UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF ILLINOIS
EASTERN DIVISION

CLEVERSAFE, INC.,	)	
	)	
Plaintiff,	)	**11 C 4890**
	)	
v.	)	**Judge John Z. Lee**
	)	
AMPLIDATA, INC.,	)	
	)	
Defendant.	)	

MEMORANDUM OPINION AND ORDER

Cleversafe, Inc. ("Cleversafe") has sued Amplidata, Inc. ("Amplidata") alleging infringement of three patents: U.S. Patent Nos. 7,953,771 ("the '771 patent"), 7,953,937 ("the '937 patent"), and 7,546,427 ("the '427 patent"). The case is before the Court for the construction of nine terms.[1]

Background

The three patents-in-suit relate to aspects of distributed data storage systems. (Def.'s Opening Claim Construction Br. 1 ("Def.'s Br.").) In such a system, data is stored across multiple "nodes" rather than a single location. (*Id.*)

On September 30, 2005, Cleversafe filed U.S. App. No. 11/241,555 disclosing a system and method for storing data on a distributed data storage system. ('937 Patent, at [57], Joint Appendix ("J.A.") 703.) The method separates data into "slices" or "subsets" that are less usable than the original data unless combined with other subsets. (*Id.* at col.2 ll.54-60, J.A. 712.) The subsets are encoded using a coding algorithm, and the subsets and coded subsets are distributed

[1] Parties originally disputed twelve terms, but during the December 12, 2012 *Markman* hearing, agreed to the construction of three terms. The parties agree that: 1) "slice server" means "a server that stores information and responds to requests to retrieve data slices"; 2) "file" means "a collection of data"; and 3) "list" means "a series of one or more entries." (*Markman* Hr'g Tr. 54, 71.)

over a network of storage nodes to increase security. (*Id.* at col.2 ll.60-67, J.A. 712.) The information can be recreated by retrieving the subsets and coded subsets and applying a decoding algorithm. (*Id.* at col.3 ll.1-6, J.A. 713.) The system is "computationally efficient compared to known systems." (*Id.* at col.3 ll.6-11, J.A. 713.) On May 31, 2011, the application issued as U.S. Patent No. 7,953,937. (*Id.* at [10], [45], J.A. 703.)

On April 13, 2006, Cleversafe filed U.S. App. No. 11/403,391 as a continuation in part of U.S. App. No. 11/241,555. ('427 Patent at [63], J.A. 1.). This application disclosed a system and method for rebuilding data previously stored on a distributed data storage network when one or more nodes becomes unavailable by applying an algorithm to the available data slices. (*Id.* at col.3 ll.16-29, J.A. 21.) On June 9, 2009, the application issued as U.S. Patent No. 7,546,427. (*Id.* at [10], [45], J.A. 1.)

On December 8, 2009, Cleversafe filed U.S. App. No. 12/633,779. ('771 Patent at [22], J.A. 562.) This application disclosed a distributed data storage network in which virtual "vaults" organize and control access to data. (*Id.* at [57], J.A. 562.) On May 31, 2011, the application issued as U.S. Patent No. 7,953,771. (*Id.* at [10], [45], J.A. 562.)

The following terms are in dispute: (1) data slice and plurality of data slices; (2) virtual digital data storage vault; (3) encode, using a coding algorithm, a plurality of subsets of data to create a plurality of coded values; (4) information dispersal algorithm; (5) strings of data; (6) subsets of data; (7) list of unusable storage nodes; (8) the request is valid/the request is invalid; and (9) supports.

Legal Standard

Claim construction is a question of law to be decided by a judge. *Markman v. Westview Instruments, Inc.*, 517 U.S. 370, 391 (1996). The Court begins its analysis with the intrinsic

evidence, which consists of the patent claims, specification, and prosecution history. *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996). Words "are generally given their ordinary and customary meaning." *Id.* "[T]he ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention . . . ." *Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005).

Terms are given "the meaning and scope with which they are used in the specification and the prosecution history." *Kinik Co. v. ITC*, 362 F.3d 1359, 1365 (Fed. Cir. 2004). The specification is usually "dispositive; it is the single best guide to the meaning of a disputed term." *Vitronics*, 90 F.3d at 1582. However, a particular embodiment used in the specification to aid understanding should not import limitations into the claim. *Superglide Corp. v. DirecTV Enters., Inc.*, 358 F.3d 870, 875 (Fed. Cir. 2004). Nonetheless, a claim may be limited to its preferred embodiment if permitting expansive claim language would undermine the public notice requirements of 35 U.S.C. § 112. *LizardTech, Inc. v. Earth Resource Mapping, Inc.*, 424 F.3d 1336, 1346 (Fed. Cir. 2005).

