

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NINTENDO CO., LTD., and NINTENDO OF AMERICA INC.,
Petitioner,

v.

ANCORA TECHNOLOGIES, INC.,
Patent Owner.

U.S. Patent No. 6,411,941 B1

Case No.: IPR2021-01338

**PATENT OWNER'S PRELIMINARY RESPONSE TO PETITION
FOR *INTER PARTES* REVIEW UNDER 37 C.F.R. § 42.107**

TABLE OF CONTENTS

Table of Authorities	iii
List of Exhibits	vi
I. Introduction	1
II. Standard for Granting <i>Inter Partes</i> Review	3
III. Overview of the '941 Patent	4
A. The Key Innovation of the '941 Patent Is Use of an OS-Level Software Agent To Interact With BIOS Memory Without Requiring Additional or Specialized Hardware	4
1. The Patentee Disclaimed Methods Requiring Use of Additional Hardware	6
2. Another “Key Distinction” Over the Prior Art Was Using an OS-Level Software Agent to Store Verification Structures and License Records in BIOS Memory.....	8
B. The '941 Patent Has Survived Four Separate Challenges to Its Validity and Patentability.....	12
C. Recent IPRs That Were Instituted (But Terminated) Based on Hellman and Chou Did Not Address Key Claim Construction Issues	14
IV. Claim Construction	15
A. “a computer”	17
B. “using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS”	20
1. “agent”	20
2. “to set up a verification structure”	31
3. “memory of the BIOS”	35
V. The Asserted Prior Art.....	40
A. Hellman	40
B. Chou	44
C. Schneck	47
VI. The Asserted Combinations Fails To Disclose Key Elements of Claim 1 of the '941 Patent.....	48
A. The Only “Agent” Petitioner Identifies is Hellman’s “Update Unit 36,” which Requires “Add-on Hardware” and Thus Fails to Disclose the “Using an Agent...” Limitation.....	48

B.	Petitioner’s Proposed Combination Does Not “Set Up A Verification Structure” and Petitioner Fails to Disclose How it is “Set Up” by the “Agent”.....	52
C.	The Proposed Combination Fails to Teach Using “Memory of the BIOS” for Storing a “Verification Structure” or “License Record”..	57
D.	Petitioner Does Not Assert That Schneck Discloses Either Claim 1 Limitation at Issue	63
VII.	Conclusion.....	64
	Certificate of Service	65
	Certificate of Compliance Pursuant to 37 C.F.R. §42.24.....	67

Table of Authorities

Cases

<i>Am. Med. Sys., Inc. v. Biolitec, Inc.</i> , 618 F.3d 1354 (Fed. Cir. 2010).....	18
<i>Ancora Techs., Inc. v. Apple, Inc.</i> , 744 F.3d 732 (Fed. Cir. 2014).....	5, 6, 9-13, 16, 23, 25-30, 35, 39, 51
<i>Ancora Techs., Inc. v. HTC Am., Inc.</i> , 908 F.3d 1343 (Fed. Cir. 2018).....	4-6, 12, 14, 23, 25, 30-36, 39, 64
<i>Applied Med. Res. Corp. v. U.S. Surgical Corp.</i> , 448 F.3d 1324 (Fed. Cir. 2006).....	2, 55
<i>Ariosa Diagnostics v. Verinata Health, Inc.</i> , 805 F.3d 1359 (Fed. Cir. 2015).....	4
<i>Cutsforth, Inc. v. Motivepower, Inc.</i> , 643 F. App'x 1008 (Fed. Cir. 2016).....	1
<i>Elbit Systems of America, LLC v. Thales Visionix, Inc.</i> , 881 F.3d 1354 (Fed. Cir. 2018).....	57
<i>Genetics Inst., LLC v. Novartis Vaccines & Diagnostics, Inc.</i> , 655 F.3d 1291 (Fed. Cir. 2011).....	2
<i>HTC Corp. v. Ancora Techs., Inc.</i> , No. CBM2017-00054, Decision Denying Institution (PTAB Dec. 1, 2017).....	12, 14
<i>In re Magnum Oil Tools Int'l, Ltd.</i> , 829 F.3d 1364 (Fed. Cir. 2016).....	3, 57, 64
<i>Indivior Inc. v. Dr. Reddy's Lab 'ys, S.A.</i> , 930 F.3d 1325 (Fed. Cir. 2019).....	27
<i>Kinetic Concepts, Inc. v. Smith & Nephew, Inc.</i> , 688 F.3d 1342 (Fed. Cir. 2012).....	3

<i>Medicines Co. v. Mylan, Inc.</i> , 853 F.3d 1296 (Fed. Cir. 2017).....	27
<i>Samsung v. Ancora</i> , IPR2020-01184, Paper 1 (PTAB Jan. 5, 2021).....	38
<i>SciMed Life Systems v. Advanced CardioVascular</i> , 242 F.3d 1337 (Fed. Cir. 2001).....	27
<i>Sony Mobile Commc 'ns Ab v. Ancora Techs., Inc.</i> , No. IPR2021-00663, Paper 17 (PTAB June 10, 2021)	40
<i>TCT Mobile (US) Inc. v. Ancora Techs., Inc.</i> , IPR2020-01609 (PTAB Feb. 2, 2021).....	15
<i>Traxcell Techs., LLC v. Nokia Sols. & Networks Oy</i> , 15 F.4th 1136 (Fed. Cir. 2021).....	22, 25
<i>Wasica Fin. GmbH v. Cont'l Auto. Sys., Inc.</i> , 853 F.3d 1272 (Fed. Cir. 2017).....	55
<i>Watts v. XL Sys., Inc.</i> , 232 F.3d 877 (Fed. Cir. 2000).....	26

Statutes

35 U.S.C. §101	14
35 U.S.C. §103	3
35 U.S.C. §314	3

Other Authorities

Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,756 (Aug. 14, 2012).....	3
PTAB Rules of Practice for Instituting on All Challenged Patent Claims, 85 Fed. Reg. 79,120, 79,121	51

Rules

37 C.F.R. §2.108.....	3
37 C.F.R. §42.100.....	16

List of Exhibits

Exhibit No.	Description	Date
2001	Microsoft Corporation's Request for <i>Ex Parte</i> Reexamination Image File Wrapper, Control No. 90010560	
2002	<i>Ancora Techs., Inc. v. Apple, Inc.</i> , 744 F.3d 732 (Fed. Cir. 2014)	Mar. 3, 2014
2003	Deposition Excerpts of Jon Weissman, <i>Ancora Technologies, Inc. v. HTC America, Inc.</i> , Case No. 2:16-cv-01919	Sept. 9, 2019
2004	Declaration of Ian Jestice, <i>Ancora Technologies, Inc. v. HTC America, Inc.</i> , Case No. 2:16-cv-01919	Aug. 26, 2019
2005	Brief of Appellees HTC America, Inc. and HTC Corporation, <i>Ancora Technologies, Inc. v. HTC America, Inc., HTC Corporation</i> , Case No. 18-1404	Apr. 23, 2018
2006	Declaration of Jon Weissman, <i>Ancora Technologies, Inc. v. HTC America, Inc.</i> , Case No. 2:16-cv-01919	Sept. 4, 2019
2007	Terplan, Kornel, Morreale, Patricia, <i>The Telecommunications Handbook</i> , CRC Press, 2000	
2008	Computer User's Dictionary, Microsoft Press, 1998	
2009	Microsoft Computer Dictionary Fourth Edition, Microsoft Press, 1999	
2010	PC Magazine Encyclopedia, definition of "Agent," https://www.pcmag.com/encyclopedia	
2011	U.S. Patent No. 6,411,941 File History with Beeble White Paper	Sept. 2001

Exhibit No.	Description	Date
2012	Joint Claim Construction Chart, <i>Ancora Technologies, Inc. v. TCT Mobile (US) Inc., Huizhou TCL Mobile Communication Co., Ltd., and Shenzhen TCL Creative Cloud Technology Co., Ltd.</i> , Case No. 8:19-cv-02192 (Dkt. #49, 49-1, 49-2)	July 17, 2020
2013	Declaration of Dr. David Martin, Ph.D., <i>Sony Mobile Communications AB, Sony Mobile Communications, Inc., Sony Electronics Inc., and Sony Corporation v. Ancora Technologies, Inc.</i> , IPR2021-00663, Ex. 2015	Apr. 23, 2021
2014	U.S. Patent No. 6,189,146 (Misra)	Feb. 13, 2001
2015	U.S. Patent No. 5,479,639 (Ewertz)	Dec. 26, 1995
2016	Decision Denying Institution of <i>Inter Partes</i> Review, <i>Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc. v. Ancora Technologies, Inc.</i> , IPR2020-01184	Jan. 5, 2021
2017	Declaration of Ian Jestice, <i>Ancora Technologies Inc. v. LG Electronics Inc., LG Electronics U.S.A. Inc., Samsung Electronics Co., Ltd., and Samsung Electronics America, Inc.</i> , Case No. 1:20-cv-00034 (Dkt. # 44-8)	Mar. 20, 2020

I. Introduction

The Board should decline to institute Nintendo’s Petition for *Inter Partes* Review of U.S. Patent No. 6,411,941 (“the ’941 Patent”). See Petition, IPR2021-01338. Petitioner raises only obviousness challenges under 35 U.S.C. §103 based on three references: Hellman, Chou, and Schneck. Ground 1 asserts a combination of Hellman and Chou; Ground 2 asserts a combination Hellman, Chou, and Schneck.

The Petition should be denied because Petitioner has not shown how any asserted reference discloses or teaches key limitations of independent Claim 1—the only independent claim challenged—when its terms are properly construed. Specifically, none of the asserted prior art discloses the limitations “a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area” or “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.”

Petitioner’s only argument that such limitations are disclosed relies on leveraging the absence of any construction to “expand[] the meaning of [each] term ... beyond its plain meaning without any supporting evidence.” *Cutsforth, Inc. v. Motivepower, Inc.*, 643 F. App’x 1008, 1012 (Fed. Cir. 2016). In particular, Petitioner argues that it “does not believe that any claim constructions are needed for the purposes of this review.” (Pet. at 23.) This is wrong.

Only by divorcing Claim 1’s key limitations from their meaning to a skilled artisan can Petitioner argue that such limitations are taught in the asserted art. But “obviousness require[s] the [Board] to compare the **properly construed claims**¹ to the available prior art.” *Genetics Inst., LLC v. Novartis Vaccines & Diagnostics, Inc.*, 655 F.3d 1291, 1302 (Fed. Cir. 2011). And when any of the terms identified above is given its proper construction (as Patent Owner proposes below) the Petition fails.

Furthermore, even if the Board were to decline to construe any of the above terms, the Petition still fails because Petitioner cannot show how any of its references or combinations discloses at least “set[ting] up a verification structure in the erasable, non-volatile memory of the BIOS.” That is because Petitioner collapses the meaning of two distinct terms—“verification structure” and “license record”—even though the specification shows, and binding precedent holds, that “the use of two terms in a claim requires that they connote different *meanings*....” *Applied Med. Res. Corp. v. U.S. Surgical Corp.*, 448 F.3d 1324, 1333 n.3 (Fed. Cir. 2006) (emphasis in original).

For these reasons, Petitioner fails to establish a likelihood of prevailing and the Petition should not be instituted.

¹ All emphasis added unless otherwise indicated.

