

**IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION**

Global Equity Management (SA) Pty. Ltd.,
Plaintiff,

v.

Expedia, Inc., et al.,
Defendants.

Case No. 2:16-cv-00095-RWS-RSP

CLAIM CONSTRUCTION MEMORANDUM OPINION AND ORDER

Before the Court is the opening claim construction brief of Global Equity Management (SA) Pty. Ltd. (“Plaintiff”) (Dkt. No. 169, filed on September 27, 2016),¹ the response of Airbnb, Inc., eBay, Inc., TripAdvisor LLC, Alibaba.com Hong Kong Limited, Alibaba.com, Inc., Alibaba.com Singapore E-Commerce Private Limited, Booking.com B.V., Expedia, Inc., Hotels.com LP, Hotwire, Inc., Orbitz Worldwide, Inc., and CruiseShipCenters International Inc. (collectively “Defendants”) (Dkt. No. 186, filed on October 13, 2016), and the reply of Plaintiff (Dkt. No. 189, filed on October 20, 2016). The Court held a hearing on the issues of claim construction and claim definiteness on November 8, 2016. Having considered the arguments and evidence presented by the parties at the hearing and in their briefing, the Court issues this Order.

¹ Citations to the parties’ filings are to the filing’s number in the docket (Dkt. No.) and pin cites are to the page numbers assigned through ECF.

Table of Contents

- I. BACKGROUND 4**
 - A. U.S. Patent No. 6,690,400..... 4
 - B. U.S. Patent No. 7,356,677..... 8
- II. LEGAL PRINCIPLES 10**
 - A. Claim Construction 10
 - B. Departing from the Ordinary Meaning of a Claim Term..... 13
 - C. Functional Claiming and 35 U.S.C. § 112, ¶ 6 (pre-AIA) / § 112(f) (AIA) 14
 - D. Definiteness Under 35 U.S.C. § 112, ¶ 2 (pre-AIA) / § 112(b) (AIA) 16
- III. AGREED CONSTRUCTIONS 17**
- IV. CONSTRUCTION OF DISPUTED TERMS..... 19**
 - A. U.S. Patent No. 6,690,400..... 19
 - A-1. preambles of Claims 1 and 16 and “means for allocating a computer device’s resources ... on said computer device” 19
 - A-2. “cabinet selection button bar” 27
 - A-3. “virtual cabinet,” “virtual cabinet record,” and “cabinet record” 30
 - A-4. “secondary storage,” and “secondary storage device” 33
 - A-5. “secondary storage partitions window” 37
 - A-6. “cabinet visible partition window” 39
 - A-7. “partition” 43
 - A-8. “means for configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window,” and “program code for configuring said at least one partition of said at least one secondary storage device through a secondary storage partitions window” 49
 - A-9. “means for manipulating said at least one cabinet record through said cabinet visible partition window,” and “program code for manipulating said virtual cabinet record through said cabinet visible partition window” 55
 - A-10. “means for modifying said at least one cabinet record through said cabinet visible partition window,” and “program code for means for modifying said at least one cabinet record through said cabinet visible partition window” 58
 - B. U.S. Patent No. 7,356,677..... 62
 - B-1. Claim 1 preamble and “said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications” 62
 - B-2. “virtual table of contents” 65

B-3.	“partition”	67
B-4.	“virtual computer system”	68
B-5.	“BIOS ACPI solutions” and “BIOS ACPI enhancements”	70
B-6.	“power-on self test (POST) in the BIOS” and “power-on self test (POST) in BIOS”	73
B-7.	“suspend for fast switching,” “fast suspending,” “fast switching,” and “power save suspend”	77
B-8.	“at least one cabinet for isolating each virtual computer system”	83
B-9.	“means for selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system” and “selection means for sequentially choosing from among said plurality of operating systems”.....	86
B-10.	“means for suspending the currently operational virtual computer system in an active state”	90
B-11.	“means for making the selected virtual computer system operable into a running state”	93
B-12.	“means for switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend”	97

V. CONCLUSION 99

I. BACKGROUND

Plaintiff alleges infringement of two U.S. Patents: No. 6,690,400 (the “’400 Patent”) and No. 7,356,677 (the “’677 Patent”) (collectively, the “Asserted Patents”). The ’400 Patent is entitled “Graphic User Interface for Resources Management of Super Operating System Based Computers.” The application leading to the ’400 Patent was filed on September 29, 1999 and the patent issued on February 10, 2004. The ’677 Patent is entitled “Computer System Capable of Fast Switching Between Multiple Operating Systems and Applications.” The application leading to the ’677 Patent was filed on October 18, 2002 and the patent issued on April 8, 2008. The ’677 Patent claims priority to an application filed on October 19, 2001.

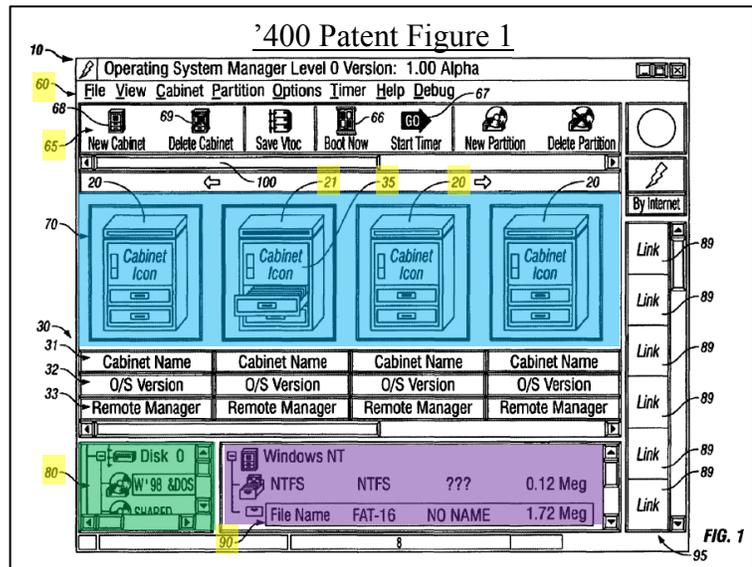
The Asserted Patents are generally directed to technology for managing computer operating-system environments.

A. U.S. Patent No. 6,690,400

In general, the ’400 Patent is generally directed to technology for graphically allocating computer resources to one or more operating-system environments. The technology can be generally understood with reference to Figure 1, reproduced here and annotated by the Court.

Figure 1 depicts a Graphic User Interface (GUI). The GUI includes a number of interactive

features, such as a Cabinet Selection Button Bar (70, in blue), a Secondary Storage Partitions Window (80, in green), and an Active Selected Cabinet Visible Partition Window (90, in purple). ’400 Patent col.5 ll.53–59. The Cabinet Selection Button Bar includes buttons (20, 21)

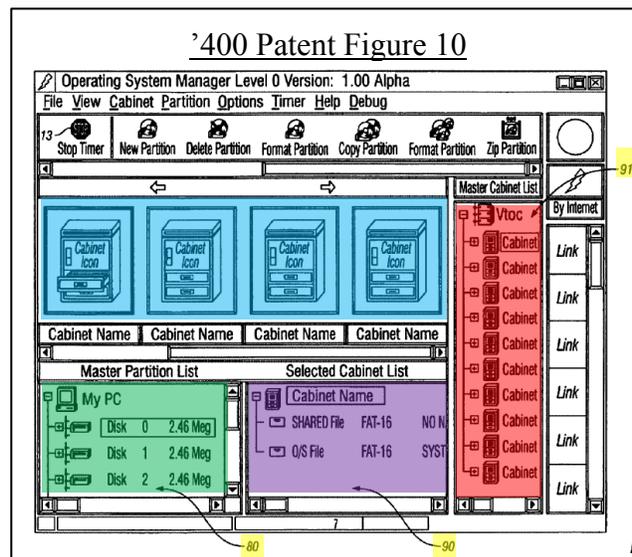


representing virtual storage devices (cabinets) comprising logical storage devices (partitions) constructed from secondary storage such as hard-disk drives. *Id.* at col.1 ll.57–67, col.2 l.64 – col.3 l.12, col.6 ll.13–21. The Secondary Storage Partitions Window displays details of secondary storage devices (e.g., disks) and their partitions. *Id.* at figs.1, 8, 10, col.7 ll.1–32. The Active Selected Cabinet Visible Partition Window displays details of the selected cabinet (21) on the Cabinet Selection Button Bar. *Id.* The GUI is configurable by the user. For example, the icons (35) on the cabinet buttons (20, 21) may be edited. *Id.* at fig.6, col.6 ll.56–63.

The GUI provides an interface to create and modify partitions. For instance, partition functions can be accessed through a “Partition” menu on the Main Pull Down Menu Bar (60). The functions may also be accessed through various buttons on the Main Toolbar (65). *Id.* at fig.13, col.8 ll.7–12. By accessing functions through the GUI, for example, the user may change the size of a partition or format a partition. *Id.* at figs.16, 17, col.8 ll.13–35.

The GUI also provides an interface to create and modify cabinets. *Id.* at col.6 l.6 – col.8 l.6. For instance, cabinet functions can be accessed through a “Cabinet” menu on the Main Pull Down Menu Bar (60), through various buttons on the Main Toolbar (65), or by “right clicking.” *Id.* at col.7 ll.56 – col.8 l.6. The patent explains that partitions can be added to or removed from a cabinet through pointer-device operations such

as “click and drag” and “double clicking.” This can be understood with reference to Figure 10, reproduced here and annotated by the Court. A partition in a secondary storage device listed in Secondary Storage Partitions Window (80, in green) may be added to the active cabinet by



selecting the partition with the pointer and dragging the partition to the Active Selected Cabinet Visible Partition Window (90, in purple). Alternatively, the partition may be dragged to a cabinet listed in the Master Cabinet Visible Partition Window (91, in red). *Id.* at figs.1, 8–10, col.7 ll.1–33.

Ultimately, the contents of the active cabinet (“i.e., the list of Partitions within that Cabinet”) replace the partition information for the computer’s secondary storage. *Id.* at col.3 ll.2–7. The computer thus views the information from the cabinet as the list of available secondary storage partitions. If the cabinet includes a bootable partition, the computer will load information from the cabinet, such as the operating system and the list of available secondary storage, into main memory during the boot process (bootstrap). *Id.* at col.1 l.36 – col.2 l.23, col.2 l.40 – col.3 l.12.

The abstract of the ’400 Patent provides:

This invention is a Graphic User Interface (GUI) that enables a user to virtualize the system and to define secondary storage physical devices through the graphical depiction of cabinets. The GUI allows the user to assign each cabinet a name, and to define the cabinet by its software, which may include single or multiple operating systems, programs and/or data files. The user is also allowed to manipulate (format, copy, resize, delete, zip) memory partitions in the secondary storage physical devices. The GUI also features graphically editable Internet hyperlinks for communication or remote management. Also, this invention uses a combination of Flash VOS VTOC and ACPI to perform “Cold Swaps” or “Context Switching”, which remove one active OS temporarily from all or part of memory and replace it with another active OS in all or part of memory. Information can be shared by multiple Operating Systems through the defined access to Shared Devices or Shared Partitions. The GUI can be used on a variety of computer systems, including multiple operating system and super operating system based computers.

Claims 1, 16, and 28 of the ’400 Patent, the independent claims, recite as follows:

1. A graphic user interface for displaying means for allocating a computer device’s resources to multiple operating system environments, partitioned on individual virtual cabinets, on said computer device, said graphic user interface comprising:
 - a main menu bar;
 - a cabinet selection button bar;

said cabinet selection button bar graphically representing at least one virtual cabinet;
each said at least one virtual cabinet representing a discrete operating system;
a secondary storage partitions window;
a cabinet visible partition window;
said secondary storage partitions window graphically illustrating at least one partition of at least one secondary storage device;
said cabinet visible partition window graphically illustrating a cabinet record corresponding to a selected virtual cabinet on said cabinet selection button bar; and
each said at least one cabinet visible partition window representing an operating system plus application software, databases and memory configured with said selected virtual cabinet.

16. A graphic user interface for displaying means for allocating a computer device's resources to at least one operating system on said computer device, said graphic user interface comprising:

a secondary storage partitions window for graphically illustrating each of at least one partition of at least one secondary storage device for each of at least one operating systems on said computer;
means for configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window;
a cabinet selection button bar;
said cabinet selection button bar graphically representing at least one virtual cabinet record;
each said at least one virtual cabinet record representing a discrete operating system;
a cabinet visible partition window for graphically illustrating a selected virtual cabinet record;
means for manipulating said at least one cabinet record through said cabinet visible partition window;
said cabinet visible partition window graphically illustrating an operating system plus application software, databases and memory configured with said selected virtual cabinet; and means for modifying said at least one cabinet record through said cabinet visible partition window.

28. A computer program product for use on a computer system with a memory, a display and multiple operating system, the computer program product comprising a computer usable medium having computer readable program code thereon for generating, a graphic user interface on the display device which facilitates manipulation of operating systems, programs and databases in said multiple operating system, the computer readable program code comprising:

program code for accessing and displaying each of at least one partition of at least one secondary storage device;
program code for configuring said at least one partition of said at least one secondary storage device through a secondary storage partitions window;
program code for displaying a cabinet selection button bar;

said cabinet selection button bar graphically representing at least one virtual cabinet record;
each said at least one virtual cabinet record representing a discrete operating system;
program code for displaying a cabinet visible partition window for graphically illustrating, at least one cabinet record, each of said at least one cabinet record representing an operating system plus application software, databases and memory configured with said selected virtual cabinet record;
program code for manipulating said virtual cabinet record through said cabinet visible partition window; and
program code for means for modifying said at least one cabinet record through said cabinet visible partition window.

B. U.S. Patent No. 7,356,677

The '677 Patent is generally directed to technology for switching between operating-system environments using suspend and resume capabilities rather than a full shut down and cold reboot. The patent's invention provides Fast Resume and Fast Suspend methods used to accomplish this Fast Switching. Each method uses the power-management functions of a modified BIOS. The Fast Resume uses the modified BIOS to load a saved operating state associated with a particular operating-system environment without shutting the system down and going through a cold boot and the associated Power-On Self-Test (POST). The Fast Suspend method uses the modified BIOS to shutdown an operating-system environment to a suspended state (saving context) instead of shutting down via the "customary and systematic termination of all services." '677 Patent col.3 ll.17–50, col.4 ll.33–55, col.5 ll.4–14, col.6 ll.39–57.

The abstract of the '677 Patent provides:

A method and apparatus is presented that allows rapid switching between multiple operating system environments on a single computer, through the use of a Super Operating System operating between the computer system's firmware level and a plurality of bootable operating systems and applications. Relevant data, such as hardware controls, bootable operating systems, and applications software, are grouped onto physical partitions in physical memory. Combinations of compatible partitions can be assigned to a cabinet to create a virtual computer system. A given partition is assignable to multiple cabinets. By operating above the virtual computer systems, the Super Operating System uses the suspend and resume functions of the power management support functions to suspend and hibernate [sic] one operating

virtual computer system, while activating and operating an alternate virtual computer system on the same computer.

Claims 1, 3, and 6 of the '677 Patent, the independent claims, recite as follows:

1. A hardware platform for a hibernate capable computer system comprising a system manager, said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications, said storage manager having a virtual table of contents for organizing and accessing a plurality of partitions of relevant data and having a plurality of virtual computer systems, each of said virtual computer systems capable of accessing a selection of the partitions, the virtual table of contents (VTOC) being capable of dynamically configuring a plurality of partition tables, said system manager comprising:

means for selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system;

means for suspending the currently operational virtual computer system in an active state;

means for making the selected virtual computer system operable into a running state, and

means for switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend.

3. A method of managing a computer system having a plurality of operating systems, a plurality of virtual computer systems, and a virtual table of contents (VTOC) capable of dynamically configuring a plurality of partition tables for fast switching between the virtual computer systems, each of said operating systems configured on a corresponding one of said virtual computer systems and operable within the corresponding one of said virtual computer systems, said method comprising:

providing a selection means for sequentially choosing from among said plurality of operating systems;

initiating a first selected operating system;

fast suspending said first selected operating system; and

executing a subsequent selected operating system;

said method utilizing a switch flag and BIOS ACPI enhancements without requiring initialization of power-on self-test (POST) in BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend.

6. A method of managing a computer system and fast switching between operating systems, the computer system having a plurality of operating systems and a plurality of virtual computer systems and at least one cabinet for isolating

each virtual computer system from other virtual computer systems, each of said operating systems configured on a corresponding said virtual computer system, said method comprising:

- a fast suspending step comprising fast suspending one of said virtual computer systems in a suspended state;
- an activating step comprising activating another virtual computer system of said virtual computer systems to a running state without rebooting and without initialization of power-on self test in BIOS;
- said fast suspending step and said activating step using a switch flag, a virtual table of contents (VTOC) capable of dynamically configuring a plurality of partition tables, and BIOS ACPI enhancements, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend.

II. LEGAL PRINCIPLES

A. Claim Construction

“It is a ‘bedrock principle’ of patent law that ‘the claims of a patent define the invention to which the patentee is entitled the right to exclude.’” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (quoting *Innova/Pure Water Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)). To determine the meaning of the claims, courts start by considering the intrinsic evidence. *Id.* at 1313; *C.R. Bard, Inc. v. U.S. Surgical Corp.*, 388 F.3d 858, 861 (Fed. Cir. 2004); *Bell Atl. Network Servs., Inc. v. Covad Commc’ns Group, Inc.*, 262 F.3d 1258, 1267 (Fed. Cir. 2001). The intrinsic evidence includes the claims themselves, the specification, and the prosecution history. *Phillips*, 415 F.3d at 1314; *C.R. Bard, Inc.*, 388 F.3d at 861. The general rule—subject to certain specific exceptions discussed *infra*—is that each claim term is construed according to its ordinary and accustomed meaning as understood by one of ordinary skill in the art at the time of the invention in the context of the patent. *Phillips*, 415 F.3d at 1312–13; *Alloc, Inc. v. Int’l Trade Comm’n*, 342 F.3d 1361, 1368 (Fed. Cir. 2003); *Azure Networks, LLC v. CSR PLC*, 771 F.3d 1336, 1347 (Fed. Cir. 2014) (“There is a heavy presumption that claim terms carry their accustomed meaning in the relevant community at the relevant time.”) (vacated on other grounds).

“The claim construction inquiry ... begins and ends in all cases with the actual words of the claim.” *Renishaw PLC v. Marposs Societa’ per Azioni*, 158 F.3d 1243, 1248 (Fed. Cir. 1998). “[I]n all aspects of claim construction, ‘the name of the game is the claim.’” *Apple Inc. v. Motorola, Inc.*, 757 F.3d 1286, 1298 (Fed. Cir. 2014) (quoting *In re Hiniker Co.*, 150 F.3d 1362, 1369 (Fed. Cir. 1998)). First, a term’s context in the asserted claim can be instructive. *Phillips*, 415 F.3d at 1314. Other asserted or unasserted claims can also aid in determining the claim’s meaning, because claim terms are typically used consistently throughout the patent. *Id.* Differences among the claim terms can also assist in understanding a term’s meaning. *Id.* For example, when a dependent claim adds a limitation to an independent claim, it is presumed that the independent claim does not include the limitation. *Id.* at 1314–15.

“[C]laims ‘must be read in view of the specification, of which they are a part.’” *Id.* (quoting *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 979 (Fed. Cir. 1995) (en banc)). “[T]he specification ‘is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.’” *Id.* (quoting *Vitronics Corp. v. Conceptoronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996)); *Teleflex, Inc. v. Ficosa N. Am. Corp.*, 299 F.3d 1313, 1325 (Fed. Cir. 2002). But, “[a]lthough the specification may aid the court in interpreting the meaning of disputed claim language, particular embodiments and examples appearing in the specification will not generally be read into the claims.” *Comark Commc’ns, Inc. v. Harris Corp.*, 156 F.3d 1182, 1187 (Fed. Cir. 1998) (quoting *Constant v. Advanced Micro-Devices, Inc.*, 848 F.2d 1560, 1571 (Fed. Cir. 1988)); *see also Phillips*, 415 F.3d at 1323. “[I]t is improper to read limitations from a preferred embodiment described in the specification—even if it is the only embodiment—into the claims absent a clear indication in the intrinsic record that the

patentee intended the claims to be so limited.” *Liebel-Flarsheim Co. v. Medrad, Inc.*, 358 F.3d 898, 913 (Fed. Cir. 2004).

The prosecution history is another tool to supply the proper context for claim construction because, like the specification, the prosecution history provides evidence of how the U.S. Patent and Trademark Office (“PTO”) and the inventor understood the patent. *Phillips*, 415 F.3d at 1317. 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.” *Id.* at 1318; *see also Athletic Alternatives, Inc. v. Prince Mfg.*, 73 F.3d 1573, 1580 (Fed. Cir. 1996) (ambiguous prosecution history may be “unhelpful as an interpretive resource”).

