IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
TYLER DIVISION

	§	
UNILOC USA, INC., ET AL.	§	
Plaintiffs,	§	
	§	**CASE NO. 6:10-CV-373**
vs.	§	**PATENT CASE**
	§	
SONY CORPORATION OF AMERICA, ET AL.	§	
	§	
Defendants.		
	§	
UNILOC USA, INC., ET AL.	§	
Plaintiffs,	§	
	§	**CASE NO. 6:10-CV-471**
vs.	§	**PATENT CASE**
	§	
DISK DOCTORS LABS, INC., ET AL.	§	
Defendants.	§	
UNILOC USA, INC., ET AL.	§	
Plaintiffs,	§	
	§	
vs.	§	**CASE NO. 6:10-CV-472**
	§	**PATENT CASE**
NATIONAL INSTRUMENTS CORP., ET AL.	§	
	§	
Defendants.	§	
	§	
UNILOC USA, INC., ET AL.	§	
Plaintiffs,	§	
	§	**CASE NO. 6:10-CV-591**
vs.	§	**PATENT CASE**
	§	
ENGRASP, INC., ET AL.	§	
Defendants.	§	

	§	
UNILOC USA, INC., ET AL.	§	
Plaintiffs,	§	
	§	**CASE NO. 6:10-CV-636**
vs.	§	**PATENT CASE**
	§	
BMC SOFTWARE, INC., ET AL.	§	
Defendants.	§	
	§	
UNILOC USA, INC., ET AL.	§	
Plaintiffs,	§	
	§	**CASE NO. 6:10-CV-691**
vs.	§	**PATENT CASE**
	§	
FOXIT CORPORATION, ET AL.	§	
Defendants.	§	
	§	
SYMANTEC CORPORATION, ET AL.	§	
Plaintiffs,	§	
	§	**CASE NO. 6:11-CV-33**
vs.	§	**PATENT CASE**
	§	
UNILOC USA, INC., ET AL.	§	
Defendants.	§	

PLAINTIFF'S OPENING BRIEF ON CLAIM CONSTRUCTION

TABLE OF CONTENTS

TABLE OF AUTHORITIES

CASES

OTHER

I. INTRODUCTION

Pursuant to the Court's Scheduling Order and P.R. 4-5(a), Plaintiffs Uniloc USA, Inc. and Uniloc Singapore Limited ("Uniloc") hereby submit their opening brief on claim construction for the disputed terms and phrases in United States Patent No. 5,490,216 (the "'216 Patent").

As a preliminary matter, the Court should be aware that since the filing of the P.R. 4-3 Joint Claim Construction and Prehearing Statement, all 41 parties have agreed to the constructions of (1) "'Local' (in the phrase 'local licensee unique ID generating means')" as "on the computer on which the digital data is executing or is to be executed"[1] and (2) "comprises part of said digital data when executed on said platform" as "contained in said digital data when executed on said platform."[2] This leaves only one disputed phrase, "permits use of said digital data . . . only if . . . has matched . . . .,"[3] and Defendants' alleged disclaimer arguments as the subject of this brief.[4]

II. BACKGROUND

A. Overview of the Claimed Invention

The '216 Patent[5] describes a system and method for software activation to help prevent piracy. It does so by generating a local licensee unique ID ("LUID") on the local computer where the software is installed and a remote licensee unique ID ("RUID") at a remote registration authority. The LUID is generated for an intended licensee by inputting certain pieces of information into a local licensee unique ID generating means, which is generally a summer or a summation algorithm. The summer or summation algorithm preferably combines information

[1] Exh. A (partial email string between Steven Hartsell and Patrick Lujin dated September 1, 2011). Unless otherwise noted, all exhibits are attached to the supporting declaration of Steven W. Hartsell ("Hartsell Decl.").
[2] Exh. B (email string between Jamie Olin and Patrick Lujin dated September 7-8, 2011).
[3] Uniloc notes that while it was able to reach agreement with the 40 remaining defendants, a lone defendant, Pervasive Software, Inc. ("Pervasive"), continues to seek leave to re-construe an additional term, namely the previously construed term "licensee unique ID." Pervasive filed its Motion for Leave to Construe Previously Construed Term on August 29, 2011 (Dkt. No. 243 in Case No. 6:10-cv-00472). Uniloc has not addressed this term as the Court has not granted leave as of the filing of this opening brief.
[4] The defendants have broken into two groups, each seeking a different alleged disclaimer. The Group A defendants consist of all defendants except those found in Group B. The Group B defendants consist of Aspyr Media, Inc., Borland Software Corp., Digital River, Inc., GEAR Software, Inc. and GEAR Software Holdings, Inc.
[5] Exh. C (copy of the '216 Patent).

entered by a prospective user that is associated with that user, along with information provided by the environment on which the protected software is to run. These pieces of information are processed through the summer or summation algorithm to produce the LUID, which is a unique identifier associated with the user.

The algorithm that generates the LUID is duplicated at a remotely located registration authority, generally under the control of the licensor. The algorithm on the remotely located registration authority then uses the same pieces of information to generate a RUID. The LUID and RUID are compared and if they match, the system will allow full, unrestricted use of the software.

B. Prior Litigation

The '216 Patent has a long litigation history, including a jury trial and multiple trips to the Federal Circuit. In September 2003, Uniloc sued Microsoft Corporation for infringement in the District Court for the District of Rhode Island. *See Uniloc USA, Inc. v. Microsoft Corp.*, 640 F. Supp. 2d 150, 159 (D.R.I. 2009). The District Court conducted a *Markman* hearing and issued an order in August 2006 construing twenty-three terms and phrases. *Uniloc USA, Inc. v. Microsoft Corp.*, 447 F. Supp. 2d 177 (D.R.I. 2006).

In October 2007, the District Court granted Microsoft's motion for summary judgment of non-infringement. Uniloc appealed, and the Federal Circuit reversed the summary judgment. In reversing, the Federal Circuit considered and rejected "several alternative grounds for affirming the summary judgment beyond those which were reached by the district court" and concluded that they were without merit. The Federal Circuit also affirmed the construction of the term "licensee unique ID," and explained that vendor-supplied information, such as Microsoft's Product Key, could provide the basis for a "licensee unique ID." *Uniloc USA, Inc. v. Microsoft Corp.*, 290 Fed. App'x 337, 343-44 (Fed. Cir. 2008) ("Uniloc I").

After a ten-day trial in April 2009, the jury found the '216 Patent valid and infringed, and

awarded Uniloc $388 million. *Uniloc,* 640 F. Supp. 2d at 160. Notwithstanding the jury verdict, the District Court granted Microsoft's post-trial motions for judgment as a matter of law of non-infringement and for a new trial on damages. *Id*. at 165-76, 183-85. The District Court also determined that: (1) U.S. Patent No. 4,658,093 ("Hellman") did not anticipate the '216 Patent; and (2) the '216 Patent is not obvious in light of Hellman combined with U.S. Patent No. 4,796,220 ("Wolfe"). *Id*. at 180-183. Both Uniloc and Microsoft appealed to the Federal Circuit.

