UNITED STATES DISTRICT COURT
DISTRICT OF MASSACHUSETTS

CIVIL ACTION NO. 16-11613-RGS

EGENERA, INC.

v.

CISCO SYSTEMS, INC.

MEMORANDUM AND ORDER ON CLAIM CONSTRUCTION

February 5, 2018

STEARNS, D.J.

Plaintiff Egenera, Inc., accuses defendant Cisco Systems, Inc., of infringing United States Patent No. 7,231,430 (the ’430 patent).[1] Before the court are the parties’ briefs on claim construction. The court received technical tutorials and heard argument, pursuant to *Markman v. Westview Instruments, Inc.*, 517 U.S. 370 (1996), on January 30, 2018.

[1] In its Complaint, Egenera also asserted infringement of U.S. Patents Nos. 6,971,044 (the ’044 patent) and 7,178,059 (the ’059 patent). On Cisco’s motion to dismiss, the court found the ’059 patent to claim patent-ineligible subject matter. *Egenera, Inc. v. Cisco Sys., Inc.*, 234 F. Supp. 3d 331, 345-346 (D. Mass. 2017). Egenera dismissed the ’044 patent without prejudice after the Patent Trial and Appeal Board instituted *inter partes* review of all claims. *See* Dkt ## 77 at 11-12; 78, 80, & 81.

THE ’430 PATENT

The ’430 patent is entitled “Reconfigurable, Virtual Processing System, Cluster, Network, and Method,” and was issued on June 12, 2007, from an application filed on January 4, 2002. It lists as the inventors Vern Brownell, Pete Manca, Ben Sprachman, Paul Curtis, Ewan Milne, Max Smith, Alan Greenspan, Scott Geng, Dan Busby, Edward Duffy, and Peter Schulter. The ’430 patent sets out 8 claims, including 4 system claims and 4 method claims.

The ’430 patent, directed to solving problems in manually configuring, deploying, and maintaining enterprise and application servers, *see id.*, col. 1, ll. 21-58, discloses “a processing platform from which virtual systems may be deployed through configuration commands,” *id.* col. 2, ll. 45-47.

> The platform provides a large pool of processors from which a subset may be selected and configured through software commands to form a virtualized network of computers (“processing area network” or “processor clusters”) that may be deployed to serve a given set of applications or customer. The virtualized processing area network (PAN) may then be used to execute customer specific applications, such as web-based server applications. The virtualization may include virtualization of local area networks (LANs) or the virtualization of I/O storage. By providing such a platform, processing resources may be deployed rapidly and easily through software via configuration commands, e.g., from an administrator, rather than through physically providing servers, cabling network and storage connections, providing power to each server and so forth.

Id. col. 2, ll. 47-62.[2]

Claim 1 of the ’430 patent is representative.

> 1. A platform for automatically deploying at least one virtual processing area network, in response to software commands, said platform comprising:
>
> a plurality of computer processors connected to an internal communication network;
>
> at least one control node in communication with an external communication network and in communication with an external storage network having an external storage address space, wherein the at least one control node is connected to the internal communication network and thereby in communication with the plurality of computer processors, said at least one control node including logic to receive messages from the plurality of computer processors, wherein said received messages are addressed to the external communication network and to the external storage network and said at least one control node including logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network;
>
> configuration logic for receiving and responding to said software commands, said software commands specifying (i) a number of processors for a virtual processing area network (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) a virtual storage space for the virtual processing area network, said configuration logic including logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer

[2] Additional descriptions of the claimed invention of the ’430 patent may be found in the court’s Memorandum and Order on Cisco’s motion to dismiss. *See Egenera*, 234 F. Supp. 3d at 334-336.

> processors, to program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology, and to program the at least one control node to define a virtual storage space for the virtual processing area network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network; and
>
> wherein the plurality of computer processors and the at least one control node include network emulation logic to emulate Ethernet functionality over the internal communication network.

The parties agree that the preambles of the claims are limiting, and that that a "virtual processing area network" is "a software simulated network of computer processors." *See* Cisco Br., Dkt # 65 at 3. The construction of the following claim terms are disputed:

- "computer processor"/"processor"
- "logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network"
- "logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors"
- "logic to . . . program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology"
- "logic to . . . program the at least one control node to define a virtual storage space for the virtual processing area network"
- "emulate Ethernet functionality over the internal communication network"

DISCUSSION

Claim construction is a matter of law. *See Markman*, 517 U.S. at 388-389. Claim terms are generally given the ordinary and customary meaning that would be ascribed by a person of ordinary skill in the art in question at the time of the invention.[3] *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312-1313 (Fed. Cir. 2005) (en banc) (citations omitted). In determining how a person of ordinary skill in the art would have understood the claim terms, the court looks to the specification of the patent, its prosecution history, and in limited instances where appropriate, extrinsic evidence such as dictionaries, treatises, or expert testimony. *Id.* at 1315-1317. Ultimately, "[t]he construction that stays true to the claim language and most naturally aligns with the patent's

[3] Egenera asserts that a person of ordinary skill in the art is one who has "(i) a Bachelor's degree in Computer Science, or equivalent training, and (ii) approximately five years of experience working in software design, including in computer system development related to network computing and storage." Egenera Br., Dkt # 66 at 6. Cisco's expert opines that such a person has "a Bachelor's degree in electrical and/or computer engineering, or computer science. They would also have approximately two years of experience working in hardware and/or software network computing." Katz Decl., Dkt # 65-1 ¶ 16. The parties do not rely on the minor differences between their characterizations of a skilled artisan as a basis to distinguish the construction of the disputed terms.

description of the invention will be, in the end, the correct construction." *Id.* at 1316 (citation omitted).

"computer processor"/"processor"

The parties agree that when the word "processor" appears alone in the claims, it is a shorthand reference to "computer processor." *See* Egenera Br. at 8; Cisco Br. at 6. The two terms will therefore be construed identically.

