

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

DROPBOX, INC. AND MICROSOFT CORPORATION,
Petitioners,

v.

DATANET, LLC,
Patent Owner.

IPR2024-000080
U.S. Patent No. 10,585,850

PETITIONERS' REPLY TO PATENT OWNER RESPONSE

TABLE OF CONTENTS

	Page
EXHIBIT LIST	iv
LIST OF PREVIOUSLY FILED EXHIBITS	iv
I. INTRODUCTION	1
II. CLAIM CONSTRUCTION	1
III. GROUNDS	7
A. Ground 1: Obviousness Over Schneider	7
1. Limitations 1.4, 10.4, And 18.5	7
a) Schneider Teaches “Retrieving The Restorable Representation Of The Selected Previous Version From The Remote Storage Location”	7
b) Schneider Teaches “Retrieving” And “Storing” Responsive To “A Selection To Restore”	11
c) Schneider’s Use Of A Simulated Disk Or Temporary Directory Is Irrelevant To The Proposed Grounds	12
2. Claim 19	18
3. Claim 3	19
B. Ground 2: Obviousness Over Schneider In View Of Crawford	21
C. Ground 3: Obviousness Over Sweat	21
1. Limitations 1.4, 10.4, 18.5, and 19.5	21
2. Claims 2, 3, And 11	23
D. Ground 4: Obviousness Over Sweat In View Of Schneider	25

CERTIFICATE OF COMPLIANCE.....	27
CERTIFICATE OF SERVICE	28

TABLE OF AUTHORITIES

Page(s)

Cases

<i>Cadence Pharms. Inc. v. Exela Pharmsci Inc.</i> , 780 F.3d 1364 (Fed. Cir. 2015)	2
<i>H-W Tech., L.C. v. Overstock.com, Inc.</i> , 758 F.3d 1329 (Fed. Cir. 2014)	3
<i>MPHJ Tech. Invs., LLC v. Ricoh Americas Corp.</i> , 847 F.3d 1363 (Fed. Cir. 2017)	3
<i>SuperGuide Corp. v. Directv Enters., Inc.</i> , 358 F.3d 870 (Fed. Cir. 2004)	3

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. 8,473,478 (“ 478 Patent ”)
1026	U.S. patent No. 9,218,348 (“ 348 Patent ”)
1027	Transcript of Deposition of Zaydoon Jawadi, October 8, 2024
1028	Microsoft Computer Dictionary, Fifth Edition (Microsoft Press 2002) (excerpts)
1029	Terence M. Shumaker, et al., <i>AutoCAD and its Applications Basics</i> (The Goodheart-Willcox Co. 1998) (excerpts) (“ Shumaker ”)

LIST OF PREVIOUSLY FILED EXHIBITS

Petitioner previously filed the following exhibits:

No.	Description
1001	U.S. Patent No. 10,585,850 (“ 850 Patent ”)
1002	File History of U.S. Patent No. 10,585,850 (“ 850 FH ”)
1003	U.S. Provisional Application No. 60/234,211 (“ 211 Provisional ”)
1004	Declaration of Dave Maier, dated October 30, 2023 (“ Maier Decl. ”)
1005	PCT Publication No. WO 99/12101 (“ Schneider ”)
1006	U.S. Patent No. 7,062,532 (“ Sweat ”)

No.	Description
1007	U.S. Provisional Application No. 60/126,170 (“ Sweat Provisional ”)
1008	U.S. Patent No. 5,953,729 (“ Cabrera ”)
1009	U.S. Patent No. 5,771,354 (“ Crawford ”)
1010	Joost S.M. Verhofstad, <i>Recovery Techniques for Database Systems</i> , Computing Surveys, Vol. 10, No. 2 (June 1978) (“ Verhofstad ”)
1011	Ann L. Chervenak, et al., <i>Protecting File Systems: A Survey of Backup Techniques</i> , Proceedings of Joint NASA and IEEE Mass Storage Conference (1998) (“ Chervenak ”)
1012	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 ”)
1013	Marc Farley, <i>Building Storage Networks</i> (McGraw-Hill 2000) (excerpts) (“ Farley ”)
1014	Microsoft Computer Dictionary, Fifth Edition (Microsoft Press 2002) (excerpts)
1015	Dropbox, Inc.’s Proposed Claim Constructions, Case No. 6:22-cv-01142-OLG (W.D. Tex.), July 7, 2023
1016	Datanet LLC’s Proposed Claim Constructions, Case No. 6:22-cv-01142-OLG (W.D. Tex.), July 7, 2023
1017	Dropbox, Inc.’s Opening Claim Construction Brief, Case No. 6:22-cv-01142-OLG (W.D. Tex.), July 26, 2023
1018	Datanet LLC’s Response to Opening Claim Construction Brief, Case No. 6:22-cv-01142-OLG (W.D. Tex.), August 16, 2023
1019	Rebuttal Decl. of Dr. Sam Malek in Support of Datanet LLC’s Response to Opening Claim Construction Brief, Case No. 6:22-cv-01142-OLG (W.D. Tex.), August 16, 2023
1020	Dropbox, Inc.’s Reply Claim Construction Brief, Case No. 6:22-cv-01142-OLG (W.D. Tex.), August 31, 2023

No.	Description
1021	Joint Claim Construction Statement, Case No. 6:22-cv-01142-OLG (W.D. Tex.), September 18, 2023
1022	List of Challenged Claims
1023	Comparison Chart of Claims

I. INTRODUCTION

The Patent Owner Response (“POR”) fails to rebut the Petition’s showing that claims 1-19 of the ’850 patent are unpatentable. For the reasons explained further below, PO’s arguments should be rejected and the Challenged Claims should be found unpatentable.

II. CLAIM CONSTRUCTION

The POR presents two related proposed claim constructions, with one nominally based on the language of limitations 1.4, 10.4, and 18.5 and the other on limitation 19.5.¹ POR, 12-15. Both constructions read a similar requirement into the claims, namely that the previous version of a file selected to be restored becomes the “current version” of the file not only at a local storage location (as the claims recite) but also at a remote storage location (claims 1, 10, 18) or Internet storage area network (claim 19). *Id.* EX1024, ¶13.

In advancing its proposed constructions, PO relies on descriptions of supposed “AccuSafe” software in the ’221 Provisional (EX1003). *See* POR 8-9 (citing EX1003, 139, 140, 147); EX2004, ¶¶51-55. PO’s apparent theory is that, following

¹ The POR relies on its proposed constructions only in arguing against Grounds 3 and 4. POR, 36-39, 40-41. PO does not assert that Grounds 1 and 2 fail to satisfy its proposed constructions.

restoration of a backed-up file version to a user's PC, AccuSafe's automatic backup operation backs up the newly restored file version, making that newly restored version the version most recently stored at both the user's PC and the archival storage. POR, 13; EX2004, ¶54. And according to PO, a POSITA would read such unclaimed functionality into the '850 patent's independent claims. POR, 13, 15.

This is a textbook example of improperly reading limitations from the specification into the claims. *See Cadence Pharms. Inc. v. Exela Pharmsci Inc.*, 780 F.3d 1364, 1369 (Fed. Cir. 2015) (“[E]ven if all of the embodiments discussed in the patent included a specific limitation, it would not be proper to import from the patent’s written description limitations that are not found in the claims themselves.”) (internal quotations omitted). For the reasons explained below, a POSITA would not consider the relied-on AccuSafe description to create additional unrecited requirements in the claims.

First, the claims themselves suggest that the claimed invention is neither directed to nor involves automatic file backup functionality. Where the claims require a previous file version to be stored as the “current version” on a given storage location, the claims *explicitly* recited that. *See* claims 1, 10, 18 (“storing the selected previous version ... as the current version *on the local storage location*”) (emphasis added). The claims do not recite then additionally storing the selected previous version as the current version on the remote storage location, and a POSITA would

not read such a requirement into the claims. EX1024, ¶15; *e.g.*, *SuperGuide Corp. v. Directv Enters., Inc.*, 358 F.3d 870, 875 (Fed. Cir. 2004) (“[I]t is important not to import into a claim limitations that are not a part of the claim.”). Indeed, as reflected in each preamble, the claims are directed to “restoring a file,” not archiving files as the applicants claimed in related patents. *Compare, e.g.*, EX1001, 8:63 (“A method of restoring a file....”), *with* EX1025, 8:40 (“[A] method for archiving files....”).]

Second, the Provisional describes AccuSafe’s automatic backup functionality as **optional**, further confirming that it should not be read into the claims. *See e.g.*, *H-W Tech., L.C. v. Overstock.com, Inc.*, 758 F.3d 1329, 1334 (Fed. Cir. 2014) (“The optional nature of such services negates any argument that such a limitation in the claim is implied as necessary to the invention.”); *MPHJ Tech. Invs., LLC v. Ricoh Americas Corp.*, 847 F.3d 1363, 1369 (Fed. Cir. 2017) (refusing to read in limitation described as “optional”). The Provisional states that the disclosed AccuSafe “features offered may be individually selected or not selected and controlled or not controlled by the user at their choosing.” EX1003, 139. AccuSafe gives users the ability “to control which files are included and excluded within the AccuSafe backup process” (*id.*, 156) and allows a user to set the “backup type” used for files, including

changing from “automatic” backup to “schedule” backup (*id.*, 161)². The Provisional further describes AccuSafe automatically changing the backup methodology for files from “automatic” to a different type. *Id.*, 176, 178 (“AccuSafe ... may adjust the file backup mode from ‘automatic’ to ‘interval’ or ‘scheduled’.”). A POSITA would not read optional functionality into the claims. EX1024, ¶16.

Third, a POSITA reading the Provisional would understand functionality related to selecting, previewing, and restoring archived files (concepts recited in the Challenged Claims) are distinct from functionality related to automatic file backup (which is not recited in the claims). The Provisional lists over twenty distinct features of the AccuSafe application. *Id.*, 139-79. Section “6.7 File restoration & revision control” (*id.*, 144), describes steps that are reflected in the Challenged Claims—viewing a list of archived file versions (*id.*, 144-146), previewing an archived file version (*id.*, 149-150), and restoring an archived file version (*id.*, 147)—but does not mention automatic backup of a newly restored file version. The automatic backup functionality relied on by PO is described in an entirely separate section “6.2 File change monitoring and copying.” *Id.*, 139-41 (cited in POR, 8-9). A POSITA would not read this distinct, separately described feature into the claims. *See* EX1003, 199-

² The ’850 patent explains “schedule based backup systems” as one of several conventional types of file-backup methodologies. EX1001, 1:63-67, 2:20-34.

200, 205 (distinguishing “performing the real-time backup of a user’s file without action or intervention” and “the restoration process of a file” as different use cases); EX1024, ¶17. This is further confirmed by separate related PO patents that do expressly claim automatically backing up files.³ See EX1025, 8:40-48 (claiming “creating an archive file” responsive to “detecting an instruction ... to perform an operation on an operating file”); EX1026, 8:47-54; EX1024, ¶17.

PO argues that its construction is necessary to give the term “as the current version” full meaning and effect as it would otherwise be encompassed by the claim’s other term of “storing the selected previous version [] on the local storage location.” POR, 13-14. This argument fails because storing a selected previous version of a file on a local storage location does not automatically make that version the “current version” of the file locally. Consider, for example, a system designating different file versions with sequential version numbers,⁴ having a previous version “1” backed up remotely and a current version “2” stored locally. Such a system could retrieve and store to local storage the previous version “1” without changing its

³ U.S. Patent Nos. 8,473,478 & 9,218,348; challenged in related proceedings IPR2024-00078 and-00079, respectively.

⁴ The ’221 Provisional describes AccuSafe having just such a “version control” implementation. EX1003, 200, 205, 226; EX1024, ¶18 fn. 3.

designated version number. To satisfy the term “as the current version,” however, the restored version would need to be re-designated as the current version. EX1024, ¶18. Therefore, that term meaningfully narrows the claims without need of PO’s constructions. *Id.* Notably, the only use of the term “current version” in the Provisional appears to refer to files stored locally and suggests that there is no corresponding “current version” at a remote location. EX1003, 37; EX1024, ¶19.

Finally, PO’s construction for claim 19 further fails because the act of “restoring” a file had a customary meaning to a POSITA in the context of computing. Specifically, to “restore” a file means “[t]o copy files from a backup storage device to their normal location, especially if the files are being copied to replace files that were accidentally lost or deleted.” EX1028, 452; EX1024, ¶20. The customary meaning of “restore” is consistent with the intrinsic evidence and further confirms that PO’s proposed constructions are improper.

III. GROUNDS

A. Ground 1: Obviousness Over Schneider

1. Limitations 1.4, 10.4, And 18.5

With respect to the claims’ “retrieving” step (limitation 1.4)⁵, as explained below the POR misinterprets the mapping presented by the Petition, ignores aspects of Schneider relied on by the Petition, and instead focuses on aspects of Schneider that are largely (if not entirely) irrelevant to the Petition’s mapping of the claim elements.

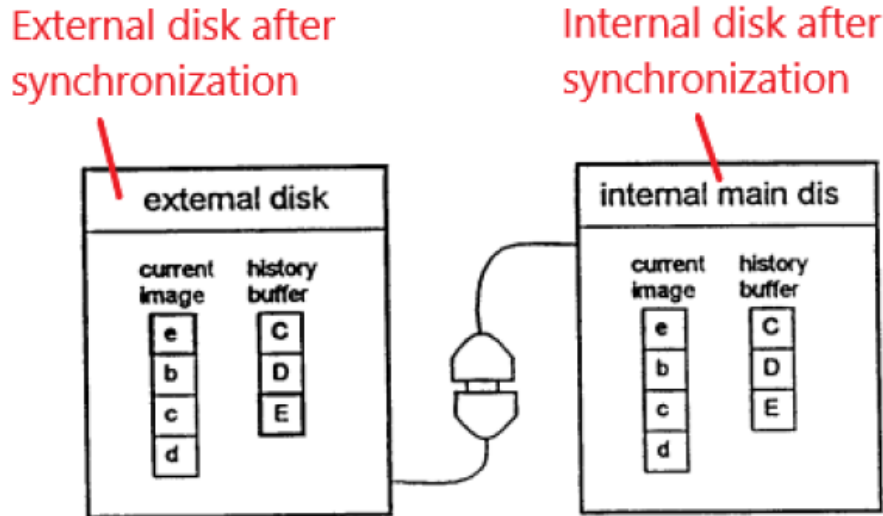
a) **Schneider Teaches “Retrieving The Restorable Representation Of The Selected Previous Version From The Remote Storage Location”**

The POR asserts that the Petition has failed to show that Schneider teaches or discloses claim limitation 1.4’s step of “retrieving [a] restorable representation of [a] selected previous version from [a] remote storage location.” POR, 18-22. The POR argument appears to misunderstand the Petition because the POR mistakenly argues that the Petition relies on a local history buffer for the “retrieving” step. For example, the POR points to the Petition’s reliance on Figure 2 and related lines 15:20-16:21. *Id.*, 19-20. These sections describe Schneider’s “Information Recovery Process,”

⁵ Arguments presented herein for limitation 1.4 apply equally to limitations 10.4 and 18.5.

which includes recreating a drive “by combining information on the original drive C and the history buffer.” EX1005:15:20-26. The POR argues that this history buffer is local to the user’s computer and “is thus not a remote storage location as required of the retrieval operation of limitation 1.4.” POR, 20. Similarly, the POR takes issue with the Petition’s reliance on lines 130:11-13 of Schneider, which disclose a user’s selecting to recover an earlier file version from a list of earlier file versions. *See* POR, 20-21 (discussing EX1005, 130:11-13). The POR argues that such a list is constructed from information contained in the history buffer, which PO again purports to be stored locally. *Id.*

By focusing on descriptions in Schneider that purportedly describe a local history buffer, the POR ignores and misrepresents the argument clearly presented in the Petition: a POSITA would utilize an external drive to store historic file versions to a remote disk as taught by Schneider and would use Schneider’s file retrieval functionality to retrieve such historic file versions from the remote disk. Petition, 16-18; *see also* EX1024, ¶21. Specifically, the Petition cites and explains Schneider’s methodology for synchronizing the internal and external disks, including a **history buffer** for both the internal and external disk. Petition, 17-18 (citing EX1005, 112:1-3, 112:24, Figs. 47A, 47I and EX1004, ¶54). As a result, content from the internal disk becomes backed up to the external disk, as shown in Fig. 47I, reproduced below.



Schneider FIG. 47I (annotated)

Petition 18. Additionally, Schneider teaches for the external disk to include additional storage, to “reach further back in time” than the internal drive. Petition 17; EX1005, 113:15-21. The Petition accordingly explains how a POSITA would implement such a system to access historic versions of files on an external drive

This obvious system would employ the software taught by Schneider to allow users to, for a given file, (i) access the historic data on the external drive and “view a list of old versions of the file” (*id.*, 113:26-28, 60:3-5); (ii) for a given previous file version, preview that version, e.g., “view its contents without actually formally retrieving the file” (*id.*, 60:21-29; *id.*, 16:1-5, 130:10-17); and (iii) “replace[] the current version” of the file with the recovered earlier version of the file (*id.*, 130:10-17). EX1004, ¶37.

Petition, 18. Further, when describing the “restorable representation[s]” of prior file versions being retrievable from a remote storage location, the Petition explains:

The presented earlier file versions are restorable because such file versions are stored in the history buffer and are used to recover versions of desired files. Id., 31:12-24. EX1004, ¶61. The restorable representations are retrievable from a remote storage location because Schneider teaches locating its historic data (from which the files are restored) on an external device, such as an external disk via a network. Id., 108:15, 112:26[.]

Petition, 21.

Finally, the Petition cites to disclosures that Schneider “allow[s] the user to view a list of earlier versions of a file [and a] selection can be made from the list and the recovered file either replaces the current version or is copied to a new file.” EX1005, 130:10-13 (cited by Petition, 29). The Petition’s argument is evident: a POSITA would implement the disclosures of Schneider to restore a previous file version using historic data stored at the external disk. EX1024, ¶22. And, where a previous file version is stored to an external disk, that external disk would be utilized to restore the file by retrieving the previous file version from the external disk. *See* EX1005, 140:20-23, Fig. 2; EX1024, ¶22. The Petition presents ample evidence from Schneider in support of this position. The POR errs by narrowly focusing on

the citations presented on Petition pages 29-30 while ignoring the context provided by the Petition's explanation of how a POSITA would have implemented Schneider using an external drive that allowed reaching "further back in time" for past versions.

**b) Schneider Teaches
"Retrieving" And "Storing"
Responsive To "A Selection To Restore"**

The POR takes a similarly tunnel-visioned approach for limitation 1.4's recitation of retrieving and storing a selected previous file version "responsive to a selection to restore" the previous version, by focusing narrowly on only those citations presented at Petition pages 29-30 while ignoring the Petition's explanation of how a POSITA would have implemented Schneider.

As discussed above and in the Petition, a POSITA would implement Schneider to store a restorable representation of previous versions of a file at a remote storage location. *Supra*, Section III.A.1.a. "The restorable representations are retrievable from a remote storage location because Schneider teaches locating its historic data (**from which the files are restored**) on an external device, such as an external disk via a network." Petition, 21 (emphasis added). Schneider's additional disclosure of selecting an earlier version of a file from a list to recover the file (EX1005, 130:10-13) demonstrates how such an implementation retrieves a restorable representation of a selected file in response to such a selection (i.e., "a selection can be made from the list" to recover an earlier file version). EX1024, ¶23.

Schneider further discloses that its file recovery process is performed responsive to a user's selection to restore a prior file version from a displayed list by explaining that the relevant simulated disk used to copy the file is created in response to the user's selection. *Id.*, 59:12-17. And the Petition's reliance on Schneider's disclosure that such "recovered file ... replaces the current version" (*id.*, 130:10-13) as a result also demonstrates how the Petition's obvious implementation of Schneider stores the previous version as current version on the local storage location in response to the same selection. EX1024, ¶23.

**c) Schneider's Use Of A
Simulated Disk Or Temporary Directory
Is Irrelevant To The Proposed Grounds**

The POR's Section VII.A.2. is confusingly styled "The Petition's broad allegations of obviousness in view of Schneider lack substantial evidence." POR, 25. This section begins by summarizing evidence relied on at Petition pages 16-18 and then quotes the Petition's assertion that, based on the evidence presented in Schneider, "a POSITA would have found it obvious to implement a system with the current versions of files stored locally on the user's PC and the historic versions on an external drive accessible via a network." POR, 25-26 (quoting Petition, 18).

Despite the section's title, the POR does not dispute that a POSITA would implement the system of Schneider to store historic versions of files on an external drive accessible via a network, but instead argues that Schneider's restoration of a

file from a remote storage location does not meet the language of limitation 1.4. POR, 26. The POR relies on Schneider's disclosure that "a prior state of the disk can be reconstructed by combining the current image with the appropriate data in the history buffer." POR, 26 (quoting EX1005, 38:30-40:3). The POR further relies on Schneider's disclosure that its file recover functionality "allow[s] the user to select a file that can be retrieved from the past, and to automatically launch the appropriate application to view the file, referencing either the file on a simulated disk or copying the file from a simulated disk to a temporary directory." POR, 27 (quoting EX1005, 60:21-24).⁶ Therefore, according to the POR, reconstruction of the file occurs locally. POR, 26-29.

However, even if reconstruction of a file is performed locally at a user's PC, as PO argues, the POR does not explain how utilizing Schneider to first retrieve such

⁶ While the POR discusses Schneider's disclosures of a "temporary directory," Schneider does not disclose, and the Petition does not rely on, using a temporary directory for restoring a file (limitation 1.4). Instead, Schneider implements a "temporary directory" at the step of viewing an old version of a file, without formally retrieving the file, after which such file is automatically discarded. EX1005, 60:21-28; EX1024, ¶25. Therefore, the Petition only describes such a temporary directory as related to the '850 patent's "preview" step of limitation 1.3. *See* Petition, 28.

files from a remote storage device fails to meet the disputed claim language. The claims require “retrieving a *restorable representation*” of a selected previous version, not a fully restored file, from a remote storage location. EX1001, limitations 1.4, 10.4, 18.5. And as the claims themselves explain, such restorable representation has “at least information *required for recovering* the version V by a computational machine.” *Id.*, limitations 1.2, 10.2, 18.3. Thus, even if recovery of a previous file version involved reconstructing the file locally, such reconstruction is performed by retrieving the archived file data (i.e., a restorable representation) from the remote storage location in order to perform such reconstruction. EX1024, ¶26.

Moreover, the Petition relies on Schneider’s *file-level* implementation. *See* Petition, 16 (explaining Schneider’s techniques for file-level recovery), 18 (explaining how a POSITA would have implemented Schneider to store historic file versions on an external drive), 22 fn. 8 (explaining that Schneider’s disclosure of an external disk for data storage and back-up applies to Schneider’s file level implementation). Schneider teaches backing up/restoring both whole files and partial files and the restoration of whole files would not require reconstruction as disclosed in relation to Schneider’s disk state recovery. *See* EX1005, 24:10-12 (“The file level implementation of present invention automatically keeps backup copies of files prior to their being changed in a circular system”), 7:8-21 (“The present invention is a method and apparatus for disk-based information recovery in computer systems....

The saved information may be disk sectors, **files**, or portions of files.”) (emphasis added), 10:25-26 (“The present invention can be implemented to work either with managing data at the disk sector level or file level (or portions of a file).”). Therefore, for file storage and backup of whole files, there would be no reconstruction of historic data with current data as the system would back up to its external storage device and subsequently restore whole files. EX1024, ¶27; *see also* EX1005, 25:4-6 (“A limited implementation of the present invention at the file level seeks to **allow access to the saved backup file copies and not simulate an entirely reverted disk.**”) (emphasis added).

To the extent the POR contends that, in the system of Schneider, previous file versions are already stored locally at the time a user selects to restore a previous file version (*see* POR 29-31), that contention is not supported by Schneider. As Schneider discloses, the act of selecting a file from a list to retrieve directly results in the operation of creating the simulated disk, copying the selected file, and closing the simulated disk. *Id.*, 59:8-17 (distinguishing retrieving a file from a list from reverting a disk to an earlier time: “In fact, the entire retrieval operation may hide the process of establishing a simulated disk. For example, the act of selecting a file to retrieve from a list, wherein the list is constructed from information in the general log, can automatically lead to the **steps of creating the appropriate simulated disk, copying the file, and closing (de-activating) the simulated disk.**”) (emphasis

added). Even where Schneider's functionality is used to view a previous file version without formally retrieving such version (*see* Petition 28-29), Schneider does not disclose a different process for subsequently restoring a desired previous file version. Thus, where a user views a previous file according to Schneider's file preview functionality prior to selecting to restore such a file, a POSITA would understand Schneider to teach creating the appropriate simulated disk to retrieve a previous file version responsive to the user making the file retrieval selection. EX1024, ¶28. This understanding makes sense in the context of a file version preview and recovery system such as Schneider's. As explained in the Petition (35-36), a POSITA understood that such a system would be used to iteratively view multiple different prior file versions without restoring such versions to determine the correct version the user is searching for. Schneider also discloses that after a file is previewed, "the viewed file is automatically discarded." EX1005, 60:25-28. Thus, a POSITA would have naturally implemented the Schneider system such that iterative previewing of multiple prior file versions followed by a selection to restore a previously viewed file version would lead the system to create the appropriate simulated disk to retrieve the prior file version from its remote storage location and copy such prior file version to the desired local storage location. EX1024, ¶28. Regardless of Schneider's use of a simulated disk, limitation 1.4 is therefore satisfied by the Petition's application of Schneider's file-level recovery implementation because a backed-up file version is

not retrieved from remote storage unless and until a user makes the selection to retrieve such file.

