Nintendo Co., Ltd. et al. v. Ancora Techs., Inc.

IPR2021-01338

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

Claims 1–3, 6–14, and 16

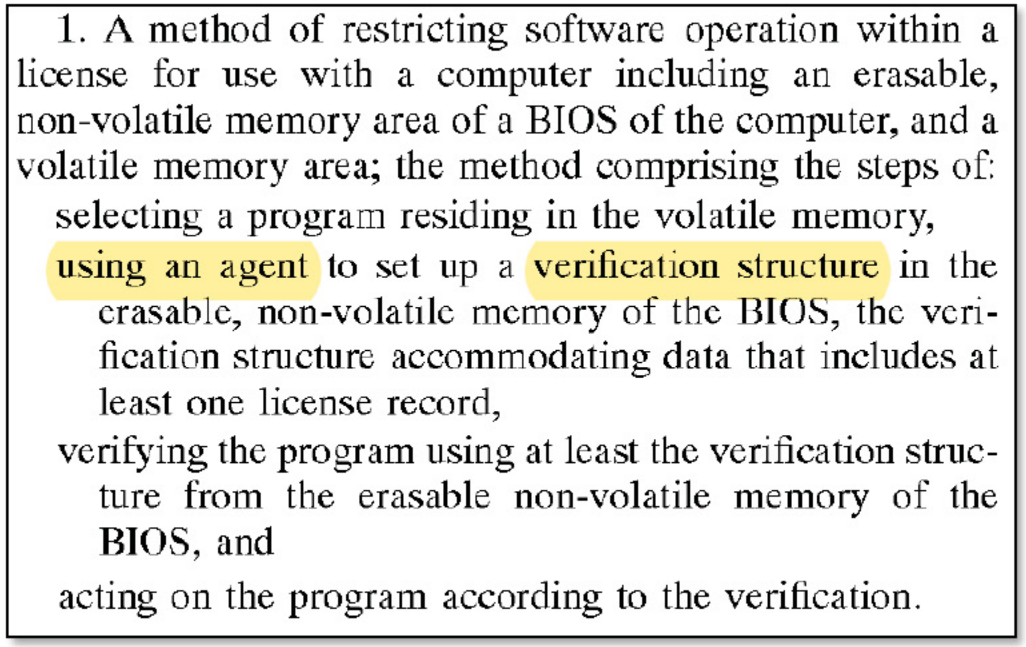
Petitioner Nintendo's Demonstratives

Oral Hearing
October 3, 2022

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Grounds of Invalidity

Ground	Basis	Reference(s)	Challenged Claims
1	§ 103	Hellman + Chou	1–2, 11, 13
2	§ 103	Hellman + Chou + Schneck	1–3, 6–14, 16



1. A method of restricting software operation within a license for use with a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area; the method comprising the steps of:

selecting a program residing in the volatile memory,

using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS, the verification structure accommodating data that includes at least one license record,

verifying the program using at least the verification structure from the erasable non-volatile memory of the BIOS, and

acting on the program according to the verification.

- Pet. 7.

Overview

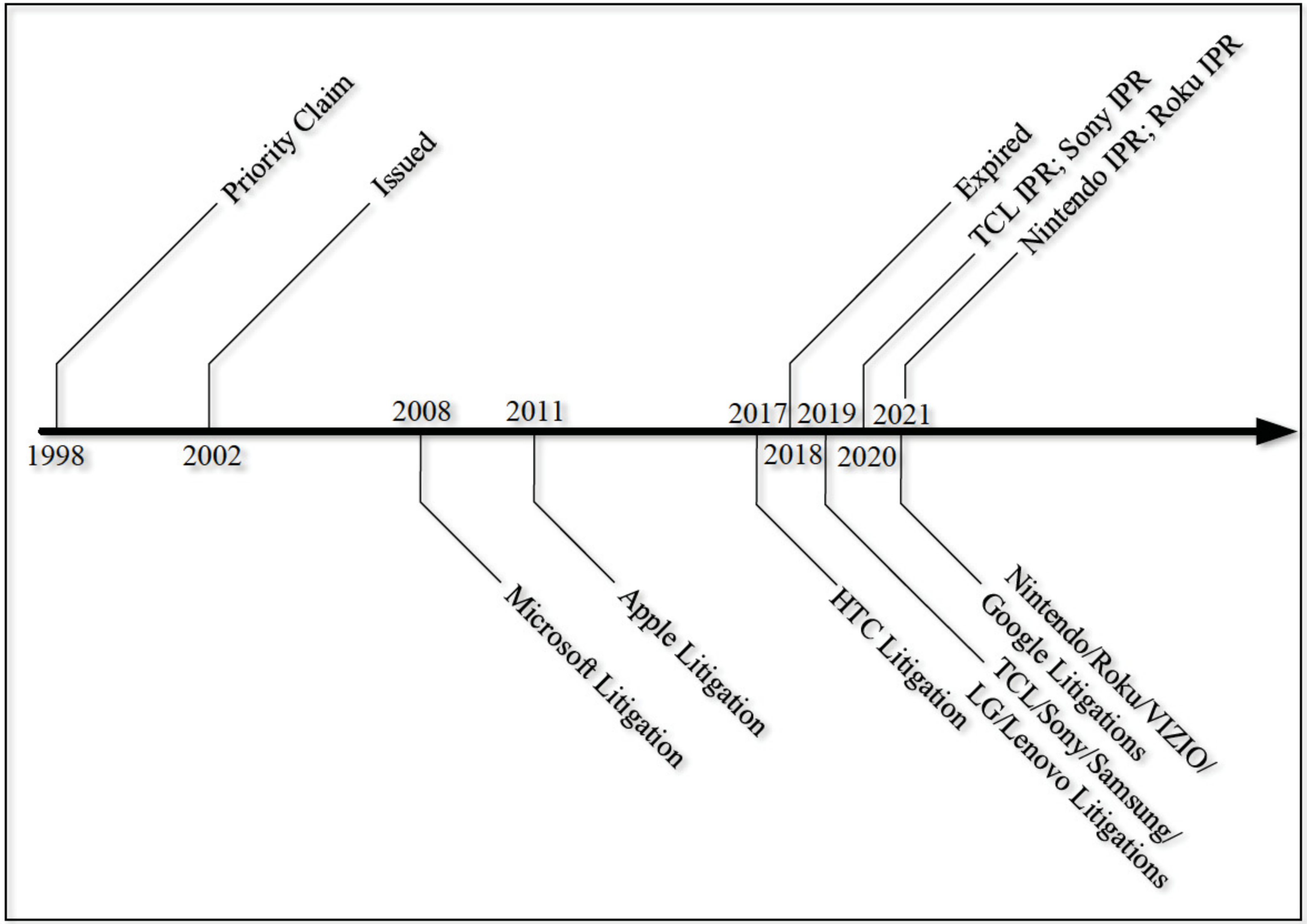
1. Background
2. The “using an agent” Limitation.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

Overview

1. Background
2. The "using an agent" Limitation.
3. The "verification structure" Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

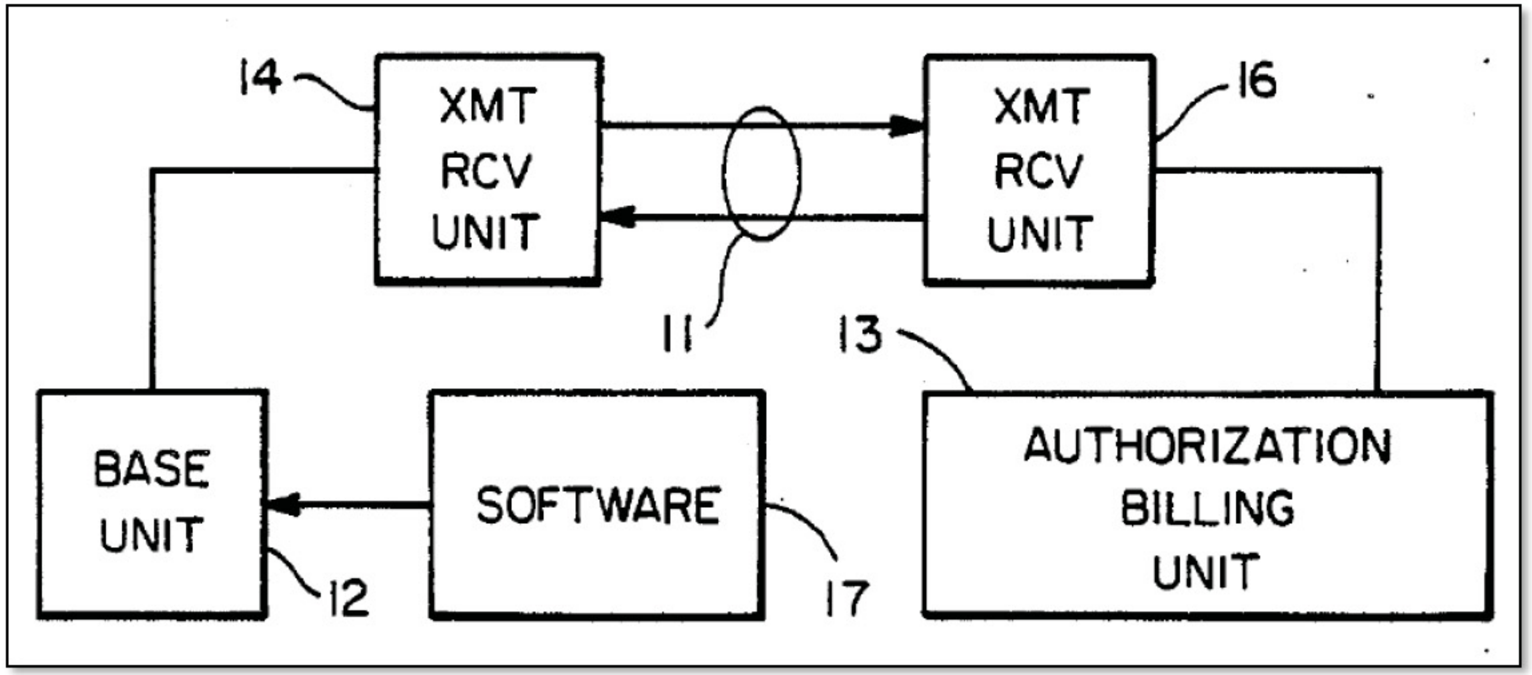
Background: ’941 Patent



- Pet. 3-4, 18-21.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

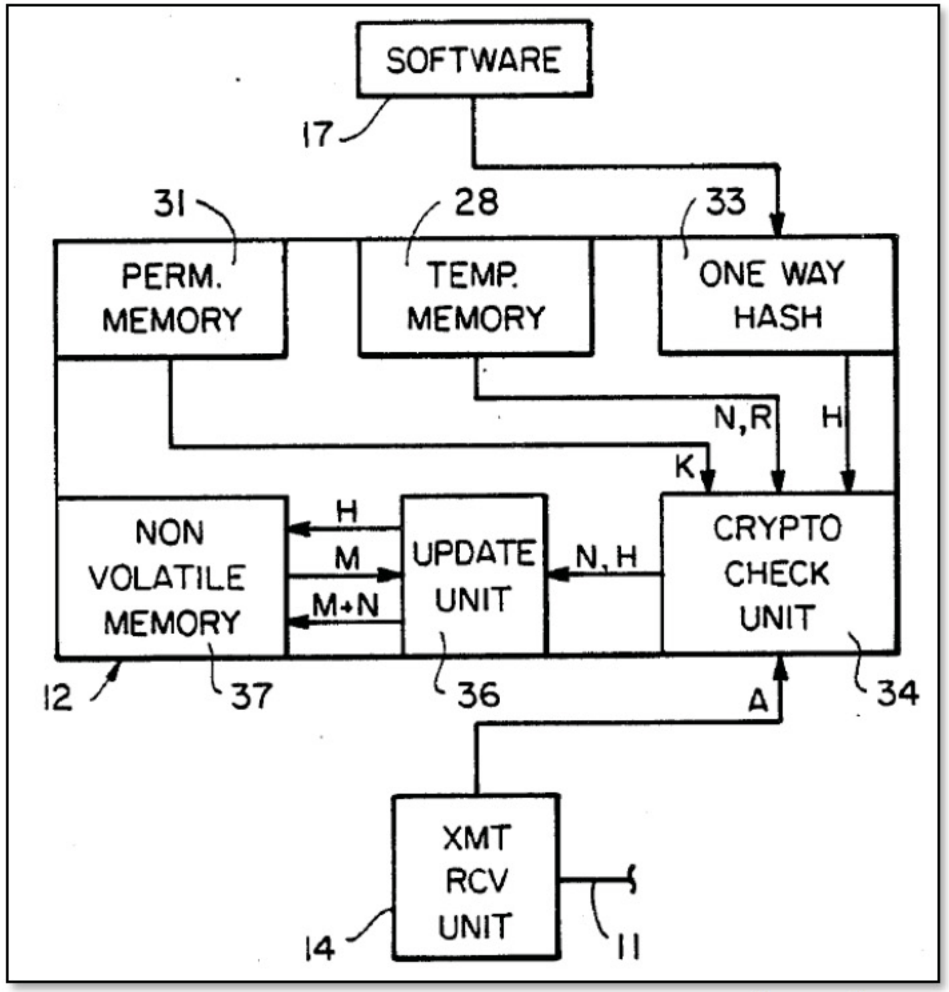
Background: Hellman



Hellman, Fig. 1.

- Pet. 24-29; Ex.1003 ¶59-77.

Background: Hellman

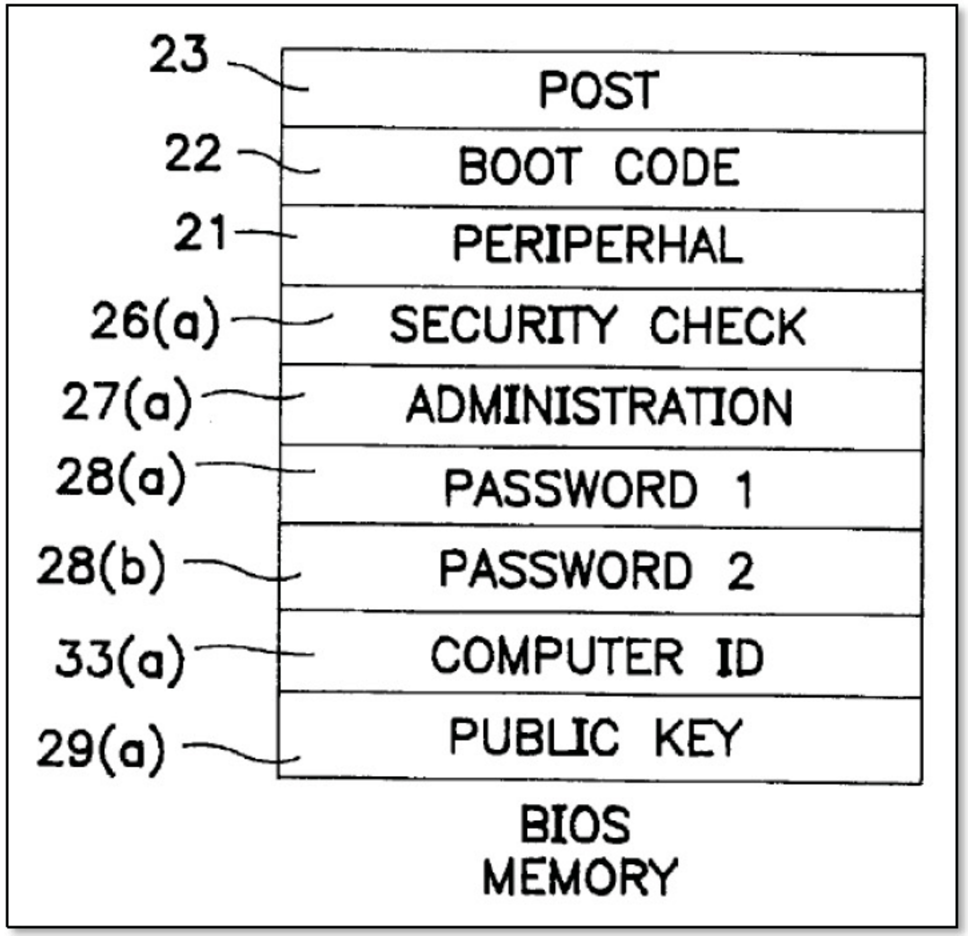


Hellman, Fig. 6.

- Pet. 24-29; Ex.1003 ¶59-77.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Background: Chou



Chou, Fig. 7.

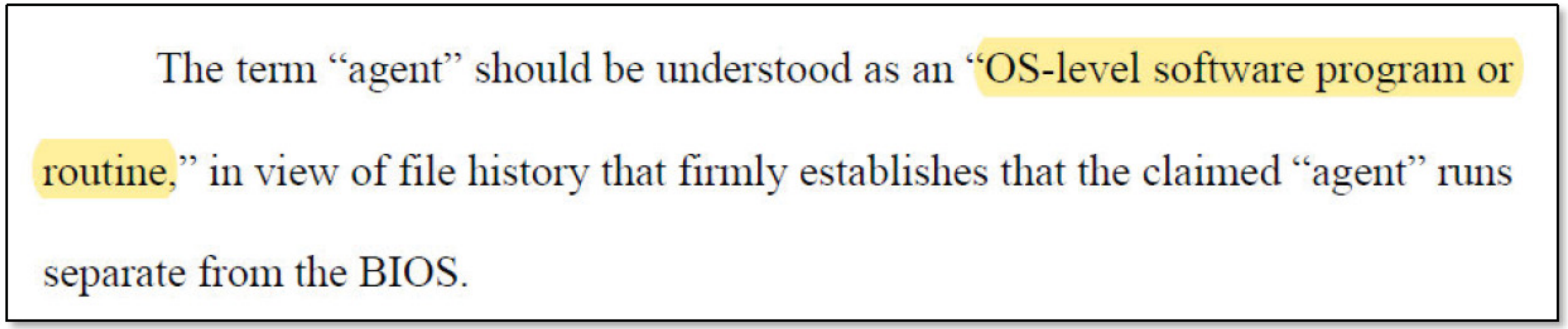
- Pet. 29-30; Ex.1003 ¶78-83.

Overview

1. Background
2. The “using an agent” Limitation.
    1. There is no disclaimer of agent to “OS-level.”
    2. There is no disclaimer or other basis for the “software-only” limitation.
    3. Hellman discloses or renders obvious an “OS-level” and “software-only” agent.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Agent: Ancora's Claim Construction Requires Disclaimer



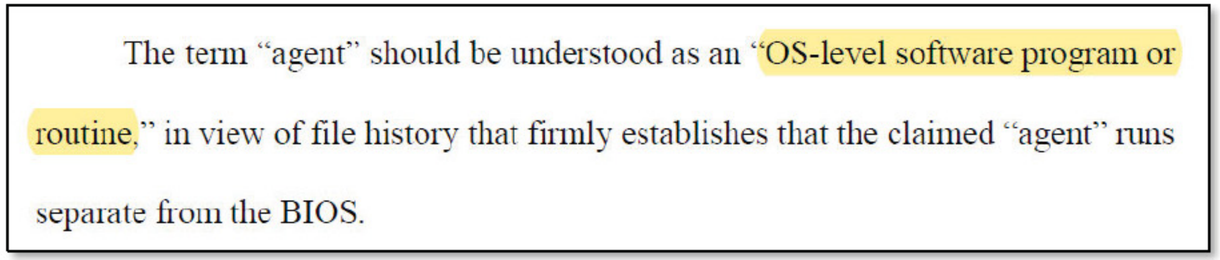
The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

- Reply 3-7.

Agent: Ancora's Claim Construction – Contradicted by Itself

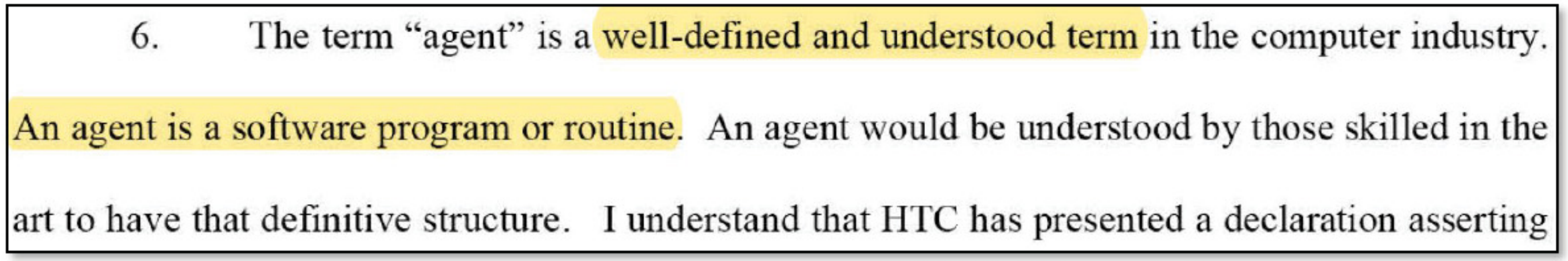
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



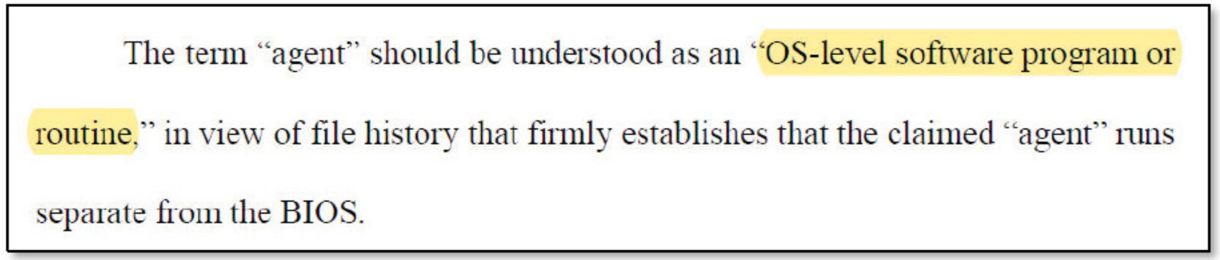
6. The term "agent" is a well-defined and understood term in the computer industry. An agent is a software program or routine. An agent would be understood by those skilled in the art to have that definitive structure. I understand that HTC has presented a declaration asserting

EX. 2004, ¶6 (sworn testimony of Ancora's expert).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

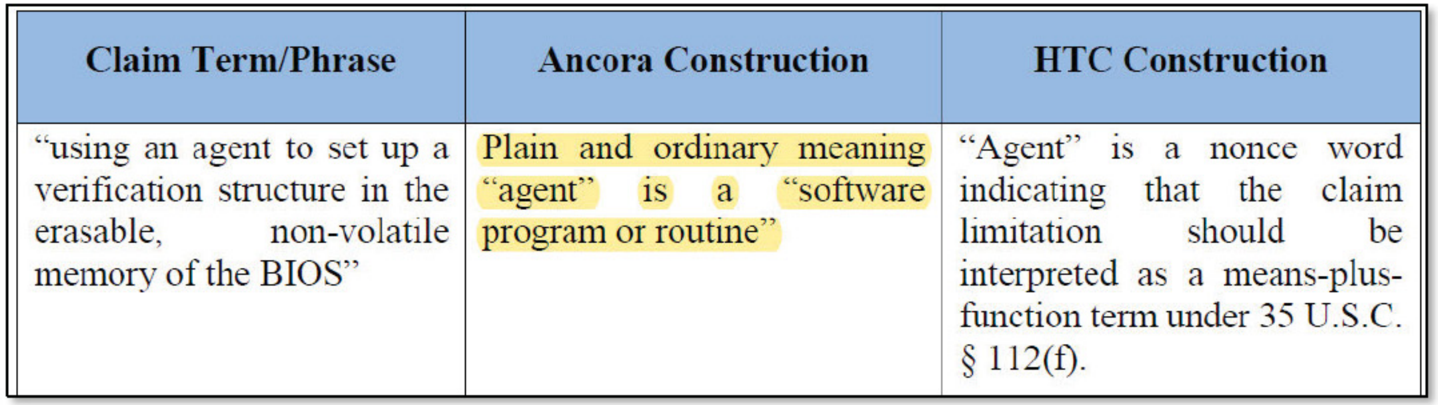
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



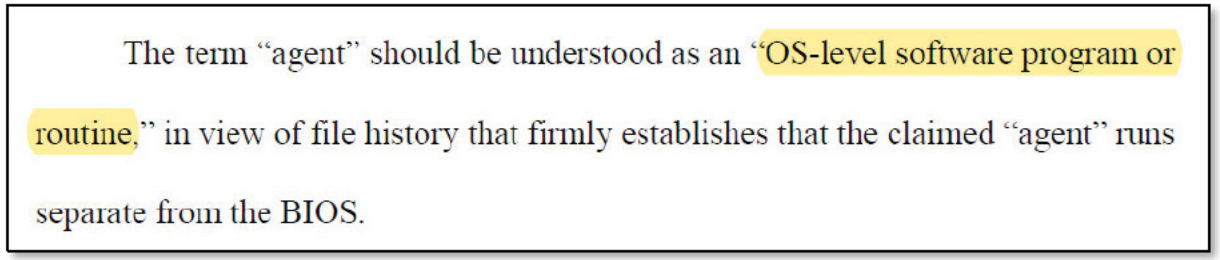
Claim Term/Phrase	Ancora Construction	HTC Construction
"using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS"	Plain and ordinary meaning "agent" is a "software program or routine"	"Agent" is a nonce word indicating that the claim limitation should be interpreted as a means-plus-function term under 35 U.S.C. § 112(f).