The prosecution history may serve to further "exclude any interpretation that was disclaimed during prosecution." *Chimie v. PPG Indus., Inc.*, 402 F.3d 1371, 1384 (Fed. Cir. 2005). However, a claim may not be narrowed "simply by pointing to the preferred embodiment or other structures or steps disclosed in the specification or prosecution history." *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002).

Extrinsic evidence, such as dictionaries and expert testimony, may be used only if the intrinsic evidence alone is insufficient to determine the meaning of the claim terms. *Vitronics*, 90 F.3d at 1583.

Finally, the doctrine of "claim differentiation" provides that "each claim in a patent is presumptively different in scope." *RF Del., Inc. v. Pac. Keystone Techs.*, Inc., 326 F.3d 1255, 1263 (Fed. Cir. 2003). "That presumption is especially strong when the limitation in dispute is the only meaningful difference between an independent and dependent claim." *Acumed LLC v. Stryker Corp.*, 483 F.3d 800, 806 (Fed. Cir. 2007). However, claim differentiation is a "rule of thumb" and not absolute. *Edwards Lifesciences LLC v. Cook, Inc.*, 582 F.3d 1322, 1332 (Fed. Cir. 2009). The scope of a claim may be limited so as to create redundant claims if required by "the clear import of the specification," *id.*, or by the prosecution history. *See Fantasy Sports Properties, Inc. v. Sportsline.com, Inc.*, 287 F.3d 1108, 1116 (Fed. Cir. 2002) (presumption of claim differentiation overcome by disclaimer during prosecution history).

Discussion

I. "Data slice" and "plurality of data slices"

The parties dispute the meaning of the phrases "data slice" and "plurality of data slices" used in claims 1, 2, 4, 5, 7, 9, 10, 12, and 13 of the '937 patent and claims 3 through 7 of the '427 patent. (Pl.'s Resp. Claim Construction Br. 9 n.6 ("Pl.'s Br.").) Amplidata argues that a plurality of data slices is "a plurality of data structures each consisting of a data subset and a corresponding coded data subset." (Def.'s Br. 4.) Cleversafe proposes an alternative construction of "a collection of subsets of data and coded values." (Pl.'s Br. 9.) The disagreement lies in whether a data slice must contain both at least one data subset and at least one coded value and, if so, whether the data subset(s) and coded value(s) must correspond with one another.

As to the first question, on October 7, 2009, Cleversafe significantly amended the application that would eventually become the '937 patent. In its remarks accompanying the

amendments, Cleversafe distinguished what would become claim 1 of the '937 patent from the prior art, Moulton, stating:

> Moulton does not teach or suggest creating *n* data slices from *n* subsets of data and *n* coded values. Moulton also does not teach or suggest outputting the data subsets and the code values as a single data element (e.g., a data slice), but does teach that the data units may be separately stored from the parity value or that only the parity values be stored.

(J.A. 1030.) The quoted passage supports the construction that a data slice must contain at least one data subset and at least one coded value. What is more, once the term "data slice" is construed to contain at least one data subset and at least one coded value, the statement "*n* data slices from *n* subsets of data and *n* coded values" leads to the conclusion that a 1:1:1 ratio must exist between a data slice, a data subset, and a coded value.

Such a construction is entirely consistent with the patent's specification. For example, the "Summary of the Invention" portion of the specification describes a process by which a digital data storage system slices "the original data into data subsets," after which a "coding algorithm uses the subsets to create coded data subsets." ('937 Patent at c.2 ll.54-60.) Once the data subsets and their coded data subsets are created, "[e]ach data subset and its corresponding coded subset may be transmitted separately across a communications network and/or stored in separate storage nodes." (*Id.* at c.2 ll.64-67.) In this context, the word "each" can only modify a combination of a data subset *and* a coded data subset, which combinations are separately transmitted across the network. Otherwise, to read the sentence to mean "each data subset" and "each coded data subset" would render the word "separately" superfluous. *See Callicrate v. Wadsworth Mfg., Inc.*, 427 F.3d 1361, 1369 (Fed. Cir. 2005) (court's construction must fully reflect the language in the patent).