Cisco argues that a "computer processor" is commonly understood by a person of ordinary skill in the art to refer to a CPU (Central Processing Unit – the circuitry within a computer that receives input, executes software instructions, and produces output).[4] *See* Katz Decl. ¶¶ 22, 26. Egenera does not dispute this ordinary meaning, *see* Jones Decl., Dkt # 68 ¶¶ 23-26, Jones Suppl. Decl., Dkt # 72 ¶ 6, but contends that in the context of the '430 patent, "computer processor" refers to the "processing node" described in the specification.

In support of its position, Egenera notes that in the claims, "computer processors" are identified as the members of the PAN that perform certain actions. Claim 1, for example, states that "a plurality of computer processors

[4] The technical dictionaries submitted by the parties uniformly reflect this ordinary meaning. *See, e.g.*, Egenera Ex. 12, Microsoft Press Computer Dictionary (3d ed. 1997) (equating "processor" with "central processing unit, microprocessor"); Cisco Ex. 4, Random House Webster's Computer & Internet Dictionary (3d ed. 1999) (same).

[are] connected to an internal communication network," and that "at least one control node is connected to the internal communications network and thereby in communication with the plurality of computer processors . . . [and] receive[s] messages from the plurality of computer processors." According to Egenera, a person of ordinary skill in the art would understand that a CPU is not independently capable of connecting to a network, nor can it communicate with a control node or send messages. *See* Jones Decl. ¶ 26. Instead, the networking and messaging functions performed by the claimed "computer processors" are attributed in the specification to "processing nodes." "Under certain embodiments, about 24 processing nodes 105a-n, two control nodes 120, and two switch fabrics 115*a,b* are contained in a single chassis and interconnected with a fixed, pre-wired mesh of point-to-point (PtP) links." '430 patent, col. 3, ll. 9-12. For example, figure 2A illustrates that processing nodes ($PN_{1...m}$) are the entities interconnected within the PAN.

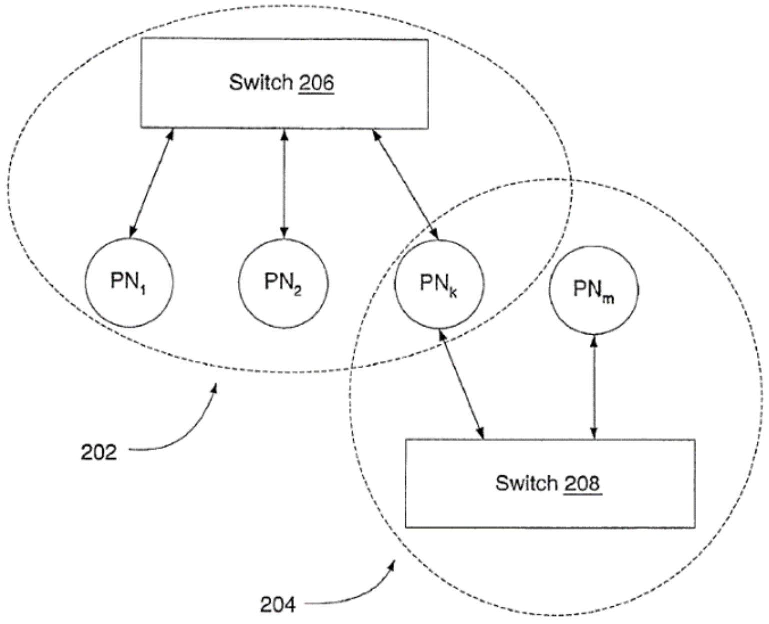

FIG. 2A

Egenera's evidence does not meet the exacting standard required to establish that the patentee gave a transformative meaning to the term "computer processor" as it is used in the '430 patent.

> To act as its own lexicographer, a patentee must "clearly set forth a definition of the disputed claim term" other than its plain and ordinary meaning. *CCS Fitness, Inc. v. Brunswick Corp.,* 288 F.3d 1359, 1366 (Fed. Cir. 2002). It is not enough for a patentee to simply disclose a single embodiment or use a word in the same manner in all embodiments, the patentee must "clearly express an intent" to redefine the term. *Helmsderfer v. Bobrick Washroom Equip., Inc.,* 527 F.3d 1379, 1381 (Fed. Cir. 2008); *see also Kara Tech. Inc. v. Stamps.com,* 582 F.3d 1341, 1347-48 (Fed. Cir. 2009).

Thorner v. Sony Computer Entm't Am. LLC, 669 F.3d 1362, 1365 (Fed. Cir. 2012).[5] First, the claims require that the "plurality of computer processors"

[5] Egenera's evocation of the standard for claim scope disavowal is inapposite. Egenera does not suggest that a "computer processor" is

be "connected to an internal communications network," but impose no limitation that the connection be established directly. *See* Jones Decl. ¶ 26 (conceding that "a CPU, in a broad sense, may be considered connected to a network"). The same is true for communication and messaging – Egenera acknowledged at the *Markman* hearing that the CPU participates in communicating to the control node and sending messages. The specification also confirms the view that processors (and not necessarily processor nodes) communicate and send messages. "Each PAN, through software commands, is configured to have a corresponding subset of processors 106 that may communicate via a virtual local area network . . . ." '430 patent, col. 3, ll. 55-57. "The processors 106 use this virtual interface to send SCSI I/O commands to the control nodes 120 for processing." *Id.* col. 23, ll. 26-28. Second, the patentee knew how to claim "nodes" by claiming "at least one control node," but elected to direct the claim language to "computer processors" instead of "processing nodes."

Finally, and fatal to Egenera's argument is that, rather than equating processors to processing nodes, the specification disambiguates them. "*Each processing node 105 is a board that includes one or more (e.g., 4) processors*

commonly understood to encompass a "processing node," and Cisco does not seek to narrow the ordinary meaning of a "computer processor."