EX. 1073, p. 12 (Ancora claim construction brief).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

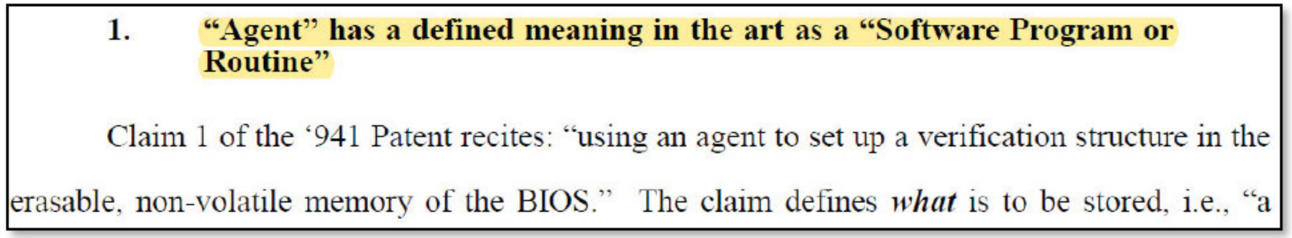
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



1. **"Agent" has a defined meaning in the art as a "Software Program or Routine"**

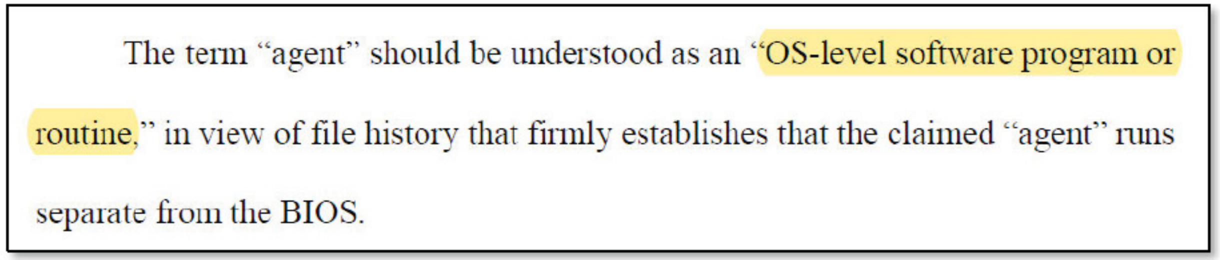
Claim 1 of the '941 Patent recites: "using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS." The claim defines ***what*** is to be stored, i.e., "a

EX. 1073, p. 12 (Ancora claim construction brief).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

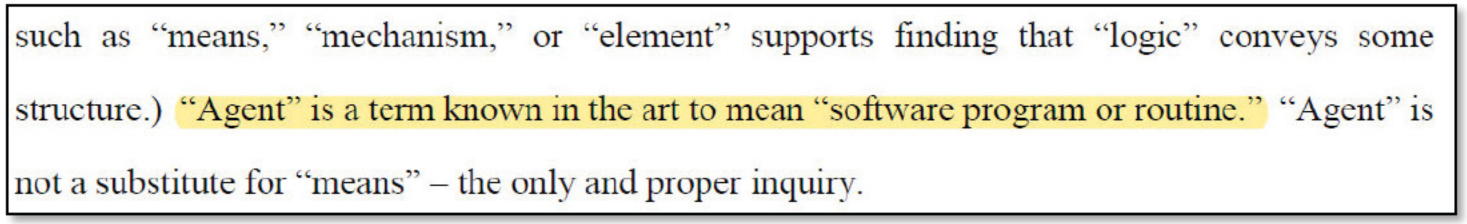
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



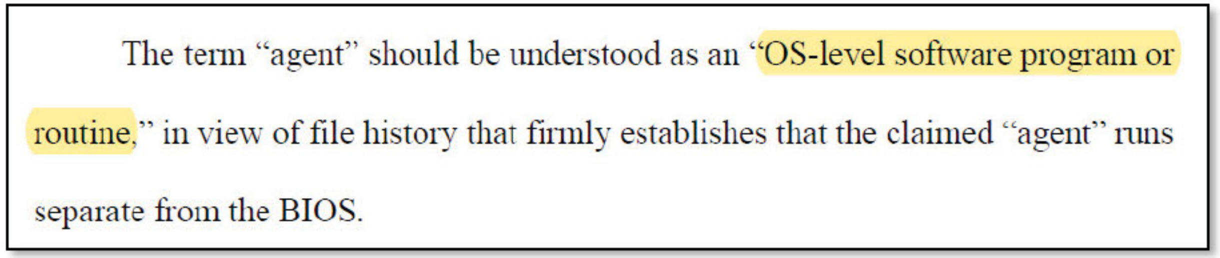
such as "means," "mechanism," or "element" supports finding that "logic" conveys some structure.) "Agent" is a term known in the art to mean "software program or routine." "Agent" is not a substitute for "means" – the only and proper inquiry.

EX. 1074, p. 7 (Ancora claim construction brief).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

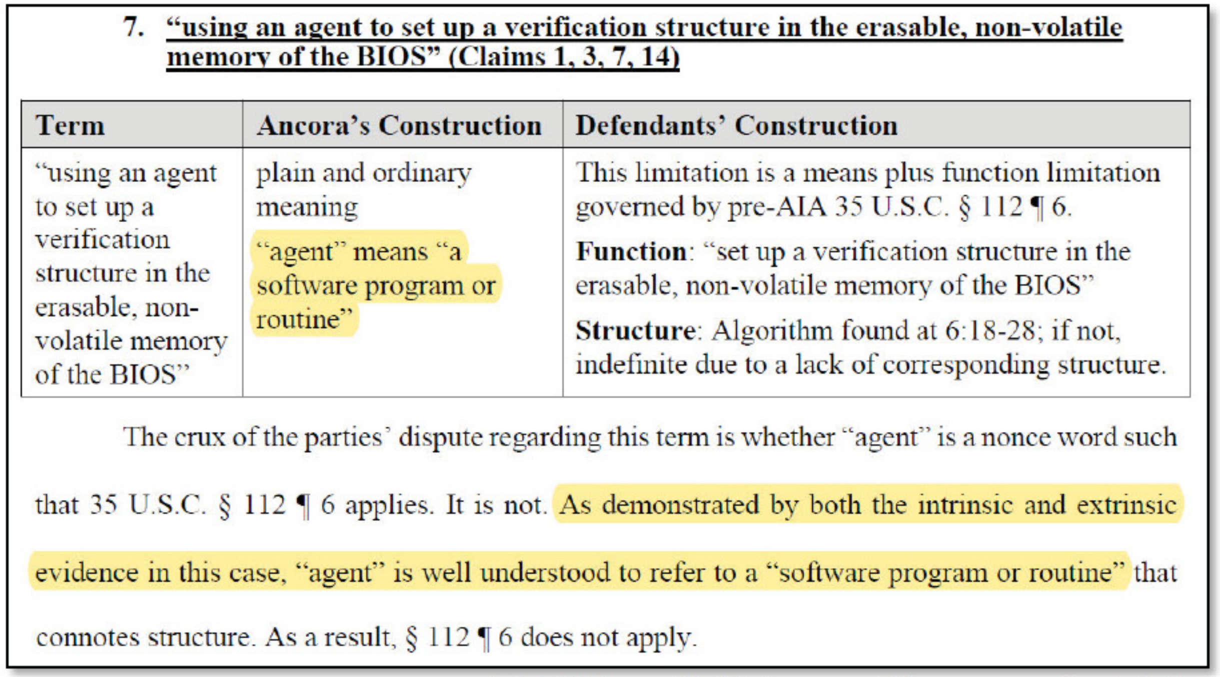
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



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

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

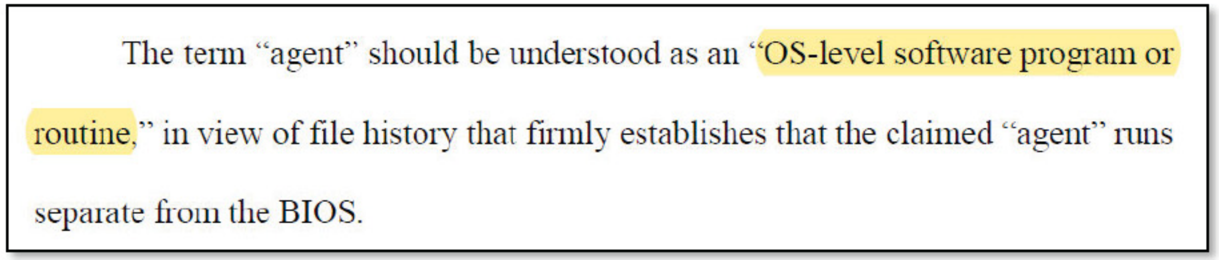
The crux of the parties' dispute regarding this term is whether "agent" is a nonce word such that 35 U.S.C. § 112 ¶ 6 applies. It is not. As demonstrated by both the intrinsic and extrinsic evidence in this case, "agent" is well understood to refer to a "software program or routine" that connotes structure.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

EX. 1070, p. 17-18 (Ancora claim construction brief).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

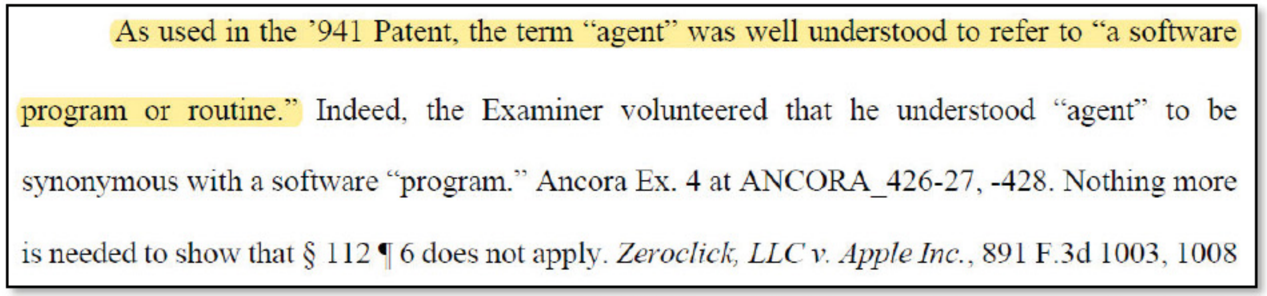
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



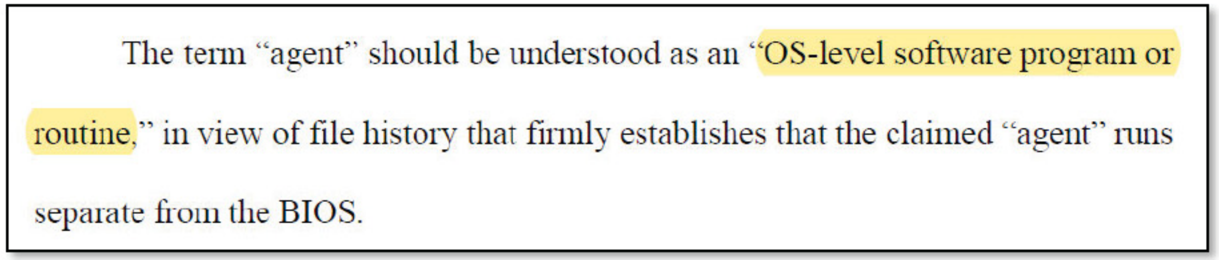
As used in the '941 Patent, the term "agent" was well understood to refer to "a software program or routine." Indeed, the Examiner volunteered that he understood "agent" to be synonymous with a software "program." Ancora Ex. 4 at ANCORA_426-27, -428. Nothing more is needed to show that § 112 ¶ 6 does not apply. *Zeroclick, LLC v. Apple Inc.*, 891 F.3d 1003, 1008

EX. 1071, p. 2-3 (Ancora claim construction brief).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

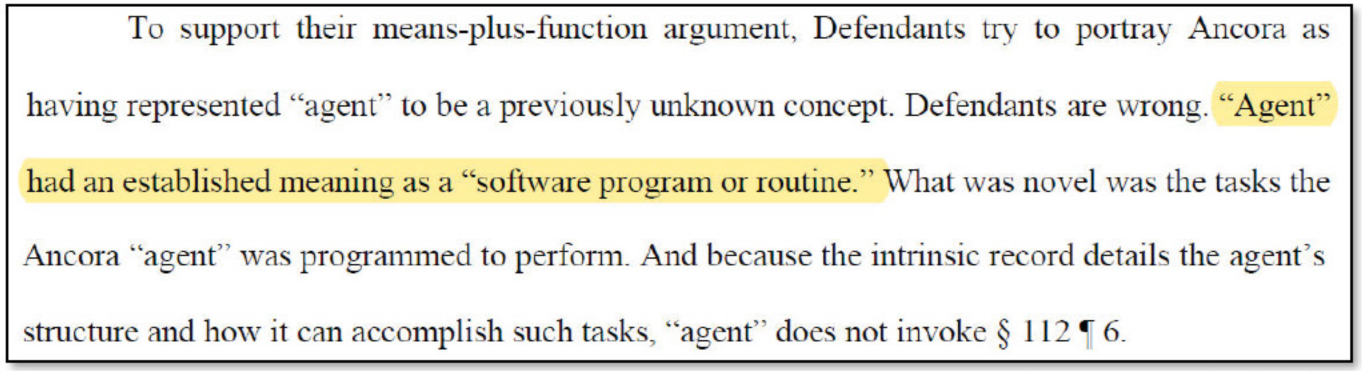
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



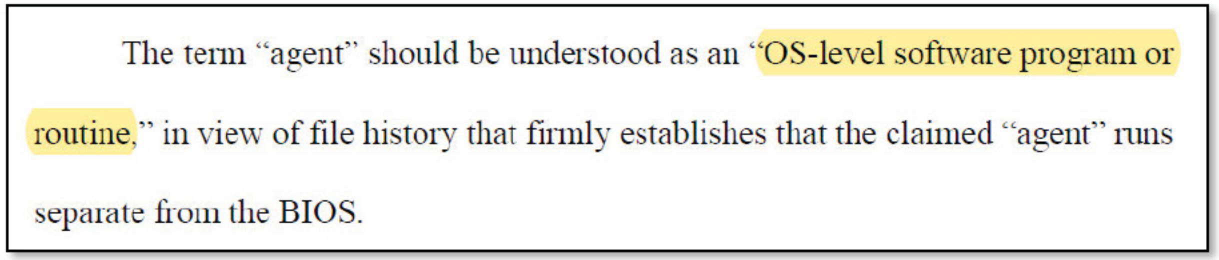
To support their means-plus-function argument, Defendants try to portray Ancora as having represented "agent" to be a previously unknown concept. Defendants are wrong. "Agent" had an established meaning as a "software program or routine." What was novel was the tasks the Ancora "agent" was programmed to perform. And because the intrinsic record details the agent's structure and how it can accomplish such tasks, "agent" does not invoke § 112 ¶ 6.

EX. 1072, p. 8-9 (Ancora claim construction brief).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

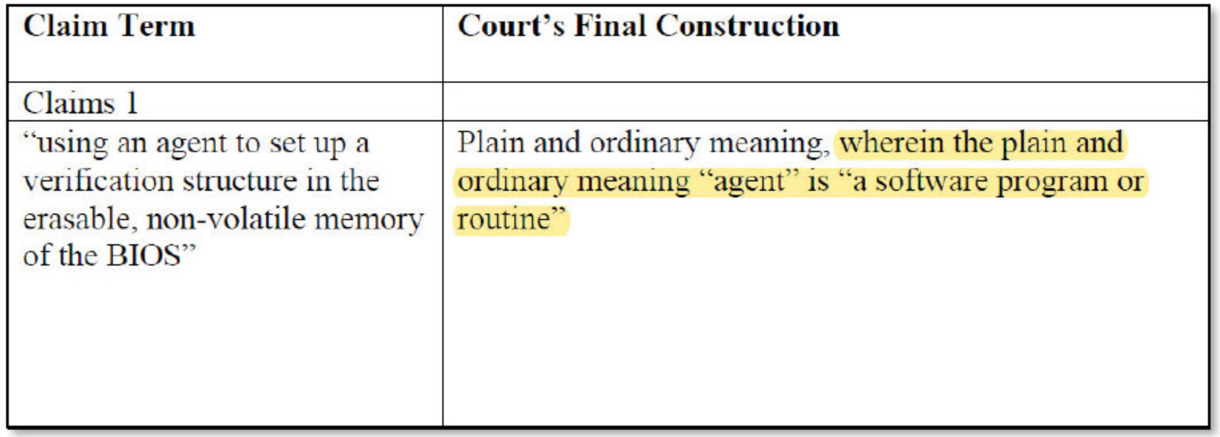
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



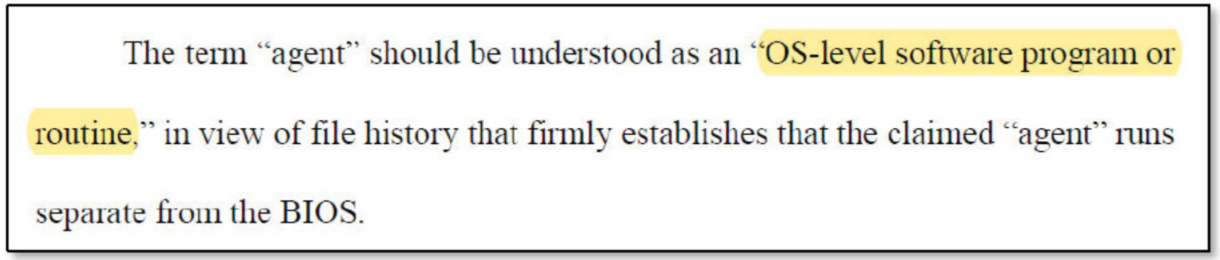
Claim Term	**Court's Final Construction**
Claims 1	
"using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS"	Plain and ordinary meaning, wherein the plain and ordinary meaning "agent" is "a software program or routine"

EX. 1012, p. 3 (W.D. Tex. claim construction order, adopting Ancora construction); *see also* EX. 1013, p. 28-36 (same).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

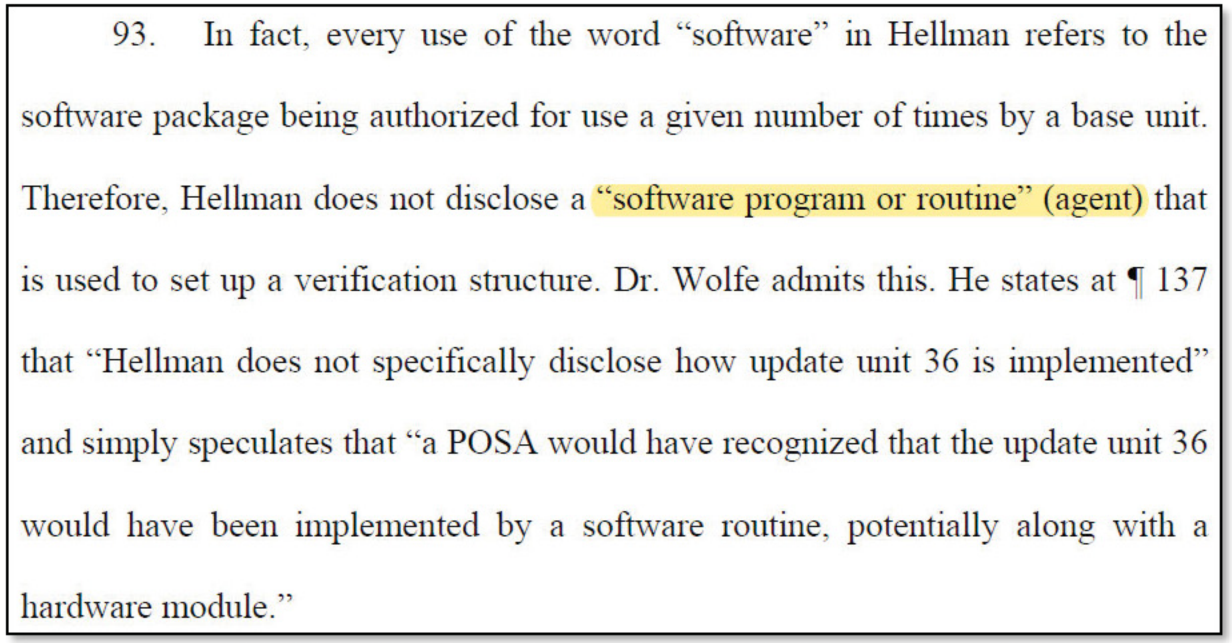
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



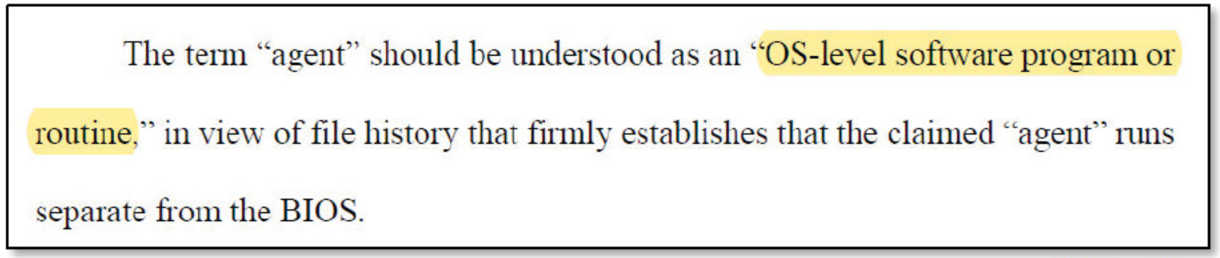
93. In fact, every use of the word "software" in Hellman refers to the software package being authorized for use a given number of times by a base unit. Therefore, Hellman does not disclose a "software program or routine" (agent) that is used to set up a verification structure. Dr. Wolfe admits this. He states at ¶ 137 that "Hellman does not specifically disclose how update unit 36 is implemented" and simply speculates that "a POSA would have recognized that the update unit 36 would have been implemented by a software routine, potentially along with a hardware module."

EX. 2013, ¶93 (Ancora's current expert, Dr. Martin, in prior IPR testimony).

- Reply 3-4.

Agent: Ancora's Claim Construction – Contradicted by Itself

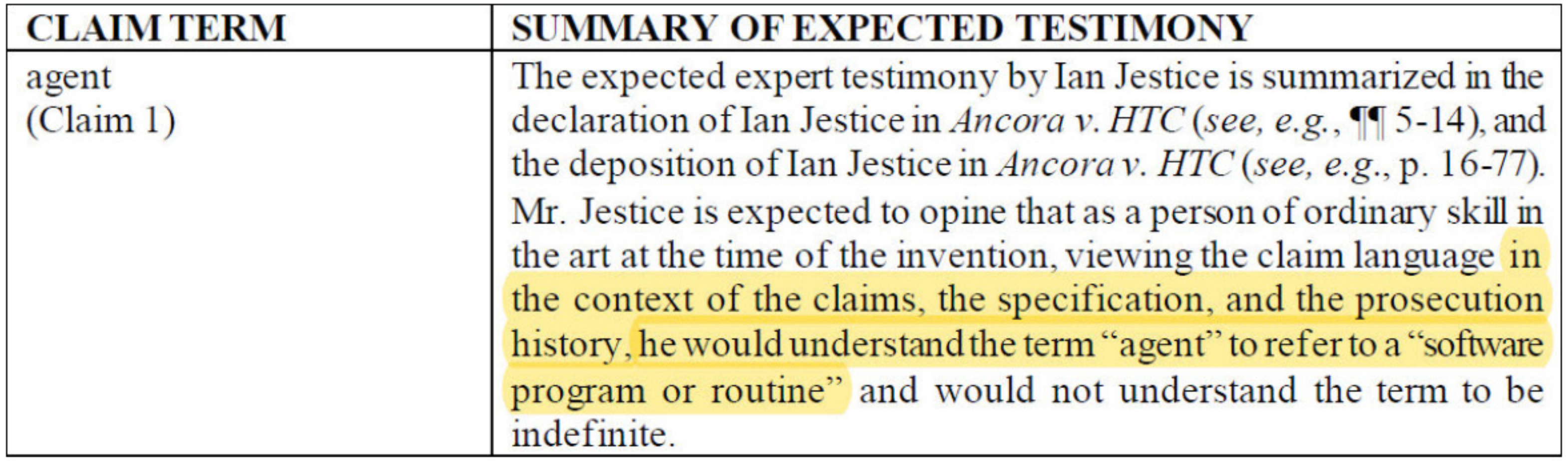
What Ancora Says Now:



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.

What Ancora Said Before:



CLAIM TERM	SUMMARY OF EXPECTED TESTIMONY
agent (Claim 1)	The expected expert testimony by Ian Jestice is summarized in the declaration of Ian Jestice in *Ancora v. HTC* (*see, e.g.*, ¶¶ 5-14), and the deposition of Ian Jestice in *Ancora v. HTC* (*see, e.g.*, p. 16-77). Mr. Jestice is expected to opine that as a person of ordinary skill in the art at the time of the invention, viewing the claim language in the context of the claims, the specification, and the prosecution history, he would understand the term "agent" to refer to a "software program or routine" and would not understand the term to be indefinite.

EX. 1075, p. 3 (Ancora's claim construction disclosure, January 2022).

- Reply 3-4.

