United States District Court,
S.D. Texas, Houston Division.

SUPERSPEED SOFTWARE, INC,
v.
ORACLE CORP.

Dec. 16, 2005.

ORDER

VANESSA D. GILMORE**, District Judge.**

Pending before the Court is Plaintiff Superspeed Software, Inc.'s Opening Brief Regarding Claim Construction (**Instrument No. 69**), filed on June 3, 2005.

I.

Plaintiff Superspeed Software, Inc. ("Superspeed") has sued Defendant Oracle Corporation ("Oracle") for infringement of three U.S. patents related to a "Method and System for Coherently Caching I/O Devices Across a Network." (Instrument No. 69, ExhA-C). The three patents that are in dispute are U.S. Patent No. 6,577,226 (Instrument No. 69, Exh A) (hereinafter "the '226 Patent"); U.S. Patent No. 5,918,244 (Instrument No. 69, Exh B) (hereinafter "the '244 Patent"); and U.S. Patent No. 6,651,136 (Instrument No. 69, Exh C) (hereinafter "the '136 Patent") (collectively "the Superspeed Caching Patents"). Superspeed's infringement allegations apply to five claims in the three patents: '226 Patent claims 27 and 35; '244 Patent claims 15 and 22; and '136 Patent claim 1. (Instrument No. 69, Exh D).

The three patents in suit are related, and all claim priority to a common parent application. The common parent, U.S. Application Serial No. 238,815, was filed May 6, 1994, and issued as the '226 Patent on November 19, 1996. Both the '244 Patent and the '136 Patent are continuation applications claiming priority to the '226 Patent filing date. The '244 Patent issued on June 29, 1999; the '136 Patent issued on November 18, 2003. (Instrument No. 69, at 1).

In general terms, the claimed inventions of the patents-in-suit concern methods for storing local Random Access Memory ("RAM") data (*i.e.,* "cache") obtained from persistent storage devices, such as disks (also known as hard drives) and tapes. "In addition to computers or servers which process and manipulate data, networks almost always [contain these] persistent storage devices." (Instrument No. 69, at 4). The "storage devices maintain the digital records or data [that] can be accessed through the network by the servers." (Id.). Plaintiff explains the function and capabilities of storage devices and caching operations as follows:

Each time a server accesses data from a disk or other persistent storage device, the server must wait for that access operation to complete before it can work with the data. The process of accessing data stored on a disk is, in computer terms, very slow. If the computer must communicate with the disk over a network, as is the

case in a cluster network with shared disks, the access time is even slower. To speed up the retrieval of data from a storage device, the data used by the computer may be temporarily loaded or "cached" in memory that is local to the computer-typically in much faster dynamic random access memory (or system memory) that is internal to the computer itself.... Data is stored in the cache while it is used by the computer (or node), significantly speeding up processing involving the cached data. The computer returns data to the persistent storage device either at regular intervals or based on non-use of that particular piece of data.

(Instrument No. 69, at 4-5).

Plaintiff Superspeed explains, however:

In a cluster network in which the same disk is accessed by multiple computers (a "shared disk system"), caching operations, if not properly controlled, can destroy the integrity of the data. For example, if two different nodes on a network access the same piece of data at the same time, and both have made copies of the data in their own local caches, then modifications to the data by one of the computers may render the data in the other computer obsolete. If both computers make modifications to the same piece of cached data and then attempt to write the data back to the storage device, a data conflict exists. To prevent these conflicts or the possibility that any computer will unknowingly have stale data in its cache, the network must implement a methodology to alert the computer nodes when the data has been modified by another computer. Such methodologies are often referred to as "cache coherency" protocols or systems.

(Id. at 5-6).

According to Plaintiff Superspeed, "[t]he Superspeed caching software has offered significant advantages relative to other caching software products available in the early 1990's, including (1) minimizing network traffic and thereby improving the ability to add a large number of servers to the cluster without causing significant slowdowns; (2)the ability to more efficiently use cache memory in a server; and (3) maintaining cache coherency when new computers join the cluster." (Id. at 6).

Specifically, the Superspeed Caching Patents describe a "method and system for coherently caching Input/Output devices ("I/O devices") across a network." (Instrument No. 69, Exh A-C). According to the Abstract description of the patents-in-suit:

The cache keeps regularly accessed disk I/O data within RAM that forms part of a computer systems main memory. The cache operates across a network of computer systems, maintaining cache coherency for the disk I/O devices that are shared by the multiple computer systems within that network. Read access for disk I/O data that is contained within the RAM is returned much faster than would occur if the disk I/O device was accessed directly. The data is held in one of three areas of the RAM for the cache, dependent on the size of the I/O access. The total RAM containing the three areas for the cache does not occupy a fixed amount of a computers main memory. The RAM for the cache grows to contain more disk I/O data on demand and shrinks when more of the main memory is required by the computer system for other uses. The user of the cache is allowed to specify which size of I/O access is allocated to the three areas for the RAM, along with a limit for the total amount of main memory that will be used by the cache at any one time.

(Id.).

Plaintiff Superspeed asserts that claim 27 of the '226 Patent, and claim 15 of the '244 Patent include

limitations directed to "the minimization of network communication traffic by keeping track of the computers that are caching a particular storage device, and notifying *only those computers* when a data block from that storage device has been modified and written back to disk." (Id. at 6-7) (emphasis in original). Plaintiff Superspeed contends that claim 35 of the '226 Patent describes "the use of multiple 'bucket sizes' in the cache of the computer" to "maximize[ ] system resources by accommodating large segments of data (in the large bucket size), and, at the same time, avoiding excessive waste when smaller segments are being cached." (Id. at 7). Finally, Plaintiff Superspeed contends that "claim 1 of the ' 136 Patent, is a protocol to enable new nodes to join the network on the fly, with the system automatically detecting the new node and freezing caching operations until communications with the new node is established." (Id. at 7-8).

The patent specifications describe the basic components of the claimed inventions in terms of three primary software components: (1) the control block ("TCB"); (2) the cache hack ("TCH"); and (3) the cache memory blocks ("TCMBs"). According to the report of Defendant's expert, Marshal Kirk McKusick, the control block is a disk control structure for each disk present on the computer system. (Instrument No. 77, Exh C, at 6-8). The cache hack is a cache control structure that contains the cache memory blocks, and also contains the disk block hash table which is used to locate specific data "buckets" in the cache. (Id. at 7). Data buckets are storage locations in the cache to which data are copied from the disk. (Id. at 6). The cache memory blocks are bucket control structures that contain pointers to particular data buckets and which are used to locate specific data buckets in order to get desired data from the cache. (Id.).