106j-l, one or more network interface cards (NICs) 107, and local memory (e.g., greater than 4 Gbytes) that, among other things, includes some BIOS firmware for booting and initialization." *Id.* col. 3, ll. 13-17 (emphasis added). "As each processor boots, BIOS-based boot logic initializes *each processor 106 of the node 105* and, among other things, establishes a (or discovers the) VI 212 to the control node logic." *Id.* col. 6, ll. 18-21 (emphasis added). The highlighted portion of figure 1, *infra*, illustrates this hierarchical relationship between processing node 105m, and processors 106j and 106l.

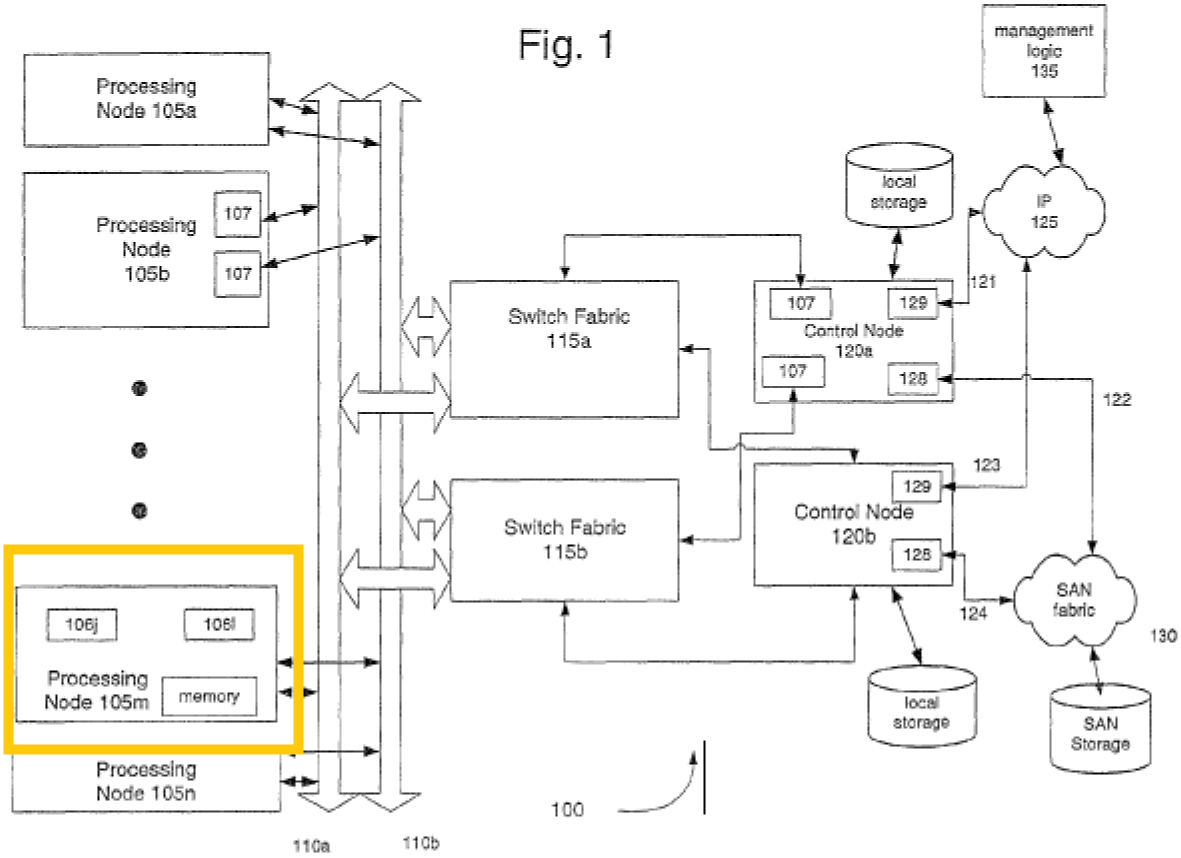
Fig. 1

Because a person of ordinary skill in the art would not read the ’430 patent as having redefined “computer processor” to mean a “processing node,” a “computer processor” will be accorded its ordinary meaning of a “CPU.”

the “logic” terms

Egenera contends that “logic” denotes “software, firmware, circuitry, or some combination thereof,” and that the “logic” terms need no further construction. For its part, Cisco maintains that “logic” is an empty nonce word, and that the associated terms should be analyzed as means-plus-function claiming.

Under 35 U.S.C. § 112, para. 6,

> [a]n element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

Section 112 permits purely functional claiming, but only on condition that the scope of such claim language is “restrict[ed] to the structure disclosed in the specification and equivalents thereof.” *Greenberg v. Ethicon Endo-Surgery, Inc.*, 91 F.3d 1580, 1582 (Fed. Cir. 1996). In identifying means-plus-function terms, the absence of the signal phrase “means,” as is the case here, creates a rebuttable presumption that the Section 112, para. 6 does not apply. *Advanced Ground Info. Sys., Inc. v. Life360, Inc.*, 830 F.3d 1341, 1347 (Fed. Cir. 2016), citing *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1348 (Fed. Cir. 2015).

> The standard is whether the words of the claim are understood by persons of ordinary skill in the art to have a sufficiently definite meaning as the name for structure. *Greenberg* [*v. Ethicon Endo-Surgery, Inc.*], 91 F.3d [1580,] 1583 [(Fed. Cir. 1996)]. When a claim term lacks the word "means," the presumption can be overcome and § 112, para. 6 will apply if the challenger demonstrates that the claim term fails to "recite sufficiently definite structure" or else recites "function without reciting sufficient structure for performing that function." *Watts* [*v. SL Systems, Inc.*], 232 F.3d [877,] 880 [(Fed. Cir. 2000)].