Overview

1. Background
2. The “using an agent” Limitation.
   1. There is no disclaimer of agent to “OS-level.”
   2. There is no disclaimer or other basis for the “software-only” limitation.
   3. Hellman discloses or renders obvious an “OS-level,” “software-only” agent.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

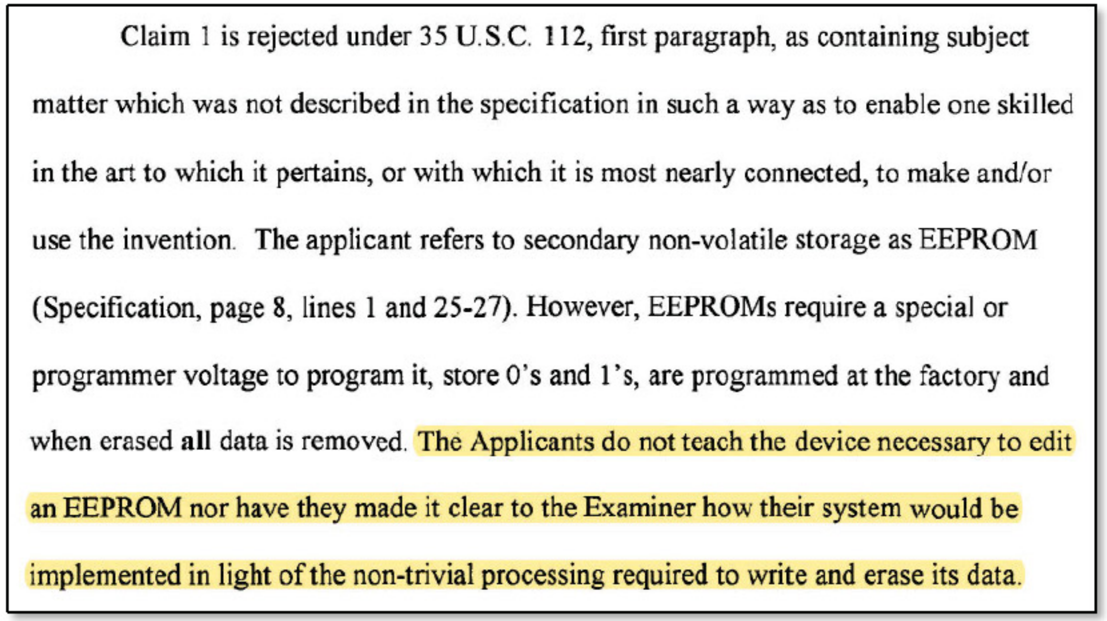
DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Agent: (Lack of) Disclosure in the Intrinsic Record

"Agent": Not mentioned in specification, or the original claims.

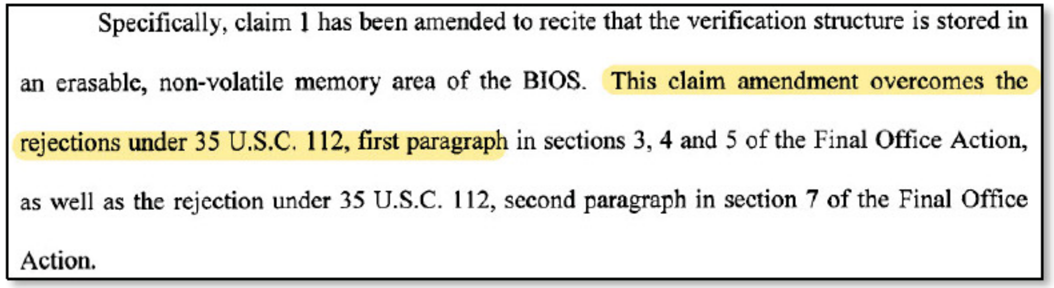
- Reply 4; Ex. 1033 ¶4-15.

Agent: (Lack of) Disclosure in the Intrinsic Record



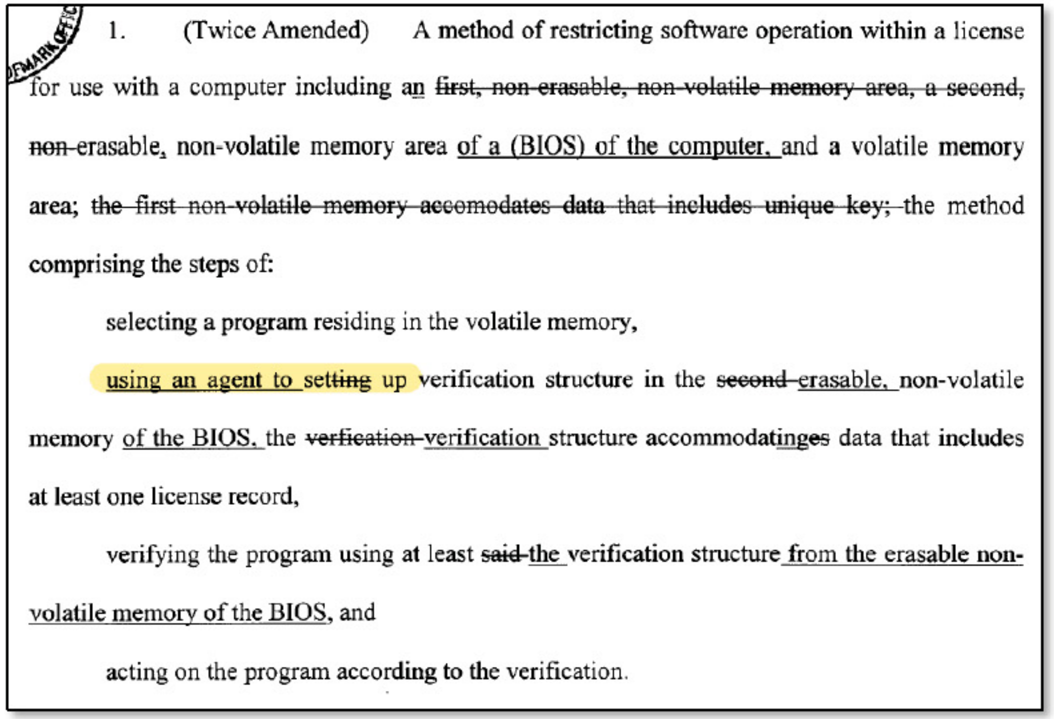
Claim 1 is rejected under 35 U.S.C. 112, first paragraph, as containing subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention. The applicant refers to secondary non-volatile storage as EEPROM (Specification, page 8, lines 1 and 25-27). However, EEPROMs require a special or programmer voltage to program it, store 0's and 1's, are programmed at the factory and when erased **all** data is removed. The 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.

Office Action (June 22, 2001).



Specifically, claim 1 has been amended to recite that the verification structure is stored in an erasable, non-volatile memory area of the BIOS. This claim amendment overcomes the rejections under 35 U.S.C. 112, first paragraph in sections 3, 4 and 5 of the Final Office Action, as well as the rejection under 35 U.S.C. 112, second paragraph in section 7 of the Final Office Action.

Office Action Response (Nov. 14, 2001).



1. (Twice Amended) A method of restricting software operation within a license for use with a computer including an ~~first, non-erasable, non-volatile memory area, a second, non-~~erasable, non-volatile memory area of a (BIOS) of the computer, and a volatile memory area; ~~the first non-volatile memory accomodates data that includes unique key;~~ the method comprising the steps of:

selecting a program residing in the volatile memory,

using an agent to set~~ting~~ up verification structure in the ~~second~~ erasable, non-volatile memory of the BIOS, the ~~verfication~~ verification structure accommodating~~es~~ data that includes at least one license record,

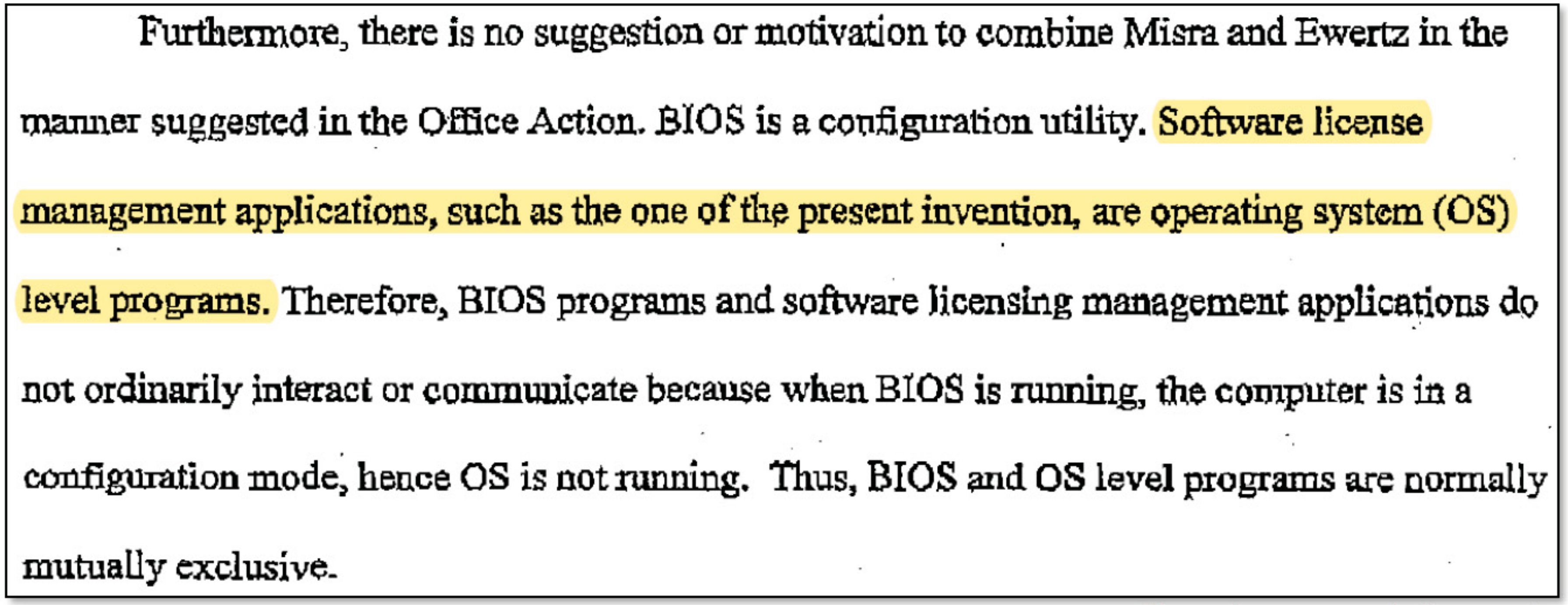
verifying the program using at least ~~said~~ the verification structure from the erasable non-volatile memory of the BIOS, and

acting on the program according to the verification.

Office Action Response (Nov. 14, 2001).

- Reply 4-7; Ex. 1033 ¶4-15.

Agent: Supposed Disclaimer of Agent without Mentioning Agent

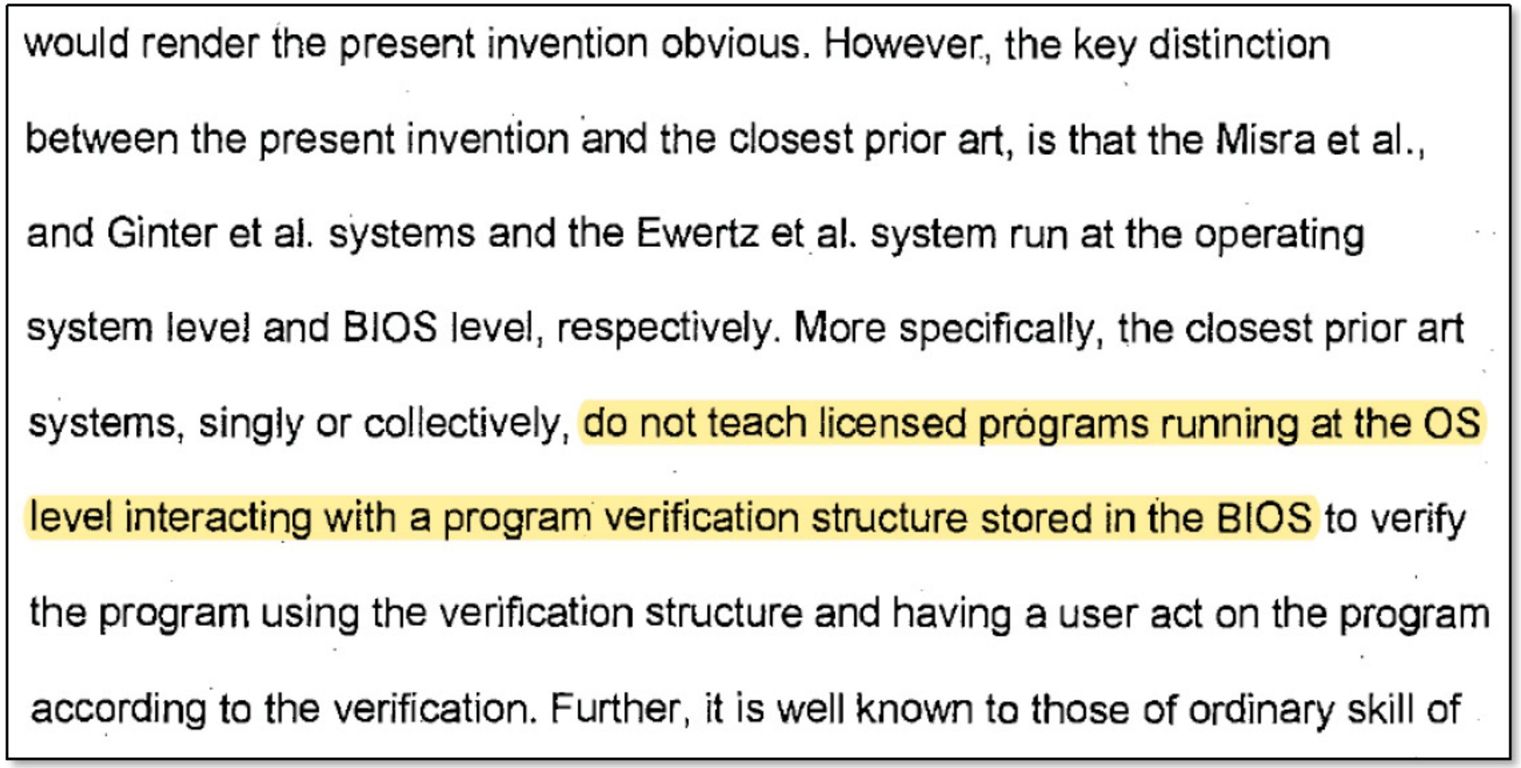


Furthermore, there is no suggestion or motivation to combine Misra and Ewertz in the manner suggested in the Office Action. BIOS is a configuration utility. Software license management applications, such as the one of the present invention, are operating system (OS) level programs. Therefore, BIOS programs and software licensing management applications do not ordinarily interact or communicate because when BIOS is running, the computer is in a configuration mode, hence OS is not running. Thus, BIOS and OS level programs are normally mutually exclusive.

Office Action Response (Feb. 5, 2002)

- Reply 4-7; Ex. 1033 ¶4-15.

Agent: Supposed Disclaimer of Agent without Mentioning Agent

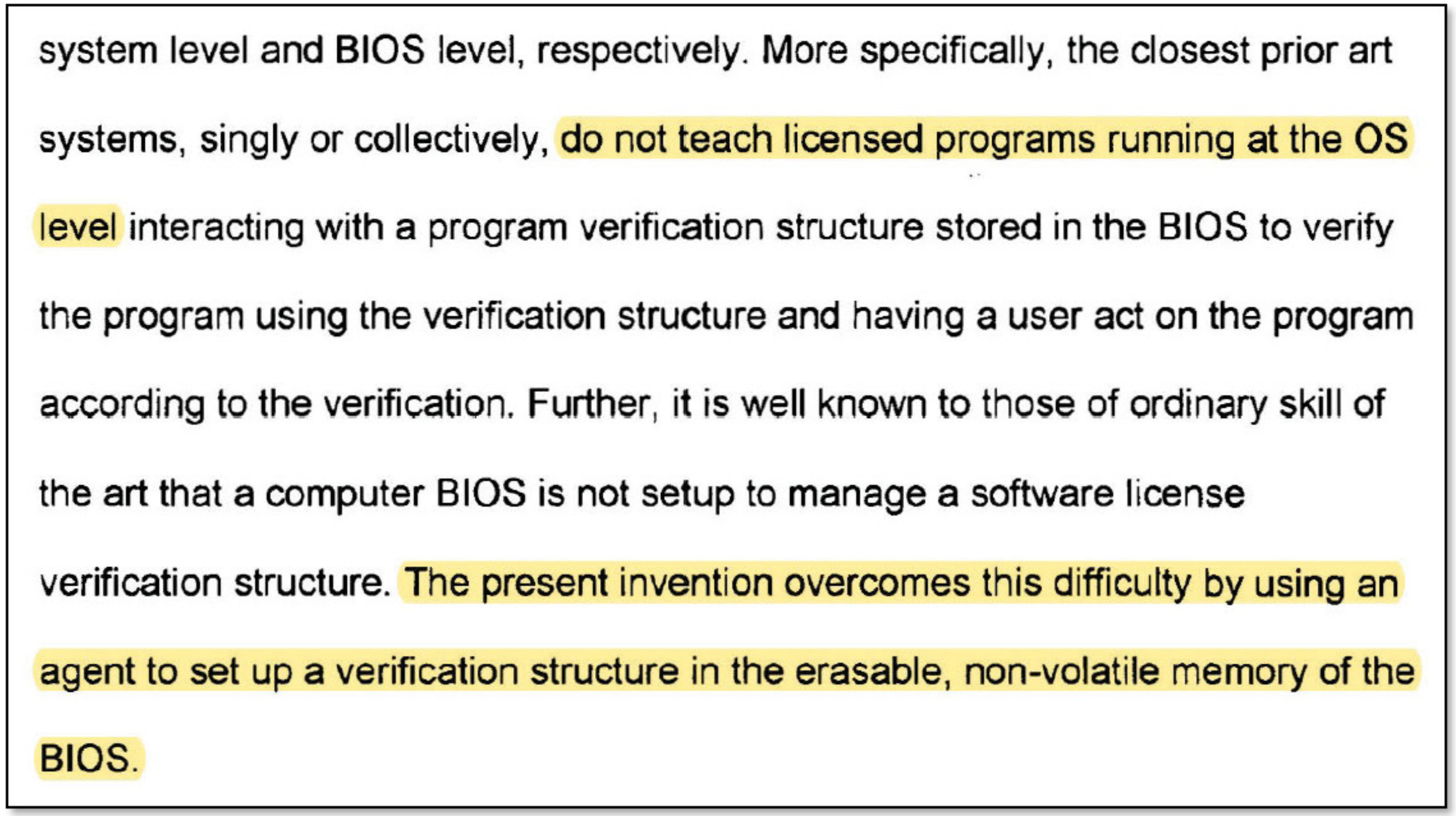


would render the present invention obvious. However, the key distinction between the present invention and the closest prior art, is that the Misra et al., and Ginter et al. systems and the Ewertz et al. system run at the operating system level and BIOS level, respectively. More specifically, the closest 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

Notice of Allowance (Mar. 28, 2002)

- Reply 4-7; Ex. 1033 ¶4-15.

Agent: Supposed Disclaimer of Agent without Mentioning Agent

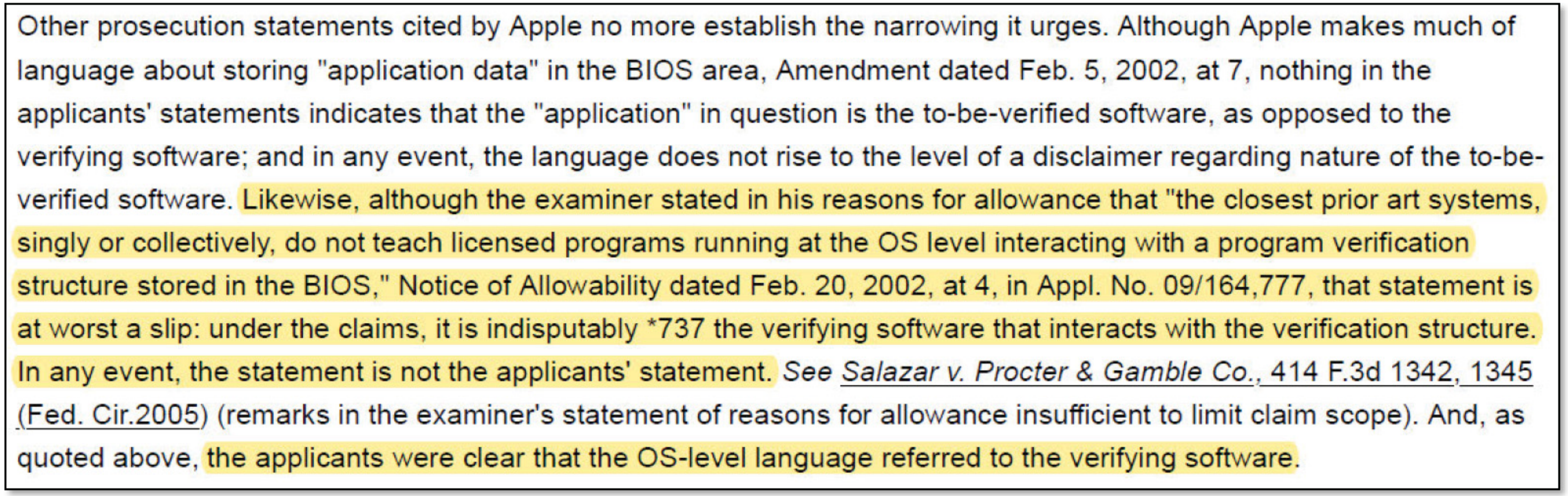


system level and BIOS level, respectively. More specifically, the closest prior art systems, singly or collectively, do not teach licensed programs running at the OS level interacting with a program verification structure stored in the BIOS to verify the program using the verification structure and having 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.

Notice of Allowance (Mar. 28, 2002)

- Reply 4-7; Ex. 1033 ¶4-15.

Agent: Supposed Disclaimer of Agent without Mentioning Agent

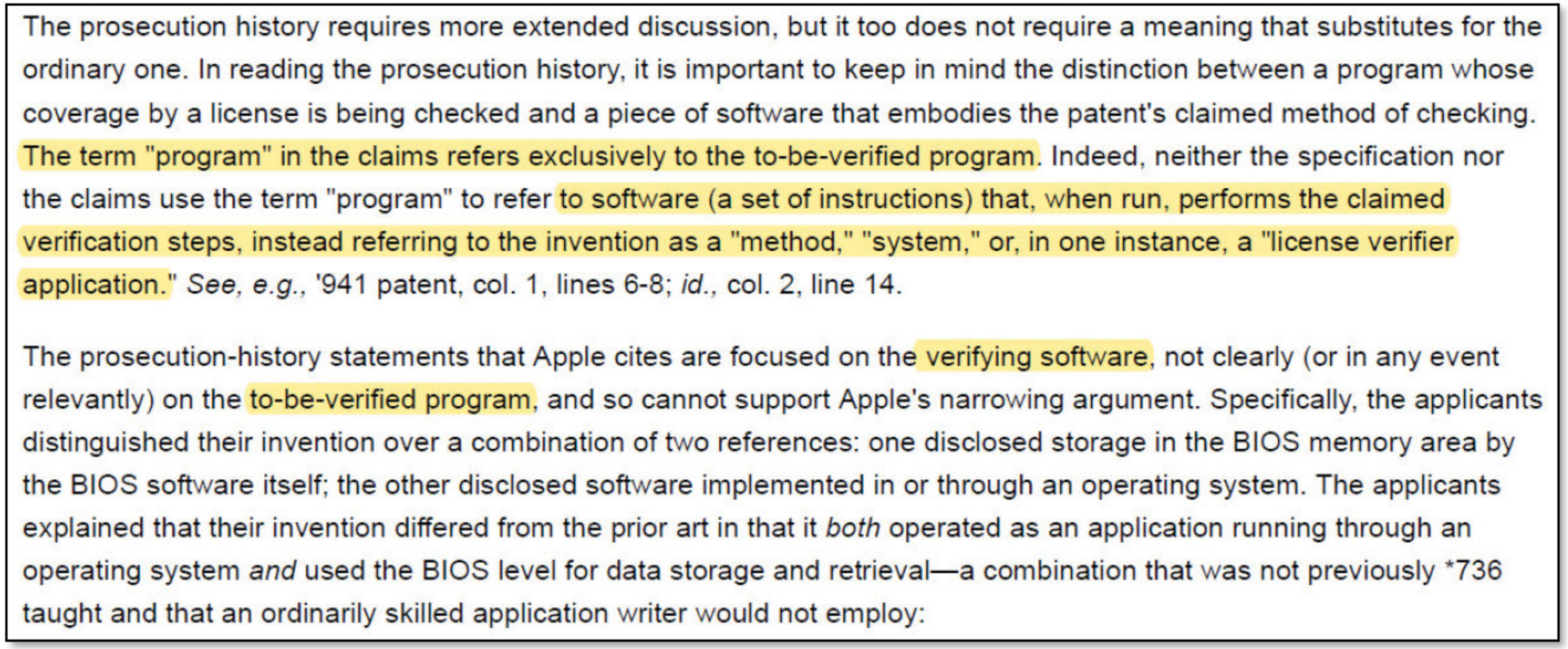


Other prosecution statements cited by Apple no more establish the narrowing it urges. Although Apple makes much of language about storing "application data" in the BIOS area, Amendment dated Feb. 5, 2002, at 7, nothing in the applicants' statements indicates that the "application" in question is the to-be-verified software, as opposed to the verifying software; and in any event, the language does not rise to the level of a disclaimer regarding nature of the to-be-verified software. Likewise, although the examiner stated in his reasons for allowance that "the closest prior art systems, singly or collectively, do not teach licensed programs running at the OS level interacting with a program verification structure stored in the BIOS," Notice of Allowability dated Feb. 20, 2002, at 4, in Appl. No. 09/164,777, that statement is at worst a slip: under the claims, it is indisputably *737 the verifying software that interacts with the verification structure. In any event, the statement is not the applicants' statement. *See Salazar v. Procter & Gamble Co.*, 414 F.3d 1342, 1345 (Fed. Cir.2005) (remarks in the examiner's statement of reasons for allowance insufficient to limit claim scope). And, as quoted above, the applicants were clear that the OS-level language referred to the verifying software.

Ancora Techs., Inc. v. Apple, Inc., 744 F.3d 732, 736-37 (Fed. Cir. 2014).

- Reply 4-7; Ex. 1033 ¶4-15.