Furthermore, although Cleversafe would have "data slices" mean "subsets of data, coded values, or both," such a construction is not only inconsistent with the prosecution history and the

specification as discussed above, but the patent itself teaches a system that, among other things, creates "a plurality of data slices from a plurality of subsets of data *and* the plurality of coded values." ('937 Patent at c.9 ll.23-24.) In other words, to create data slices, one needs both data subsets *and* coded values. If the patent contemplated that a plurality of data slices could be created from a plurality of subsets of data *and/or* the plurality of coded values, it certainly could have stated so.

For its part, Cleversafe also argues that the doctrine of claim differentiation requires a broader reading of the term "data slice." (Pl.'s Br. 9-10.) Claim 3 of the '427 patent is an independent claim, while claim 4 depends on claim 3. Claim 4 provides: "The method of claim 3, wherein each data slice comprises a data subset and a coded data subset." ('427 Patent col.16 ll.29-29, J.A. 27.) Thus, according to Cleversafe, the doctrine of claim differentiation dictates that the term "data slice" must be broader than something that "comprises a data subset and a coded data subset." Claim differentiation, however, is not an insurmountable doctrine, but a presumption that can be overcome. *See Fantasy Sports*, 287 F.3d at 1115. Here, the clear and unambiguous statements made by Cleversafe during the prosecution history, coupled with the statements in the patent's specification, are sufficient to overcome that presumption. *See id.* at 1116.

The remaining question is whether the data subset and coded value that are contained in a data slice must correspond with one another. In arguing the affirmative, Amplidata points to several instances in the '937 and '427 patents in which data subsets are paired with their corresponding coded subsets. For example, Amplidata cites Figure 4 of the '937 patent, which shows data subsets A, B, and E paired with "their corresponding coded values" cA, cB, and cE.

(Def.'s Br. 5.) However, the exemplary embodiments referenced by Amplidata may aid in understanding the invention, but they do not operate to limit the scope of the claims.

During the *Markman* hearing, though not in its briefs, Amplidata also argued that, during prosecution of the '937 patent, Cleversafe had limited the definition of data slices in this precise manner. (*Markman* Hr'g Tr. 28.) According to Amplidata, between February 6, 2008 and May 4, 2009, Cleversafe repeatedly tried to distinguish its application from prior art by stating the prior art "[did] not teach grouping a data unit with a corresponding encoded data unit, and storing each grouping on a separate storage node." (J.A. 885, 906, 931, 967, 989.)

But when Cleversafe amended the application in October 2009, it cancelled the previous claims and removed the phrase "corresponding encoded data unit" from the claims. (J.A. 1021.) In its remarks accompanying the amendments, Cleversafe then distinguished what would become claim 1 of the '937 patent from Moulton, in this manner:

> Moulton does not teach or suggest using a coding algorithm to generate a code value for each data subset for a given string of data, but does teach creating a parity value by exclusive ORing multiple data units. Thus, Moulton does not teach or suggest a one-to-one correlation of code values to data subsets in a string of data.

(J.A. 1030.) In so doing, Cleversafe relied upon a "one-to-one correlation" between data subsets and coded values, rather than a requirement that a data subset be combined strictly with its "corresponding" coded value.[2] Accordingly, the Court finds that the data subset and the coded value that comprise a data slice need not correspond with one

[2] Such broadening of claims during the prosecution of a patent is permissible. *See Liebel-Flarshein Co. v. Medrad*, Inc., 358 F.3d 898 (Fed. Cir. 2004). There, the plaintiff owned several patents for medical devices to inject fluids. *Id.* at 901. The original patent description required the syringes to have "pressure jackets," which prevented the syringes from breaking. *Id.* During prosecution the plaintiff removed the pressure jacket limitation, and thus broadened the coverage of the patents. The Court supported this broadening, stating that "[t]he omission of reference to a pressure jacket . . . is a strong indication that the applicants intended those claims to reach injectors that did not use pressure jackets." *Id.* at 909. It did not matter whether the broadening ultimately invalidated the patents, because "any question regarding the support or lack of support for the claims in the original disclosure bears on the issues of priority and validity, not on the issue of claim construction." *Id.* at 903.

another. In summary, the Court will construe a "data slice" to mean a "data structure consisting of a data subset and a coded value."