Williamson, 792 F.3d at 1349 (Fed. Cir. 2015).[6]

Both Egenera and Cisco turn to dictionaries to ascertain how a person of ordinary skill in the art at the time of the invention would have understood the term "logic." These definitions, as shown in the table below, fall into two categories. The first category is one of abstract concept, referring to computer operations generally or the plan or sequence of steps taken by a computer to perform a function. The second category is structural, denoting computer circuitry.

[6] In *Williamson*, the Federal Circuit overruled a line of cases characterizing as "strong" the presumption that a limitation without the phrase "means" does not fall under Section 112. *Id.*

Source	**Abstract Definition**	**Circuitry Definition**
The American Heritage College Dictionary 797 (3d ed.1997) (cited in *Skyhook v. Wireless, Inc. v. Google, Inc.*, 2012 WL 4076180 at * 16 (D. Mass. Sep. 14, 2012)).	a. The nonarithmetic operations performed by a computer, such as sorting, that involve yes-no decisions.	b. Computer circuitry
Wiley Electrical and Electronics Engineering Dictionary 432 (2004) (cited in *Skyhook*)	1. The functions performed by a computer which involve operations such as mathematical computations and true/false comparisons....	2. The circuits in a computer which enable the performance of logic functions or operations, such as AND, OR, and NOT. These include gates and flip-flops. Also, the manner in which these circuits are arranged.... 3. The totality of the circuitry contained in a computer.
McGraw–Hill Dictionary of Scientific and Technical Terms 1101 (4th ed. 1989) (cited in *Skyhook*) *McGraw–Hill Dictionary of Scientific and Technical Terms* 1101 (6th ed. 2003) (Egenera Ex. 11)	1. The basic principles and applications of truth tables, interconnections of on/off circuit elements, and other factors involved in mathematical computation in a computer.	2. General term for the various types of gates, flip-flops, and other on/off circuits used to perform problem-solving functions in a digital computer.
The Compact Oxford English Dictionary 1108 (2d ed.) (cited in *Skyhook*)	The system or principles underlying the representation of logical operations and two-valued variables by electrical or other physical signals and their interactions; the forms and interconnections of logic elements in any particular piece of equipment, in so far as they relate to the interaction of signals and not to the physical nature of the components used; . . . local operations collectively, as performed by electronic or other devices.	[A]lso, the actual components and circuitry.

Microsoft Press Computer Dictionary (3d. ed. 1997) (Egenera Ex. 12)	In programming, the assertions, assumptions, and operations that define what a given program does. Defining logic of a program is often the first step in developing the program's source code.	
The IEEE Standard Dictionary of Electrical and Electronics Terms (6th ed. 1996) (Cisco Ex. 5)	(1)(A) The result of planning a data-processing system or of synthesizing a network of logic elements to perform a specified function.	(B) Pertaining to the type or physical realization of logic elements used, for example, diode logic, and logic.

Egenera cites several cases where courts have held that claim terms directed to "logic" recite sufficient structure to insulate the term from analysis under Section 112. In each of the cited cases, the court adopted or relied upon a circuitry definition of "logic."[7] In *TecSec, Inc. v. Int'l Bus. Machines Corp.*, 731 F.3d 1336, 1348 (Fed. Cir. 2013), the Federal Circuit construed "digital logic means" as "digital circuits that perform Boolean algebra." The Court cited as evidence of structure the fact that the claim term did not recite a function to be performed, but was rather comprised of structural elements. *Id.* Similarly, in *St. Clair Intellectual Prop.*

[7] With respect to "circuit" as a claim term, the Federal Circuit has found it not "necessary to hold that the term 'circuit' by itself always connotes sufficient structure, the term 'circuit' with an appropriate identifier such as 'interface,' 'programming' and 'logic,' certainly identifies some structural meaning to one of ordinary skill in the art." *Apex Inc. v. Raritan Computer, Inc.*, 325 F.3d 1364, 1373 (Fed. Cir. 2003).

Consultants, Inc. v. Canon, Inc., 2004 WL 1941340, at *20-21 (D. Del. Aug. 31, 2004), the court found that the claim term "logic means" described structure because it "clearly refers to a logic circuit." Likewise, in *PCTEL, Inc. v. Agere Sys., Inc.*, 2005 WL 2206683, at *21-22 (N.D. Cal. Sept. 8, 2005), the court accepted plaintiff's contention that "'logic' is synonymous with circuitry" – "[a] review of the technical dictionaries supports PCTEL's view that 'logic,' by itself, can connote structure. (O'Grady Decl., Ex. 16, *McGraw Hill Dictionary of Scientific and Technical Terms* (1994) ('General term for the various types of games, flip-flops, and other on/off circuits . . . .')."[8]

A person of ordinary skill in the art, in contrast, would not understand "logic" as used in the '430 patent to refer to circuitry. Unlike the claims in *TecSec*, the "logic" terms do not recite structural components, and each is described by a specific function.[9] The specification discloses that "logic" has

[8] Egenera also relies heavily on *Skyhook* to support its contention that "logic" connotes structure to a person of ordinary skill in the art. In *Skyhook*, in the context of "computer implemented logic" terms, the court recited dictionaries reflecting both the abstract and structural definitions of "logic," and concluded that "'logic' has a known structural meaning in the context of computer science." 2012 WL 4076180, at *15-16. Notably, the patentee did not seek, and the court did not adopt, a construction of "logic" consistent with only its abstract conception.