On January 4, 2011, the Federal Circuit reversed the District Court a second time on the issue of infringement, reinstating the jury's finding, but upheld the District Court's grant of a new trial on damages. *Uniloc USA, Inc. v. Microsoft Corp.*, 632 F.3d 1292, 1323 (Fed. Cir. 2011) ("Uniloc II"). Characterizing the District Court's opinion as "comprehensive and well-reasoned," the Federal Circuit also explicitly rejected Microsoft's arguments that the '216 Patent was anticipated by Hellman, noting that in accordance with the Federal Circuit's prior construction, Hellman failed to teach a "licensee unique ID" as claimed in the '216 Patent. *Id*. at 1301.

The mandate of the Federal Circuit issued on May 23, 2011. The final pretrial conference for the new trial on damages is set for December 2011, with the trial expected to occur in early January 2012.

C. *Ex Parte* Reexamination

On January 22, 2010, a reexamination of the '216 Patent was instigated by Microsoft based primarily on United States Patent Nos. 4,658,093 to Hellman ("Hellman") and 5,291,598 to Grundy ("Grundy").[6] The Patent Office granted the reexamination request on April 9, 2010.[7] Ultimately,

[6] The Examiner also cited United States Patent No. 4,796,220 to Wolfe but did not issue any substantive rejections based on Wolfe. See UNI075102 at n. 1. All references to UNIXXXXXX are found in Hartsell Decl. at Exh. D. Additionally, Uniloc also notes that Grundy was considered in the original prosecution and is found on the face of the '216 Patent. See Exh. C. Hellman was considered and rejected by the District of Rhode Island in the Microsoft litigation, *see Uniloc USA, Inc. v. Microsoft Corp.*, 640 F. Supp. 2d 150, 179-183 (D.R.I. 2009), and that decision was affirmed by the Federal Circuit, *see Uniloc II*, 632 F.3d 1292, 1321-1323 (Fed. Cir. 2011). The Patent Office has recently confirmed all of the claims of the '216 Patent over both Hellman and Grundy. See UNI0076148-57.

[7] UNI074468-69.

the Patent Office confirmed all claims of the '216 Patent and issued a Notice of Intent to Issue Reexamination Certificate ("NIIRC") on August 5, 2011.[8]

The primary focus of the Patent Office's reexamination was a 35 U.S.C. § 103(a) obviousness rejection based on Hellman in view of Grundy.[9] Uniloc disagreed with the Examiner's technical interpretations of how the disclosed systems in Hellman and Grundy operated. Uniloc's responses in the reexamination focused on correcting the Examiner's technical misunderstanding of the prior art. Ultimately, Uniloc was successful in explaining the technology of Hellman and Grundy to the Examiner and why these references could not properly be combined under § 103.

Prior to addressing Defendants' constructions/disclaimers, Uniloc presents a summary of the substance of Hellman and Grundy. Having a high-level technical understanding of these two references will allow the Court to appreciate logical deficiencies in the Defendants' positions.

1. Overview of Hellman[10]

Hellman discloses a system that prevents software misuse by limiting the number of times that a base unit, *e.g.*, a personal computer, is authorized to use a software program.[11] To that end, authorizations are made to be specific to individual base units "so that an authorization for one base unit cannot be transferred to another base unit."[12]

An authorization unit at the software manufacturer (*e.g.*, a server) uses a cryptographic function generator 23 to generate an authorization A.[13] An identical cryptographic function generator 38 at the personal computer's cryptographic check unit 34 is used to generate a check

[8] UNI076148-57.
[9] UNI075022-30.
[10] The explanation of Hellman is derived from paragraphs 13-24 of the March 17, 2011 declaration of Dr. Udo Pooch that was submitted in connection with the reexamination. See UNI075642-720. The Patent Office stated that Dr. Pooch's declaration was "sufficient to overcome the rejection of claims 1-20 based upon Dr. Pooch's argument that it would be improper to combine the references in the manner of the claimed invention." UNI076156 (Notice of Intent to Issue Reexamination Certificate).
[11] Exh. E at Abstract (Hellman patent); see also UNI075644 at ¶ 14.
[12] Id.; see also UNI075644 at ¶ 14.
[13] Id. at Figs. 1 and 2; see also UNI075644 at ¶ 15.

value C that must match the authorization A at comparator 39.[14] If the check value C does not match Authorization A, A is not considered to be a proper authorization.[15]

The cryptographic function generators 23 or 38 are an essential and critical element of Hellman's system. Hellman's cryptographic function generators are designed to be highly secure and to prevent would-be copiers from generating their own authorizations.[16] Hellman describes three possible types of cryptographic functions that can serve as function generator 23 or 38, and unequivocally states that one of these cryptographic functions is "*required to carry out the present invention.*"[17]

The first cryptographic function, described in FIG. 4 of Hellman, uses a modified Data Encryption Standard ("DES").[18] According to Hellman, "[t]he DES would have to be modified…to have its key length equal to the length of SK" so that it would work in the cryptographic function generator.[19] Hellman teaches that this embodiment "would also inherently have the property that the new authorizations could not be predicted from old authorizations because, in a conventional cryptographic system, given past plaintext-ciphertext pairs, it must be difficult to determine the plaintext which goes with a new ciphertext."[20] The DES embodiment thus uses a robust and highly secure, cryptographic algorithm to generate an authorization code sufficiently secure to meet the rigorous needs of Hellman's system. At the time of Hellman, DES was considered a "military-grade" encryption technique.

The second cryptographic function is shown in FIG. 9 and described as follows: "[s]ignals representing H, R and N are presented as a message to be signed by a public key cryptosystem 43

[14] Id. at Figs. 6 and 7; see also UNI075644 at ¶ 15.
[15] Id at Col. 10:24-26; see also UNI075644 at ¶ 15.
[16] Id. at Col. 7:4-16; UNI075644-45 at ¶ 16.
[17] Id. at Col. 2:61-65 (emphasis added); UNI075644-45 at ¶ 16.
[18] Id. at Col. 8:23-26; UNI075645 at ¶ 17.
[19] Id. at Col. 8:23-26; UNI075645 at ¶ 17.
[20] Id. at Col. 8:46-51; UNI075645 at ¶ 17.

using secret key SK to produce the digital signature," which becomes authorization A.[21] Like the DES embodiment, a public key cryptosystem is a highly secure, robust, cryptographic algorithm that meets the rigorous needs of Hellman's system. It is still in wide use today.

The third cryptographic function is a one-way hash function. Hellman discloses that "[o]ne implementation of the cryptographic function generator 23 of FIG. 2 would also involve a one-way hash function . . . ."[22] Again, a cryptographic one-way hash function is a robust, highly secure, cryptographic algorithm that meets the rigorous needs of Hellman's system. "One-Way" means that it is impossible to recreate the input of the hash function using the output of the function.