II. "Virtual digital data storage vault"

The claim term "virtual digital data storage vault" appears in claims 1 and 6 of the '771 patent. Cleversafe and Amplidata made some progress defining the term at the *Markman* hearing, but still dispute its meaning. Originally, Amplidata read the term to mean "a software construct associated with particular slice servers and user accounts," while Cleversafe offered the construction "a logical container physically stored over one or more slice servers." (Def.'s Br. 7; Pl.'s Br. 13.) The parties now agree that a vault is a "software container" and that it is "stored over more than one slice server." (*Markman* Hr'g Tr. 59-60.) However, Cleversafe contests Amplidata's assertion that vaults must be "associated" with particular user accounts. (*Id.* at 62.)

Amplidata references several instances in the specification that appear to indicate that vaults are associated with particular user accounts. (Def.'s Br. 9.) However, Amplidata simply points to preferred embodiments, and nothing in the specifications suggests the vaults must be associated with particular user accounts. Quite to the contrary, the detailed description of the '771 patent provides ". . . a vault *may* also include an access control list specifying which accounts are allowed to access the vault, and what permission are associated with that account." ('771 Patent col.4 ll.25-28, J.A. 568 (emphasis added).) Though a vault may be associated with particular user accounts, it is not required by claim 1 of the '771 patent, which is an independent claim. Dependent claims, such as claim 3, do impose the limitation that a vault be associated with "user account information." ('771 Patent col.5 l.59, J.A. 569.) However, nothing in claim 1 suggests that such association is required. Therefore, this Court construes "virtual digital data storage vault" to mean a "software construct stored over multiple slice servers."

III. "Encode, using a coding algorithm, a plurality of subsets of data to create a plurality of coded values"

The phrase "encode, using a coding algorithm, a plurality of subsets of data to create a plurality of coded values" appears in claims 1, 7, and 9 of the '937 patent. Amplidata proposes that the phrase be defined to mean "encode, using the algorithm described at lines 4:5-5:42, *n* subsets of data to create *n* coded values." (Def.'s Br. 13.) Cleversafe counters that the term is fully defined and no construction is required. (Pl.'s Br. 17.)

Amplidata bases its limited construction on references to prior art found in the Background of the Invention. (Def.'s Br. 13.)

> In order to make sure that stored data is only available to authorized users, data is often stored in an encrypted form . . . . In addition to securing data using encryption, several methods for improving the security of data storage using information dispersal algorithms have been developed. . . . Such information dispersal algorithms are used to slice the original data into multiple data subsets and distribute these subsets to different storage nodes . . . .

('937 Patent col.2 ll.9-26, J.A. 712.) The '937 patent differentiates itself from the prior art by claiming superior computational efficiency:

> Unfortunately, these methods and other known information dispersal methods are computationally intensive and are thus not applicable for general storage of large amounts of data using the kind of computers in broad use by businesses, consumers and other organizations today. Thus there is a need for a data storage system that is able to reliably and securely protect data that does not require the use of computation intensive algorithms.

('937 Patent col.2 ll.43-50, J.A. 712.)

Amplidata interprets the "need for a data storage system . . . that does not require the use of computation intensive algorithms" to mean that the algorithm itself must not be computation intensive. (Def.'s Br. 14.) This interpretation would make sense if the specification identified

"the need for a data storage that uses an algorithm that is not computation intensive," but this is not the case.

Looking to the patent itself, the only coding algorithm that appears in the Detailed Description takes the following general form: $c[x] = d[n\_\mathrm{mod}(x+1)] + d[n\_\mathrm{mod}(x+2)] + d[n\_\mathrm{mod}(x+4)]$. ('937 Patent col.4 ll.61-62, J.A. 713.) However, the patent does not state that this is the only form of coding algorithm encompassed. Quite to the contrary, the Detailed Description explicitly states that this algorithm is only one embodiment. (*Id.* col.4 ll.56-59, J.A. 713.)

Nevertheless, Amplidata argues that if the term "coding algorithm" means any coding algorithm, then the invention would encompass computation intensive algorithms, thus violating the stated goal of computational simplicity. (Def.'s Br. 14.) Amplidata further claims support for its interpretation from *LizardTech v. Earth Resource Mapping, Inc.*, which established that "description of one method . . . does not entitle the inventor . . . to claim any and all means for achieving that objective." 424 F.3d 1336, 1346 (Fed. Cir. 2005).