II. Standard for Granting *Inter Partes* Review

A petition for *inter partes* review may be granted only when “the information presented in the petition ... shows that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.” 35 U.S.C. §314(a); 37 C.F.R. §2.108(c). Petitioner bears the burden of showing that this statutory threshold has been met. *See* Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,756 (Aug. 14, 2012) [hereinafter “Practice Guide”] (“The Board ... may institute a trial where the petitioner establishes that the standards for instituting the requested trial are met”).

For a claim of obviousness under 35 U.S.C. §103, Petitioner must show where in the prior art each claim limitation (as properly construed) is found. *See, e.g., Kinetic Concepts, Inc. v. Smith & Nephew, Inc.*, 688 F.3d 1342, 1361 (Fed. Cir. 2012). Failure to do so defeats a claim of obviousness. *Id.*; *accord* Practice Guide, 77 Fed. Reg. at 48,764 (explaining that “potential patent owner preliminary responses” include showing that “[t]he prior art lacks a material limitation in all of the independent claims”).

Further, a Petitioner cannot rely on the Board to substitute its own reasoning to remedy the Petition’s deficiencies, because the Board may not “raise, address, and decide unpatentability theories never presented by the petitioner.” *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1381 (Fed. Cir. 2016). Thus, “a challenge can fail

even if different evidence and arguments might have led to success.” *Ariosa Diagnostics v. Verinata Health, Inc.*, 805 F.3d 1359, 1367 (Fed. Cir. 2015).

III. Overview of the ’941 Patent

A. The Key Innovation of the ’941 Patent Is Use of an OS-Level Software Agent To Interact With BIOS Memory Without Requiring Additional or Specialized Hardware

The ’941 Patent relates to a specific technique for “identifying and restricting an unauthorized software program’s operation.” (Ex.1001 at 1:6-8.)

Before the ’941 Patent, there were two basic methods of verifying and restricting the operation of a program on a computer. The first approach involved “software-based methods” that “require[d] writing a license signature on the computer’s hard drive.” *Ancora Techs., Inc. v. HTC Am., Inc.*, 908 F.3d 1343, 1344 (Fed. Cir. 2018) (“*HTC*”). 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 Ex.1001 at 1:19-26.)

The second approach involved hardware-based methods that required the use of additional hardware not found in a conventional computer. For example, such techniques “require[d] inserting a dongle into a computer port to authenticate the software authorization.” *Id.* (citing Ex.1001 at 1:27-32.) As the patent explained, such hardware-based “methods [we]re costly, inconvenient, and not suitable for software sold and downloaded over the internet.” (Ex.1001 at 1:29-32.)

The '941 Patent thus disclosed a better solution—one that disclaimed the need for additional hardware because it instead “makes use of the existing computer hardware” by “using the memory space associated with the computer’s basic input/output system (BIOS), rather than other memory space, to store appropriately encrypted license information to be used in the verification process.” *Ancora Techs., Inc. v. Apple, Inc.*, 744 F.3d 732, 733 (Fed. Cir. 2014) (citing Ex.1001 at 1:46-2:5, 4:45-48, 5:19-24). As the Federal Circuit recognized, 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.

The '941 Patent thus proceeded against the conventional wisdom in the art to do something “the closest prior art, singly or collectively,” never contemplated: using “programs running at the OS level interacting with a program verification structure stored in the BIOS to verify [a] program using the verification structure”—that is, “using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS” as Claim 1 requires. (Ex.2011 at 155; *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 BIOS memory in this unexpected manner was a critical innovation that “improves computer security” in a number of ways, including “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. Equally important, by making use of conventional “existing computer hardware,” *Apple*, 744 F.3d at 733, the invention avoided the need for the “expensive, inconvenient” additional hardware that prior art hardware-based approaches required. *HTC*, 908 F.3d at 1344.

1. The Patentee Disclaimed Methods Requiring Use of Additional Hardware

In the course of the '941 Patent's two separate trips to the Federal Circuit, the Federal Circuit has twice recognized—and repeatedly emphasized—that the patentee expressly disclaimed the need for additional hardware to perform the claimed methods, explaining:

Thus, the inventors stated that their method makes use of the existing computer hardware (eliminating the expense and inconvenience of using additional hardware), while storing the verification information in a space that is harder and riskier for a hacker to tamper with than storage areas used by earlier methods.

Apple, 744 F.3d at 733-34; *accord HTC*, 908 F.3d at 1345.

The Federal Circuit is not alone in identifying the patentee's disclaimer of methods that require use of additional hardware. The Patent and Trademark Office ("Patent Office") recognized this same disclaimer of hardware-based methods during examination. Specifically, the Examiner initially rejected the invention over the Ginter reference, which utilized additional or specialized hardware.

In response, the patentee argued that Ginter "suffers from the deficiency" that was expressly called out in the specification: it required "add-on hardware" that was "not part of the PC." (Ex.2011 at 87-88.) As the applicant explained in detail:

[I]n col. 70, line 23 – col. 71, line 25 Ginter et al. describe the architecture as add-on hardware which is named "SPU".... Col. 64, lines 16-21 explicitly detail[s] the fact that the SPU is a hardware, add-on not part of the PC.

(Ex.2011 at 88.) The applicant emphasized this point again when explaining:

There is no mention whatsoever in Ginter et al. ... referred to by the Examiner of a process where a software program verifies its authenticity using a license (verification structure) stored in the second volatile non-volatile memory. The functionality described in these portions of Ginter et al. is the different functionality that **add-on hardware**, referred to as SPU, can perform.

(Ex.2011 at 89.)

After submitting these explanations to the Examiner, the patentee spoke with the Examiner. At that interview, "editing independent claim language [to

encapsulate this distinction] was discussed,” (Ex.2011 at 112), and the claims thereafter were “amended as agreed during the interview” to add the “using an agent” limitation. (Ex.2011 at 120.)

In short, the applicant added the “using an agent” portion of the limitation at issue to make clear what the invention encompassed and what was disclaimed—crystalizing that the invention did not encompass the use of “a hardware[] add-on not part of the PC.” (Ex.2011 at 88.) Rather, the claimed “agent” had to exist on and run from (at least in part) the conventional computer itself.

The Examiner understood and accepted this express disclaimer—explaining in its Reasons for Allowance that the invention proceeded against the conventional wisdom in the art to do something “the closest prior art systems, singly or collectively,” never contemplated: “using an agent [*i.e.*, software] to set up a verification structure in the erasable, non-volatile memory of the BIOS.” (Ex.2011 at 155.)

2. Another “Key Distinction” Over the Prior Art Was Using an OS-Level Software Agent to Store Verification Structures and License Records in BIOS Memory

In addition to disclaiming the use of specialized or “add-on” hardware, the invention also improved over prior art software techniques by making use of BIOS memory, rather than other memory, to perform software-verification functions. As

the Federal Circuit emphasized, the patentee distinguished its invention from prior software-based techniques, including the techniques disclosed by the two primary prior art references discussed during examination: Ewertz, which “disclosed storage in the BIOS memory area by the BIOS software itself,” and Misra, which disclosed “software implemented in or through an operating system.” *Apple*, 744 F.3d at 735–36. “The applicants explained that their invention differed from the[se] prior art [methods] in that it *both* operated as an application running through an operating system *and* used the BIOS level for data storage and retrieval – **a combination that was not previously taught and that an ordinarily skilled application writer would not employ.**” (*Id.*, italics in original.)

During prosecution, the patentee reinforced and expressly described how its invention “proceed[ed] against conventional wisdom in the art” because:

BIOS is a configuration utility. Software license management applications, such as the one of the present invention, are operating system (OS) level programs.... [W]hen BIOS is running, the computer is in a configuration mode, hence OS is not running. Thus, BIOS and OS level programs are normally mutually exclusive.

(Ex.2011 at 146-147.) Thus, because of the prior art *status quo* of “mutual[] exclusiv[ity]” between BIOS and OS levels, “[u]sing BIOS to store application data such as that stored in Misra’s local cache for licenses [wa]s not obvious.” (*Id.*)

The Examiner agreed. In its subsequent Notice of Allowance, the Examiner explained clearly that the claims recited licensed programs running at the OS level interacting with a verification structure at the BIOS level—an approach that was never previously contemplated or disclosed in the extensive art cited during examination. As the Examiner explained in its “Reasons for Allowance”:

[T]he key distinction between the present invention and the closest prior art, is that the Misra et al., and Ginter et al. systems and the Ewertz et al. system run at the operating system level and BIOS level, respectively. More specifically, 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 the BIOS** to verify the program using the verification structure and having a user act on the program according to the verification. Further, it is well known to those of ordinary skill of the art that a computer BIOS is not setup to manage a software license verification structure. The present invention overcomes this difficulty by using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS.

(Ex.2011 at 155.)

The Federal Circuit also emphasized this precise point of novelty in observing that, prior to the '941 invention, “[a]n ordinary skilled artisan would not consider the BIOS as a storage medium to preserve application data for at least two reasons.” *Apple*, 744 F.3d at 735-36. As it explained:

First, ... [a]n ordinary person skilled in the art makes use of OS features to write data 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....

Second, no file system is associated with the [prior art] BIOS.... This is further evidence that OS level application programmers would not consider the BIOS as a storage medium for license data.

Id.

In short, a “key distinction” between the ’941 Patent and the “closest prior art,” *id.*, was the invention’s use of an OS-level software agent that was able to access and use BIOS memory (previously understood as inaccessible to OS-level programs) to store verification structures and license records that were to be used to verify a program.

Because the “agent” necessarily had to be an OS-level “software program or routine” located on and run from (at least in part) the recited “computer,” this same innovation also distinguished the invention from prior art hardware-based techniques. (Ex. 1001 at 1:44-47; *id.* at 3:33-47 (teaching that there may be a “license authentication bureau which can participate in ... establishing the license record in the second non-volatile memory”).) In other words, consistent with the patentee’s criticism of prior art hardware techniques as “expensive, inconvenient, and not [] suitable for software [] sold [and] download[ed] [] over the internet” because they

required the use of additional, non-conventional hardware, the invention required the agent to comprise OS-level software on the computer such that no additional hardware was necessary. (Ex.1001 at 1:27-32; *Ancora*, 744 F.3d at 733–34.)

This was important. In affirming the validity of the '941 Patent, the Federal Circuit twice emphasized that it “eliminat[ed] the expense and inconvenience of using additional hardware.” *Apple*, 744 F.3d at 733-34; *see HTC*, 908 F.3d at 1345. This Board reached the same conclusion. In an earlier Covered-Business Method Patent Review proceeding, it explained that the '941 Patent improved on “hardware based products that are expensive, inconvenient, and not suitable for downloaded software.” *HTC Corp. v. Ancora Techs., Inc.*, No. CBM2017-00054, Decision Denying Institution, Paper 7 at 9 (PTAB Dec. 1, 2017) (citing Ex.1001 at 1:19-32).

B. The '941 Patent Has Survived Four Separate Challenges to Its Validity and Patentability

Given the well-recognized novelty of the '941 Patent, it is not surprising that the '941 Patent's claims have survived no fewer than four separate attempts to re-examine and invalidate them since their original examination in 2002.

The first unsuccessful challenge came in 2009, when Microsoft requested *ex parte* reexamination of the '941 Patent. Control No. 90/010,560. (Ex.2001.) In that proceeding, the Patent Office confirmed the patentability of all 19 claims of the '941 Patent—focusing specifically on the claim requirement “memory of the BIOS.” As

the Patent Office made clear, “consistent with the specification of the ’941 patent,” BIOS is “defined by the functional descriptive material contained within it” such that “one skilled in the art” would consider a memory area to be “memory of the BIOS” only if the non-volatile memory area contains data “used by the functions of BIOS itself”—which the Patent Office explained includes to “test hardware at startup, start[] the operating system, and supports the transfer of data among hardware devices.”² (Ex.2001 at 163-164.) Thus, the mere fact that data resides in non-volatile memory “does not necessarily mean that it is part of the BIOS.” (*Id.*) More is required to meet the claims. Accordingly, the Patent Office confirmed the patentability of all claims of the ’941 Patent.

