means "data indicating that a program is licensed"

The terms "license" and "license record" appear in different places in the claims. "License" appears only in the preamble of Claim 1. "License record" appears in the body of Claim 1 and in various dependent claims. Because Defendants contend that "license" and "license record" have different meanings, Ancora addresses these terms separately below.

i) The Court Does Not Need to Construe the Term "License."

As noted, the term "license" appears only in the preamble to Claim 1. As such, it does not need to be construed unless the portion of the preamble in which it appears is limiting. It is not.

A preamble is generally understood to be non-limiting. Thus, to show that "license" should be construed, Defendants must show that the portion of the preamble in which it resides "recites essential structure or steps, or if it is 'necessary to give life, meaning, and vitality' to the claim." *TomTom, Inc. v. Adolph*, 790 F.3d 1315, 1323 (Fed. Cir. 2015) (citation omitted).

In undertaking this analysis, two points are key. First, "a preamble is not limiting 'where a patentee defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention.'" *Id.* (same). Second, the fact that a portion of the preamble is limiting does not require that the entire preamble to be limiting—particularly if the remaining portion "is language stating a purpose or intended use." *Id.* at 1323-24.

TomTom controls here. As stated above, the parties agree that the portion of the preamble of Claim 1 underlined below is limiting:

> 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 . . . :

Unlike the underlined text, however, the portion of the preamble that includes the "license" term does not provide an antecedent basis for any later term. Nor does it "recite[] essential structure." *Id.* at 1323. Rather, like the preamble at issue in *TomTom*, it "employs the standard pattern of such [non-limiting] language: the words 'a method for a purpose or intended use comprising,' followed by the body of the claim, in which the claim limitations describing the invention are recited." *Id.*

In fact, the Claim 1 preamble is nearly identical to the one at issue in *TomTom*, which recited "[a] method for generating and updating data for use in a destination tracking system of at least one mobile unit comprising . . . ." *Id.* at 1322. The Claim 1 preamble similarly starts by stating an intended use ("A method of restricting software operation within a license for use with [a computer]") before reciting structure included in the body of the claim ("a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area").

Like the preamble in *TomTom*, the portion of the Claim 1 preamble describing an intended use thus is non-limiting. *See id.*; *E-Pass Techs., Inc. v. 3Com Corp.*, 343 F.3d 1364, 1370 (Fed. Cir. 2003) ("An invention may possess a number of advantages or purposes, and there is no requirement that every claim directed to that invention be limited to encompass all of them."). Indeed, the fact that "license" never appears alone in the body of the claim reinforces that the first part of the preamble "does not provide an antecedent basis for any of the claims" and thus is non-limiting. *Id.*

Nevertheless, if the Court does construe the term, it should reject Defendants' construction. First, nothing in the intrinsic record supports it. The specification never references the words "contract," "agreement," "right," "software license," and "sell/sale." And limiting the term to a

"legal contract between a software provider and a user that specifies the rights of the user to use, distribute or resell the software" is inconsistent with the understanding that the invention was intended not only to combat the "proliferation of illegally copied software," but also to safeguard against "attack[s] at the hands of skilled system's programmers (e.g. 'hackers')." '941 Patent at 1:21-24. Defendants' construction also is inconsistent with the PTAB's express finding (later echoed by the Federal Circuit in *HTC*) that the invention is not directed to a business method, but instead "solves a technical problem using a technical solution." *HTC Corp. v. Ancora Techs. Inc.*, 2017 WL 6032605, at *1 (Patent Tr. & App. Bd. Dec. 1, 2017); *accord HTC*, 908 F.3d at 1348-49.

Instead, a person of ordinary skill in the art ("POSITA") would understand "license" to refer simply to "authorization or verification to run." As the Federal Circuit recognized, "[t]he patent describes and claims methods of limiting a computer's running of software not authorized for that computer to run." *HTC*, 908 F.3d at 1344.

The specification supports this understanding. It states that the "invention relates to a method and system of identifying and restricting an unauthorized software program's operation." '941 Patent at 1:6-8. It also describes how, in an embodiment, "there commences an initial license establishment procedure, where a verification structure is set in the BIOS so as to indicate that the specified program is licensed to run on the specified computer." *Id.* at 1:59-62. The '941 Patent thus describes how "the process of verifying a license" can include comparing encrypted records such that:

> In the case of match, the program is verified to run on the computer. If on the other hand the sought encrypted data record is not found . . . , this means that the program under question is not properly licensed and appropriate application define action is invoked (e.g. informing to the user on the unlicensed status, halting the operation of the program under question etc.).

Id. at 2:14-26.

Because the term "license" appears only in the non-limiting preamble, it does not need to be

construed. If construed, however, it simply means "authorization or verification to run."

ii) The Court Should Adopt Plaintiff's "License Record" Construction.

Turning to the "license record" term, Ancora's proposed construction mirrors the construction of this term in *Apple I*—varying only by replacing the word "from" with "associated with." *Compare* 2012 WL 6738761, at *12 (construing the term to mean "a record from a licensed program with information for verifying that licensed program"), *with* '941 Patent at 1:53-57 ("[E]ach application program that is to be licensed to run on the specified computer[] is associated with a license record . . . ."). In contrast, Defendants ask the Court to discard the prior construction and construe license record to mean "data indicating that a program is licensed."

Defendants' construction is incorrect and should be rejected for at least two reasons. First, during the meet and confer process, Defendants stated that they would be willing to agree to the construction "a record associated with a program with information for verifying that the program is licensed"—demonstrating that the crux of the parties' dispute centers on whether the license record must contain "information for verifying th[e] licensed program" or "information for verifying that a program is licensed." And as Ancora explained to Defendants, their "is licensed" construction distorts the specification by suggesting that the program from which the license record is derived must have a specific legal or contractual status. As explained above at page 6, the specification makes clear that the '941 Patent uses the word "licensed" to refer to authorization or verification to run—a concept that can include a program's legal or contractual status, but is not limited to it.

This understanding is squarely supported by the specification, which the *Apple I* court recognized teaches that "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.'" 2012 WL 6738761,

at *11 (quoting '941 Patent at 6:11-17). Thus, for example, a license record could comprise only information sufficient to identify a program's manufacturer—provided that it allows the program to be verified to run on the computer. *See* '941 Patent at 5:27-33 ("The volatile memory accommodates a license program (16) having license record fields (13-15) . . . . By way of example said fields stand for Application names . . . , Vendor name . . . , and number of licensed copies . . . .").