Agent: Supposed Disclaimer of Agent without Mentioning Agent



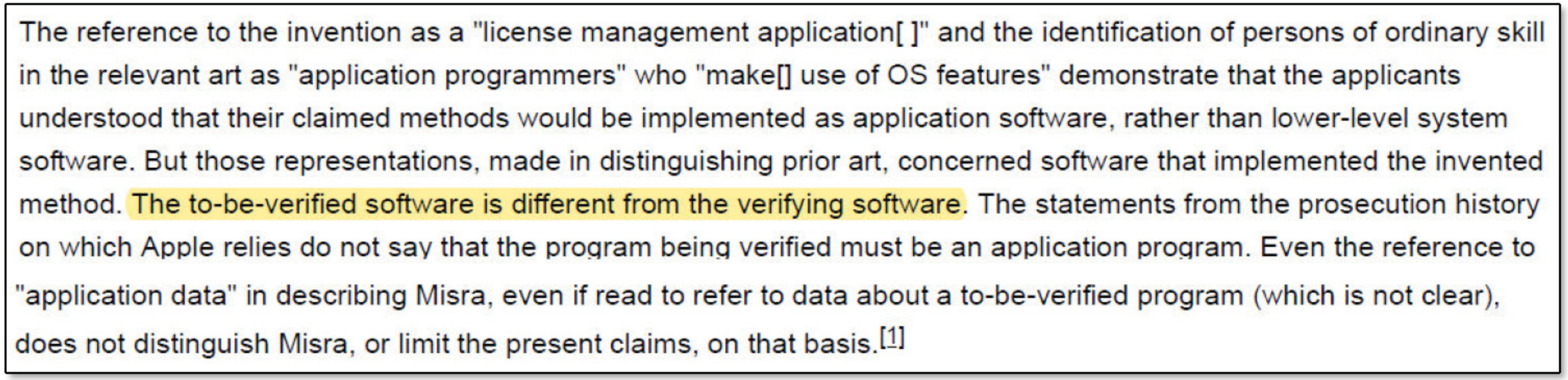
The prosecution history requires more extended discussion, but it too does not require a meaning that substitutes for the ordinary one. In reading the prosecution history, it is important to keep in mind the distinction between a program whose coverage by a license is being checked and a piece of software that embodies the patent's claimed method of checking. The term "program" in the claims refers exclusively to the to-be-verified program. Indeed, neither the specification nor the claims use the term "program" to refer to software (a set of instructions) that, when run, performs the claimed verification steps, instead referring to the invention as a "method," "system," or, in one instance, a "license verifier application." *See, e.g.*, '941 patent, col. 1, lines 6-8; *id.*, col. 2, line 14.

The prosecution-history statements that Apple cites are focused on the verifying software, not clearly (or in any event relevantly) on the to-be-verified program, and so cannot support Apple's narrowing argument. Specifically, the applicants distinguished their invention over a combination of two references: one disclosed storage in the BIOS memory area by the BIOS software itself; the other disclosed software implemented in or through an operating system. The applicants explained that their invention differed from the prior art in that it *both* operated as an application running through an operating system *and* used the BIOS level for data storage and retrieval—a combination that was not previously *736 taught and that an ordinarily skilled application writer would not employ:

Ancora Techs., Inc. v. Apple, Inc., 744 F.3d 732, 735-36 (Fed. Cir. 2014).

- Reply 4-7; Ex. 1033 ¶4-15.

Agent: Supposed Disclaimer of Agent without Mentioning Agent



The reference to the invention as a "license management application[ ]" and the identification of persons of ordinary skill in the relevant art as "application programmers" who "make[] use of OS features" demonstrate that the applicants understood that their claimed methods would be implemented as application software, rather than lower-level system software. But those representations, made in distinguishing prior art, concerned software that implemented the invented method. The to-be-verified software is different from the verifying software. The statements from the prosecution history on which Apple relies do not say that the program being verified must be an application program. Even the reference to "application data" in describing Misra, even if read to refer to data about a to-be-verified program (which is not clear), does not distinguish Misra, or limit the present claims, on that basis.[1]

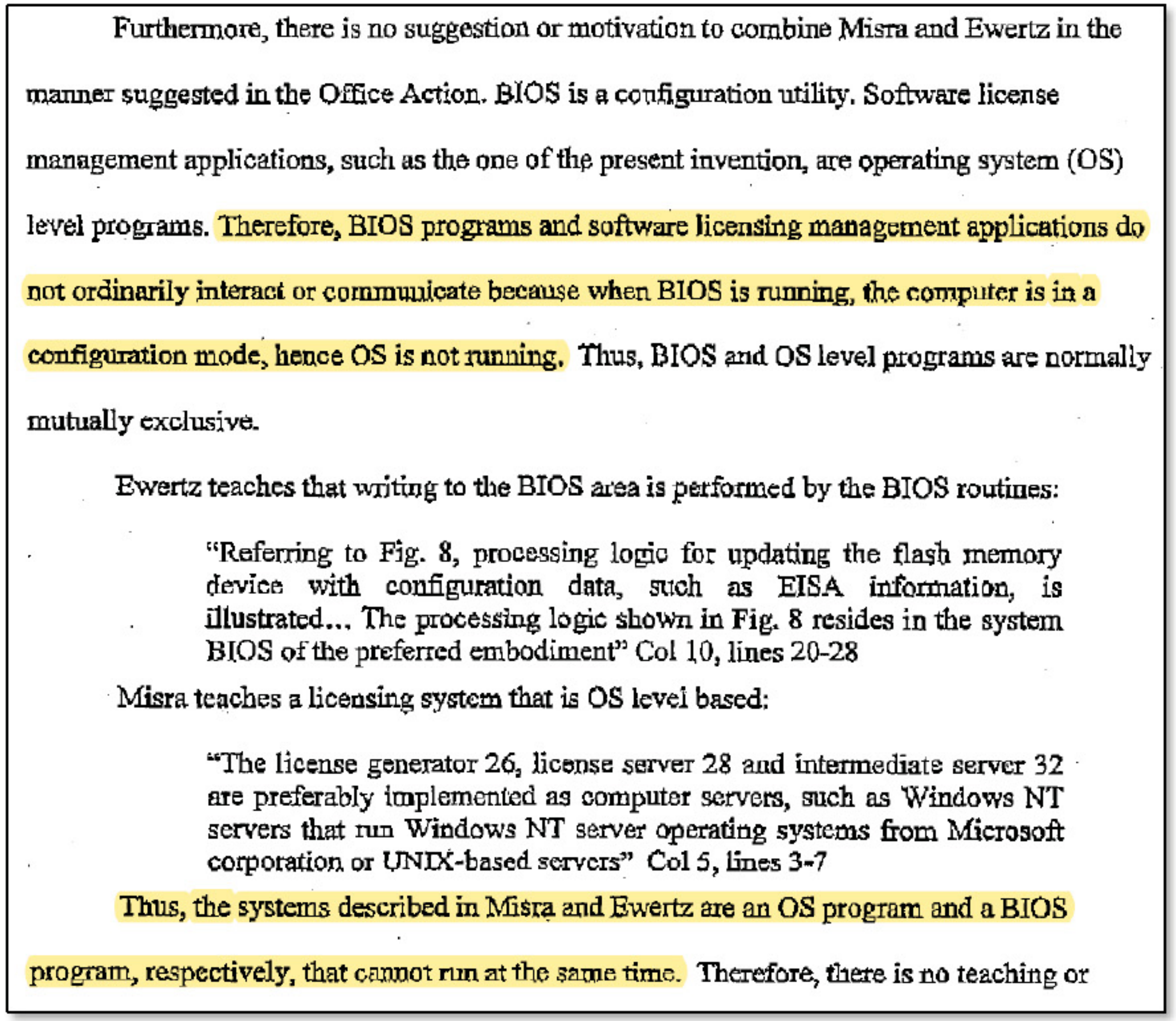
Ancora Techs., Inc. v. Apple, Inc., 744 F.3d 732, 736 (Fed. Cir. 2014).

- Reply 4-7; Ex. 1033 ¶4-15.

Agent: Ancora's Proposed Construction Is Itself Ambiguous

Ancora's First Characterization:

"OS-Level" is Anything that Runs After BIOS Setup Is Complete



Furthermore, there is no suggestion or motivation to combine Misra and Ewertz in the manner suggested in the Office Action. BIOS is a configuration utility. Software license management applications, such as the one of the present invention, are operating system (OS) level programs. Therefore, BIOS programs and software licensing management applications do not ordinarily interact or communicate because when BIOS is running, the computer is in a configuration mode, hence OS is not running. Thus, BIOS and OS level programs are normally mutually exclusive.

Ewertz teaches that writing to the BIOS area is performed by the BIOS routines:

> "Referring to Fig. 8, processing logic for updating the flash memory device with configuration data, such as EISA information, is illustrated... The processing logic shown in Fig. 8 resides in the system BIOS of the preferred embodiment" Col 10, lines 20-28

Misra teaches a licensing system that is OS level based:

> "The license generator 26, license server 28 and intermediate server 32 are preferably implemented as computer servers, such as Windows NT servers that run Windows NT server operating systems from Microsoft corporation or UNIX-based servers" Col 5, lines 3-7

Thus, the systems described in Misra and Ewertz are an OS program and a BIOS program, respectively, that cannot run at the same time. Therefore, there is no teaching or

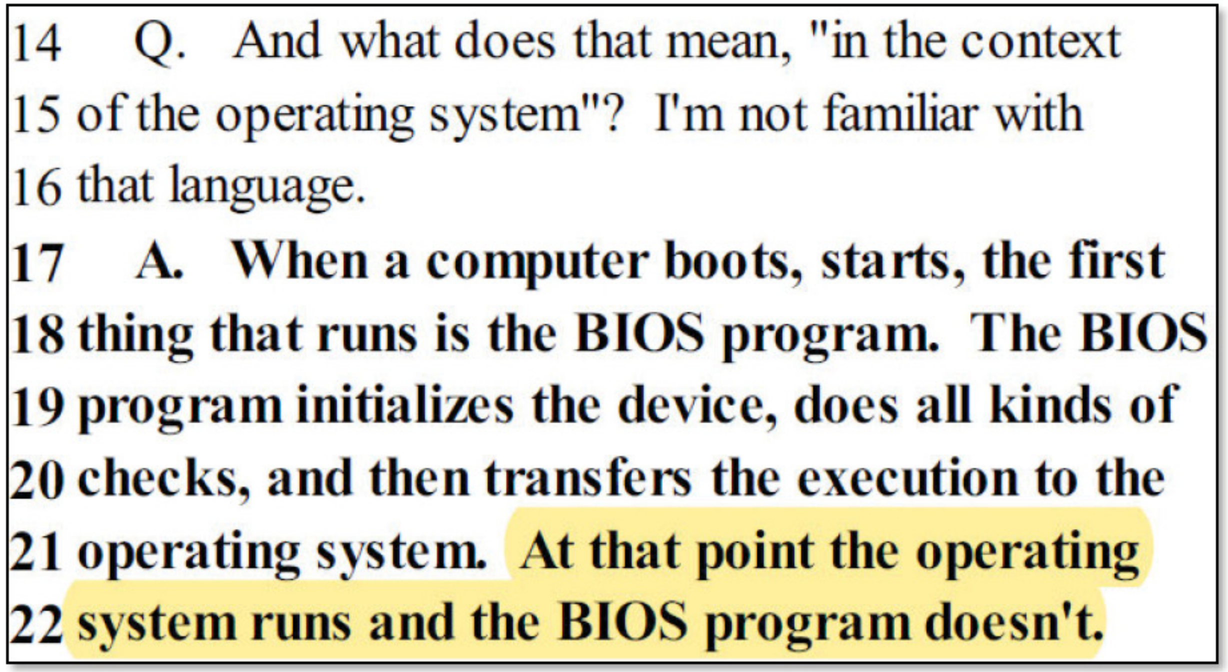
Office Action Response (Feb. 5, 2002)

- Reply 7-11; Ex. 1033 ¶27.

Agent: Ancora's Proposed Construction Is Itself Ambiguous

Ancora's First Characterization:

"OS Level" as Anything that Runs After BIOS Setup Is Complete



Q. And what does that mean, "in the context of the operating system"? I'm not familiar with that language.

A. When a computer boots, starts, the first thing that runs is the BIOS program. The BIOS program initializes the device, does all kinds of checks, and then transfers the execution to the operating system. At that point the operating system runs and the BIOS program doesn't.

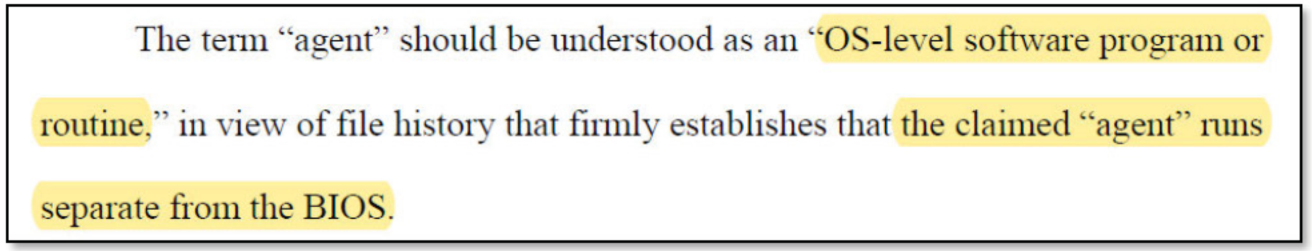
EX. 1034, 59:14-22 (Deposition of inventor/owner, Miki Mullor).

- Reply 7-11; Ex. 1033 ¶27.

Agent: Ancora's Proposed Construction Is Itself Ambiguous

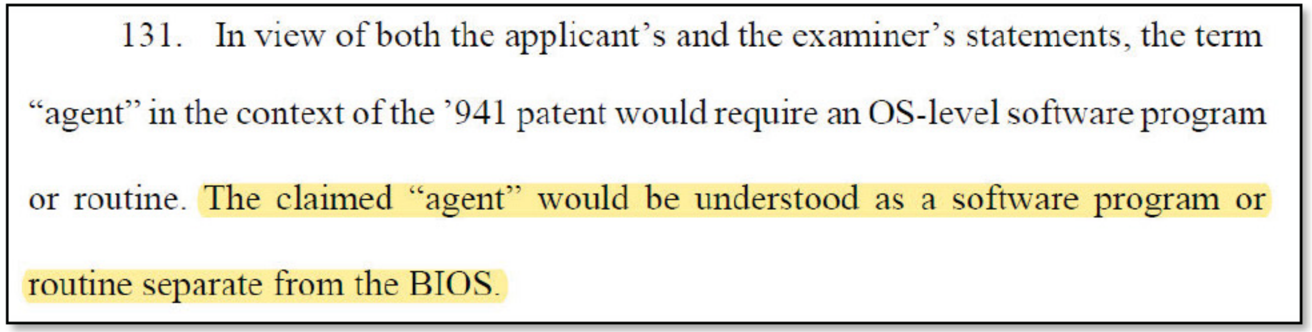
Ancora's Second Characterization:

"OS Level" as Anything Separate from the BIOS



The term "agent" should be understood as an "OS-level software program or routine," in view of file history that firmly establishes that the claimed "agent" runs separate from the BIOS.

POR at 32.



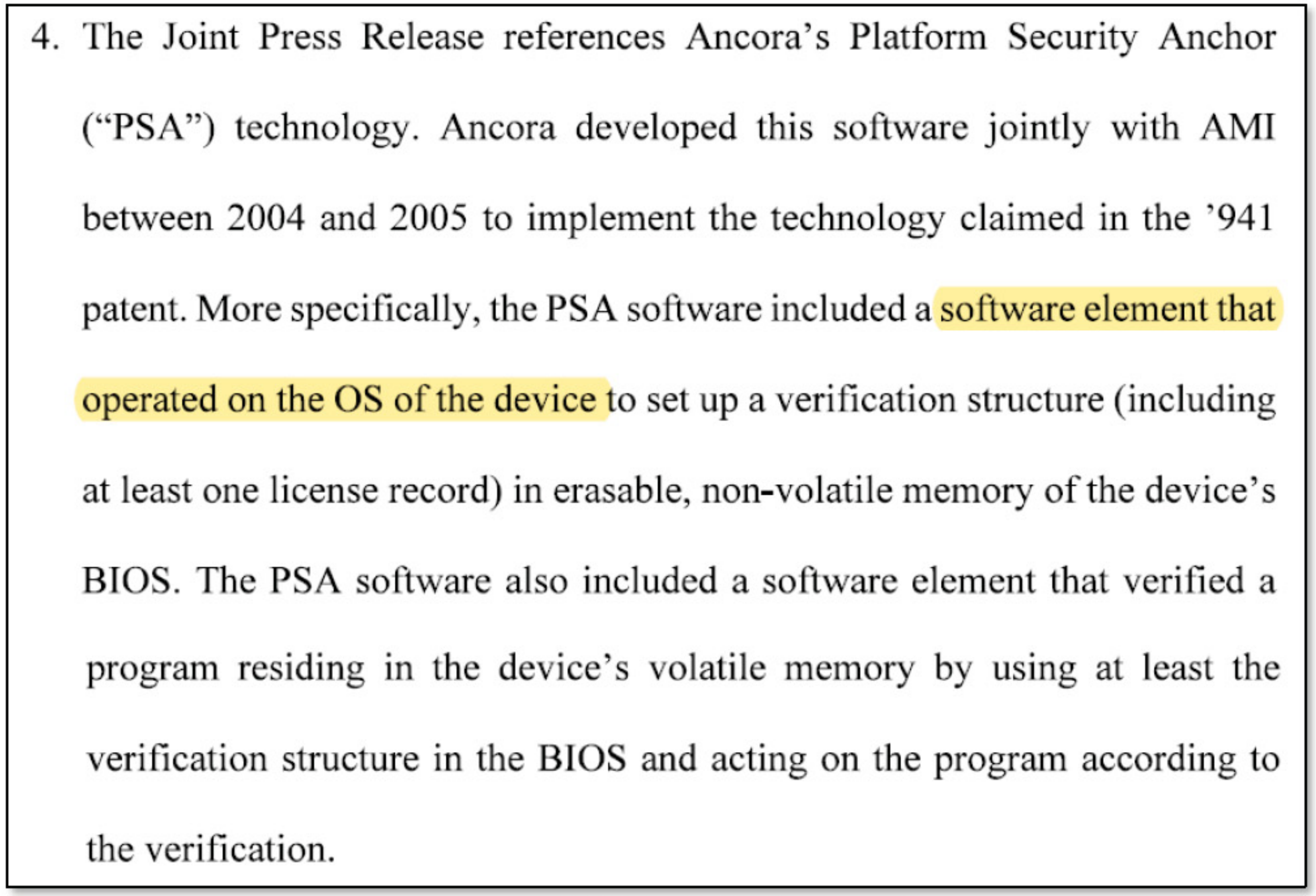
131. In view of both the applicant's and the examiner's statements, the term "agent" in the context of the '941 patent would require an OS-level software program or routine. The claimed "agent" would be understood as a software program or routine separate from the BIOS.

EX. 2018, ¶131 (Declaration of Ancora's expert, Dr. Martin).

- Reply 7-11; Ex. 1033 ¶27.

Agent: Ancora's Proposed Construction Is Itself Ambiguous

"OS Level" as Something Else…



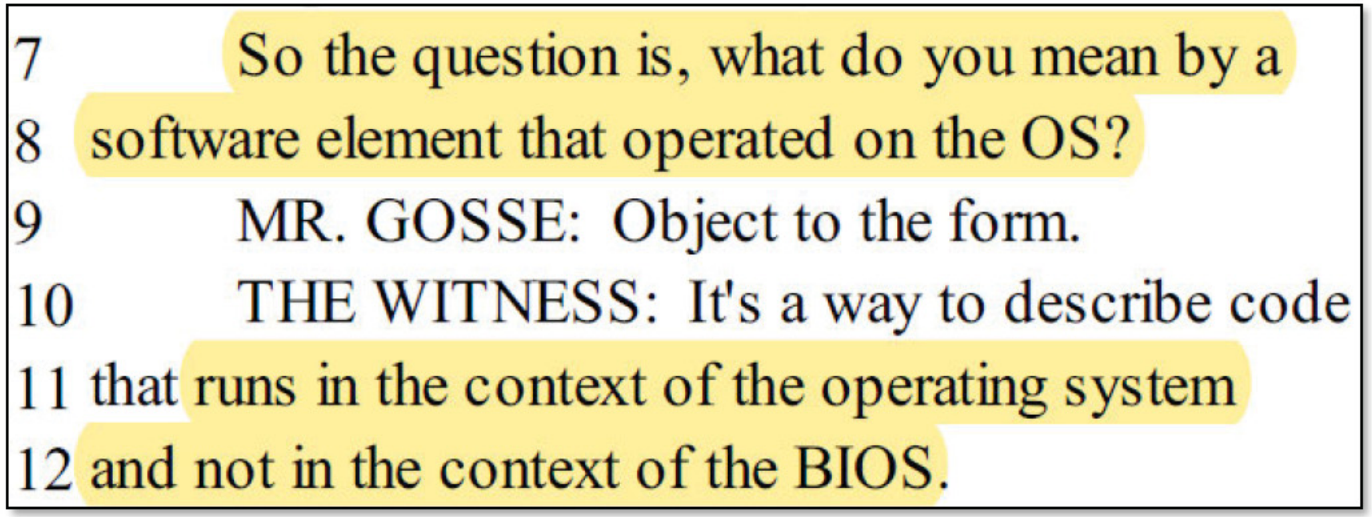
4. The Joint Press Release references Ancora's Platform Security Anchor ("PSA") technology. Ancora developed this software jointly with AMI between 2004 and 2005 to implement the technology claimed in the '941 patent. More specifically, the PSA software included a software element that operated on the OS of the device to set up a verification structure (including at least one license record) in erasable, non-volatile memory of the device's BIOS. The PSA software also included a software element that verified a program residing in the device's volatile memory by using at least the verification structure in the BIOS and acting on the program according to the verification.

EX. 2030, ¶4 (Declaration of inventor/owner, Miki Mullor).

- Reply 7-11; Ex. 1033 ¶27.

Agent: Ancora's Proposed Construction Is Itself Ambiguous

"OS Level" as Something Else…



So the question is, what do you mean by a software element that operated on the OS?

MR. GOSSE: Object to the form.

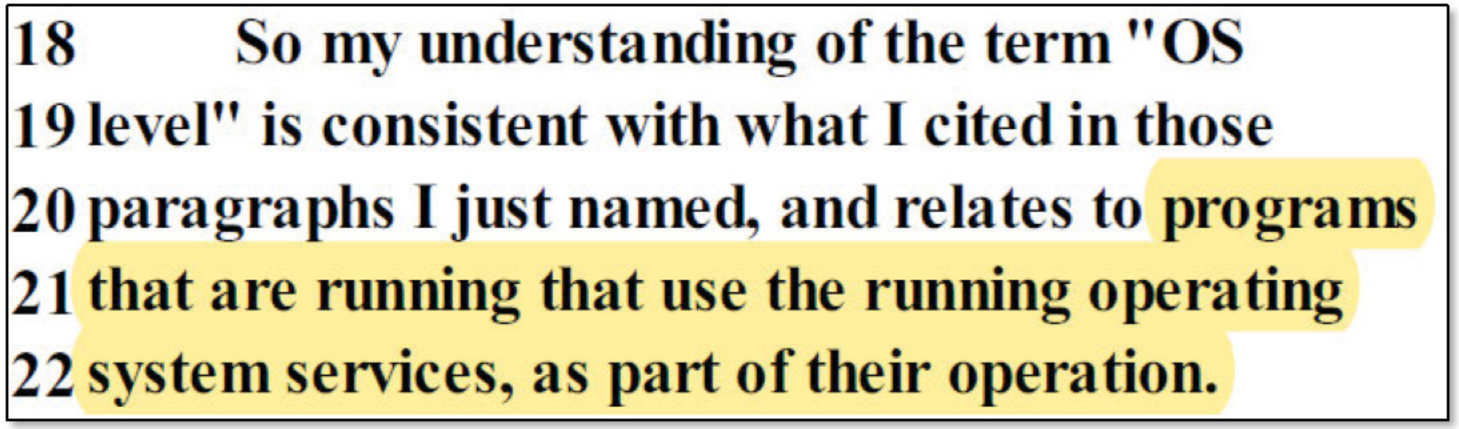
THE WITNESS: It's a way to describe code that runs in the context of the operating system and not in the context of the BIOS.

EX. 1034, 59:7-12 (Deposition of inventor/owner, Miki Mullor).

- Reply 7-11; Ex. 1033 ¶27.