At issue is the interpretation of disputed terms in the contested claims of the Superspeed Caching Patents. Prior to the Markman hearing, the parties exchanged proposed constructions and held several meetings and conferences to identify genuine areas of disagreement. The following claim terms are in dispute: (1) Input/Output device; (2) list; (3) cache driver; (4) write instruction; (5) invalidate data; (6) disabling cache operations; (7) enabling caching operations. Plaintiff Superspeed's and Defendant Oracle's proposed constructions of these disputed terms are set out in the following chart.

DISPUTED TERM	PATENT NO.	SUSPENDED'S PROPOSED CONSTRUCTION	ORACLE'S PROPOSED CONSTRUCTION
Input/Output device or I/O device	'226 '244 '136	a disk or other persistent storage device shared by multiple computer systems in a network	disk or other persistent storage device
list	'226	No definition is needed. Alternatively, "list" means a set of data items.	an item-by-item series of numbers, words, or characters
cache driver	'244	a program that implements a cache	device driver that controls a cache
write instruction	'226	an access or operation that initiates a transfer of information to storage	command that transfers data
invalidate data	'226 '244	to indicate that a portion of data in a cache is no longer up to date	make data unusable
disabling cache operations	'136	suspending caching	prohibiting use of data in the cache
enabling caching operations	'136	resuming caching	permitting use of data in the cache

(Instrument No. 69, at 3; Instrument No. 77, at 8, 14, 20, 22, 25, 30).

The contested claims include: (i) Claim 1 of the '136 Patent; (ii) Claims 27 and 35 of the '226 Patent; and (iii) Claims 15 and 22 of the '244 Patent. (Instrument No. 69, Exh D). The Claims read as follows:

U.S. PATENT NO. 5,577,266

27. A method for accelerating access to data on a network comprising the steps of:
creating a cache in the RAM of a computer connected to the network;

creating a data structure in the computer for each of a plurality of **I/O devices** connected to said network for which data may be cached by said computer, each said data structure including a **list** of all computers on said network that permit caching with respect to the **I/O device** corresponding to said data structure;

intercepting a **write instruction** to one of said plurality of **I/O devices** from said computer; and

communicating over the network individually with each computer in the **list** of computers in the data structure corresponding to said one of said **I/O devices** to **invalidate data** in caches on the network corresponding to said one of said plurality of **I/O devices.**

35. The method of claim 27 wherein the step of creating a cache in the RAM of a computer comprises creating a plurality of caches in the RAM, each having a different bucket size for storing date.

(Instrument No. 69, Exh A, at 28:13-30, 49-57) (emphasis added).
U.S. PATENT NO. 5,918,244

15. A caching system comprising:
a network;

a plurality of computers connected over said network, each computer having a memory;

a plurality of **I/O devices** connected to said network; and

a plurality of **cache drivers,** each resident in one of said computers, for creating a cache in the memory of the computer in which the **cache driver** resides for caching data from selected ones of said **I/O devices,** each **cache driver** including executable remote messaging code that forms a computer communication channel with any other of said computers on said network via which messages relating to caching may be communicated with **cache drivers** on any of said computers on said network, said executable remote messaging code further saving a remote connection address for each of the communication channels;

wherein at least one of said **cache drivers** further includes executable interception code for intercepting a **write instruction** to one of said plurality of **I/O devices** and executable invalidate code that sends a message to **invalidate data** through selected ones of the communication channels to the caches of any computer that is caching said one of said plurality of **I/O devices.**

22. The caching system of claim 15 wherein each of said computers runs an operating system and each of said **cache drivers** allocate to the cache, space in the memory of the computer in which the **cache driver** resides, which creating the cache.

(Instrument No. 69, Exh B, at 26:55-27:11, 27:42-46) (emphasis added).
U.S. PATENT NO. 6,651,136

1. A method for coherently caching **I/O devices** available for shared access on a network comprising: providing a plurality of computers on the network each with cache software;

disabling cache operations upon finding that a new computer joined the network; and

enabling caching operations at each computer after each computer has connections in place with the cache software of every other computer on the network.

(Instrument No. 69, Exh C, at 26:12-26:22) (emphasis added).
The parties have agreed upon definitions for certain terms that are used within the above disputed terms and proposed constructions. The agreed definitions and terms are as follows:

Term	Agreed Construction
cache	a portion of system main memory (e.g., RAM) used for temporary storage of data
cache software	a program that creates and controls a cache
caching	storing in cache
intercepting	to stop, deflect, or interrupt the progress or intended course of
network	communication facilities that link points at which computers or devices may be connected
the memory of the computer	the system main Random Access Memory of a computer
the RAM of the computer	the system main Random Access Memory of a computer

(Instrument No. 69, at 9-10)

II.

Whoever without authority makes, uses, or sells any patented in vention within the United States during the term of the patent therefor, infringes the patent. 35 U.S.C. s. 271(a). The determination of whether a claim of a patent has been infringed is a two-step process. First, the Court must determine the meaning and scope of the patent claims asserted to be infringed. *See* Bell Atl. Network Servs., Inc. v. Covad Comms. Group, Inc., 262 F.3d 1258, 1267 (Fed.Cir.2001); Markman v. Westview Instruments, Inc., 52 F .3d 967, 976 (Fed.Cir.1995), *aff'd,* 116 S.Ct. 1384 (1996). This step is commonly known as claim construction or interpretation. Second, the court must compare the claims alleged to be infringed to the accused device. *See* Bell Atlantic, 262 F.3d at 1267; Markman, 52 F.3d at 976.

Claim interpretation is a matter of law involving the review of patent specifications, prosecution history, language of the patent claims, and, if necessary, extrinsic evidence. *See* Texas Instruments v. U.S. Int'l Trade Comm'n, 988 F.2d 1165, 1171 (Fed.Cir.1993). The court must decide and explicate its findings regarding claim construction on the record. *See* Genentech, Inc. v. Wellcome Foundation Ltd., 29 F.3d 1555 (Fed.Cir.1994).

"[A]s a general rule, all terms in a patent claim are to be given their plain, ordinary and accustomed meaning to one of ordinary skill in the relevant art." Rexnord Corp. v. Laitram Corp., 274 F.3d 1336, 1342 (Fed.Cir.2001). *See also* Toro Co. v. White Consol. Indus., Inc., 199 F.3d 1295, 1299 (Fed.Cir.1999) ("[W]ords in patent claims are given their ordinary meaning in the usage of the field of the invention, unless the text of the patent makes clear that a word was used with a special meaning."). In addition, unless required to do otherwise, a court should give a claim term "the full range of its ordinary meaning as understood by an artisan of ordinary skill." Rexnord, 274 F.3d at 1342 (citing Johnson Worldwide Assocs., Inc. v. Zebco Corp., 175 F.3d 985, 989 (Fed.Cir.1999)).