Second, the proposed construction of "data indicating that a program is licensed" also is incorrect because it suggests that the data must reflect information about the program's status. The specification states, however, that the licensed record simply needs to allow for verification; it does not need to indicate itself whether a program is verified or authorized to run. '941 Patent at 2:5-10 ("The actual format of the license may include a string of terms that correspond to a license registration entry (e.g. lookup table entry or entries) at a license registration bureau . . . .").

2. "volatile memory" (Claims 1, 6, 9, 11)

Term	Ancora's Construction	Defendants' Construction
"volatile memory"	"memory whose data is not maintained or becomes inaccessible when the power is removed"	"memory whose data is not maintained when the power is removed"

The parties agree that volatile memory means memory whose data is not maintained when the power is removed. They disagree only as to whether the term also includes memory whose data "becomes inaccessible" after the power is removed. As the Federal Circuit already has held, it does.

In *Apple II*, the Federal Circuit held that, "as a general matter, '[t]o one of ordinary skill in the art, a volatile memory is memory whose data is not maintained when the power is removed . . . .'" 744 F.3d at 737 (quoting *Apple I*, 2012 WL 6738761, at *4). The Federal Circuit then went on to identify an important exception to that "general" understanding—explaining that the '941 Patent's disclosure "of a hard disk as 'volatile' memory" showed that the '941 Patent contemplated that memory typically thought of as non-volatile also fell within the scope of the term

"volatile memory" to the extent such memory was being used in a volatile manner. *Id*. at 737-38.

The Federal Circuit even identified an example of when non-volatile memory could be understood to be operating in a volatile manner, explaining:

> it is well known that a computer's hard disk is routinely used as 'virtual' memory to provide temporary storage when there is insufficient RAM to complete an operation, in which case (it is undisputed) the data become inaccessible through the usual means once power is removed (even if the data can still be found on the hard disk by more sophisticated means).

Id. at 738 (internal citations omitted).

Thus, unlike Defendants' construction, Plaintiff's construction adheres to the entirety of the Federal Circuit's guidance and recognizes that "volatile memory" includes both (1) memory whose data is not maintained when the power is removed and (2) memory whose data becomes inaccessible when the power is removed such that it is recoverable only through "sophisticated means." *Id.*; *see* Ex. 7 (Jestice Decl.) at ¶¶ 10-12. Accordingly, Ancora's construction should be adopted.

3. "BIOS" (Claims 1, 3, 7, 9, 12, 16)

Term	Ancora's Construction	Defendants' Construction
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 test hardware, starts the operating system, and support the transfer of data among hardware devices."	"An acronym for Basic Input / Output System. It is the set of essential startup operations stored in ROM that run automatically when a computer is turned on, which test hardware, starts the operating system, and support the transfer of data among hardware devices."

Ancora's construction of BIOS is taken verbatim from the district court's claim construction order in *Apple I*, which is entitled to deference. *Kinetic Concepts, Inc. v. Wake Forest Univ. Health Scis.*, 2013 WL 6164592, at *3 (W.D. Tex. Nov. 25, 2013) (stating that the court "gives reasoned deference to the decisions of other district courts that have construed these claims"). Departing from that construction, Defendants seek to insert two additional limitations: (1) that BIOS be "stored in ROM" and (2) that BIOS "run automatically." Neither of these new limitations is proper.

i) Defendants' "Stored in ROM" Limitation Conflicts with the Patent.

Nothing in the specification requires that BIOS be stored entirely in a particular memory medium—let alone ROM (read only memory). To the contrary, the '941 Patent states that, at most, the BIOS may include a "ROM section" or "ROM portion." '941 Patent at 1:46-48; *id*. at 2:12-19 (referencing a "ROM portion of the BIOS"); *id*. at 2:29-32 (same), 2:36-39 (same), 4:59-61 (same). It also states, however, that other sections of BIOS may exist, including an E^2PROM section. *Id*. at 1:65-2:1. Further, and to be clear, '941 Patent explains that its references to both sections are entirely exemplary and "non-limiting." *Id*. at 4:49-54 ("According to one, non-limiting, preferred embodiment of the present invention, the first non-volatile memory area is a ROM section of a BIOS; the second non-volatile memory area is a E^2PROM section of a BIOS; and the volatile memory is a RAM e.g. hard disk and/or internal memory of the computer.").

Nothing more is needed to reject Defendants' ROM limitation. *Broadcom Corp. v. Emulex Corp.*, 732 F.3d 1325, 1333 (Fed. Cir. 2013) ("[A]n interpretation which excludes a [disclosed] embodiment from the scope of the claim is rarely, if ever, correct." (brackets in original, citation and quotation marks omitted)). Nevertheless, more exists.

The '941 Patent also describes a preferred embodiment without limiting it to including ROM or any other specific non-volatile memory type; the patent simply states that it includes generic "non-volatile memory media used as a BIOS." '941 Patent at 4:45-49. Moreover, the patent also contemplates embodiments that "hav[e] only one non-volatile memory." *Id.* at 3:24-28. Were such memory comprised entirely of ROM, it would not be possible to "[s]et up a verification structure in the erasable, non-volatile memory of the BIOS" as Claim 1 requires. *See Apple II*, 744 F.3d at 733 (noting "the contents of the BIOS memory space may be modified"); Ex. 7 (Jestice Decl.) at ¶ 7.

The prosecution history also does not limit BIOS to being stored in ROM. To the contrary, the Examiner specifically observed that "flash memory can be used as a computer BIOS" such that

"a computer BIOS would not contain an EEPROM and/or ROM section." Ex. 4 (Examiner Jan. 15, 2001 Non-Final Rejection) at ANCORA_423. This further demonstrates a POSITA would not understand a BIOS to be stored only "in ROM." *Salazar v. Procter & Gamble Co.*, 414 F.3d 1342, 1347 (Fed. Cir. 2005) (explaining that examiner statements may evidence "how one of skill in the art understood the term at the time the application was filed").

For each of the above reasons, Defendants' "ROM" limitation should be rejected.

ii) There Is No Support in the Patent for an "Automatically" Limitation.

Defendants' "automatically" limitation also has no support in the intrinsic record. Rather, the lone basis for this limitation appears to be a statement by Ian Jestice (one of Ancora's experts) in a declaration submitted in *Apple I* that it was generally understood that, "[a]t the time the computer is first started, BIOS automatically performs the initial steps necessary to boot the operating system." Ex. 8 (*Apple I* Decl.) at ¶ 10; *see also id*. at ¶ 11-12. As is clear from this statement, however, the fact that BIOS can operate automatically at startup does not mean that it always must do so.