Agent: Ancora's Proposed Construction Is Itself Ambiguous

"OS Level" as Something Else…



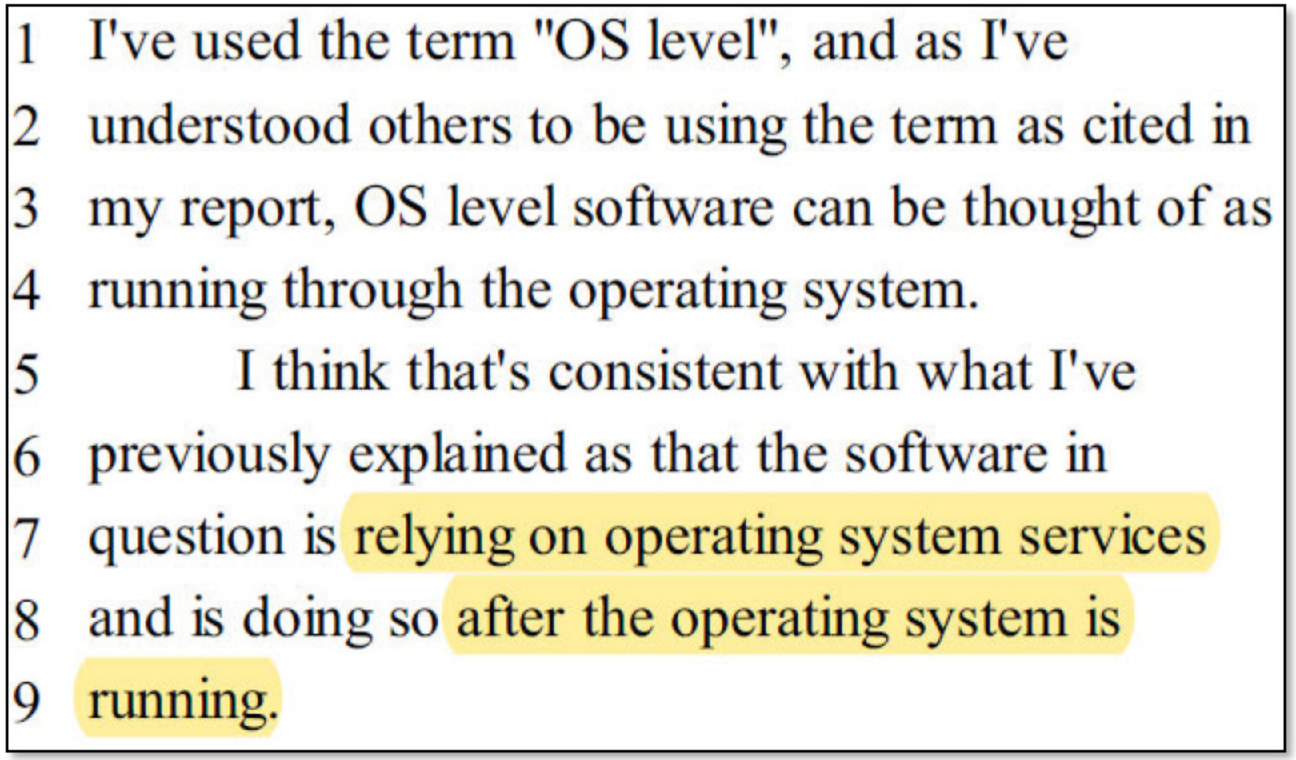
So my understanding of the term "OS level" is consistent with what I cited in those paragraphs I just named, and relates to programs that are running that use the running operating system services, as part of their operation.

EX. 1035, 100:18-22 (Deposition of Ancora's expert, Dr. Martin).

- Reply 7-11; Ex. 1033 ¶27.

Agent: Ancora's Proposed Construction Is Itself Ambiguous

"OS Level" as Something Else…



I've used the term "OS level", and as I've understood others to be using the term as cited in my report, OS level software can be thought of as running through the operating system.

I think that's consistent with what I've previously explained as that the software in question is relying on operating system services and is doing so after the operating system is running.

EX. 1035, 102:1-9 (Deposition of Ancora's expert, Dr. Martin).

- Reply 7-11; Ex. 1033 ¶27.

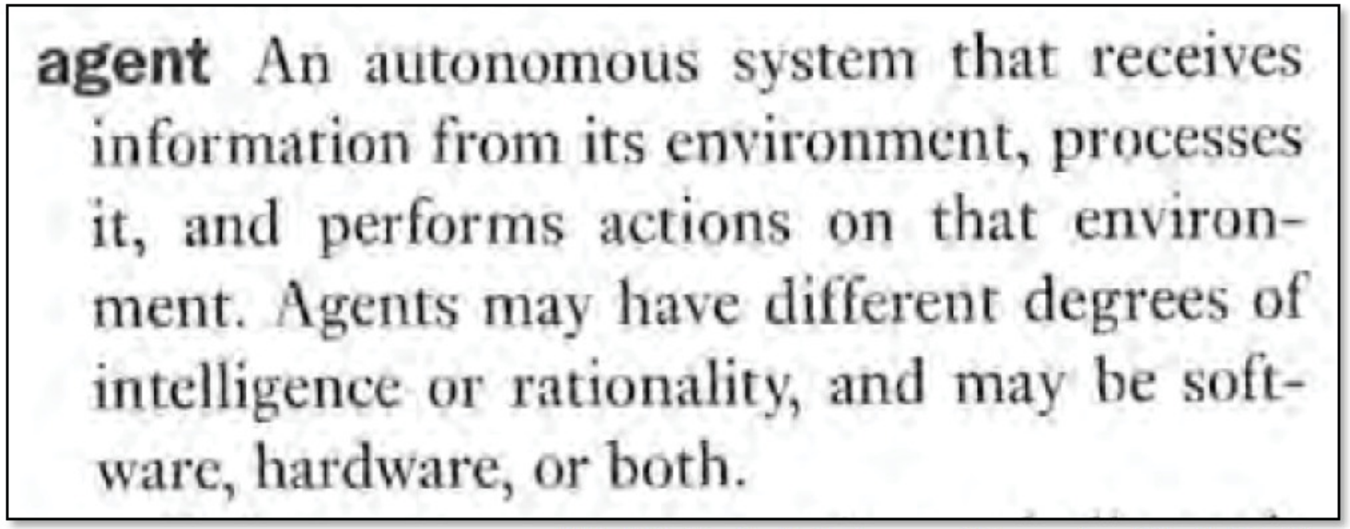
Overview

1. Background
2. The “using an agent” Limitation.
    1. There is no disclaimer of agent to “OS-level.”
    2. There is no disclaimer or other basis for the “software-only” limitation.
    3. Hellman discloses or renders obvious an “OS-level,” “software-only” agent.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Agent: No Basis for Software-Only Limitation

"Agent": Not mentioned in specification,
or the original claims.



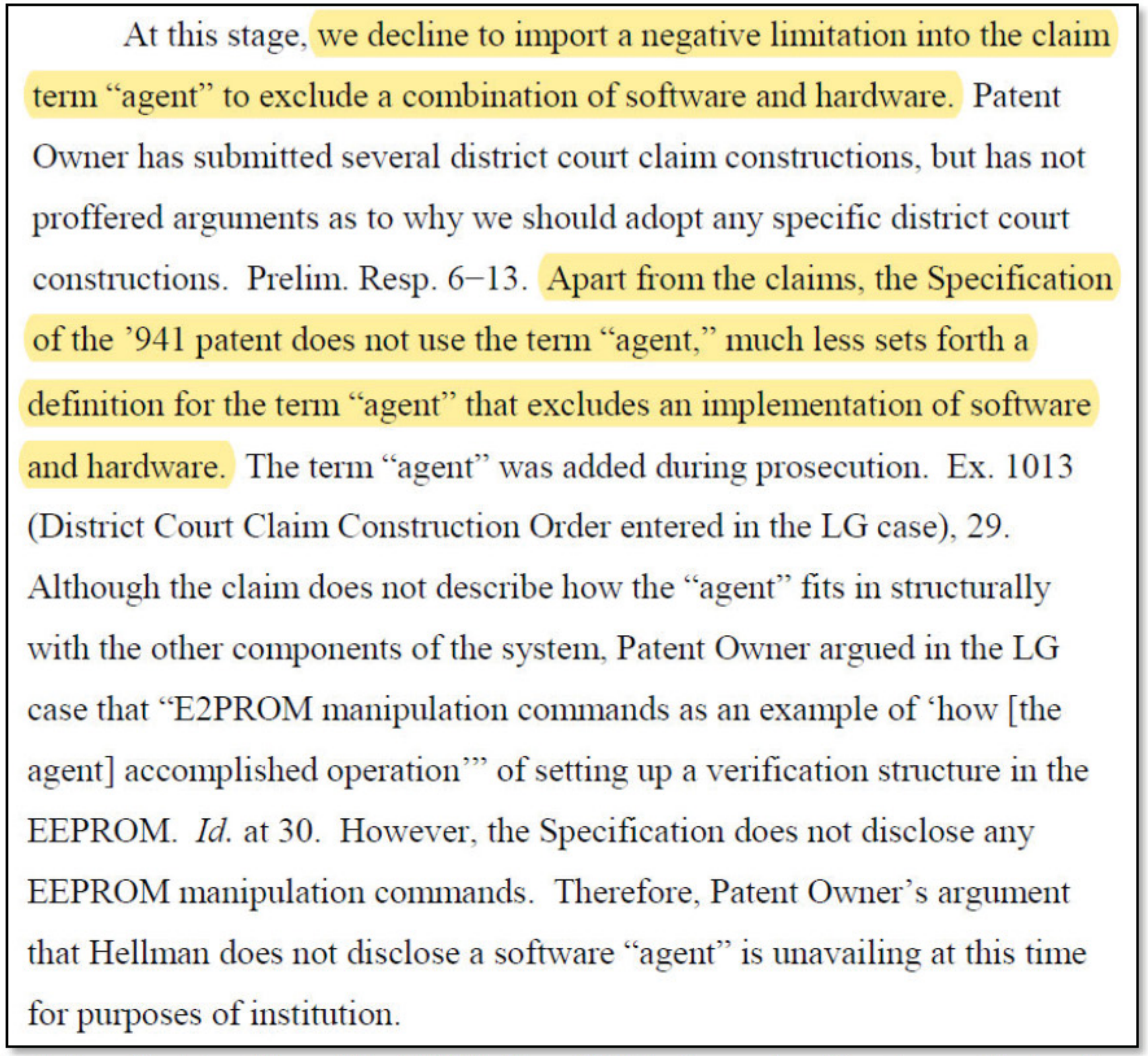
agent An autonomous system that receives information from its environment, processes it, and performs actions on that environment. Agents may have different degrees of intelligence or rationality, and may be software, hardware, or both.

EX. 1038 (Oxford Dictionary of Computing (4th ed. 1996)).

- Reply 14-17; Ex. 1033 ¶4-15.

Agent: No Basis for Software-Only Limitation

The Board previously considered and rejected Ancora's software-only negative limitation.



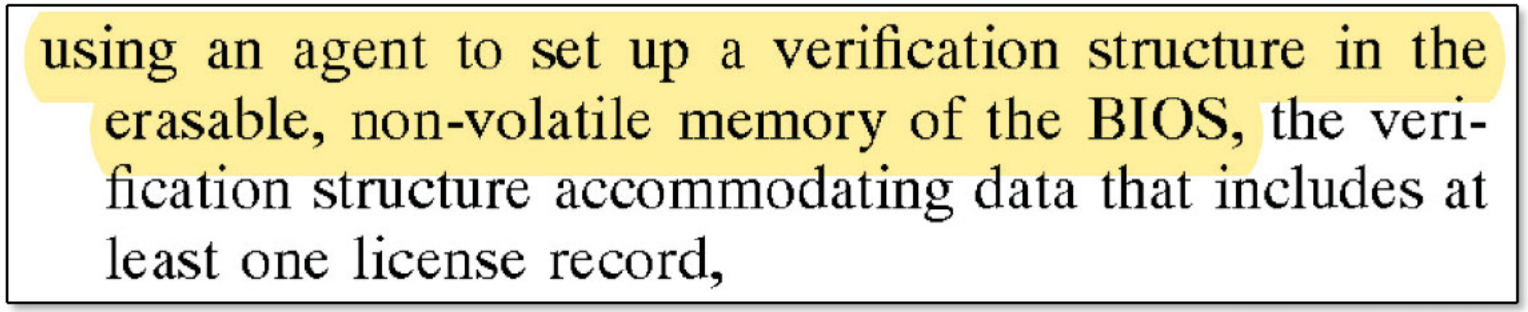
At this stage, we decline to import a negative limitation into the claim term "agent" to exclude a combination of software and hardware. Patent Owner has submitted several district court claim constructions, but has not proffered arguments as to why we should adopt any specific district court constructions. Prelim. Resp. 6–13. Apart from the claims, the Specification of the '941 patent does not use the term "agent," much less sets forth a definition for the term "agent" that excludes an implementation of software and hardware. The term "agent" was added during prosecution. Ex. 1013 (District Court Claim Construction Order entered in the LG case), 29. Although the claim does not describe how the "agent" fits in structurally with the other components of the system, Patent Owner argued in the LG case that "E2PROM manipulation commands as an example of 'how [the agent] accomplished operation'" of setting up a verification structure in the EEPROM. *Id.* at 30. However, the Specification does not disclose any EEPROM manipulation commands. Therefore, Patent Owner's argument that Hellman does not disclose a software "agent" is unavailing at this time for purposes of institution.

Institution Decision (Paper 17) at 28-29, *Sony Mobile Commc'ns AB v. Ancora Techs., Inc.*, IPR2021-00663 (June 10, 2021).

- Reply 14-17; Ex. 1033 ¶4-15.

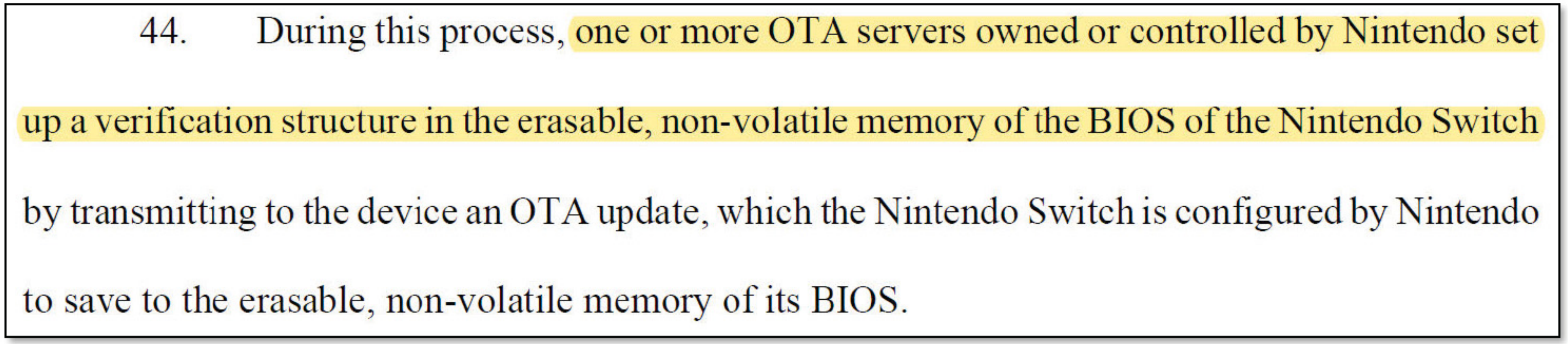
Agent: No Basis for Software-Only Limitation

Ancora's infringement contentions accused a combination of hardware and software.



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,

'941 patent, claim 1.



44. During this process, one or more OTA servers owned or controlled by Nintendo set up a verification structure in the erasable, non-volatile memory of the BIOS of the Nintendo Switch by transmitting to the device an OTA update, which the Nintendo Switch is configured by Nintendo to save to the erasable, non-volatile memory of its BIOS.

EX. 1076, ¶44 (Ancora complaint for infringement against Nintendo).

- Reply 14-17; Ex. 1033 ¶4-15.

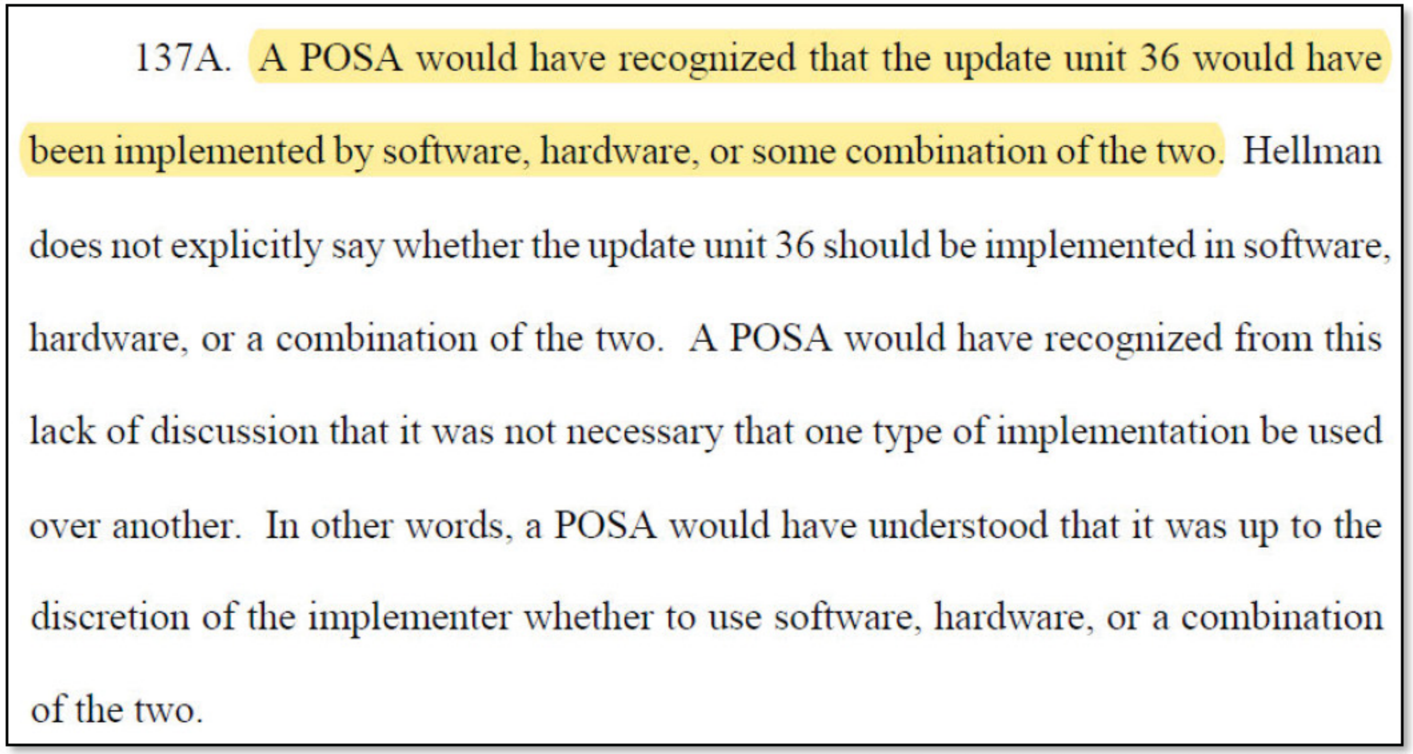
Overview

1. Background
2. The “using an agent” Limitation.
    1. There is no disclaimer of agent to “OS-level.”
    2. There is no disclaimer or other basis for the “software-only” limitation.
    3. Hellman discloses or renders obvious an “OS-level,” “software-only” agent.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Agent: Hellman Includes Ancora's Narrowed Agent

Hellman discloses a software-only agent.



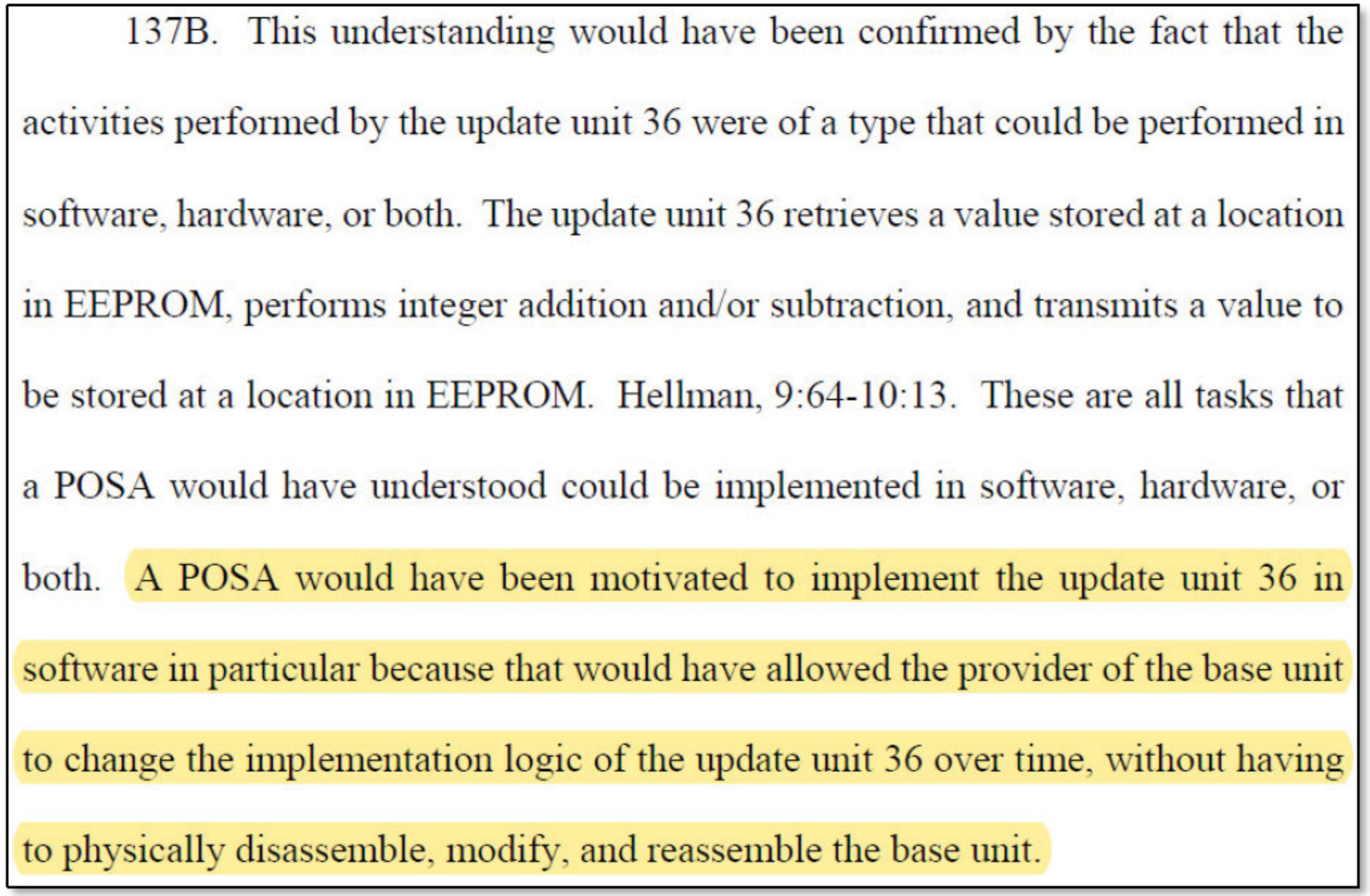
137A. A POSA would have recognized that the update unit 36 would have been implemented by software, hardware, or some combination of the two. Hellman does not explicitly say whether the update unit 36 should be implemented in software, hardware, or a combination of the two. A POSA would have recognized from this lack of discussion that it was not necessary that one type of implementation be used over another. In other words, a POSA would have understood that it was up to the discretion of the implementer whether to use software, hardware, or a combination of the two.

EX. 1003, ¶137A (Dr. Wolfe Opening Decl.).

- Pet. 40-41; Ex. 1003 ¶137-138B;
Reply 17-18; Ex. 1033 ¶17-26.

Agent: Hellman Includes Ancora's Narrowed Agent

Hellman also renders obvious a software-only agent.



137B. This understanding would have been confirmed by the fact that the activities performed by the update unit 36 were of a type that could be performed in software, hardware, or both. The update unit 36 retrieves a value stored at a location in EEPROM, performs integer addition and/or subtraction, and transmits a value to be stored at a location in EEPROM. Hellman, 9:64-10:13. These are all tasks that a POSA would have understood could be implemented in software, hardware, or both. A POSA would have been motivated to implement the update unit 36 in software in particular because that would have allowed the provider of the base unit to change the implementation logic of the update unit 36 over time, without having to physically disassemble, modify, and reassemble the base unit.

EX. 1003, ¶137B (Dr. Wolfe Opening Decl.).

- Pet. 40-41; Ex. 1003 ¶137-138B; Reply 17-18; Ex. 1033 ¶17-26.

Agent: Hellman Includes Ancora's Narrowed Agent

Hellman's update unit 36 meets Ancora's characterization of "OS-level."

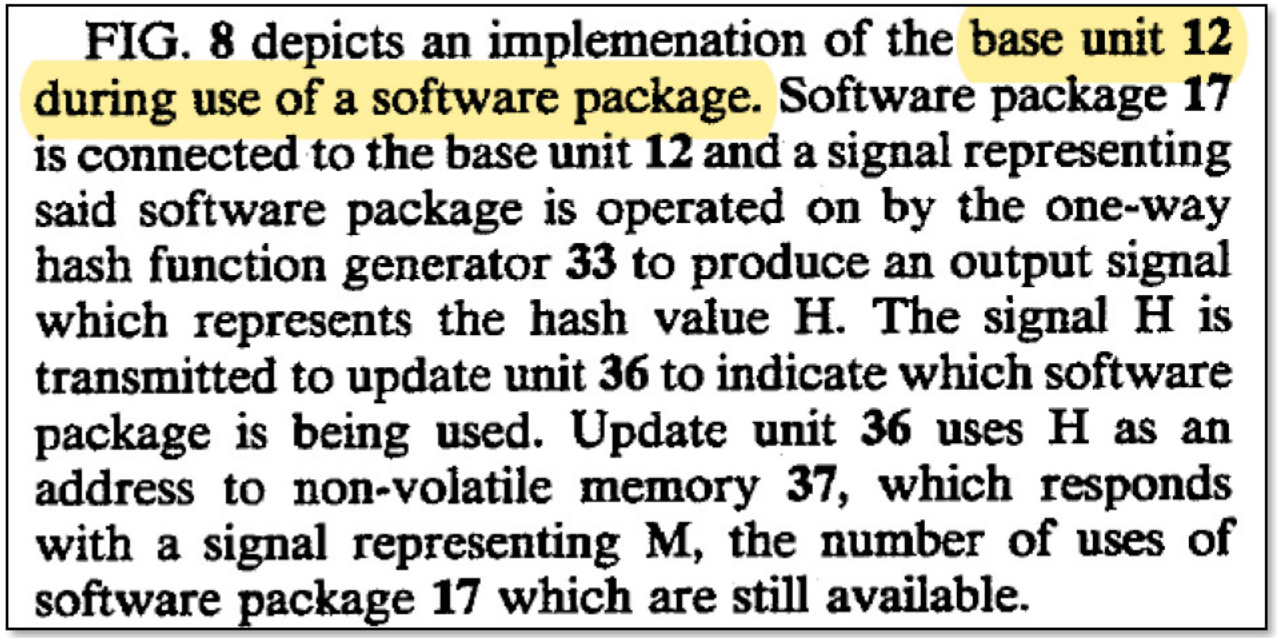


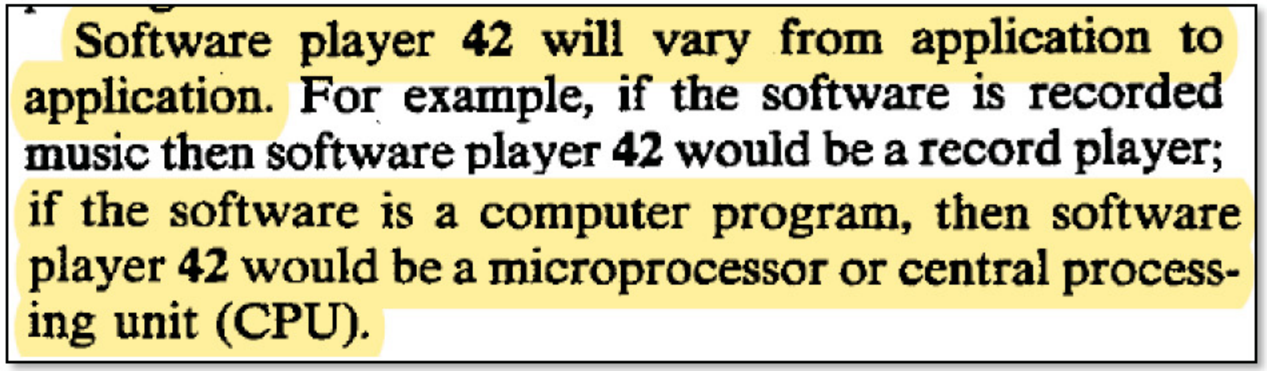
FIG. 8 depicts an implemenation of the base unit 12 during use of a software package. Software package 17 is connected to the base unit 12 and a signal representing said software package is operated on by the one-way hash function generator 33 to produce an output signal which represents the hash value H. The signal H is transmitted to update unit 36 to indicate which software package is being used. Update unit 36 uses H as an address to non-volatile memory 37, which responds with a signal representing M, the number of uses of software package 17 which are still available.