Each of the alternative embodiments that Hellman describes to implement its cryptographic function generators 38 (*i.e.*, DES, public/private key encryption, or cryptographic one-way hash algorithms) are sophisticated cryptographic algorithms. As noted earlier, Hellman unambiguously states that such cryptographic functions are "*required to carry out the present invention.*"[23]

2. Overview of Grundy[24]

Grundy is directed to a method and system for decentralized manufacture of copy-controlled software. As part of Grundy's system, Grundy describes the generation of a "registration code" that is used for communication between a new software user and what Grundy describes as a Manufacturing Control Agency.[25]

As it relates to the '216 Patent reexamination, Grundy's use of "checksums" is of interest. A checksum is a value that (a) is computed by a function that is dependent on the contents of a data object and (b) is stored or transmitted together with the object, for detecting changes in data. A

[21] Id. at Col. 11:42-47; see also UNI075646 at ¶ 19.
[22] Id. at Col. 8:13-15; see also UNI075646 at ¶ 22.
[23] Id. at Col. 2:61-65 (emphasis added); see also UNI075647 at ¶ 24.
[24] The explanation of Grundy is derived from paragraphs 25-39 of the March 17, 2011 declaration of Dr. Udo Pooch that was submitted in connection with the reexamination. See UNI075642-720. The Patent Office stated that Dr. Pooch's declaration was "sufficient to overcome the rejection of claims 1-20 based upon Dr. Pooch's argument that it would be improper to combine the references in the manner of the claimed invention." UNI076156 (Notice of Intent to Issue Reexamination Certificate).
[25] Exh. F at Col. 14:31-60 (Grundy patent); see also UNI075647 at ¶ 25.

checksum algorithm is a signature algorithm that does not attempt to provide cryptographic protection against inversion. The term "checksum" originally referred to checking algorithms that summed the bytes, but is now generally used to refer to any non-cryptographic checking algorithm.[26] This is consistent with a contemporaneous definition of "checksum" from the filing of the '216 Patent and Grundy, which defines a "checksum" as:

> [A] calculated value that is used to test data integrity. Errors can occur when data is transmitted or when it is written to disk. One means of detecting such errors is use of a checksum, a value calculated for a given chunk of data by sequentially combining all the bytes of data with a series of arithmetic or logical operations. After the data is transmitted or stored, a new checksum can be calculated (using the possibly faulty transmitted or stored data) and compared with the original one. If the checksums don't match, an error occurred, and the data should be transmitted or stored again; if they do match, the transmission or storage was probably error-free. Checksums are a simple validation mechanism, and they cannot be used to correct erroneous data.[27]

For example, prior to transmitting a 1000 character set of data, the sending computer may add each character together to form a checksum. The checksum may be only four characters long (undoubtedly, the true sum of a 1000 character data set would result in a number greater than four characters, but checksums typically only make use of a few digits of the total sum). The sending computer would then send the 1000 character set along with the four-character checksum to a receiving computer. The receiving computer would then use the same checksum algorithm used by the sending computer to sum up the 1000 character set and recreate the four-character checksum. If the checksum computed by the receiving computer matched the checksum it received from the transmitting computer, the receiving computer would assume that there were no errors in the transmission. If, however, the two checksum did not match, the receiving computer would conclude that the 1000 character set of data had not been transmitted accurately and would typically

[26] UNI075650 at ¶ 36.
[27] UNI075650 at ¶ 37; UNI075720.

request the transmitting computer to resend the data. Almost all Internet communications involve the exchange of checksums to help ensure data accuracy.

Of importance to the reexamination, Grundy describes the use of checksum algorithms in the generation of certain data fields that ultimately make up the "registration code." Specifically, Grundy teaches that the registration code consists of a packed bit array "with each field of the record occupying no more bits than is necessary to encode the information content of the field."[28] There are six fields concatenated to form the registration code bit array.

The first four fields are (1) a user data checksum, (2) a hardware ID code, (3) an anti-virus checksum, and (4) a previous owner's ID number. After these four fields are packed into the bit array, the array is encrypted, then (5) a Product/Version ID code is added, then (6) a data entry checksum is generated for the entire array and added to the array. Finally, the packed bit array is converted to an alphanumeric form to produce the registration code.[29] The block diagram in Figure 1 below represents how Grundy generates its registration code, which is a packed bit array:

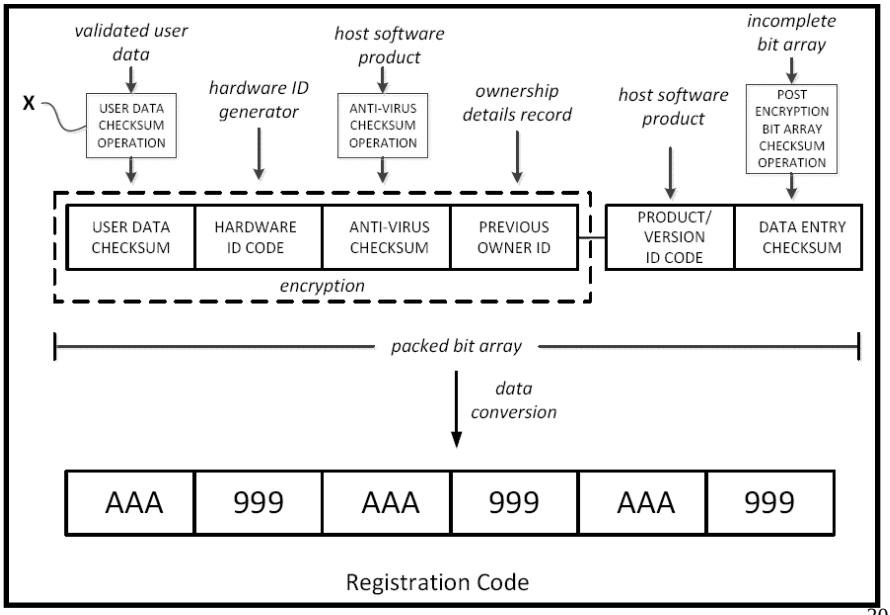

Figure 1: Block diagram of Grundy's concatenated registration code[30]

[28] Id. at Col. 18:61-64; see also UNI075647-48 at ¶ 27.
[29] Id. at Col. 18:58-64, 19:13-18; see also UNI075648 at ¶ 29.
[30] UNI075648.

As illustrated in the block labeled "X" in the upper-left hand corner of the above Figure, the operation that comprises the checksum of the user data component fields is *a preliminary step* for generating only the data that will later occupy the USER DATA CHECKSUM field of the packed bit array.

During the reexamination, the Examiner erroneously asserted that Grundy produces a registration code by "performing a checksum of the user data component fields."[31] However, the Examiner's technical understanding of Grundy was flawed because, as discussed below, the checksum function does not create the registration code. Rather, the checksum function is a preliminary step to the creation of Grundy's "registration code." Further, the checksums described in Grundy are not cryptographic functions, but rather appear to be used to check for typographical data entry errors or transmission errors.

While the typical checksums are useful in detecting accidental modification such as corruption of stored data or errors in a communication channel, they provide no security against a malicious agent because their simple mathematical structure makes them trivial to circumvent. To provide a reliable level of security, the use of a cryptographic hash function is necessary.