Mr. Jestice has confirmed this point. He explained that a POSITA would understand that—in addition to running automatically at startup—a user can prompt BIOS operations to run, including taking steps as simple as using the operating system to restart the computer (which would then cause the BIOS to run upon restart). Ex. 7 (Jestice Decl.) at ¶ 9. Mr. Jestice also explained that a POSITA would understand that a user can use special commands to interrupt normal BIOS operations or specify different parameters. *Id*. at ¶ 9. In short, Mr. Jestice confirmed that, while BIOS operations can run automatically, they are not limited to doing so as Defendants' construction requires.

4. "non-volatile memory of the BIOS" (Claim 1)

Term	Ancora's Construction	Defendants' Construction
"non-volatile memory of the BIOS"	This term does not require separate construction	"memory of the BIOS" means "a memory that: (i) stores the BIOS; (ii) is not recognized by an operating

		system as a storage device; and (iii) does not have a file system"

The parties agree that the term "non-volatile memory" means "memory whose data is maintained when the power is removed." Further, as reflected above, the term "BIOS" already has been identified for construction. Defendants nonetheless ask the Court to separately construe the phrase "memory of the BIOS" even though it always appears as part of the phrase "non-volatile memory of the BIOS." *Apple I*, 2012 WL 6738761, at *5 ("The claims of the '941 Patent refer[] to the 'non-volatile memory area of the BIOS' of a computer fourteen times (and refer[] to the 'non-volatile memory area of ***a*** BIOS' once). The Court should reject that request.

The joinder of these two terms does nothing to change their meaning. To the contrary, consistent with the plain meaning of the word "of," the Federal Circuit has described the "non-volatile memory of [a/the] BIOS" as "memory space associated with the computer's basic input/output system (BIOS), rather than other memory space." *Apple II*, 744 F.3d at 733 (pointing to various disclosures in the '941 Patent); *cf. Facebook, Inc. v. Pragmatus AV, LLC*, 582 F. App'x 864, 867 (Fed. Cir. 2014) ("The plain meaning of 'of [the] device' is that the 'addressing information' is specific to the device."). In short, it has demonstrated that no further construction is warranted.

Further, even setting aside that the terms at issue already are being construed, no support exists for Defendants' construction. First, Defendants' construction requires that the "non-volatile memory area" be the memory that "(i) stores the BIOS." As noted above, however, the Federal Circuit already indicated that the memory area only needs to be "associated with" the BIOS. *Apple II*, 744 F.3d at 733. Moreover, as noted above, Defendants' proposed "BIOS" construction requires that this same memory also be ROM (read-only memory). This conflicts with Claim 1's express requirement that an agent "set up a verification structure in the erasable, non-volatile memory of the BIOS," which cannot occur if the memory is comprised only of ROM. Ex. 7 (Jestice Decl.) at ¶ 7.

Second, nothing in the intrinsic record supports the negative limitations Defendants ask the Court to impose—namely, that BIOS (ii) not be recognized by an operating system (OS) as a storage device, and (iii) not have a file system. Such "negative limitations" cannot be imposed absent an express disclaimer of claim scope. *Omega Eng'g, Inc, v. Raytek Corp.*, 334 F.3d 1314, 1323 (Fed. Cir. 2003) (rejecting "negative limitation" where there was no "express disclaimer or independent lexicography in the written description"); *Linear Tech. Corp. v. ITC*, 566 F.3d 1049, 1060 (Fed. Cir. 2009) (same). Here, the specification contains no mention of either limitation.

The prosecution history likewise does not contain the "express disclaimer" necessary to justify Defendants' construction. In support of their "(ii) not be recognized … as a storage device" limitation, Defendants appear to rely solely on the patentee's explanation during prosecution that "[a]n ordinary skilled artisan would not consider the BIOS as a storage medium to preserve application data" because:

> [a]n ordinary person skilled in the art makes use of OS features to write date to storage mediums. There is no OS support whatsoever to write data to the system BIOS. Therefore, an ordinary person skilled in the art would not consider the BIOS as a possible storage medium.

Ex. 3 (February 5, 2002 Statement in Support of Amendment) at ANCORA_438.

This statement does nothing to support Defendants' position because it does not describe the invention; it describes the shortcoming present in the prior art. And as the patentee stated immediately before the above quoted text, one of the core improvements of the '941 Patent is that it overcame this prior art shortcoming by enabling OS-level programs to write (i.e., store) data to the BIOS. *Id*. ("[T]he present invention proceeds against conventional wisdom in the art. Using BIOS to store application data such as that stored in Misra's local cache for licenses is not obvious."); *Apple II*, 744 F.3d at 735-36 (citing this same history and stating that "[t]he applicants explained that their invention differed from the prior art in that it both operated as an application running through an

operating system and used the BIOS level for data storage and retrieval").

Indeed, the Examiner emphasized that he was allowing the '941 Patent over the prior art precisely because "the closest prior art systems, singly or collectively, do not teach licensed programs running at the OS level" interacting with BIOS, and "the present invention overcomes this difficulty by using an agent to setup a verification structure in the erasable non-volatile memory area of the BIOS." Ex. 2 (Examiner's Reason for Allowance) at ANCORA_451.

As is clear from the above, the statement at issue thus does not purport to define the characteristics of BIOS post-invention (or make any disclaimer of scope), and certainly does not make the "clear and unmistakable disavowal of scope" required to show disclaimer. *Purdue Pharma L.P. v. Endo Pharmaceuticals Inc.*, 438 F.3d 1123, 1136 (Fed. Cir. 2006). Rather, at most, this passage highlights one of the improvements taught by the '941 Patent.

For the same reasons, there is no support for Defendants' negative construction that the "BIOS" recited in the '941 Patent must "(iii) . . . not have a file system." In support of this limitation, Defendants appear to rely entirely on the below statement in the prosecution history:

> [N]o file system is associated with BIOS. Every writable device connected to the PC is associated with an OS file system to arrange and manage data structures. An example for such a file system would be FAT, FAT32, N1FS, HPFS, etc. that suggests writing data to the writable device. No such file system is associated with the BIOS. This is further evidence that OS level application programmers would not consider the BIOS as a storage medium.