In *LizardTech*, the plaintiff owned a patent relating to the compression of digital images, including a claim for creating a "seamless" discrete wavelet transform ("DWT"). *Id.* at 1337. Other compression methods created errors within the images, but the patent at issue claimed a "seamless" DWT, thereby improving upon the prior art. *Id.* at 1344. The patent included one embodiment of the method, but the claim at issue referred to "taking a seamless DWT generically." *Id.* The Federal Circuit invalidated this claim for failure to meet the written description requirement of 35 U.S.C. § 112 because "a person of skill in the art would not understand how to make a seamless DWT generically and would not understand LizardTech to

have invented a method for making a seamless DWT, except by [the sole disclosed embodiment]." *Id.* at 1345.

To illustrate its reasoning, the Federal Circuit offered a useful analogy:

> [S]uppose that an inventor created a particular fuel-efficient automobile engine and described the engine in such detail in the specification that a person of ordinary skill in the art would be able to build the engine. Although the specification would meet the requirements of section 112 with respect to a claim directed to that particular engine, it would not necessarily support a broad claim to every possible type of fuel-efficient engine, no matter how different in structure or operation from the inventor's engine.

Id. at 1346. Similarly, Amplidata's argument goes, Cleversafe claims to have invented a system that does not require "computation intensive algorithms," but it has only disclosed one such coding algorithm. (Def.'s Br. 13-14; '937 Patent col.2 ll.43-50, J.A. 712.)

If the '937 patent specified that it required a computational non-intensive algorithm, the reasoning in *LizardTech* might apply and the algorithm limited to the exemplary algorithm provided in the specification. But the invention makes no such claim. First, the claims themselves do not ever mention computationally efficient algorithms. Second, the invention is not simply a new algorithm. Rather, the '937 patent describes a *system* of storing data that, as a whole, allows for (but does not require) less computation intensive algorithms.

This becomes apparent after reviewing the prosecution history of the '937 patent. As discussed above, Cleversafe significantly amended its application on October 7, 2009. Prior to this date, Cleversafe had repeatedly argued that its invention overcame the prior art, Moulton, because "an encoded data unit corresponding to each data unit is created by, at minimum, summing the modulo of at least two data units within each subset." (J.A. 853.) On October 7, 2009, however, Cleversafe dropped this argument and removed any references to specific algorithms from the claims. (J.A. 1030.) Instead, Cleversafe differentiated Moulton by saying it

did not teach using a "coding module [that] also encodes the n subsets of data using a coding algorithm to create n coded values." (*Id.*) Thus, Cleversafe sought to broaden the scope of its claim beyond the previously specified algorithm, as permitted under *Liebel-Flarshein Co.*, 358 F.3d at 901.

Amplidata's preferred construction is also refuted by the doctrine of claim differentiation. Claim 6 of the '937 patent identifies "[the] computer of claim 1 wherein the coding algorithm further comprises a convolutional coding algorithm." (J.A. 716.) The existence of the term "convolutional coding algorithm" in a dependant claim strongly suggests that "coding algorithm" must be broader than "convolutional coding algorithm," further proving that Amplidata's limited construction is incorrect.[3] Therefore, the algorithm is not limited to algorithm described at lines 4:5-5:42, as Amplidata urges.

That said, Amplidata also argues that the '937 patent is limited to algorithms that create *n* coded values from *n* data subsets. (Def.'s Br. 18.) Amplidata bases this belief on the same arguments from the October 7, 2009, amendment referenced above: the "coding module also encodes the *n* subsets of data using a coding algorithm to create *n* coded values." (Def.'s Br. 18; J.A. 1030.) Cleversafe attempts to rebut this argument by pointing to the issued patent, in which "*n* subsets" and "*n* coded values" were replaced by "plurality of subsets" and "plurality of values." (Pl.'s Br. 17; J.A. 716.) However, this ignores the prosecution history. Cleversafe made this change from "*n*" to "plurality" in its May 3, 2010 amendment. (J.A. 1042.) In this very same amendment, Cleversafe "reaffirm[ed] its contention that the new claims are