[9] Reciting an intended function does not by itself conclusively establish the non-structural nature of a claim element. *See Phillips*, 415 F.3d at 1311

to be implemented. *See* ’430 patent, col. 23, ll. 23-24 (“The processor-side logic 620 of the protocol is implemented as a host adapter module . . . .”); *id.* col. 25, ll. 3-4 (“Under certain embodiments, the control node-side storage logic 715 is implemented as a device driver module.”). Once implemented, “logic” may take the form of “software logic,” *id.* col. 3, ll. 61, 63, or may be “BIOS-based,” *id.* col. 6, l. 18. These characteristics are consistent with an understanding of logic as an abstraction for the set of steps designed to accomplish a stated function.

Tellingly, to capture the breadth of “logic” in the ’430 patent, Egenera proposes the construction: “software, firmware, circuitry or some combination thereof.” In *Williamson*, the Court held that a claimed “distributed learning control module” did not recite sufficient structure because “‘module’ is simply a generic description for software or hardware that performs a specified function.” 792 F.3d at 1350. “Logic” as Egenera defines it fares no better – “software, firmware, circuity or some combination thereof” is so broad and formless as to be “a generic ‘black box’ for performing the recited computer-implemented functions.” *Id*; *see also Visual Networks Operations, Inc. v. Paradyne Corp.*, 2005 WL 1411578, at

(“While the baffles in the ’798 patent are clearly intended to perform several functions, the term ‘baffles’ is nonetheless structural; it is not a purely functional placeholder in which structure is filled in by the specification.”).

*30 (D. Md. June 15, 2005) ("'Logic for determining at least one dedicated time slot(s)' describes only a function, not a structure. Any number of different algorithms, in the form of either computer code or hard-wired circuit logic, could perform the recited function.").

Having concluded that the totality of the evidence rebuts the presumption against Section 112, I will analyze the "logic" terms accordingly. Construction of means-plus-function claim terms proceeds in two steps. "First, we must identify the claimed function, staying true to the claim language and the limitations expressly recited by the claims. Once the functions performed by the claimed means are identified, we must then ascertain the corresponding structures in the written description that perform those functions." *Omega Eng'g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1321 (Fed. Cir. 2003) (citations omitted). The parties agree that the function of the "logic" terms are those recited, but disagree as to the corresponding structures. With the exception of the "logic to modify . . ." term, Cisco contends that the specification does not disclose sufficient structure to perform the stated functions and are therefore indefinite. Consistent with the heightened standard of proof required to overcome the presumption of patent validity, "a challenge to a claim containing a means-plus-function limitation as lacking structural support requires a finding, by clear and convincing evidence, that the specification lacks disclosure of structure sufficient to be understood by one skilled in the art as being adequate to perform the recited function." *Budde v. Harley-Davidson, Inc.*, 250 F.3d 1369, 1376-1377 (Fed. Cir. 2001).

> "*logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network*"

For the "logic to modify . . ." term, Egenera identifies as structure "control node 120 operating as described at 3:26-31, 13:60-67, 14:53-67, 17:29-18:33, 18:37-19:3, 21:37-50, 22:11-29, 25:20-29, 25:30-26:46, 29:12-15, 29:57-60, and/or 30:45-48, and equivalents thereof." Cisco, for its part, points to "virtual LAN proxy 340 connected to virtual LAN server 335 and connected to the external network 125 through the physical LAN driver 345" for messages addressed to the external communication network; and "control node-side storage logic 615 that modifies the message by translating the address information contained within the message from the processor to the corresponding mapped SAN address" for messages addressed to the external storage network.

The court agrees with Cisco that corresponding structure must perform the specific function recited by the claim term.

> Structure disclosed in the specification qualifies as 'corresponding structure' if the intrinsic evidence clearly links or

> associates that structure to the function recited in the claim. [*Noah Sys., Inc. v. Intuit Inc.,* 675 F.3d 1302, 1311 (Fed. Cir. 2012)] (citing *B. Braun Med., Inc. v. Abbott Labs.,* 124 F.3d 1419, 1424 (Fed. Cir. 1997)).

Williamson, 792 F.3d at 1352. With respect to messages to the external communications network, the excerpted portion of figure 3B is illustrative.

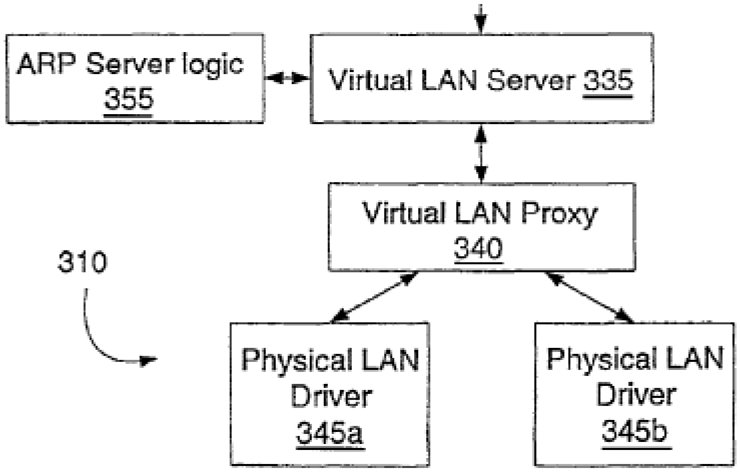

FIG. 3B

> When the external network 125 is running in filtered mode and the virtual LAN Proxy 340 receives outgoing packets (ARP or otherwise) from a virtual LAN server 335**,** it replace [sic] the internal format MAC address with the MAC address of the physical Ethernet device 129 as the source MAC address.

Id. col. 18, ll. 53-58. The structure for modifying and transmitting messages to the external communications network is therefore "virtual LAN server 335, virtual LAN proxy 340, and physical LAN driver 345" and equivalents.

Figure 6 depicts the communication paths between the control node (of which the "logic to modify . . ." is an element) and the external storage network.