Ex. 3 (February 5, 2002 Statement in Support of Amendment) at ANCORA_438. Like the prior statement to which Defendants point, this passage does not describe limitations present in the '941 invention. It describes the shortcoming present in prior art systems—shortcomings the '941 invention was invented to overcome. As such, this statement cannot justify limiting the '941 invention to the very shortcomings it went "against conventional wisdom in the art" to surmount. *Id.*

In sum, the Court should reject Defendants' limitations and decline to construe this phrase.

5. "program" (Claims 1, 3, 6, 9-10, 14, 16)

Term	Ancora's Construction	Defendants' Construction
"program"	"a set of instructions for a computer"	"a set of instructions that can be executed by a computer"

The Federal Circuit construed the term "program" in *Apple II*—holding that "a program is merely a 'set of instructions' for a computer" and "[t]hat clear meaning governs here, we conclude, because there is nothing sufficient to displace it." *Apple II*, 744 F.3d at 735 (citation omitted). Ancora asks this Court to adhere to the Federal Circuit's construction. In contrast, Defendants urge the Court to construe the term in the manner that Ancora had proposed in the *Apple* case before it obtained the benefit of the Federal Circuit's guidance. No support exists for that position. Not only do "[e]arlier Federal Circuit [claim construction] decisions have *stare decisis* effect upon this court," *Kinetic Concepts, Inc.*, 2013 WL 6164592, at *3, but Defendants have offered no justification for their construction other than to point to the fact that Ancora previously proposed it.

6. "selecting a program residing in the volatile memory" (Claims 1, 6)

Term	Ancora's Construction	Defendants' Construction
"selecting a program residing in the volatile memory"	plain and ordinary meaning	"running a program in the volatile memory"

Defendants next ask the Court to re-write the phrase "selecting a program residing in the volatile memory" by changing the word "selecting" to "running" and deleting the word "residing." No support exists for this re-write. *K–2 Corp. v. Salomon S.A.*, 191 F.3d 1356, 1364 (Fed. Cir. 1999) ("Courts do not rewrite claims; instead, we give effect to the terms chosen by the patentee.").

First, the words "selecting" and "residing" are simple English words that, as a general matter, do not require construction. *See Chef Am., Inc. v. Lamb Weston, Inc.*, 358 F.3d 1371, 1373 (Fed. Cir. 2004) ("[O]rdinary, simple English words whose meaning is clear and unquestionable . . . mean exactly what they say."); *Nokia Sols. & Networks US LLC v. Huawei Techs. Co.*, 2017 WL 2226413,

at *26 (E.D. Tex. May 19, 2017) (construing the term "[select/selecting] a communication resource . . ." to have its plain and ordinary meaning (alteration in original)); *Prowess, Inc. v. RaySearch Labs., AB*, 953 F. Supp. 2d 638, 672 (D. Md. 2013) (same).

Second, the patent itself demonstrates that, at a minimum, "selecting" is not synonymous with "running." In describing embodiments of the "selecting" step, the specification never refers to the "running" of a program—a concept it references in regard to other steps. *Compare* '941 Patent at 2:66-67, 6:7-17 (describing the "selecting"), *with id.* at 1:53-57, 1:59-62, 2:19-26, 2:27-32, 2:36-39 (describing how the program can be "run" after it is verified); *see also id*. at 3:57, 4:1 (referencing a "selected program"). Instead, the specification explains that "[s]electing (17) a program" can include "the step of: establishing a licensed-software-program in the volatile memory of the computer wherein the licensed-software program includes contents used to form a license-record." *Id*. at 6:7-10; *accord id.* at Claim 6 ("A method according to claim 1 wherein selecting a program includes the steps of establishing a licensed software-program in the volatile memory of the computer . . . .").

Further, the specification teaches that, in at least one embodiment, this "establishing" (or "loading") of the program in the volatile memory, as well as its verification, occurs before the selected program can run. *Id*. at 2:10-26. As the '941 Patent explains:

> [W]hen a program is loaded into the memory of the computer, a so-called license verifier application, that is a priori running in the computer, accesses the program under question, retrieves therefrom the license record, encrypts the record utilizing the specified unique key (as retrieved from the ROM section of the BIOS) and compares the so encrypted record to the encrypted records that reside in the E^2PROM. In the case of match, the program is verified to run on the computer.

Id. Indeed, this passage alone demonstrates that the patentee understood what running a program meant and intended that "selecting a program" confer a different meaning.

In short, Defendants' construction is inconsistent with the primary objective of at least some

of the patent's preferred embodiments: to require that a program be verified before it is allowed to run. '941 Patent at 2:19-20 (describing the verification process and stating that, "[i]n the case of match, the program is verified to run on the computer"). Defendants' construction thus would negate some of the invention's embodiments—violating a cardinal claim construction rule.

Moreover, when asked during the meet-and-confer process to explain the basis for this construction, Defendants did not point to the patent or prosecution history. Instead, they pointed solely to a construction that Ancora had proposed in 2009 in a prior case. But that that prior proposal is not controlling here. Not only was Ancora's construction never adopted by the district court, but Ancora's briefing shows its core position is unchanged. As Ancora explained in its brief in that case, its construction was intended to convey only that the limitation would be satisfied if a single program was "loaded into memory for execution." Ex. 15 at 16. In contrast, Defendants in this case have made clear that they believe that "running" requires that the program actually "be executing."

In sum, Ancora never agreed with the construction Defendants now urge, which is inconsistent with the patent's express disclosures and should be rejected.

7. "using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS" (Claims 1, 3, 7, 14)

Term	Ancora's Construction	Defendants' Construction
"using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS"	plain and ordinary meaning "agent" means "a software program or routine"	This limitation is a means plus function limitation governed by pre-AIA 35 U.S.C. § 112 ¶ 6. **Function**: "set up a verification structure in the erasable, non-volatile memory of the BIOS" **Structure**: Algorithm found at 6:18-28; if not, indefinite due to a lack of corresponding structure.

The crux of the parties' dispute regarding this term is whether "agent" is a nonce word such that 35 U.S.C. § 112 ¶ 6 applies. It is not. As demonstrated by both the intrinsic and extrinsic evidence in this case, "agent" is well understood to refer to a "software program or routine" that

connotes structure. As a result, § 112 ¶ 6 does not apply.