[3] Even ignoring claim differentiation, a dependent claim may not be broader than its independent claim. (35 U.S.C. § 112(d) ("A claim in dependent form shall be construed to incorporate by reference all the limitations of the claim to which it refers.").) This means the "convolutional coding algorithm" in dependent claim 6 must be narrower than the "coding algorithm" of independent claim 1. And that a "coding algorithm" must consist of something in addition to a "convolutional coding algorithm." If "coding algorithm" were limited to the one specific algorithm described in the specification, the family of "convolutional coding algorithms" in claim 7 would necessarily be broader than the independent claim, thereby violating § 112(d). *See also* Markman Hr'g Tr. at 88-89.

distinguished over Moulton based on the arguments presented in the office response of 10/3/2009." (J.A. 1050.) As the record does not show any correspondence on October 3, 2009, this must be a reference to the October 7, 2009, amendment in which Cleversafe claimed that the "coding module also encodes the *n* subsets of data using a coding algorithm to create *n* coded values." (J.A. 1030.) By defending this one-to-one ratio of data sets to coded values in the same amendment that changed "*n*" to "plurality," Cleversafe told the Patent Office that the coding algorithm must create a one-to-one correlation.

In light of the above analysis, the Court declines to adopt either construction proposed by the parties. The coding algorithm disclosed in the '937 patent is not limited to the preferred embodiments in the specification, but must create a one-to-one correlation between data subsets and coded values. Therefore, the Court construes "encode, using a coding algorithm, a plurality of subsets of data to create a plurality of coded values" to mean "encode, using a coding algorithm, *n* subsets of data to create *n* coded values."

IV. "Information dispersal algorithm"

The parties dispute the phrase "information dispersal algorithm," which appears in claims 3 and 5 of the '427 patent. The '427 patent is a continuation-in-part of the '937 patent, and the parties argue along largely the same lines as the previous term. Amplidata suggests the construction "the algorithm that segments a file into *n* data subsets, encodes the *n* data subsets to create *n* coded values, and creates *n* data slices from *n* data subsets and *n* coded values, as described at 4:27-9:32," while Cleversafe believes no construction is necessary. (Def.'s Br. 19; Pl.'s Br. 18.) In the alternative, Cleversafe has suggested the construction "a method of dividing a file into a number of subsets such that the data in each subset is less usable or less recognizable

or completely unusable or completely unrecognizable by itself except when combined with some or all of the other subsets." (Pl.'s Br. 19.)

Amplidata urges its preferred construction for "exactly the same reasons" as the "coding algorithm" previously discussed, and for substantially the same reasons this construction is incorrect. (Def.'s Br. 19.) Like the '937 patent, the '427 patent also claims to be "able to reliably and securely protect data that does not require the use of computation intensive algorithms." ('427 Patent col.3 ll.1-3 J.A. 21.) For this reason, Amplidata believes the algorithm should be limited to computationally efficient algorithms, and that the specification does not adequately describe what algorithms fall within that category. (Def.'s Br. 20.) Therefore, Amplidata believes the only computationally efficient algorithm disclosed is that of the embodiment, and seeks to limit the definition to this specific algorithm. (Def.'s Br. 21.) As discussed previously, however, stating that a system *does not require* a computationally intensive algorithm is not the same as stating the system *requires* a computationally efficient algorithm. There is thus no reason to limit "information dispersal algorithm" to the embodiments.

Looking to the claims themselves, Claims 3 and 5 of the '427 patent describe two functions of the information dispersal algorithm: 1) "assembling the listed file from the data slices" and 2) "creating n new data slices from the rebuilt file," where n is the number of data slices making up the file. ('427 Patent col.16, ll.21-24, J.A. 27.) The specification broadly defines the algorithm as able to break up data into smaller pieces so it is less identifiable. ('427 Patent at [57], J.A. 1.) Although Amplidata likens this algorithm to the coding algorithm described in the '937 patent, the coding algorithm is distinct from the information dispersal algorithm, and the information dispersal algorithm is not required to utilize the same mechanism as the coding algorithm. The information dispersal algorithm is required only to have the

capabilities described in claim 3. Therefore, this Court defines "information dispersal algorithm" as "an algorithm capable of assembling a file from data slices and creating new data slices from a rebuilt file."