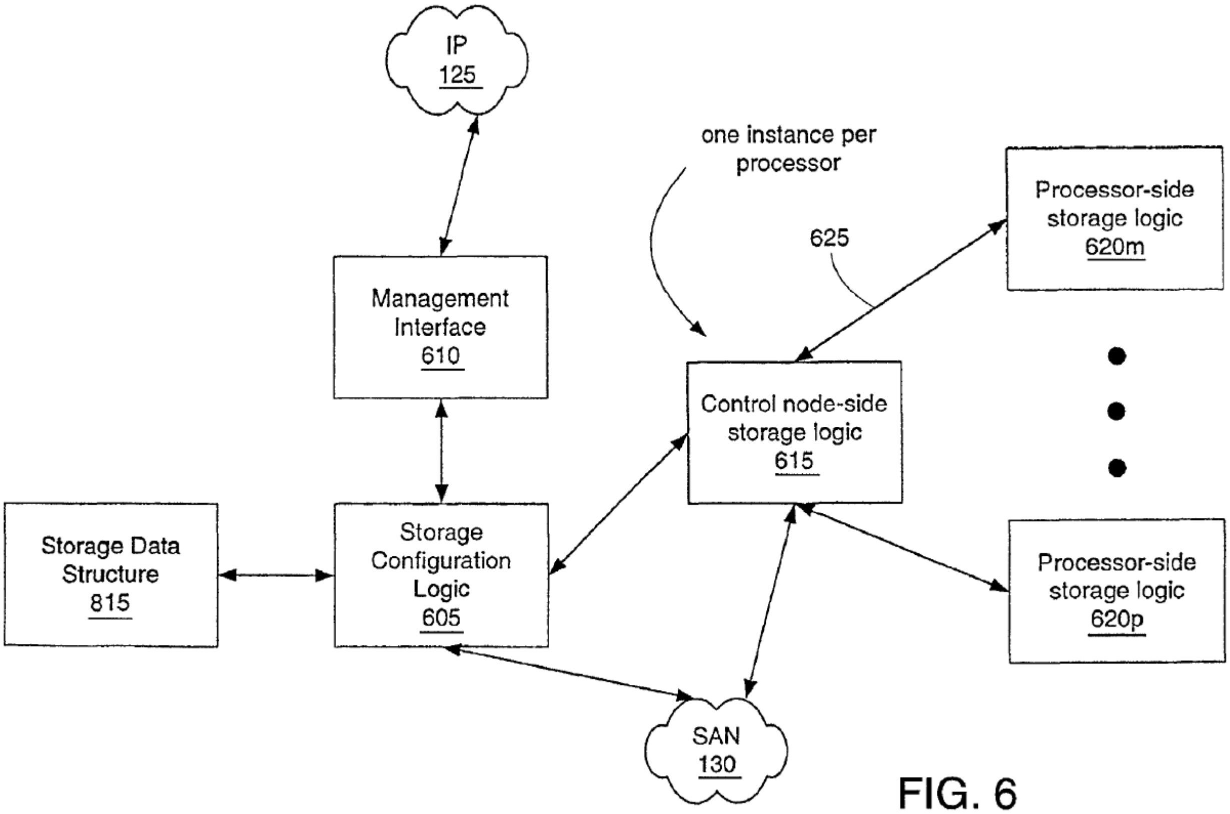
IP
125
one instance per processor
Processor-side storage logic
620m
625
Management Interface
610
Control node-side storage logic
615
Storage Data Structure
815
Storage Configuration Logic
605
Processor-side storage logic
620p
SAN
130

FIG. 6

"A configuration component 605, typically executed on a control node 120, is in communication with external SAN 130." *Id.* col. 21, l. 67-col. 22, l. 2. "Storage configuration logic 605 is also responsible for communicating the SAN storage allocations to control node-side logic 615." *Id.* col. 22, ll. 15-17.

> The control node-side storage logic 715 receives messages from the processor-side logic and then analyzes the header information to determine how to act, e.g., to allocate buffers or the like. In addition, the logic translates the address information contained in the messages from the processor to the

> corresponding, mapped SAN address and issues the commands (e.g., via FCP or FCP-2) to the SAN 130.

Id. col. 25, ll. 23-29. Thus, the structure for modifying and transmitting messages to the external storage network is "storage configuration logic 605" and equivalents.

> *"logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors"*

With respect to the "logic to select . . ." term, Egenera identifies as structure "control node 120 and/or management application 135 operating as described at 2:47-52; 3:4-8, 3:33-37, 3:55-57; 5:56-61, 27:11-44, 28:49-55, 29:24-26, 30:2-4 and/or 30:57-59, and equivalents thereof." Cisco contends the term has no corresponding structure and is therefore indefinite. Although the description is brief, the specification provides that "[t]o create and configure such networks, an administrator defines the network topology of a PAN and specifies (e.g., via a utility within the management software 135) MAC address assignments of the various nodes." *Id.* col. 5, ll. 56-59. Thus, the structure for the "logic to select under programmatic control, a corresponding set of computer processors from the plurality of computer processors" term is "a utility within the management software 135" and its equivalents.

> *"logic to . . . program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology"*

Egenera identifies as the structures for the "logic to . . . establish the specified virtual local area network topology" term "control node 120 and/or management application 135 operating as described at 2:47-52; 3:4-8, 3:33-37, 3:55-57; 5:56-61, 6:6-47, 27:11-44, 28:49-55, 29:24-29, 30:2-7, and/or 30:57-62, and equivalents thereof." Cisco maintains that this term has no corresponding structure and is, again, therefore indefinite.

The specification explains how the local area network topology is established.