"When a claim term lacks the word 'means,'" there is a presumption that § 112 ¶ 6 does not apply. *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1348 (Fed. Cir. 2015) (*en banc*). A party seeking to overcome that presumption must supply "evidentiary support for [its] position." *Zeroclick, LLC v. Apple Inc.*, 891 F.3d 1003, 1007–08 (Fed. Cir. 2018). Specifically, it must "demonstrate[] that the words of the claim are not understood by persons of ordinary skill in the art to have a sufficiently definite meaning as the name for structure." *SecurityProfiling, LLC v. Trend Micro Am., Inc.*, 2018 WL 4585279, at *3 (N.D. Tex. Sept. 25, 2018) (emphasis in original). If on the other hand, "the intrinsic record" or "extrinsic evidence" show that the words at issue refer to particular structure, the presumption stands, and § 112 ¶ 6 does not apply. *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 469 F.3d 1005, 1024 (Fed. Cir. 2006).

Here, Defendants cannot overcome the presumption. A POSITA would understand (as the Examiner did) that an "agent" is "a software program or routine" and it is understood that software provides structure—particularly when, as here, the claims specify what this software does ("set up a verification structure in the erasable, non-volatile memory of the BIOS"), and the patent provides examples of how it accomplishes the specified operation ("using E^2PROM manipulation commands" to store, add, and modify data in the "erasable, non-volatile memory of the BIOS").

i) "Agent" Means "a Software Program or Routine"

Defendants cannot credibly dispute that "agent" means "a software program or routine." The term was added during prosecution specifically to encapsulate the point that "the closest prior art systems, singly or collectively, do not teach licensed programs running at the OS level interacting with a program verification structure stored in BIOS" and that "the present invention overcomes this difficulty by using an agent to setup a verification structure in the erasable non-volatile memory area of the BIOS." Ex. 2 (Reasons for Allowance) at ANCORA_451; Ex. 6 (Examiner Interview

Summary) at ANCORA_365.

Indeed, the Examiner repeatedly acknowledged his understanding that the word "agent" refers to such programs. Ex. 4 at ANCORA_425-28 (stating that "Misra et al. also teach encryption keys and programs ('agent') used in the license collation process that belong to various parties."); *see also id.* at 200 ("Smith et al. teach a system for distributing, registering and purchasing software over a network using an agent program embedded in each software application.").

The extrinsic evidence demonstrates the same understanding. For example, Mr. Jestice has confirmed that the term "agent" is a well-defined and understood term in the computer industry that means "a software program or routine." Ex. 9 (Jestice *HTC* Decl.) at ¶ 13. Similarly, dictionaries and other texts uniformly define "agent" to refer to a software program or routine. *E.g.*, Ex. 10 (THE TELECOMMUNICATIONS HANDBOOK (1999) at 571 ("An agent is a program, which, with a certain degree of autonomy, performs tasks on behalf of a user or an application.").[3] Even Defendants appear to understand that an "agent" is software program or routine. Combined, they have nearly 90 patents or patent applications reciting the use of a "software agent." Seigel Decl. at ¶¶ 14-18.

ii) It Is Understood That "Software" Is Structure.

Nor can Defendants dispute that because "agent" means "a software program or routine" it provides structure. Primarily, a long list of courts have found that references to "software" and like words connote sufficient structure to avoid § 112 ¶ 6. *E.g.*, *Collaborative Agreements, LLC v. Adobe Sys. Inc.*, 2015 WL 2250391, at *13 (W.D. Tex. May 12, 2015) ("Courts have consistently

[3] Ex. 11 (MICROSOFT PRESS COMPUTER USER'S DICTIONARY (1998)) at 13 ("agent *n* 1. A program that performs a background task for a user when the task is done or some expected event has taken place."); Ex. 12 (MICROSOFT COMPUTER DICTIONARY (4th Ed. 1999)) at 18-19 (same); Ex. 13 (ENCYCLOPEDIA OF COMPUTER SCIENCE (4th Ed. 2000)) at 1194-95 ("Multi-agent systems are computational systems in which several artificial 'agents', which are programs, interact or work together over a communications network to perform some set of tasks . . . ."); Ex. 14 (PC Magazine) (An agent is "a software routine that waits in the background and performs an action when a specified event occurs. For example, agents could transmit a summary file on the first day of the

interpreted 'software' and similar terms to have sufficient structure so as to avoid an invocation of Section 112 ¶ 6."); *accord Zeroclick, LLC v. Apple Inc.*, 891 F.3d 1003, 1008 (Fed. Cir. 2018) (holding that the terms "program" and "user interface code" were not "nonce words" but instead connoted sufficient structure to avoid § 112 ¶ 6); *WhitServe LLC v. GoDaddy.com, Inc.*, 2014 WL 5668335, at *4 (D. Conn. Nov. 4, 2014) (holding that claim language requiring "software executing on said computer" was sufficient to avoid invocation of § 112 ¶ 6); *RLIS, Inc. v. Allscripts Healthcare Solutions, Inc.*, 2013 WL 3772472, at *14 (S.D. Tex. July 16, 2013) (holding that "computer executable database software," "an editing software utility," "executable software," and "computer software" avoided § 112 ¶ 6); *Eolas Techs., Inc. v. Adobe Sys.*, Inc., 810 F. Supp. 2d 795, 810 (E.D. Tex. 2011) ("computer readable program code" sufficient); *Aloft Media, LLC v. Adobe Sys., Inc.*, 570 F. Supp. 2d 887, 898 (E.D. Tex. 2008) ("computer code" sufficient); *SecurityProfiling, LLC*, 2018 WL 4585279, at *3 ("code" sufficient).

Moreover, not only do the claims themselves recite what such software does, but Defendants concede that, at a minimum, the "algorithm found at 6:18-28" provides structure for the term. Without more, this admission precludes Defendants from overcoming the presumption against means-plus-function claiming. *See Verizon Calif. Inc. v. Ronald A. Katz Tech. Licensing, P.A.*, 326 F. Supp. 2d 1060, 1106 (C.D. Cal. 2003) ("[A]s Verizon itself concedes, 'interface unit' connotes some structure. Therefore, '[i]n the absence of any more compelling evidence of the understanding of one of ordinary skill in the art, the presumption that § 112, ¶ 6 does not apply is determinative.'" (quoting *Apex Inc. v. Raritan Computer, Inc.*, 325 F.3d 1364, 1373 (Fed. Cir. 2003))."

Finally, even if Defendants could overcome the presumption, Defendants are incorrect that the "algorithm found at 6:18-28" is the only structure identified in the patent. The patent also

---

month or monitor incoming data and alert the user when a certain transaction has arrived.").

discusses embodiments of this limitation at least at 1:60-2:1 and 3:51-61. Thus, all three algorithms must be permitted. *Creo Prod., Inc. v. Presstek, Inc.*, 305 F.3d 1337, 1345 (Fed. Cir. 2002) (reiterating that, "in the case of a means-plus-function claim, . . . the written description may disclose distinct and alternative structures for performing the claimed function").