Hellman at 10:33-43.

- Reply 11-14; Ex. 1033 ¶27-41.

Agent: Hellman Includes Ancora's Narrowed Agent

Hellman's update unit 36 meets Ancora's characterization of "OS-level."



Software player 42 will vary from application to application. For example, if the software is recorded music then software player 42 would be a record player; if the software is a computer program, then software player 42 would be a microprocessor or central processing unit (CPU).

Hellman at 10:66-11:3.

- Reply 11-14; Ex. 1033 ¶27-41.

Agent: Hellman Includes Ancora's Narrowed Agent

Hellman's update unit 36 meets Ancora's characterization of "OS-level."

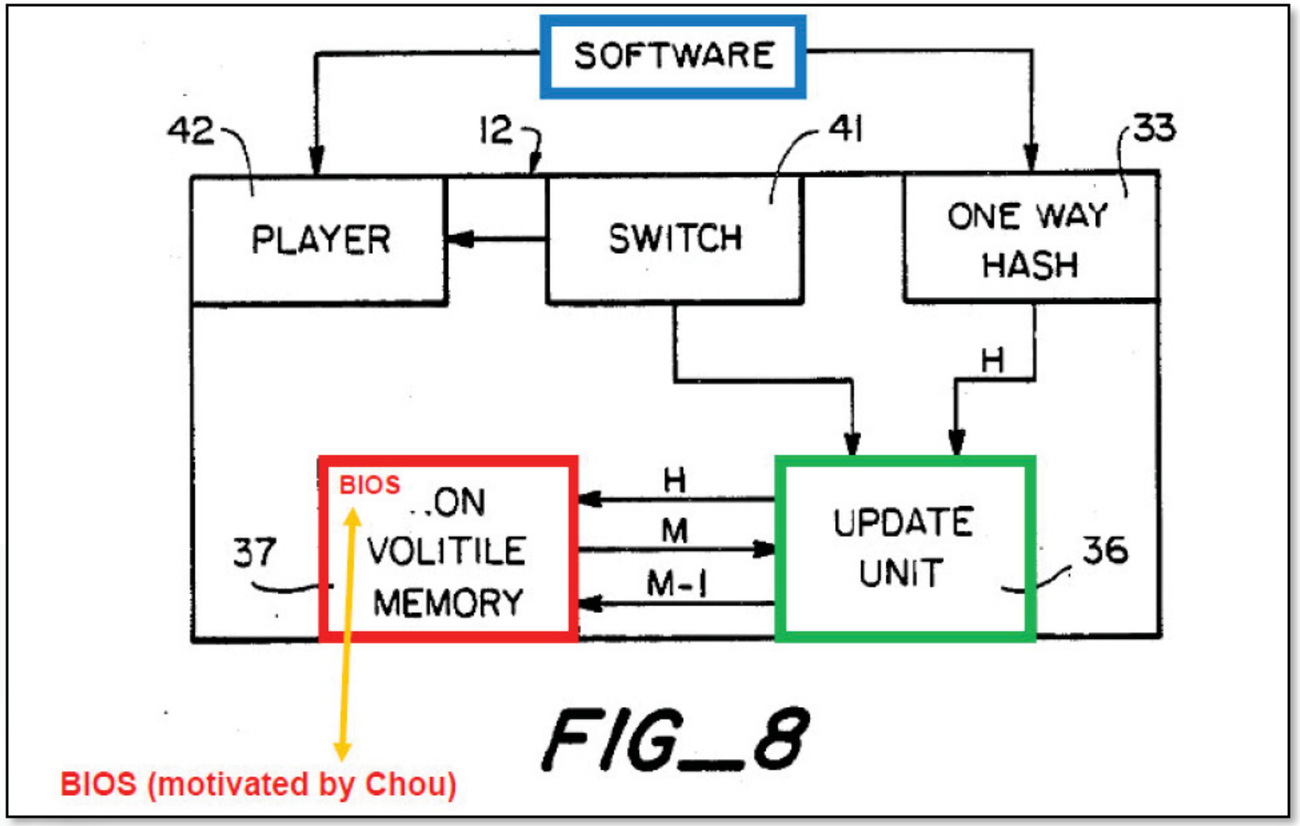


FIG_8

EX. 1033, ¶30 (Dr. Wolfe Reply Decl.).

- Reply 11-14; Ex. 1033 ¶27-41.

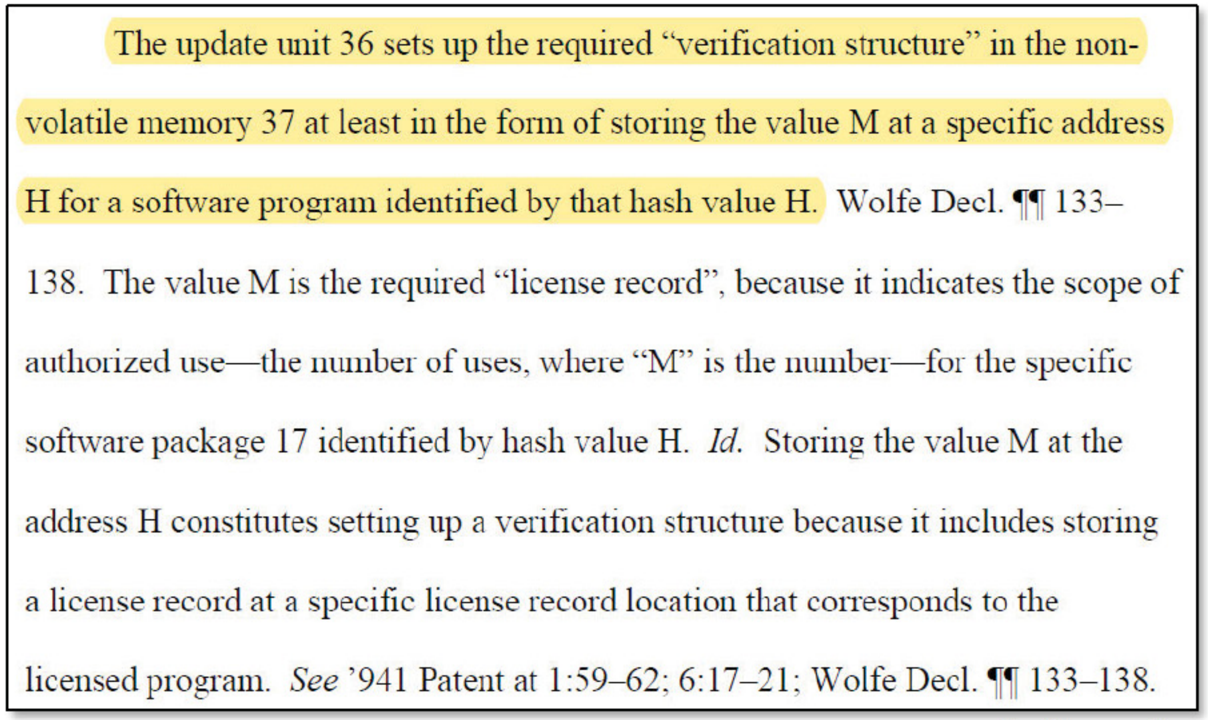
Overview

1. Background
2. The “using an agent” Limitation.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Verification Structure

Hellman's update unit 36 sets up a "verification structure"



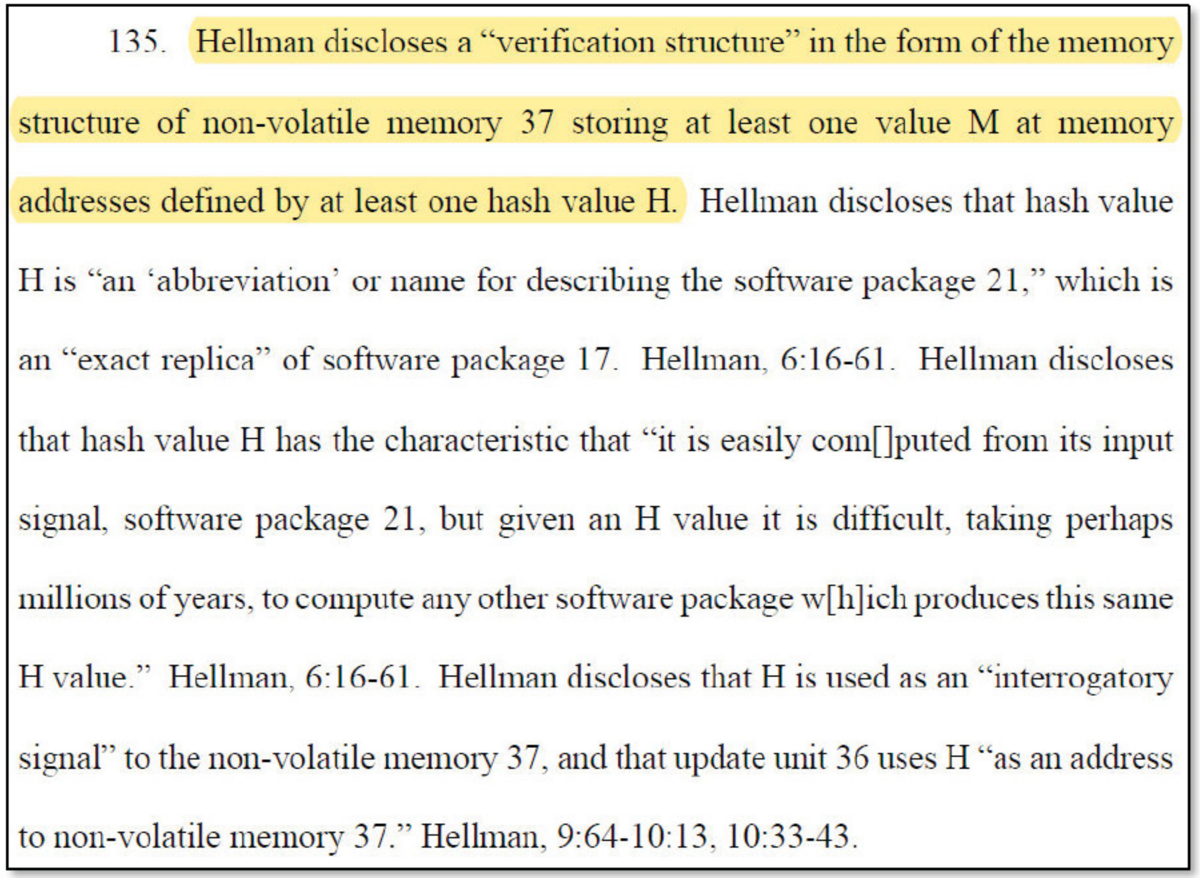
The update unit 36 sets up the required "verification structure" in the non-volatile memory 37 at least in the form of storing the value M at a specific address H for a software program identified by that hash value H. Wolfe Decl. ¶¶ 133–138. The value M is the required "license record", because it indicates the scope of 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 that corresponds to the licensed program. *See* '941 Patent at 1:59–62; 6:17–21; Wolfe Decl. ¶¶ 133–138.

Petition at 41.

- Pet. 40-41; Ex. 1003 ¶135-136; Reply 18-20; Ex. 1033 ¶42-47.

Verification Structure

Hellman's update unit 36 sets up a "verification structure"



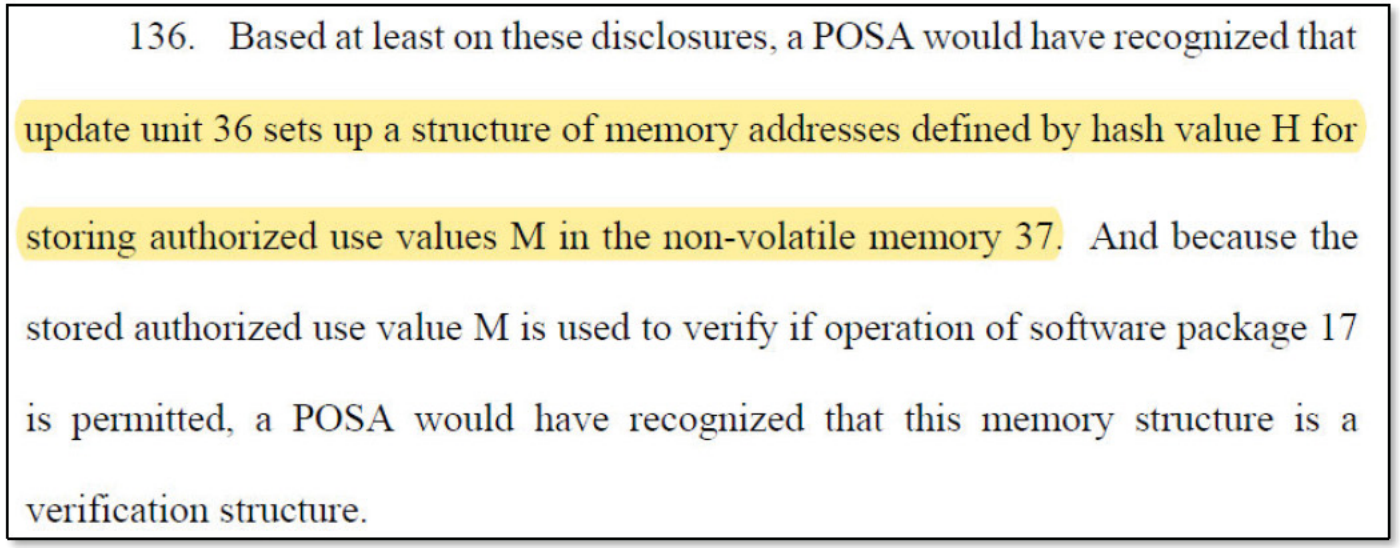
135. Hellman discloses a "verification structure" in the form of the memory structure of non-volatile memory 37 storing at least one value M at memory addresses defined by at least one hash value H. Hellman discloses that hash value H is "an 'abbreviation' or name for describing the software package 21," which is an "exact replica" of software package 17. Hellman, 6:16-61. Hellman discloses that hash value H has the characteristic that "it is easily com[]puted from its input signal, software package 21, but given an H value it is difficult, taking perhaps millions of years, to compute any other software package w[h]ich produces this same H value." Hellman, 6:16-61. Hellman discloses that H is used as an "interrogatory signal" to the non-volatile memory 37, and that update unit 36 uses H "as an address to non-volatile memory 37." Hellman, 9:64-10:13, 10:33-43.

EX. 1003, ¶135 (Dr. Wolfe Opening Decl.).

- Pet. 40-41; Ex. 1003 ¶135-136; Reply 18-20; Ex. 1033 ¶42-47.

Verification Structure

Hellman's update unit 36 sets up a "verification structure"



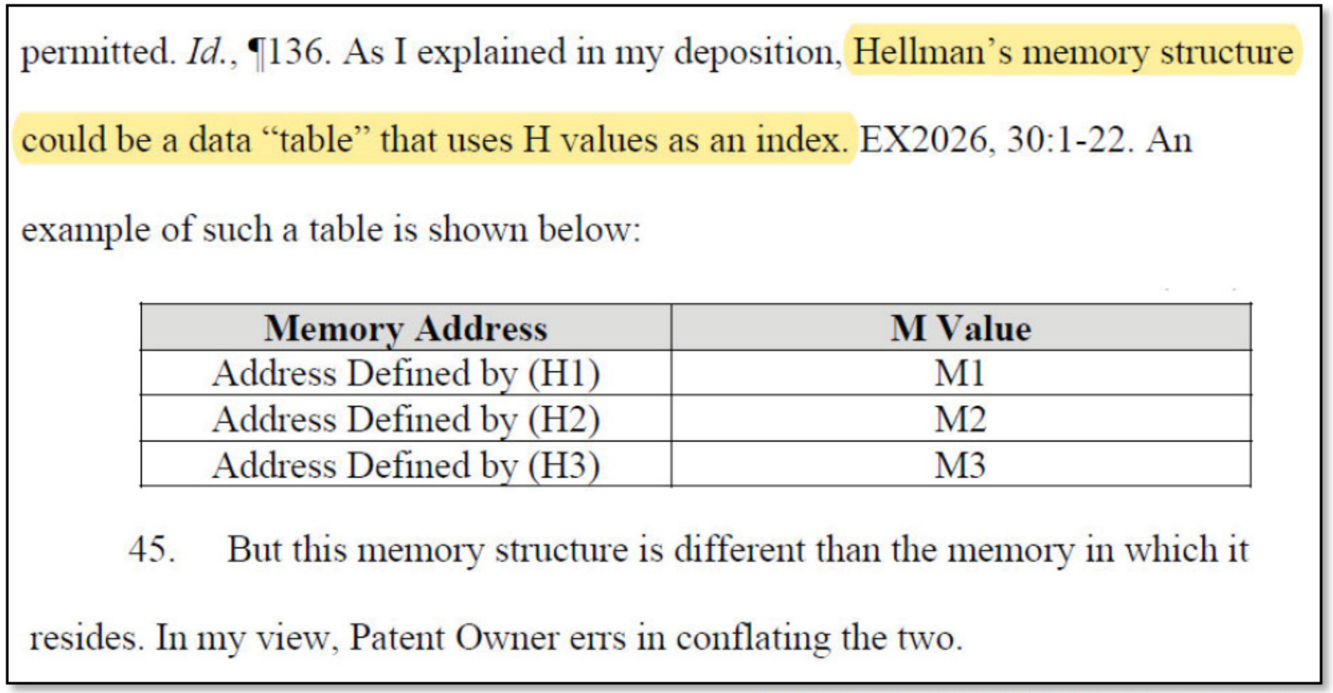
136. Based at least on these disclosures, a POSA would have recognized that update unit 36 sets up a structure of memory addresses defined by hash value H for storing authorized use values M in the non-volatile memory 37. And because the stored authorized use value M is used to verify if operation of software package 17 is permitted, a POSA would have recognized that this memory structure is a verification structure.

EX. 1003, ¶136 (Dr. Wolfe Opening Decl.).

- Pet. 40-41; Ex. 1003 ¶135-136; Reply 18-20; Ex. 1033 ¶42-47.

Verification Structure

Hellman's update unit 36 sets up a "verification structure"



permitted. *Id.*, ¶136. As I explained in my deposition, Hellman's memory structure could be a data "table" that uses H values as an index. EX2026, 30:1-22. An example of such a table is shown below:

Memory Address	M Value
Address Defined by (H1)	M1
Address Defined by (H2)	M2
Address Defined by (H3)	M3

45. But this memory structure is different than the memory in which it resides. In my view, Patent Owner errs in conflating the two.

EX. 1033, ¶44-45 (Dr. Wolfe Reply Decl.).

- Pet. 40-41; Ex. 1003 ¶135-136; Reply 18-20; Ex. 1033 ¶42-47.

Verification Structure

Patent Owner Position:

1. Unclear,
2. Unsupported by Expert Testimony,
3. Unsupported by the Intrinsic Record.

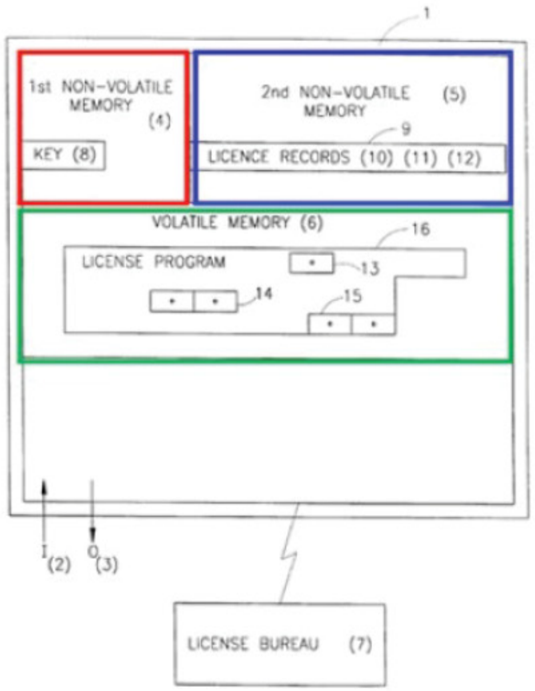
- Pet. 40-41; Ex. 1003 ¶135-136; Reply 18-20; Ex. 1033 ¶42-47.

Overview

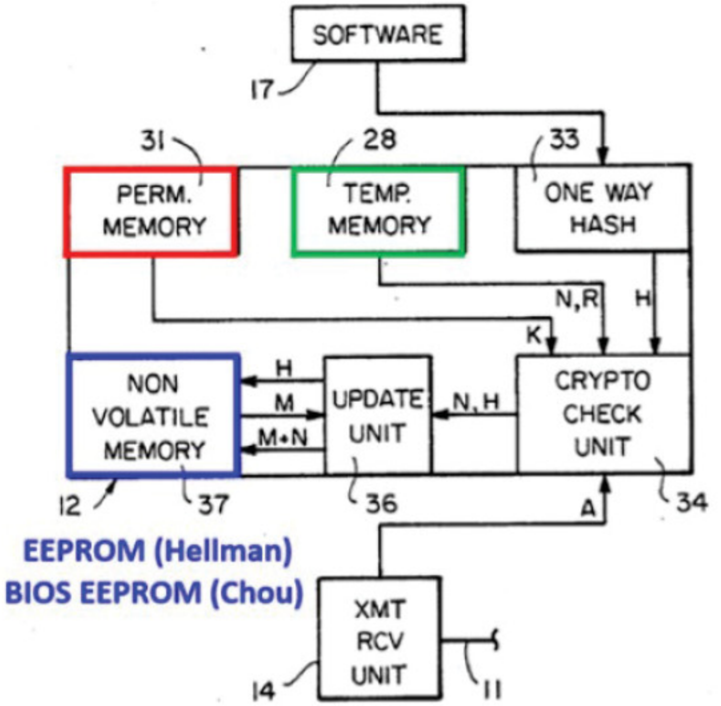
1. Background
2. The “using an agent” Limitation.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Motivation to Combine: Overview


1
1st NON-VOLATILE MEMORY (4)
KEY (8)
2nd NON-VOLATILE MEMORY (5)
9
LICENCE RECORDS (10) (11) (12)
VOLATILE MEMORY (6)
16
LICENSE PROGRAM
13
14
15
I (2)
O (3)
LICENSE BUREAU (7)

’941 Patent, Fig. 1
(Annotated by Petitioner)



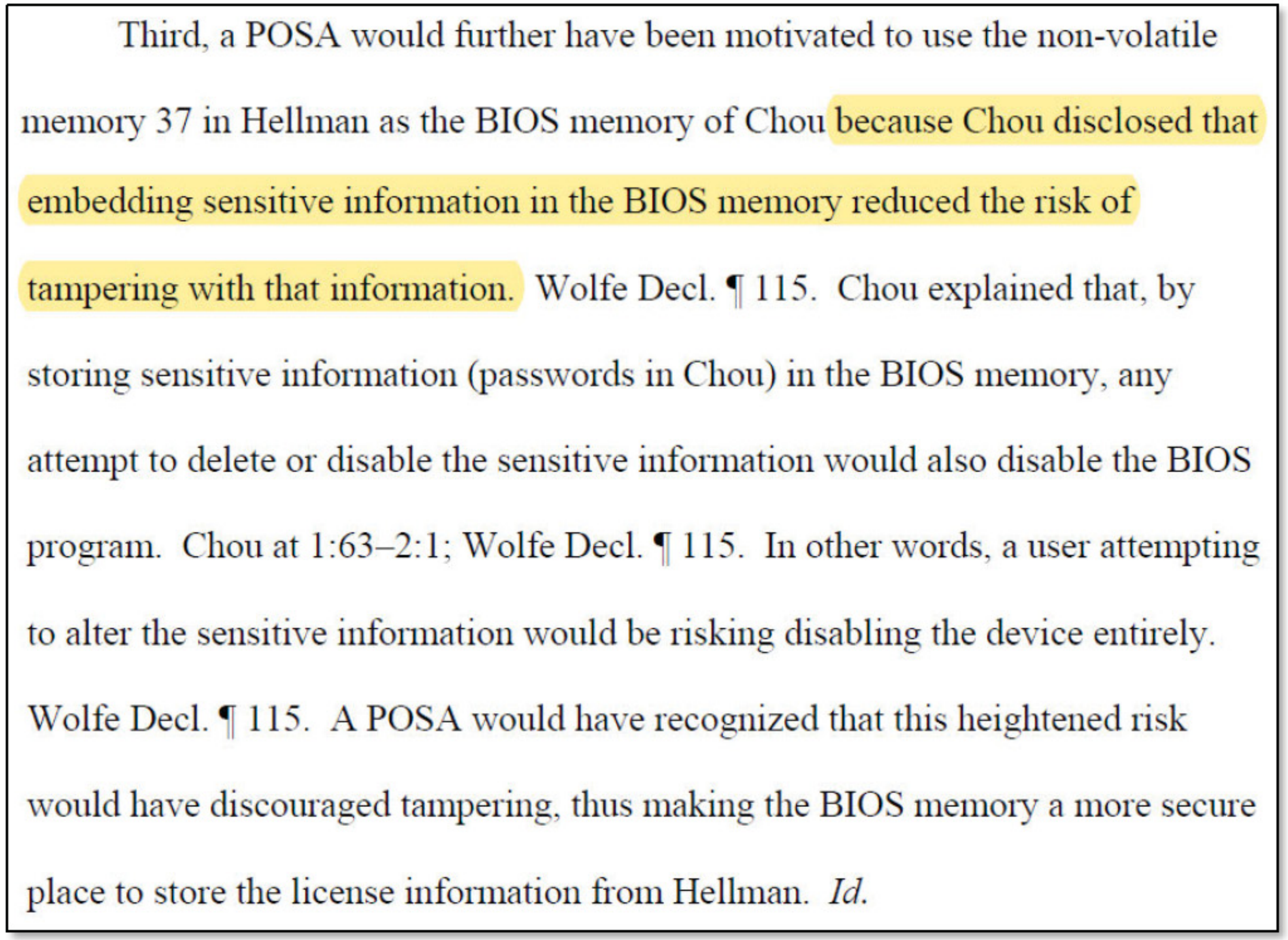
EEPROM (Hellman)
BIOS EEPROM (Chou)

Hellman, Fig. 6
(Annotated by Petitioner)

Petition at 31.