In 2012, Apple, Inc. next unsuccessfully challenged the claims of the ’941 Patent as indefinite in district court litigation. The Federal Circuit affirmed the district court’s finding that the claim terms “volatile memory” and “non-volatile memory” were not indefinite. *Apple*, 744 F.3d at 737-40. Apple also filed a petition for covered business review that was dismissed following the parties’ settlement.

In 2017, HTC Corporation sought to revive Apple’s prior CBM petition—

² As the Patent Office also explained, this does not preclude data stored in BIOS from “being also used or modified by programs located outside of the BIOS, such as applications running in an operating system.” (Ex.2001 at 164.)

effectively duplicating Apple’s petition for covered business method review. The Board denied institution, finding that the ’941 Patent was not a covered business method patent because it disclosed a technical solution in the form of storing the license record in the memory of the BIOS. *HTC Corp. v. Ancora Techs., Inc.*, No. CBM2017-00054, Decision Denying Institution, Paper 7, pp. 10–12 (Dec. 1, 2017).

More recently, in 2018, the Federal Circuit rejected another attempt by HTC Corporation to invalidate the ’941 Patent as patent ineligible under 35 U.S.C. §101. In *Ancora Techs. v. HTC Am., Inc.*, the Federal Circuit reversed the district court’s ruling that the claims of the ’941 Patent were patent ineligible and held that the “claims at issue ... are not directed to ineligible subject matter” but represent “a concrete assignment of specified functions among a computer’s components to improve computer security.” 908 F.3d 1343, 1344 (Fed. Cir. 2018).

C. Recent IPRs That Were Instituted (But Terminated) Based on Hellman and Chou Did Not Address Key Claim Construction Issues

Most recently, the ’941 Patent was the subject of several institution decisions based on the same prior art grounds Nintendo relies on here: Hellman and Chou.³

³ All of the petitions were identical as they were based on the same initial IPR filed by TCT Mobile. See *TCT Mobile (US) Inc. v. Ancora Techs., Inc.*, IPR2020-01609

Notably, though, in those matters, the only construction the Board considered was the meaning of the term “license record.” At the time, and for simplicity’s sake, the Patent Owner did not identify any additional terms for construction. As such, many of the shortcomings of the Hellman/Chou obviousness combinations were left unaddressed. The present Preliminary Response addresses these shortcomings by better explaining the proper scope and meaning of key claim terms—and the invention itself—to ensure that the invention is construed consistently with its specification, prosecution history, and the long history of district court, Federal Circuit, Patent Office, and Board decisions interpreting the ’941 Patent.

IV. Claim Construction

“In an *inter partes* review proceeding,” the challenged claims “shall be construed using the same claim construction standard that would be used to construe the claim in a civil action ... including construing the claim in accordance with the

(PTAB Feb. 2, 2021). When the Board instituted that IPR, other parties filed “copycat” petitions and motions for joinder. *See HTC Corporation and HTC America, Inc., v. Ancora Technologies, Inc.*, IPR2021-00570 (PTAB Feb. 19, 2021); *Sony Mobile Communications AB, Sony Mobile Communications, Inc., Sony Electronics Inc., and Sony Corporation v. Ancora Technologies, Inc.*, IPR2021-00663 (PTAB Mar. 15, 2021). The Sony IPR was instituted.

ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.” 37 C.F.R. §42.100(b). In construing a claim, the Board must consider “[a]ny prior claim construction determination concerning a term of the claim in a civil action ... that is timely made of record” in this proceeding. *Id.*

Petitioner’s approach to claim construction is to ignore it. Specifically, Petitioner states: “Nintendo does not believe that any claim constructions are needed for the purposes of this review.” (Pet. at 23.) Petitioner also ignores the patent’s lengthy history of prior constructions. To be sure, Petitioner recites that “[t]he claims of the ’941 Patent have been construed by three courts”⁴ and that at least one term—“license record”—was construed by the PTAB in the TCL IPR. (Pet. at 22-23.) But other than making a cursory note that prior constructions have occurred, Petitioner declines to address those prior constructions or explain why they should not be considered here in discerning the proper scope of the challenged claims.

Contrary to Petitioner’s misguided and erroneous approach to claim construction, Patent Owner submits that, under a proper construction of each of the

⁴ Petitioner’s count omits the Federal Circuit’s affirmation and reversal of several claim constructions of the district court in *Ancora Technologies, Inc. v. Apple Inc.*, No. 11:cv-06357 (N.D. Cal.); see *Apple*, 744 F.3d at 740.

following underlined Claim 1 terms, Petitioner’s arguments fail:

- (1) “a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area”; and
- (2) “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.”

Each of these limitations is a key component of the ’941 Patent’s advances over the prior art. And even though these phrases have been construed in the context of other disputes, such constructions have been ignored by Petitioner here.

For the reasons set forth below, Patent Owner respectfully requests that the following constructions be adopted.

A. “a computer”

The term “a computer” should be construed to mean “a digital data processor that includes one or more non-volatile memory and volatile memory areas.” Further, it should be understood to require that it be able to perform its claimed setting up and verifying functions without specialized or non-conventional “add-on” hardware.

Claim 1 of the ’941 Patent recites a method “for use with a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory.” The body of Claim 1 refers back to the antecedent “computer”—reciting “[s]electing a program residing in the volatile memory” of the computer, “using an

agent to set up a verification structure in the erasable, non-volatile memory of the BIOS” of the computer, and “verifying the program using at least the verification structure from the erasable non-volatile memory of the BIOS” of the computer.

Thus, even though it is recited in the preamble, the claimed “computer” is a claim limitation. (See Ex. 1013 at 11 (citing *Am. Med. Sys., Inc. v. Biolitec, Inc.*, 618 F.3d 1354, 1358 (Fed. Cir. 2010)).)

The specification also makes clear that the claimed “computer” is a “conventional” one, *i.e.*, one that does not require or utilize specialized or add-on hardware. (Ex.1001 at 1:27-32, 1:46-52 (distinguishing invention from “[h]ardware based products” that “validate authorized software usage by accessing a dongle that is coupled e.g. to the parallel port of the P.C.”).) Indeed, the specification defines “computer” “[i]n the context of the present invention” to be a “digital data processor” like those “found in personal computers, or on one or more processing cards in multi-processor machines,” and explains that the claimed computer therefore must include “a processor” containing the recited “a first non-volatile memory, a second non-volatile memory, and data linkage access to a volatile memory.” (Ex.1001 at 3:18-28 (noting that there may be one or more non-volatile and volatile memories).) The specification also teaches, of course, that the “volatile memory” may be “distributed into numerous physical components, using a bus, LAN, etc.” that collectively comprise the relevant “computer.” (Ex.1001 at 3:28-31.)

The prosecution history also confirms that the recited “computer” must be capable of performing the recited functions without specialized or “add-on” hardware. As the patentee explained when distinguishing the invention over Ginter:

Ginter et al. describe the architecture as add-on hardware which is named “SPU” Col. 64, lines 16-21 explicitly detail the fact that **the SPU is a hardware add-on, not part of the PC.** In col. 70 Ginter et al. describes the memory architecture for the SPU and uses terms taken from the PC engineering world. However, this is not referring to those actual PC components which name is used in their design.

(Ex.2011 at 88.) The patentee further confirmed that the invention required a conventional computer in the White Paper submitted during prosecution, which explained that the invention uses “nonvolatile memory chips (EEPROM) present in PCs” and “does not require any additional hardware to be added to the PC.” (Ex.2011 at 173.)

Given its use in the claims, specification, and prosecution history, the term “computer” as used “[i]n the context of the present invention” thus should be construed to refer to “a digital data processor that includes one or more non-volatile memory and volatile memory areas.” Consistent with the disclaimer in the specification, such computer also must be construed to require that it be able to perform its claimed setting up and verifying functions without needing to add any specialized or non-conventional hardware to the computer.

B. “using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS”

As noted above, in addition to claiming “a computer,” Claim 1 of the ’941 Patent also recites the need for such computer to include a software agent that is used “to set up a verification structure in the erasable, non-volatile memory of the BIOS.” As was recognized during prosecution, this limitation was a “key distinction” over the prior art software and hardware techniques discussed above, (Ex.2011 at 154-156), because the limitation (1) defines **how** the recited step must be performed (*i.e.*, using an unconventional and novel OS-level software agent residing at least in part on an otherwise conventional computer), (2) defines **what** that “agent” must set up (*i.e.*, “a verification structure”), and (3) defines **where** such verification structure is to be stored (*i.e.*, “in the erasable, non-volatile memory of the BIOS”).

Each construction of the “using an agent” limitation is addressed below.

1. “agent”

The Board should construe “agent” to be “an OS-level software program or routine running on the recited conventional computer to write and/or erase data” (*i.e.*, performing the task of writing and/or erasing data without the need for

additional or specialized hardware).⁵

Claim 1 of the '941 Patent recites: “**using an agent** to set up a verification structure in the erasable, non-volatile memory of the BIOS.” Thus, in addition to identifying what must be set up (the verification structure) and where such structure must be set up (in BIOS memory), the limitation defines who performs the step of storing a “verification structure”: the **agent**.

As the Federal Circuit and other courts uniformly recognize, this “agent” is an OS-level software program or routine running on the recited conventional computer to write and/or erase data (*i.e.*, it executes without the need for additional or specialized hardware). As demonstrated below, the claims, specification, prosecution history, prior constructions of other courts, and extrinsic evidence all support construing “agent” to be “an OS-level software program or routine running on the recited conventional computer to write and/or erase data.”

The “agent” is software that runs on a conventional computer: To begin, the “agent” necessarily is software. As the Petitioner recognizes, (*see* Pet. at 22-23), “agent” was previously construed by several district courts as a “software program or routine.” (Ex.1012 at 3; Ex.1013 at 28-36 (“the patent clearly limits agent to mean

⁵ Although not relevant to the Petition, the '941 Patent expressly contemplates that its steps “may be shared between the bureau and the computer.” (Ex.1001, 4:31-35.)

software as attested to by the Examiner”).) There is a reason why: the claim language and specification require it.

The language of Claim 1 is explicit. The method operates “with a computer including an erasable, non-volatile memory area of a BIOS of the computer.” The “agent” executes on the claimed “computer.” Nothing in the claims suggests that the agent is hardware; to the contrary, the only hardware disclosed is the “computer,” and nothing in the claims suggests the “agent” can be anything other than software executing on a conventional “computer.” As the Federal Circuit explained in *Apple*, “under the claims, it is indisputably the verifying software that interacts with the verification structure” stored in the BIOS. 744 F.3d at 736-37; *accord Traxcell Techs., LLC v. Nokia Sols. & Networks Oy*, 15 F.4th 1136, 1143–44 (Fed. Cir. 2021) (“[R]eciting ‘a computer’ ... that performs a function, and then further reciting that ‘the computer’ ... performs multiple additional functions, suggests that such ‘computer’ must be tied to all those functions.” (emphasis in original)).