8. "set up a verification structure" (Claims 1, 3, 7, 14)

Term	Ancora's Construction	Defendants' Construction
"set up a verification structure"	plain and ordinary meaning	"forming a verification structure using a unique key for each computer and license record information in the program"

Recognizing that the "using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS" term is not a means-plus-function term, Defendants also offer a non-means-plus-function construction. It too should be rejected because it is inconsistent with the specification and the claims, which make clear that this step can comprise just "establishing or certifying the existence of a pseudo-unique key in the first non-volatile memory area; and establishing at least one license-record location in the first or the second nonvolatile memory area." '941 Patent at 6:18-22; *see also* Claim 7. And it is black-letter law that a construction that would "exclude[] a [disclosed] embodiment from the scope of the claim is rarely, if ever, correct." *Broadcom*, 732 F.3d at 1333 (citation and quotation marks omitted).

Indeed, at a basic level it appears that Defendants are conflating the "set up a verification structure" step with the process for "establishing a license-record," which dependent Claim 8 teaches can comprise "forming a license record by encrypting of the contents used to form a license record with other predetermined data contents, using the key." *Id*. at 5:40-52 (teaching that "encrypted license records" can be "form[ed] . . . from the contents" of the program "using the key").

Defendants' "using a unique key for each computer" limitation fails for the same reason: the specification and the claims expressly contemplate embodiments that do not use a "key" to set up a

verification structure. *See id.* at 2:62-3:3 ("In its broadest aspect, the invention provides for a method of restricting software operation within a license limitation including; . . . setting up a verification structure in the non-volatile memories, verifying the program using the structure, and acting on the program according to the verification."). Further, those embodiments that do use a key are not limited to using a "unique key." Rather, as dependent Claim 7 teaches, a "pseudo-unique key" may be used as part of the process to "set up the verification structure."

Finally, what Defendants intend by their "using . . . license record information in the program" limitation is unclear, but it appears that the text is entirely redundant of the already-present requirement that "the verification structure accommodat[e] data that includes at least one license record." Accordingly, it should be rejected. *See Harris Corp. v. IXYS Corp.*, 114 F.3d 1149, 1152 (Fed. Cir. 1997) (rejecting "construction [that] would contribute nothing but meaningless verbiage").

In short, "set up a verification structure" requires no construction; the claim already states in clear language what is required. The Court should adopt its plain and ordinary meaning.

9. "verifying the program using at least the verification structure" (Claims 1, 9, 16)

Term	Ancora's Construction	Defendants' Construction
"verifying the program using at least the verification structure"	"confirming whether a program is licensed using at least the verification structure"	confirming through an operating system (OS) level application whether a program is licensed using at least the verification structure

Ancora's construction of this term—the "verifying" step, for short—is the same as the court's in *Apple I* and thus is entitled to "reasoned deference." *Kinetic*, 2013 WL 6164592, at *3. In contrast, Defendants seek to insert an additional limitation requiring that the "verifying" step be performed "through an operating system (OS) level application." Defendants' additional limitation is contradicted by the claims, specification, prosecution history, and extrinsic evidence.

As an initial matter, the words "operating system (OS) level application" are found nowhere in the patent itself. Rather, it appears that Defendants are seeking to engraft that limitation entirely from the patentee's statement during prosecution that "software license management applications, such as the one of the present invention, are operating system (OS) level programs." Ex. 3 (February 5, 2002 Statement in Support of Amendment) at ANCORA_437. But context matters. *i4i Ltd. P'ship v. Microsoft Corp.*, 598 F.3d 831, 843 (Fed. Cir. 2010) (stating that, "[i]n evaluating whether a patentee has disavowed claim scope, context matters" and it is not enough to just "pluck[]" statements "from the prosecution history"). And here, that context makes clear the patentee never limited this step to being performed by an OS-level program or application.

Rather, as the prosecution history demonstrates, where the patentee intended that an OS-level program or application participate in a particular step, it amended the claims to say so specifically by adding the term "agent." *TecSec, Inc. v. Int'l Bus. Machs. Corp.*, 731 F.3d 1336, 1346 (Fed. Cir. 2013) ("[F]or prosecution disclaimer to attach, our precedent requires that the alleged disavowing actions or statements made during prosecution be both clear and unmistakable." (citation omitted)); *Elbex Video, Ltd. v. Sensormatic Elecs. Corp.*, 508 F.3d 1366, 1372 (Fed. Cir. 2007) (explaining that only a "clear and unmistakable surrender of claim scope" supports a finding of disclaimer).

i) The Patentee Added the Term "Agent" to the Claims to Identify When an OS-Level Program or Application Was Required.

The method of Claim 1 comprises four steps: "selecting," "using an agent to set up a verification structure" (the "setting up" step), "verifying," and "acting." And the very statement that Defendants appear to be relying on demonstrates that the patentee's statement regarding the involvement of OS-level programs/applications was directed to only one of these steps: the setting up step. As the patentee explained in the very next sentence, "BIOS programs and software licensing management applications do not ordinarily interact or communicate because when BIOS is running,

the computer is in a configuration mode, hence OS is not running." Ex. 3 at ANCORA_437.

In short, the patentee explained that, in the prior art, "BIOS and OS-level programs are normally mutually exclusive" such that "writing to the BIOS area is performed by the BIOS routines" and "there is no OS support whatsoever to write data to the system BIOS." *Id.* at ANCORA_437-38. The patentee thus explained that the invention "proceeds against conventional wisdom in the art" by enabling OS-level programs/applications to use "BIOS to store application data." *Id*. at ANCORA_438; *see id*. at ANCORA_437-38 (explaining that a key innovation of the '941 invention was that it allowed OS-level programs to interact with BIOS and use "BIOS as a storage medium"); *id.* at 207 (stating that "[t]he cited references do not render the present invention obvious as they do not teach or suggest, among other things, storing a verification structure, such as a software license information, in the BIOS . . . as is recited in the present claims").

Consistent with these statements, the patentee thus amended the Claim 1 "setting up" step to require the use of "an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS." Ex. 5 at ANCORA_412. Notably, however, the patentee added no such requirement to any of the other Claim 1 steps, including the "verifying" step at issue. This was a deliberate choice.