In construing patent claims, the Court looks to the intrinsic evidence of claim meaning-the claims, the specification of the patent, and the prosecution history of the patent. *See* Vitronics Corp. v. Conceptronic Inc., 90 F.3d 1576, 1582-83 (Fed.Cir.1996). If the intrinsic evidence is clear, "it is improper to rely on extrinsic evidence in construing the patent claims." Id. at 1583. In fact, when the meaning of a disputed claim term is clear from the intrinsic evidence, *i.e.,* the intrinsic evidence is unambiguous, then that meaning and no other must prevail; it is improper for the court to rely on extrinsic evidence to alter or supersede that meaning. *See* Bell & Howell Document Mgmt. Prods. Co. v. Altek Sys., 132 F.3d 701, 706 (Fed.Cir.1997).

It is well established that "the language of the claim defines the scope of the protected invention." *Bell Communications Research, Inc. v. Vitalink Communications Corp.,* 55F.3d615,619 (Fed.Cir.1995). The Court first must look at the claim language and ascribe the plain and ordinary meaning to the phrase. *See* Hockerson-Halberstadt, Inc. v. Avia Group Int'l, Inc., 222 F.3d 951, 955 (Fed.Cir.2000). The Federal Circuit has indicated that the claim language itself defines the scope of the claim, and a construing court does not accord the specification, prosecution history, and other relevant evidence the same weight as the claims themselves. *See* Eastman Kodak Co. v. Goodyear Tire & Rubber Co., 114 F.3d 1547, 1552 (Fed.Cir.1997), *overruled on other grounds by* Cybor Corp. v. FAS Techs. Inc., 138 F.3d 1448, 1456 (Fed.Cir.1998). "A court must therefore presume that the terms in the claim mean what they say and, unless otherwise compelled, give full effect to the ordinary and accustomed meaning of claim terms." Johnson Worldwide Assocs., Inc., 175 F.3d at 989.

Although the focus should be on the ordinary meaning, the specification and prosecution history cannot be ignored. *See* Transmatic, Inc. v. Gulton Indus., Inc., 53 F.3d 1270, 1277 (Fed.Cir.1995) (stating that claim terms are given their ordinary meaning unless the specification, prosecution history, and other claims indicate a contrary intent). A patent specification is the written description of the patented invention that "describe[s] the manner and process of making and using" the patented invention. Autogiro Co. of America v. United States, 384 F.2d 391, 397-98 (Ct.Cl.1967). "The descriptive part of the specification aids in ascertaining the scope and meaning of the claims inasmuch as the words of the claims must be based on the description. The specification is, thus, the primary basis for construing the claims." Standard Oil Co. v. Am. Cynamid Co., 774 F.2d 448, 452 (Fed.Cir.1985). *See also* Multiform Desiccants, Inc. v. Medzam, Ltd., 133 F.3d 1473, 1478 (Fed.Cir.1998)("In most cases, the best source for discerning the proper context of claim terms is the patent specification wherein the patent applicant describes the invention.").

The prosecution history consists of the complete record of the proceedings before the PTO and includes the prior art cited during the examination of the patent. Autogiro, 384 F.2d at 399. Like the specification, the prosecution history provides evidence of how the PTO and the inventor understood the patent. *See* Lemelson v. Gen. Mills, Inc., 968 F.2d 1202, 1206 (Fed.Cir.1992). However, because the prosecution history represents an ongoing negotiation between the PTO and the applicant, rather than the final product

of that negotiation, it often lacks the clarity of the specification and thus is less useful for claim construction purposes. *See* Inverness Med. Switz. GmbH v. Awarner Lambart Co., 309 F.3d 1373, 1380-82 (Fed.Cir.2002).

Use of the specification and the prosecution history, however, must be balanced with the principle that it is impossible to read a particular embodiment into the claim. *See* Comark Communications, Inc. v. Harris Corp., 156 F.3d 1182, 1186-87 (Fed.Cir.1998). In other words, while claims should be read in view of the specification it is improper to limit the scope of a claim to the preferred embodiment or specific examples disclosed in the specification. *See* Ekchian v. Home Depot, Inc., 104 F.3d 1299, 1303 (Fed.Cir.1997). The Federal Circuit has consistently found that a patent is not restricted to the examples but is defined by the words of the claims. *See* Specialty Composites v. Cabot Corp., 845 F.2d 981, 987 (Fed.Cir.1988).

What is important is what the elements of the claim require, not what they "do not cover." *See, e.g.,* NeoMagic Corp. v. Trident Microsystems, Inc., 287 F.3d 1062, 1074 (Fed.Cir.2002). Claims are not to be interpreted in view of the accused infringing device. *See Young Dental Mfg. Co. v. Q3 Special Prods., Inc.,* 112F.3d 1137, 1141 (Fed.Cir.1997). Courts have routinely rejected an accused infringer's attempt to show that his device is outside the scope of the claims by asserting a distinction that is not specifically claimed. *See, e.g.,* Shamrock Techs., Inc. v. Med. Sterilization, Inc., 903 F.2d 789, 793 (Fed.Cir.1990).

The court may also rely on extrinsic evidence to interpret the meaning of a claim. In *Markman v. Westview Instruments,* the Federal Circuit discussed the principles governing claim interpretation, including the role of the specification, prosecution history, and "extrinsic evidence." 52 F.3d at 979-80. It emphasized that extrinsic evidence serves a limited purpose; it facilitates a judge's understanding of the meaning of patent claim language. *Id.* The court explained that:

Extrinsic evidence consists of all evidence external to the patent and prosecution history, including expert and inventor testimony, dictionaries, and learned treatises. This evidence may be helpful to explain scientific principles, the meaning of technical terms, and terms of art that appear in the patent and prosecution history. Extrinsic evidence may demonstrate the state of the prior art at the time of the invention. It is useful to show what was then old, to distinguish what was new, and to aid the court in the construction of the patent.

Id. at 980 (internal citations and quotations omitted). "The court may, in its discretion, receive extrinsic evidence in order 'to aid the court in coming to a correct conclusion' as to the 'true meaning of the language employed' in the patent." *Id.* (quoting Seymour v. Osborne, 78 U.S. (11 Wall.) 516, 546 (1871) (reviewing a decree in equity).