- Pet. 30-35; Ex. 1003 ¶105-120F;
 Reply 20-23; Ex. 1033 ¶48-61.

Motivation to Combine: Rationale #1

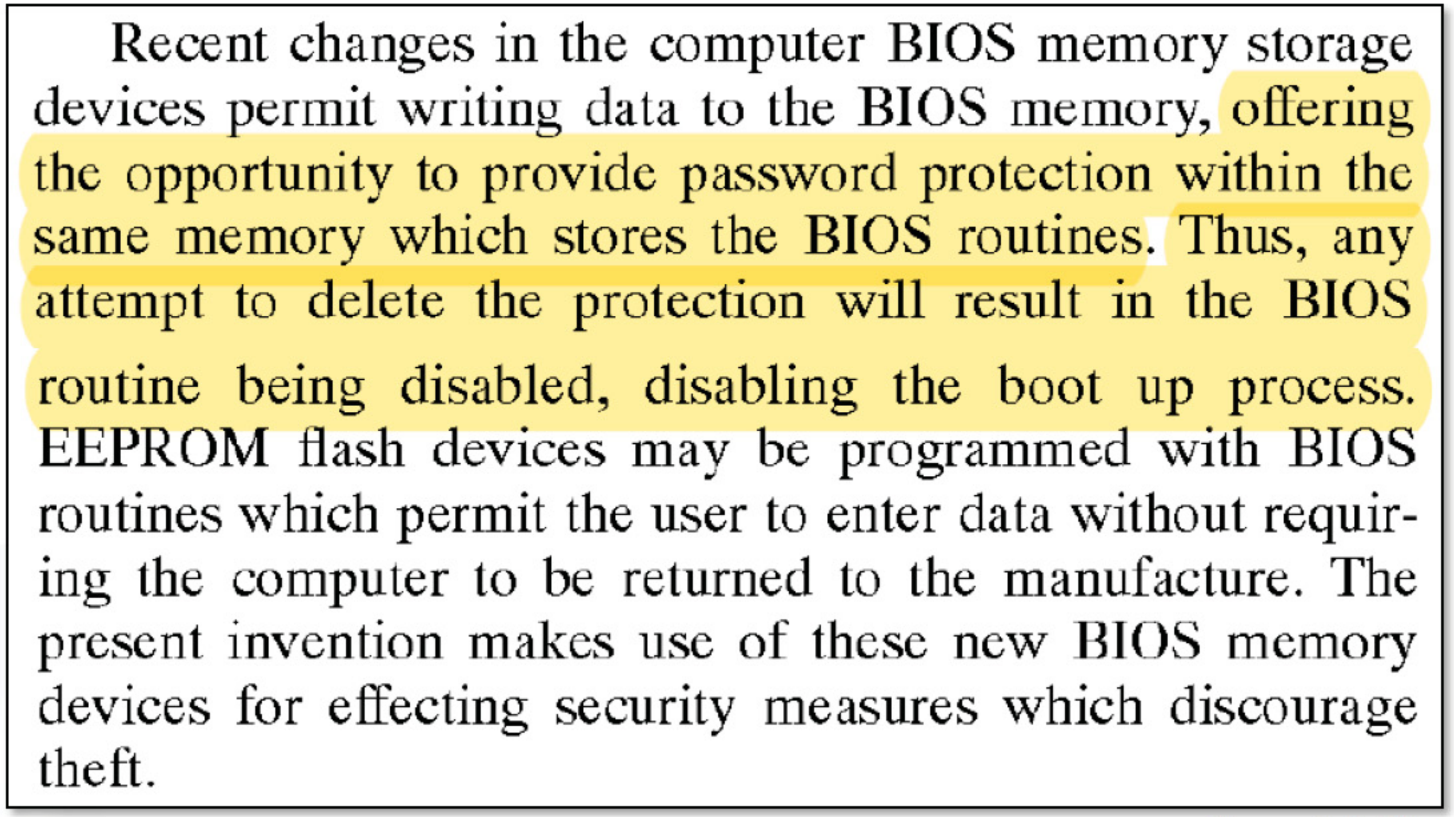


Third, a POSA would further have been motivated to use the non-volatile memory 37 in Hellman as the BIOS memory of Chou because Chou disclosed that embedding sensitive information in the BIOS memory reduced the risk of tampering with that information. Wolfe Decl. ¶ 115. Chou explained that, by storing sensitive information (passwords in Chou) in the BIOS memory, any attempt to delete or disable the sensitive information would also disable the BIOS program. Chou at 1:63–2:1; Wolfe Decl. ¶ 115. In other words, a user attempting to alter the sensitive information would be risking disabling the device entirely. Wolfe Decl. ¶ 115. A POSA would have recognized that this heightened risk would have discouraged tampering, thus making the BIOS memory a more secure place to store the license information from Hellman. *Id.*

Petition at 33-34.

- Pet. 33-34; Ex. 1003 ¶115;
Reply 20-21; Ex. 1033 ¶56.

Motivation to Combine: Rationale #1

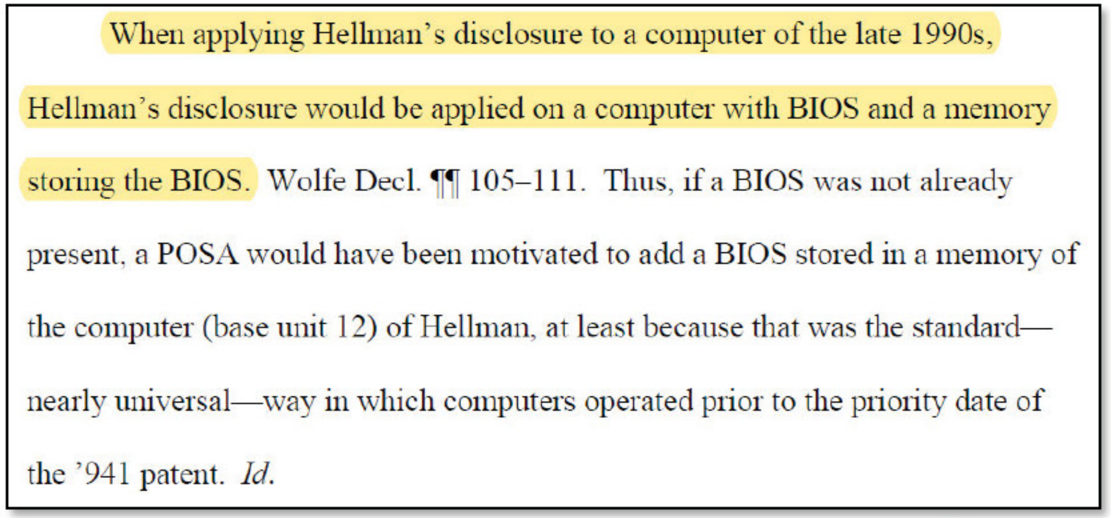


Recent changes in the computer BIOS memory storage devices permit writing data to the BIOS memory, offering the opportunity to provide password protection within the same memory which stores the BIOS routines. Thus, any attempt to delete the protection will result in the BIOS routine being disabled, disabling the boot up process. EEPROM flash devices may be programmed with BIOS routines which permit the user to enter data without requiring the computer to be returned to the manufacture. The present invention makes use of these new BIOS memory devices for effecting security measures which discourage theft.

Chou at 1:63-2:7.

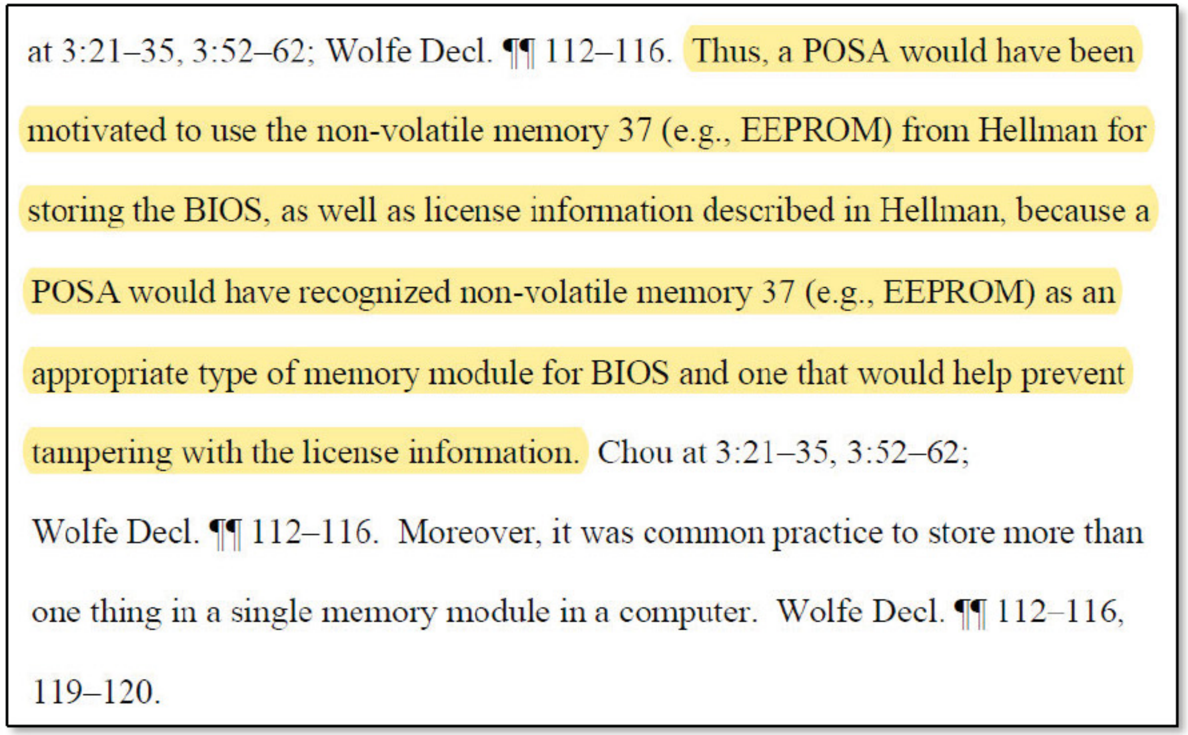
- Pet. 33-34; Ex. 1003 ¶115; Reply 20-21; Ex. 1033 ¶56.

Motivation to Combine: Rationale #2



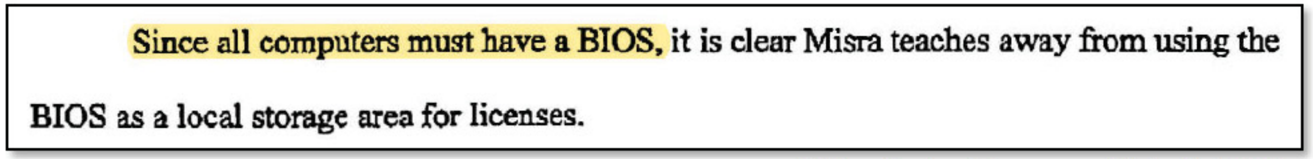
When applying Hellman's disclosure to a computer of the late 1990s, Hellman's disclosure would be applied on a computer with BIOS and a memory storing the BIOS. Wolfe Decl. ¶¶ 105–111. Thus, if a BIOS was not already present, a POSA would have been motivated to add a BIOS stored in a memory of the computer (base unit 12) of Hellman, at least because that was the standard—nearly universal—way in which computers operated prior to the priority date of the '941 patent. *Id.*

Petition at 32.



at 3:21–35, 3:52–62; Wolfe Decl. ¶¶ 112–116. Thus, a POSA would have been motivated to use the non-volatile memory 37 (e.g., EEPROM) from Hellman for storing the BIOS, as well as license information described in Hellman, because a POSA would have recognized non-volatile memory 37 (e.g., EEPROM) as an appropriate type of memory module for BIOS and one that would help prevent tampering with the license information. Chou at 3:21–35, 3:52–62; Wolfe Decl. ¶¶ 112–116. Moreover, it was common practice to store more than one thing in a single memory module in a computer. Wolfe Decl. ¶¶ 112–116, 119–120.

Petition at 32-33.

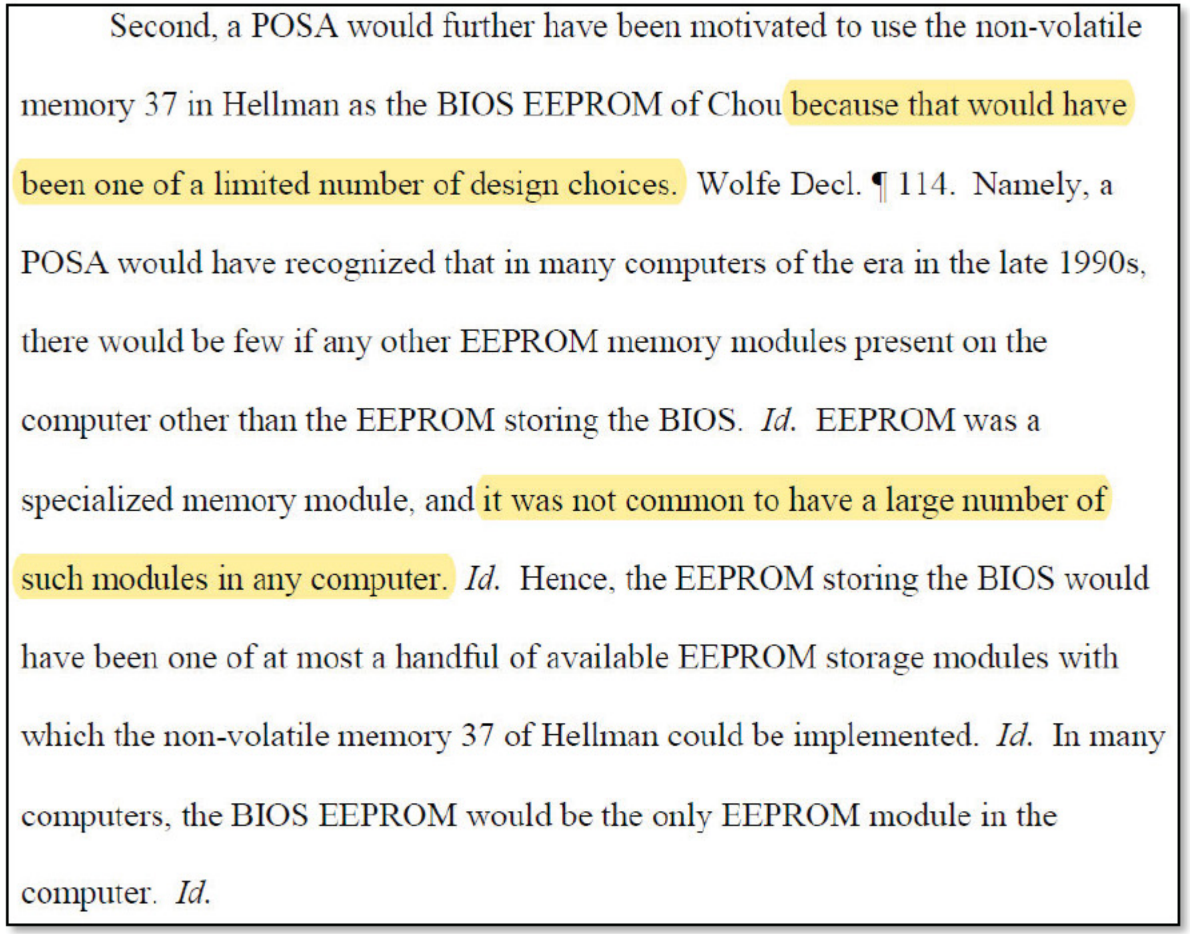


Since all computers must have a BIOS, it is clear Misra teaches away from using the BIOS as a local storage area for licenses.

Office Action Response (Feb. 5, 2002)

- Pet. 31-33; Ex. 1003 ¶107-113;
Reply 20-23; Ex. 1033 ¶49.

Motivation to Combine: Rationale #3

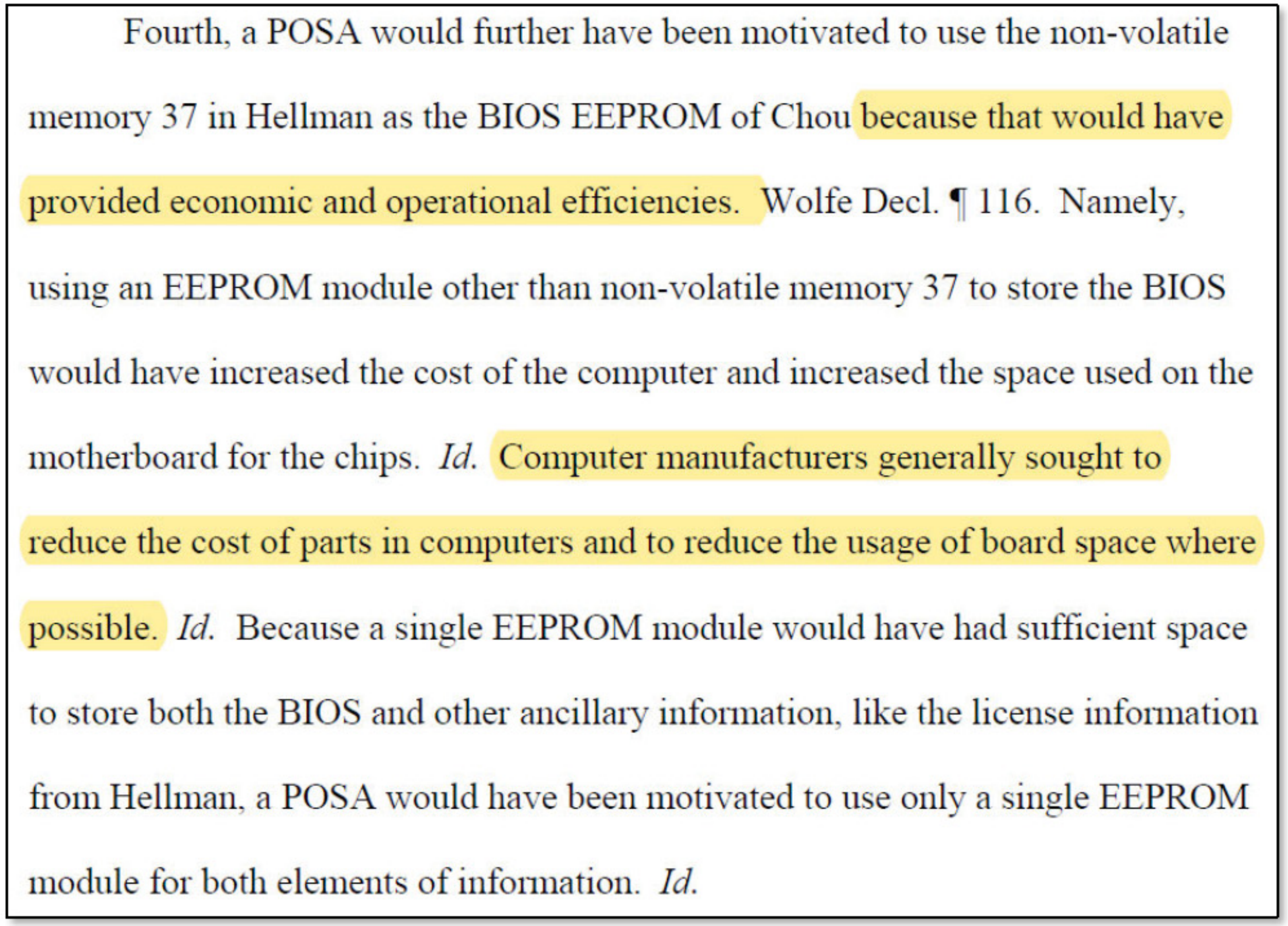


Second, a POSA would further have been motivated to use the non-volatile memory 37 in Hellman as the BIOS EEPROM of Chou because that would have been one of a limited number of design choices. Wolfe Decl. ¶ 114. Namely, a POSA would have recognized that in many computers of the era in the late 1990s, there would be few if any other EEPROM memory modules present on the computer other than the EEPROM storing the BIOS. *Id.* EEPROM was a specialized memory module, and it was not common to have a large number of such modules in any computer. *Id.* Hence, the EEPROM storing the BIOS would have been one of at most a handful of available EEPROM storage modules with which the non-volatile memory 37 of Hellman could be implemented. *Id.* In many computers, the BIOS EEPROM would be the only EEPROM module in the computer. *Id.*

Petition at 33.

- Pet. 33; Ex. 1003 ¶114;
  Reply 21; Ex. 1033 ¶50-55.

Motivation to Combine: Rationale #4

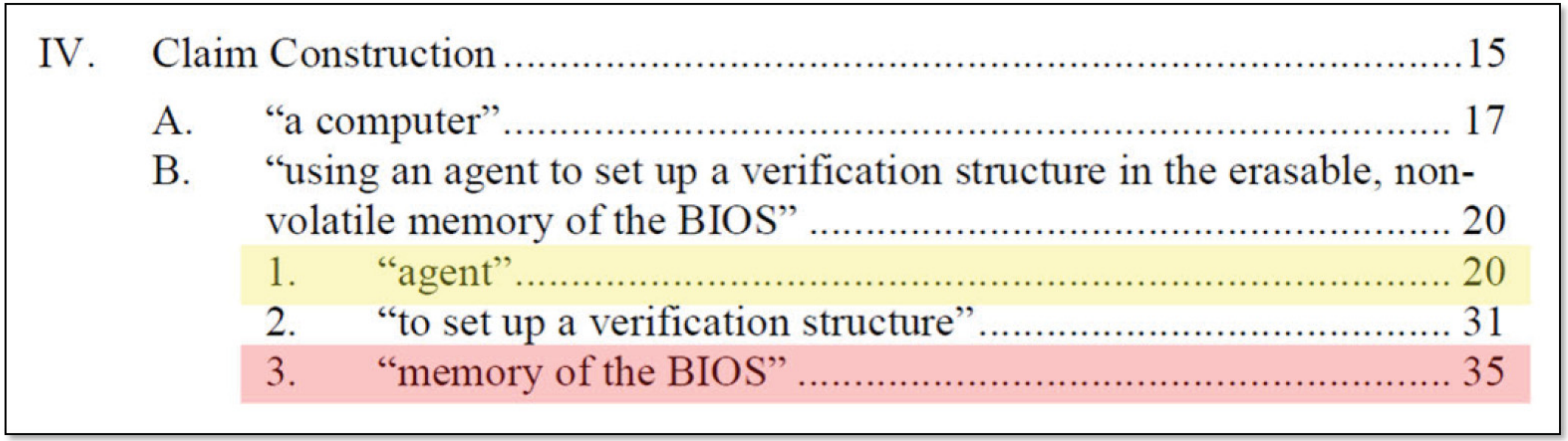


Fourth, a POSA would further have been motivated to use the non-volatile memory 37 in Hellman as the BIOS EEPROM of Chou because that would have provided economic and operational efficiencies. Wolfe Decl. ¶ 116. Namely, using an EEPROM module other than non-volatile memory 37 to store the BIOS would have increased the cost of the computer and increased the space used on the motherboard for the chips. *Id.* Computer manufacturers generally sought to reduce the cost of parts in computers and to reduce the usage of board space where possible. *Id.* Because a single EEPROM module would have had sufficient space to store both the BIOS and other ancillary information, like the license information from Hellman, a POSA would have been motivated to use only a single EEPROM module for both elements of information. *Id.*

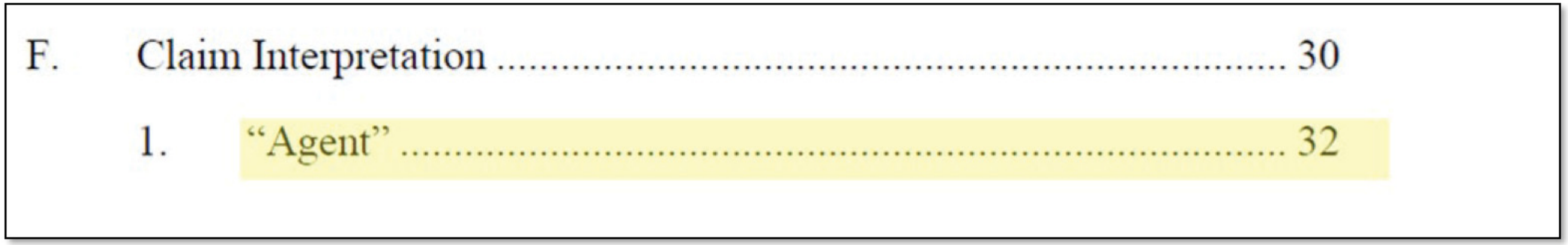
Petition at 34.