The specification also confirms that the “agent” that is “us[ed]” to set up the verification structure must be “software” capable of operating on a conventional computer without the need for any additional or specialized hardware—i.e., the “expensive, inconvenient” prior art methods that the ’941 Patent expressly stated it overcame. (Ex.1001 at 1:27-32, 1:46-52 (distinguishing invention from “[h]ardware based products” that “validate authorized software usage by accessing a dongle that

is coupled e.g. to the parallel port of the P.C.”).)

The Federal Circuit’s analysis of the intrinsic record confirms that a key point of novelty of the ’941 Patent was making use of existing computer hardware, thereby disclaiming verification methods that would require “additional hardware”:

Thus, the inventors stated that their method makes use of the existing computer hardware (eliminating the expense and inconvenience of using additional hardware), while storing the verification information in a space that is harder and riskier for a hacker to tamper with than storage areas used by earlier methods.

Apple, 744 F.3d at 733-34; *accord HTC*, 908 F.3d at 1345.

Accused infringers have also agreed that the agent is software running on the computer (in circumstances where there was no incentive to dispute the plain meaning of “agent” in the context of the patent as OS-level software). Microsoft confirmed in its *ex parte* reexamination that the claimed “agent” was a “program to perform a task.” (Ex.2001 at 19.) HTC and its expert both adopted that same construction as well. (Ex.2003 at 89:2-10 (“‘Agent’ is broadly understood to be software.”); Ex.2005 at 22 (HTC describing the agent in its brief to the Federal Circuit as “*e.g.*, a software routine”).)

Extrinsic evidence from POSITAs reinforces that the “agent” comprises a software program or routine. Effectively every expert in each of the various prior challenges to the ’941 Patent has understood agent in this manner. (Ex.2011 at 142;

Ex.2003 at 66:14-17 (explaining “the patent examiner equated ‘programs’ ’with ‘agent.’”); Ex.2004, ¶10; *see also* Ex.2017.)

And various publications existing at the time of the invention confirm that “agent” is well-defined and understood in the computer industry as software. The Telecommunications Handbook (1999) states that “[a]n agent is a program, which ... performs tasks on behalf of a user or an application.” (Ex.2007 at 2-11.) Similarly, the Microsoft Press Computer User’s Dictionary (1998) states that an “agent” is “[a] program that performs a background task for a user [] when the task is done or some expected event has taken place.” (Ex.2008 at 13.) Other references demonstrate the same understanding. (Ex.2009, Microsoft Computer Dictionary (4th Ed. 1999) at 18-19 (same); Ex.2010, PC Magazine at 1 (An agent is “a software routine that waits in the background and performs an action when a specified event occurs.”).)

The “agent” is OS-level software: The intrinsic record also makes clear that the claimed “agent” is an OS-level software program or routine used by the existing computer. The specification describes the agent’s prescribed actions of “set[ting] up the verification structure” as “using E²PROM manipulation commands” to store, add and modify items in the “erasable, non-volatile memory of the BIOS.” (Ex.1001 at 1:65-2:9.) These are commands used to modify BIOS by OS-level software as described in the White Paper submitted by the applicant during prosecution.

(Ex.2011 at 173.)

The Federal Circuit confirmed that the disclosed “agent” is OS-level software when explaining that the intrinsic record disclosed that the “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—a combination that was not previously taught and that an ordinarily skilled application writer would not employ.” *Apple*, 744 F.3d at 735-36 (italics in original); *accord HTC*, 908 F.3d at 1345 (explaining that the “method calls for storage of a license record in a ‘verification structure’ created in a portion of BIOS memory that... ‘may be erased or modified’—for example, ... E2 PROM[], which may be altered by ‘using E2PROM manipulation commands.’” (citing ’941 Patent at 1:65-2:5).)

“The prosecution history confirms this understanding,” too. *Traxcell*, 15 F.4th at 1144. During prosecution, the Examiner in its Reasons for Allowance expressly described the “agent” as referring to an “OS-level software program or routine.” Indeed, the Examiner focused on that attribute of the invention as its point of novelty and the crux of its allowance decision, explaining:

[T]he 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 the BIOS to verify the program using the verification structure and having a user act on the program according to the verification. Further, it is well known to those of ordinary skill of

the art that a computer BIOS is not setup to manage a software license verification structure. The present invention overcomes this difficulty by using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS.

(Ex.2011 at 155.) In short, the Examiner expressly defined the “agent” as a “program[] running at the OS level”—that is, OS-level software.

The “agent” runs without additional or specialized hardware: The intrinsic record also confirms that the claimed “agent” runs on the recited conventional computer without the need for additional or specialized hardware.

The specification makes this clear by criticizing prior art techniques that required the use of additional or specialized hardware. (Ex.1001 at 1:27-32, 1:46-52.) As summarized by the Federal Circuit: the “inventors stated that their method makes use of **the existing computer hardware** (eliminating the expense and inconvenience of using additional hardware).” *Apple*, 744 F.3d at 733-34.

This clear disclaimer of methods that use “additional hardware” must be accorded its proper weight. It is well established that claims must be read in view of the specification, and “[o]ne purpose for examining the specification is to determine if the patentee has limited the scope of the claims.” *Watts v. XL Sys.*, 232 F.3d 877, 882 (Fed. Cir. 2000). The Federal Circuit thus routinely holds that:

Where the specification makes clear that the invention does not include a particular feature, that feature is deemed to be outside the reach of the

claims of the patent, even though the language of the claims, read without reference to the specification, might be considered broad enough to encompass the feature in question.

SciMed Life Sys. v. Advanced CardioVascular Sys., 242 F.3d 1337, 1341 (Fed. Cir. 2001).

The *SciMed* decision confirms the construction Patent Owner proposes here. In *SciMed*, the specification discussed only the coaxial lumen structure; it characterized the side-by-side structure as inferior; and it concluded that the coaxial lumen structure was used in “all embodiments of the present invention contemplated and disclosed herein.” *Id.* at 1344 (emphasis in original). Thus, even though the claim language could have been interpreted to include side-by-side lumen structure, the court felt that the specification made clear that it did not “fall within the scope of the claim language.” *Id.*; *accord Indivior Inc. v. Dr. Reddy’s Labs., S.A.*, 930 F.3d 1325, 1337 (Fed. Cir. 2019); *Meds. Co. v. Mylan, Inc.*, 853 F.3d 1296, 1309 (Fed. Cir. 2017).

The same is true here. Claim 1 expressly states that the method is “for use with a computer including an erasable, non-volatile memory area of a BIOS of the computer” and “us[es] an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS.” The only methods described in the specification are software-based methods. The specification characterizes hardware-based methods

as inferior, as the “inventors stated that their method makes use of the existing computer hardware (eliminating the expense and inconvenience of using additional hardware).” *Apple*, 744 F.3d at 733-34. And the only embodiments contemplated are ones that do not require add-on hardware to perform the verification function of the invention. The term “agent” thus must be understood to refer to an “an OS-level software program or routine running on the recited conventional computer to write and/or erase data” (i.e., running without the need for additional or specialized hardware).

The prosecution history likewise confirms that the “agent” is OS-level software running on a conventional computer without requiring any additional hardware. During prosecution, the claims were rejected based on “Ginter.” (Ex.2011 at 93.) In response, the applicant argued that Ginter “suffers from the deficiency” that was described in the specification, namely it required “add-on hardware”:

Moreover, in col. 70, line 23 – col. 71, line 25 Ginter et al. describe the architecture as **add-on hardware** which is named “SPU” (col. 63, line 66-col. 64, line 15). Col. 64, lines 16-21 explicitly detail the facts that **the SPU is a hardware, add-on not part of the PC.**

(Ex.2011 at 88.) The Examiner nevertheless maintained the rejection—stating that “[t]he Applicants do not teach the device necessary to edit an EEPROM nor have they made it clear to the Examiner how their system would be implemented in light

of the non-trivial processing required to write and erase its data.” (Ex.2011 at 102.)

The Examiner and Applicant subsequently held an interview where “112 corrections” and “editing independent claim language was discussed,” (Ex.2011 at 112), and the claims thereafter were “amended as agreed during the interview” to add the very “agent” limitation now at issue.⁶ (Ex.2011 at 120.) Simply put, the Applicant added the “agent” limitation for the very purpose of making clear that the claimed invention did not use or require additional or specialized hardware to accomplish its “set up” task; rather, it was to be performed using an OS-level software program or routine running on the conventional computer components that the specification disclosed.

The “agent” writes and/or erases data: Finally, the intrinsic record also demonstrates that the agent must be used to write and/or erase data (*e.g.*, to set up the verification structure in BIOS memory). First, the specification states expressly

⁶ Among other disclosures, in support of their amendment adding “agent” to Claim 1, the patentee submitted to the Examiner a 2001 White Paper authored by one of the inventors (Miki Mullor). That paper explained how, “[i]n order for a user to access the BIOS EEPROM[,] proprietary software” needed to be developed and how the resulting invention thus was “accomplished through software” such that it “**does not require any additional hardware** to be added to the PC.” (Ex.2011 at 173.)

that the agent must be used so that “the data in the second non-volatile memory may optionally be erased or modified (using E²PROM manipulation commands), so as to enable to add, modify or remove licenses.” (Ex.1001 at 2:1-5). The agent is the entity that performs the task of storing such data to the BIOS, e.g., to “add, modify or remove licenses.”

Second, as noted above, the applicant claimed the recited “agent” specifically to overcome the Examiner’s §112 rejection that “[t]he Applicants do not teach the device necessary to edit an EEPROM nor have they made it clear to the Examiner how their system would be implemented in light of the non-trivial processing required to **write and erase its data.**” (Ex.2011 at 102.) Thus, the prosecution history confirms what the specification already stated expressly: that the agent is what is to be used for the operations of writing and/or erasing BIOS memory as is required to perform the task of “set[ting] up a verification structure in the erasable, non-volatile memory of the BIOS.”

In sum, and as the Federal Circuit repeatedly has recognized, the specification and prosecution history make clear that the claimed “method makes use of the existing computer hardware (eliminating the expense and inconvenience of using additional hardware).” *Apple*, 744 F.3d at 733-34, *HTC*, 908 F.3d at 1345, and that the recited agent is an OS-level software program or routine that writes and/or erases data such that the verification structure and license record may be “stored in...the

computer's BIOS.” *HTC*, 908 F.3d at 1348–49.

Consistent with the claims, specification, prosecution history, extrinsic record, and constructions of other courts, the Board thus should construe “agent” to be “an OS-level software program or routine running on the recited conventional computer to write and/or erase data” (*i.e.*, running without the need for additional or specialized hardware).

2. “to set up a verification structure”

The Board should construe the phrase “to set up a verification structure” to require that the “verification structure” be distinct from the “license record.”⁷

Claim 1 of the '941 Patent recites: “using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS.” In addition to reciting how the step is to be performed (*i.e.*, using an OS-level software agent), the claim identifies what is to be “set up”: “a verification structure accommodating data that includes at least one license record.”

A key aspect of the recited limitation is that the “verification structure” is distinct from the “license record.”

⁷ The Board previously construed “license record” as “a record having information for verifying that licensed program.” (Ex.1016 at 6-9.) Petitioner does not dispute that construction. (Pet. at 23.)

The language of Claim 1 demonstrates that the terms are distinct—reciting that “the verification structure [must] accommodate[e] data that includes at least one license record.” The Federal Circuit recognized this very distinction—explaining that the “verification structure” and the “license record” are separate limitations with distinct purposes:

The 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 by interacting with the distinct computer memory that contains the program to be verified.

HTC, 908 F.3d at 1348–49.