The Federal Circuit recently clarified the role of extrinsic evidence in claim construction. In Phillips v. AWH Corp., 415 F.3d 1303, 2005 U.S.App. LEXIS 13954 (Fed Cir. July 12, 2005), the Federal Circuit recognized that it has "viewed extrinsic evidence in general as less reliable than the patent and its prosecution history in determining how to read claim terms." 415 F.3d 1303, 2005 U.S.App. LEXIS 13954 at *40. The court explained:

First, extrinsic evidence by definition is not part of the patent and does not have the specification's virtue of being created at the time of patent prosecution for the purpose of explaining the patent's scope and meaning. Second, while claims are construed as they would be understood by a hypothetical person of skill in the art, extrinsic publications may not be written by or for skilled artisans and therefore may not reflect the

understanding of a skilled artisan in the field of the patent. Third, extrinsic evidence consisting of expert reports and testimony is generated at the time of and for the purpose of litigation and thus can suffer from bias that is not present in intrinsic evidence.... Finally, undue reliance on extrinsic evidence poses the risk that it will be used to change the meaning of claims in derogation of the indisputable public records consisting of the claims, the specification and the prosecution history, thereby undermining the public notice function of patents.

415 F.3d 1303, *Id*. at *40-41 (internal citations and quotations omitted).

Thus, the *Phillips* court concluded, "[i]n sum, extrinsic evidence may be useful to the court, but it is unlikely to result in a reliable interpretation of patent claim scope unless considered in the context of the intrinsic evidence." 415 F.3d 1303, *Id*. at *41-42. The *Phillips* court did not completely invalidate the use of extrinsic evidence "because extrinsic evidence can help educate the court regarding the field of the invention and can help the court determine what a person of ordinary skill in the art would understand claim terms to mean." 415 F.3d 1303, *Id*. at 42. However, the court recognized that "the specification is 'the single best guide to the meaning of a disputed term,' and ... the specification 'acts as a dictionary when it expressly defines terms used in the claims or when it defines terms by implication.' " 415 F.3d 1303, *Id*. at *47-48 (quoting Vitronics Corp. v. Conceptronic, Inc., 90 F.3d 1576, 1582 (Fed.Cir.1996); *see also* Irdeto Access, Inc. v. Echostar Satellite Corp., 383 F.3d 1295, 1300 (Fed.Cir.2004) ("Even when guidance is not provided in explicit definitional format, the specification may define claim terms by implication such that the meaning may be found in or ascertained by a reading of the patent documents.").

III.

A.

Disputed Term:	Patent No.	Superspeed's Construction:	Oracle's Construction:
Input/Output device or I/O device	'226, '244, '136	a disk or other persistent storage device shared by multiple computer systems in a network	disk or other persistent storage device

The term "I/O device" is an abbreviation of the term "Input/Output device." All of the disputed claims directly or indirectly reference the term "I/O device." (Instrument No. 69, Exh D). The parties agree that the definition of "I/O device" includes "disk or other persistent storage device." Superspeed contends, however, that two additional limitations should be included in the definition of "I/O device," showing that the storage device must be "shared by multiple computer systems" and that those systems must be "in a network." (Instrument No. 69, at 15).

Claim 27 of the '226 Patent claims "each of a plurality of I/O devices connected to said network" and claim 15 of the '244 patent claims the same language. ('226 Patent, at 28:17-18; '244 Patent, at 27:7) (emphasis added). Moreover, claim 1 of the '136 Patent claims "[a] method for coherently caching I/O devices **available for shared access on a network.**" ('136 Patent, at 26:33-34) (emphasis added). Although certain limitations in the claim language require the networking or sharing of I/O devices, networking or sharing is not inherent in the I/O devices themselves. In fact, the limitations of the claims relate to the utilization of I/O devices in a shared network, but do not explicitly describe networking or sharing as an intrinsic feature of an I/O device.

Because the claims explicitly state that the I/O devices are "connected to said network" or "available for shared access on a network," there is no need to define the term with an additional limitation stating that the storage device is "shared by multiple computer systems in a network." If the term "I/O device" were defined in such a way it would render the explicit limitations of the claims redundant or mere surplusage. *See* Texas Instruments Inc. v. United States Int'l Trade Comm'n, 988 F.2d 1165, 1171 (Fed .Cir.1993) (noting that a proposed construction would render the disputed claim language mere surplusage because "courts can neither broaden nor narrow claims to give the patentee something different than what he has set forth") (quoting Autogiro Cp. of Am. v. United States, 384 F.2d 391, 396 (Ct.Cl.1967). Without the additional limitation that the device is "shared by multiple computer systems in a network," the parties essentially agree that the term "I/O device" means "disk or other persistent storage device." Accordingly, the Court finds that the proper construction for the term "I/O device" is " ***disk or other persistent storage device.***"

B.

Deputed Term:	Patent No.	Superspeed's Construction:	Oracle's Construction:
"list"	'226	No definition is needed. Alternatively, "list" means a set of data items.	an item-by-item series of numbers, words, or characters

The term "list" appears in claim 27 of the '226 Patent. It does not appear in the contested claims of the '136 Patent or the '244 Patent. Claim 27 of the '226 Patent states in material part that "each ... data structure including a **list** of all computers on said network that permit caching to the I/O device ..." ('226 Patent, at 28:20-23) (emphasis added). Claim27 further claims "communicating over the network individually with each computer in the **list** of computers in the data structure." (*Id*. at 28:26-27) (emphasis added).

The Court must first look at the claim language and ascribe the plain and ordinary meaning to the phrase. *See* Hockerson-Halberstadt, 222 F.3d at 955. The '226 Patent claims do not provide an explicit definition or attach a special meaning to the term "list." There is also no indication that the ' 226 Patent specifications use the term "list" in any other way than the term's ordinary and common meaning. The '226 Patent specifications provide that "the cache driver (10) uses its remote message communication channels (18) to send a message to each of the remote cache drivers in the **list** contained in the TCB (16) disk control structure." (Id. at 3:33-42) (emphasis added). The specifications further provide that "[t]he **list** of remote computers that can access the disk (12) is obtained from the TCB (16) disk control structure and a message is sent to all these remote computers using the remote message communication channels (18)." (Id. at 5:37-40) (emphasis added).

Given this use of "list" in the claim and specifications, there is no reason to define that term for the jury. *See* Generation II Orthotics Inc. v. Medical Tech. Inc., 263 F.3d 1356, 1367 (Fed.Cir .2001) ("We can discern no sound basis on which to conclude that the word 'controlled,' as used in the claims, is in need of defining ...").