In one example, Grundy states for one of the checksums that "[t]his checksum will be used by the Manufacture Control Agency to avoid operator data-entry errors 304."[32] For another checksum, Grundy states that "[i]f the user data validity check 502 passes, a checksum of the user data is created 505. This checksum will be used during the Authorization process as a cross-reference to validate the user data as communicated to the Manufacture Control Agency 310 FIG. 3."[33]

[31] UNI075172.
[32] Exh. F at Col. 19:6-8, 15:3-24; see also UNI075649 at ¶ 34.
[33] Id. at Col. 18:25-29; see also UNI075649 at ¶ 34.

Grundy's checksums are quite distinct from cryptographic functions, which are used for different applications. If Grundy's checksums were to be used for security applications (as the Examiner asserted during the reexamination) they would be vulnerable to attack. For example, a checksum's linearity may be exploited by a malicious adversary.

One of the Examiner's primary misconceptions in the reexamination was the role of checksum functions in Grundy. The Examiner mistakenly believed that the checksum in Grundy was used to generate the Grundy "registration code." Grundy however, uses a checksum function for data entry error checking as a *preliminary step* to creating Grundy's "registration code." Importantly, the checksum function *does not* produce the Grundy "registration code." Rather, the checksum produces a value that is ultimately combined along with several other data items, encrypted and packed into a bit array, which is Grundy's "registration code."

Operating under this misconception, the Examiner substituted the checksum algorithm disclosed in Grundy in place of the sophisticated encryption algorithms disclosed in Hellman to arrive at a combination that the Examiner initially believed rendered obvious the claims of the '216 Patent. But, after Uniloc educated the Examiner on the nature of Grundy's checksum and how/why it was used, the Patent Office reversed itself and concluded that such a combination did not render obvious any of the claims of the '216 Patent.

III. APPLICABLE LAW

A. Claim Construction

"It is a 'bedrock principle' of patent law that 'the claims of a patent define the invention to which the patentee is entitled the right to exclude.'" *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (quoting *Innova/Pure Water Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)). In claim construction, courts examine the patent's intrinsic evidence to define the patented invention's scope. *See id.*; *C.R. Bard, Inc. v. U.S. Surgical Corp.*,

388 F.3d 858, 861 (Fed. Cir. 2004); *Bell Atl. Network Servs., Inc. v. Covad Commc'ns Group, Inc.*, 262 F.3d 1258, 1267 (Fed. Cir. 2001). This intrinsic evidence includes the claims themselves, the specification, and the prosecution history. *See Phillips*, 415 F.3d at 1314; *C.R. Bard, Inc.*, 388 F.3d at 861. Courts give claim terms their ordinary and accustomed meaning as understood by one of ordinary skill in the art at the time of the invention in the context of the entire patent. *Phillips*, 415 F.3d at 1312-13; *Alloc, Inc. v. Int'l Trade Comm'n*, 342 F.3d 1361, 1368 (Fed. Cir. 2003).

The claims themselves provide substantial guidance in determining the meaning of particular claim terms. *Phillips*, 415 F.3d at 1314. First, a term's context in the asserted claim can be very instructive. *Id*. Other asserted or unasserted claims can also aid in determining the claim's meaning because claim terms are typically used consistently throughout the patent. *Id*. Differences among the claim terms can also assist in understanding a term's meaning. *Id*. For example, when a dependent claim adds a limitation to an independent claim, it is presumed that the independent claim does not include the limitation. *Id*. at 1314-15.

"[C]laims 'must be read in view of the specification, of which they are a part.'" *Id*. (quoting *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 979 (Fed. Cir. 1995) (en banc)). "[T]he specification is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term." *Id*. (quoting *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996)); *Teleflex, Inc. v. Ficosa N. Am. Corp.*, 299 F.3d 1313, 1325 (Fed. Cir. 2002). This is true because a patentee may define his own terms, give a claim term a different meaning than the term would otherwise possess, or disclaim or disavow the claim scope. *Phillips*, 415 F.3d at 1316. In these situations, the inventor's lexicography governs. *Id*. Also, the specification may resolve ambiguous claim terms "where the ordinary and accustomed meaning of the words used in the claims lack sufficient clarity to permit the scope of the claim to be ascertained from the words alone." *Teleflex, Inc.*, 299 F.3d at 1325. But, "[a]lthough the

specification may aid the court in interpreting the meaning of disputed claim language, particular embodiments and examples appearing in the specification will not generally be read into the claims." *Comark Communs. v. Harris Corp.*, 156 F.3d 1182, 1187 (Fed. Cir. 1998) (quoting *Constant v. Advanced Micro-Devices, Inc.*, 848 F.2d 1560, 1571 (Fed. Cir. 1988)); *see also Phillips*, 415 F.3d at 1323. The prosecution history is another tool to supply the proper context for claim construction because a patent applicant may define a term in prosecuting the patent. *Home Diagnostics, Inc., v. Lifescan, Inc.*, 381 F.3d 1352, 1356 (Fed. Cir. 2004) ("As in the case of the specification, a patent applicant may define a term in prosecuting a patent.").

B. Doctrine of Prosecution History Disclaimer

The doctrine of prosecution history disclaimer "limits the interpretation of claims so as to exclude any interpretation that may have been disclaimed or disavowed during prosecution in order to obtain claim allowance." *Omega Eng'g, Inc. v. Raytek Corp.,* 334 F.3d 1314, 1323 (Fed. Cir. 2003). For the doctrine to apply, the disclaimer must be clear and unmistakable. *Computer Docking Station Corp. v. Dell, Inc.*, 519 F.3d 1366, 1374 (Fed. Cir. 2008). Indeed, the Federal Circuit has stated, "we will find that the applicant disclaimed protection during prosecution only if the allegedly disclaiming statements constitute a clear and unmistakable surrender of subject matter." *Ecolab, Inc. v. FMC Corp.*, 569 F.3d 1335, 1342 (Fed. Cir. 2009) (quoting *Bayer AG v. Elan Pharm. Research Corp*., 212 F.3d 1241, 1251 (Fed. Cir. 2000)).

In other words, the doctrine is limited in that "prosecution disclaimer does not apply to an ambiguous disavowal." *Computer Docking Station,* 519 F.3d at 1375; *Cordis Corp. v. Boston Sci. Corp.*, 561 F.3d 1319, 1329 (Fed. Cir. 2009) ("A disclaimer must be 'clear and unmistakable,' and unclear prosecution history cannot be used to limit claims.") "Prosecution disclaimer does not apply, for example, if the applicant simply describes features of the prior art and does not

distinguish the claimed invention based on those features." *Computer Docking Station,* 519 F.3d at 1375.