V. "String(s) of data"

The parties disagree about the proper construction of the term "string(s) of data" used in claims 1, 2, 5, 7, 9, 10, and 13 of the '937 patent. Amplidata argues that a "string of data" must be "a sequence of human-readable characters." (Def.'s Br. 21.) Cleversafe counters that no construction is required because a person skilled in the art would understand the meaning of a "string of data." (Pl.'s Br. 20.) In the alternative, Cleversafe suggests the definition "a sequence of information in a format that can be processed by a computer." (*Id.*)

Amplidata attempts to import limitations based upon exemplary language in the specification. For example, Amplidata points to the use of multiple examples of "strings" used in the '937 patent, such as "the quick brown box jumped over the lazy dog." (Def.'s Br. 22.) While the examples cited by Amplidata are all human-readable, the purpose of the examples in the patent is to facilitate understanding of the invention, not impose new limitations. An inventor could not be expected to provide examples of an invention that could not be read by humans when the readers of a patent are, in fact, humans.

Additionally, Cleversafe does offer one illustration where the exemplary string is not human-readable: "For example, where the original data is the starting string of ASCII *values* for the characters of the text O L I V E R . . . ." (Pl.'s Br. 20 (emphasis added).) This particular example directly rebuts Amplidata's argument. Moreover, nothing in the prosecution history suggests that "strings of data" must be "human-readable." As such, the intrinsic evidence does not support Amplidata's construction that "strings of data" must be "human-readable," but

supports the opposite. Accordingly, Amplidata's reliance upon extrinsic evidence (Def.'s Br. 22) is misplaced. *Vitronics*, 90 F.3d at 1583. For these reasons, the Court concludes that the phrase "strings of data" is unambiguous and does not require any construction beyond its plain and ordinary meaning.

VI. "Subsets of data"

The parties disagree over the meaning of the term "subsets of data" used in claims 1, 7, and 9 of the '937 patent. Amplidata argues that "subsets of data" must be "portions of a string of data," while Cleversafe believes no construction is necessary. (Def.'s Br. 22; Pl.'s Br. 21-22.) However, Cleversafe's own brief states "[i]n the context of the claim language, a 'subset of data' is a *portion of a 'string of data*.'" (Pl.'s Br. 22 (emphasis added).) It thus appears that the parties agree that a "subset of data" is "a portion of a string of data," and this Court concurs.

VII. "List of unusable storage nodes"

Next, the parties disagree about the proper construction of the phrase "list of unusable storage nodes" that appears in claims 3 and 5 of the '427 patent. Specifically, Amplidata argues that "unusable storage nodes" in the context of the '427 patent must be "permanently unusable." (Def.'s Br. 23.) Cleversafe claims that no construction is required. (Pl.'s Br. 22.)

Amplidata supports its argument by pointing to Cleversafe's Reply to Office Action of September 15, 2008, in which Cleversafe stated "[t]he claimed invention further requires that one or more storage nodes are rendered permanently unusable, and that a list of such unusable storage nodes is assembled . . . ." (*Id.*) Amplidata claims that Cleversafe made this statement to circumvent the prior art and thus limited its claims to "permanently unusable storage nodes." (*Id.*)

Cleversafe responds that the patent was allowed for other reasons because the examiner did not respond to this statement. (Pl.'s Br. 23.) However, this cannot overcome Cleversafe's clear assertion to the examiner that "unusable" storage nodes are permanently unusable. Therefore, this Court adopts Amplidata's construction and defines "list of unusable storage nodes" to mean "list of storage nodes that have been rendered permanently unusable."

VIII. "The request is valid" and "the request is invalid"

The parties further disagree on the meaning of the phrase "the request is valid" or "the request is invalid" as used in claims 1 and 6 of the '540 patent. (Def.'s Br. 24.) Amplidata claims that the phrase "the request is valid/invalid" should be read to mean "the user has/does not have appropriate permissions to access the requested data." (*Id.*) Cleversafe believes no construction is necessary. (Pl.'s Br. 24.)

Amplidata supports this construction by pointing to several examples in the specification relating to user "permissions." (*Id.*) For example, "a vault may also include an access control list specifying which accounts are allowed to access the vault, and what permissions are associated with that account." (*Id.*, '771 Patent col.4 ll.25-28, J.A. 568.) While the specification does indicate that appropriate permissions are required for a request to be valid, the specification does not limit validity solely to user permissions.

Cleversafe argues that claim differentiation prohibits Amplidata's interpretation because claim 4 of the '771 patent would leave claim 1 superfluous. (Pl.'s Br. 24.)