Oracle argues that "[t]he plain meaning of the term 'list' as that term is used in the specification and claims of the patents-in-suit, is an item-by-item series of numbers, words, or characters." (Instrument No. 77, at 23). Oracle provides no specific basis in the specifications or the claims to support its argument that the term "list" is restricted to an item-by-item series, or that the '226 Patent in any way implies such a construction. Oracle wholly relies on the testimony of its expert witness and its preferred definition in the IEEE Standard Dictionary of Electrical and Electronics Terms to support its argument. IEEE (INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS) STANDARD DICTIONARY OF ELECTRICAL

AND ELECTRONICS TERMS (Kim Breitfelder & Stephen Huffman eds.) (IEEE STANDARDS BOARD, 6th ed.1996) (hereinafter "IEEE DICTIONARY"). The Federal Circuit recently warned, however, that "extrinsic evidence consisting of expert reports and testimony is generated at the time of and for the purpose of litigation and thus can suffer from bias that is not present in intrinsic evidence," and that "undue reliance on extrinsic evidence poses the risk that it will be used to change the meaning of claims in derogation of the indisputable public records consisting of the claims, the specification and the prosecution history." Phillips, 415 F.3d 1303, 2005 U.S.App. LEXIS 13954 at *40-41 (internal citations and quotations omitted). Because the term "list," as it is used in the specification and claims, may be understood by both a person of ordinary skill in the art and a layman, there is no need to provide an additional definition. Accordingly, the Court finds that the term "list" means " ***list.***"

C.

Disputed Term:	Patent No.	Superspeed's Construction:	Oracle's Construction:
cache driver	'244	a program that implements a cache	device driver that controls a cache

The term "cache driver" appears in claims 15 and 22 of the '244 Patent. The term does not appear in the contested claims of the '226 Patent or the '136 Patent. Claim 15 of the '244 Patent explains that the cache driver (i) is resident in computers; (ii) creates a cache in the memory of the computer in which it resides; (iii) includes executable remote messaging code; (iv) at least one of he cache drivers includes executable interception code; and (v) at least one of the cache drivers includes executable invalidate code. ('244 Patent, at 26:60-27:31). Claim 22 of the '244 Patent adds that the cache driver (vi) allocates to the cache, space in the memory of the computer. (Id. at 27:42-46).

Superspeed contends that the definition of "cache driver" means "a program that implements a cache." (Instrument No. 69, at 24-26). The parties have already agreed that the definition of cache means "a portion of system main memory (*e.g.,* RAM) used for temporary storage of data." (Instrument No. 69, at 9-10). Thus, it is not necessary to re-define cache within the definition of "cache driver." However, it is necessary to determine how the term "driver" in "cache driver" is used within the context of the claims and specifications of the patents-in-suit.

The specifications of the '244 Patent refer to the "cache driver" as "cache driver software." ('244 Patent, at 4:8-10; 5 :14-16) ("When the OpenVMS system(14) performs a read data I/O access to a disk(12)the **cache driver (10) software** intercepts the I/O") (emphasis added). The specifications also describe the "cache driver" as something that is loaded on to a computer or operating system, rather than a separate physical device that is inserted or attached to the computer. (*See id*. at 3:18 ("When the cache driver is first loaded on the operating system all the disks (12) present on the computer system are located ..."); *Id*. at 3:36-38 ("The cache driver (10) maintains remote message communication channels (18) with other cache drivers loaded on other computers")). Thus, the "cache driver" is accurately described as software or a program that is loaded on a computer..

The specifications describe the function of the "cache driver" software as follows:

The **cache driver** (10) maintains remote message communication channels (18) with other **cache drivers** loaded on other computers that can access a common set of disks (12). Whenever the Open VMS system

(14) changes the data on the disk (12) for example by doing a write data access to the disk (12), the **cache driver** (10) uses its remote message communications channels (18) to send a message to each of the remote **cache drivers** in the list contained in the TCB [the **control** block] (16) disk **control** structure.

.....

When the Open VMS system (14) performs a read data I/O access to a disk (12) the **cache driver** (10) software intercepts the I/O. Using the size of the read data access the **cache driver** (10) selects which of the three caches, small, medium, or large, the data transfer fits. Having selected the appropriate sized cache the TCH [the cache hack] (26) cache **control** structure is selected. Using the read data I/O access disk block as a pointer into the disk block value hash table(30) of the TCH (26), the **cache driver** (10) attempts to locate a matching TCMB (24) bucket **control** structure.

.....

If a TCMB (24) bucket control structure with its corresponding cache data bucket (22) was obtained from one of the three sources described above, **cache data space can be assigned** for this disk (12) read data. The disk (12) is accessed normally, however the **read data is** not only sent to the requesting user on the Open VMS system (14), but also **copied to the cache data bucket.**
('244 Patent, at 3:34-42, 4:8-17, 4:52-58) (emphasis added).
According to the specifications, the "cache driver" controls the cache by communicating with the cache control structures and disk control structures. The "cache driver" also creates cache by assigning cache data to available cache data buckets and copying the data to the cache data bucket.

In order to assist the Court, both parties refer to the definition of "driver" as set forth in the IEEE DICTIONARY. The IEEE DICTIONARY defines "driver" in many contexts:

(1) (communication practice) An electronic circuit that supplies input to another electronic circuit;

(2) (A) (software) A software module that invokes and, perhaps, controls and monitors the execution of one or more other software modules (B) (software) A computer program that controls a peripheral device, and, sometimes, reformats data for transfer to and from the device. *See also:* test drives.

(3) A program, circuit or device used to power or control other programs, circuits or devices. *See also:* bus driver, device driver

(4) An electrical circuit whose purpose is to signal a binary state for transmitting information. Also referred to as a generator in international standards.

IEEE DICTIONARY, at 318. (Instrument No. 69, Exh F).

Here, the cache driver is software that creates or controls cache in the memory of the computer in which it resides and uses remote message communication to communicate with an I/O device or other cache driver programs and computers. Superspeed argues that the appropriate definition for "driver" in this context is "[a] software module that invokes and, perhaps, controls and monitors the execution of one or more other software modules." IEEE DICTIONARY, at 318. (Instrument No. 69, at 25-26). By linking the IEEE definition of "driver" to the agreed definition of "cache," Superspeed contends that "cache driver" may be defined as "software that invokes or controls the cache." (*Id.*). Superspeed further contends that this definition can be simplified by removing the terms "invokes or controls" and replacing it with the term

"implementing." (*Id*. at 26). However, the Court finds that the term "implementing" is overly vague in the context of the specifications that state that the cache driver communicates with structures that *control* the cache. Thus, the Court does not find that Superspeed's additional modification to include the term "implementing" is necessary.

Oracle argues that the appropriate definition of "driver" in this context is "[a] computer program that controls a peripheral device, and, sometimes, reformats data for transfer to and from the device." IEEE DICTIONARY, at 318. (Instrument No. 77, at 33). Oracle ultimately attempts to use this IEEE definition to support its proposed construction that "cache driver" means "device driver that controls a cache." The Court initially notes that Oracle's proposed definition of "cache driver" includes both of the terms that the definition purports to define. Furthermore, including the term "device driver" does nothing to clarify what a "driver" is-which is the entire basis of dispute.