Claim 18 further illustrates the deliberate choice made by the patentee to require that only certain steps must be performed by an OS-level "agent." In contrast to Claim 1, the agent in Claim 18 expressly performs multiple claim steps, including the "verifying" step. However, the Examiner made clear in an Examiner's amendment that the OS-level agent in Claim 18 (then Claim 20) is not required to perform the "acting" step because "an agent" was not required to perform that step. Ex. 2 at ANCORA_449 (stating "the steps of 'encrypting…', 'storing…', and 'subsequently verifying…' are performed by the agent," but "[t]his does not apply, however, to the 'acting…' limitation").

The Examiner's Reasons for Allowance also draw a clear distinction between the agent's limited role in Claim 1 versus its expanded role in Claim 18 (then Claim 20). *Salazar*, 414 F.3d at 1347 (explaining examiner statements "may be evidence of how one of skill in the art understood the term at the time the application was filed"). As the Examiner explained, the Claim 1 invention only required "using an agent to set up a verification structure." *Id.* at 223. In contrast, Claim 18 (then Claim 20) also required "utilizing an agent to verify the application software program." *Id.* at 224.

In sum, the difference between the express text of the two claims, along with the Examiner's explanations and amendment, thus shows that the patentee's choice not to have the OS-level agent perform the "verifying" step in Claim 1 was deliberate. As such, requiring the "verifying" step in Claim 1 be performed through an OS-level agent would rewrite the claim in a manner that nullifies the patentee's purposeful distinctions. *See also* '941 Patent at Claims 7 & 14 (further specifying how the "agent" is to perform the step of "setting up the verification structure" recited in claim 1). For this reason alone, Defendants' construction should be rejected.

ii) At Most, the Patentee Contemplated the Combined Use of BIOS and an OS-Level Program or Application (and Other Utilities).

Even if the "verifying" step required the use of an "agent," Defendants' construction should be rejected because it improperly suggests that such "verifying" must be performed exclusively by "an operating system (OS) level application." Such a limitation is inconsistent with the core principle of the patent discussed above: namely, that it "proceeds against conventional wisdom" by allowing for the "combine[d]" use of "BIOS and OS level programs." Ex. 3 at ANCORA_437-38.

Indeed, the verifying step expressly contemplates that the BIOS participate in the verifying process—requiring that such "verifying" use "at least the verification structure from the erasable non-volatile memory of the BIOS." Further, various dependent claims contemplate that components other than an OS-level agent operating on the claimed "computer" may participate in the "verifying"

step, including a "remote license authentication bureau." '941 Patent at Claim 4.

In short, no basis exists to limit this step to being performed exclusively through "an operating system (OS) level application" as Defendants request.

10. "acting on the program according to the verification" (Claims 1, 10)

Term	Ancora's Construction	Defendants' Construction
"acting on the program according to the verification"	plain and ordinary meaning	"(i) allowing the use of the program if licensed or (ii) restricting the program's operation if not licensed, using an operating system (OS) level application"

This phrase means exactly what it says: acting on the program according to (or based on) "the verification," which the parties agree means "verifying the program using at least the verification structure from the erasable nonvolatile memory of the BIOS." No further construction is needed, and Defendants' construction is incorrect for at least three reasons.

First, Defendants' proposal that "acting" be performed "using an . . . OS level application" is wrong for the reasons Ancora explained—namely, Claim 1 requires only that the "setting up" step be performed by an agent. And as shown above at page 24, the prosecution history makes plain that both the patentee and the Examiner understood that neither Claim 1 nor Claim 18 required that the "acting" step be performed by the agent. *See also* Ex. 2 at ANCORA_449 (Examiner's Amendment to Claim 18 stating that "the steps of 'encrypting…', 'storing…', and 'subsequently verifying…' are performed by the agent" but that "[t]his does not apply, however, to the 'acting…' limitation").

Second, Defendants' "if licensed" and "if not licensed" limitations are wrong for the reasons Ancora also already explained above at pages 7-8 in the context of Defendants' "license record" argument. Simply put, the reference to "licensed" misleadingly and improperly suggests that any "acting" be based on the program's legal or contractual status. No such requirement exists in the claims or the specification, which instead explains that the focus of the "verifying" inquiry is to

determine only whether a program is "authorized or verified to run." '941 Patent at 2:10-26.

Indeed, in discussing an embodiment of the "acting" step, the specification never mentions the word "licensed." *Id.* at 6:40-45. Instead, the patent teaches that "[a]cting (20) on the program includes the step of restricting the program's operation with predetermined limitations if the comparing yields non-unity or insufficiency"—further explaining that, "[i]n this context 'non-unity' relates to being unequal with respect to a specific equation (e.g. A=B+1); and 'insufficiency' relates to being outside of a relational bound (e.g. A>B+1)." *Id.* Further, when speaking more generally about the claimed invention, the specification explains that, "[i]n the case of match, the program is verified to run on the computer." *Id.* at 2:19-20; *accord id.* at Claim 10.

Third, Defendants' proposal that "acting" can comprise only "allowing" or "restricting" a program's operation contradicts the specification, which details a diverse set of potential actions that can be taken. Those actions include, for example, "informing . . . the user" of a restriction (*id.* at 2:24-26), "halting the operation of the program" (*id.*), "ask[ing] for additional user interactions" (*id.* at 5:60-61), "erasing the software in volatile memory, warning the license applicant/user, placing a fine on the applicant/user . . . , or scrambling sections of the BIOS of the computer (or of functions interacting therewith)" (*id.* at 6:46-52). Defendants' binary construction is thus far too narrow.

11. The Order of the Claim 1 Steps

Term	Ancora's Construction	Defendants' Construction
Order of the Claim 1 Steps	The steps do not need to be performed in a specific order.	The "verifying the program" step and "acting on the program" step of claim 1 must sequentially follow the "selecting a program" step and "using an agent" step.

What Defendants intend by their proposed Claim 1 "ordering" limitation has changed several times and, even today, is not entirely clear. It appears, however, that Defendants are arguing (1) that the "selecting a program" and "using an agent" steps occur before the "verifying the program" and

"acting on the program" steps and, further, (2) that the "verifying the program" step occur before the "acting on the program" step. In short, Defendants' construction would appear to limit the invention to two possible sequences: "selecting" followed by "using" followed by "verifying" followed by "acting" (i.e., a 1-2-3-4 order) or "using" followed by "selecting" followed by "verifying" followed by "acting" (i.e., a 2-1-3-4 order). Neither limitation is correct.