Although extrinsic evidence can also be useful, it is “less significant than the intrinsic record in determining the legally operative meaning of claim language.” *Phillips*, 415 F.3d at 1317 (quoting *C.R. Bard, Inc.*, 388 F.3d at 862). Technical dictionaries and treatises may help a court understand the underlying technology and the manner in which one skilled in the art might use claim terms, but technical dictionaries and treatises may provide definitions that are too broad or may not be indicative of how the term is used in the patent. *Id.* at 1318. Similarly, expert testimony may aid a court in understanding the underlying technology and determining the particular meaning of a term in the pertinent field, but an expert’s conclusory, unsupported assertions as to a term’s definition are not helpful to a court. *Id.* Extrinsic evidence is “less reliable than the patent and its prosecution history in determining how to read claim terms.” *Id.* The Supreme Court recently explained the role of extrinsic evidence in claim construction:

In some cases, however, the district court will need to look beyond the patent’s intrinsic evidence and to consult extrinsic evidence in order to understand, for example, the background science or the meaning of a term in the relevant art during the relevant time period. *See, e.g., Seymour v. Osborne*, 11 Wall. 516, 546 (1871)

(a patent may be “so interspersed with technical terms and terms of art that the testimony of scientific witnesses is indispensable to a correct understanding of its meaning”). In cases where those subsidiary facts are in dispute, courts will need to make subsidiary factual findings about that extrinsic evidence. These are the “evidentiary underpinnings” of claim construction that we discussed in *Markman*, and this subsidiary factfinding must be reviewed for clear error on appeal.

Teva Pharm. USA, Inc. v. Sandoz, Inc., 135 S. Ct. 831, 841 (2015).

B. Departing from the Ordinary Meaning of a Claim Term

There are “only two exceptions to [the] general rule” that claim terms are construed according to their plain and ordinary meaning: “1) when a patentee sets out a definition and acts as his own lexicographer, or 2) when the patentee disavows the full scope of the claim term either in the specification or during prosecution.”² *Golden Bridge Tech., Inc. v. Apple Inc.*, 758 F.3d 1362, 1365 (Fed. Cir. 2014) (quoting *Thorner v. Sony Computer Entm’t Am. LLC*, 669 F.3d 1362, 1365 (Fed. Cir. 2012)); *see also GE Lighting Solutions, LLC v. AgiLight, Inc.*, 750 F.3d 1304, 1309 (Fed. Cir. 2014) (“[T]he specification and prosecution history only compel departure from the plain meaning in two instances: lexicography and disavowal.”). The standards for finding lexicography or disavowal are “exacting.” *GE Lighting Solutions*, 750 F.3d at 1309.

To act as his own lexicographer, the patentee must “clearly set forth a definition of the disputed claim term,” and “clearly express an intent to define the term.” *Id.* (quoting *Thorner*, 669 F.3d at 1365); *see also Renishaw*, 158 F.3d at 1249. The patentee’s lexicography must appear “with reasonable clarity, deliberateness, and precision.” *Renishaw*, 158 F.3d at 1249.

To disavow or disclaim the full scope of a claim term, the patentee’s statements in the specification or prosecution history must amount to a “clear and unmistakable” surrender. *Cordis*

² Some cases have characterized other principles of claim construction as “exceptions” to the general rule, such as the statutory requirement that a means-plus-function term is construed to cover the corresponding structure disclosed in the specification. *See, e.g., CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1367 (Fed. Cir. 2002).

Corp. v. Boston Sci. Corp., 561 F.3d 1319, 1329 (Fed. Cir. 2009); *see also Thorner*, 669 F.3d at 1366 (“The patentee may demonstrate intent to deviate from the ordinary and accustomed meaning of a claim term by including in the specification expressions of manifest exclusion or restriction, representing a clear disavowal of claim scope.”). “Where an applicant’s statements are amenable to multiple reasonable interpretations, they cannot be deemed clear and unmistakable.” *3M Innovative Props. Co. v. Tredegar Corp.*, 725 F.3d 1315, 1326 (Fed. Cir. 2013).

C. Functional Claiming and 35 U.S.C. § 112, ¶ 6 (pre-AIA) / § 112(f) (AIA)³

A patent claim may be expressed using functional language. *See* 35 U.S.C. § 112, ¶ 6; *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1347–49 & n.3 (Fed. Cir. 2015) (en banc in relevant portion). Section 112, Paragraph 6, provides that a structure may be claimed as a “means ... for performing a specified function” and that an act may be claimed as a “step for performing a specified function.” *Masco Corp. v. United States*, 303 F.3d 1316, 1326 (Fed. Cir. 2002).

But § 112, ¶ 6 does not apply to all functional claim language. There is a rebuttable presumption that § 112, ¶ 6 applies when the claim language includes “means” or “step for” terms, and that it does not apply in the absence of those terms. *Masco Corp.*, 303 F.3d at 1326; *Williamson*, 792 F.3d at 1348. The presumption stands or falls according to whether one of ordinary skill in the art would understand the claim with the functional language, in the context of the entire specification, to denote sufficiently definite structure or acts for performing the function. *See Media Rights Techs., Inc. v. Capital One Fin. Corp.*, 800 F.3d 1366, 1372 (Fed. Cir. 2015) (§ 112, ¶ 6 does not apply when “the claim language, read in light of the specification, recites sufficiently definite structure” (quotation marks omitted) (citing *Williamson*, 792 F.3d at 1349;

³ The Court refers to the pre-AIA version of § 112 but understands that there is no substantial difference between functional claiming under the pre-AIA version and under the AIA version of the statute.

Robert Bosch, LLC v. Snap-On Inc., 769 F.3d 1094, 1099 (Fed. Cir. 2014)); *Williamson*, 792 F.3d at 1349 (§ 112, ¶ 6 does not apply when “the words of the claim are understood by persons of ordinary skill in the art to have sufficiently definite meaning as the name for structure”); *Masco Corp.*, 303 F.3d at 1326 (§ 112, ¶ 6 does not apply when the claim includes an “act” corresponding to “how the function is performed”); *Personalized Media Communications, L.L.C. v. International Trade Commission*, 161 F.3d 696, 704 (Fed. Cir. 1998) (§ 112, ¶ 6 does not apply when the claim includes “sufficient structure, material, or acts within the claim itself to perform entirely the recited function ... even if the claim uses the term ‘means.’” (quotation marks and citation omitted)).

When it applies, § 112, ¶ 6 limits the scope of the functional term “to only the structure, materials, or acts described in the specification as corresponding to the claimed function and equivalents thereof.” *Williamson*, 792 F.3d at 1347. Construing a means-plus-function limitation involves multiple steps. “The first step ... is a determination of the function of the means-plus-function limitation.” *Medtronic, Inc. v. Advanced Cardiovascular Sys., Inc.*, 248 F.3d 1303, 1311 (Fed. Cir. 2001). “[T]he next step is to determine the corresponding structure disclosed in the specification and equivalents thereof.” *Id.* A “structure disclosed in the specification is ‘corresponding’ structure only if the specification or prosecution history clearly links or associates that structure to the function recited in the claim.” *Id.* The focus of the “corresponding structure” inquiry is not merely whether a structure is capable of performing the recited function, but rather whether the corresponding structure is “clearly linked or associated with the [recited] function.” *Id.* The corresponding structure “must include all structure that actually performs the recited function.” *Default Proof Credit Card Sys. v. Home Depot U.S.A., Inc.*, 412 F.3d 1291, 1298 (Fed. Cir. 2005). However, § 112 does not permit “incorporation of structure from the written

description beyond that necessary to perform the claimed function.” *Micro Chem., Inc. v. Great Plains Chem. Co.*, 194 F.3d 1250, 1258 (Fed. Cir. 1999).

For § 112, ¶ 6 limitations implemented by a programmed general purpose computer or microprocessor, the corresponding structure described in the patent specification must include an algorithm for performing the function. *WMS Gaming Inc. v. Int’l Game Tech.*, 184 F.3d 1339, 1349 (Fed. Cir. 1999). The corresponding structure is not a general purpose computer but rather the special purpose computer programmed to perform the disclosed algorithm. *Aristocrat Techs. Austl. Pty Ltd. v. Int’l Game Tech.*, 521 F.3d 1328, 1333 (Fed. Cir. 2008).

D. Definiteness Under 35 U.S.C. § 112, ¶ 2 (pre-AIA) / § 112(b) (AIA)⁴

Patent claims must particularly point out and distinctly claim the subject matter regarded as the invention. 35 U.S.C. § 112, ¶ 2. A claim, when viewed in light of the intrinsic evidence, must “inform those skilled in the art about the scope of the invention with reasonable certainty.” *Nautilus Inc. v. Biosig Instruments, Inc.*, 134 S. Ct. 2120, 2129 (2014). If it does not, the claim fails § 112, ¶ 2 and is therefore invalid as indefinite. *Id.* at 2124. Whether a claim is indefinite is determined from the perspective of one of ordinary skill in the art as of the time the application for the patent was filed. *Id.* at 2130. As it is a challenge to the validity of a patent, the failure of any claim in suit to comply with § 112 must be shown by clear and convincing evidence. *Id.* at 2130 n.10. “[I]ndefiniteness is a question of law and in effect part of claim construction.” *ePlus, Inc. v. Lawson Software, Inc.*, 700 F.3d 509, 517 (Fed. Cir. 2012).

When a term of degree is used in a claim, “the court must determine whether the patent provides some standard for measuring that degree.” *Biosig Instruments, Inc. v. Nautilus, Inc.*, 783

⁴ The Court refers to the pre-AIA version of § 112 but understands that there is no substantial difference between definiteness under the pre-AIA version and under the AIA version of the statute.

F.3d 1374, 1378 (Fed. Cir. 2015) (quotation marks omitted). Likewise, when a subjective term is used in a claim, “the court must determine whether the patent’s specification supplies some standard for measuring the scope of the [term].” *Datamize, LLC v. Plumtree Software, Inc.*, 417 F.3d 1342, 1351 (Fed. Cir. 2005); *accord Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364, 1371 (Fed. Cir. 2014) (citing *Datamize*, 417 F.3d at 1351).

In the context of a claim governed by 35 U.S.C. § 112, ¶ 6, the claim is invalid as indefinite if the claim fails to disclose adequate corresponding structure to perform the claimed functions. *Williamson*, 792 F.3d at 1351–52. The disclosure is inadequate when one of ordinary skill in the art “would be unable to recognize the structure in the specification and associate it with the corresponding function in the claim.” *Id.* at 1352.

III. AGREED CONSTRUCTIONS

The parties have agreed to the following constructions set forth in their Joint Claim Construction Chart (Dkt. No. 197).

Term ⁵	Agreed Construction
“computer system with a memory, a display and multiple operating systems” <ul style="list-style-type: none"> • ’400 Patent Claim 28 	plain and ordinary meaning
“operating system[s]” <ul style="list-style-type: none"> • ’400 Patent Claims 1, 16, 28 • ’677 Patent Claims 1, 3, 6 	plain and ordinary meaning

⁵ For all term charts in this order, the claims in which the term is found are listed with the term but: (1) only the highest-level claim in each dependency chain is listed, and (2) only asserted claims identified in the parties’ Joint Claim Construction Chart (Dkt. No. 197) are listed. The parties filed three updated version of the claims construction chart: Plaintiff’s Errata to Joint Claim Construction Chart (Dkt. No. 203, filed November 2, 2016), Defendants’ Notice of Correction (Dkt. No. 205, filed November 4, 2016), and Plaintiff’s Errata to Joint Claim Construction Chart (Dkt. No. 208, filed November 8, 2016). The parties’ proposed claim constructions listed in this Order are from the latest-filed charts.

Term⁵	Agreed Construction
“discrete operating system” <ul style="list-style-type: none"> • ’400 Patent Claims 1, 16, 28 	plain and ordinary meaning
“a computer program product for use on a computer system” <ul style="list-style-type: none"> • ’400 Patent Claim 28 	plain and ordinary meaning
“database” <ul style="list-style-type: none"> • ’400 Patent Claims 1, 16, 28 	a collection of data organized by a database management system for easy retrieval
“hardware platform” <ul style="list-style-type: none"> • ’677 Patent Claim 1 	plain and ordinary meaning
“hibernate capable computer system” <ul style="list-style-type: none"> • ’677 Patent Claim 1 	plain and ordinary meaning
“partitions of relevant data” <ul style="list-style-type: none"> • ’677 Patent Claim 1 	plain and ordinary meaning
“dynamically configuring a plurality of partition tables” <ul style="list-style-type: none"> • ’677 Patent Claim 1, 3, 6 	plain and ordinary meaning
“sequentially choosing” <ul style="list-style-type: none"> • ’677 Patent Claim 3 	plain and ordinary meaning
“activating another virtual computer system of said virtual computer systems to a running state without rebooting and without initialization of power-on self test in BIOS” <ul style="list-style-type: none"> • ’677 Patent Claim 6 	plain and ordinary meaning

With the understanding that “plain and ordinary” meaning refers to the ordinary meaning in the art, and having reviewed the intrinsic and extrinsic evidence of record, the Court agrees with and hereby adopts the parties’ agreed constructions.

IV. CONSTRUCTION OF DISPUTED TERMS

A. U.S. Patent No. 6,690,400

A-1. preambles of Claims 1 and 16 and “means for allocating a computer device’s resources ... on said computer device”⁶

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“A graphic user interface for displaying means for allocating a computer device’s resources to multiple operating system environments, partitioned on individual virtual cabinets, on said computer device, said graphic user interface comprising”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 1 	<p>a graphic user interface</p> <p>No further construction is necessary.</p>	<p>Defendants no longer seek construction of the preamble phrase “graphic user interface,” but only seek construction of the means-plus-function limitations contained therein.</p>
<p>“means for allocating a computer device’s resources to multiple operating system environments, partitioned on individual virtual cabinets, on said computer device”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 1 	<p>No construction necessary as part of non-limiting preamble.</p> <p>Alternatively:</p> <ul style="list-style-type: none"> • more than one virtual storage device capable of containing virtualized operating systems, application software, databases and memory, or partitions thereof displayed on a graphical user interface, on the computer device 	<p>35 U.S.C. §112, ¶ 6</p> <p>Function:</p> <ul style="list-style-type: none"> • allocating a computer device’s resources to multiple operating system environments, partitioned on individual virtual cabinets, on said computer device <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claims 1 and 16 are invalid under 35 U.S.C. § 112

⁶ It appears that the parties’ positions changed in the time between Plaintiff’s opening brief and the parties’ submission of their Joint Claim Construction Chart submitted pursuant to P.R. 4-5(d). See Dkt. No. 169 at 8–10; Dkt. No. 197-1 at 1–8. It is unclear from the Joint Claim Construction Chart whether there remains a dispute with respect to Claim 16’s preamble. Defendants’ position on Claim 1 as stated in the P.R. 4-5(d) chart notes that Claim 16 is invalid because of the “means for allocating . . .” limitation, but there is no such position stated in the Claim 16 entry. See Dkt. No. 197-1 at 1–5. Because of the similarities between the Claim 1 preamble and the Claim 16 preamble, the Court here considers both preambles, and the “means for allocating . . .” language in the preambles.

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“A graphic user interface for displaying m[e]ans for allocating a computer device’s resources to at least one operating system on said computer device, said graphic user interface comprising”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 16 	-	-
<p>“m[e]ans for allocating a computer device’s resources to at least one operating system on said computer device”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 16 	-	<p>35 U.S.C. §112, ¶ 6</p> <p>Function:</p> <ul style="list-style-type: none"> • allocating a computer device’s resources to multiple operating system environments, partitioned on individual virtual cabinets, on said computer device. <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claims 1 and 16 are invalid under 35 U.S.C. § 112.

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits that the preambles of Claims 1 and 16 are not limiting. Dkt. No. 169 at 9. Plaintiff further submits that the Court should not construe the “means for allocating ...” terms for a variety of reasons: First, indefiniteness, as an issue of validity, should be raised in summary judgment, not in claim construction, *Id.* at 10–11. Second, the terms are part of terms which are separately being construed. *Id.* at 11. (3) Third, the terms are part of preambles that are not limiting. *Id.* at 11. Fourth, construction of the term would result in inferential claiming of a positive limitation, which Plaintiff contends is improper under patent office practice. *Id.* at 11–12 and

nn.22–24 (citing 37 C.F.R. § 1.75(i) and *Metrologic Instruments, Inc. v. PSC Inc.*, No. 99-4876 (JBS), 2003 U.S. Dist. LEXIS 26636, at *13 (D.N.J. Aug. 26, 2003)). Fifth, Defendants’ have not shown that there is a need to construe the terms to resolve a “specific and concrete infringement or invalidity dispute.” *Id.* at 12–13 (citing *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999)).

Plaintiff further submits that the preamble language “graphic user interface for displaying means for allocating ...” is the graphic user interface (GUI) of the claims acting “as a means of displaying for the purposes of allocating.” *Id.* at 13. And a GUI is known in the art as “capable of displaying information.” *Id.* at 13 & n.30. Thus, Plaintiff contends, the “means for allocating ...” terms are not governed by 35 U.S.C. § 112. *See id.* at 13–14.

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent, at [57] Abstract, col.1 l.1 – col.5 l.5, col.2 ll.40–52, col.5 ll.9–29, col.5 l.60 – col.6 l.12. **Extrinsic evidence:** U.S. Patent No. 6,463,460; U.S. Patent No. 6,556,223; U.S. Patent No. 6,654,803; U.S. Patent No. 6,888,565; U.S. Patent No. 6,938,216; Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).⁷

Defendants respond that the preambles of Claims 1 and 16 of the ’400 Patent are limiting for a variety of reasons: First, the preambles recite the “framework of the invention.” Dkt. No. 186 at 9 (citing *On Demand Mach. Corp. v. Ingram Indus.*, 442 F.3d 1331, 1343 (Fed. Cir. 2006)).

⁷ Plaintiff cites “[d]eclaration, generally, and testimony of Dr. Rosenberg.” *See, e.g.*, Dkt. No. 169 at 13 n.27. Plaintiff did not submit any exhibits with its opening claim construction brief (filed September 27, 2016), but submitted exhibits at Dkt. No. 179 (filed October 4, 2016) and with its reply brief at Dkt. No. 189 (filed October 20, 2016). The Court understands that “[d]eclaration . . . of Dr. Rosenberg” refers to Declaration of Craig Rosenberg, Ph.D, Dkt. No. 179 and “testimony of Dr. Rosenberg” refers to the September 30, 2016 Deposition Upon Oral Examination of Craig Rosenberg, Dkt. No. 189-2 (excerpts). In its opening brief, Plaintiff has not, however, provided pin cites to either the 90-paragraph declaration or to the 66 pages of the deposition transcript Plaintiff submitted with its reply.

Second, the preambles were used to overcome a prior art rejection during prosecution. *Id.* at 9 (citing *Rotatable Techs. LLC v. Motorola Mobility LLC*, 567 F. App'x 941, 943 (Fed. Cir. 2014)). Third, with respect to Claim 16, the preamble provides the antecedent basis for the term “computer” recited in the body of the claim. *Id.* (citing *Catalina Mktg. Int'l v. Coolsavings.com, Inc.*, 289 F.3d 801, 808 (Fed. Cir. 2002)). Finally, the preambles “disclose a fundamental characteristic of the claimed invention.” *Id.* at 9–10 n.5 (citing *Poly-Am., L.P. v. GSE Lining Tech., Inc.*, 383 F.3d 1303, 1310 (Fed. Cir. 2004)).

Defendants respond that the function of the “means for allocating ...” terms is “allocating a computer device’s resources ...,” not “displaying for the purposes of allocating” as Plaintiff suggests. *Id.* at 10. Defendants argue that under 35 U.S.C. § 112, ¶ 6, the patent must provide a structure, such as an algorithm, for performing this function. *Id.* at 7–8. And, Defendants posit, an interface “for displaying” does not perform the “allocating” function. *Id.* Defendants contend that “allocating” involves “altering the address boundaries of the physical devices and memory at the firmware level.” *Id.* at 10–11 (citing '400 Patent col.3 ll.33–35). According to Defendants, a GUI “for displaying” an icon that identifies a command to allocate resources is not the requisite structure to actually allocate resources. *Id.* at 11. Thus, Defendants conclude, there is no structure corresponding to the “means for allocating ...” terms, and Claims 1 and 16 and therefore indefinite. *Id.*

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: '400 Patent, at [57] Abstract, col.1 ll.57–61, col.3 ll.30–35, col.3 ll.57–58; '400 Patent File Wrapper September 4, 2002 Response (Defendants' Ex. E, Dkt. No. 186-6).

Plaintiff replies that Defendants “have not proven construction of the terms are necessary” and did not address the impropriety of inferential claiming or the impropriety of construing a term

within another term being construed. Dkt. No. 189 at 1–2 & nn.2–5. Plaintiff also provides pin cites for what it purports are its expert’s identification of structure corresponding to the function of these terms. *Id.* at 2 & n.7. Plaintiff further replies that the preambles are not limiting because: (1) they were not clearly used to distinguish prior art and (2) the “computer” of Claim 16 is not part of the claimed invention, so resorting to the preamble for the antecedent basis is not necessary to give life to the invention. *Id.* at 8–9 (citing *Welding Innovation Sols., Inc. v. Am. Axle & Mfg.*, No. 13-cv-13131, 2015 U.S. Dist. LEXIS 44532, at *14, 17 (E.D. Mich. Apr. 6, 2015)). Plaintiff argues that Defendants have not established that an algorithm is required because they have not established that the claims are directed to a general-purpose computer and that a general-purpose computer cannot perform the recited function. *Id.* at 5.

Plaintiff cites further intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent File Wrapper September 4, 2002 Response (Defendants’ Ex. E, Dkt. No. 186-6). **Extrinsic evidence:** Rosenberg Dep. 13:8–14:20, 15:1–16, 16:7–17:23, 22:13–24, 23:19–25:19, 28:3–29:12, 56:14–22, 58:1–5, 65:23–73:23 (Plaintiff’s Ex. A, Dkt. No. 189-2 at 4–8, 15–19); Rosenberg Decl. ¶¶ 20–24, 28–32 (Plaintiff’s Ex. B, Dkt. No. 189-3 at 5–10).