> The control node-side networking logic maintains data structures that contain information reflecting the connectivity of the LAN (e.g., which nodes may communicate to which other nodes). The control node logic also allocates and assigns VI [(virtual interface) (or RVI [(reliable virtual interface)]) mappings to the defined MAC addresses and allocates and assigns VIs or (RVIs) between the control nodes and between the control nodes and the processing nodes. In the example of FIG. 2A, the logic would allocate and assign VIs 212 of FIG. 2B. . . .
>
> As each processor boots, BIOS-based boot logic initializes each processor 106 of the node 105 and, among other things, establishes a (or discovers the) VI 212 to the control node logic. The processor node then obtains from the control node relevant data link information, such as the processor node's MAC address, and the MAC identities of other devices within the same data link configuration. Each processor then registers its IP address with the control node, which then binds the IP address to the node and an RVI (e.g., the RVI on which the registration arrived). In this fashion, the control node will be able to bind IP addresses for each virtual MAC for each node on a subnet. In addition to

> the above, the processor node also obtains the RVI or VI-related information for its connections to other nodes or to control node networking logic.

Id. col. 6, ll. 6-32. Figure 2B is illustrative of virtual interfaces between processor nodes and virtual switches.

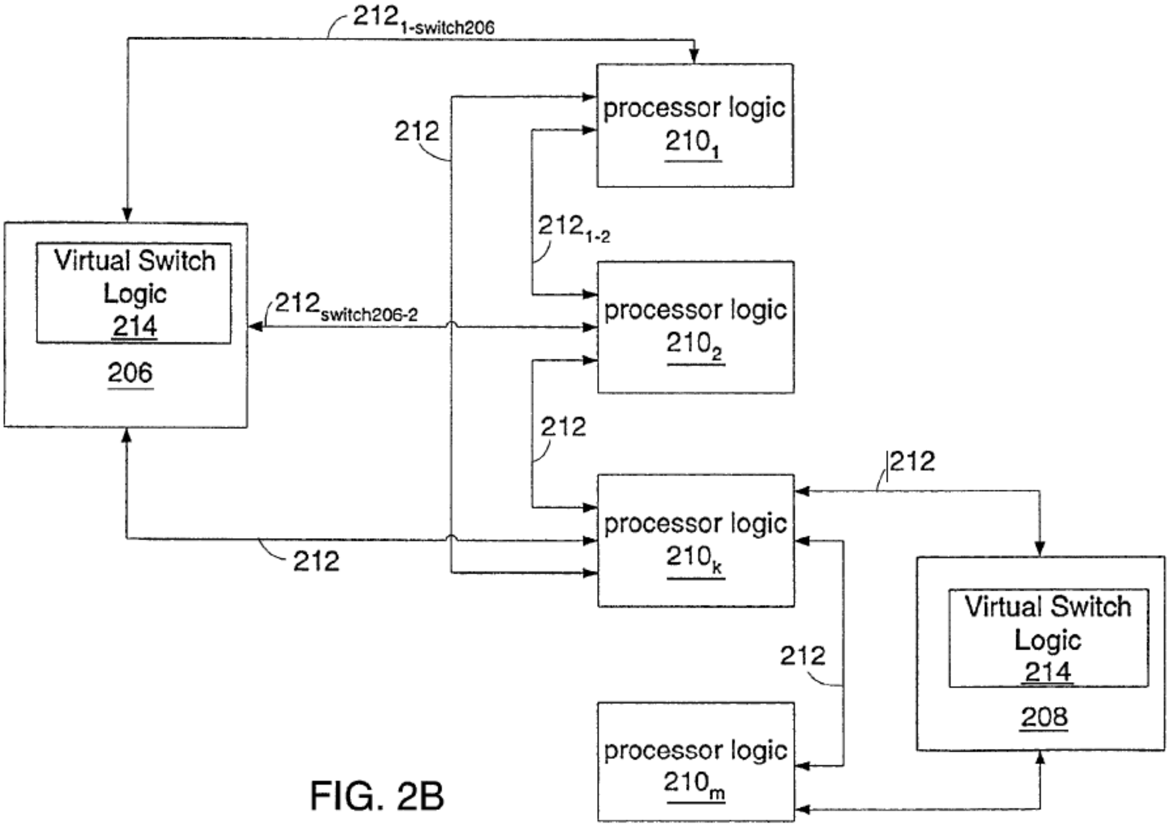

FIG. 2B

The patent further explains that "data structures 910 [] record[s] the networking information . . . such as the network topologies of PANs, the MAC address assignments within a PAN and so on." *Id.* col. 27, ll. 47-51. Thus, the structures that "program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology" are "control node-side networking logic 310, (reliable) virtual interface 212, and data structure 910" and their equivalents.

> *"logic to . . . program the at least one control node to define a virtual storage space for the virtual processing area network"*

Egenera contends that the structure for the "logic to . . . define a virtual storage space" is "control node 120 and/or management application 135 operating as described at 3:4-8, 3:33-37, 3:58-67, 21:51-65, 22:11-15, 22:33-53, 22:57-67, 23:3-11, 26:64-27:4, 27:11-44, 28:49-55, 29:24-31, 30:2-9, and/or 30:57-64, and equivalents thereof." Cisco likewise contends that the term lacks corresponding structure and is indefinite.

With respect to the virtual storage space for each PAN, the specification provides that "the configuration component 605 and interface 610 are responsible for discovering those portions of SAN storage that are allocated to the platform 100 and for allowing an administrator to suballocate portions to specific PANs or processors 106." *Id.* col. 22, ll. 11-15. Configuration component 605

> provides a mapping function that translates the device numbers (e.g., SCSI target and LUN) that the processor uses into the device numbers visible to the control nodes through their attached SCSI and Fibre Channel I/O interfaces 128. It also provides an access control function, which prevents processors from accessing external storage devices which are attached to the control nodes but not included in the processors' configuration. The model that is presented to the processor (and to the system administrator and applications/users on that processor) makes

> it appear as if each processor has its own mass storage devices attached to interfaces on the processor.

Id. col. 22, ll. 35-46. Figure 8 depicts an instance of the storage address mapping logic that may be employed.