First, the *Apple I* court already considered and rejected the argument that Claim 1 is subject to Defendants' 1-2-3-4 ordering requirement. 2012 WL 6738761, at *13. Explaining that it is understood that, "[u]nless the steps of a method actually recite an order, the steps are not ordinarily construed to require one," the court pointed to the fact that the specification teaches that the "selecting" step can occur at various points in the overall process to hold that "[n]othing in the '941 Patent directly or implicitly requires that the steps be performed in the order recited." *Id.* (quoting *Altiris, Inc. v. Symantec Corp.*, 318 F.3d 1363, 1369-70 (Fed. Cir. 2003)).

Second, Defendants' 2-1-3-4 and "sequentially follow" limitations also are incorrect. It is a basic principle that use of the word "'comprising' in a method claim indicates that the claim is open-ended and allows for additional steps." *Solvay S.A. v. Honeywell Int'l Inc.*, 742 F.3d 998, 1005 (Fed. Cir. 2014) (citation omitted). As such, no basis exists to require that any steps occur "sequentially."

Further, it also is black-letter law that, unless the claim language demands it, it is error to "require that each step occur independently or separately." *Kaneka Corp. v. Xiamen Kingdomway Grp. Co.*, 790 F.3d 1298, 1306 (Fed. Cir. 2015) (explaining that, generally speaking, method "claims do not exclude a continuous process, in which later steps are initiated . . . while previous steps are still ongoing"). And here nothing precludes such simultaneous performance. To the contrary, Claim 18 shows that—when a particular order is required—the claims say so explicitly. Unlike in Claim 1, in Claim 18 the patentee expressly required that the process "[s]tor[e] the encrypting license

information" before "subsequently verifying" the relevant program.

Likewise, Mr. Jestice confirmed that it was understood at the time of the invention that a device could be programmed such that it implements multiple functions simultaneously (either in whole or in part). Ex. 7 (Jestice Decl.) at ¶ 14. And he further confirmed that a POSITA would not understand anything in Claim 1 or the specification to suggest that the patentee did not intend to adhere to that understanding here. *Id*. Rather, a POSITA would understand that, at a minimum, each of the recited Claim 1 steps could overlap such that some "steps are initiated . . . while previous steps are still ongoing." *Kaneka*, 790 F.3d at 1306.

12. "first non-volatile memory area of the computer" (Claim 7)

Term	Ancora's Construction	Defendants' Construction
"first non-volatile memory area of the computer"	plain and ordinary meaning	"read only memory area of the computer"

Defendants' proposal differs from the claim language only by substituting "read only" for "first non-volatile." Thus, the only question is whether this substitution is proper. It is not.

Defendants' proposal violates the presumption that claim terms carry the same meaning throughout all claims. Defendants agree that "non-volatile memory" in Claim 1 means "memory whose data is maintained when the power is removed." Yet they seek a different construction for the same term in Claim 7. "But the principle that the same phrase in different claims of the same patent should have the same meaning is a strong one, overcome only if 'it is clear' that the same phrase has different meanings in different claims." *In re Varma*, 816 F.3d 1352, 1363 (Fed. Cir. 2016) (citation and quotations omitted). There is no "clear" showing here.

Instead, the specification shows why Defendants' construction is incorrect. The passage at column 3, lines 18 to 32, teaches that computer memory is—and can be—used in many different manners, and that it is not the physical properties of the particular memory that matters, but rather

how it is used. The patent does this by referring to memory as either "volatile" or "non-volatile." *Id.*

The specific embodiments the patent recites only reinforce that the patentee did not narrow the claim scope of "first non-volatile memory area" to only mean "read only memory." One embodiment describes the invention describing the memory only as "volatile" and "non-volatile":

> In its broadest aspect, the invention provides for a method . . . including; for a computer having a first non-volatile memory area, a second non-volatile memory area, and a volatile memory area; the steps of: selecting a program residing in the volatile memory, setting up a verification structure in the non-volatile memories, verifying the program using the structure, and acting on the program according to the verification.

'941 Patent at 2:62-3:3. Another embodiment describes the invention as having only a single non-volatile memory that nonetheless is treated as having a "first and a second non-volatile memory areas"—explaining: "There are also processors having only one non-volatile memory or having more than two non-volatile memories; all of which should be considered logically as relating to having a first and a second non-volatile memory areas." *Id.* at 3:24-28. This embodiment would be rendered inoperable if the sole non-volatile memory area were read-only, which would prevent performance of the "setting up" step required by the claims. *See Broadcom*, 732 F.3d at 1333 ("[A]n interpretation which excludes a [disclosed] embodiment from the scope of the claim is rarely, if ever, correct." (brackets in original, citation and quotation marks omitted)). Finally, the embodiment recited at column 4, lines 45 to 48, refers to the "first non-volatile memory" without any mention of ROM—stating only that it comprises a generic and unspecified form of "memory media."

For each of these reasons, Defendants' construction is incorrect and should be rejected.

Date: March 20, 2020

/s/ *Andres Healy*
Charles Ainsworth
State Bar No. 00783521
Robert Christopher Bunt
State Bar No. 00787165
PARKER, BUNT & AINSWORTH, P.C.
100 E. Ferguson, Suite 418
Tyler, TX 75702
903/531-3535
E-mail: charley@pbatyler.com
E-mail: rcbunt@pbatyler.com

Lexie G. White (Texas 24048876)
Susman Godfrey LLP
1000 Louisiana Street, Suite 5100
Houston, Texas 77002
Tel: (713) 651-9366
Fax: (713) 654-6666
lwhite@susmangodfrey.com

Andres Healy (*pro hac vice*)
Steven M. Seigel (*pro hac vice*)
Susman Godfrey LLP
1201 Third Avenue, Suite 3800
Seattle, Washington 98101
Tel: (206) 516-3880
Fax: 206-516-3883
ahealy@susmangodfrey.com
sseigel@susmangodfrey.com

Zachary B. Savage (*pro hac vice*)
Susman Godfrey LLP
1301 Avenue of the Americas, 32nd Floor
New York, New York 10019
Tel: (212) 336-8330
Fax: (212) 336-8340
zsavage@susmangodfrey.com

COUNSEL FOR PLAINTIFF ANCORA TECHNOLOGIES, INC.

CERTIFICATE OF SERVICE

I hereby certify that all counsel of record, who are deemed to have consented to electronic service are being served this 20th day of March, 2020, with a copy of this document via the Court's CM/ECF system.

/s/ *Michelle Wimmer*
Michelle Wimmer