In short, the Federal Circuit recognized that the two claim terms are different. The verification structure is a data structure or database that must be able to store or “contain[]” one or more “license record[s].” *Id.* Indeed, as the Federal Circuit succinctly explained “[t]he method calls for storage of a license record **in** a ‘verification structure’ created **in** a portion of BIOS memory.” *Id.* at 1345

The Federal Circuit’s description of the limitation is grounded in the specification, which describes the “verification structure” as a “database” that is distinct from the “license record[s] (LR)” that are contained within it. For example,

the specification refers to the “verification structure” repeatedly as a “database” or “database section,” explaining that the data structure contains the license records “LR” that are used in the verification process:

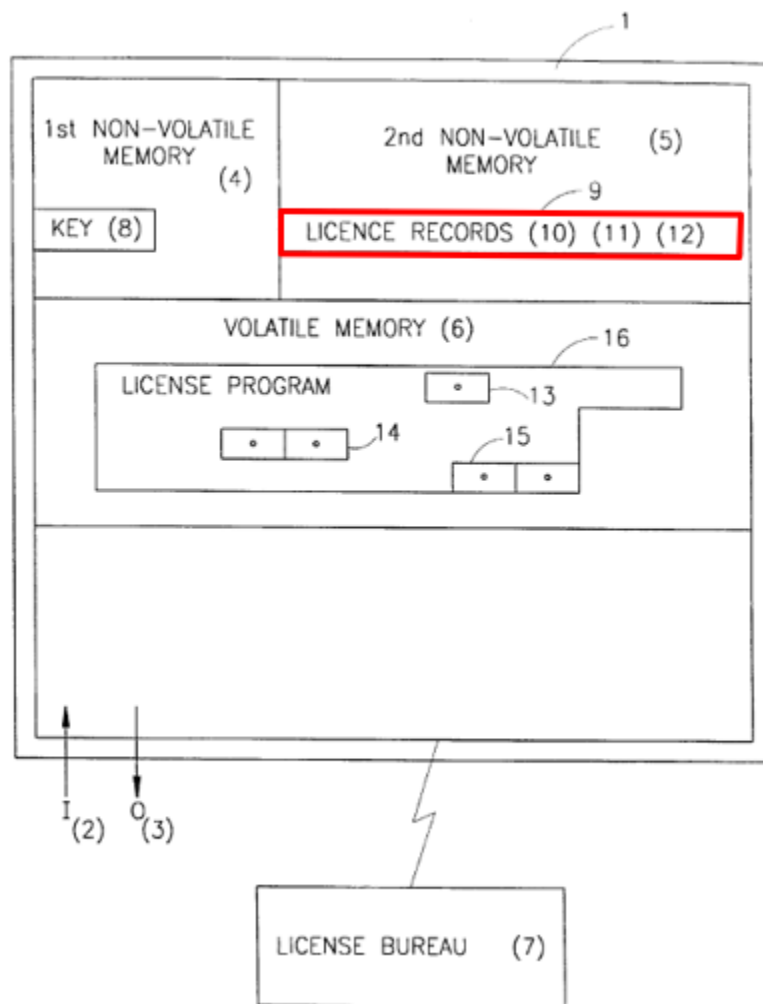
All or a portion the database contents (including of course $(LR)_{k1}$) that reside in the E²PROM portion in the first computer may be copied in a known per se means to the second computer....

Obviously, the value $(LR)_{k2}$ does not reside in the E²PROM database section of the second computer...

(Ex.1001 at 2:20-43, 2:52-53.)

The specification also states expressly that this “verification structure” database is distinct from the data it contains, which the specification separately refers to as, *e.g.*, a “license record (LR),” “data record,” “record,” or “data contents.” (*E.g.*, (Ex.1001 at 1:55, 2:22, 5:37, 6:25.)

The distinction between the “verification structure” database and “license record” data is also illustrated in Figure 1, as the license record area (9)—contains multiple license records (10), (11), (12) in the non-volatile memory.



(Ex.1001, FIG. 2, annotated.)

In sum, the claim language and the specification make clear that the verification structure is a data structure that is capable of storing one or more license records. The verification structure is not the license record itself.

The Board should thus construe “verification structure” to be distinct from “license record,” each of which must be “stored in...the computer’s BIOS.” *HTC*, 908 F.3d at 1348–49.

3. “memory of the BIOS”

The Board should construe the term “memory area of a BIOS” to mean the “area of memory used by the BIOS.”

Claim 1 of the ’941 Patent recites: “using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS.” The “using an agent...” limitation thus defines not only what must be set up, but also where the recited “verification structure” must be stored: “in the erasable, non-volatile memory of the BIOS” of the “computer” claimed in the preamble.

The memory recited in the claim is very specific; as the Federal Circuit acknowledged, the location where the “verification structure” is stored is “the memory space associated with the computer’s basic input/output system (BIOS), rather than other memory space.” *Apple*, 744 F.3d at 733 (citing various portions of the specification).

Storing the verification structure in BIOS memory was a “key distinction” over the inferior prior art techniques, as the Examiner expressly stated in its Reasons for Allowance. Ex.2011 at 154-155 (“[T]he 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 the BIOS.”). The specification also states that prior art “software” techniques (*i.e.*, techniques that operated exclusively at the OS level) were able to store license records to memory; they simply could not—and

indeed did not think to—store license records or the verification structures that contained them in BIOS memory. (Ex.1001 at 1:19-26.)

As the Federal Circuit recognized, using the “memory of the BIOS” location was significant to the ’941 Patent’s innovation, because:

Using BIOS memory, rather than other memory in the computer, improves computer security, the patent indicates, 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. *Id.*, col. 3, lines 4-17.

HTC, 908 F.3d at 1345.

The “memory of the BIOS” language also received particular attention in the Microsoft reexamination. There, the Examiner had no difficulty finding that the asserted prior art reference Schwartz did not invalidate the ’941 invention because Schwartz did not disclose storage of the verification structure in memory that was “used by” BIOS. Specifically, the Examiner explained that, “consistent with the specification of the ’941 patent,” memory of the “BIOS is therefore defined by the functional descriptive material contained within it” (Ex.2001 at 164) and is not defined solely by its physical structure. In particular, “one skilled in the art” would consider memory to be “memory of the BIOS” only if it was non-volatile memory

containing data “used by the functions of BIOS itself.”⁸ (Ex.2001 at 163-164.)

The key takeaway from the Examiner’s analysis is clear: the mere fact that data resides in non-volatile memory “does not necessarily mean that it is part of the BIOS.” (Ex.2001 at 164.) More is required to show that a specific memory area is “memory of the BIOS.”

Based on this proper understanding of “memory of the BIOS” as comprising memory “used by” the BIOS, *id.* the reexamination concluded with the Examiner confirming the patentability of all claims of the ’941 Patent, explaining that the prior art relied on the reexamination applicant—Schwartz—was “merely cumulative to the art cited by the Examiner during prosecution” and that no “reasonable examiner would have found this reference important in determining the patentability of claims 1-19” of the ’941 Patent because Schwartz did not disclose storing a verification structure in “memory of the BIOS”:

It is therefore the case that the table created cannot be considered to be in the BIOS either. Schwartz is therefore merely cumulative to the art cited by the Examiner during prosecution, insofar as it teaches to the claim limitations. It is NOT agreed that a reasonable examiner would

⁸ As noted previously, the Patent Office further explained that this does not preclude data stored in BIOS from “being also used or modified by programs located outside of the BIOS, such as applications running in an operating system.” (Ex.2001 at 164.)

have found this reference important in determining the patentability of claims 1-19.

(Ex.2001 at 165.)

For nearly identical reasons, in declining Samsung’s recent IPR petition—in which it raised Schwartz as one of its prior art references—the Board similarly concluded that the “memory of the BIOS” limitation was an important aspect of the ’941 invention. *Samsung v. Ancora*, IPR2020-01184, Paper 1 at 7-30 (PTAB Jan. 5, 2021). As in reexamination, the Patent Owner explained that “memory of the BIOS” was a key limitation such that:

[T]he mere fact that the EEPROM 250a contains a BIOS module does not mean the entirety of the EEPROM is the memory of the BIOS. Indeed, this Office has rejected this argument when Microsoft advanced it in the *Ex Parte* Reexamination. That makes sense. Such a remarkably broad construction is inconsistent with the clear meaning of “memory of the BIOS” and would render effectively *all* portions of any physical memory structure that contains a segment dedicated or used by the BIOS to be memory of the BIOS.

(IPR2020-01184, Paper 7 at 24, emphasis in original (footnote omitted).)

The Board agreed with—and expressly relied upon—that same reasoning in denying institution. The Board explained that the “memory of the BIOS” limitation is not met simply because BIOS is stored within some part of a physical memory module. Rather, the Board explained that one had to look at the function being

performed by the specific portion of memory at issue—explaining:

In its Petition, Petitioner takes the position that Schwartz’s EEPROM 250a teaches the claimed “non-volatile memory of the BIOS” under the Federal Circuit’s interpretation in *Ancora v. Apple* “because it stores *part of* BIOS module 309.” Pet. 9–11 n.4 (citing Ex. 1005, 8:17–19, Fig. 9) (emphasis added). However, other parts of Schwartz’s EEPROM 250a store configuration module 307 and zip/zone module 305. Ex. 1005, 8:17–19, Fig. 9. Petitioner does not allege that these other modules themselves are associated with the computer’s BIOS. Pet. 10–11. Significantly, Petitioner does not explain adequately why the entirety of EEPROM 250a, including the memory space that stores configuration module 307 and zip/zone module 305, is a “non-volatile memory of BIOS.” Pet. 10–14.

(Ex. 2016, IPR2020-01184, Paper 11 at 26, emphasis in original.)

As the specification, the prosecution history, the reexamination record, and the Board’s reasoning in the Samsung IPR all make clear, the “memory area of a BIOS” limitation refers to a specific portion of memory: the memory space used by the BIOS. To hold otherwise would negate the meaning of the phrase “of the BIOS” and dismantle the “key distinction” of the present invention, including that the danger associated with tampering memory used by the BIOS is one aspect that demonstrated the invention’s novelty over the prior art. (Ex.1001 at 3:4-17; *HTC*, 908 F.3d at 1345; *Apple*, 744 F.3d at 733.)

Accordingly, “memory area of a BIOS” should be construed as “area of memory used by the BIOS.”

V. The Asserted Prior Art

A. Hellman

Hellman discloses a method and an apparatus in which use of a software package can be authorized for a particular base unit a specific number of times. (Ex.1004 at 4:37-40.)

Notably, the “authorization[]” methods disclosed in Hellman, (Pet. at 24), are predicated entirely on a hardware system that requires both modification of and additional hardware beyond what is included in a conventional computer. Indeed, the Board has already concluded that “the evidence in this current record suggests that Hellman’s agent is a combination of software and hardware implementation.” *Sony Mobile Commc’ns Ab v. Ancora Techs., Inc.*, No. IPR2021-00663, Paper 17, at 28 (PTAB June 10, 2021). The record here is no different, and Petitioner does not claim otherwise. See Pet. at 41 (relying on same evidence as relied on in *Sony*). As such, Hellman does not disclose an OS-level software agent, let alone one running on a conventional computer.