In addition, Oracle entirely focuses on the cache driver communication with I/O devices, to support the conclusion that the cache driver operates on the device driver level. (*Id*. at 32). However, it is not necessary to limit the cache driver to communication on the device driver level, because the specifications indicate that the cache driver is not limited to communication with peripheral I/O devices. Rather, the cache driver uses remote message communications to communicate with I/O devices, as well as other operating systems and cache drivers. ('244 Patent, at 3:34-42, 4:8-17, 4:52-58) The cache driver also communicates with other internal software programs and disks, such as the TCH cache control structure, TCB disk control structure, and TCMB bucket control structure. (*Id*.).

Because the "cache driver" controls the cache by communicating with the cache control structures, and creates cache by copying read data to the cache data buckets, the Court finds that the term "cache driver" means " ***a software program that creates or controls a cache.***"

D.

Disputed Term:	Patent No.	Superspeed's Construction:	Oracle's Construction:
write instruction	'226, '244	an access or operation that initiates a transfer of information to storage	command that transfers data

The term "write instruction" appears in claim 27 of the '226 Patent and claim 15 of the '244 Patent; it does not appear in the contested claim of the '136 Patent. Claim 27 of the '226 Patent describes "[a] method for accelerating access to data on a network comprising the steps of ... intercepting a **write instruction** to one of said plurality of I/O devices from said computer." ('226 Patent, at 28:24-25) (emphasis added). Claim 15 of the '244 Patent describes a caching system "wherein at least one of said cache drivers further includes executable interception code for intercepting a **write instruction** to one of said plurality of I/O devices ..." ('244 Patent, at 27:5-7) (emphasis added).

The specifications in the patents-in-suit do not use the term "write instruction." Only the claims to the patents-in-suit use this term. However, the specifications refer to several write instructions that are intercepted by the cache driver software for the disk I/O device. For example, the specifications refer to the cache driver as intercepting a write data I/O access, write I/O data function, or write I/O data transfer. Specifically, the specifications state as follows:

When the Open VMS system (14) performs a **write data I/O access** to a disk (12) the cache driver (10) software **intercepts the I/O.** The cache driver (10) will search for possible matching TCMB (24) bucket control structures with their corresponding cache data buckets (22) in all three TCH (26) cache control structures, for the disk and the range of disk blocks in the write data I/O access. Using the write data I/O access disk block as a pointer into the disk block value hash table (30) of each of the three TCH's (26), the cache driver (10) attempts to locate matching TCMB (24) bucket control structures. For each TCMB (24) bucket control structure found, the TCMB (24) and its corresponding cache data bucket (22) are invalidated. ('226 Patent, at 5:13-25) (emphasis added).

.....

If the OpenVMS I/O function is 'io_writelblk' (write logical blocks of disk I/O data), or 'io_writepblk' (write physical blocks of disk I/O data) or io_dse' (write data security erase pattern) (437), the program dispatches to the "write data" (572, FIG.5K) program flow.
Referring to FIG. 5K, the "write data" (572) program flow ... checks that the byte count for the **intercepted write I/O data function** is a non-zero positive value (574).... The program records the positive byte count of the **intercepted write I/O data function** in the TCB (16, FIG.1) disk control structure for the disk I/O device.

The program checks whether the intercepted disk I/O device is currently subject to mount verification on the OpenVMS system (582), indicating that the OpenVMS system is checking the integrity of the volume mounted in the disk I/O device. If so, the program exists via the "I/O function exit" (564, FIG.5J) program flow, allowing the write I/O data to go directly to the disk I/O device. (*Id*. at 17:37-42; 21:53-22:3) (emphasis added)

.....

Referring to FIG. 5L, the "cache data invalidate" program invalidates the cached data blocks in all three caches, small, medium, and large, that match the disk block range in this **intercepted write I/O data transfer** for the disk I/O device.... If this TCMB is associated with the disk I/O device in the intercepted write I/O data transfer, the program checks whether the disk block range in the TCMB falls anywhere within the disk block range in the TCMB falls anywhere within the range of disk blocks in the **intercepted write I/O data transfer.** (*Id*. at 22:23-26, 46-50) (emphasis added).
In broader terms, the specifications refer to the cache driver software as intercepting an I/O operation. Specifically, the specifications provide that "[w]henever any I/O operation is performed on a disk I/O device, that **I/O operation will be intercepted** by the cache software of the invention and the **program will commence running** at the 'process io' (400) entry point." (*Id*. at 15:57-61) (emphasis added). The specifications refer to Figures 5A-50 of the patents-in-suit, when describing the I/O operations and the "program flow performed by the active data caching of a disk I/O device in the cache software of the invention." (*Id*. 15:55-57). Figures 5A-50 describe the "process io" entry point (FIG.5A), "cache on" program flow (FIG.5B), "read data" program flow (FIG.5C), "read cache miss" program flow (FIG.5F), "read complete" program entry point (FIG.5H), "read cache hit" program flow (FIG.51), "I/O function exit" program flow (FIG.5J), "write data" program flow (FIG.5K), "cache data invalidate" program flow (FIG.5L), "write invalidate" program flow (FIG.5M), "message receive" entry point, "remote invalidate" program flow (FIG.5N), and "basic statistics" program flow (FIG.50). ('226 Patent, at 15:55-24:48).

Each of the various program flows commence running upon intercepting an I/O operation (*i.e.*, read I/O data function, read I/O data transfer, write I/O data function, or write I/O data transfer). Thus, the term "I/O

operation" is not limited to a write instruction, rather the term "I/O operation" encompasses read instructions as well. Because the specifications use the term "operation" instead of "instruction," the term "write instruction" is accurately described by the broader term "operation."

The IEEE DICTIONARY also confirms that a "write instruction" is accurately described as an "operation." The IEEE DICTIONARY does not define the term "write instruction." However, the IEEE DICTIONARY defines "instruction" as "[a] statement or expression consisting of an operation and its operands which can be interpreted by a computer in order to perform some function or operation." IEEE DICTIONARY at 528. (Instrument No. 69, at 22; Instrument No. 77, at 28) (emphasis added). The definition of "write" is "to record data in a storage device or on a data medium." IEEE DICTIONARY at 1210. (*Id.*). Thus, the IEEE DICTIONARY is consistent with the specifications of the patents-in-suit, which indicate that a "write instruction" is an "operation." The IEEE DICTIONARY is also consistent with the interpretation that the "write instruction" is used to transfer data to storage. However, the IEEE DICTIONARY does not describe the process by which a particular function (*i.e.,* transferring data to storage) is to be completed.