IV. THE PROPER CONSTRUCTION OF "PERMITS USE OF SAID DIGITAL DATA...ONLY IF...HAS MATCHED . . ."

Uniloc's Proposed Construction	Defendants' Proposed Construction
The ordinary meaning of the phrase is clear and unambiguous. Thus, the phrase does not require construction.	When . . . has matched then the use of said digital data is permitted

This disputed phrase is found in claim 1 of the '216 Patent where the relevant clause reads as follows:

> said system further including [mode switching means] operable on said platform which permits use of said digital data in said [use mode] on said platform only if a [licensee unique ID] first generated by said [local licensee unique ID generating means] [has matched] a [licensee unique ID] subsequently generated by said [remote licensee unique ID generating means];[34]

No construction of this phrase is necessary because it consists of basic English words that enjoy frequent use and therefore will be easily understood and applied by a jury. *Phillips*, 415 F.3d at 1314 (Fed. Cir. 2005) ("In some cases, the ordinary meaning of claim language as understood by a person of skill in the art may be readily apparent even to lay judges, and claim construction in such cases involves little more than the application of the widely accepted meaning of commonly understood words."); *see also Orion IP, LLC v. Staples, Inc.*, 406 F. Supp. 2d 717, 738 (E.D. Tex. 2005) (stating that "although every word used in a claim has meaning, not every word requires construction" in declining to construe claim terms).

Defendants' proposed construction is unnecessary because there is no ambiguity in the disputed phrase that requires clarification, nor are there terms of art that need explanation. This observation is confirmed by examining Defendants' proposed construction, which appears to do

[34] Exh. C at Col. 13:54-14:1. Items in brackets denote claim terms/phrases that have been previously construed and are found in the Agreed Constructions section of the parties' P.R. 4-3 Joint Claim Construction and Prehearing statement.

nothing more than rearrange the words of the disputed phrase. *O2 Micro Int'l Ltd. v. Beyond Innovation Tech. Co.*, 521 F.3d 1351, 1362 (Fed. Cir. 2008) ("We, however, recognize that district courts are not (and should not be) required to construe every limitation present in a patent's asserted claims . . . . Claim construction is not an obligatory exercise in redundancy." (internal quotations and citations omitted)). The disputed phrase should be accorded its plain and ordinary meaning, which is best recognized by the express words of the claim term.

V. UNILOC'S EXPLANATION OF THE PRIOR ART DURING THE REEXAMINATION DOES NOT GIVE RISE TO ANY DISCLAIMER

As explained in the introduction, there are two competing arguments advanced by two different groups of defendants, both of which require the Court to find that Uniloc unambiguously disavowed claim scope during reexamination. Uniloc's and Defendants' competing positions are set forth in the table below.

Uniloc's Position	Group A Defendants' Position	Group B Defendants' Position
There is no disclaimer in the reexamination file history.	The licensee unique ID/security key cannot be generated by a checksum, summation algorithm, summer, or equivalent thereof, used to test data integrity	The licensee unique ID generated by the means recited in each of the claims must be derived from at least one piece of information that is specific to the user, such as name, billing information, or product information unique to the installation entered by the user. The information cannot be specific to the computer or independently generated by the computer.

The Group A Defendants' would erroneously narrow the term licensee unique ID generating means whereas the Group B Defendants' proposed disclaimer erroneously limits the information which can be used to generate a licensee unique ID. As explained below, neither position can be sustained. Because Uniloc did not disclaim any subject matter during reexamination, there should be no manipulation or limitation of the established constructions - constructions that the Patent Office and Uniloc relied upon during the reexamination.

A. A Review of the Reexamination History Reveals That There is No Support for an Exclusion Related to Data Integrity Testing

Not having seen Defendants' arguments, Uniloc is now forced to prove a negative, *i.e.*, that there is no disclaimer. But, based on the excerpts listed in the Group A Defendants' portion of the P.R. 4-3 Statement, it appears Defendants are attempting to select a number of unrelated reexamination statements, from different sources, and then string them together out of context in an attempt to manufacture a disclaimer argument (yet, even these out-of-context statements do not rise to the level of a disclaimer). A close review of the portions of the reexamination history cited by Defendants, however, demonstrates nothing more than Uniloc's efforts to correct the Examiner's technical misinterpretation of the prior art. Namely, the Examiner's attempts to incorrectly combine the Hellman and Grundy references by substituting the simple typo checking checksum disclosed in Grundy for the more sophisticated cryptographic functions used in Hellman's system, as discussed above in sections II(C)(1)-(2).

Based on the portions of the reexamination cited by Defendants in the P.R. 4-3 Statement as allegedly supporting their "checksum" and "used to test data integrity" exclusions, Defendants appear to be laboring under the same flawed technical understanding of Grundy that the Examiner ultimately abandoned. Uniloc did not distinguish its invention, nor amend any of the original claims, to overcome Grundy's checksums. Rather, Uniloc merely demonstrated to the Examiner that he was operating under a flawed understanding regarding Grundy's disclosure and that under a correct understanding, Grundy and Hellman could not be combined in the manner suggested. The Examiner then correctly withdrew the rejections based on the combination of Hellman and Grundy.

The Court should recognize that Uniloc's arguments were not an attempt to distinguish the claimed invention, but instead were a demonstration of the technical infeasibility of combining Hellman and Grundy. Such an explanation does not amount to a disavowal of claim scope under *Computer Docking Station*, 519 F.3d at 1375. The Court should thus reject the Defendants attempts

to narrow the scope of the claims based on Uniloc arguments related to combining Hellman and Grundy.

1. The statements in response to the first office action do not constitute a disclaimer

None of the statements made in Uniloc's reply to the first office action constitute a clear, unambiguous disavowal of claim scope. Instead, these statements were made to show why it was technically unworkable (and thus legally improper) for the Examiner to combine Hellman and Grundy under § 103.

Uniloc pointed out to the Examiner that Grundy's checksum could not be combined with Hellman to somehow cure Hellman's lack of a licensee unique ID because Grundy's checksum was not being used to generate a licensee unique ID. For example, in two statements cited to by Defendants, Uniloc merely points out to the Examiner that Grundy could not properly be combined under § 103 because Grundy's checksum was used to verify that user entered information is entered correctly. In other words, the checksum in the case of Grundy is used as a typographical error checker and not to generate a unique identifier associated with the licensee.

> Statement #1: "Grundy does not cure this deficiency of Hellman. The Office argues that the unique identifier associated with the licensee is disclosed by Grundy's 'checksum.' But *Grundy's checksum is solely used to verify the accuracy of user-entered information - it is not a unique identifier associated with a licensee.*"[35]
>
> Statement #2: "Thus, Grundy's 'checksum' is not uniquely associated with an intended licensee. Rather, *Grundy's checksum can only be used to indicate whether the user (i.e., the intended licensee) correctly entered the requested data.*"[36]

Contrary to the Defendants' apparent position, the above statements do not demonstrate that Uniloc disclaimed the use of checksums or algorithms used to test data integrity, but rather that Grundy was using its checksum for a fundamentally different function - as a typo checker, and not to generate a licensee unique ID. If fact, the above statements are found at the beginning and

[35] UNI075103 (emphasis added).
[36] UNI075103-4 (emphasis added).

ending, respectively, of one paragraph and thus are cited out of context. The rest of the paragraph clearly shows that Uniloc was attempting to explain to the Examiner that the checksum in Grundy was not being used to generate a licensee unique ID and provides:

> More specifically, Grundy describes a "method and apparatus that monitors and controls the use of information stored on a storage medium." (Grundy, 4:21-24.) As part of that method, Grundy teaches generating a checksum of the user data upon entry of the user data, and then packing and encoding the checksummed user data along with other data (i.e., hardware ID, anti-virus checksum, and previous owner ID number) to generate a "registration code." (*See*, Grundy 18:10-33 and 18:58-64.) Grundy discloses that the "registration code is decrypted and then unpacked into its component fields 308." (Grundy, 15:4-6.) Grundy discloses that once the data is unpacked the "user data cross-reference code and the second checksum 309 are compared 310," and "[i]f these do not match it is an indication that the User Data as entered by the Manufacturer Control Agency operator 301 does not match the User Details as originally entered by the new user."[37]

As explained in the above paragraph, Grundy discloses generating a "registration code" which has a one of its parts, a checksum of the user's information. The checksum, however, is not being used to generate the registration code. When the two statements relied upon by the Defendants are viewed in context, it is clear that Uniloc was explaining to the Examiner that the Grundy checksums were being used for error checking and not generating of a licensee unique ID. Additional statements cited by Defendants further show that Uniloc was attempting to explain that Grundy's checksum was not being used to generate a licensee unique ID.

> Statement #3: "However, Grundy's checksum cannot meet these limitations as it cannot be equated to claim l's 'licensee unique ID.' As explained more fully below, *Grundy's checksum is used for nothing more than verifying that the licensee correctly entered data*. It is not uniquely associated with any intended licensee and cannot be used to identify any intended licensee."[38]

> Statement #4: "*Thus, Grundy uses the checksum of the user data as an indicator that the user data has been correctly entered*. Grundy does not teach or suggest that the checksum, or the registration code that includes the checksum as one of the fields, represents a unique Identifier associated with intended registered user."[39]

[37] UNI075103.
[38] UNI075111 (emphasis added).
[39] UNI075112 (emphasis added).

Even though the above statements are again taken out of context, by themselves they show that Uniloc was not disclaiming the use of checksums. This fact is even clearer when viewed in light of the surrounding materials where Uniloc specifically told the Examiner:

> Furthermore, the registration code is not 'produced by performing a checksum of the user data component fields,' as alleged by the Office Action. Rather, the registration code is produced by encrypting multiple fields, of which one of those fields is a checksum of user data, none of which produces a unique value. *The Office's interpretation of Grundy is thus incorrect*.[40]

When read in context, it is evident that Uniloc is explaining to the Examiner that Grundy's checksum does not generate a licensee unique ID because that is not the purpose for which it is being used. Indeed, it is clear that this was the argument Uniloc made because the Examiner explained that he was confirming all claims of the '216 Patent over his initial rejection of them because:

> [Uniloc] has persuasively argued that the summation [*i.e.*, the checksum] disclosed by Grundy is used in the context of merely verifying the correctness of information related to the user and is not being used to generate an ID per se. Since the information is not being used for the same purpose, one skilled in the art therefore would not use the algorithm of Grundy as part of the generation of the claimed licensee unique ID.[41]

Uniloc did not, however, disclaim the use of checksums or algorithms used to test data integrity to generate licensee unique IDs.

The remaining statements from the first office cited by Defendants likewise do not show any disclaimer by Uniloc. These statements include the following:

> Statement #5: "A person of ordinary skill in the art would understand 'checksum' to represent a small number of check digits that are typically appended to data in order to ensure the data's integrity when it is stored or transmitted. To calculate a checksum of some data, the data is added up (e.g., broken up into C-byte chunks, where C is a small number such as 1,2,4, or 8, and summed); the sum is chopped to a fixed length (e.g., a byte or C bytes) and appended to the data before storage or transmission. Checksum algorithms used in practice are variations on this scheme. When the data is received or retrieved later, the checksum is re-calculated to ensure

[40] UNI075111 (emphasis added).
[41] UNI076155.

that the result is the same as the original checksum; if the result differs then the data must have been corrupted. (*See,* Rosenblatt Dec., ¶52.)"[42]

Statement #6: "A checksum is therefore much smaller in length than its input data. For example, a 16-bit (2-byte) or 64-bit (8-byte) checksum may be calculated on thousands, millions, or billions of bytes of data. This fulfills the checksum's intended purpose well, given that most errors in data storage or transmission are small and localized, making it highly likely that the resulting checksum will differ from the one originally calculated, and extremely unlikely that corrupted data will produce the same checksum as the original one. For example, if one or two bits are altered, the checksum will differ. (*See,* Rosenblatt Dec., ¶56.)"[43]

Statement #7: "Therefore, a checksum cannot preserve the uniqueness of the input data. Grundy shows the input data to the checksum routine in Fig. 2, 212, 'ENTER NEW USER DETAILS.' This is 'new user data, such as the user's name, address and telephone number' (Grundy at 12:37-38.) Such data might take up roughly a hundred bytes of data. A checksum of this data would not preserve its uniqueness; many different sets of user data could produce the same checksum. Therefore the checksum is not a generator of unique identifiers. (*See,* Rosenblatt Dec., ¶62.)"[44]

Statement #8: "As previously discussed, a checksum cannot preserve the uniqueness of the input data and thus the checksum is not a generator of unique identifiers. (See, Rosenblatt Dec., 62.)"[45]

As an initial matter, each of the above statements was based on statements made in connection with the November 23, 2010 declaration of William Rosenblatt submitted by Uniloc.[46] However, the Patent Office explicitly stated that it did not rely on Mr. Rosenblatt's declaration[47]:

> Application/Control Number: 90/010,831 Page 5
> Art Unit: 3992
>
> The affidavit under 37 CFR 1.132 filed 29 November 2010 by William R. Rosenblatt has been considered and entered into the record. The affidavit is unpersuasive because it entirely consists of opinions regarding the applicability of the cited prior art.

[42] UNI075112; see also UNI76176 at ¶ 52.
[43] UNI075112; see also UNI76177 at ¶ 56.
[44] UNI075112; see also UNI76180 at ¶ 62.
[45] UNI075117; see also UNI76180 at ¶ 62.
[46] UNI076163-86.
[47] UNI075163.

Thus, any reliance on any of these statements by Defendants should be weighed accordingly. Furthermore, even if the Examiner had not rejected the Mr. Rosenblatt's declaration, none of the statements can be read as anything more than a primer of checksums. There is no attempt to distinguish the '216 Patent in these statements. Instead, these statements were provided to show another reason why the Hellman and Grundy references could not be combined. Specifically, these statements demonstrate that checksums are not the same as the cryptographic hash functions disclosed in Hellman and thus it would be improper to combine the references, by substituting Grundy's simple checksum for the more complicated cryptographic hash functions of Hellman. Thus, these statements do not contain a "clear and deliberate statement" that meets the high standard Defendants must prove to show a disclaimer of claim scope. *Honeywell Int'l, Inc. v. Universal Avionics Sys. Corp.*, 493 F.3d 1358, 1365 (Fed. Cir. 2007).