Claim 1 provides:

> A dispersed data storage system comprises: a plurality of slice servers . . . and wherein the slice server functions to: receive a request to access a virtual digital data storage vault; determine whether the virtual digital data storage vault is the first or the second virtual digital data storage vault; when the virtual digital data storage vault is the first or the second virtual digital data storage vault, *determine*

> *whether the request is valid; and when the request is valid, execute the request to generate a response*.

('771 Patent col.5 ll. 31-47, J.A. 569 (emphasis added).)

Dependent claim 4 further validates "based on permissions associated with the user account." (*Id.* at col.6 ll.1-2, J.A. 569.) If Amplidata's interpretation were accurate, then there would be no need for dependent claim 4 because claim 1 would already incorporate user permissions. Additionally, the presumption in favor of claim differentiation is powerful here because "the limitation in dispute is the only meaningful difference between an independent and dependent claim." *Acumed*, 483 F.3d at 806. Thus, this Court agrees with Cleversafe that whether a request is "valid" is not limited only to user permissions, and accordingly no construction of this phrase is required.

IX. "Supports"

The final term that is disputed by the parties is "supports." This term appears in claim 1 of the '771 patent, which claims a "plurality of slice servers, wherein a first set of the plurality of slice servers supports a first virtual digital data storage vault and a second set of the plurality of slice servers supports a second virtual digital data storage vault." ('771 Patent col.5 ll.32-35, J.A. 569.) Amplidata believes "supports" should be interpreted to mean "makes up" in this context (Def.'s Br. 25) and cites to several passages from the specification. For example, Amplidata points to the statement that "[e]ach vault makes use of some number of slice servers, and a particular slice server may be associated with any number of vaults. . . . By example, a first vault may be comprised of 16 slice servers." (*Id.* at 25-26; '771 Patent col.3 ll.50-55, J.A. 568.) Amplidata then concludes without explanation that this language indicates slice servers support vaults by making up the structure of the vaults. (Def.'s Br. 26.)

Such a reading, however, is at odds with Amplidata's own suggested construction of the term "virtual digital data storage vault" as "a software construct associated with particular slice servers and user accounts." (*Id.* at 7.) "Make up" suggests that the slice servers form the vaults. If a vault is a software construct, then it is difficult to imagine how slice servers could "make up" vaults. This Court has defined "virtual digital data storage vault" to mean a "software construct stored over multiple servers." The servers thus store the vaults, but do not themselves "make up" the vaults."

On the other hand, Cleversafe argues that the term "support" only serves to link slice servers and vaults. (Pl.'s Br. 25.) Further, Cleversafe does not believe any construction is necessary because a person skilled in the art would understand the relationship between slice servers and vaults to be "classic client-server-service architecture." (*Id.*) Out of an abudance of caution, Cleversafe has also proposed an alternative definition of "support" as "provides access to." (*Id.*)

Cleversafe's definition accurately describes the relationship between the slice servers and virtual digital data storage vaults. Slice servers cannot "make up" vaults because the vaults are virtual software constructs. However, the slice servers described by the '771 patent control access to the virtual digital data storage vaults by determining whether requests for access are valid. ('771 Patent col.5 ll.31-47, J.A. 569.) Therefore, the Court construes the term "supports" to mean "provides access to" in the context of the '771 patent.

Terms as Construed

#	Term	Definition
I	"Data slice" and "plurality of data slices"	"Data structure consisting of a data subset and a coded value" and a "plurality of data structures each consisting of a data subset and a coded value"
II	"Virtual digital data storage vault"	"Software construct stored over multiple slice servers"
III	"Encode, using a coding algorithm, a plurality of subsets of data to create a plurality of coded values"	"Encode, using a coding algorithm, *n* subsets of data to create *n* coded values"
IV	"Information dispersal algorithm"	"An algorithm capable of assembling a file from data slices and creating new data slices from a rebuilt file"
V	"String(s) of data"	No construction required
VI	"Subsets of data"	"Portions of a string of data"
VII	"List of unusable storage nodes"	"List of storage nodes that have been rendered permanently unusable"
VIII	"The request is valid" or "the request is invalid"	No construction required
IX	"Supports"	"Provides access to"

Conclusion

For the foregoing reasons, the nine disputed claim terms are construed as set forth in this Memorandum Opinion and Order.

SO ORDERED **ENTER: 5/20/14**

JOHN Z. LEE
U.S. District Judge