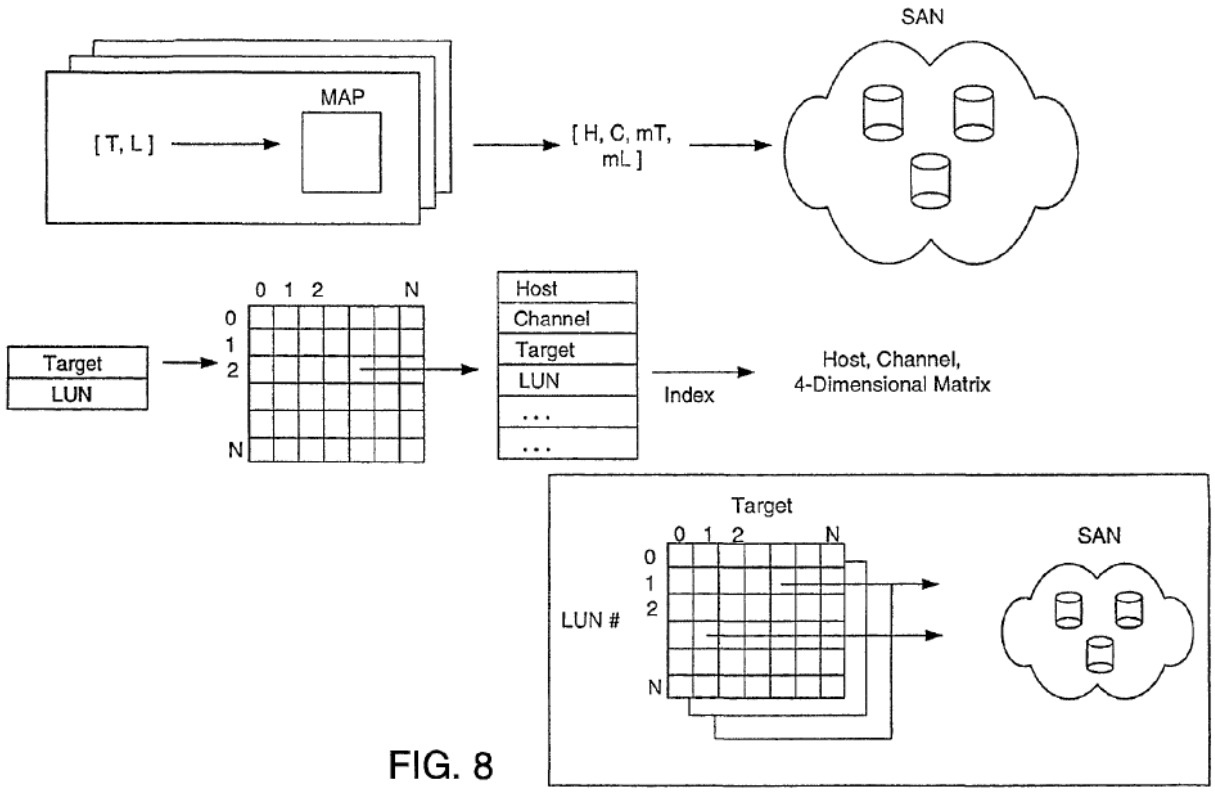

FIG. 8

"[D]ata structure 915 [] record[s] the storage correspondence of various processors 106." *Id.* col. 51-53. The structure for the "logic to . . . program the at least one control node to define a virtual storage space for the virtual processing area network" is therefore "storage configuration logic 605, management interface component 610, and storage data structure 815/915" and equivalents.

In sum, the "logic" terms are construed as follows.

Claim Term	**Function**	**Structure**
"logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network"	"modify said received messages to transmit said modified received messages to the external communication network and to the external storage network"	"virtual LAN server 335, virtual LAN proxy 340, and physical LAN driver 345" and equivalents for messages to the external communications network "storage configuration logic 605" and equivalents for messages to the external storage network
"logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors"	"select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors"	"a utility within the management software 135" and equivalents
"logic to . . . program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology"	"program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology"	"control node-side networking logic 310, (reliable) virtual interface 212, and data structure 910" and equivalents
"logic to . . . program the at least one control node to define a virtual storage space for the virtual processing area network"	"program the at least one control node to define a virtual storage space for the virtual processing area network"	"storage configuration logic 605, management interface component 610, and storage data structure 815/915" and equivalents

"*emulate Ethernet functionality over the internal communication network*"

The parties dispute whether the "internal communication network" is necessarily limited to a "non-Ethernet physical network" (Cisco's position). Cisco contends that because the Ethernet functionality is "emulate[d]," it is necessarily absent from the internal communication network. The specification explains that the object of the platform's "emulation feature" is to produce the expected network response consistent with request of nodes external to the platform.

> Under certain embodiments, the virtual networks so created emulate a switched Ethernet network, though the physical, underlying network is a PtP mesh. The virtual network utilizes IEEE MAC addresses, and the processing nodes support IETF ARP processing to identify and associate IP addresses with MAC addresses. *Consequently, a given processor node replies to an ARP request consistently whether the ARP request came from a node internal or external to the platform*.

Id. col. 4, ll. 8-16 (emphasis added). The platform is agnostic as to the specific architecture of the internal communication network. Indeed, the specification expressly describes using an Ethernet fabric as an alternative internal network architecture. *See id.* col. 28, ll. 36-40 ("The design may be changed to use an internal Ethernet fabric which would simplify much of the architecture, e.g., obviating the need for emulation features."). Consequently, the court will not narrow the scope of the "internal communication network."

ORDER

The claim terms at issue will be construed for the jury and for all other purposes in the pending litigation in a manner consistent with the above rulings of the court.

SO ORDERED.

/s/ Richard G. Stearns
____
UNITED STATES DISTRICT JUDGE