Analysis

There are two issues raised by the dispute. First, whether the preambles of Claims 1 and 16 are limiting. Second, whether the “means for allocating ...” terms are governed by 35 U.S.C. § 112, ¶ 6. Neither the preambles nor the “means for allocating ...” terms are limiting.

To begin, the Court is obliged to construe these terms. It is true that the Court need not rewrite every claim term in construing a claim. The Federal Circuit has stated as much:

The *Markman* decisions do not hold that the trial judge must repeat or restate every claim term in order to comply with the ruling that claim construction is for the court. Claim construction is a matter of resolution of disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the

claims, for use in the determination of infringement. It is not an obligatory exercise in redundancy.

United States Surgical Corp. v. Ethicon, Inc., 103 F.3d 1554, 1568 (Fed. Cir. 1997). Thus, the Court’s claim construction obligations do not extend to claim terms that are not disputed. *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999) (“only those terms need be construed that are in controversy, and only to the extent necessary to resolve the controversy”). And while the Court may require the parties to show an actual controversy, the Court need not require the parties to muster their invalidity and infringement evidence and present their case theories in order to establish that a dispute exists. Indeed, the Federal Circuit effectively rejected making such an approach mandatory. *Id.* (rejecting a rule requiring the completion of discovery prior to claim construction). The Court has not adopted such an approach here. Ultimately, “[w]hen the parties present a fundamental dispute regarding the scope of a claim term, it is the court’s duty to resolve it.” *O2 Micro Int’l Ltd. v. Beyond Innovation Tech. Co.*, 521 F.3d 1351, 1362 (Fed. Cir. 2008). And this duty extends to determining whether the scope of the claim is reasonably certain. *Nautilus Inc. v. Biosig Instruments, Inc.*, 134 S. Ct. 2120, 2129 (2014) (“we read § 112, ¶ 2 to require that a patent’s claims, viewed in light of the specification and prosecution history, inform those skilled in the art about **the scope of the invention** with reasonable certainty” (emphasis added)); *ePlus, Inc. v. Lawson Software, Inc.*, 700 F.3d 509, 517 (Fed. Cir. 2012) (“indefiniteness is a question of law and in effect part of claim construction”); *Cox Communs., Inc. v. Sprint Commun. Co. LP*, 838 F.3d 1224, 1288–89 (Fed. Cir. 2016) (“an indefiniteness analysis under 35 U.S.C. § 112, ¶ 2 is ‘inextricably intertwined with claim construction’” (quoting *Atmel Corp. v. Info. Storage Devices, Inc.*, 198 F.3d 1374, 1379 (Fed. Cir. 1999))). Accordingly—and with respect to every term in dispute—the Court rejects Plaintiff’s argument that the Court should not

construe the claims because Defendants have not proven there is a “specific and concrete infringement or invalidity dispute” to be resolved by the claim construction.

The preambles are not limiting. The preamble simply states the intended use of the invention. For Claim 1, the intended use is “for displaying means for allocating a computer device’s resources to multiple operating system environments, partitioned on individual virtual cabinets, on said computer device.” For Claim 16, it is “for displaying m[e]ans for allocating a computer device’s resources to at least one operating system on said computer device.” There is nothing in the prosecution history that establishes that the preamble was used to overcome a prior art rejection. To the contrary, the patentee expressly distinguishes the prior art using limitations found in the body of the claim, not using the preamble:

Claim 1 includes a ***cabinet selection button bar*** and a ***cabinet visible partition window***. These elements provide visual depiction of the virtual management of cabinets of the present invention. The prior art does not teach or suggest the limitations these elements support in the current invention.

’400 Patent File Wrapper September 4, 2002 Response 12 (emphasis added), Dkt. No. 186-6 at 13. Thus, the preamble of Claim 1 is not limiting. *See TomTom, Inc. v. Adolph*, 790 F.3d 1315, 1323 (Fed. Cir. 2015) (“a preamble is not limiting where a patentee defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention” (quotation marks omitted)); *Am. Med. Sys. v. Biolitec, Inc.*, 618 F.3d 1354, 1359 (Fed. Cir. 2010) (“If the preamble is reasonably susceptible to being construed to be merely duplicative of the limitations in the body of the claim (and was not clearly added to overcome a prior art rejection), we do not construe it to be a separate limitation.” (quotation and modification marks omitted)).

The reference to “said computer” in Claim 16 does not change the Court’s conclusion that the preamble is not limiting. The only difference between Claims 1 and 16 that is relevant to this

analysis is that the first limitation in the body of Claim 16 recites (with emphasis added): “a secondary storage partitions window for graphically illustrating each of at least one partition of at least one secondary storage device for each of at least one operating systems *on said computer*.” The only “computer” mentioned in the claim before this is the “computer device” listed in the preamble. Because the preamble forms the antecedent basis for terms in the body of the claims, the preamble may be limiting. *See Pacing Techs., LLC v. Garmin Int’l, Inc.*, 778 F.3d 1021, 10224 (Fed. Cir. 2015) (“when limitations in the body of the claim rely upon and derive antecedent basis from the preamble, then the preamble *may* act as a necessary component of the claimed invention” (emphasis added)). Here, however, the computer is a statement of intended environment that is introduced inferentially. *See Faber on Mechanics of Patent Claim Drafting* § 3:4 (6th ed. 2014). Indeed, the “means for allocating ...” terms are similar statements of intended environments. Such environmental elements are not positive limitations. *See, e.g., In re Hutchinson*, 154 F.2d 135, 137–38 (C.C.P.A. 1946) (in the claim language “[a]s an article of manufacture, adapted to be adhered to a metal backing element, a flexible sheet material ...,” the metal backing element is “mentioned inferentially” and is not “positively included”); *In re Otto*, 50 C.C.P.A. 938, 940, 312 F.2d 937, 939 (C.C.P.A. 1963) (“inclusion of the material or article worked upon by a structure being claimed does not impart patentability to the claims”); *Canon Inc. v. GCC Int’l, Ltd.*, 450 F. Supp. 2d 243, 248–49 (S.D.N.Y. 2006) (claim to a cartridge “detachably mountable to a main assembly” via a “projection ... for engagement” with the surfaces of a hole in the assembly “does not mean that the main assembly or any portion thereof is claimed as part of the invention”); *Metrologic Instruments, Inc. v. PSC Inc.*, No. 99-4876 (JBS), 2003 U.S. Dist. LEXIS 26636, at *115 (D.N.J. Aug. 26, 2003) (holding that in “means for ... producing digital data ... for use by

decoder means for decoding” the term “‘decoder means’ is not an additional limitation ... and its claim construction is therefore not at issue”).

Accordingly, the Court determines that the preambles of Claims 1 and 16 are not limiting, the “means for allocating ...” terms are not positive limitations, and construction of these terms is not required.

A-2. “cabinet selection button bar”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“cabinet selection button bar” <ul style="list-style-type: none"> • ’400 Patent Claims 1, 16, 28 	a graphical button bar representing more than one virtual cabinet	a row of user selectable graphical icons, each representing a virtual cabinet

The Parties’ Positions

Plaintiff submits that “cabinet” is otherwise being construed and “selection button bar” does not have a “specified technical meaning” in the art. Dkt. No. 169 at 22. Plaintiff argues that Defendants’ proposed construction is improper as it limits the button bar to a “row” of “icons” and neither limitation is supported by the ’400 Patent.

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent fig.1, col.5 ll.53–59, col.8 ll.56–60. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that the exemplary cabinet selection button bar is depicted as a row of user-selectable icons, where each icon represents a virtual cabinet. Dkt. No. 186 at 16 (citing ’400 Patent fig.1). According to Defendants, the patentee clarified in prosecution that the cabinets are depicted visually with icons and that this was a distinction over the prior art. *Id.* And Defendants argue that Plaintiff’s proposed construction is unhelpful because it fails to explain “what aspect of the bar represents the multiple cabinets.” *Id.* at 16–17.

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: '400 Patent fig.1; '400 Patent File Wrapper September 4, 2002 Response (Defendants' Ex. E, Dkt. No. 186-6).

Plaintiff replies that Defendants “have not proven construction of the terms are necessary.” Dkt. No. 189 at 1 & n.2.

Analysis

The issue here appears to be whether a “cabinet selection button bar” is necessarily a “row of user selectable graphical icons.” It is not.

To begin, the '400 Patent distinguishes between a “button” and an “icon.” They are denoted as separate items in the figures: the icon is item 35 and the buttons are items 20 and 21. '400 Patent fig.1, col.6 ll.13–21, col.6 ll.56–63. The icon is a “graphic on the ... cabinet [] button.” *Id.* at col.6 ll.60–63. That is, a “button” is not an “icon.”

The statements in the prosecution history do not overcome this clear teaching. The relevant section provides:

The “cabinets” referenced in the prior art are distinctly different in definition and function from the cabinets of the present invention. In *Hahn et al*, the cabinets are “much like their physical counterparts.” Column 2, lines 53-54. The images in *Hahn et al*. are file cabinet images, being a graphical depiction as a “metaphor for a physical file cabinet” with graphical “metaphors for physical file folders” and “physical documents.” *See* column 4, lines 24-44; *See also Hahn et al*. Fig. 12A, 21A, 2B, 22A, 22B, 23A, and 23B.

The disclosure of the current invention defines a cabinet as “a virtual storage device, capable of containing, ... all (or partitions of) shared (or non-shared) operating systems, application software (both OS dependent and No-OS embedded), databases and memory.” Page 9, lines 16–19. In other words, a “cabinet”, in the present invention, represents a grouping of all software (an operating system, if necessary, applications, application support files and user data) and software allocable memory necessary to operate the particular computer processes of that partition or grouping. The icons of the present invention depict virtual, individual, stand-alone computer systems, operating in individual operating environments. *See* Fig. 1 (depicting four individual cabinets 20 and 21 containing independent, non-compatible, virtual systems configured for a FlashVos OS,

Windows OS, Linux OS and Novell OS.) The differences in appearance between the prior art and current icons functionally represent the differences in the virtual systems.

'400 Patent File Wrapper September 4, 2002 Response 11–12, Dkt. No. 186-6 at 12–13 (italics and ellipsis in original). Contrary to Defendants' contention, the patentee is not distinguishing the prior art based on the icons, but rather on the very nature of the cabinets. This is not a statement that "icons" are somehow essential to the invention such that "icon" should be read into the claims when not otherwise present. Further, the patentee here notes that the cabinets are depicted by items 20 and 21—buttons. And to the extent this statement equates "icons" of the patent and "buttons" of the patent it is erroneous and should be disregarded. *See Biotec Biologische Naturverpackungen GmbH v. Biocorp, Inc.*, 249 F.3d 1341, 1348 (Fed. Cir. 2001) ("A person of reasonable intelligence would not be misled into relying on the erroneous statement [in the prosecution history], for it is contrary not only to the plain language of the claims and the specification, but also to other statements in the same prosecution document."). Simply, "buttons" are not "icons" in the sense that those words are used in patent.

Defendants also offer no argument or evidence to interpret a "bar" as a "row." Thus, the Court declines to do so.

That said, Plaintiff's proposed construction is unhelpful—and incorrect. It is not the bar that represents the cabinets, it is the buttons that comprise the bar. And the buttons are representative and actionable, they are user-selectable graphics in the GUI that a user can invoke to select the cabinet represented by the button. *See, e.g.*, '400 Patent col.6 ll.13–16 (providing that "Cabinet Buttons 20 represent inactive cabinets" and "Cabinet Button 21 represents an active cabinet"); col.7 ll.34–39 (noting that a boot action occurs "when selected Cabinet Button 21 is double-clicked"). Indeed, "buttons" in the patents are uniformly described as actionable graphic

items. And the “bars” of the patents are collections of actionable items (e.g., buttons and pull-down menus).

Accordingly, the Court construes “cabinet selection button bar” as follows:

- “cabinet selection button bar” means “collection of user-selectable graphical items, each graphical item representing a virtual cabinet.”

A-3. “virtual cabinet,” “virtual cabinet record,” and “cabinet record”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“virtual cabinet” • ’400 Patent Claim 1	graphical representation of virtualized software and/or data	virtual storage device capable of containing all (or partitions of) shared (or non-shared) operating systems, application software (both OS dependent and No-OS embedded), databases and memory
“cabinet record” • ’400 Patent Claims 1, 16, 28		
“virtual cabinet record” • ’400 Patent Claims 16, 28		

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits that these terms are defined synonymously in the ’400 Patent. Dkt. No. 169 at 20 (citing ’400 Patent col.5 ll.22–29). But, Plaintiff argues, defining “virtual cabinet” as a “virtual storage device,” as it is defined in the patent, is unhelpful as it uses the same term, namely, “virtual.” *Id.* at 20–21. Plaintiff contends the patent also explains that the “invention is a Graphic User Interface (GUI) that enables a user to virtualize the system and to define secondary storage physical devices through the graphical depiction of cabinets.” *Id.* at 21 (citing ’400 Patent, at [57])

Abstract). Thus, Plaintiff argues, the “virtual storage devices” are graphical depictions of virtualized software or data. *Id.*

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent, at [57] Abstract, col.5 ll.22–29. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that the terms are defined in the patent as “virtual storage devices capable of containing ... all (or partitions of) shared (or non-shared) operating systems, application software (both OS dependent and No-OS embedded), databases and memory.” Dkt. No. 186 at 17. Defendants argue that Plaintiff’s proposed construction does not address Plaintiff’s purported concern with the phrase “virtual storage device” in that the proposed construction includes the term “virtualized.” *Id.* Furthermore, Defendants argue that Plaintiff’s proposed construction omits the “*all* (or partitions of) ...” language from the patent’s definition. *Id.* (emphasis by Defendants).

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’400 Patent col.2 ll.46–47, col.5 ll.22–29.

Analysis

The issue here is whether to construe the claim term according to the definition provided in the ’400 Patent or to construe it according to Plaintiff’s rewrite of that definition. The Court determines that the explicit definition provided in the patent is the better construction.

The ’400 Patent provides that “virtual cabinet,” “cabinet record,” and “virtual cabinet record” are used synonymously as defined in the patent:

For instance, parents who use a computer for business purposes cannot, in general, protect their critical data while allowing children to play games on the same system. It would therefore be beneficial to the prior art to provide a GUI Interface for allocation of secondary storage device that can restrict access to a pre-specified section, independent of any operating system, in the form of Virtual Cabinets (Cabinet Records, hereinafter referred to as “Cabinets”). *A cabinet is defined*

herein as a virtual storage device, capable of containing, typically through the use of virtual table of content pointers, all (or partitions of) shared (or non-shared) operating systems, application software (both OS dependent and No-OS embedded), databases and memory.

'400 Patent col.2 ll.40–52 (emphasis added). This explicit definition is stated a second time in the patent:

For the purposes of this invention and disclosure, the terms “Virtual Cabinet”, “Cabinet Record” and “cabinet” are synonymous. ***A cabinet is defined herein*** as a virtual storage device, capable of containing, typically through the use of virtual table of content pointers, all (or partitions of) shared (or non-shared) operating systems, application software (both OS dependent and No-OS embedded), databases and memory.

Id. at col.5 ll.24–29. And then again in the prosecution history: “The disclosure of the current invention defines a cabinet as ‘a virtual storage device, capable of containing, ... all (or partitions of) shared (or non-shared) operating systems, application software (both OS dependent and No-OS embedded), databases and memory.’” ’400 Patent File Wrapper September 4, 2002 Response 11 (ellipsis in original), Dkt. No. 186-6 at 12.

Plaintiff’s concern that this definition is unhelpful because it uses the term “virtual” is unfounded. There is no evidence that “virtual” is used in the patent other than according to its common lay meaning. That is, a “virtual” storage device appears as a storage device but is not defined by physical properties. Indeed, the meaning of “virtual storage device” is more accessible than the meaning of “virtualized software and/or data.” What does it mean for software or data to be “virtualized”? In addition, Plaintiff’s proposed construction apparently removes the concept of “storage” and “containing.” As plainly stated in the explicit definition, a “cabinet” is a “virtual storage device, capable of containing” software and data.

Accordingly, the Court construes the terms as follows:

- “virtual cabinet,” “cabinet record,” and “virtual cabinet record” each means “virtual storage device, capable of containing, typically through the use of virtual

table of content pointers, all (or partitions of) shared (or non-shared) operating systems, application software (both OS dependent and No-OS embedded), databases and memory.”

A-4. “secondary storage,” and “secondary storage device”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“secondary storage” • ’400 Patent Claims 1, 16, 28	a storage medium separate from the computer device	storage other than random access memory
“secondary storage device” • ’400 Patent Claims 1, 16, 28	a device containing a storage medium separate from the computer device	hardware storage device other than random access memory

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits that “secondary storage” and a “secondary storage device” are each separate and independent from a computer device and attached to the computer device, as evinced by U.S. Patent No. 6,401,183 (the “’183 Patent”), which is a prior art patent that is denoted in the ’400 Patent as “related” to the subject matter of the ’400 Patent. Dkt. No. 169 at 14–16 & nn.35–37 (citing ’400 Patent col.1 ll.7–11; ’183 Patent col.8 ll.2–5, col.11 ll.48–52, col.12 ll.1–3). Plaintiff argues that Defendants’ proposed construction is incorrect because the ’400 Patent makes no reference to RAM or random access memory and because memory that is not RAM, such as ROM or EPROM, can be part of a computer device, and is not necessarily separate from and attached to a computer device. *Id.* at 15. Plaintiff further argues that neither the ’400 Patent nor the ’183 Patent refer to “secondary storage” as hardware—and such include both hardware and software. *Id.* (citing *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240, 1243 (3d

Cir. 1983)). With respect to “secondary storage,” as distinct from “secondary storage device,” Plaintiff submits that it appears only in the preamble of Claim 1. *Id.* at 16.

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent col.1 ll.7–11. **Extrinsic evidence:** U.S. Patent No. 6,401,183⁸; U.S. Patent No. 5,628,022; Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that the ordinary meaning of “secondary storage” in the art is that it is memory other than RAM. Dkt. No. 186 at 17–18 (citing *Microsoft Computer Dictionary* 424 (4th ed. 1999), Dkt. No. 206-1 at 6). Defendants further respond that the ’400 Patent explains that secondary storage devices may be found in the computer system and are therefore not necessarily “separate from the computer device.” *Id.* at 18. Finally, Defendants respond that Plaintiff’s proposed construction would exclude the common configuration of a hard disk contained within a computer device, and such configuration is included in the patent’s preferred embodiments. *Id.*

In addition to the claims themselves, Defendants cite the following intrinsic and extrinsic evidence to support their position: **Intrinsic evidence:** ’400 Patent col.7 ll.7–10. **Extrinsic evidence:** *Microsoft Computer Dictionary* (4th ed. 1999) (“storage device”) (Defendants Ex. H, Dkt. No. 206-1 at 6).

Plaintiff replies that: (1) “a computer system” is different than a “a computer device,” and (2) Defendants fail to cite any evidence in support of the purported ordinary understanding the term. Dkt. No. 189 at 10.

⁸ The ’183 Patent issued from Patent Application No. 90/283,318, which is identified in the ’400 Patent as being related to the subject matter of the ’400 Patent. ’400 Patent col.1 ll.7–11. The ’183 Patent, however, is not cited on the face of the ’400 Patent, is not incorporated by reference into the ’400 Patent, and does not share a claim of priority with the ’400 Patent. As such, the Court treats that ’183 Patent as extrinsic to the ’400 Patent.

Analysis

The issues in dispute appear to be: (1) whether “secondary storage” includes anything other than RAM, and (2) whether “secondary storage” is necessarily separate from the computer device. The Court determines that “secondary storage” is used in the ’400 Patent according to its customary meaning in the art: “secondary storage” is not “main memory.”

A “secondary storage device,” and “secondary storage” generally, is not necessarily “separate from the computer device.” To begin, the statements regarding “secondary storage” in the ’183 Patent are not definitional. But the statements evince that “secondary storage” has a customary meaning in the art and provide clues as to that meaning. The ’183 Patent suggests that “secondary storage” may be separate from a computer device in that the “secondary storage devices” are described as “attached to the computer device.” *See, e.g.*, ’183 Patent col.8 ll.1–3. The ’183 Patent also suggests that a secondary storage device may be part of a computer device. *See, e.g., id.* at col.10 l.64 – col.11 l.6 (referring to a “computer device having multiple secondary storage devices” and a “computer device including both secondary storage device A ... and secondary storage device B”). This comports with the use of “secondary storage” in the ’400 Patent. *See, e.g.*, ’400 Patent col.2 ll.1–6 (noting that certain operating systems “are designed to monopolize and re-configure the entire computer system, including secondary storage devices”), col.7 ll.7–8 (referring to “secondary storage devices found in the computer system”).