Hellman demonstrates this. The core “Base Unit 12”—which is the entity Petitioner identifies as the “computer” that allegedly performs the authorization steps, Pet. at 36—is disclosed as a unit that is physically modified with custom

hardware components that must be used to carry out Hellman's disclosed authorization process. Figure 8 (and its accompanying descriptions) illustrate this, showing within "Base Unit 12" a physical hardware "switch 41" standing between "update unit 36" and "player 42."⁹

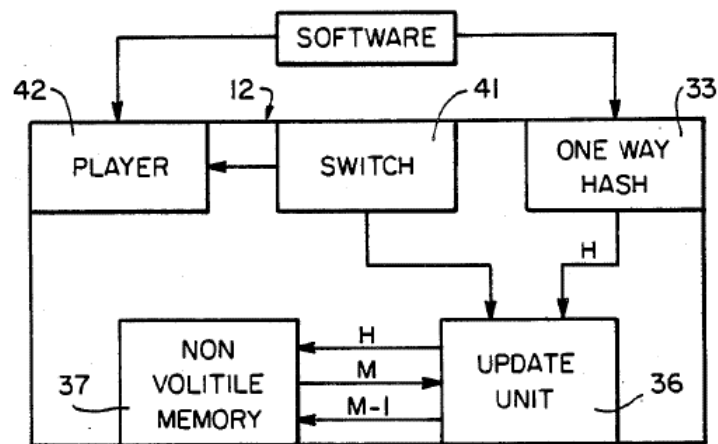


FIG_8

The specification makes clear that both the "switch 41" and the "update unit 36" are specialized add-on hardware not found within a conventional computer. Hellman expressly states that "switch 41" is not part of a conventional computer, because it explains that "Base unit 12 should be physically constructed so that switch 41 is not readily accessible to the user." (Ex.1004 at 11:4-19.) It then explains that

⁹ Hellman discloses "if the software is a computer program, then software player 42 would be a microprocessor or central processing unit (CPU)." (Ex.1004 at 11:1-3.)

“update unit 36” must be physically constructed in such a way so that “the wire sending the control signal from update unit 36 to software player 42” cannot be cut, which can be accomplished by “use of epoxy to encapsulate switch 41 and the wire connecting it to update unit 36.” (*Id.*) These physical, hardware-based modifications are required to prevent tampering by a user. (*Id.*)

In addition, Hellman discloses a numeric value “M” stored in a “non-volatile memory 37” as representing “the number of uses of software package 17 which are still available.” (Ex.1004 at 10:30-43.) As annotated below, Hellman states when “M is greater than 0” the “update unit 36 sends a control signal to switch 41 which activates software player 42, allowing it to use software package 17.” (Ex.1004 at 10:44-49.) Once the “player 42” (*i.e.*, CPU) activates the software, the “update unit 36” decrements M (*i.e.*, M-1) and stores this new value in the “non-volatile memory 37.” (*Id.*)

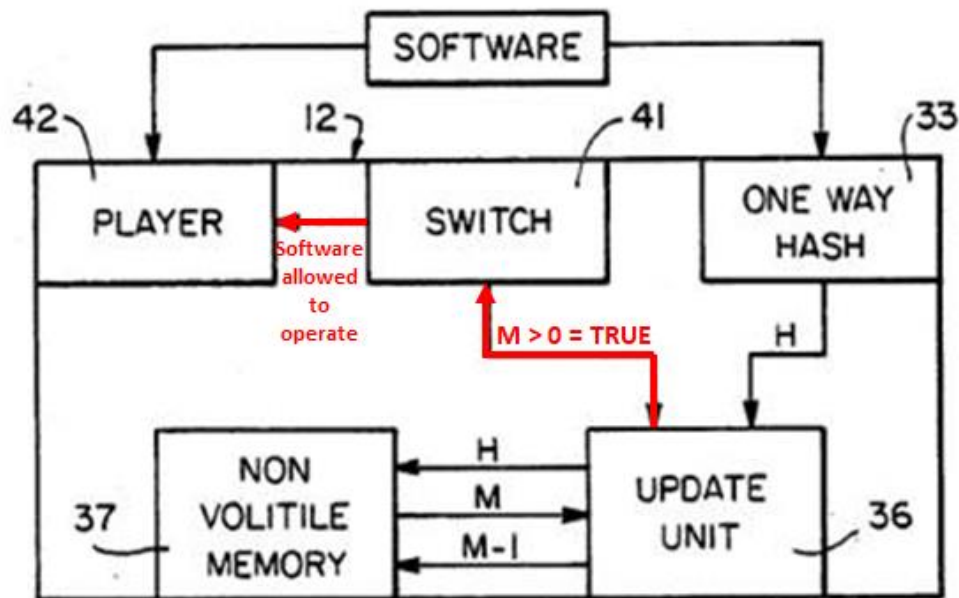


FIG_8

(Ex.1004, Fig. 8, annotated and emphasis in red added.)

Hellman thus discloses that the invention cannot work without this additional hardware (and the manner in which the hardware is manufactured), which ensures that a user cannot physically bypass Hellman's physical authorization mechanism. This is illustrated in the above annotations. Hellman requires additional hardware (i.e., “switch 41”) between the CPU and “update unit 36” for authorizing software.

In addition to requiring specialized, add-on hardware, Hellman also fails to disclose an OS-level software program or routine running on the recited conventional computer to write and/or erase data to set up a verification structure for at least two reasons.

First, Hellman discloses neither an operating system nor that the “update unit 36” is software running at the OS-level. To the contrary, Hellman discloses “update unit 36” as specialized hardware that operates only to provide a signal to the CPU (“software player 42”) which in turn allows the to-be-verified software program to run. *See Sony*, IPR2021-00663, Paper 17 at 28.

Second, and as explained in greater detail below, Hellman does not disclose a “verification structure.” Its control signal “M” is not a “verification structure” that includes or contains a “license record” stored in BIOS memory. Instead, control signal “M” is at most a numerical counter (an integer value of “0,” “1,” “2”...) that is transmitted to the CPU indicating the number of allowable uses of a software package.

B. Chou

Chou discloses two alternative embodiments for carrying out its claimed anti-theft methods. Critically, however, neither contemplate or disclose OS-level software storing a verification structure in BIOS memory. Instead, Chou discloses the very prior art techniques the ’941 Patent improved over: (1) hardware-based techniques and (2) a password-lock software-based anti-theft technique.

Again, the distinctions between Chou and the ’941 invention are clear from Chou itself. Chou’s first embodiment requires use of an “external ROM 19,” *i.e.*, a key (illustrated below), that serves to “prevent[] operation of the personal computer

when it [external ROM 19] is disconnected from the computer.” (Ex.1005 at 2:58-61.)

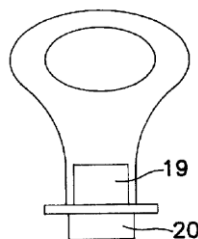


FIG.2

Chou’s external hardware key must be “connected to the serial port 16” of the computer as it “contains information necessary to permit the BIOS routine stored within BIOS memory 15 to complete execution.” (Ex.1005 at 4:49-51; *see also* 3:36-44; 4:42-58.) As illustrated by Figure 5’s flow chart, Chou explains how the external hardware key is used to restrict the computer’s operation. Chou recites how, at step 44, the BIOS may determine if the computer is “locked” such that the external hardware key (“ROM 19”) is required. If so, Chou executes steps 45-51 to verify the contents of “external ROM 19” – *i.e.*, the key. (Ex.1005 at 5:36-52.) Only if the “external ROM 19” is verified will Chou then allow the BIOS to complete the boot process. (Ex.1005 at 5:53-62.)

In short, Chou’s first embodiment is one of the “expensive, inconvenient” “hardware-based products” that the ’941 Patent improved over. (Ex.1001 at 1:27-30.)

Chou's second (alternative) embodiment likewise discloses nothing more than one of the prior art techniques that the '941 Patent improved over. Instead of relying on hardware, the second method "instead[]" relies upon a user entered password for protection." (Ex.1005 at 7:14-17.) This embodiment thus requires "a user entered password instead of an externally connected key to enable complete execution of the BIOS routines." (Ex.1005 at 7:20-21.)

Critically, this alternative embodiment does not involve the use of an OS-level software agent to store a verification structure in BIOS memory; to the contrary, all of the necessary data structures are created at the time of manufacturing and can only be implemented prior to the installation of BIOS memory onto the computer's motherboard. As Chou explains:

The process of creating the security function as part of the contents of the BIOS memory is illustrated in FIG. 6.... The BIOS memory is prepared by storing within it, the [BIOS] routines illustrated in FIG 3.... Further, the BIOS memory is loaded with the computer I.D. number for which it is to be installed, as well as the public key derived in step 65.

Once the routines are **loaded for effecting the functions of FIG. 4** [the 'security function'] the BIOS memory is installed in the mother board of the computer 10 in step 67.

(Ex.1005 at 6:20-35.)

Only by “prepar[ing]” the BIOS memory during manufacture to effectuate its security functions—prior to “install[ation] in the mother board”—can Chou allow a user to later enable the “lock mode” after the computer’s distribution. (Ex.1005 at 9:1-12.) If the user “select[s] the lock state 112 each subsequent boot up of the computer 10 will request a password verification from the user.” (*Id.*; Ex.1005 at 9:13-17; Figure 10; 2:46-48.). The password protects or locks the operation of BIOS using passwords stored in structures created at manufacture, nothing more. There is nothing in Chou about using an “agent” to set up a verification structure or license record in BIOS memory.

In sum, neither of Chou’s two embodiments discloses using an OS-level program to set up a verification structure in BIOS memory, which is then used to verify software.

C. Schneck

Schneck is another hardware-based system and is irrelevant to Patent Owner’s preliminary response because Petitioner does not rely on Schneck as disclosing, alone or in combination, any part of Claim 1. As Petitioner states, it relies on “Hellman [to] disclose[] most of the features of claim 1, and Hellman in combination with Chou [to] disclose[] all of the features of claim 1. Schneck contains teachings relevant to various dependent claims.” (Pet. at 45.)

VI. The Asserted Combinations Fails To Disclose Key Elements of Claim 1 of the '941 Patent

The Petition relies on two combinations as purportedly demonstrating that each claim of the '941 Patent is obvious: (1) the combination of Hellman and Chou (Ground 1) and (2) the combination of Hellman, Chou, and Schneck (Ground 2). The Petition is wrong. Among other deficiencies, none of the three references discloses the limitations “a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area” or “using an agent to set up a verification structure 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.”

A. The Only “Agent” Petitioner Identifies is Hellman’s “Update Unit 36,” which Requires “Add-on Hardware” and Thus Fails to Disclose the “Using an Agent...” Limitation

The Petition first should be denied because none of the proposed combinations teach or disclose the “using an agent” limitation.

The Petition relies solely on Hellman as purportedly disclosing this limitation. Specifically, Petitioner claims that Hellman’s “update unit 36 act[s] as the required ‘agent.’” (Pet. at 40-41.) It does not. Nothing in Hellman teaches “an OS-level software program or routine,” let alone one that “run[s] on the recited conventional computer” without the need for specialized or “add-on” hardware as required.

Hellman does not disclose the use of an OS-level software program at all. As explained above at pp. 21-31, Hellman teaches only the use of specialized hardware to perform its methods. *See Sony*, IPR2021-00663, Paper 17 at 28.