- Pet. 34; Ex. 1003 ¶116; Reply 21; Ex. 1033 ¶57.

Motivation to Combine: The License Record Is in Memory of the BIOS



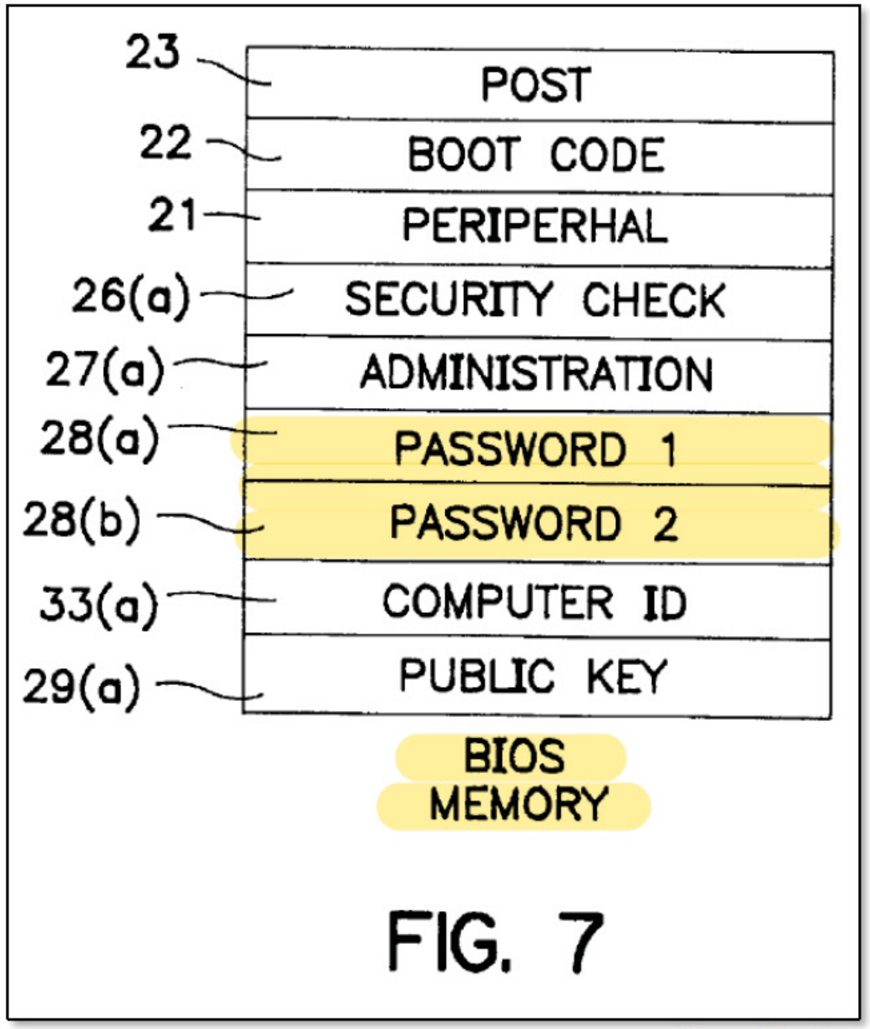
POPR (Table of Contents).



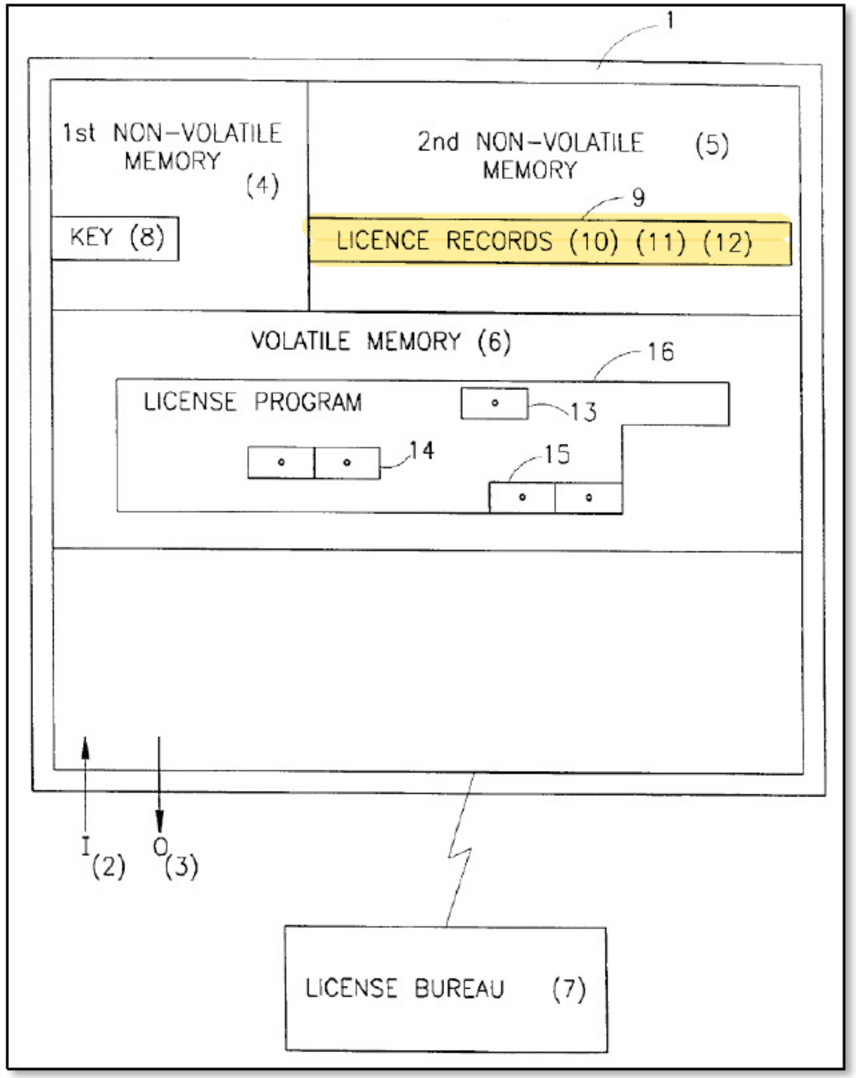
POR (Table of Contents).

- Reply 22-23; Ex. 1033 ¶59-61.

Motivation to Combine: The License Record Is in Memory of the BIOS



Chou, Fig. 7.



’941 Patent, Fig. 1.

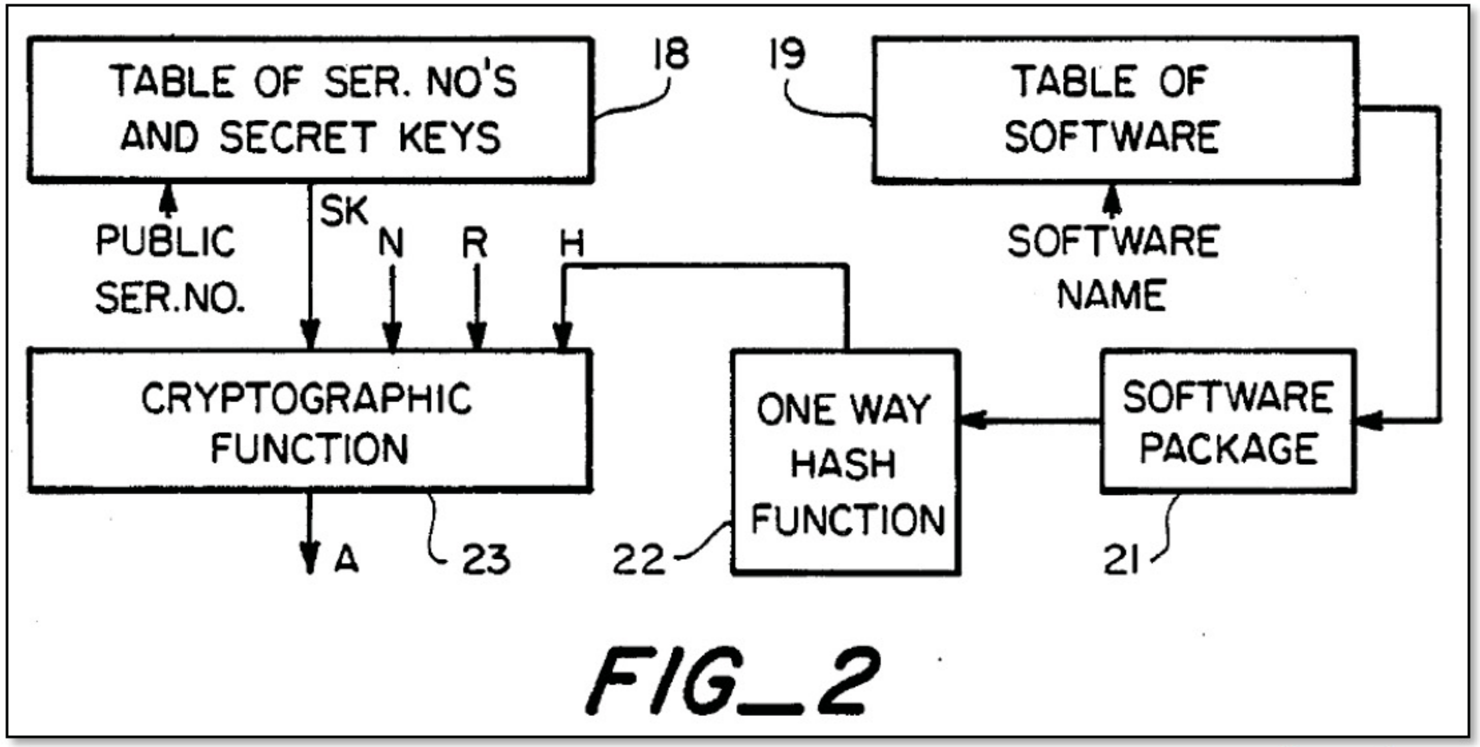
- Pet. 29-35, 40-41; Ex. 1003 ¶105-120F;
  Reply 22-23; Ex. 1033 ¶59-61.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Overview

1. Background
2. The “using an agent” Limitation.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

Dependent Claims: Operable as Described in the Petition

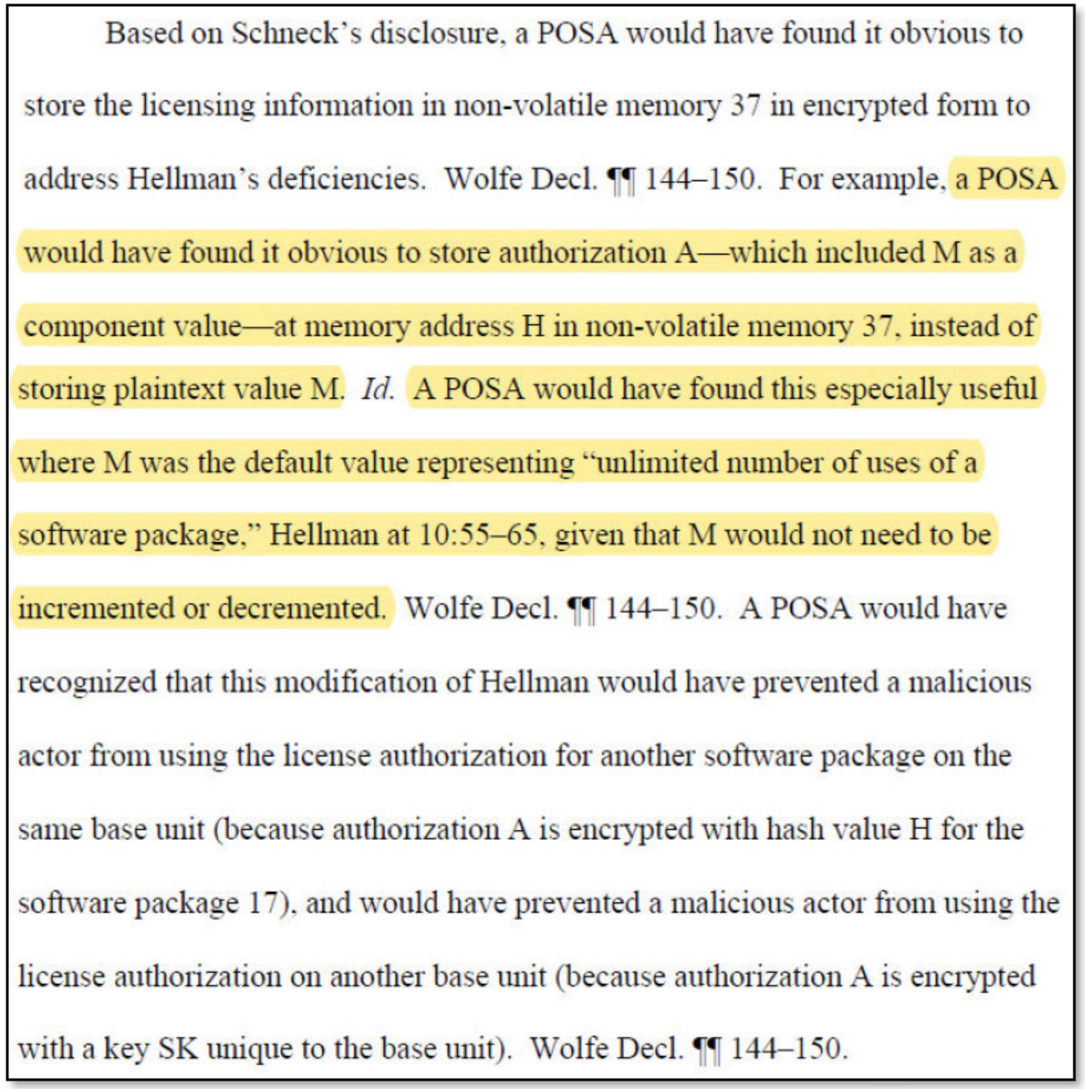


Hellman, Fig. 2.

- Pet. 44-49; Ex. 1003 ¶139-150; Reply 23-24; Ex. 1033 ¶62-64.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Dependent Claims: Operable as Described in the Petition



Based on Schneck's disclosure, a POSA would have found it obvious to store the licensing information in non-volatile memory 37 in encrypted form to address Hellman's deficiencies. Wolfe Decl. ¶¶ 144–150. For example, a POSA would have found it obvious to store authorization A—which included M as a component value—at memory address H in non-volatile memory 37, instead of storing plaintext value M. *Id.* A POSA would have found this especially useful where M was the default value representing "unlimited number of uses of a software package," Hellman at 10:55–65, given that M would not need to be incremented or decremented. Wolfe Decl. ¶¶ 144–150. A POSA would have recognized that this modification of Hellman would have prevented a malicious actor from using the license authorization for another software package on the same base unit (because authorization A is encrypted with hash value H for the software package 17), and would have prevented a malicious actor from using the license authorization on another base unit (because authorization A is encrypted with a key SK unique to the base unit). Wolfe Decl. ¶¶ 144–150.

Petition at 47-48.

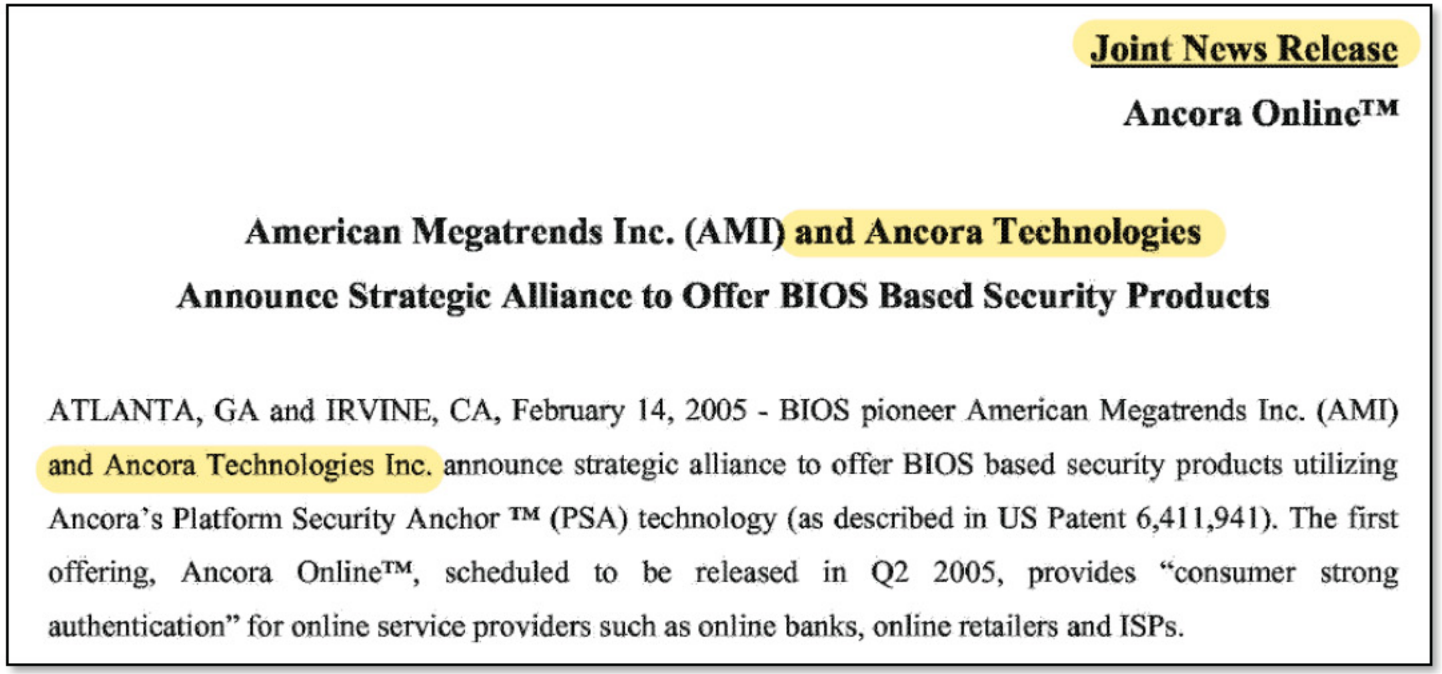
- Pet. 44-49; Ex. 1003 ¶139-150; Reply 23-24; Ex. 1033 ¶62-64.

Overview

1. Background
2. The “using an agent” Limitation.
3. The “verification structure” Limitation.
4. Motivation to Combine Hellman and Chou
5. Dependent Claims
6. Secondary Considerations

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Self-Praise Is Not Industry Praise



Joint News Release

Ancora Online™

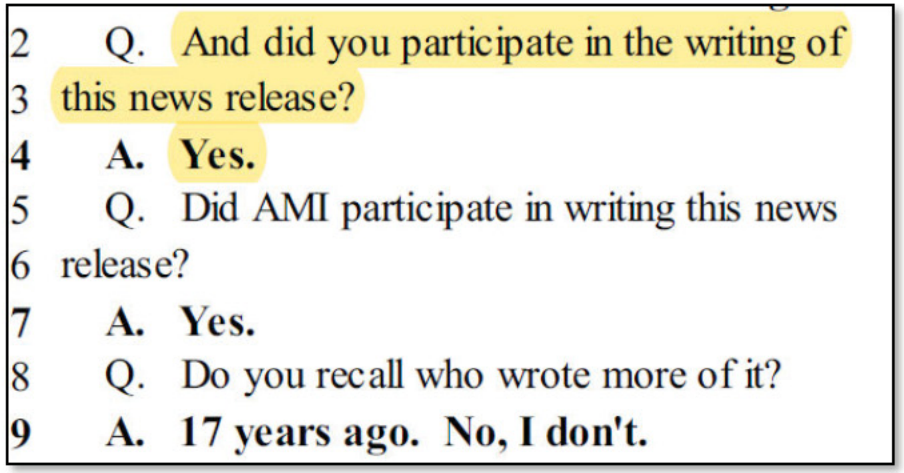
American Megatrends Inc. (AMI) and Ancora Technologies
Announce Strategic Alliance to Offer BIOS Based Security Products

ATLANTA, GA and IRVINE, CA, February 14, 2005 - BIOS pioneer American Megatrends Inc. (AMI) and Ancora Technologies Inc. announce strategic alliance to offer BIOS based security products utilizing Ancora's Platform Security Anchor ™ (PSA) technology (as described in US Patent 6,411,941). The first offering, Ancora Online™, scheduled to be released in Q2 2005, provides "consumer strong authentication" for online service providers such as online banks, online retailers and ISPs.

EX. 2027.

- Reply 24-25.

Self-Praise Is Not Industry Praise



Q. And did you participate in the writing of this news release?

A. Yes.

Q. Did AMI participate in writing this news release?

A. Yes.

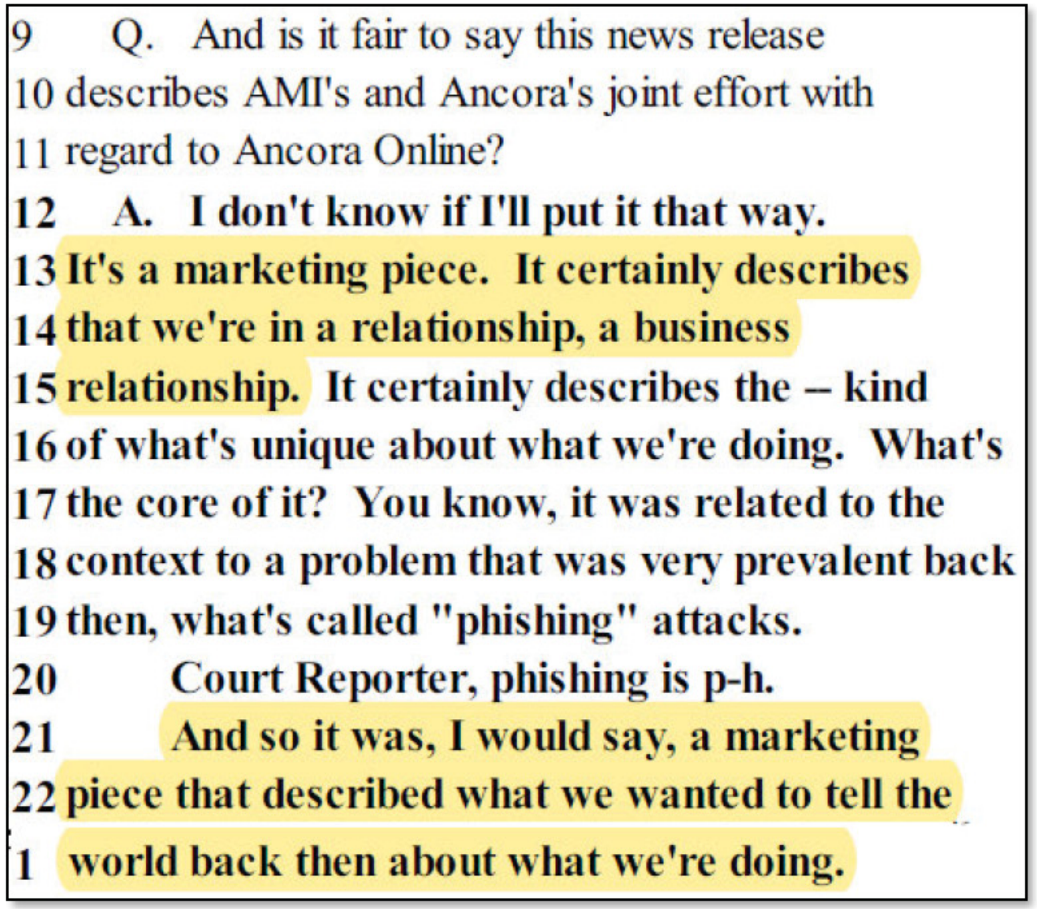
Q. Do you recall who wrote more of it?

A. 17 years ago. No, I don't.

EX. 1034, 49:2-9 (Inventor/Owner, Miki Mullor, Deposition).

- Reply 24-25.

Self-Praise Is Not Industry Praise



Q. And is it fair to say this news release describes AMI's and Ancora's joint effort with regard to Ancora Online?

A. I don't know if I'll put it that way. It's a marketing piece. It certainly describes that we're in a relationship, a business relationship. It certainly describes the -- kind of what's unique about what we're doing. What's the core of it? You know, it was related to the context to a problem that was very prevalent back then, what's called "phishing" attacks.

Court Reporter, phishing is p-h.

And so it was, I would say, a marketing piece that described what we wanted to tell the world back then about what we're doing.

EX. 1034, 48:9-49:1 (Inventor/Owner, Miki Mullor, Deposition).

- Reply 24-25.

Self-Praise Is Not Industry Praise

[redacted]

~~THIRD-PARTY CONFIDENTIAL – PARTY ACCESS LIMITED~~

- Reply 24-25.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Self-Praise Is Not Industry Praise

[redacted]

~~THIRD-PARTY CONFIDENTIAL – PARTY ACCESS LIMITED~~

- Reply 24-25.

Licenses Show Recognition of Obviousness of the Claims

[redacted]

~~THIRD-PARTY CONFIDENTIAL – PARTY ACCESS LIMITED~~

- Reply 25-26.

Licenses Show Recognition of Obviousness of the Claims

[redacted]

- Reply 25-26.

~~THIRD-PARTY CONFIDENTIAL – PARTY ACCESS LIMITED~~

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE