

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

DROPBOX, INC. AND MICROSOFT CORPORATION,
Petitioners,

v.

DATANET, LLC,
Patent Owner.

IPR2024-00078
U.S. Patent No. 8,473,478

PETITIONERS' REPLY TO PATENT OWNER RESPONSE

TABLE OF CONTENTS

	Page
EXHIBIT LIST	iv
LIST OF PREVIOUSLY FILED EXHIBITS	iv
I. INTRODUCTION	1
II. PATENT OWNER’S PROPOSED CONSTRUCTION OF “SEARCHING” IGNORES INTRINSIC EVIDENCE AND IMPROPERLY IMPORTS REQUIREMENTS BASED ON A DICTIONARY DEFINITION.....	1
III. GROUND 1: CLAIMS 1-6 AND 8-11 ARE OBVIOUS OVER KENLEY AND LONG.....	4
A. Long’s Modification-Driven Backup Method Improved, And Was Compatible With, Kenley’s File-Based Backup System	4
1. Long Teaches A Modification-Driven Backup Approach That Was Recognized As A Fundamental Technique Applicable To Files Before The Alleged Invention Of The ’478 Patent.....	5
2. There Is No Fundamental Incompatibility Between Techniques Used In File-Based Systems And Techniques Used In Sector-Based Systems.....	7
3. Long Does Not Teach Away From Combination With File-Based Systems.....	9
4. Combining Kenley And Long Does Not Destroy The Principle Of Operation Of Kenley But Instead Improves It.....	11
B. Kenley-Long Renders Element 1.4 Obvious, Even Under Patent Owner’s Unduly Narrow Construction, Because The Combined System Searches For the Archive File In Order To Move It.....	12

C.	Kenley-Long Renders Elements 1.6, 1.9, And 1.10 Obvious, Because Kenley-Long Updates A Database To Indicate A Location Of The Archive File In Order To Track It	15
IV.	GROUND 2: CLAIMS 1-6 AND 8-11 ARE OBVIOUS OVER CABRERA, SCHNEIDER, AND WHITING	19
A.	POSITAs Would Have Implemented The Cabrera-Schneider-Whiting Combination In A Manner That Renders The Claims Obvious	19
1.	Cabrera-Schneider-Whiting Improves Upon The Operation Of Cabrera By Preserving Files Before Modification	19
2.	A POSITA Would Have Expected Cabrera-Schneider-Whiting To Work As Proposed	21
3.	Whiting’s Express Teachings Emphasize Obviousness Of The “Database” Claim Elements	23
B.	Cabrera-Schneider-Whiting Evaluates The Staging Area, And Thus Searches For The Archive File	24
C.	Cabrera-Schneider-Whiting Transmits The Archive File To A Permanent Storage Location Responsive To A Second Event, And Moves The Archive File To An Intermediate Storage Location As Part Of This Transmission	25
V.	CONCLUSION	26
	CERTIFICATE OF COMPLIANCE	27
	CERTIFICATE OF SERVICE	28

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>In re Black</i> , 778 F. App'x 911 (Fed. Cir. 2019)	11
<i>In re Fulton</i> , 391 F.3d 1195 (Fed. Cir. 2004)	10
<i>In re Keller</i> , 642 F.2d 413, (CCPA 1981)	6
<i>KSR Int'l Co. v. Teleflex, Inc.</i> , 550 U.S. 398 (2007)	11
<i>MCM Portfolio LLC v. Hewlett-Packard Co.</i> , 812 F.3d 1284 (Fed. Cir. 2015)	6
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005)	2

EXHIBIT LIST

LIST OF NEWLY FILED EXHIBITS

Exhibits concurrently filed with Petitioners' Reply to Patent Owner Response:

No.	Description
1024	Reply Declaration of David Maier, dated October 24, 2024 (“ Maier Reply Decl. ”)
1025	U.S. Patent No. 5,907,672 (“ Matze ”)
1026	Deposition Transcript of Zaydoon Jawadi, dated October 8, 2024 (“ Jawadi Tr. ”)
1027	Microsoft Press Computer Dictionary (1994) (excerpts)
1028	U.S. Provisional Application No. 60/234,211 (“ 211 Provisional ”)

LIST OF PREVIOUSLY FILED EXHIBITS

Petitioner previously filed the following exhibits:

No.	Description
1001	U.S. Patent No. 8,473,478 (“ 478 patent ”)
1002	File History of U.S. Patent No. 8,473,478 (“ 478 FH ”)
1003	Declaration of David Maier, dated October 31, 2023 (“ Maier Decl. ”)
1004	U.S. Patent No. 5,325,519 (“ Long ”)
1005	U.S. Patent No. 5,953,729 (“ Cabrera ”)
1006	U.S. Patent No. 5,778,395 (“ Whiting ”)
1007	Marc Farley, <i>Building Storage Networks</i> (2000) (excerpts) (“ Farley ”)

1008	Helen Custer, <i>Inside the Windows NT File System</i> (1994) (excerpts) (“ Custer ”)
1009	Newton’s Telecom Dictionary (2000) (excerpts)
1010	U.S. Patent No. 5,860,124 (“ Matthews ”)
1011	Ann Chervenak, et al., <i>Protecting File Systems: A Survey of Backup Techniques</i> (“ Chervenak ”)
1012	U.S. Patent No. 5,485,606 (“ Midgley ”)
1013	U.S. Patent No. 5,745,313 (“ Sliger ”)
1014	U.S. Patent No. 5,276,867 (“ Kenley ”)
1015	Abraham Silberschatz, et al., <i>Operating System Concepts</i> (3d ed 1991) (“ Silberschatz ”)
1016	U.S. Patent No. 6,016,553 (“ Schneider-US ”)
1017	Joost S.M. Verhofstad, <i>Recovery Techniques for Database Systems</i> , Computing Surveys (June 1978) (“ Verhofstad ”)
1018	U.S. Patent No. 5,771,354 (“ Crawford ”)
1019	Microsoft Press Computer Dictionary (1994) (excerpts)
1020	PCT Publication No. WO 99/12101 (“ Schneider ”)
1021	U.S. Patent No. 5,684,991 (“ Malcolm ”)
1022	Russell Sandberg, et al., <i>Design and Implementation of the Sun Network Filesystem</i> , in <i>Innovations in Internetworking</i> (Craig Partridge ed., Artech House 1988) (excerpts) (“ Sandberg ”)
1023	List of Challenged Claims

I. INTRODUCTION

The Patent Owner Response (Paper 18, “POR”) fails to rebut the Petition’s showing that claims 1-6 and 8-11 of the ’478 patent are obvious. For the reasons explained further below, Patent Owner’s arguments should be rejected, and the Challenged Claims should be found unpatentable.

II. PATENT OWNER’S PROPOSED CONSTRUCTION OF “SEARCHING” IGNORES INTRINSIC EVIDENCE AND IMPROPERLY IMPORTS REQUIREMENTS BASED ON A DICTIONARY DEFINITION

The claim construction dispute centers on whether Limitation 1.4’s “searching the first temporary storage location for the archive file” should be narrowly limited to “seeking a particular file” (as Patent Owner proposes, based on a dictionary definition) or whether it encompasses examining the first temporary storage location to see if *any* files reside there (as Petitioner proposes, based on the intrinsic evidence). EX1024, ¶¶8-11.

To reach its proposed construction, Patent Owner disregards the intrinsic evidence—including unambiguous disclosures in the specification and Applicants’ own statements in prosecution regarding “searching” a storage location for a file—and relies largely on a dictionary definition of the word “search.” POR, 6-7. A proper claim construction, however, looks at the “meaning to the ordinary artisan after reading the entire patent” and should avoid “heavy reliance on the dictionary divorced from the intrinsic evidence.” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1321

(Fed. Cir. 2005). That intrinsic evidence shows that Patent Owner’s dictionary-based construction is wrong, and that the claimed “searching” encompasses a determination of whether any files are in the temporary storage area. EX1024, ¶12.

In prosecution statements, the Applicants linked the claimed “searching ...” to parts of the application that describe “examining” a storage location to determine whether “any files” reside there. Specifically, in briefing before the Board’s predecessor (BPAI), the Applicants identified two passages and Fig. 2A element 100 as providing “exemplary support” for the disputed “searching ...” claim language. EX1002, 147/744 (citing EX1002, 726/744 (¶43, ll. 6-11), 728/744 (¶45), 712/744 (Fig. 2A element 100)). Neither of these passages nor Fig. 2A element 100 are limited to searching for a particular file. To the contrary, each describes a binary yes/no determination of whether the storage area contains *any* files. EX1024, ¶11.

For example, Fig. 2A element 100 states “SDM *examines* it’s [sic] INPUT BUFFER to see *if any files reside* there” (emphasis added):

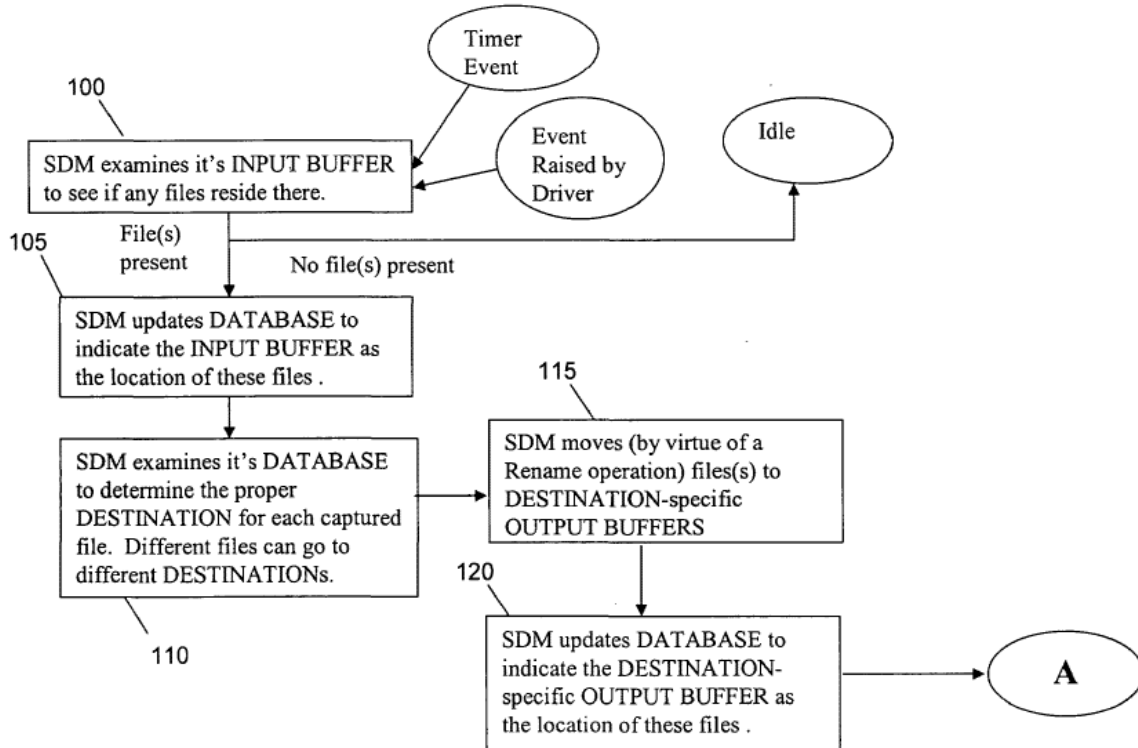


FIGURE 2A

EX1002, 712/744; *see also* EX1001, FIG. 2A. Similarly, application ¶45 (EX1002, 728/744), which corresponds to 6:21-27 of the '478 patent, describes “examin[ing] input buffer 20 to determine whether *any* archive files are stored therein” (EX1001, 6:21-23 (emphasis added)) and then resting idle “[i]f no archive files are present” (EX1001, 6:23-24) and thus no archive files are “detected” (EX1001, 6:24-27). And application ¶43, ll. 6-11 (EX1002, 726/744) similarly describes generally examining the buffer for “the presence of archive files,” not for any particular archive file. *See also* EX1001, 5:39-41; EX1028, 64/296 (provisional application describing

“initiat[ing] a process to examine the **AccuSafe Buffer Area** for new files”).
EX1024, ¶10.

This intrinsic evidence shows that the claimed “searching ...” encompasses and is met by examining the first temporary storage location to make the determination of whether *any* files reside there. Patent Owner’s narrow, dictionary-based definition should be rejected at least because (1) the intrinsic evidence provides a clear enough understanding of the term for purposes of this dispute and (2) Patent Owner’s dictionary-based definition contradicts the intrinsic evidence.
EX1024, ¶¶12-13.

III. GROUND 1: CLAIMS 1-6 AND 8-11 ARE OBVIOUS OVER KENLEY AND LONG

A. Long’s Modification-Driven Backup Method Improved, And Was Compatible With, Kenley’s File-Based Backup System

Many of the POR Ground 1 arguments are based on the false premise that Long’s modification-driven backup method was not compatible with Kenley’s file-based backup system. But the evidence shows that Long’s teachings on modification-driven backups apply to file-based backup systems. Indeed, Schneider-US (EX1016), published well before the ’478 patent’s earliest alleged priority date, expressly describes using Long’s technique for file backup. *E.g.*, Petition, 23-25. The POR does not even address Schneider-US. And, as explained below, each of Kenley and Long describes aspects of both file- and sector-based systems. Long’s

teachings would provide the expected improvements to Kenley's file-based system, and the combination renders all challenged claims unpatentable. EX1024, ¶14.

1. Long Teaches A Modification-Driven Backup Approach That Was Recognized As A Fundamental Technique Applicable To Files Before The Alleged Invention Of The '478 Patent

Patent Owner misapprehends and ignores the actual proposed Kenley-Long combination. The Kenley-Long combination proposed in the Petition uses Long's teachings on modification-driven backup. Those broad concepts involve the creation of a backup copy of data prior to modification of that data, and the migration of that data through different storage locations. Petition, 15-19 (discussing the "general-purpose backup technique" of Long), 19-25 (discussing the combination of Kenley and Long). Specifically, Long teaches initiating a backup/archiving process in response to a "modification request," i.e., "a request from an application to modify the contents of the hard disk." EX1004, 7:28-32. In order to quickly create the backup copy before performing the requested modification, Long teaches first copying the data into a memory buffer. EX1004, 7:32-46. The backup copy is then migrated to hard disk, and eventually to archive media when such archive media is available. EX1004, 7:47-53, 8:29-55. EX1024, ¶¶15, 21.

This general technique in Long is the same approach touted by Patent Owner nearly a decade later as "a large step forward in the methodology of backup technology," purportedly made by the '478 patent (POR, 3-4). *See also* Petition, 1.

Further, these general teachings in Long were already recognized by Schneider-US—a prior-art patent granted before the alleged ’478 invention—as a “fundamental method[] of recording the original state of information prior to its being altered” that was applicable to files and file-based systems. EX1016, 6:33-35 (cited by Petition, 23); *see also* EX1016, 4:27-38 (describing Long), 8:7-18 (“The basic elements of the Move Method are described in the ’519 patent” (i.e., Long)); EX1024, ¶¶16-19.

Patent Owner and its expert are completely silent with regard to the “modification request” teachings in Long, and ignore the confirmation in Schneider-US of “[t]he reasonable expectation of success, and overall obviousness, of implementing Long’s technique for a file-based system such as Kenley’s.” Petition, 23. The POR instead broadly generalizes Kenley and Long as corresponding respectively to file-based and sector-based backup systems, and argues that a supposed incompatibility between file-based and sector-based backup systems would render the combination not obvious. *See* POR, 28-30. This argument misapprehends the proposed combination. The Petition does not rely on bodily incorporation of Long’s system into Kenley, but rather applies Long’s teachings on event-driven backups that involve an application-level “modification request” and copying of the associated backup to different tiers of storage, in succession. *See MCM Portfolio LLC v. Hewlett-Packard Co.*, 812 F.3d 1284, 1294 (Fed. Cir. 2015) (quoting *In re Keller*, 642 F.2d 413, 425 (CCPA 1981)) (“The test for obviousness

is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference; nor is it that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art.”). As the Petition explained, this straightforward modification-driven conceptual approach was known to broadly apply to multi-tier backup–archival systems, including file-based systems. *E.g.*, Petition, 29. In addition to being incorrect about the alleged incompatibility (*infra* Section III.A.2), Patent Owner’s failure to address the actual proposed combination and the evidence confirming the obviousness of the combination is fatal to its arguments against Ground 1. EX1024, ¶¶15-16, 22-29.

**2. There Is No Fundamental Incompatibility
Between Techniques Used In File-Based Systems
And Techniques Used In Sector-Based Systems**

Patent Owner argues incompatibility between Kenley and Long based on the incorrect generalization that sector-level techniques “are rarely applicable” to file-level systems because “[f]ile systems and physical disks/sectors exist at separate conceptual levels.” POR, 30. This argument fails for at least two reasons.

First, Patent Owner identifies no evidence that this alleged rare applicability “has long been recognized in the art” (POR, 30), and the evidence actually shows the opposite. Background prior art such as Schneider-US expressly recognized the

applicability of Long to file-based systems, as discussed above. Petition, 23-24; *supra* Section III.A.1. Other background references show that POSITAs understood the differences between file-based and sector-based approaches to backup, and knew how to successfully apply concepts from one type of approach to the other. *E.g.*, EX1025, 1:13-5:21 (describing conventional backup techniques for file-based and sector-based approaches as of 1995), 5:31-6:15 (describing application of conventionally file-based techniques such as incremental backup and individual file retrieval to sector-based backup); EX1024, ¶22.

Second, Kenley and Long each incorporates aspects of both abstraction levels. For example, Kenley teaches a file-based backup/archiving system, but describes actions involving sectors and blocks (*e.g.*, EX1014, 16:5-32, 19:3-25), and even incorporates image recovery, a concept typically considered a sector-based approach, into its file-based backup/archiving system. EX1014, 11:20-24; *id.*, 11:25-12:14; EX2020, ¶¶57-59. Similarly, Long’s preferred embodiment for backup/archiving is allegedly sector-based, but Long also describes a “request from an application to modify the contents of the hard disk,” which spans from the supposed higher level (an application) to the lower level (the hard disk). EX1004, 7:28-29. Further, despite providing a sector-based embodiment, Long more broadly claimed a method that did not invoke sectors. *See, e.g.*, EX1004, 12:19-37 (claim 31); EX2022, 49:2-9 (discussing claim 31 and noting that “Long, itself, claims to be

broader than simply monitoring sector operations”); *see also* EX1004, Abstract (summarizing Long’s method without restricting it to sectors); Petition, 26 (quoting EX1004, Abstract); EX2022, 47:23-48:2 (explaining that a POSITA would have understood that Long’s methods would “be equally applicable to managing file page data”). EX1024, ¶¶20-21, 23-24.

In general, a POSITA knew that methods described in the context of one conceptual level were often applicable to other conceptual levels. *E.g.*, EX1007, 256-257 (“It is possible to combine the concept of image backup with file system backup.”); EX1024, ¶¶25-26. Despite Patent Owner’s conclusory assertions of incompatibility, all record evidence indicates that a POSITA would have recognized the benefits and been comfortable implementing Long’s teachings into Kenley’s file-based system.

3. Long Does Not Teach Away From Combination With File-Based Systems

The Petition explained: “Long teaches a straightforward modification-driven archival approach that applied to known multi-tier backup–archival systems.” Petition, 23 (citing EX1004, 11:54-56); EX1024, ¶27. Patent Owner nevertheless argues that Long teaches away from a combination of Long with Kenley because Long notes that “[e]rrors occurring in data-storage components [...] are far easier to detect than ‘errors originating within modules that generate or transform data’” (identified by Patent Owner as “file system level modules”). POR, 30 (quoting

EX1004, 3:7-11). But “such disclosure does not criticize, discredit, or otherwise discourage” the combination of Long’s technique of triggering backup upon a modification request with a file-based backup/archival system such as Kenley, and thus does not teach away from the combination. *In re Fulton*, 391 F.3d 1195, 1201 (Fed. Cir. 2004).

Nothing about Long’s discussion of detecting *errors* would have discouraged a POSITA from implementing Long’s broader teachings regarding detecting *modification requests* “from an application to modify the contents of the hard disk” (EX1004, 7:28-29) in a file-based backup/archival system such as Kenley. Indeed, Long teaches copying “[t]he data to be modified on the disk ... one sector at a time” (EX1004, 7:37-38), but these are “operations performed by the audit subsystem in response to a modification request.” EX1004, 7:30-32. Long’s description of a modification request triggering a backup/archival process does not criticize, discredit, or discourage its use for triggering file backup/archiving in a file-based system. Neither does Long’s discussion of “error detection and masking” or “error containment,” relied on by Patent Owner for its teaching away argument. *See* EX1004, 2:64-68, 3:1-33 (describing error detection and masking and error containment as aspects of fault-tolerant systems distinct from error recovery); EX1024, ¶28.

Long’s embodiment of “[p]rocessing circuitry [that] detects accesses to the device which would alter data thereon” (EX1004, 1:67-68) also does not teach away from a combination of Long’s teachings with Kenley, because the disclosure of a particular embodiment does not limit the ordinary creativity of a POSITA in applying teachings from that embodiment. *In re Black*, 778 F. App’x 911, 918 (Fed. Cir. 2019) (quoting *KSR Int’l Co. v. Teleflex, Inc.*, 550 U.S. 398, 420-21 (2007)) (“A [POSITA] attempting to solve a problem is not wholly restricted to the literal confines of the prior art. We may assume that ‘familiar items may have obvious uses’ and that a [POSITA] may draw upon her knowledge and creativity to solve the same problem.”). In general, nothing about Long’s disclosure of a sector-based embodiment would have discouraged a POSITA from applying Long’s teachings to Kenley’s file-based backup/archival system. EX1024, ¶¶28-29.

4. Combining Kenley And Long Does Not Destroy The Principle Of Operation Of Kenley But Instead Improves It

Patent Owner alleges that “[t]he proposed combination would necessarily change the principle of operation of either reference,” supposedly because “Kenley and Long operate at fundamentally different levels” and the combination “would require reengineering Kenley, Long, or both references.” POR, 31. This misapprehends the proposed combination, and ignores the evidence that a POSITA knew that techniques from different conceptual levels could easily be implemented in another conceptual level. EX1024, ¶¶30-31; *supra* Sections III.A.1-2.

The actual proposed combination “implement[]s Kenley’s file-based backup system with the modification-driven archival approach taught by Long,” which achieves “increased reliability by creating backup data after each modification, thus avoiding the potential loss of data created in between the timed backups of Kenley.” Petition, 19. The proposed combination does not destroy the principle of operation of Kenley in a way that renders the combination not obvious, but rather improves Kenley “by adding Long’s event-driven backup process that efficiently saved backups prior to any modifications of a file stored on hard disk.” Petition, 20. EX1024, ¶32.

**B. Kenley-Long Renders Element 1.4 Obvious,
Even Under Patent Owner’s Unduly Narrow
Construction, Because The Combined System
Searches For the Archive File In Order To Move It**

Under the proper construction, Kenley-Long searches the first temporary storage location for the archive file because it must do so in order to move that file from the memory buffer (first temporary storage location) to the audit buffer (intermediate storage location). Petition, 31-32. “[A]bsent such searching, the system would neither know what it is moving nor when the move has completed.” EX1003, ¶66; *see also* Institution Decision (Paper 9), 18. And without searching for the archive file, the Kenley-Long system would not know whether there is anything to move to the audit buffer. EX1024, ¶33.

The Kenley-Long combination also satisfies Patent Owner’s improperly narrow construction of the “searching ...” element, as the system searches for the particular archive file. A POSITA would have implemented Kenley-Long such that a file is copied into the memory buffer with such marking metadata in order to mark (i.e., identify) the copied file (i.e., the archive file) as the specific archive file (e.g., as opposed to another version of the file). EX1003, ¶¶67-68 (discussing creation of the marker in Long and updating the audit header with the marker information). EX1024, ¶34.

Further, a POSITA would have implemented Kenley-Long such that Kenley-Long updates a database to include the relevant marking metadata and the current location of the copied file (archive file) to allow the system to locate the copied file, which facilitates “accurate retrieval and movement” of the file “for both backup and recovery.” EX1003, ¶¶77, 55 (explaining how “a POSITA would naturally augment the catalog database of Kenley to store relevant information from the audit header taught by Long”), 58 (same), 68, 76-80 (explaining how Kenley-Long would update the database to indicate that the archive file is located in the memory buffer). Responsive to the triggering event (first event) to flush the memory buffer, Kenley-Long searches the first temporary storage location for the particular archive file by (1) examining the database to identify what archive files are in the memory buffer, and (2) upon identifying presence of the particular archive file in the memory buffer

(due to the database indicating that the particular archive file is located in the memory buffer), searching the memory buffer to locate the particular archive file (identifying the particular archive file based on its metadata) so that the particular archive file may be moved from the memory buffer to disk. EX1003, ¶70; EX1024, ¶34.

Though not needed, the Petition also noted how Long’s teachings on semaphores confirmed that the memory buffer is searched for an archive file prior to flushing the memory buffer. EX1004, 8:19-20; Petition, 33. Patent Owner argues that “the presence of a semaphore would merely indicate that there is data present.” POR, 16. That misses the point. When the computer system needs to write to disk (one of the three types of a “first event” that satisfies the claims), the presence of the semaphore triggers Kenley-Long to search the memory buffer for the archive file, so that it may migrate the archive file, resolve the semaphore, and carry on with its write to disk. Petition, 32-33; EX1024, ¶¶35-36.

Patent Owner cites no evidence, aside from unsupported expert assertions, for its argument that Long’s flushing of the memory buffer “would likely be done as a block copy in Long ... without searching through the memory.” POR, 12; EX1024, ¶37. This argument fails at least by focusing on Long alone and failing to squarely address the Kenley-Long combination, where *files* are stored in (and migrated from) the memory buffer. As explained in the preceding paragraphs, a POSITA would have

implemented that system so as to search the memory buffer for particular archive files, in order to, *inter alia*, know when the file copy had completed and to allow for updating of the database so that the copied file could later be efficiently located. This aligns with other file copy aspects of Kenley, which discloses “selecting files to be migrated” from one storage area to another. EX1014, 3:5-8. Kenley-Long updates its database to indicate the location of an archive file, and so would need to know, regardless of copy method, what files it moved from the memory buffer to the audit buffer on disk in order to appropriately update the database. EX1024, ¶¶37-38.

For all the reasons described in the Petition and elaborated upon above, a POSITA would have implemented Kenley-Long to search the memory buffer (temporary storage location) for the archive file responsive to a first event, in order to move the archive file to the audit buffer on the secondary disk storage, where the archive file is stored. Petition, 33-34; EX1024, ¶39.

**C. Kenley-Long Renders Elements 1.6, 1.9, And 1.10
Obvious, Because Kenley-Long Updates A Database
To Indicate A Location Of The Archive File In Order To Track It**

Elements 1.6, 1.9, and 1.10 recite “updating [a/the] database to indicate that the archive file is located” “in the temporary storage location” (1.6), “in the intermediate storage location” (1.9), and “in the second storage location” (1.10). *E.g.*, Petition, 12. The Petition explained how a POSITA would have found it obvious, in view of the complementary teachings of Kenley and Long, to update a

database with location information for an archive file to enhance “transparent access” to the archive file as it migrates between storage elements. Petition, 21-22, 36-39, 42-46. Patent Owner’s arguments miss the forest for the trees by narrowly focusing on select language in the individual references. EX1024, ¶40.

Patent Owner argues that Kenley’s catalogs would simply “indicate *which* files are stored” while “say[ing] nothing about locations of files being stored.” POR, 18-19. Without indicating the location of archive files, Kenley’s database storing backup catalogs would be useless. POSITAs would have understood that such catalogs would store location information enabling the system to locate the file. EX1027, 67 (defining “catalog” to include “location of files”). To that end, Kenley teaches a database storing catalogs of backup files that ““enhances transparent access to file attribute data, independent of whether the file is stored in the secondary storage element or has been previously migrated from the secondary storage element.”” Petition, 22 (quoting EX1014, 3:25-28), 30; EX1014, 6:35-40 (describing the database–backup catalogs), 10:34-61 (describing using the catalogs to locate and retrieve files), FIG. 1B. After creating backup copies of files, Kenley updates the database with “a catalog of all files backed up,” and these database backup catalogs are used in Kenley to locate backup files for retrieval. EX1014, 8:26-29 (cited by Petition, 22), 10:43-52 (cited by Petition, 23); EX1024, ¶¶41-42.

Notably, in Kenley, the “file attribute data” stored and maintained by these catalogs includes a “backup location identifier” comprising various location information (e.g., “the volume ID, the file starting offset, storage length and the file length”) that “enables the backup processor 12 to properly backup and restore files.” EX1014, 3:17-28, 7:45-62, 8:8-13, 8:25-28; Petition, 21. To recover “desired files,” the Kenley system “searches the file catalog” for the files, and “[o]nce backup volumes that contain the set of files to extract are identified,” the system “open[s] each volume in turn” and extracts the desired files. EX1014, 10:34-58. Far from merely storing lists of backup files, Kenley’s catalogs include the backup location identifier for each backup file, which in turn enables the system to identify which backup volume contains the backup file. EX1024, ¶42.

Similarly, Long teaches updating an “audit header that ‘contains information that is used to manage the audit buffer 30, archive media ... and the general environment of the audit subsystem.’” Petition, 15-16, 36 (quoting EX1004, 4:44-47). Long teaches updating the audit header to indicate the location of copied data in the memory buffer (temporary storage location), audit buffer on disk (intermediate storage location), or archive media (second/permanent storage location). Petition, 36-37, 42-46. These updates in Long “facilitate[] the audit subsystem’s accurate retrieval and movement of data for both backup and recovery.” Petition, 37; EX1024, ¶43.

Patent Owner repeatedly admits that the audit header is updated with location information for data when it is placed into the memory buffer. *See* POR, 12, 14, 20. Based on its discussion of updating the audit header with the latest marker information, Patent Owner seems to agree that Long teaches updating the audit header every time that a marker comprising data (archive file) is created and placed in the memory buffer (temporary storage location), moved from the memory buffer to the audit buffer (intermediate storage location), and moved to archive media (second/permanent storage location). Petition, 36-39, 42-46; EX1024, ¶44.

The Petition explained that a POSITA would have been motivated to implement these updates taught by Long with Kenley's relational database catalog, to ensure "that information on current locations of different backup files" is stored in the database. Petition, 38-39. Specifically, when implemented for a file-based system such as Kenley's, the updates-upon-storing taught by Long in the database taught by Kenley would naturally indicate the current location of the archive file, thereby ensuring that the archive file can be reliably tracked and located. *E.g.*, Petition, 36-39, 42-46. Consequently, a POSITA would have found elements 1.6, 1.9, and 1.10 obvious in view of the combined teachings of Kenley and Long. EX1024, ¶45.

**IV. GROUND 2: CLAIMS 1-6 AND 8-11 ARE
OBVIOUS OVER CABRERA, SCHNEIDER, AND WHITING**

**A. POSITAs Would Have Implemented The
Cabrera-Schneider-Whiting Combination
In A Manner That Renders The Claims Obvious**

**1. Cabrera-Schneider-Whiting
Improves Upon The Operation Of
Cabrera By Preserving Files Before Modification**

Cabrera teaches a file backup/archiving method that is initiated upon a “triggering event,” such as “a command to stage data” for backup/archiving or “any other triggering event.” EX1005, 10:56-61. A POSITA would have been motivated to trigger the file backup/archiving method of Cabrera with “a file-manipulation event (e.g., save, write, or close event), as taught by Schneider, ... in order to ‘eliminate the need to pause and make backups or decide which files should be backed up.’” Petition, 64 (quoting EX1020, 5:24-25); EX1003, ¶133. A POSITA was motivated to make the resulting combination because it provides various advantages, including “increas[ing] the reliability of backup/archiving.” Petition, 67; EX1003, ¶¶133, 139-140; EX1024, ¶¶46-47.

Patent Owner alleges that “[t]he entire premise of Cabrera is a periodic backing up of data” and modifying Cabrera as proposed is not obvious because it would “fundamentally alter the operation of Cabrera.” POR, 47-48. But the teaching in Cabrera quoted by Patent Owner as evidence of this alleged “central nature of Cabrera” itself indicates the use of a periodic backup as an embodiment “[i]n the

alternative.” POR, 47-48 (quoting EX1005, 3:45-50). Patent Owner characterizes Cabrera’s “expiration of an elapsed time” (EX1005, 10:53-55) itself as an “other alternative,” while providing no explanation for its conclusion that “[t]he backup is fundamentally a periodic process in Cabrera.” POR, 49. Cabrera’s disclosure of periodic backup as an *alternative* embodiment does not render it a fundamental aspect of Cabrera that would discourage a POSITA from implementing the proposed combination. EX1024, ¶48.

Patent Owner also alleges that the proposed combination “would completely change Cabrera” because “[i]t would eliminate, for example, steps 124 and 126 from Cabrera,” which relate to periodically determining whether a threshold amount of data is ready for backup. POR, 48-49; *see* EX1005, 10:39-49. But even Cabrera explains that “decision block 124 and time delay 126” are just one embodiment, and that yet other mechanisms for triggering backup/archiving can be used. EX1005, 10:45-61; EX1024, ¶49.

Moreover, the central premise of Cabrera is better understood as the efficient backing up and archiving of files to remote storage. *E.g.*, EX1005, Abstract, 14:35-40. Even if the use of elapsed time were the only example triggering event for initiating backup/archiving disclosed by Cabrera (it is not), a POSITA still would have been motivated to improve Cabrera by using the known technique or “fundamental method[] of recording the original state of information prior to its

being altered” taught by Schneider, because doing so would predictably preserve files before modification and improve reliability of backup/archiving. EX1020, 34:12-13; Petition, 62-68. The inclusion of an additional trigger for initiating backup does not fundamentally alter Cabrera, but instead improves it. EX1024, ¶¶50-51.

**2. A POSITA Would Have Expected
Cabrera-Schneider-Whiting To Work As Proposed**

A POSITA would have had a reasonable expectation of success implementing the file-modification event as taught by Schneider within the archive-backup system of Cabrera, because POSITAs knew how to intercept file operations and use them to trigger other actions. Petition, 64 (citing EX1020, 30:4-11; EX1021, 4:2-7, 20-25). Patent Owner’s expert agreed that capturing file operations would have been easy for a POSITA to implement. EX1026, 236:20-239:2 (discussing Interrupt 21 routine and how to implement file modification intercepts to perform other routines); *see also* EX1021, 4:2-25 (describing same). Patent Owner nevertheless argues that a POSITA would not have had a reasonable expectation of success because the proposed combination “fundamentally[] would not work.” POR, 54. But none of Patent Owner’s observations about Cabrera explain how or why the proposed combination would not work, and the argument fails because a POSITA would have reasonably expected success. EX1024, ¶52.

Patent Owner argues that the proposed combination would not work because “Cabrera’s backup operation is performed on data only *after* the data is present in

the local storage area.” POR, 50 (emphasis in original), 51 (quoting EX1005, 6:63-64) (the data file is “stored in a local storage medium, such as the hard disk for the local computer system”). Naturally, the file is already stored on the disk—otherwise, there would be no “file on the disk” to copy “before completing the write to the disk.” Petition, 67. In view of Schneider’s teachings, when a file on the disk is about to be modified or overwritten through a write, the Cabrera-Schneider-Whiting combination creates an archive copy of the file “before completing the write to the disk” so that the backup occurs before the file is overwritten on disk. Petition, 62, 64-65, 67. Nothing in Cabrera forbids other triggering events from initiating backup/archiving. Thus, a POSITA implementing a Cabrera-like system would have been motivated to initiate backup/archiving by a file-modification event, as taught by Schneider. This implementation of Cabrera-Schneider-Whiting beneficially ensures that “historic versions of a file” are backed up/archived before they are modified. Petition, 62; EX1024, ¶53.

Patent Owner’s primary evidence for this argument is FIG. 2 of Cabrera, which depicts copying of a data file to the archive system. Patent Owner argues that the data file is copied to the archive system “*after* the arrows from the data producer to the data file, not before.” POR, 53 (emphasis in original). This is uncontroversial, as Cabrera-Schneider-Whiting also copies the file from the disk to create the archive file, so naturally the data file must already be on the disk. Further, Patent Owner’s

expert acknowledged that the arrows from the data producer to the data file also represent “editing of the Data Files in Cabrera” and thus may correspond to disk writes that would overwrite the data files already stored on disk. EX2020, ¶231. Patent Owner’s observations about Cabrera’s FIG. 2 in isolation do not provide any basis for the Cabrera-Schneider-Whiting combination not working, or otherwise indicate that a POSITA would not have had a reasonable expectation of success implementing the combination. EX1024, ¶54.

**3. Whiting’s Express Teachings Emphasize
Obviousness Of The “Database” Claim Elements**

Patent Owner suggests that “the inclusion of Whiting for the database limitations basically admits that the combination of Schneider and Cabrera lacks these limitations.” POR, 54 n.3. Not so. The Petition explained that, although Cabrera does not expressly use the term “database,” Cabrera discloses a “master file table” (MFT) storing location information for files. Petition, 65. The database elements of the challenged claims are obvious in view of this disclosure alone. Whiting’s express teachings regarding a database “provide implementation details for storing and updating the location of various archived versions of a file.” Petition, 65-66. As such, Whiting’s teachings emphasize how obvious the database elements would have been to a POSITA, who knew that “the location of a backup-archive copy needed to be tracked in order to facilitate transparent access to the backup-

archive copy as it migrates (i.e., moves) through hierarchical storage locations.”

Petition, 65 (citing EX1006, 6:52-60); EX1024, ¶55.

B. Cabrera-Schneider-Whiting Evaluates The Staging Area, And Thus Searches For The Archive File

Cabrera discloses evaluating the staging area to determine whether more data should be copied from the local storage area on hard disk. Petition, 69-70; EX1005, 11:22-28. Cabrera performs this evaluation “based on any triggering event” (a first event). EX1005, 11:24-27. The determination of whether to append more data depends on how much storage space is allocated or allocable for the staging area, which in turn depends on what files are already stored in the staging area. *E.g.*, EX1005, 2:12-13, 2:44-49, 2:55-61. Under the proper construction as discussed *supra* Section II, Cabrera’s evaluation of the staging area necessarily requires searching for an archive file previously stored in the staging area but not yet migrated; otherwise, the system would not know whether there is allocated or allocable space available to add the additional archive files to the staging area. Petition, 69-70. This evaluation in Cabrera thus satisfies the element “identifying presence of the archive file in the temporary storage area,” and the proposed Cabrera-Schneider-Whiting combination renders obvious element 1.4. EX1024, ¶56.

Patent Owner notes that “staging area” and “staging file” discussed in Cabrera are different terms that refer to the same storage area. POR, 37-40. Petitioners agree.

But to be clear, the Petition does not identify the “staging file” as the archive file. Rather, the copy of the data file stored in the staging area is the archive file. Petition, 68-69; EX1024, ¶57.

C. Cabrera-Schneider-Whiting Transmits The Archive File To A Permanent Storage Location Responsive To A Second Event, And Moves The Archive File To An Intermediate Storage Location As Part Of This Transmission

The proposed Cabrera-Schneider-Whiting combination transmits the archive file from the staging area to remote storage in response to a second event, which satisfies element 1.5. Petition, 70-71. Patent Owner argues that this transmission of the archive file from the staging area to remote storage occurs without the archive file being stored in an intermediate subsystem. POR, 42. This is incorrect, as Cabrera explains that transferring the data (the archive file) from the staging area to remote storage may be implemented by “delivering the data ... to an intermediate system or subsystem which will, in turn, deliver the appropriate data to remote storage.” EX1005, 7:52-60; Petition, 73-74; EX1024, ¶58.

Although Patent Owner suggests that challenged claim 1 requires moving the archive file from the intermediate storage location to the remote storage location responsive to a second event, elements 1.5 and 1.8 do not specify such a requirement. Instead, element 1.5 simply recites “transmitting the archive file to a [permanent] storage location responsive to a second event,” while element 1.8 simply recites “moving the archive file from the temporary storage location to an intermediate

storage location.” These elements are satisfied by the initiating transfer of the archive file from the staging area (temporary storage location) to the remote storage (permanent storage location) responsive to a second event, where the archive file is moved from the staging area to the intermediate system (intermediate storage location) as part of the transmission, as performed by Cabrera-Schneider-Whiting. *See* EX1005, 7:47-60, 11:46-55. EX1024, ¶59.

V. CONCLUSION

For the reasons explained above and in the Petition, the challenged claims should be found unpatentable as obvious.

Respectfully submitted,

Dated: October 25, 2024

By: /Andrew M. Mason/

Andrew M. Mason, Reg. No. 64,034
andrew.mason@klarquist.com
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Tel: 503-595-5300
Fax: 503-595-5301

Counsel for Petitioner

**CERTIFICATE OF COMPLIANCE WITH
TYPE-VOLUME LIMITATION PURSUANT TO 37 C.F.R. § 42.24**

This brief complies with the type-volume limitation of 37 C.F.R. § 42.24(a)(1)(i).

The brief contains 5599 words, excluding the parts of the brief exempted by 37 C.F.R. § 42.24(a).

The brief has been prepared in a proportionally spaced typeface using Microsoft Word in a 14-point Times New Roman font.

Dated: October 25, 2024

By: /Andrew M. Mason/
Andrew M. Mason, Reg. No. 64,034
andrew.mason@klarquist.com
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Tel: 503-595-5300
Fax: 503-595-5301

Counsel for Petitioner

**CERTIFICATE OF SERVICE
IN COMPLIANCE WITH 37 C.F.R. § 42.6(E)(4)**

The undersigned certifies that on October 25, 2024, a complete copy of **Petitioners' Reply to Patent Owner Response and Exhibits 1024-1028** were served on counsel for Patent Owner via electronic mail as follows:

Ari J. Jaffess
Liston Abramson LLP
405 Lexington Ave, 46th Floor
New York, NY 10174
Email: ari.jaffess@listonabramson.com

Ronald Abramson
M. Michael Lewis
Gina K. Kim
Liston Abramson LLP
405 Lexington Ave, 46th Floor
New York, NY 10174
Email:
ron.abramson@listonabramson.com
michael.lewis@listonabramson.com
gina.kim@listonabramson.com

Rowan P. Smith
Messner Reeves LLP
7250 N. 16th St., Ste. 410
Phoenix, AZ 85020
Email: rsmith@messner.com

By: /Andrew M. Mason/
Andrew M. Mason, Reg. No. 64,034
andrew.mason@klarquist.com
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Tel: 503-595-5300
Fax: 503-595-5301

Counsel for Petitioner