A “secondary storage device,” and “secondary storage” generally, is distinct from main memory. For example, the ’400 Patent notes that “[s]uper operating systems allow computer users to load multiple operating systems from secondary storage into main memory.” ’400 Patent col.1 ll.50–52; *see also*, col.2 ll.1–3 (noting that “operating systems ... are loaded from secondary storage into main memory”), col.3 ll.7–10 (noting that a partition on a “secondary storage device”

is “loaded into main memory”). The distinction between main memory and secondary storage is also present in the ’183 Patent. *See, e.g.*, ’183 Patent col.3 ll.10–12 (noting that a partition on a “secondary storage device” is “loaded into main memory”); col.4 ll.50–54 (noting that during startup, “the computer device loads the first sector of the first secondary storage device into main memory”). This comports with the definition of “storage device” in the *Microsoft Computer Dictionary*, which notes a “distinction . . . between primary (main) storage devices and secondary (auxiliary) storage devices.” *Microsoft Computer Dictionary* 424 (4th ed. 1999), Dkt. No. 206-1 at 6. According to this dictionary, the distinction is that the term “secondary (auxiliary) storage devices” “refers to disk drives and other external devices” and the term “primary (main) storage devices” refers to “random access memory.” *Id.* That is, main memory is both RAM and internal and secondary storage is external and not RAM. Injecting “external” into a claim construction, however, would likely create confusion since the patent allows that “secondary storage” may be part of a computer device or system and it is not clear how “external” is being used in the dictionary. ’400 Patent col.7 ll.7–8. It is clear, however, from both the intrinsic and extrinsic evidence of record that “secondary storage” is storage that is not main memory.

Accordingly, the Court construes “secondary storage” and “secondary storage device” as follows:

- “secondary storage” means “storage other than main memory”;
- “secondary storage device” means “storage device other than a main memory device.”

A-5. “secondary storage partitions window”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“secondary storage partitions window” • ’400 Patent Claims 1, 16, 28	a graphical representation of at least one segment of storage on a storage medium separate from the computer device	a window in the GUI depicting the contents of the all secondary storage devices found in the computer system

The Parties’ Positions

Plaintiff submits that Defendants’ proposed construction is improper as it simply reorders the words of the term in dispute, adds extraneous limitations and is therefore unhelpful to the jury. Dkt. No. 169 at 15–16.

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent col.2 l.64 – col.3 l.1, col.3 ll.6–10. **Extrinsic evidence:** U.S. Patent No. 6,401,183; Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that this term is defined in the ’400 Patent. Dkt. No. 186 at 18–19 (citing ’400 Patent col.7 ll.7–9).

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’400 Patent fig.1, col.7 ll.7–9.

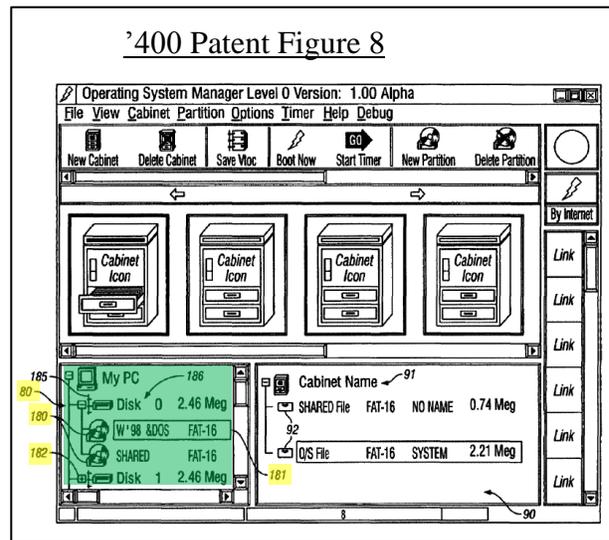
Analysis

The issue in dispute appears to be whether the “secondary storage partitions window” is defined in the ’400 Patent as Defendants’ propose. The Court determines that it is not so defined.

The term “secondary storage partitions window” is not defined as Defendants contend. Referencing Figure 8, reproduced here and annotated by the Court, the patent provides:

As stated above, Cabinet Button 21 depicts an active cabinet, whose contents, which may include operating systems, partitions, software and data, can be manipulated (added, removed, renamed, relocated in memory) by FVOS GUI 10. The contents of the active cabinet are depicted in FIG. 8 in Active Selected Cabinet Visible Partition Window 90. *The contents of the all secondary storage devices found in the computer system are depicted in Secondary Storage Partitions Window 80 as Secondary Storage Icon 180 and Secondary Storage Text Descriptor 186. The partitions of each secondary storage device are depicted with icons 180 and text descriptors 181, viewable through content button expanders 182.* The partitions can be virtually copied from the secondary storage to the active cabinet by an input device operation, such as “click and drag” movement using a pointer device such as a mouse, or by double clicking using such pointer device. The pointer device is placed over either icon 180 or text descriptor 181 in Secondary Storage Partitions Window 80, and the “click and drag” or double clicking operation of the pointer device places the partition described by icon 180 and text descriptor 181 into the active cabinet. As depicted in FIG. 9, the partition can be removed from the active cabinet by “right clicking” a pointer device over the selected partition, evoking cabinet button 190, and selecting Remover 93. In the preferred embodiment, the Configuration View shown in FIG. 10 allows for the same functions without being limited to the Selected Cabinet. Partitions are moved from the Secondary Storage Partitions Window 80 to the desired Cabinet depicted in Master Cabinet Visible Partition Window 91. Master Cabinet Visible Partition Window 91 depicts all cabinets, both selected (active) and non-selected (inactive), booted and non-booted.

'400 Patent col.7 ll.1–32 (emphasis added). This passage is not defining “secondary storage partitions window,” it is explaining how the window is used in the exemplary embodiment. The window (80, in green) provides access to the



partitions of the secondary storage devices so that the user may add partitions to cabinets. That is, the “secondary storage partitions window” lists the secondary storage devices and their partitions. It should be noted, however, that while the partitions are *viewable* in the window, whether the partitions of a particular secondary storage device are actually displayed is a function of the state of the expander button (182).

The claims further describe the “secondary storage partitions window.” For example, Claim 1 provides: “said secondary storage partitions window graphically illustrating at least one partition of at least one secondary storage device.” Indeed, Defendants’ proposed construction would not only render this limitation superfluous, it would effectively rewrite it as “said secondary storage partitions window graphically illustrating the contents of every secondary storage device.” The Court declines to narrow the plain meaning of the claim language as Defendants suggest.

Plaintiff’s proposed construction does not clarify scope. Setting aside Plaintiff’s incorporation of its proposed constructions for “secondary storage” and “partition,” Plaintiff rewrites “window” as “graphical representation.” The parties have not presented any evidence or argument regarding the meaning of “window” and the Court does not perceive that there is a dispute to be resolved here. In any event, the plain meaning of a “window” in a GUI is readily accessible. Further, while a “window” may be a “graphical representation,” not all “graphical representations” are windows. For example, the patent describes other graphical representations that are not windows, such as icons, buttons, and bars. The Court rejects Plaintiff’s expansive construction of “window.”

Accordingly, the Court construes “secondary storage partitions window” as follows:

- “secondary storage partitions window” means “window that depicts secondary storage devices and that is configurable to depict their partitions.”

A-6. “cabinet visible partition window”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“cabinet visible partition window” • ’400 Patent Claims 1, 16, 28	a graphical representation of a segment of storage on the virtual cabinet	a window in the GUI showing all of the contents of a selected cabinet

The Parties' Positions

Plaintiff submits that “cabinet visible partition window” is not necessarily limited to a “selected cabinet.” Dkt. No. 169 at 21. This, according to Plaintiff, is evinced by the descriptions of an “*Active Selected* Cabinet Partition Window” and a “*Master* Cabinet Partition Window,” which means that not all cabinet windows are “selected.” *Id.* (quoting ’400 Patent col.7 ll.1–10 (emphasis by Plaintiff)). Plaintiff further submits that while embodiments are described as showing “contents” they do this by, for example, showing information about the contents, such as file names and do not necessarily show the “all of the contents.” *Id.* (citing ’400 Patent figs. 8, 10).

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent figs.8,10, col.5 ll.53–60, col.6 ll.9–10, col.6 l.24, col.7 ll.1–10, col.7 ll.30–33, col.7 ll.43–48, col.8 ll.56–60. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that the claims require the “cabinet visible partition window” show the contents of the “selected” cabinet, namely the “operating system plus application software, databases and memory.” Dkt. No. 186 at 19–20 (quoting ’400 Patent Claims 1, 16, 28).

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’400 Patent fig.1, col.7 ll.1–32.

Analysis

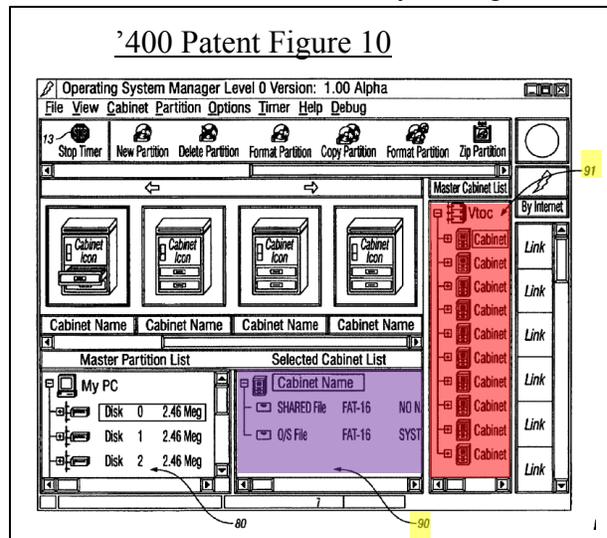
The issues in dispute appear to be (1) whether a “cabinet visible partition window” is limited to a “selected cabinet” and (2) whether the window must show “all of the contents” of that cabinet. A “cabinet visible partition window” is neither limited to a “selected cabinet” nor must it display “all of the contents” of a cabinet.

A “cabinet visible partition window” is not inherently limited to a “selected cabinet.”

Referencing Figure 10, reproduced here and annotated by the Court, the patent provides:

As stated above, Cabinet Button 21 depicts an active cabinet, whose contents, which may include operating systems, partitions, software and data, can be manipulated (added, removed, renamed, relocated in memory) by FVOS GUI 10. The contents of the active cabinet are depicted in FIG. 8 in *Active Selected Cabinet Visible Partition Window* 90. The contents of the all secondary storage devices found in the computer system are depicted in Secondary Storage Partitions Window 80 as Secondary Storage Icon 180 and Secondary Storage Text Descriptor 186. The partitions of each secondary storage device are depicted with icons 180 and text descriptors 181, viewable through content button expanders 182. The partitions can be virtually copied from the secondary storage to the active cabinet by an input device operation, such as “click and drag” movement using a pointer device such as a mouse, or by double clicking using such pointer device. The pointer device is placed over either icon 180 or text descriptor 181 in Secondary Storage Partitions Window 80, and the “click and drag” or double clicking operation of the pointer device places the partition described by icon 180 and text descriptor 181 into the active cabinet. As depicted in FIG. 9, the partition can be removed from the active cabinet by “right clicking” a pointer device over the selected partition, evoking cabinet button 190, and selecting Remover 93. In the preferred embodiment, the Configuration View shown in FIG. 10 *allows for the same functions without being limited to the Selected Cabinet*. Partitions are moved from the Secondary Storage Partitions Window 80 to the desired Cabinet depicted in *Master Cabinet Visible Partition Window* 91. *Master Cabinet Visible Partition Window 91 depicts all cabinets, both selected (active) and non-selected (inactive), booted and non-booted.*

'400 Patent col.7 ll.1–32 (emphasis added). The patent describes two “cabinet visible partition windows.” The first, the Active Selected Cabinet Visible Partition Window (90, in purple), is



limited to the selected cabinet. The other, the Master Cabinet Visible Partition Window (91, in red), “depicts all cabinets, both selected (active) and non-selected (inactive).” That is, the patent expressly states that a “cabinet visible partition window” is not inherently limited to a selected cabinet.

The above passage, along with Figure 10, further show that a “cabinet visible partition window” depicts cabinets with expander buttons in the same manner as the “secondary storage partitions window” depicts secondary storage devices with expander buttons. The expander button is shown in its expanded state in the Active Selected Cabinet Visible Partition Window and in its collapsed state in the Master Cabinet Visible Partition Window. So, the “cabinet visible partition window” is configurable to depict the contents of a cabinet but does not necessarily depict the contents.

Finally, the “cabinet visible partition window” depicts the contents of a cabinet by listing certain details of files and partitions (e.g., the names) in the cabinet, not by displaying the contents of those files and partitions. For example, the Active Selected Cabinet Visible Partition Window in Figure 10 depicts two files, the “SHARED File” and the “O/S File.” But the window does not depict the contents of those files. Similarly, the Active Selected Cabinet Visible Partition Window in Figure 9 depicts a partition, the “W'98 &DOS” partition. But the window does not depict the contents of that partition. '400 Patent col.7 ll.17–25.

That said, for the reasons explained above in the section on “secondary storage partitions window,” the Court rejects Plaintiff’s proposed construction.

Accordingly, the Court construes “cabinet visible partition window” as follows:

- “cabinet visible partition window” means “window that depicts one or more virtual cabinets and that is configurable to depict their files and partitions.”

A-7. “partition”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“partition” <ul style="list-style-type: none"> • ’400 Patent Claims 1, 16, 28 • ’677 Patent Claims 1, 3, 6⁹ 	a segment of storage	a stable and rigid portion of a secondary storage device that is logically distinct from other portions based on firmware-level address boundaries of the storage device

The Parties’ Positions

Plaintiff submits that “partition” is defined in the ’183 Patent and because that patent is “related” to the ’400 Patent, the construction of “partition” should comport with that definition. Dkt. No. 169 at 18–19 (’183 Patent col.1 ll.23–28, col.5 ll.20–22). Plaintiff argues that a “partition” is not necessarily logically distinct and that such is evinced by the patent's description of hard disks “partitioned into a number of logical devices.” *Id.* at 19 (quoting ’400 Patent col.1 ll.58–60). According to Plaintiff, if “partition” inherently included “logically distinct” this description would have stopped at “partitioned.” *Id.* Plaintiff further argues that a “partition” is not necessarily defined by “firmware-level address boundaries” as the patent describes that the address boundaries *can be* at the firmware level, meaning that they need not be at the firmware level. *Id.* at 19–20. Finally, Plaintiff argues that a “partition” need not be “stable and rigid” as the patent describes that partitions “need to be set up as stable and rigid partitions,” meaning that a partition is not inherently stable and rigid. *Id.* at 20 (quoting ’400 Patent col.1 ll.61–65).

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent col.1 ll.58–65; U.S. Patent No.

⁹ Plaintiff and Defendants each consolidated their arguments and evidence for “partition” as used in the ’677 Patent with that for the ’400 Patent.

6,401,183¹⁰. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that the “partitions” of the ’400 and ’677 Patents are consistently described as “stable and rigid” and “partitions” should be construed to give effect to how the term is used in the patents. Dkt. No. 186 at 20–21 (citing ’400 Patent col.1 ll.61–65, col.2 ll.1–11; ’677 Patent col.8 ll.4–14, col.8 ll.28–33, col.8 l.65 – col.9 l.6). Defendants further respond that the ’183 Patent states that a “partition” is a distinct segment of secondary storage, and that “distinct” in the art means “logically distinct.” *Id.* at 21 (citing ’183 Patent col.5 ll.20–22). Finally, Defendants respond that the ’400 Patent describes the “invention” as including the ability to define address boundaries at the firmware level, and thus the claimed partition is defined at the firmware level. *Id.*

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’400 Patent col.1 ll.61–65, col.2 ll.1–11, col.3 ll.33–35, col.3 ll.39–44; ’677 Patent col.8 ll.4–14, col.8 ll.28–33, col.8 l.65 – col.9 l.6; U.S. Patent No. 6,401,183.

Analysis

There are three main issues in dispute. First, whether a “partition” is necessarily “logically distinct” from other portions of secondary storage. Second, whether a “partition” is necessarily “based on firmware-level address boundaries” of the secondary storage. And third, whether a “partition” is necessarily “stable and rigid.” With respect to the first issue, a “partition” is logically distinct from other partitions, but not necessarily from all other portions of secondary storage. With

¹⁰ The ’183 issued from Patent Application No. 90/283,318, which is identified in the ’400 Patent as being related to the subject matter of the ’400 Patent. ’400 Patent col.1 ll.7–11. The ’183 Patent is incorporated into the disclosure of the ’677 Patent. ’677 Patent col.6 ll.58–65. Because “partition” is proposed for construction for both the ’400 Patent and the ’677 Patent, the Court treats the ’183 Patent as intrinsic evidence.

respect to the second issue, a partition is defined by address boundaries (e.g., starting location and size) but the Asserted Patents do not require that these be “firmware-level address boundaries.” And with respect to the third issue, while the ’400 Patent describes partitions as “stable and rigid,” the Court declines to include that phrase into the construction as doing so would obfuscate rather than clarify claim scope.

To begin, “partition” is used in the Asserted Patents according to its customary meaning in the art. For instance, the ’400 Patent refers to partitions when discussing the state of the art:

The cornerstone of a super operating systems GUI is system virtualization, in which physical devices, such as a hard disk and memory, are mapped repeatedly or ***partitioned into a number of logical devices***, each containing a separate operating system. ***These partitions***, however, need to be set up as stable and rigid partitions or mappings so that the operating systems do not mix, intermingle, call on each other, or exchange data, unless the user desires such exchange.

’400 Patent col.1 ll.57–65 (emphasis added). Thus, a “partition” is a logical device created from a physical device. The ’677 Patent describes the hard disk of Figure 3—designated as “PRIOR ART”—as having partitions. ’677 Patent fig.3, col.9 ll.7–16. And the ’183 Patent provides a pithy explanation of what “partition” means:

Prior art secondary storage devices, that are bootable including hard disks and solid-state disks, include a Master Boot Record (MBR) stored on the first sector of the device. The prior art MBR includes simple executable code and data file in a standard format.

Some secondary storage devices have a single file system that occupies the entire device. Illustrative of such systems are FAT-12 on floppy disks and FAT-16 on Zip™ drives. Other secondary storage devices can be divided into one or more partitions, with each partition having an independent file system organization. Generically, ***a partition can be defined as a continuous segment of storage. A partition is identified by such information as starting location, size, type, and active flag. Partitions are the basic storage units formatted by operating systems to meet the specific operating system storage allocation and retrieval services needs.***

Secondary storage devices for IBM PC and compatible systems as originally designed for DOS contain a structure that can consist of up to four partitions. This four-partition structure was subsequently enhanced so that the last partition can

contain more than one logical volume. This enhanced last partition is called an extended partition.

'183 Patent col.1 ll.13–34. Thus, a “partition” is a continuous segment of secondary storage. The

'183 Patent further explains how “partition” is elsewhere used in that patent, using a definition that comports with the customary meaning of “partition” earlier explained in patent:

The remainder of the application will refer, at various places, to “Partitions” and “Cabinets”. *A “Partition” is understood to be a distinct segment of physical secondary storage.* A “Cabinet” is a list of Partitions chosen from the set of all of the Partitions defined in secondary storage. A Partition may be thought of as a “file drawer” for a Cabinet for which it is listed.

Id. at col.5 ll.19–25. Thus, a partition is a “distinct segment of physical secondary storage.” That is, “partition” in the Asserted Patents refers to a prior art concept. It refers to a logical device comprising a distinct continuous segment of physical secondary storage.

A partition is not necessarily “distinct from other portions” of secondary storage, but it is distinct from other partitions. For example, secondary storage such as a hard disk may be subdivided into multiple partitions. Each partition is still part of the hard drive, so in that sense the partition is not logically distinct from the hard drive. Partitions do not, however, overlap. The Asserted Patents each explain the importance of partitions being distinct from other partitions. For example, the '400 Patent explains:

Similarly, when certain operating systems, such as the Windows family of operating systems, are loaded from secondary storage to main memory, they are designed to monopolize and re-configure the entire computer system, including secondary storage devices, to suit their particular requirements and parameters. If more than one operating system is to co-exist in the same physical device (as is the case in super operating systems), then such operating systems must be restricted from accessing any portion of the secondary storage address blocks that contain the other operating systems.

'400 Patent col.2 ll.1–11. Additionally, the '677 Patent explains that “care should be taken to ensure a partition with one type of data is not assigned to another, non-compatible Virtual PC system.” '677 Patent col.8 ll.12–14.

The Asserted Patents do not require a partition to be defined at the firmware level.

Defendants rely on the following passage for their contention:

This invention is a Graphic User Interface that enables a user to virtualize a computer system and to define secondary storage physical devices, in single or multiple/super operating system environments. The system, at its firmware level, enables a user to define and alter the address boundaries of the physical devices and memory. The address boundaries of the physical devices are defined by the address boundaries of one of the logical devices, which has been partitioned or remapped within that physical device. Because such boundary definitions can be implemented at the firmware level of the physical device, every address access request, regardless of its origin and regardless of the origin of the request (i.e., the software component issuing the request), can be subject to the firmware-defined boundaries.

'400 Patent at col.3 ll.30–44. This passage does not state that partition boundaries are defined in firmware. Rather, it states that the definitions of boundaries of the physical device of the virtualized computer system “can be implemented at the firmware level” and that the user’s ability to define and alter the boundaries is enabled by the system of the invention at the firmware level. It is not that partitions are necessarily defined at the firmware level, it is that the invention enables such definition (or alteration). The '183 Patent explains that partitions are “identified by such information as starting location, size, type, and active flag,” that “a partition table identifying up to four partitions” is in the Master Boot Record (MBR), which is “stored on the first sector of the [secondary storage] device.” '183 Patent col.1 ll.13–17, col.1 ll.35–40. Moreover, the '677 Patent also notes that partitions are defined by the partition table in the MBR:

The Partition table is maintained in the format depicted in FIG. 2 to satisfy the runtime OS requirements. It is important to note that in a multiple OS environment, the MBR is a dynamic data structure and may be updated depending on the active OS environment. Microsoft Windows operating systems rely on the MBR data for locating the file systems and for mounting logical disks. The Super OS builds the MBR partition table, making only those partitions and logical drives visible that the user has placed into the current Cabinet. If the user boots a Cabinet, or resumes a previously suspended or hibernated Cabinet to a current state, the Super OS updates the MBR so that the runtime Operating System has access only to those partitions and file systems that the user has included in that virtual environment. All other partitions, file systems, and unused disk space are hidden by the Super OS to protect

the files, folders, and other information from the currently running OS, its applications, users, and potential viruses.

'400 Patent col.8 l.56 – col.9 l.6. Further, the '677 Patent provides that “restrictive partitioning ... can be implemented in hardware (the firmware of the disk drives), or in software as part of the Super OS.” 677 Patent col.8 ll.28–33.

While the '400 Patent mentions that partitions are “rigid and stable,” the Court understands this to mean that partitions remain distinct. Injecting “rigid and stable” into the construction, although not *per se* incorrect, would also improperly inject ambiguity into the claims. For instance, “rigid and stable” may connote immutability to the jury. But the '400 Patent specifically provides that “[p]artitions can be edited” including by “deleting, formatting, ... resizing and zipping.” '400 Patent col.8 ll.7–12.

Accordingly, the Court construes “partition table” as follows:

- “partition” means “logical device corresponding to a distinct continuous segment of physical secondary storage.”

A-8. “means for configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window,” and “program code for configuring said at least one partition of said at least one secondary storage device through a secondary storage partitions window”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“means for configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 16 	<p>a pointing device such as a mouse, keyboard, user selection, program code or the like and equivalents thereof for using menus, menu bars, pull-down menus, link buttons, hot keys, function keys, command lines, program code, and the like for configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window) and equivalents thereof</p>	<p>35 U.S.C. § 112, ¶ 6.</p> <p>Function:</p> <ul style="list-style-type: none"> • configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claim 16 is invalid under 35 U.S.C. § 112
<p>“program code for configuring said at least one partition of said at least one secondary storage device through a secondary storage partitions window”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 28 	<p>plain and ordinary meaning / no construction necessary</p> <p>Alternative:</p> <ul style="list-style-type: none"> • a pointing device such as a mouse, keyboard, program code or the like and equivalents thereof 	<p>35 U.S.C. § 112, ¶ 6.</p> <p>Function:</p> <ul style="list-style-type: none"> • configuring said at least one partition of said at least one secondary storage device through a secondary storage partitions window <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claim 28 is invalid under 35 U.S.C. § 112

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties' Positions

Plaintiff submits that its “proposed structures are clearly linked or associated to the configuring function by the ’400 Patent or its prosecution history.” Dkt. No. 169 at 25–26. With respect to the “program code for ...” term, Plaintiff submits that “program code” is structure and therefore 35 U.S.C. § 112, ¶ 6 does not apply. *Id.* at 23–24 (citing *Smartflash LLC v. Apple Inc.*, 77 F. Supp. 3d 535, 561–63, Appendix C (E.D. Tex. 2014), adopted, 2015 U.S. Dist. LEXIS 17754 (E.D. Tex. Feb. 13, 2015)).

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent figs.8, 10, col.5 ll.9–33, col.5 ll.40–46, col.7 ll.6–30, col.8 ll.30–34. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that the “configuring” function of these terms does not connote structure. Dkt. No. 186 at 11 (citing Goodin Decl.¹¹ ¶¶ 58–59, Dkt. No. 186-7 at 18–19). And, Defendants further respond, like “means for,” “program code for” is a generic term that does not describe a specific structure and therefore both the “means for ...” and the “program code for ...” terms are governed by 35 U.S.C. § 112, ¶ 6. *Id.* at 11 (citing *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1348–49 (Fed. Cir. 2015)). Defendants argue that under 35 U.S.C. § 112, ¶ 6, the patent must provide a structure, such as an algorithm, for performing the recited function. *Id.* at 7–8.

In support of their contention that “program code” is not structural, Defendants state as follows: neither Plaintiff’s expert nor Defendants’ expert has here opined that “program code” is structural, unlike in *Smartflash*, where Defendants’ expert opined that “code” was structural. *Id.* at 11–12 (citing Goodin Decl. ¶ 54, Dkt. No. 186-7 at 16–17; Rosenberg Decl. ¶ 77, Dkt. No.

¹¹ Declaration of Richard M. Goodin, P.E.

186-8 at 27, 29; Rosenberg Dep. at 100:7 – 101:8, Dkt. No. 186-4 at 21–22; *Smartflash LLC v. Apple Inc.*, No. 6:13-CV-447-JRG-KNM, 2015 U.S. Dist. LEXIS 91669, at *9–*10 (E.D. Tex. July 6, 2015)). Neither the claims nor the description of the ’400 Patent describe how the “program code for configuring” operates “within the claimed invention to achieve its objective.” *Id.* at 12. (quoting *Uniloc USA, Inc. v. Autodesk, Inc.*, No. 2:15-cv-1187-JRG-RSP, 2016 U.S. Dist. LEXIS 87921, at *57 (E.D. Tex. July 7, 2016)). Thus, “program code for configuring” is a recitation of generic code without sufficiently definite structure to avoid application of § 112, ¶ 6. *Id.* at 12–13 (citing *Advanced Ground Info. Sys. v. Life360, Inc.*, 830 F.3d 1341, 1348 (Fed. Cir. 2016)).

Defendants further respond that the patent does not disclose structure sufficient to perform the function of “configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window.” *Id.* at 13. Specifically, Defendants contend that a “pointing device,” Plaintiff’s proposed structure, cannot perform this function. *Id.*

In addition to the claims themselves, Defendants cite the following **extrinsic evidence** to support their position: Goodin Decl. ¶¶ 41–44, 54, 58–59 (Defendants’ Ex. F, Dkt. No. 186-7 at 11–13, 16–19); Rosenberg Decl. ¶¶ 36, 77 (Defendants’ Ex. G, Dkt. No. 186-8 at 12–13, 29), Rosenberg Dep. at 81:9 – 87:1, 100:7 – 101:8 (Defendants’ Ex. C, Dkt. No. 186-4 at 9–15, 20–21).

Plaintiff replies that Defendants have not established that an algorithm is required because they have not established that the claims are directed to a general-purpose computer and that a general-purpose computer cannot perform the recited function. Dkt. No. 189 at 5. And Plaintiff replies to provide cites for what it purports are its expert’s identification of structure corresponding to the function of these terms. *Id.* at 2 & n.7. With respect to the “program code for ...” term, Plaintiff argues that the “objective or operation of the code” as stated in the claims provides

sufficient structure to avoid application of § 112, ¶ 6. *Id.* at 6–7. And Plaintiff contends the structural nature of “code” is evinced by numerous dictionaries and claim construction opinions, and cites a number online dictionaries, cases citing dictionaries, and court holdings. *Id.* at 6–7 & nn.34–38.

Plaintiff cites further **extrinsic evidence** to support its position: Rosenberg Dep. 74:3 – 82:20, 85:24 – 86:8, 114:6 – 119:18 (Plaintiff’s Ex. A, Dkt. No. 189-2 at 20–23, 30–31); Rosenberg Decl. ¶¶ 36–38, 59–62 (Plaintiff’s Ex. B, Dkt. No. 189-3 at 11–13, 21–22).

Analysis

There are two issues in dispute. First, whether the “program code for configuring ...” term is governed by 35 U.S.C. § 112, ¶ 6. Second, whether the patent complies with that statute by disclosing adequate structure for performing the recited functions for the “means for configuring ...” and “program code for configuring ...” terms. With respect to the first issue, the Court determines that the “program code for configuring ...” term is governed by § 112, ¶ 6. With respect to the second issue, what Plaintiff identifies as the disclosed structure for performing the recited function does not meet the disclosure requirement of § 112, ¶ 6.

To begin, the parties appear to agree that the “means for configuring ...” term is governed by § 112, ¶ 6. It is.

With respect to the “program code for configuring ...” term, the Court starts from the presumption that § 112, ¶ 6 does not apply because the term does not include the “means” language traditionally used to signal application of the statute. *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1347-49 & n.3 (Fed. Cir. 2015) (en banc in relevant portion). This “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.” *Id.* at 1349 (quotation marks omitted). The Court notes that in many instances, “code,” like “circuit” or “processor,” may connote sufficiently definite structure and is not a “nonce” or “functional” word that is subject to the limitations of § 112, ¶ 6. Whether recitation of a “code for” performing a function is governed by § 112, ¶ 6, as for a “circuit for” performing a function and a “processor for” performing a function, depends on whether the stated objectives and operation of the code connote sufficiently definite structure. *See, e.g., Linear Tech. Corp. v. Impala Linear Corp.*, 379 F.3d 1311, 1319–21 (Fed. Cir. 2004) (“circuit [for performing a function]” found to be sufficiently definite structure because the claim recited the “objectives and operations” of the circuit); *Apple Inc. v. Motorola, Inc.*, 757 F.3d 1286, 1295, 1298–99, 1301 (Fed. Cir. 2014) (noting that structure for software “is understood through, for example, an outline of an algorithm, a flowchart, or a specific set of instructions or rules” and finding “heuristic [for performing a function]” to be sufficiently definite structure because the patent described the operation and objectives of the heuristic). In the context of the “program code for configuring ...” term, however, the program code is defined only by the function that it performs. How the code interacts with other code or structure of the claimed invention is not described. There are no indicia of the structural nature of this code in the claim or in the disclosure of the ’400 Patent. As such, the Defendants have rebutted the presumption that § 112, ¶ 6 does not apply to the “program code for ...” terms.

The ’400 Patent does not disclose sufficient structure for “configuring said at least one partition of said at least one secondary storage device through a secondary storage partitions window.” Indeed, the patent discloses no more than a graphic interface to such partition-configuring functions. *See* ’400 Patent col.8 ll.7–30. The disclosure of this interface, while it might suffice if the recited function was to provide parameters to a partition-configuring function, does

not disclose how the partitions are configured. As explained in the '677 Patent, the function of the GUI is to provide an interface to other functions, such as configuring partitions. '677 Patent col.7 ll.4–9 (the “U.S. patent application Ser. No. 09/409,013 [issued as the '400 Patent] ... describes a graphical user interface that facilitates the use of a Super Operating System”), col.7 ll.55–56 (“The Super OS GUI provides a graphical user interface (“GUI”) to the Super OS functions.”); *see also*, '400 Patent col.4 ll.1–10 (listing “System and OS Functions” including “Partition Tools and Functions”). Plaintiff’s expert’s opinion on this is not credible. He mistakenly views disclosure regarding an interface to a cabinet-configuring function (i.e., assigning existing partitions to cabinets) with the recited partition-configuring function. Rosenberg Decl. ¶¶ 33–38, Dkt. No. 179 at 10–13. Ultimately, the patent does not adequately disclose any structure, algorithm or otherwise, by which a partition can be configured.

Accordingly, the Court determines that the “program code for configuring ...” term and the “means for configuring ...” term are each governed by § 112, ¶ 6 and holds that the '400 Patent does not disclose adequate structure for the “configuring said at least one partition of said at least one secondary storage device through said secondary storage partitions window” function recited in these terms.

A-9. “means for manipulating said at least one cabinet record through said cabinet visible partition window,” and “program code for manipulating said virtual cabinet record through said cabinet visible partition window”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“means for manipulating said at least one cabinet record through said cabinet visible partition window”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 16 	<p>a pointing device such as a mouse, keyboard, user selection, program code or the like and equivalents thereof for using menus, menu bars, pull-down menus, link buttons, hot keys, function keys, command lines, program code, and the like for manipulating said at least one cabinet record through said cabinet visible partition window and equivalents thereof</p>	<p>35 U.S.C. § 112, ¶ 6</p> <p>Function:</p> <ul style="list-style-type: none"> • manipulating said at least one cabinet record through said cabinet visible partition window <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claim 16 is invalid under 35 U.S.C. § 112
<p>“program code for manipulating said virtual cabinet record through said cabinet visible partition window”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 28 	<p>plain and ordinary meaning / no construction necessary</p> <p>Alternative:</p> <ul style="list-style-type: none"> • menus, menu bars, pulldown menus, link buttons, hot keys, function keys, command lines, program code, and the like for manipulating said at least one cabinet record through said cabinet visible partition window and equivalents thereof 	<p>35 U.S.C. § 112, ¶ 6</p> <p>Function:</p> <ul style="list-style-type: none"> • manipulating said virtual cabinet record through said cabinet visible partition window <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claim 28 is invalid under 35 U.S.C. § 112

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits that its “proposed structures are clearly linked or associated to the configuring function by the ’400 Patent or its prosecution history.” Dkt. No. 169 at 24. With respect

to the “program code for ...” term, Plaintiff submits that “program code” is structure and therefore 35 U.S.C. § 112, ¶ 6 does not apply. *Id.* at 23–24.

In addition to the claims themselves, Plaintiff cites the following **intrinsic evidence** to support its position: ’400 Patent, at [57] Abstract, figs.1, 3, 4, 6, 8, 9, 15, col.3 l.45 – col.4 l.8, col.4 ll.39–69, col.5 ll.19–22, col.5 ll.29–35, col.6 ll.6–12, col.6 ll.22–32, col.6 ll.51–55, col.7 l.1 – col.8 l.60.

Defendants respond that both the “means for ...” term and the “program code for ...” term are governed by 35 U.S.C. § 112, ¶ 6, reiterating its argument set forth in the section on “program code for configuring” Dkt. No. 186 at 13–14. Defendants contend that while Plaintiff incorporates the recited function into its proposed constructions, Plaintiff’s expert opines that the function is the same as for the “means for configuring ...” term, namely, “configur[ing], or chang[ing], memory partitions or data.” *Id.* at 14 (quoting Rosenberg Decl. ¶ 42, Dkt. No. 186-8 at 15–16 (Defendants’ modifications)). And although Plaintiff’s expert identifies the same function as the “configuring ...,” Plaintiff identifies different structures for performing that function. *Id.* Yet, according to Defendants, the structures identified by Plaintiff are standard GUI elements and “cannot themselves change memory partitions or data.” *Id.*

In addition to the claims themselves, Defendants cite the following intrinsic and extrinsic evidence to support their position: **Intrinsic evidence:** ’400 Patent col.5 ll.30–35. **Extrinsic evidence:** Goodin Decl. ¶¶ 54, 58, 62, 66, 69–71, 74 (Defendants’ Ex. F, Dkt. No. 186-7 at 16–24); Rosenberg Decl. ¶ 42 (Defendants’ Ex. G, Dkt. No. 186-8 at 15–16), Rosenberg Dep. at 43:18 – 44:24, 91:12 – 92:13, 94:4–14 (Defendants’ Ex. C, Dkt. No. 186-4 at 7–8, 16–18).

Plaintiff replies as it did with respect to the “means for configuring ...” / “program code for configuring ...” terms, as set forth above.

Plaintiff cites further **extrinsic evidence** to support its position: Rosenberg Dep. 91:12 – 95:5, 119:19 – 121:1 (Plaintiff’s Ex. A, Dkt. No. 189-2 at 24–25, 31); Rosenberg Decl.¹² ¶¶ 42–44, 78–80¹³ (Plaintiff’s Ex. B, Dkt. No. 189-3 at 14–15, Dkt. No. 189-4 at 1–4).

Analysis

The issues in dispute here largely parallel the issues in dispute for the “means for configuring ...” and “program code for configuring ...” terms discussed above. For the same reasons set forth previously, the “means for manipulating ...” and the “program code for manipulating ...” terms are both governed by 35 U.S.C. § 112, ¶ 6. As with the “configuring” function, the recited “manipulating” functions are not supported by an adequate disclosure of structure in the ’400 Patent.

The ’400 Patent does not disclose adequate structure for performing the recited functions. The recited functions essentially distill to manipulating a virtual cabinet. As with the partition-configuring function, the described GUI provides an interface to the cabinet-manipulating function of the Super OS rather than performing the cabinet-manipulating function. ’677 Patent col.7 ll.4–9 (the “U.S. patent application Ser. No. 09/409,013 [issued as the ’400 Patent] ... describes a graphical user interface that facilitates the use of a Super Operating System”), col.7 ll.55–56 (“The Super OS GUI provides a graphical user interface (“GUI”) to the Super OS functions.”); ’400 Patent col.4 ll.1–10 (listing “System and OS Functions” including “Cabinet Tools and Functions”). Plaintiff’s expert is not credible on this issue. He conflates the cabinet-manipulating function with the partition-configuring function. *See, e.g.*, Rosenberg Decl. ¶¶ 42 (opining the “means for

¹² Plaintiff’s Ex. B was submitted as four ECF files, Dkt. No. 189-3, Dkt. No. 189-4, Dkt. No. 189-5, and Dkt No. 189-6.

¹³ The Declaration of Craig Rosenberg, Ph.D., includes two of each paragraph for paragraphs 76 through 81. Thus, it is unclear which paragraphs Plaintiff is referencing with its pin cite. Dkt. No. 189-2 at 1–4.

manipulating is to configure, or change, memory partitions or data”), Dkt. No. 179 at 14. Ultimately, the ’400 Patent does not disclose adequate structure for manipulating cabinets.

Accordingly, the Court determines that the “program code for manipulating ...” term and the “means for manipulating ...” term are each governed by § 112, ¶ 6 and holds that the ’400 Patent does not disclose adequate structure for the “manipulating said virtual cabinet record through said cabinet visible partition window” and the “manipulating said at least one cabinet record through said cabinet visible partition window” functions respectively recited in these terms.

A-10. “means for modifying said at least one cabinet record through said cabinet visible partition window,” and “program code for means for modifying said at least one cabinet record through said cabinet visible partition window”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“means for modifying said at least one cabinet record through said cabinet visible partition window”</p> <ul style="list-style-type: none"> • ’400 Patent Claim 16 	<p>a pointing device such as a mouse, keyboard, user selection, program code or the like and equivalents thereof for using menus, menu bars, pull-down menus, link buttons, hot keys, function keys, command lines, program code, and the like for modifying said at least one cabinet record through said cabinet visible partition window and equivalents thereof</p>	<p>35 U.S.C. § 112, ¶ 6</p> <p>Function:</p> <ul style="list-style-type: none"> • modifying said at least one cabinet record through said cabinet visible partition window <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claim 16 is invalid under 35 U.S.C. § 112

Disputed Term	Plaintiff's Proposed Construction	Defendants' Proposed Construction
<p>“program code for means for modifying said at least one cabinet record through said cabinet visible partition window”</p> <ul style="list-style-type: none"> ’400 Patent Claim 28 	<p>plain and ordinary meaning / no construction necessary</p> <p>Alternative:</p> <ul style="list-style-type: none"> menus, menu bars, pulldown menus, link buttons, hot keys, function keys, command lines, program code, and the like for manipulating said at least one cabinet record through said cabinet visible partition window and equivalents thereof 	<p>35 U.S.C. § 112, ¶ 6</p> <p>Function:</p> <ul style="list-style-type: none"> modifying said at least one cabinet record through said cabinet visible partition window <p>Structure:</p> <ul style="list-style-type: none"> none disclosed; this claim term is indefinite and claim 28 is invalid under 35 U.S.C. § 112

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits that its “proposed structures are clearly linked or associated to the configuring function by the ’400 Patent or its prosecution history.” Dkt. No. 169 at 25. With respect to the “program code for ...” term, Plaintiff submits that “program code” is structure and therefore 35 U.S.C. § 112, ¶ 6 does not apply. *Id.* at 23–24.

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’400 Patent col.3 ll.10–12, col.5 l.60 – col.6 l.12. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that both the “means for ...” term and the “program code for ...” term are governed by 35 U.S.C. § 112, ¶ 6, reiterating its argument set forth in the section on “program code for configuring” Dkt. No. 186 at 15. Defendants contend that Plaintiff’s expert opines that the function is the same as for the “means for configuring ...” term, namely, “configur[ing], or chang[ing], memory partitions or data.” *Id.* at 15–16 (quoting Rosenberg Decl. ¶ 48, Dkt. No.

186-8 at 17–18 (Defendants’ modifications)). Again, according to Defendants, the structures identified by Plaintiff, namely “pointing devices,” cannot change memory partitions. *Id.*

In addition to the claims themselves, Defendants cite the following intrinsic and extrinsic evidence to support their position: **Intrinsic evidence:** ’400 Patent col.2 ll.47–52. **Extrinsic evidence:** Goodin Decl. ¶¶ 51–52, 74–75 (Defendants’ Ex. F, Dkt. No. 186-7 at 15–16, 24–25); Rosenberg Decl. ¶¶ 48–49 (Defendants’ Ex. G, Dkt. No. 186-8 at 17–18), Rosenberg Dep. at 96:17 – 97:4 (Defendants’ Ex. C, Dkt. No. 186-4 at 19–20).

Plaintiff replies as it did with respect to the “means for configuring ...” / “program code for configuring ...” terms, as set forth above.

Plaintiff cites further **extrinsic evidence** to support its position: Rosenberg Dep. 95:11 – 99:13, 119:21 – 121:1 (Plaintiff’s Ex. A, Dkt. No. 189-2 at 25–26, 31); Rosenberg Decl.¹⁴ ¶¶ 48–50, 78–81¹⁵ (Plaintiff’s Ex. B, Dkt. No. 189-3 at 16–17, Dkt. No. 189-4 at 1–4).

Analysis

The issues in dispute here largely parallel the issues in dispute for the “means for configuring ...” and “program code for configuring ...” terms discussed above. For the same reasons set forth above, the “means for modifying ...” and “program code for modifying ...” terms are both governed by 35 U.S.C. § 112, ¶ 6. As with the “configuring” function, the recited “modifying” function is not supported by an adequate disclosure of structure in the ’400 Patent.

The ’400 Patent does not disclose adequate structure for performing the recited function. The recited function is modifying a virtual cabinet. As with the partition-configuring function, the

¹⁴ Plaintiff’s Ex. B was submitted as four ECF files, Dkt. No. 189-3, Dkt. No. 189-4, Dkt. No. 189-5, and Dkt No. 189-6.

¹⁵ The Declaration of Craig Rosenberg, Ph.D., includes two of each paragraph for paragraphs 76 through 81. Thus, it is unclear which paragraphs Plaintiff is referencing with its pin cite. Dkt. No. 189-2 at 1–4.

described GUI provides an interface to the cabinet-modifying function of the Super OS rather than performing the cabinet-modifying function. ’677 Patent col.7 ll.4–9 (the “U.S. patent application Ser. No. 09/409,013 [issued as the ’400 Patent] ... describes a graphical user interface that facilitates the use of a Super Operating System”), col.7 ll.55–56 (“The Super OS GUI provides a graphical user interface (“GUI”) to the Super OS functions.”);’400 Patent col.4 ll.1–10 (listing “System and OS Functions” including “Cabinet Tools and Functions”). Plaintiff’s expert is not credible on this issue. He conflates the cabinet-modifying function with the partition-configuring function. *See, e.g.*, Rosenberg Decl. ¶¶ 48 (opining the “means for modifying is to configure, or change, memory partitions or data”), Dkt. No. 179 at 16. Ultimately, the ’400 Patent does not disclose adequate structure for modifying cabinets.

Accordingly, the Court determines that the “program code for modifying ...” term and the “means for modifying ...” term are each governed by § 112, ¶ 6 and holds that the ’400 Patent does not disclose adequate structure for the “modifying said at least one cabinet record through said cabinet visible partition window” function recited in these terms.

B. U.S. Patent No. 7,356,677

B-1. Claim 1 preamble and “said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“A hardware platform for a hibernate capable computer system comprising a system manager, said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications, said storage manager having a virtual table of contents for organizing and accessing a plurality of partitions of relevant data and having a plurality of virtual computer systems, each of said virtual computer systems capable of accessing a selection of the partitions, the virtual table of contents (VTOC) being capable of dynamically configuring a plurality of partition tables, said system manager comprising”</p> <ul style="list-style-type: none"> • ’677 Patent Claim 1 	<p>plain and ordinary meaning / no construction necessary</p> <p>Alternative:</p> <ul style="list-style-type: none"> • A system manager 	<p>No construction necessary beyond those terms within this phrase being otherwise construed.</p>
<p>“said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications”</p> <ul style="list-style-type: none"> • ’677 Patent Claim 1 	<p>plain and ordinary meaning / no construction necessary</p>	<p>a firmware-level storage manager that enables dynamic manipulation of the master boot record and partition table such that each OS virtually runs in its native environment, without any changes, and is restricted and “believes” to be the entire computer storage</p>

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties' Positions

Plaintiff submits that the preamble of Claim 1 of the '677 Patent is not limiting and thus there is no need to construe "said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications." Dkt. No. 169 at 27. Plaintiff further submits that if construed as limiting, the term should be given its plain and ordinary meaning. *Id.*

Defendants respond that the phrase "an OS-independent storage" was added to the Claim 1 preamble during prosecution to overcome a prior art rejection and the preamble is therefore limiting. Dkt. No. 186 at 25–26. Defendants further respond that "said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications" is defined in the intrinsic record in that the patentee defined "OS Independent Storage Management." *Id.* at 26. Thus, Defendants conclude, the term should be construed to give effect to the patentee's definition. *Id.* at 26–27.

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: '677 Patent File Wrapper December 19, 2006 Amendment and Response (Defendants' Ex. I, Dkt. No. 186-10), November 20, 2006 Office Action (Defendants' Ex. J, Dkt. No. 186-11), September 8, 2006 Amendment and Response (Defendants' Ex. K, Dkt. No. 186-12).

Plaintiff replies that Defendants have failed to establish a specific and concrete infringement or validity dispute that requires construction of this term. Dkt. No. 189 at 2. Plaintiff further replies that the preamble is not limiting because it was not clearly used to distinguish prior art during prosecution. *Id.* at 8.

Analysis

The issues in dispute are whether the preamble of Claim 1 of the '677 Patent is limiting and whether the OS-independent-storage-manager term should be construed consistent with statements made in the prosecution history. The preamble is limiting and the OS-independent storage manager term should be construed as that concept was explained during prosecution.

The preamble is limiting because it was used during prosecution to distinguish a prior art reference to overcome a rejection. The patentee defined OS-independent storage management and OS-dependent storage management to distinguish the two management approaches. '677 Patent File Wrapper September 8, 2006 Amendment and Response at 6–7, Dkt. No. 186-12 at 7–8. Specifically, the patentee provided:

OS Independent Storage Management means that each OS virtually runs in its native environment, without any changes, and that it is restricted and “believes” to be the entire computer storage (its cabinet). In reality, it may not be, as all other storage allocated to other OSs can be protected from the active OS by the OS-Independent storage manager invented by this Applicant. In fact, this is the only solution that allows predatory, adversary and hostile OSs to actually co-exist and share information on the same computer system. In an OS-Independent storage system environment, VTOC and DTOC enable dynamic manipulation of the master boot record (MBR) and partition table for each OS as necessary. This invention can present totally different MBR and partition tables for different active OSs to prevent unauthorized manipulation of storage and partitions allocated to other OS environments by the running OS. Accordingly, the prior art limitation of four (4) primary partitions by the MBR is eliminated in the present invention.

Id. (emphasis in original). Based on this definition of OS-independent storage management, the patentee argued that the prior art “does not teach the virtual systems, Cabinets, VTOC or OS-independent storage management of Applicant. As a result, [the prior art] has no solution or teaching for switching complete virtual systems with OS-independent storage management.” *Id.* at 10, Dkt. No. 186-12 at 11. That is, the claimed invention is distinguishable over the prior art in part because it has an OS-independent storage manager. This limitation is recited in the preamble and is not subsumed by a limitation in the body of the claim. Thus, the preamble is limiting.

Accordingly, the Court determines the Claim 1 preamble is limiting, and construes “said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications” as follows:

- “said computer system having an OS-independent storage manager operating through a firmware level and a plurality of operating systems and applications” means “said computer system having a storage manager operating through a firmware level and that enables dynamic manipulation of the master boot record and partition table such that each OS virtually runs in its native environment, without any changes, and that it is restricted and “believes” to be the entire computer storage.”

B-2. “virtual table of contents”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“virtual table of contents” <ul style="list-style-type: none"> • ’677 Patent Claims 1, 3, 6 	a data structure maintained on secondary storage, but outside the primary file systems, containing all necessary information about the Virtual Personal Computer (‘Virtual PC’)	a data structure maintained on secondary storage, but outside the primary file systems, that contains all necessary information about each virtual computer system(s) of the physical computer including storage information, partitions and file systems

The Parties’ Positions

Plaintiff submits the “virtual table of contents” is defined in the ’677 Patent. Dkt. No. 169 at 28 (citing ’677 Patent col.7 ll.23–29). Also, Plaintiff contends that the definition refers to “necessary information” for a singular environment, not to “each virtual computer system.” *Id.*

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’677 Patent col.7 ll.23–29. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that in the portion of the '677 Patent that Plaintiff identifies as definitional, the patent provides the virtual table of contents “defines Cabinets (Virtual PC’s),” meaning it defines multiple Virtual PCs. Dkt. No. 186 at 27 (quoting '677 Patent col.7 ll.28–29 (Defendants’ emphasis)). According to Defendants, the requirement of multiple Virtual PCs is further evinced by the claim language itself (e.g., Claim 1’s “a plurality of virtual computer systems”) and by statements made during prosecution. *Id.* at 27–28.

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: '677 Patent fig.4, col.7 ll.28–29; '677 Patent File Wrapper February 14, 2006 (Defendants’ Ex. L, Dkt. No. 186-13).

Analysis

The issue in dispute appears to be whether a “virtual table of contents” includes information for each of the plurality of virtual computer systems of the claims. It does.

The '677 Patent provides a definition for “virtual table of contents”:

The following terms are used herein, and are described in more detail in the foregoing described related pending applications:

The Virtual Table of Contents (“VTOC”) is a data structure maintained on secondary storage, but outside the primary file systems. The VTOC ***contains all necessary information about the Virtual Personal Computer (“Virtual PC”) environment of a physical computer.*** This includes storage information, partitions and file systems. Furthermore, the VTOC defines Cabinets (Virtual PC’s). A Cabinet or Virtual PC is a set of partitions and file systems that are visible and available to a given operating system environment when that environment is actively executing on the physical computer. For PC’s with multiple hard disks, the VTOC is stored as a Disk Table of Contents (“DTC”) on each disk.

'677 Patent col.7 ll.20–34 (emphasis added). The parties disagree whether this means that the “virtual table of contents” contains all the necessary information for each Virtual PC. On this question, the patent provides further guidance: “In a Virtual PC, the information needed by each OS to be operable and to operate independently from all other OS’s and embedded applications,

and to control its Virtual PC environment, is stored in a VTOC data structure.” *Id.* at col.8 ll.15–18. That is, for every Virtual PC environment, the “necessary information” for the environment is stored in the virtual table of contents. And, as set forth below, a Virtual PC is the “virtual computer system” of the claims.

Accordingly, the Court construes “virtual table of contents” as follows:

- “virtual table of contents” means “data structure maintained on secondary storage, but outside the primary file systems, that contains all necessary information about each virtual computer system environment of a physical computer, including storage information, partitions and file systems.”

B-3. “partition”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“partition” • ’677 Patent Claims 1, 3, 6	a segment of storage	a stable and rigid portion of a secondary storage device that is logically distinct from other portions based on firmware-level address boundaries of the storage device

The Parties’ Positions

The parties presented their argument and evidence regarding the meaning of “partition” in the ’677 Patent with their argument and evidence regarding the meaning of “partition” in the ’400 Patent, as set forth above. The Court’s analysis and construction is presented there.

B-4. “virtual computer system”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“virtual computer system” <ul style="list-style-type: none"> • ’677 Patent Claims 1, 3, 6 	a set of partitions and file systems that are visible and available to a given operating system environment when that environment is actively executing on the physical computer	set of storage information, partitions and file systems that are visible and available to a given operating system environment, having a unique master boot record and its own partition table, when that environment is actively executing on the physical computer

The Parties’ Positions

Plaintiff submits that “virtual computer system” is synonymous with “Virtual PC” and “Virtual PC” is defined in the ’677 Patent. Dkt. No. 169 at 28 (citing ’677 Patent col.7 ll.29–32).

In addition to the claims themselves, Plaintiff cites the following **intrinsic evidence** to support its position: ’677 Patent, at [57] Abstract, col.3 ll.37–45, col.4 ll.13–16, col.10 ll.1–4; ’400 Patent, at [57] Abstract, col.2 ll.40–52, col.3 ll.1–5, col.5 ll.9–29.

Defendants respond that while the term is defined in the patent, that definition was further explained during prosecution in that the “virtual computer system” of the patent “owns its own unique master boot record (MBR) and partition table distinct from other virtual environments.” Dkt. No. 186 at 28 (quoting ’677 Patent File Wrapper September 8, 2006 Amendment and Response at 21, Dkt. No. 186-12 at 22). Thus, Defendants contend, “virtual computer system” should be construed to give effect to both the statement in the patent and the statement in the prosecution history.

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’677 Patent col.7 ll.29–32; ’677 Patent File Wrapper September 8, 2006 Amendment and Response (Defendants’ Ex. K, Dkt. No. 186-12).

Analysis

The issue in dispute appears to be whether a “virtual computer system” has its own unique master boot record (MBR) and partition table distinct from other environments. It does.

The '677 Patent provides a definition for “Virtual PC,” which the parties agree is synonymous with “virtual computer system”:

The following terms are used herein, and are described in more detail in the foregoing described related pending applications:

The Virtual Table of Contents (“VTOC”) is a data structure maintained on secondary storage, but outside the primary file systems. The VTOC contains all necessary information about the Virtual Personal Computer (“Virtual PC”) environment of a physical computer. This includes storage information, partitions and file systems. Furthermore, the VTOC defines Cabinets (Virtual PC’s). ***A Cabinet or Virtual PC is a set of partitions and file systems that are visible and available to a given operating system environment when that environment is actively executing on the physical computer.*** For PC’s with multiple hard disks, the VTOC is stored as a Disk Table of Contents (“DTC”) on each disk.

'677 Patent col.7 ll.20–34 (emphasis added). During prosecution, the patentee further explained that the “virtual computer system” has its “own unique master boot record (MBR) and partition table distinct from other virtual environmen[t]s”:

The Examiner next suggests that each OS of Shimotono comprises a virtual computer system. This assumption is incorrect, as each OS is simply the bootable image that may exist in a partition of a system. A virtual computer system contains one or more OSs, but each OS is not a virtual computer system. ***A virtual system of Applicant owns its own unique master boot record (MBR) and partition table distinct from other virtual environmen[t]s.***

'677 Patent File Wrapper September 8, 2006 Amendment and Response at 21(emphasis added), Dkt. No. 186-12 at 22. Defendants’ have not provided any argument or evidence to modify the patent’s definition from “set of partitions and file systems” to “set of storage information, partitions and files systems.” As such, the Court declines to modify the express definition in that respect.

Accordingly, the Court construes “virtual computer system” as follows:

- “virtual computer system” means “set of partitions and file systems that are visible and available to a given operating system environment when that environment is actively executing on the physical computer, having a unique master boot record and its own partition table.”

B-5. “BIOS ACPI solutions” and “BIOS ACPI enhancements”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“BIOS ACPI solutions” • ’677 Patent Claim 1	plain and ordinary meaning / no construction necessary	modifications made to standard BIOS ACPI power management software
“BIOS ACPI enhancements” • ’677 Patent Claim 3, 6		

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits that this term should not be construed for two reasons: First, it is part of longer term offered for construction (the “means for switching ...” term). Dkt. No. 169 at 32. Second, it is only found within Claim 28’s preamble, which is not limiting. *Id.* Plaintiff further submits the term should be given its “plain and ordinary” meaning without further construction as the meaning of the claim language is clear. *Id.* Finally, Plaintiff contends that Defendants’ proposed construction does not clarify claim scope but rather imports limitations from exemplary embodiments. *Id.*

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’677 Patent col.14 ll.26–32. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that “BIOS ACPI enhancements” is not part of a longer phrase offered for construction and that it is appropriate to construe “BIOS ACPI solutions” even though it appears in the “means for switching ...” term. Dkt. No. 186 at 32. Defendants further respond that the ’677 Patent does not have a Claim 28 and that either “BIOS ACPI enhancements” or “BIOS ACPI solutions” appears in the body of every independent claim of the ’677 Patent. *Id.* at 32–33. Furthermore, Defendants contend that the BIOS ACPI solutions/enhancements of the patent are described in the patent and prosecution history as improvements over a standard BIOS that enables the Fast Suspend and Fast Resume functions. *Id.*

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’677 Patent col.11 ll.27–62; ’677 Patent File Wrapper September 8, 2006 Amendment and Response (Defendants’ Ex. K, Dkt. No. 186-12).

Analysis

The issue in dispute appears to be whether the BIOS ACPI solutions/enhancements are modifications to other BIOS ACPI power management functions. They are.

To begin, the Court notes that “ACPI” is an acronym for Advanced Configuration and Power Interface. ’677 Patent col.5 ll.4–8 & Certificate of Correction. As explained in the ’400 Patent,

ACPI is a power management specification that enables the operating system to control the amount of power given to each device attached to the computer. With ACPI, the operating system can turn off peripheral devices, such as CD-ROM players, when they are not in use, or can automatically power up the computer as soon as an input device such as a mouse is moved.

’400 Patent at col.2 ll.53–64.¹⁶

¹⁶ The disclosure of the ’400 Patent is incorporated by reference into the disclosure of the ’677 Patent. ’677 Patent col.6 ll.58–59, col.7 ll.4–9.

The '677 Patent is squarely directed to using a modified version of the BIOS power-management functionality for the fast switching of the invention. *See, e.g.*, '677 Patent col.3 ll.17–20 (“It would be an improvement to the art to use the existing BIOS power management capabilities to facilitate fast switching between operating systems in conjunction with a Super OS.”), col.5 ll.4–8 & Certificate of Correction (“One of the key features of this invention is the novel use of the power management (advanced configuration and power interface (ACPI) or advanced power management (APM) or alike) support functions of the runtime (standard) OS and the BIOS to temporarily evict that OS from memory and switch to the next OS requested by the user.”); col.14 ll.26–32 (“The BIOS may be enhanced with a parameter that ... provides the user the option to resume automatically after a suspend, or to simply reboot the IPL and give control of the system to the Super OS after power up or after a resume from suspend.”). The Fast Resume and Fast Suspend method that comprise the Fast Switching method of the patent each make “modified use of the BIOS power management standard.” *Id.* at col.4 ll.33–46. And the patentee explained during prosecution that:

Applicant’s invention does not require nor incorporate any of the elements of the Puckette or Shimotono teachings and references. In fact, the ***Applicant’s fast suspend and fast resume totally rely on applying BIOS ACPI enhancements*** to eliminate the need for modifying any of the plurality of OSs designs (including their memory management) that comprise the multiple virtual system environment. Applicant’s invention requires no modification to the design of the native OSs.

'677 Patent File Wrapper September 8, 2006 Amendment and Response at 27 (bold emphasis added, underline in original), Dkt. No. 186-12 at 28. The Modification of the BIOS power-management functions are described as defining and using flags to identify suspended states and restore therefrom, and to create suspended states, bypassing the standard power-off. *See* '677 Patent col.10 l.56 – col.11 l.67. The patent describes enhancing various BIOS products, such as the Phoenix NoteBIOS. *Id.* at col.14 ll.7–32.

Accordingly, the Court construes “BIOS ACPI solutions” / “BIOS ACPI enhancements” as follows:

- “BIOS ACPI solutions” / “BIOS ACPI enhancements” means “modifications made to the BIOS ACPI power management functions.”

B-6. “power-on self test (POST) in the BIOS” and “power-on self test (POST) in BIOS”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“power-on self test (POST) in the BIOS” • ’677 Patent Claims 1	plain and ordinary meaning / no construction necessary	diagnostic test sequence performed by BIOS to determine if various system components are properly connected and operating and, if successful, pass control to the system’s bootstrap loader
“power-on self test (POST) in BIOS” ’677 Patent Claims 3, 6		

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits the terms should be given their “plain and ordinary” meaning without further construction as the meaning of the claim language is clear. Dkt. No. 169 at 33. Also, Plaintiff contends that Defendants’ proposed construction does not clarify claim scope but rather imports limitations from exemplary embodiments. *Id.*

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’677 Patent col.4 ll.35–43, col.13 ll.26–30. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that “power-on self test” and “self-test” had well-understood meanings in the art at the time the ’677 Patent was filed. Dkt. No. 186 at 33–34 (citing *Microsoft Computer*

Dictionary 352, 402 (4th ed. 1999), Dkt. No. 206-1 at 4–5). Defendants contend that the terms are used in the patent according to this meaning and should be construed accordingly. *Id.*

In addition to the claims themselves, Defendants cite the following intrinsic and extrinsic evidence to support their position: **Intrinsic evidence:** '677 Patent col.11 ll.6–11, col.13 ll.26–28. **Extrinsic evidence:** *Microsoft Computer Dictionary* (4th ed. 1999) (“power-on self test” and “self test”) (Defendants Ex. H, Dkt. No. 206-1 at 4–5).

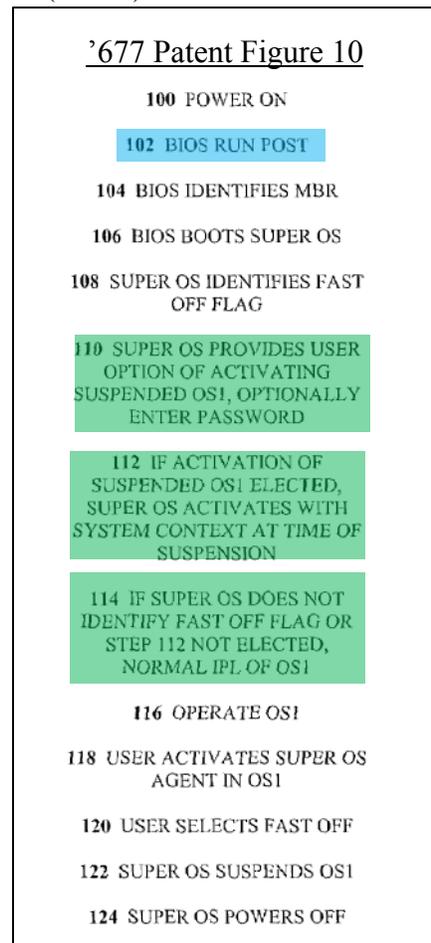
Plaintiff replies that Defendants “have not proven construction of the terms are necessary.” Dkt. No. 189 at 1 & n.2.

Analysis

The issue in dispute here appears to be whether Defendants’ proposed construction accurately captures the customary meaning of “power-on self test (POST)” as that term is used in the art. While the Court generally agrees with Defendants’ proposed construction, that construction goes too far in that it includes “pass control to the system’s bootstrap loader.”

The '677 Patent uses “power-on self test” (i.e., POST) according to its customary meaning, but it is not clear that the step after a successful POST is to “pass control to the system’s bootstrap loader.” The patent describes an exemplary boot sequence with reference to Figure 10, reproduced here and annotated by the Court:

Power on 100 of the computer system. Such a power on 100 includes *the typical BIOS boot sequence Power-On Self-Test (“POST”) check* 102, identification of a device containing a valid Master Boot Record (“MBR”) 104, and loading of the Super OS 106. Upon loading of the Super OS, control of the



computer is turned over to the Super OS Initial Program Loader (“IPL”). The Super OS IPL proceeds to check the system resources. *If Super OS does not identify a Fast Suspend Flag (described herein), Super OS proceeds to load OS1, the default OS 114, in the manner of a cold start*, including activation and verification of utilities and service programs, libraries, devices, and networks.

’677 Patent col.13 ll.26–37 (emphasis added). That is, after a successful POST (102, in blue), the BIOS proceeds to identify the MBR and load the super operating system. An operating system (other than the super operating system) is loaded only after checking to determine which operating system to load (110, 112, 114, all in green).

Defendants’ dictionary defines “power-on self test” as:

A set of routines stored in a computer's read-only memory (ROM) that tests various system components such as RAM, the disk drives, and the keyboard to see if they are properly connected and operating. If problems are found, these routines alert the user by sounding a series of beeps or displaying a message, often accompanied by a diagnostic numeric value, to the standard output or standard error device (usually the screen). If the power-on self test is successful, it passes control to the system’s bootstrap loader. *Acronym: POST. See also* bootstrap loader.

Microsoft Computer Dictionary 352 (4th ed. 1999) (italics in original), Dkt. No. 206-1 at 4. For the most part, this comports with the use of POST in the ’677 Patent. *See, e.g.*, ’677 Patent col.11 ll.5–11 (listing that “POST” includes “Check Flags,” “Check Memory,” “Check Devices,” “Configure Devices,” “Check Virus, Network, Options,” and “Build Tables, Memory Information.”). It is not clear that the “bootstrap loader” portion of this dictionary definition is applicable to the virtual PC (cabinet) environment of the ’677 Patent. Neither the ’677 Patent nor the parties explain the meaning of a “bootstrap loader.” The ’183 Patent, the disclosure of which is incorporated into the ’677 Patent, explains the boot sequence in a multi-OS environment:

To achieve such improvements, the invention comprises a Storage Manager that dynamically manipulates and partitions the secondary storage of a computer device without re-writing or revising the secondary storage after each manipulation. The *Storage Manager is implemented by executable code between the firmware level and the run-time operating system level of a computer device*. Means are described to transfer control of the computer device to the Storage Manager prior to the run-time operating system or any application programs gaining control of the computer

device. The Storage Manager includes a Virtual Table of Contents (VTOC), in which relevant identifying information is contained for each partition of the secondary storage. At least one Cabinet is created, containing a list of partitions. Each Cabinet may have a separate list of partitions, and each partition may be included in more than one Cabinet. One of the Cabinets is designated as the Active Cabinet. ***Prior to continuation of the run-time operating system boot sequence, the contents (i.e., the list of partitions within that Cabinet) replace the partition list of the secondary storage device. If the secondary storage device is bootable, then the partition within that Cabinet marked as bootable is bootstrapped and loaded into main memory.*** The contents of the partitions and Cabinets may be modified by a user through a graphical user interface.

'183 Patent col.2 l.57 – col.3 l.14 (emphasis added); *see also*, '400 Patent col.2 l.34 – col.3 l.10 (the Partition within [the active] Cabinet marked as bootable is bootstrapped and loaded into main memory”). The '183 Patent explains that the operating system is loaded at the bootstrapping step. '183 Patent col.9 l.62 – col.10 l.8. That is, bootstrapping occurs after the active virtual computer system (cabinet) is determined and loaded. If the “bootstrap loader” is that which loads the selected operating system, the bootstrap loader takes control only after the Super Operating System determines which operating-system environment to load. This does not comport with the use of POST in the '677 Patent. Given the ambiguity of “bootstrap loader” and what it means to “pass control” to the bootstrap loader, the Court declines to read such into the POST terms.

Accordingly, the Court construes “power-on self test (POST) in the BIOS” / “power-on self test (POST) in BIOS” as follows:

- “power-on self test (POST) in the BIOS” / “power-on self test (POST) in BIOS” means “test routines performed by BIOS to determine if various system components are properly connected and operating.”

B-7. “suspend for fast switching,” “fast suspending,” “fast switching,” and “power save suspend”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“suspend for fast switching” • ’677 Patent Claims 1, 3, 6	plain and ordinary meaning / no construction necessary	making modified use of the BIOS power management standard to provide fast shutdown of an operating system to its suspended state by swapping out or remapping the current address space of the active program or OS without device power-down
“fast suspending” • ’677 Patent Claims 3, 6	plain and ordinary meaning / no construction necessary	making modified use of the BIOS power management standard to provide fast shutdown of an operating system to its suspended state by swapping out or remapping the current address space of the active program or OS without device power-down
“fast switching” • ’677 Patent Claims 1, 3, 6	plain and ordinary meaning / no construction necessary	making modified use of the BIOS power management standard to fast suspend one operating system environment, followed by a fast resume of another operating system environment from a suspended state, without device power-down
“power save suspend” • ’677 Patent Claim 1, 3, 6	plain and ordinary meaning / no construction necessary	the customary and systematic termination of all services that forces the closure of all open files and cached information, shutdown of all devices and drivers and system power off

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits these terms should not be construed because Defendants have not shown that there is a need to construe the terms to resolve a “specific and concrete infringement or

invalidity dispute.” Dkt. No. 169 at 30–32. With respect to “suspend” / “suspending,” Plaintiff submits that the claims do not state how suspension occurs and it would be improper to read a limitation from the exemplary embodiments regarding the manner of suspension. *Id.* at 30. With respect to “fast switching,” Plaintiff contends that the term is used according to its ordinary meaning and should not be rewritten to import limitations from exemplary embodiments. *Id.* at 33. Plaintiff also submits that the meaning of “power save” is understandable by the jury. *Id.* at 31.

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** ’677 Patent col.3 ll.17–20, col.4 l.33 – col.5 l.35. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that construction of these terms in their favor will also resolve all infringement issues in their favor. Dkt. No. 186 at 29–30. Defendants contend that the ’677 Patent describes “fast switching” as including “a Fast Resume method and a Fast Suspend method.” *Id.* at 30 (quoting ’677 Patent col.4 ll.33–35). Additionally, Defendants contend that both Fast Resume and Fast Suspend make “modified use of the BIOS power management standard.” *Id.* at 29–30 (quoting ’677 Patent col.4 ll.35–46). Defendants further contend that Fast Suspend provides a fast shutdown of an operating-system environment to the suspended state and Fast Resume involves a quick transfer to a suspended operating system environment. *Id.* Defendants additionally contend that “fast switching” is further described as a Fast Suspend of one operating system followed by a Fast Resume of another operating system, without powering down. *Id.* at 29–31 (citing ’677 Patent col.5 ll.4–8, col.10 ll.1–55, col.12 ll.1–42). This, according to Defendants, was explained during prosecution in that Fast Suspend was contrasted with Hibernate, which involves a power down. *Id.* at 30.

Defendants further respond that “power save suspend” is the alternative to “suspend for fast switching.” Dkt. No. 186 at 31. Moreover, Defendants contend that the alternative to “suspend for fast switching” is described in the ’677 Patent as “the customary and systematic termination of all services that forces the closure of all open files and cached information, shutdown of all devices and drivers, and system power off.” *Id.* (quoting ’677 Patent col.4 ll.46–50).

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’677 Patent col.4 ll.33–50, col.5 ll.4–8, col.10 ll.1–55, col.12 ll.1–42; ’677 Patent File Wrapper September 8, 2006 Amendment and Response (Defendants’ Ex. K, Dkt. No. 186-12).

Plaintiff replies that Defendants have failed to establish a specific and concrete infringement or validity dispute that requires construction of these terms. Dkt. No. 189 at 2–3 & nn.10–11.

Analysis

There appear to be two main issues in dispute. First, whether “power save suspend” is equivalent to a regular system shutdown, and second, whether “fast switching,” “fast suspending,” and “suspend for fast switching” have their ordinary meaning in the art or should be limited by the ’677 Patent’s disclosure. With respect to the first issue, “power save suspend” is the traditional suspend, or hibernate, by which the system’s power-management function saves context and powers down the system. With respect to the second issue, the “fast switching” and “fast suspending” are the invention and are not prior art concepts.

The ’677 Patent explains that BIOS power-management can suspend a computer system to save power—this is the “power save suspend.” To begin, the term “power save suspend” is used in the claims to define a “switch flag.” For instance, Claim 6 of the ’677 Patent provides:

said fast suspending step and said activating step using a switch flag, a virtual table of contents (VTOC) capable of dynamically configuring a plurality of partition tables, and BIOS ACPI enhancements, ***wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend.***

'677 Patent col.17 l.12 – col.18 l.5 (emphasis added); *see also, id.* at col.11 ll.57–61 (“Switch_Flag is a flag that can be set up in storage to differentiate Between Suspend for Fast Switching and Power Save Suspend”). The switch flag is used as part of the “Restore” function in the modified BIOS of the invention. *Id.* at col.11 ll.40–55 (listing “Restore (if Switch_flag and Suspended)”). Notably, the switch flag is not described as used for the “Suspend” function. *Id.* (listing “Suspend (then IPL, if SOS_flag)”). That is, the claims contemplate two suspend states: a “suspend for fast switching” and a “power save suspend.” The first state refers to the suspend for switching from one operating system to another, and the second to the suspend to conserve power. Thus, there is a distinction between a suspend state (where context is preserved) and a shutdown state (where context is not preserved).

The standard BIOS power-management functions can also suspend a computer's operations to save power:

Most of today's widely used operating systems are capable of hibernating or suspending their operations, using the system power management capabilities. This feature is implemented to save energy on desktop computers, and to extend the battery life of portable computers.

'677 Patent col.8 ll.34–38; *see also, id.* at col.10 ll.44–46 (“Prior art power management facilities are designed to suspend operating system environments to save power.”). This suspension allows the computer to quickly resume operation:

Suspend and resume capabilities are widely used today and present in most personal computers, be they battery-operated notebook computers or typical desktop computers. The capabilities provided by the BIOS power management standard are intended to facilitate power savings by permitting users to suspend the operations of the computer during brief interruptions, and quickly resume operation at the conclusion of the interruption.

'677 Patent col.3 ll.20–27. As explained in the '400 Patent,

ACPI is a power management specification that enables the operating system to control the amount of power given to each device attached to the computer. With ACPI, the operating system can turn off peripheral devices, such as CD-ROM players, when they are not in use, or can automatically power up the computer as soon as an input device such as a mouse is moved.

'400 Patent at col.2 ll.53–64.¹⁷ The patent explains that this power-saving function is accomplished by suspend and resume/restore functions in the BIOS. *Id.* at col.10 l.58 – col.11 l.61. On a “Power Saving” suspend, the BIOS saves the context and then powers down. This context can be restored to the computer and it can “Resume Operation of Suspended OS.” *Id.*

The '677 Patent describes modifying the BIOS functions to use its power-management functions to switch between operating-system environments (virtual computer systems). *See, e.g.*, '677 Patent col.3 ll.17–20 (“It would be an improvement to the art to use the existing BIOS power management capabilities to facilitate fast switching between operating systems in conjunction with a Super OS.”). Specifically, the specification of the '677 Patent describes:

The present invention establishes a ***Fast Switching method***, comprised of a ***Fast Resume method*** and a ***Fast Suspend method***. The Fast Resume method makes modified use of the BIOS power management standard, inherent in many computers, to allow a computer user to quickly transfer to an operating state one of a number of operating system (“OS”) environments, in a suspended state, as an alternative to the slow process of starting with Power-On Self-Test (“POST”) and loading of all the components of an operating system from the cold boot state via the initialization programs. Conversely, the Fast Suspend method makes modified use of the BIOS power management standard, providing fast shutdown of an operating system to its suspended or hibernated state. ***This Fast Suspend method is an alternative to the customary and systematic termination of all services that forces the closure of all open files and cached information, shutdown of all devices and drivers, and system power off.*** The BIOS power management standard still supports the system shutdown and/or system restart with a cold boot at the users’ discretion, when it is required for events such as a configuration change, addition or deletion of hardware components, new applications, or system upgrades.

¹⁷ The disclosure of the '400 Patent is incorporated by reference into the disclosure of the '677 Patent. '677 Patent col.6 ll.58–59, col.7 ll.4–9.

'677 Patent col.4 ll.33–55 (emphasis added). Thus, the invention of the '677 Patent enables switching from operating-system environment to another without a system shutdown and cold boot—this is the “fast switching.” This “fast switching” is also accomplished using a Fast Suspend method (i.e., “fast suspending”) and a Fast Resume method. The Fast Suspend additionally places the operating system environment in its “suspended or hibernated state.” In this context, the Court understands that the described “alternative” to fast suspending is a system shutdown, not a “power save suspend.”

Accordingly, the Court construes the terms as follows:

- “suspend for fast switching” means “suspend to allow switching from one operating system environment to another operating system environment, without power-down”;
- “power save suspend” means “suspend to save power”;
- “fast suspending” means “swapping out or remapping the current addressed space of the active program or OS, so no device power-down (power off or spin down) is required”;
- “fast switching” means “making modified use of the BIOS power management standard to fast suspend one operating system environment, followed by a fast resume of another operating system environment from a suspended state, without device power-down.”

B-8. “at least one cabinet for isolating each virtual computer system”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“at least one cabinet for isolating each virtual computer system” • ’677 Patent Claim 6	a graphical representation of virtualized software and/or data	at least one cabinet for physically separating each virtual computer system from any other virtual computer system

The Parties’ Positions

Plaintiff submits this term should not be construed because Defendants’ have not shown that there is a need to construe the term to resolve a “specific and concrete infringement or invalidity dispute” and because it appears only in a non-limiting preamble. Dkt. No. 169 at 34. Plaintiff further submits that there is no support in the intrinsic record for Defendants’ proposed “physically separating” term and that if the term must be rewritten, it should be construed as “virtual cabinet” is construed for the ’400 Patent. *Id.*

In addition to the claims themselves, Plaintiff cites the following **intrinsic evidence** to support its position: ’677 Patent col.7 l.23 – col.8 l.3; ’400 Patent, at [57] Abstract, col.2 ll.40–52, col.3 ll.1–5, col.5 ll.9–29.

Defendants respond that construing this term in their favor would result in non-infringement and that this term is limiting because it was added to the preamble during prosecution to overcome a prior art rejection. Dkt. No. 186 at 34. Defendants further respond that the “isolating” of the term requires physical separation because that is how the ’677 Patent describes “isolation.” *Id.* (citing ’677 Patent col.1 l.64 – col.2 l.3).

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: ’677 Patent col.1 l.64 – col.2 l.3; ’677 Patent File Wrapper December 19, 2006 Amendment and Response (Defendants’ Ex. I, Dkt. No. 186-10).

Plaintiff replies that Defendants have failed to establish a specific and concrete infringement or validity dispute that requires construction of these terms. Dkt. No. 189 at 2–3 & nn.10–11. Plaintiff further replies that the Claim 6 preamble is not limiting because it was not clearly used to distinguish prior art. *Id.* at 8.

Analysis

The issues in dispute are whether this term is limiting and whether cabinets necessarily physically separate virtual computer systems. The term is limiting and cabinets do not necessarily physically separate virtual computer systems.

The preamble of Claim 6 of the '677 Patent is limiting. Indeed, the patentee added the term at issue during prosecution to distinguish the prior art and described the term as a “limitation” required by the claim:

Claim 8 [that issued as Claim 6] as amended herein now requires: “... fast switching between operating systems, ... ***and at least one cabinet for isolating each virtual computer system from other virtual computer systems***, ... and using a virtual table of contents (VTOC) and BIOS ACPI enhancements”

Neither Puckette, Aguilar et al., nor Shimotono discloses ***these new claim limitations required by Claim 8 as amended***. Because these references fail to disclose the above-quoted requirements of Claim 8 as amended, these claim limitations would also be lacking from any combination of these references. Hence, Applicant's Claim 8 as amended is patentably distinct from the cited references.

'677 Patent File Wrapper December 19, 2006 Amendment and Response at 5–6 (emphasis added, ellipses in original), Dkt. No. 186-10 at 6–7.

The cabinets isolate virtual computer systems, but they do not physically separate these systems. As set forth above in the discussion of “partition,” logical separation of partitions and the virtual computer systems they comprise is important to a multi-OS environment. *See, e.g.*, '677 Patent col.1 ll.40–51 (noting the importance of ensuring “appropriate measures are employed to segregate the domain of each operating system”), col.8 ll.12–14 (“care should be taken to ensure

a partition with one type of data is not assigned to another, non-compatible Virtual PC system.”). That said, the patent expressly explains that a given partition, i.e., a logical representation of a physical segment of storage, may be assigned to more than one cabinet:

As used herein, the terms “Cabinet” and “Virtual PC” are interchangeable. For example, a user may select a Virtual PC containing Microsoft Windows 98. The Microsoft Windows 98 Cabinet may contain multiple partitions, and one boot partition. *A user can share partitions between Cabinets*. Partitions are treated like “File Drawers” that contain files and folders. Each “Drawer” of a Cabinet may have a different file format (file system), such as FAT-16, FAT-32, NTFS or EXT2FS.

’677 Patent col.7 l.62 – col.8 l.3 (emphasis added); *see also*, ’400 Patent col.5 ll.44–49 (noting that the system “allows sharing of information by Shared Device or Shared Partitions among multiple Operating Systems”). That is, a partition may be part of more than one cabinet, and therefore of more than one virtual computer system. As such, the virtual computer systems that share a partition share physical storage and are not physically separated.

Accordingly, the Court construes “at least one cabinet for isolating each virtual computer system” as follows:

- “at least one cabinet for isolating each virtual computer system” means “at least one cabinet for logically separating each virtual computer system from any other virtual computer system.”

B-9. “means for selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system” and “selection means for sequentially choosing from among said plurality of operating systems”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“means for selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system”</p> <ul style="list-style-type: none"> • ’677 Patent Claim 1 	<p>a graphical user interface (GUI) or the like for sequentially choosing from among said plurality of operating systems, or equivalents thereof</p>	<p>35 U.S.C. § 112, ¶ 6.</p> <p>Function:</p> <ul style="list-style-type: none"> • selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claim 1 is invalid under 35 U.S.C. § 112
<p>“selection means for sequentially choosing from among said plurality of operating systems”</p> <ul style="list-style-type: none"> • ’677 Patent Claim 3 	<p>graphical user interface (GUI) or the like for sequentially choosing from among said plurality of operating systems, or equivalents thereof</p>	<p>35 U.S.C. § 112, ¶ 6.</p> <p>Function:</p> <ul style="list-style-type: none"> • sequentially choosing from among said plurality of operating systems <p>Structure:</p> <ul style="list-style-type: none"> • none disclosed; this claim term is indefinite and claim 3 is invalid under 35 U.S.C. § 112

Because the parties’ arguments and proposed constructions with respect to these terms are related, the Court addresses the terms together.

The Parties’ Positions

Plaintiff submits that its “proposed structures are clearly linked or associated to the configuring function by the ’677 Patent or its prosecution history.” Dkt. No. 169 at 35, 37.

In addition to the claims themselves, Plaintiff cites the following intrinsic and extrinsic evidence to support its position: **Intrinsic evidence:** '677 Patent figs.9, 12, col.4 ll.33–55, col.5 ll.4–14, col.5 l.28 – col.6 l.5, col.6 ll.39–57, col.8 l.15 – col.9 l.16, col.9 l.58 – col.10 l.8. **Extrinsic evidence:** Rosenberg Decl. (Dkt. No. 179); Rosenberg Dep. (Dkt. No. 189-2).

Defendants respond that the functions of these terms may not be performed by a general-purpose computer and therefore the '677 Patent must disclose algorithms for performing the functions. Dkt. No. 186 at 22. Defendants further respond that what Plaintiff identifies as disclosure of structure in the patent is rather just a high-level discussion of fast switching without any disclosure of an algorithm for “selecting” a virtual computer system. *Id.* at 22–23. As such, Defendants contend that the patent fails to provide the disclosure required by 35 U.S.C. § 112, ¶ 6, and that the terms render the claims indefinite. *Id.* at 22–23.

In addition to the claims themselves, Defendants cite the following intrinsic and extrinsic evidence to support their position: **Intrinsic evidence:** '677 Patent figs.9, 12, col.4 ll.33–55, col.5 ll.4–14, col.5 l.28 – col.6 l.5, col.6 ll.39–57, col.8 l.15 – col.9 l.16, col.9 l.58 – col.10 l.8. **Extrinsic evidence:** Goodin Decl. ¶¶ 81, 83, 86–87 (Defendants' Ex. F, Dkt. No. 186-7 at 26–29).

Plaintiff replies to provide cites for what it purports is its expert's identification of structure corresponding to the function of these terms. Dkt. No. 189 at 2 & n.7. Plaintiff argues that Defendants have not established that an algorithm is required because they have not established that the claims are directed to a general-purpose computer and that a general-purpose computer cannot perform the recited function. *Id.* at 5.

Plaintiff cites further **extrinsic evidence** to support its position: Rosenberg Dep. 135:18 – 136:2, 139:7–24, 140:10 – 143:1 (Plaintiff’s Ex. A, Dkt. No. 189-2 at 35–37); Rosenberg Decl.¹⁸ ¶¶ 86, 90 (Plaintiff’s Ex. B, Dkt. No. 189-3 at 5–7).

Analysis

The issue here is whether the ’677 Patent adequately discloses structure corresponding to the recited functions. The patent provides that the Super OS GUI is used for “selecting” a virtual computer system but provides no structure for “sequentially choosing” a virtual computer system.

To begin, the parties agree that 35 U.S.C. § 112, ¶ 6 governs both the “means for selecting ...” term and the “selection means for sequentially choosing ...” term. These terms are governed by § 112, ¶ 6.

The ’677 Patent provides that the Super OS GUI is the structure corresponding to the function of “selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system.” Specifically, at column 10, lines 1–8, the ’677 Patent provides:

In the exemplary embodiment, the Super OS GUI provides a selection means for choosing one of the available operating systems and corresponding virtual computer systems.

Using the Super OS GUI, the user can double click on any cabinet to boot (or Resume) that environment. The GUI menu has separate functions for cabinets (Virtual PC’s), VTOC, and Partition utilities.

That is, the Super OS GUI is the structure used to select “any cabinet to boot (or Resume)” that operating system environment. The patent further provides that the Super OS GUI “provides a graphical user interface (“GUI”) to the Super OS functions”, including the Super OS Agent. *Id.* at

¹⁸ Plaintiff’s Ex. B was submitted as four ECF files, Dkt. No. 189-3, Dkt. No. 189-4, Dkt. No. 189-5, and Dkt No. 189-6.

col.7 ll.35–56. Therefore, the patent explains how the Super OS Agent responds to the user’s selection of a cabinet:

Referring again to FIG. 1, the user accomplishes a switch of operating system environments by clicking on the desired Virtual PC (OS).

FIGS. 6 through 9 depict a GUI for remote switching. The Super OS Agent provides users with the functionality to switch their Operating System environments. In Windows 98, as depicted in FIG. 6, a user can click on the Super OS Agent icon to activate the Super OS Agent. Once activated, the Super OS Agent will find out about other OS environments available to the user and allow him to switch to another target OS. *The GUI depicts the currently running OS, plus other available OS’s (Virtual PC’s). The user may then click on another Virtual PC. As depicted in FIG. 8, the GUI provides a confirmation box to confirm the selection.* Referring to FIG. 9, the user is then given an option to suspend or shut down the currently running OS. The next Virtual PC is set to Resume or restart by means of changes to the VTOC as requested by the Super OS Agent.

’677 Patent col.13 ll.6–23 (emphasis added). That is, the user selects the Virtual PC by clicking on it in the Super OS GUI. The patent further provides:

User activates Super OS Agent in OS1,
Selects switchover to OS2
Agent notes the (Suspend OS1/Boot OS2)
Super OS updates VTOC (or VTOC.DAT)
Dismounts Shared (non-networked) file systems

Id. at col.12 ll.19–22 (emphasis added); *see also, id.* at col.10 ll.15–17 (“In response to a user request, the Super OS Agent notes the Cabinet to be loaded next in the VTOC and sets the SOS_FLAG in the BIOS scratch storage area.”). Thus, the Super OS GUI allows the user to select a virtual computer system by, for example, clicking on the GUI representation of that system.

The ’677 Patent does not adequately provide structure corresponding to the function of “sequentially choosing from among said plurality of operating systems.” The term “sequentially” appears once in the ’677 Patent—in Claim 3. Consequently, there is no indication in the patent as to how “sequentially choosing” is performed.

Accordingly, the Court determines that: (1) “selection means for sequentially choosing from among said plurality of operating systems” fails the requirements of 35 U.S.C. § 112, ¶ 6 because the ’677 Patent does not disclose a structure for “sequentially choosing from among said plurality of operating systems,” and (2) “means for selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system” is governed by 35 U.S.C. § 112, ¶ 6 with the following function and structure:

- “means for selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system”:
 - **function:** selecting one of said virtual computer systems to become next operable before suspending a currently operational virtual computer system;
 - **structure:** Super OS GUI, and equivalents.

B-10. “means for suspending the currently operational virtual computer system in an active state”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“means for suspending the currently operational virtual computer system in an active state” • ’677 Patent Claim 1	BIOS ACPI enhancements for suspending the currently operational virtual computer system in an active state, or equivalents thereof	35 U.S.C. § 112, ¶ 6. Function: <ul style="list-style-type: none"> • suspending the currently operational virtual computer system in an active state Structure: <ul style="list-style-type: none"> • disclosed at 10:15–27

The Parties’ Positions

Plaintiff submits that its “proposed structures are clearly linked or associated to the configuring function by the ’677 Patent or its prosecution history.” Dkt. No. 169 at 36.

In addition to the claims themselves, Plaintiff cites the following **intrinsic evidence** to support its position: '677 Patent.

Defendants respond that the only disclosed algorithm for the “suspending” function is found at column 10, lines 15–27 of the '677 Patent. Dkt. No. 186 at 24. Defendants further respond that “BIOS ACPI enhancements” is not disclosure of an algorithm. *Id.*

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: '677 Patent col.10 ll.15–27.

Plaintiff replies that Defendants have not established that an algorithm is required because they have not established that the claims are directed to a general-purpose computer and that a general-purpose computer cannot perform the recited function. Dkt. No. 189 at 5.

Analysis

The parties dispute the appropriate structure for performing the “suspending the currently operational virtual computer system in an active state” function. Plaintiff’s proposed “BIOS ACPI enhancements” is not a sufficient identification of structure and Defendants’ proposal is too narrow.

The '677 Patent discloses several structures for performing the “suspend” function. For example, the patent provides:

The following steps are involved in a Super OS Suspend procedure:

1. In response to a user request, the Super OS Agent notes the Cabinet to be loaded next in the VTOC and sets the SOS_FLAG in the BIOS scratch storage area.
2. The Super OS Agent *invokes the Suspend facilities of the currently active OS environment*, causing the latter to save its current state[] in the designated Suspend storage area of the currently active Cabinet.
3. *After the suspend operation completes*, the BIOS recognizes that the SOS_FLAG is set, and therefore immediately performs the actions that normally follow a power-on of the computer, instead of turning off the computer’s power, as would normally be the case after a Suspend operation.

'677 Patent col.10 ll.13–27 (emphasis added). Here, the “suspend operation” is triggered by user input and is followed by “the actions that normally follow a power-on of the computer.” The suspending function is invoking the “Suspend facilities” to save the current state. The patent further provides a “SUSPEND” feature of “the simplified BIOS power management functions”:

```
SUSPEND:
  Save Context
  Set Flag
    If SOS_FLAG
      Then IPL
        EXIT
  Power Down.
```

Id. at col.10 1.58 – col.11 1.25. In this description, the “suspend” function includes checking the SOS_FLAG to determine whether to proceed to the BIOS’s IPL function or to Power Down. *See also, id.* at col.11 1.50 (“4. Suspend (then IPL if SOS_Flag)”). The patent further describes a “suspend” operation:

```
Suspends OS1.
  Interrupt BIOS to Suspend/IPL (OS1),
  BIOS Saves Context to Save Partition or file OS1 Save
  BIOS IPL’s Super OS (no power off).
```

Id. at col.12 ll.22–25. Finally, with reference to Figure 10, the patent describes:

Suspension step 122 involves copying the contents of the volatile memory and registers, also referred to as the system context, system contents, or “contents”, into a non-volatile reserved “Save Space” such as the Power Save partition on a secondary storage device (e.g., disk or flash memory).

Id. at col.13 1.63 – col.14 1.2. Common to these various descriptions of “suspending” is that the context of the active system is saved.

Accordingly, the Court determines that the “means for suspending the currently operational virtual computer system in an active state” is governed by 35 U.S.C. § 112, ¶ 6 with the following function and structure:

- **function:** suspending the currently operational virtual computer system in an active state;
- **structure:** ’677 Patent col.10 ll.18–21 (“Suspend facilities of the currently active OS environment”), or col.11 ll.15–20 (“SUSPEND” feature of BIOS), or col.11 l.50 (“Suspend (then IPL if SOS_Flag)”), or col.12 ll.22–25 (“Suspends OS1”), or col.13 l.63 – col.14 l.2 (“Suspension step 122”), and equivalents.

B-11. “means for making the selected virtual computer system operable into a running state”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“means for making the selected virtual computer system operable into a running state” <ul style="list-style-type: none"> • ’677 Patent Claim 1 	BIOS ACPI enhancements/solutions and VTOC Data Structure, or equivalents thereof	35 U.S.C. § 112, ¶ 6. Function: <ul style="list-style-type: none"> • making the selected virtual computer system operable into a running state Structure: <ul style="list-style-type: none"> • disclosed at 10:31–44

The Parties’ Positions

Plaintiff submits that its “proposed structures are clearly linked or associated to the configuring function by the ’677 Patent or its prosecution history.” Dkt. No. 169 at 36.

In addition to the claims themselves, Plaintiff cites the following **intrinsic evidence** to support its position: ’677 Patent figs.9, 12, col.4 ll.33–55, col.5 ll.4–14, col.5 l.28 – col.6 l.5, col.6 ll.39–57, col.8 l.15 – col.9 l.16, col.9 l.58 – col.10 l.8.

Defendants respond that the only disclosed algorithm for the “making...” function is found at column 10, lines 31–44 of the ’677 Patent. Dkt. No. 186 at 24–25. Defendants further respond that “BIOS ACPI enhancements/solutions” is not disclosure of an algorithm. *Id.* at 25.

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: '677 Patent col.10 ll.31–44.

Plaintiff replies that Defendants have not established that an algorithm is required because they have not established that the claims are directed to a general-purpose computer and that a general-purpose computer cannot perform the recited function. Dkt. No. 189 at 5.

Analysis

The parties dispute the appropriate structure for performing the “making the selected virtual computer system operable into a running state” function. Plaintiff’s proposed “BIOS ACPI enhancements/solutions and VTOC Data Structure” is not a sufficient identification of structure and Defendants’ proposal is too narrow.

The '677 Patent discloses several structures for “making the selected virtual computer system operable into a running state.” For example, the patent provides:

The Following Steps are Involved in a Super OS Resume Procedure

1. The BIOS loads the standard IPL, which activates Flash Vos.
2. Flash Vos recognizes the desired new Cabinet noted by the Flash Vos agent during the Suspend procedure, and adjusts the MBR to reflect the newly activated Cabinet.
3. Flash Vos invokes the BIOS resume procedure, which restores the state of any previously suspended OS environment from the Suspend area of the newly activated Cabinet. This process resumes the OS environment that was previously suspended in the newly activated Cabinet. If no suspended OS environment exists in the newly activated Cabinet, then the BIOS boots the operating system associated with that Cabinet. Prior art power management facilities are designed to suspend operating system environments to save power. Super OS virtual operating systems use these features to temporarily suspend an operating system environment (Virtual PC) and allow the use of system resources by another OS environment (virtual PC). In this manner, a Super OS allows rapid switching between multiple OS environments (virtual PC’s) on the same computer (physical PC), without requiring the shutdown of one OS and booting of another OS.

'677 Patent col.10 ll.29–54. That is, the Flash Vos Super Operating System adjusts the computer's master boot record (MBR) to identify the selected cabinet, the BIOS resume procedure identifies and restores a state (context) saved in the cabinet by a suspend procedure, and if no such suspended state exists, the BIOS boots the operating system of the cabinet. The patent further provides a "Start" feature of "the simplified BIOS power management functions":

Start:

Resume From Suspend (Interrupt):
 Check MBR (Suspend Partition):
 IF Suspend Context Present;
 Then If (Flag) Restore Contents
 Resume Operation of Suspended OS.

Id. at col.10 l.58 – col.11 l.25. This procedure presumes that the master boot record includes the information for the selected virtual computer system. The patent further describes a means for

making a selected virtual PC operable in the context of a Super OS switching algorithm provided at column 12, lines 16–40, reproduced here and annotated by the Court. In one instance (the upper highlighted section), the algorithm describes how to make a selected OS system (OS2) operable without a saved suspend context: the Super OS detects changes in the VTOC, updates the master boot record, and boots the OS2 system. In a second instance (the lower highlighted

<u>'677 Patent col.12 ll.16–40</u>	
<p>BIOS runs POST BIOS boots Super OS Super OS selects and boots the default OS (OS1) User activates Super OS Agent in OS1, Selects switchover to OS2 Agent notes the (Suspend OS1/Boot OS2) Super OS updates VTOC (or VTOC.DAT) Dismounts Shared (non-networked) file systems Suspends OS1. Interrupt BIOS to Suspend/IPL (OS1), BIOS Saves Context to Save Partition or file OS1 Save BIOS IPL's Super OS (no power off).</p> <p style="background-color: yellow;">Super OS runs and detects VTOC changes (next OS) Super OS Switches Virtual PC's MBR Super OS boots OS2</p> <p>User activates Super OS Agent in OS2, Selects switchover to OS1 Environment Super OS Agent notes the (Suspend Current/Resume OS1) Super OS Agent updates VTOC (or VTOC.DAT) Suspends OS2 SOS Agent dismounts Shared file systems Interrupts BIOS to Suspend/IPL (OS2), BIOS saves Context to Save Partition or File OS2 Save BIOS IPL's Super OS (no power off)</p> <p style="background-color: yellow;">Super OS runs and detects VTOC changes (next OS) Super OS switches Virtual PC MBR to OS1 BIOS Resume BIOS restores context from OS1 Save or File OS1 Save Resumes OS1 OS1 mounts shared file systems. Resuming OS1 from last suspend</p>	<p>BIOS runs POST BIOS boots Super OS Super OS selects and boots the default OS (OS1) User activates Super OS Agent in OS1, Selects switchover to OS2 Agent notes the (Suspend OS1/Boot OS2) Super OS updates VTOC (or VTOC.DAT) Dismounts Shared (non-networked) file systems Suspends OS1. Interrupt BIOS to Suspend/IPL (OS1), BIOS Saves Context to Save Partition or file OS1 Save BIOS IPL's Super OS (no power off).</p> <p style="background-color: yellow;">Super OS runs and detects VTOC changes (next OS) Super OS Switches Virtual PC's MBR Super OS boots OS2</p> <p>User activates Super OS Agent in OS2, Selects switchover to OS1 Environment Super OS Agent notes the (Suspend Current/Resume OS1) Super OS Agent updates VTOC (or VTOC.DAT) Suspends OS2 SOS Agent dismounts Shared file systems Interrupts BIOS to Suspend/IPL (OS2), BIOS saves Context to Save Partition or File OS2 Save BIOS IPL's Super OS (no power off)</p> <p style="background-color: yellow;">Super OS runs and detects VTOC changes (next OS) Super OS switches Virtual PC MBR to OS1 BIOS Resume BIOS restores context from OS1 Save or File OS1 Save Resumes OS1 OS1 mounts shared file systems. Resuming OS1 from last suspend</p>

section), the algorithm describes how to make a selected OS system (OS1) operable from a saved

suspend context: the Super OS detects changes in the VTOC, updates the master boot record, and the BIOS Resume function restores the OS1 suspend context and resumes operation of the OS1 system. *See also, id.* at fig.11 (showing “Restore MBR,” “Restore VPC Context,” and “Resume VPC” steps are executed when “Resume selected VPC?” is “yes” and “Load SOS GUI” when “Resume selected VPC” is “no”). Common to each of these descriptions is that the selected virtual computer system is made operable by booting or resuming based on a master boot record (MBR) that reflects the virtual computer system. Indeed, the ‘677 Patent explains:

Virtualized secondary storage devices are a significant pre-requisite for this switching method. *Software virtualized storage devices must apply a mechanism for the import and export of entries to the MBR partition table.* This process is described in previous sections as a means to successfully export suspended OS (Virtual PC) partitions for protection and safeguard of operation states. *It is also a means to import the proper OS (next Virtual PC), and to resume its operation with system state and MBR exactly as they were when it was suspended.*

Id. at col.12 l.63 – col.13 l.5.

Accordingly, the Court determines that “means for making the selected virtual computer system operable into a running state” is governed by 35 U.S.C. § 112, ¶ 6 with the following function and structure:

- **function:** making the selected virtual computer system operable into a running state
- **structure:** ‘677 Patent col.10 ll.31–44, or col.10 l.58 – col.11 l.25 (BIOS “Start” with MBR from selected virtual computer system), or col.12 ll.16–40 (“boots” with MBR from selected virtual computer system or “Resumes” from restored context with MBR from selected virtual computer system), or col. (“restore/resume”), fig.11 (“Restore MBR,” “Restore VPC Context,” “Resume VPC”), and equivalents.

B-12. “means for switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend”

Disputed Term	Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“means for switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend”</p> <ul style="list-style-type: none"> ’677 Patent Claim 1 	<p>BIOS ACPI enhancements/solutions with a switch flag and VTOC Data Structure, but without initialization of power-on self-test (POST) in the BIOS, or equivalents thereof</p>	<p>35 U.S.C. § 112, ¶ 6.</p> <p>Function:</p> <ul style="list-style-type: none"> switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend <p>Structure:</p> <ul style="list-style-type: none"> disclosed at 12:15–41

The Parties’ Positions

Plaintiff submits that its “proposed structures are clearly linked or associated to the configuring function by the ’677 Patent or its prosecution history.” Dkt. No. 169 at 37.

In addition to the claims themselves, Plaintiff cites the following **intrinsic evidence** to support its position: ’677 Patent figs.9, 12, col.4 ll.33–55, col.5 ll.4–14, col.5 l.28 – col.6 l.5, col.6 ll.39–57, col.8 l.15 – col.9 l.16, col.9 l.58 – col.10 l.8.

Defendants respond that the only disclosed algorithm for the “switching” function is found at column 12, lines 15–41 of the ’677 Patent. Dkt. No. 186 at 25. Defendants further respond that “BIOS ACPI enhancements/solutions” is not disclosure of an algorithm. *Id.* at 25.

In addition to the claims themselves, Defendants cite the following **intrinsic evidence** to support their position: **Intrinsic evidence:** '677 Patent col.12ll.15–41.

Plaintiff replies that Defendants have not established that an algorithm is required because they have not established that the claims are directed to a general-purpose computer and that a general-purpose computer cannot perform the recited function. Dkt. No. 189 at 5.

Analysis

The parties dispute the appropriate structure for performing the “switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend” function. Plaintiff’s proposed “BIOS ACPI enhancements/solutions with a switch flag and VTOC Data Structure” is not a sufficient identification of structure and Defendants’ proposal does not necessarily capture “using a switch flag.”

The '677 Patent discloses a structure for “switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend.” As identified by Defendants, the '677 Patent provides a switching algorithm at column 12, lines 16–40 (reproduced in the section on “means for making ...” above). That algorithm, however, does not explicitly mention use of a “switch flag.” For example, in the context of switching from the OS2 environment to the OS1 environment, the algorithm describes “BIOS restores context from OS1 Save or File OS1 Save.” '677 Patent col.12 ll.35–40. In contrast, the patent explicitly provides for use of a switch flag in the description of the Super OS BIOS operation—in which the context is restored only if the switch flag is set. *Id.* at

col.40–61 (stating “3. Restore (if Switch_flag and Suspended)” and “Switch_Flag is a flag that can be set up in storage to differentiate Between Suspend for Fast Switching and Power Save Suspend”); *see also, id.* col.11 ll.1–5 (“IF Suspend Context Present; Then If (Flag) Restore Contents”).

Accordingly, the Court determines that “means for switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend” is governed by 35 U.S.C. § 112, ¶ 6 with the following function and structure:

- **function:** switching of the virtual computer systems using a switch flag and BIOS ACPI solutions, and without initialization of power-on self test (POST) in the BIOS, wherein the switch flag is a flag that is set up in storage to differentiate between suspend for fast switching and power save suspend;
- **structure:** ’677 Patent col.10 l.63 – col.11 l.60 (“suspend”/ “restore” / “resume”), or col.12 ll.15-41 and col.11 l.48, and equivalents.

V. CONCLUSION

The Court adopts the constructions above for the disputed and agreed terms of the Asserted Patents. The Court further holds that Claims 16 and 28 of the ’400 Patent and Claim 3 of the ’677 Patent are each invalid as indefinite. Furthermore, the parties should ensure that all testimony that relates to the terms addressed in this Order is constrained by the Court’s reasoning. However, in the presence of the jury the parties should not expressly or implicitly refer to each other’s claim construction positions and should not expressly refer to any portion of this Order that is not an

actual construction adopted by the Court. The references to the claim construction process should be limited to informing the jury of the constructions adopted by the Court.

SIGNED this 22nd day of December, 2016.


ROY S. PAYNE
UNITED STATES MAGISTRATE JUDGE