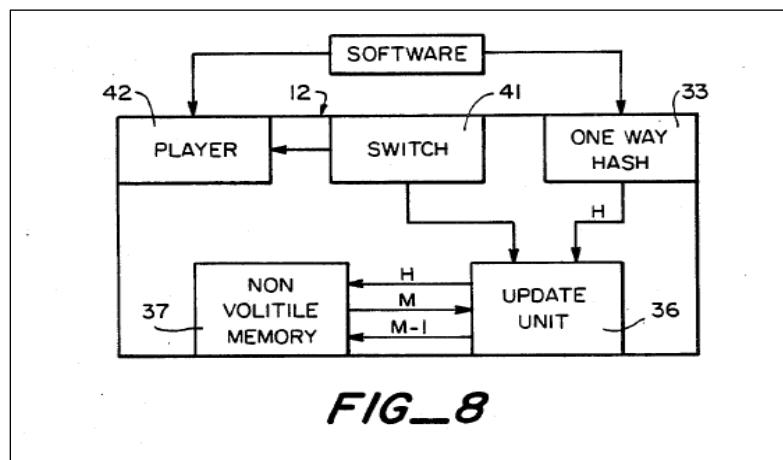
The specialized add-on hardware Hellman discloses is “update unit 36,” which Hellman explains is a component of “base unit 12.” (Ex.1004 at 11:4-19; Ex.2013, ¶¶89-90.) The fact that “update unit 36” is specialized hardware is made clear from every one of its descriptions in the specification, each of which confirms that it is specialized hardware. (Ex.1004, *e.g.*, 11:4-8.)

Hellman extensively emphasizes that it is directed to a hardware-based approach rather than “an OS-level software program or routine running on the recited conventional computer to write and/or erase data” as Claim 1 of the ’941 Patent requires:

Base unit 12 should be physically constructed so that switch 41 is not readily accessible to the user. Otherwise a user could cut the wire sending the control signal from update unit 36 to software player 42 and force the control signal to always activate software player 42. The use of epoxy to encapsulate switch 41 and the wire connecting it to update unit 36 would be one such approach.... Detection of tampering is a simpler task and techniques are well known for accomplishing this task. For example, many warranties are void if equipment is opened by the user. Manufacturers of such equipment use special factory seals or a

dab of paint across a joint to tell if a piece of equipment has been opened.

(Ex.1004 at 11:4-19.) The need for the added “update unit 36” hardware component also is clear just from Figure 8, which depicts the “update unit 36” as central to every other disclosed interaction required by Hellman’s authorization processes:



In short, Hellman discloses nothing more than the additional hardware-based approaches that the patentee expressly disclaimed and specifically added the “using an agent” term to exclude.

Additionally, a POSITA would understand that Hellman’s use of the word “software” refers not to an “agent to set up a verification structure,” but rather to the software package being authorized by the base unit. (Ex.2013, ¶93.) The Federal Circuit explained the distinction between the claimed “agent” and this to-be-verified software “program” by explaining that “‘program’....refers to the to-be-verified

software,” while the “agent” is the “verifying software that interacts with the verification structure.” *Apple*, 744 F.3d 732, 735-36. “The to-be-verified software is different from the verifying software.” *Id.*

Nor would a POSITA understand Hellman as disclosing the recited “OS-level software program or routine” that is used to set up a verification structure. The Wolfe Declaration simply speculates that the update unit 36 may be a software routine that potentially operates along with a hardware module—but identifies nothing in Hellman to support that proposition. (Ex.1003, ¶137.) There is a reason. The Wolfe Declaration states that “**Hellman does not specifically disclose how update unit 36 is implemented.**” (*Id.*)

In sum, when “agent” is properly construed as “an OS-level software program or routine running on the recited conventional computer to write and/or erase data,” there can be no doubt that Hellman fails to disclose the required “agent.” And because Petitioner relies on nothing other than Hellman to satisfy the “agent” limitation, Hellman’s failure to teach an “agent” means that Petitioner has failed to “present evidence and arguments sufficient to show that it is reasonably likely that it will prevail in showing unpatentability.” PTAB Rules of Practice for Instituting on All Challenged Patent Claims, 85 Fed. Reg. 79,120, 79,121; *id.* at 79,125 (“[T]he burden is on the petitioner to meet the applicable standard that it would prevail with respect to at least one of the claims challenged in the petition.”). For this reason

alone, the Petition should be denied.

B. Petitioner’s Proposed Combination Does Not “Set Up A Verification Structure” and Petitioner Fails to Disclose How it is “Set Up” by the “Agent”

The Petition also should be denied because the proposed combination does not teach a “verification structure” or using an agent to “set up a verification structure.”

As with the “agent” limitation, the Petition relies solely on Hellman for the “set up a verification structure” limitation. But Hellman at most discloses storing an alleged “license record”; it does not disclose “set[ting] up” a “verification structure” in the memory of the BIOS.

Petitioner’s argument for how Hellman allegedly discloses a “verification structure” conflates the meaning of “verification structure” and “license record” by arguing that storing a “license record” at a specific memory address satisfies both limitations. Specifically, Petitioner claims that both limitations are taught by the same single disclosure: the existence of “value M at a specific address H.” (Pet. at 41 (citing Wolfe Decl. ¶¶133-138).) The Petition explains its reasoning as follows:

[Hellman’s] value M is the required “license record”, because it indicates the scope of the authorized use – the number of uses, where “M” is the number – for the specific software package 17 identified by hash value H. *Id.*

Storing the value M at the address H constitutes setting up a verification structure because it includes storing a license record at a specific license record location...

(*Id.*) Petitioner thus argues that “M” (numerical-value data corresponding to the number of remaining uses) is the “license record” and further argues that the existence or modification of “M” at a particular address “H” constitutes “setting up a verification structure.”

Petitioner is wrong. Even accepting Petitioner’s descriptions of Hellman, it does not disclose “set[ting] up a verification structure ... the verification structure accommodating data that includes at least one license record” for at least three reasons. (Ex.1001, Claim 1.)

First, as Claim 1’s plain language makes clear, the limitation requires both that specific data (a “license record”) along with a specific data structure that contains the data (a “verification structure”) be stored in BIOS memory. Petitioner’s description of Hellman points only to the storage of data (an alleged “license record”), and identifies nothing in Hellman showing the storing or establishing of any data structure (a “verification structure”) in BIOS memory.

Indeed, Petitioner cannot make such an argument because it admits—as it must—that no “verification structure” is ever stored or set up in Hellman. Value “M” is merely “stored in plaintext at an address location determined by hash value H.”

(Pet. at 46; Ex.1003, ¶120E (describing memory where “M” is stored as a pre-established “range of acceptable memory addresses”).) Petitioner does not explain how a memory address location is a “verification structure,” nor does it explain how saving a plaintext value at such a pre-existing memory address constitutes “set[ting] up a verification structure.”

Petitioner’s arguments as to other limitations of Claim 1 also show why the “verification structure” cannot be merely an “address location.” (Pet. at 46.) The verifying step of Claim 1 requires “[v]erifying the program using at least the verification structure.” (Ex.1001, Claim 1.) Petitioner argues that this limitation is satisfied by “using the value M [the alleged license record]...to verify software package 17.” (Pet. at 42.) But the claim requires “using at least the verification structure,” and all that Petitioner has identified as being “used” is “value M”—*i.e.*, the alleged “license record.” Petitioner makes no attempt to argue or explain how the “address location” where “M” is stored is in any way “used” in the verifying step.

Additionally, the specification shows that the “verification structure” is a data structure that itself must be stored in memory; it is not the memory address itself. As noted above, the specification’s embodiments describe the “verification structure” as a “database” or “database section” that is distinct from the “license record (LR)” that is contained within it. (Ex.1001 at 2:20-43; *see id.* at 1:55, 2:22, 5:37, 6:25.)

Like the claims, the specification makes clear that the “verification structure” and “license record” are distinct terms with distinct meanings.

Second, by claiming that the step of storing value “M” in address location “H” satisfies the limitations of both setting up a “license record” and setting up a “verification structure,” (Pet. at 41), Petitioner renders the term “set up a verification structure” void and meaningless. Because a memory address is required for any data to be stored, any act of storing data (e.g., a “license record”) would necessarily satisfy the act of storing a data structure (e.g., a “verification structure”), and thus render the latter limitation entirely superfluous. But “[i]t is highly disfavored to construe terms in a way that renders them void, meaningless, or superfluous.” *Wasica Fin. GmbH v. Cont’l Auto. Sys., Inc.*, 853 F.3d 1272, 1288 n.10 (Fed. Cir. 2017).

Petitioner cannot demonstrate a reasonable likelihood of showing unpatentability by treating two different terms as meaning the same thing. *Applied Med.*, 448 F.3d at 1333 n.3 (“[T]he use of two terms in a claim requires that they connote different meanings....”). Yet that is exactly what is required under Petitioner’s theory of how Hellman discloses setting up both a “license record” and a “verification structure” simply by storing data (“M”) in a memory address location.

Third, that Petitioner nowhere identifies an “agent” as performing the “set up a verification structure” step is also highlighted by Petitioner’s arguments with

respect to dependent Claim 7. Claim 7 specifies certain actions the agent is to perform in setting up the verification structure, yet Petitioner's own expert admits that the asserted agent (Hellman's "update unit 36") performs none of Claim 7's recited steps.

Specifically, Claim 7 recites: "A method according to claim 6 wherein **using an agent to set up the verification structure includes** the steps of: [i] establishing or certifying the existence of a pseudo-unique key in a first non-volatile memory area of the computer; and [ii] establishing at least one license-record location...." (Ex. 1001, Claim 7 (brackets added).) Relying entirely on the Wolfe Declaration, Petitioner argues that a "modified update unit 36 [from Hellman] and/or other associated memory access modules" would have been used to perform Claim 7's sub-steps for the "using an agent" limitation. (Pet. at 60 citing Ex.1003, ¶¶190-196.)

But Petitioner misrepresents its own evidence: nowhere in the Wolfe Declaration's discussion of Claim 7 does Dr. Wolfe state that "update unit 36" is involved in Claim 7's "establishing" or "certifying" steps. (Ex.1003, ¶¶190-196.) Dr. Wolfe never mentions "update unit 36" in discussing Claim 7 at all. Petitioner thus relies only on attorney argument to argue that Hellman's alleged "agent" is involved in performing Claim 7's steps. And "attorney argument is not evidence." *Elbit Systems of America, LLC v. Thales Visionix, Inc.*, 881 F.3d 1354, 1359 (Fed. Cir. 2018). This failing further underscores that nothing in the Petition shows how

the asserted combination teaches “**using an agent** to set up a verification structure” step.

In sum, Petitioner fails to show how Hellman discloses an “agent” that is used to “set up a verification structure.” Petitioner presents no evidence showing how Hellman’s alleged “agent” performs the act of “set[ting] up a verification structure”—a failing that holds true whether or not the Board adopts Patent Owner’s proposed construction for “setting up a verification structure.” And it fails to show how the same act of storing data to a memory location can satisfy both the “license record” and “verification structure” limitations without rendering the latter claim term void and meaningless. Because Petitioner relies exclusively on Hellman for this limitation—and does not rely on any other reference alone or in combination with Hellman—each of these failings is an independent basis to deny the Petition.¹⁰

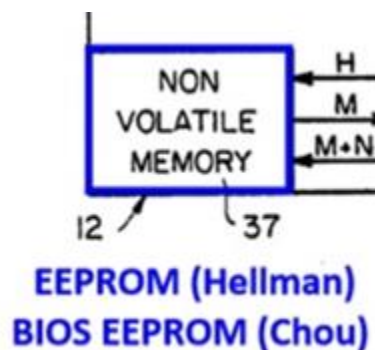
C. The Proposed Combination Fails to Teach Using “Memory of the BIOS” for Storing a “Verification Structure” or “License Record”

The Petition also fails because it identifies nothing in Chou showing that the

¹⁰ Although Petitioner’s Ground II asserts the Schneck reference, the Petition relies on only Hellman and Chou for its arguments as to Claim 1. (Pet. at 45, 50; *see In re Magnum*, 829 F.3d at 1381 (holding that the Board may not “raise, address, and decide unpatentability theories never presented by the petitioner.”))

asserted “license record” or “verification structure” is stored in “memory of the BIOS.”

Petitioner alleges that a POSITA would modify Hellman’s “memory 37” (based on the teaching of Chou) such that it satisfies the recited “memory of BIOS.” Petitioner even annotates Hellman’s “memory 37” as being the same BIOS EEPROM disclosed by Chou.



(Pet. at 31.)

Such a modification is necessary because Petitioner states that Hellman’s value “M” is the claimed “license record” that is stored in the “memory of the BIOS” in the alleged combination. Petitioner alleges the modification would have been possible because, according to Petitioner, EEPROM memory could have been used to store BIOS routines and non-BIOS data (*i.e.*, the “verification structure”). In particular, Petitioner states that “it was common practice to *store more than one thing in a single memory module* in a computer” and that a “single EEPROM module would have had sufficient space to store both the BIOS and other ancillary

information, like the license information from Hellman.” (Pet. at 33-34 citing Ex.1003, ¶¶112-116, 119-120.)

Petitioner’s argument should be rejected for at least two independent reasons.

First, nothing in Chou, Hellman, or their combination teaches storing the alleged “M” value in the memory area that is “used by the BIOS,” as the “memory of the BIOS” term requires. Instead, Petitioner’s argument and proposed combination assumes that the entirety of Chou’s EEPROM chip constitutes “memory of the BIOS,” such that storing data in EEPROM is the same as storing it in BIOS memory.

This argument, however, has been considered and rejected twice—once by the Board (in Samsung’s IPR Petition) and once by the Examiner (in Microsoft’s *ex parte* reexamination).

Specifically, Microsoft and Samsung both argued that U.S. Patent No. 6,153,835 (“Schwartz”) disclosed storing BIOS in EEPROM. (Ex. 2001; IPR2020-01184, Paper 1 at 7-30.) In particular, both argued that storing verification data on the same memory module where the BIOS was stored constituted storing it in “memory of the BIOS,” as long as any portion of that memory also stored one or more BIOS routines.

The Board rejected that reasoning, explaining that storing license-verification data on the same EEPROM where BIOS data is also stored is insufficient to render

the entire EEPROM “memory of the BIOS” because there is no indication that the location where the license-verification data is stored is used by the BIOS or, in the Board’s words, “associated with the...BIOS”:

Petitioner takes the position that Schwartz’s EEPROM 250a teaches the claimed “non-volatile memory of the BIOS” under the Federal Circuit’s interpretation in *Ancora v. Apple* “because it stores *part of* BIOS module 309.” ... However, other parts of Schwartz’s EEPROM 250a store configuration module 307 and zip/zone module 305. Ex. 1005, 8:17–19, Fig. 9. Petitioner does not allege that these other modules themselves are associated with the computer’s BIOS.... Significantly, Petitioner does not explain adequately why the entirety of EEPROM 250a, including the memory space that stores configuration module 307 and zip/zone module 305, is a “non-volatile memory of BIOS.

(Ex.2016, IPR2020-01184, Paper 11 at 26, italics in original.) Petitioner makes the same mistake here, failing to explain why the entirety of Chou’s EEPROM constitutes “memory of the BIOS.”

And Petitioner’s failing here is even more acute because Patent Owner’s proposed construction requires “memory of the BIOS” to be “area of memory used by the BIOS.” Petitioner identifies no disclosure in Chou, Hellman, or their combination that teaches storing Hellman’s value M in an “area of memory used by the BIOS.”

Indeed, Petitioner makes no argument that any of the memory areas where

non-BIOS data is stored—*i.e.*, “ancillary information” according to Petitioner, *see* Pet. at 34—is in any way used by the BIOS. To the contrary, the only “ancillary information” or data that Petitioner identifies is a user “password stored in the BIOS memory.” (*Id.* at 29, 34.) But Petitioner makes no argument that Chou or its combination with other references teaches that the location where such passwords are stored is in an “area of memory used by the BIOS.”

Second, Petitioner’s argument fails for another independent reason, too: Chou teaches away from using a software agent to set up a verification structure in BIOS memory. Specifically, Chou’s description of Figure 7 reflects that the “configuration of the BIOS EEPROM” provides for no spare memory space for functions and routines other than those already prescribed by Chou:

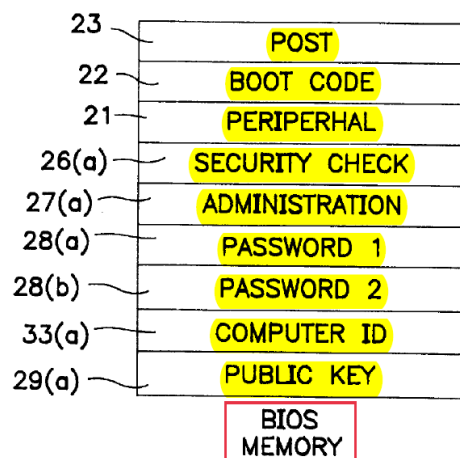


FIG. 7

(Ex.1005 at 7:15-35, Fig. 7, annotated.) As such, there is no space available in Chou’s “memory of the BIOS” to store Hellman’s value “M” or any other data.

Tellingly, Petitioner concedes that Chou’s “BIOS MEMORY” cannot accommodate Hellman’s value “M” —arguing only that Hellman’s value “M” would be stored “[i]n EEPROM memory” generally, (Pet. at 31, *see id.* at 32-35), rather than in Chou’s “BIOS MEMORY” specifically (as illustrated in Chou’s Figure 7). That is fatal. Because any purported combination of Hellman and Chou will result in Hellman’s value “M” being stored on EEPROM generally, it would not disclose storage in memory that is even associated with “BIOS Memory” —much less used by it as Claim 1 requires.

Moreover, Chou’s “BIOS Memory” cannot simply be modified to accommodate Hellman’s value “M.” As Chou explains, to the extent BIOS memory can be used for storing “ancillary” non-BIOS data at all, such memory configurations are in fact preconfigured prior to distribution to users:

The process of creating the security function as part of the contents of the BIOS memory is illustrated in FIG. 6.... The BIOS memory is prepared by storing within it, the routines illustrated in FIG 3.... Further, the BIOS memory is loaded with the computer I.D. number for which it is to be installed, as well as the public key derived in step 65.

Once the routines are loaded for effecting the functions of FIG. 4 [‘security functions’], the BIOS memory is installed in the mother board of the computer 10 in step 67.

(Ex.1005 at 6:20-35.) In short, Chou discloses that “configuration of the BIOS

EEPROM” is done at manufacture, such that altering it in order to write Hellman’s value “M” to Chou’s “BIOS Memory” would require de-coupling the EEPROM memory module from the chip, returning it to the manufacturer, and reconfiguring the “BIOS EEPROM.” That is the antithesis of using a software agent to perform the required “using an agent” limitation.

In sum, because Chou’s configuration provides no room for any additional “non-BIOS data,” and because configuring the BIOS to permit additional non-BIOS data to be stored “as part of the contents of BIOS memory” requires “prepar[ing]” the “BIOS memory” prior to its installation in the motherboard, Chou teaches away from the ’941 Patent’s claimed method of using a software agent to “stor[e] a license record in a ‘verification structure’ created in a portion of BIOS memory.” *HTC*, 908 F.3d at 1345. Neither Chou nor Petitioner’s asserted combination with Hellman thus discloses using an agent to set up a verification structure in BIOS memory.

D. Petitioner Does Not Assert That Schneck Discloses Either Claim 1 Limitation at Issue

Finally, even though Petitioner’s Ground II asserts a combination of Hellman, Chou, and Schneck, Petitioner disclaims any reliance on Schneck as disclosing Claim 1 or any of its limitations. (Pet. at 45 (noting it relies only on “Hellman” for “most of the features of claim 1” and “Hellman in combination with Chou [for]...all of the features of claim 1” and explaining that “Schneck contains teachings relevant

to various dependent claims”; *id.* at 50.) The Board thus does not need to consider Schneck at all. *In re Magnum*, 829 F.3d at 1381 (Board cannot “adopt arguments on behalf of petitioners that could have been, but were not, raised by the petitioner during an IPR” but “must base its decision on arguments that were advanced by a party”).

VII. Conclusion

Because the asserted prior art fails to disclose key limitations of the claimed invention such as “using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS,” the Board should deny the Petition for *inter partes* review of the ’941 Patent.

Dated: November 18, 2021

Respectfully submitted,

/John P. Rondini/
John P. Rondini (Reg. No. 64,949)
Marc Lorelli (Reg. No. 43,759)
BROOKS KUSHMAN P.C.
1000 Town Center, 22nd Floor
Southfield, MI 48075
(248) 358-4400

*Attorneys for Patent Owner
Ancora Technologies Inc.*

Certificate of Service

The undersigned hereby certifies that on November 18, 2021, a complete and entire copy of **PATENT OWNER'S PRELIMINARY RESPONSE TO PETITION FOR INTER PARTES REVIEW and all Exhibits**, was served via electronic mail to the attorneys listed below at PerkinsService-Nintendo-Ancora-IPR@perkinscoie.com:

LEAD COUNSEL	BACK-UP COUNSEL
Jerry A. Riedinger Reg. No. 30,582 riedinger-ptab@perkinscoie.com Perkins Coie LLP 1201 3rd Ave., #4900 Seattle, WA 98101 Telephone: 206-359-8664	Jose Villarreal, Reg. No. 43,969 villarreal-ptab@perkinscoie.com Perkins Coie LLP 500 W 2nd St 1900 Austin, TX 78701 Phone: 737-256-6122 Kyle Canavera, Reg. No. 72,167 canavera-ptab@perkinscoie.com Perkins Coie LLP 11452 El Camino Real, Suite 300 San Diego, California 92130 Phone: 858-720-5700 Theresa H. Nguyen (to be admitted <i>pro hac vice</i>) nguyen-ptab@perkinscoie.com Perkins Coie LLP 1201 Third Avenue, Suite 4900 Seattle, WA 98101-3099 Phone: 206-359-6068 Tara Kurtis, Reg. No. 74,846 kurtis-ptab@perkinscoie.com Perkins Coie LLP 131 S. Dearborn Street, Suite 1700 Chicago, IL 60603-5559 Phone: 312-324-9607

/John P. Rondini/

Marc Lorelli (Reg. No. 43,759)
John P. Rondini (Reg. No. 64,949)
BROOKS KUSHMAN P.C.
1000 Town Center, 22nd Floor
Southfield, MI 48075
(248) 358-4400

*Attorneys for Patent Owner
Ancora Technologies Inc.*

Certificate of Compliance Pursuant to 37 C.F.R. §42.24

This paper complies with the type-volume limitation of 37 C.F.R. §42.24. The paper contains 13,664 words, excluding the parts of the paper exempted by §42.24(a).

This paper also complies with the typeface requirements of 37 C.F.R. §42.6(a)(ii) and the type style requirements of §42.6(a)(iii)&(iv).

Dated: November 18, 2021

Respectfully submitted,

/John P. Rondini/
Marc Lorelli (Reg. No. 43,759)
John P. Rondini (Reg. No. 64,949)
BROOKS KUSHMAN P.C.
1000 Town Center, 22nd Floor
Southfield, MI 48075
(248) 358-4400

*Attorneys for Patent Owner
Ancora Technologies Inc.*