In this case, the specifications provide that the "**I/O operation will be intercepted** by the cache software of the invention and the **program will commence running** at the 'process io.' " ('266 Patent, at 15:57-61) (emphasis added). The specifications make clear that the program goes through a step-by-step process of checking different specifications before allowing the transfer or invalidation of data. (*Id.* at 21:53-23:59). The specifications further indicate that the program flow uses information contained in the "write instruction" to assign different values, ranges, and sizes for the transfer of data. (*Id.*). Thus, the "write instruction" or "operation" does not immediately transfer the data to storage, it merely provides the necessary information to commence or initiate a process that eventually allows the transfer of data. Accordingly, the Court finds that an accurate construction of the term "write instruction" is " ***an operation that initiates a transfer of data to storage.***"

E.

Disputed Term:	Patent No.	Superspeed's Construction:	Oracle's Construction:
"invalidate data"	'226,'244	to indicate that a portion of data in a cache is no longer up to date	make data unusable

Claims 27 and 35 of the '226 Patent and claims 15 and 22 of the '244 Patent reference directly or indirectly the term "invalidate data." ('226 Patent, at 28:28-29; '244 Patent, at 27:9). Superspeed asserts that the term "invalidate data" means "to indicate that a portion of data in a cache is no longer up to date." (Instrument No. 69, at 11). Oracle asserts that the term "invalidate data" means "make data unusable." (Instrument No. 77, at 8).

Claim 27 of the '226 Patent describes "[a] method for accelerating access to data on a network comprising the steps of ... communicating over the network individually with each computer in the list of computers in the data structure corresponding to said one of said I/O devices to **invalidate data** in caches on the network corresponding to said one of said plurality ." ('226 Patent, at 28:26-30) (emphasis added). Claim 15 of the '244 Patent describes a caching system "wherein at least one of said cache drivers further includes executable interception code for intercepting a write instruction to one of said plurality of I/O devices and executable invalidate code that sends a message to **invalidate data** through selected ones of the communication channels to the caches of any computer that is caching said one of said plurality of I/O

devices." ('244 Patent, at 27:5-11) (emphasis added).

The claims and specifications of the patent do not expressly define the term "invalidate data." The specifications of the preferred embodiment provide, however, that the cache driver sends a message to invalidate data, whenever the operating system changes data on the disk. ('226 Patent, at 3:37-50; '244 Patent, at 3:39-52). Specifically, the specifications provide:

Whenever the Open VMS [operating] system (14) changes the data on the disk ... the cache driver (10) uses its remote message communication channels (18) to send a message to each of the remote cache drivers in the list contained in the TCB (16) disc control structure.... The cache driver (10) would use this incoming message to **invalidate any possible previously locally cached data** for the area on the remotely connected disk (12) that has been changed by the remote OpenVMS system.

(*Id*.) (emphasis added). This specification indicates that the invention sends a message to "invalidate data" when there is old cached data that corresponds to a change in that data. Thus, the invalidated data corresponds to data that has been changed or modified.

The specifications further indicate that the invalidated cache data is not automatically deleted or removed from the system as a result of being invalidated. Rather, "[t]he invalidated TCMB (24) [cache data] and its cache data bucket (22) are normally placed on the free queue (27) of the associated TCH (26) cache control structure to be used by some future cache data operation." ('226 Patent, at 5:23-28; '244 Patent, at 5:26-30). In other words, the invalidated data is typically moved to another location to be written over at some point in the future. Although nothing in the specifications suggests that the invalidated data is cleared or erased as a result of an invalidate message, nothing in the specifications suggests that the invalidated data may be used again.

However, "if the OpenVMS system (14) indicates there are insufficient available free pages for the OpenVMS system (14), the cache data bucket (22) RAM space is returned to the OpenVMS system (14) free pages and the corresponding TCMB (24) space is returned to the OpenVMS system (14) pool." ('226 Patent, at 5:28-34; '244 Patent, at 5:30-35). The specifications do not specifically state what happens to the invalidated cache data once it is returned to the OpenVMS system pool.

The "Summary of the Invention" in the '226 Patent and the '244 Patent provide two possible ways that an "invalidate data" message may be used in the invention:

In accordance with the embodiment of the invention, once the total cache size has grown to its upper limit further new demands on cache data are handled by cache bucket replacement, which operate on a least recently used algorithm. This cache bucket replacement will also occur if the total cache size is inhibited from growing owing to a high demand on computer main memory by other applications and users of the computer system.

.....

In accordance with the embodiment of the invention, when a write access is performed to a disk which is being cached and the disk data area being written was previously read into the cache, i.e., an update operation on the disk data, the current cache buckets for the previous read disk data area are invalidated on all computers on the network.

('226 Patent, at 2:13-20, 2:45-50; '244 Patent, at 2:15-22, 2:46-52). Thus, the summary of the invention and specifications are consistent with a claim construction of "invalidate data" that indicates a change or modification of previous data with more recent data.
In order to assist the Court, Superspeed has also provided a definition of the term "invalid" from the IEEE DICTIONARY. The IEEE DICTIONARY provides that the term "invalid" is "[a]n attribute assigned to a cache line if there is not an up-to-date copy in the module's cache." IEEE DICTIONARY at 528. (Instrument No. 69, at Exh F). The IEEE DICTIONARY does not provide a definition of the term invalidate. However, the attributes described for the term "invalid" are consistent with the use of the term "invalidate data" in the specification of the patents-in-suit.

Upon a review of the patents-in-suit and the arguments of the parties, the Court finds that the term "invalidate data" means " ***to indicate previously cached data has been modified.***"

F.

Disputed Term;	Patent No.	Superspeed's Construction:	Oracle's Construction:
disabling cache operations	'136	suspending caching	prohibiting use of data in the cache
enabling caching operations	'136	resuming caching	permitting use of data in the cache

The terms "disabling cache operations" and "enabling caching operations" are found in Claim 1 of the '136 Patent. The terms do not appear in the contested claims of the '226 or '244 Patents. Claim 1 of the '136 Patent describes "[a] method for coherently caching I/O devices available for shared access on a network comprising ... **disabling cache operations** upon finding that a new computer joined the network; and **enabling caching operations** at each computer after each computer has connections in place with the cache software of every other computer on the network." ('136 Patent, at 26:17-22) (emphasis added). The parties agree that "enabling caching operations" is the opposite of "disabling cache operations" and that the two terms should be construed together. (Instrument No. 69, at 28; Instrument No. 77, at 15). There is also no dispute that "cache operations" and "caching operations" are the same. (*Id*.).

The parties also appear to agree that the term "disable" means "prohibit," and the term "enable" means "permit."(Instrument No. 69, at 29; Instrument No. 77, at 15). In order to assist the Court, both parties refer to the IEEE DICTIONARY definition of "disable" and "enable." (*Id*.). The IEEE DICTIONARY defines "disable" as "[a] condition or command that **prohibits** some specific event from proceeding." IEEE DICTIONARY at 299. (emphasis added). As a corollary, the IEEE DICTIONARY defines "enable" as "[a] command or condition that **permits** some specific event to occur [or] to proceed." (Instrument No. 69, at 29) (emphasis added). In this case the specific event is "cache operations" or "caching operations." The parties have agreed that the term "cache" means "a portion of system main memory (*e .g.,* RAM) used for temporary storage of data." (Instrument No. 69, at 9-10). The parties have further agreed that the term "caching" means "storing in cache." (*Id*.). The parties dispute whether the term "caching" is the same as "cache operations," or whether "caching" is just one of several different "cache operations." The parties further dispute whether the duration of the prohibited or permitted cache operations should be reflected in the construction of the terms "disabling cache operations" and "enabling caching operations."

With regard to the duration of prohibited or permitted cache operations, Claim 1 of the '136 Patent implies

that the disabling of cache operations is only temporary, because cache operations are only disabled "upon finding that a new computer joined the network." ('136 Patent, at 26:18-19). "[A]fter each computer has connections in place with the cache software of every other computer on the network," cache operations are enabled again. (*Id*. at 26:20-22). The specifications of the '136 Patent further confirm the temporary nature of disabling cache operations, and explain that "the 'disable' flag is used to indicate that the remote computer connections are inconsistent, which will **temporarily** disable caching operations until the remote computer connections are completely formed in a consistent state." (Id. at 7:10-11; 8:52-53) (emphasis added).

Superspeed argues that "while the '136 claim 1 does not explicitly require 'temporarily disabling' it does explicitly contemplate that the disabled operations will be subsequently enabled." (Instrument No. 69, at 29). Thus, Superspeed argues that the term "disabling cache operations" means "temporarily prohibiting (*i.e*., suspending) cache operations" and "enabling caching operations" is "continuing or resuming those caching operations." (Id.). Although the terms "suspending" and "resuming" may constitute a convenient short-form to reflect the temporary nature of prohibiting or permitting cache operations, the Court finds that it is unnecessary to use these specific terms in the construction. Because the specifications specifically provide for operations to be "temporarily disabled," the Court finds that the term "temporarily prohibiting" cache operations is a preferable construction. Because the specifications do not specifically state that "enabling caching operations" is temporary in nature, the Court finds that the term "permitting" caching operations is a preferable construction. However, the Court must also determine what specific cache operations are temporarily prohibited or permitted.

Superspeed contends that the specifications and flow chart figures of the '136 Patent indicate that "if cache operations are disabled ... data cannot be read from the cache nor placed into the cache from the disk." (Instrument No. 69, at 31). Specifically, the specifications provide:

Referring to FIG. 5C, the "read data" (440) program flow will now be described.... The program checks whether the cache status flag disable' is set (452), if so, the program exits ... The cache status flag 'disable' indicates that some OpenVMS system in the VMScluster ... does not have the cache driver ... of the invention loaded. This normally would indicate that some OpenVMS system is currently joining the VMScluster ... and has not yet successfully loaded the cache software of the invention.

[T]he cache status flag 'disable' indicates an inconsistent view of the cache for the invention across the VMScluster and VAXcluster, **preventing active cache operations** (and possible subsequent corruption) **of the data contained in a disk I/O device.**"

(Instrument No. 69, Exh C, at 17:61-18:9) (emphasis added). Figure 5C of the '136 Patent further indicates that the "read data" program flow checks whether the cache status flag "disable" is set. (Id. at FIG. 5C).

The specifications do not specifically define the term "active cache operations." However, the parties appear to agree that the active cache operations that are disabled include: the reading of data from the cache, and the storing of data into the cache. Superspeed admits that "placing data into the cache for rapid access and then accessing it from the processor" are operations that are prohibited while caching operations are disabled. (Instrument No. 69, at 31). Oracle agrees that the "plain meaning of 'cache operations' exceeds just storing data in a cache and includes reading data from a cache (a prerequisite to using the cached data), and writing data to the cache." (Instrument No. 77, at 16). According to the demonstrative exhibit that Oracle presented in oral argument, Oracle uses the term "writing data" from the disk to cache synonymously with "caching" or "storing data" in cache. (DX 1, at 60). Accordingly, the parties agree that disabling cache

operations prohibits reading data from the cache, and storing data in a cache.

However, disabling cache operations does not prohibit all cache operations. Superspeed argues that even while caching operations are disabled, some operations on the cache may still be undertaken. Specifically, Superspeed contends that "the software continues to check the free memory and can release cache if the memory used in the cache is needed elsewhere." (Instrument No. 69, at 30). This allows the program to modify the size of the cache for small, medium, or large cache data buckets. (Instrument No. 69, Exh C, at 8:61-65). Superspeed further contends that the system can "unload" or clear cache data in the system when cache operations are "disabled." (Instrument No. 69, at 30; Exh C, at 17:31-36, FIG. 5B). Finally, Superspeed contends that the software can invalidate cache data when writing to an I/O device. (Id.; Exh C, at 17:40-45, 21:58-22:27, FIG. 5K). Oracle does not dispute that the program can modify the size of cache, clear the cache, and invalidate cache data while caching operations are disabled. (Instrument No. 77, at 20). This interpretation appears to be consistent with the specifications and flow chart figures described in the '136 Patent.

Because "disabling cache operations" does not prohibit the use of all cache operations, a construction that the term "disabling cache operations" means "prohibiting use of data in the cache" would be too broad. However, a construction of the term "disabling cache operations" to mean "suspending caching" does not indicate that reading data from the cache is also prohibited, and would thus be too narrow. Taking into consideration the duration of prohibited cache operations and the specific cache operations that are prohibited by "disabling cache operations," the Court finds that the term "disabling cache operations" means " ***temporarily prohibiting the storing and reading of data on the cache,***" and that the term "enabling caching operations" means " ***permitting the storing and reading of data on the cache.***"

IV.

IT IS HEREBY ORDERED that the disputed terms shall have the following claim constructions:

DISPUTED TERM	PATENT NUMBER	CLAIM CONSTRUCTION
Input/Output device or I/O device	'226, '244, '136	disk or other persistent storage device
list	'226	list
cache driver	'244	a software program that creates or controls a cache
write instruction	'226	an operation that initiates a transfer of data to storage
invalidate data	'226, '244	to indicate previously cached data has been modified
disabling cache operations	'136	temporarily prohibiting the storing or reading of data on the cache
enabling caching operations	'136	permitting the storing or reading of data on the cache

The Clerk shall enter this Order and provide a copy to all parties.

S.D.Tex.,2005.

Superspeed Software, Inc. v. Oracle Corp.

Produced by Sans Paper, LLC.