2. Statements in response to the second office action also fall short of a disclaimer

Defendants have also indicated that the following statements support their contention that Uniloc disclaimed the use of checksums and algorithms used to test data integrity:

> Statement #9: "Uniloc submits that based on the Examiner's statement in the Order, Grundy's data validation checksums do not produce a unique ID that could be used by Hellman."[48]
>
> Statement #10: "Uniloc also argued that Grundy's checksum did not generate 'a licensee unique ID' because Grundy's checksum algorithm, by its very nature, destroys any uniqueness."[49]
>
> Statement #11: "The simple checksums described in Grundy are an entirely different class of algorithm and cannot provide the security required by Hellman. (Pooch Dec. ¶¶ 32-39). As Dr. Pooch explains that 'Grundy…describes several conventional for checksums.' (Pooch Dec. ¶ 8.) For example, 'the checksums described in Grundy are not cryptographic functions, but rather appear to be used to check, for example, for typographical data entry errors or transmission errors.' (Pooch Dec., ¶ 32.)"[50]

[48] UNI075202.
[49] UNI075203.
[50] UNI075218.

> Statement #12: "In the rejection of claim 1 on page 14, on the other hand, the Examiner proposes to replace Hellman's cryptographic function generator 38 with the checksum of Grundy to provide the summation algorithm limitation absent from the teachings of Hellman. (Second Action, p. 15; bottom) However, if these references are combined as the Examiner suggests, with Grundy's error-checking checksum replacing Hellman's cryptographic function generator, the Examiner can no longer take credit for the 'uniqueness' feature provided by Hellman because the source of that uniqueness, the one-way compressive hash function having a 100:1 X/Y bit ratio, would be replaced by Grundy's checksum. Uniloc therefore requests that the obviousness rejection of claims 2, 12, and 17 be considered and withdrawn."[51]

As with the statements made in the first office action, these statements simply point out that the Grundy checksums are not used to generate Grundy's "registration code" and could not be combined with Hellman because checksums are not sophisticated cryptographic functions. Also worth noting is that these statements do not mention "data integrity" nor do they appear to be germane to such a subject.

In sum, a review of Uniloc's statements in the reexamination prosecution history demonstrates there was no disclaimer of subject matter. Uniloc simply made factual statements about what was (and was not) contained in the Grundy reference and why it was technically infeasible, and therefore legally improper, to combine Grundy and Hellman as suggested by the examiner. Uniloc never made statements to the effect that "the claims are allowable over Grundy because Grundy's X does not meet the claimed feature Y" and that are typically associated with a disclaimer.

3. Grundy dealt with a checksum, not a licensee unique ID

Assuming, *arguendo*, that the Defendants are correct that some of Uniloc's statements about Grundy amount to a disclaimer, at most such an alleged "disclaimer" should only apply to checksum algorithms such as the one shown in Grundy and not summation algorithms in general "used to test data integrity" as urged by the Defendants. As shown from the portions of the intrinsic

[51] UNI075221.

record identified by the Defendants above, most, if not all, of the identified portions deal with the subject of Grundy's checksum algorithm. As noted by the Patent Office in the NIIRC, Grundy's checksum algorithm is distinct from the algorithms used to create licensee unique IDs ("unique licensee ID generating means"). Again, on this subject the Examiner explained:

> [Uniloc] has persuasively argued that the summation [*i.e.*, the checksum] disclosed by Grundy is used in the context of merely verifying the correctness of information related to the user and is not being used to generate an ID per se. Since the information is not being used for the same purpose, one skilled in the art therefore would not use the algorithm of Grundy as part of the generation of the claimed licensee unique ID.[52]

In other words, the Patent Office found that Grundy's checksum algorithm is different than the unique licensee generating means claimed in the '216 Patent. In light of this recognized distinction, if there was a disclaimer of subject matter (and Uniloc is adamant that there was not), the Court should limit such a disclaimer to checksums like the one shown in Grundy and decline Defendants' invitation to apply such a disclaimer to summation algorithms "used to test data integrity" as that was never at issue in the cited portions of the reexamination file history.

4. Defendants' alleged disclaimer conflicts with the disclosed invention and creates a subjective test

Defendants would have the Court adopt a disclaimer excluding a licensee unique ID/security key generated by a checksum, summation algorithm, summer, or equivalent thereof, used to test data integrity. Defendants' data integrity exclusion is nonsensical in light of the patent specification, which teaches that the LUID and the RUID are supposed to be compared and if they match, allow full access to the software.[53] Since the local unique ID generating means and the remote unique ID generating means are the same algorithm (located on different computers), the inputs to those algorithms need to be the same in order for the outputs to match.

The term "data integrity" is hopelessly vague and does nothing to serve the purpose of claim

[52] UNI076155.
[53] See Exh. C at claims 1 and 19.

construction. Under Defendants' "data integrity" limitation, an alleged infringer could always argue that the local and remote generating means act as a test of data integrity because if the inputs are not identical the outputs would not match. Thus, under Defendants' proposed limitation, any generating means algorithm would be disclaimed. Such an interpretation would effectively eviscerate the entire '216 Patent.

Further, the "used to test data integrity" limitation creates a subjective test that would not aid in objectively determining infringement. Under Defendants' proposal, a given algorithm may or may not infringe depending on its intended purpose. In order to ascertain such a purpose, it would be necessary to determine why a particular algorithm was implemented.

The Federal Circuit has said that hash values produced by an MD5 hash algorithm satisfy the licensee unique ID requirement. *See Uniloc II*, 632 F.3d at 1304 ("As this court held in *Uniloc I,* 290 Fed. App'x at 342, there was substantial evidence for a jury to conclude that the output of the MD5 and SHA1 algorithms was a licensee unique ID."). Defendants would have the Court alter the Federal Circuit's findings to conclude that MD5 hash values may infringe under some circumstances but not others. In fact, Defendants would require the fact finder to determine the state of mind of the software programmer to ascertain why the programmer used a particular hash algorithm, such as MD5. Such a test would lead to inconsistent results and should be rejected. In addition, the Patent Office does not appear to have adopted Defendants' view as evidenced by the NIIRC, which states: "[a]s the Federal Circuit has pointed out, the MD5 algorithm (described in RFC 1321, attached to this action) could be such a means."[54]

B. Group B Defendants' Attempt to Limit the Types of Information Used to Generate a "Licensee Unique ID" Should be Rejected

During the reexamination, the Examiner mistakenly asserted that Hellman discloses a "licensee unique ID" (*i.e.*, "a unique identifier associated with a licensee" per the Federal Circuit).

[54] UNI076155.

As with Grundy, Uniloc had to correct the Examiner's technical misunderstanding of Hellman. When read in context, all of the statements listed by Defendants are consistent with Uniloc's efforts to educate the Examiner that Hellman did not disclose a "unique licensee ID" consistent with the Federal Circuit's holdings. Importantly, there are no "clear" or "unmistakable" statements that meet the Federal Circuit's high burden for disclaimer, much less one that supports Defendants' lengthy proposal.

The District of Rhode Island construed the phrases "licensee unique ID" and "security key" to mean "a unique identifier associated with a licensee." *Uniloc,* 447 F.Supp. 2d at 183-89. In *Uniloc I,* the Federal Circuit discussed at length the proper construction of "licensee unique ID" and ultimately affirmed the district court's concise construction stating "the district court correctly construed the 'licensee unique ID' as a unique identifier associated with a licensee that can be, but is not limited to, personally identifiable information about the licensee or user." *Uniloc I,* 290 Fed. App'x at 344. In its opinion, the Federal Circuit acknowledged, "Microsoft is, however, correct that the licensee unique ID cannot be based solely on platform-related user information." *Id.* at 343. Further, the Court also stated "[w]e do not read these limitations as requiring that this information be uniquely *about* the user instead of just unique *to* the user." *Id.* at 344 (emphasis in original). Despite these various remarks, at the end of the day the Federal Circuit decided that a succinct construction was proper and decided not to alter the construction of "licensee unique ID" beyond "a unique identifier associated with a licensee." *See Uniloc II,* 632 F.3d at 1299-1300.

The Group B Defendants' alleged disclaimer appears to be derived from the NIIRC.[55] To the extent Defendants intend to argue that the Examiner's "Reasons for Allowance"[56] mandate their alleged disclaimer, Federal Circuit precedent is not on their side. *See ACCO Brands, Inc. v. Micro*

[55] UNI076154.

[56] The Examiner's statements are actually contained in the section of the NIIRC entitled "**STATEMENT OF REASONS FOR PATENTABILITY AND/OR CONFIRMATION."** See UNI076154. Uniloc contends that this section is the same as a "Reasons for Allowance" section.

Sec. Devices, Inc., 346 F.3d 1075, 1079 (Fed. Cir. 2003) ("there is no obligation to respond to an examiner's statement of Reasons for Allowance, and the statement of an examiner will not necessarily limit a claim."); *Inverness Medical Switzerland v. Princeton Biomeditech Corp.,* 309 F.3d 1365, 1372-73 (Fed. Cir. 2002) (rejecting contention that statement in examiner's Reasons for Allowance governed construction of disputed claim term).

Moreover, the Examiner's Reasons for Allowance are internally inconsistent and conflict with the Federal Circuit's construction. The Reasons for Allowance specifically provide:

> During reexamination, claims are given the broadest reasonable interpretation consistent with the specification and limitations in the specification are not read into the claims (*In re Yamamoto,* 740 F.2d 1569, 222 USPQ 934 (Fed. Cir. 1984)). *Where there exists a final decision by the Court of Appeals for the Federal Circuit regarding the construction of claims, an interpretation is not reasonable where it is inconsistent with that decision*.[57]

Despite having acknowledged that his claim construction must be consistent with the Federal Circuit's, the Examiner went on to state:

> The licensee unique ID generated by the means recited in each of the claims must be derived from at least [sic] piece of information that is specific to the user, such as name, billing information, or product information unique to the instantiation [sic][58] entered by the user. The information cannot be specific to the computer or independently generated by the computer.[59]

As noted above, this is not how the Federal Circuit construed "licensee unique ID." The Federal Circuit explained that "the licensee unique ID cannot be based *solely* on platform related user information" and that licensee unique ID is a "unique identifier *associated with* the licensee that can be, *but is not limited to*, personally identifiable information about the user of licensee." The Examiner's statement that the licensee unique ID cannot be derived from information "*specific to* the computer or independently generated by the computer" conflicts with the Federal Circuit's observation that it cannot be derived *solely from* such information. Additionally, the difference in

[57] UNI076154 (emphasis added).
[58] It is believed that the Examiner meant to state "installation."
[59] UNI076154.

using information *associated with* a user as opposed to information *specific to* a user in generating a "licensee unique ID" was heavily contested in the Microsoft litigation. The Federal Circuit ultimately concluded that information "associated with the user" is the proper standard.

Having acknowledged that his construction must be consistent with the Federal Circuit's, Uniloc believes that the Examiner did not purposefully depart from the Federal Circuit's well-established construction, but was unfortunately imprecise in his choice of words. The Group B Defendants appear to be seizing on *the Examiner's* inexact articulation of the Federal Circuit's prior holdings as a basis to adopt a disclaimer that (1) neither the Federal Circuit nor Uniloc have embraced and (2) that conflicts with that Court's own construction. As noted above, the Federal Circuit has examined the construction of "licensee unique ID" at length. Uniloc asserts that there is no reason to depart from the Federal Circuit's established construction, especially when the Examiner himself has indicated no intention or reason to do so. Accordingly, the Court should reject the Group B Defendants' attempt to re-construe and narrow this term because neither the Examiner's Reasons for Allowance nor his inconsistent statements made therein provide a basis to do so.

VI. CONCLUSION

For the foregoing reasons, Uniloc requests that the Court adopt its proposed constructions and reject those proffered by the Defendants. Furthermore, Uniloc submits that a close reading of the reexamination file history demonstrates that Uniloc only described features of the prior art in an effort to demonstrate why, from a technical perspective, the Examiner would not be able to successfully maintain a §103 rejection. *Computer Docking Station,* 519 F.3d at 1375. Importantly, Uniloc did not make limiting distinctions over either Grundy or Hellman and Defendants cannot show that the high threshold for prosecution history disclaimer is satisfied. Therefore, Uniloc respectfully requests that the previous constructions remain unaltered.

Dated: September 12, 2011.

Respectfully Submitted:

/s/ Barry J. Bumgardner
Edward R. Nelson, III
Texas State Bar No. 00797142
Barry J. Bumgardner
Texas State Bar No. 24041918
Steven W. Hartsell
Texas State Bar No. 24040199
S. Brannon Latimer
Texas State Bar No. 24060137
Jaime K. Olin
Texas State Bar No. 24070363
NELSON BUMGARDNER CASTO, P.C.
3131 West 7th Street, Suite 300
Fort Worth, Texas 76107
(817) 377-9111
(817) 377-3485 (fax)
enelson@nbclaw.net
barry@nbclaw.net
shartsell@nbclaw.net
blatimer@nbclaw.net
jolin@nbclaw.net

T. John Ward, Jr.
Texas State Bar No. 00794818
J. Wesley Hill
Texas State Bar No. 24032294
WARD & SMITH LAW FIRM
111 West Tyler St.
Longview, Texas 75601
Tel: (903) 757-6400
Fax: (903) 757-2323
jw@wsfirm.com
wh@wsfirm.com

ATTORNEYS FOR PLAINTIFFS
UNILOC USA, INC. AND
UNILOC SINGAPORE PRIVATE LIMITED

CERTIFICATE OF SERVICE

I hereby certify that on the 12th day of September 2011, I electronically filed the foregoing document with the clerk of the court for the U.S. District Court, Eastern District of Texas, Tyler Division, using the electronic case filing system of the court. The electronic case filing system sent a "Notice of Electronic Filing" to the attorneys of record who have consented in writing to accept this Notice as service of this document by electronic means.

/s/ Barry J. Bumgardner