The POR further argues that Schneider's disclosed file-level implementation would not be applicable to remote storage based on Schneider's disclosure that its file level implementation includes "tight integration into the operating system." POR, 28 (quoting EX1005, 24:7-12). As Schneider states, its "File Method is one in which the functionality of saving prior states such that one can restore or view data from the past is incorporated into the OS." EX1005, 102:7-8. Such operating system integration allows the system of Schneider, for example, to "save an entire prior file state just before a file is opened and the intention to modify is clear." *Id.*, 24:21-23. While the POR suggests that such OS integration would preclude remote file backup, applicants themselves disclosed AccuSafe as being "fused to the operating system." EX1003, 42. Similarly, a related patent (U.S. 8,473,478) claims archiving a file and moving it to separate storage based on "detecting an instruction by an operating system." EX1025, 8:40-54. A POSITA would therefore not have considered operating system integration to be incompatible with file-level remote storage and restoration. EX1024, ¶29.

Moreover, nothing in Schneider indicates that such integration into the operating system would prevent Schneider's system from utilizing a remote external storage disk (as explicitly disclosed by Schneider) for back-up storage of files stored

by the operating system or subsequent recovery of such files from the remote storage location. EX1024, ¶30. As explained in the Petition, by the time of the '850 Patent's priority date, computer backup technologies had achieved "location transparency" such that there was no meaningful distinction between local and remote computer storage. Petition, 5-6. Accordingly, a POSITA would have implemented Schneider's file-level backup and recovery processes to store files to and retrieve them from an external drive regardless of whether parts of Schneider disclose OS integration. EX1024, ¶30.

For the reasons explained above, Schneider's disclosed use of a simulated disk or temporary directory for viewing or recovering a previous file version is irrelevant to the invalidity position presented in the Petition as such file is still retrieved from a remote storage location directly responsive to a user's selection to recover the file.

2. Claim 19

For limitation 19.3, the POR relies on largely the same argument presented for limitation 1.4. POR, 31-32. However, as discussed in Section III.A.1.c, *supra*, the Petition relies on the file-level implementation of Schneider for backup storage on a remote disk to store historical versions of files. The Petition also demonstrates that it would have been obvious to a POSITA to implement Schneider's remote storage as an Internet storage location (Petition, 24-26), which the POR does not dispute. Accordingly, where a user selects to view a historic file version that is stored

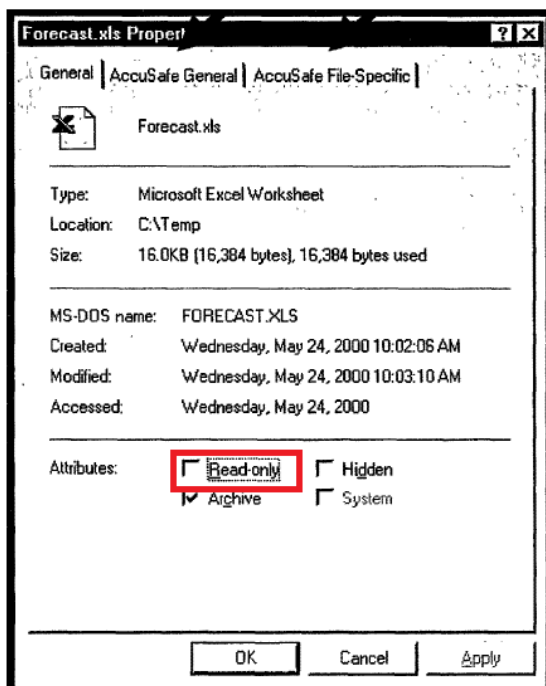
remotely at such an Internet storage location, such historic file version is received therefrom so that it can be viewed locally (for example, by using a simulated drive or temporary directory). The act of receiving the historic file version from Internet storage to view such file is sufficient to meet limitation 19.3's requirement of "previewing selected revision of said file, wherein said selected revision of said file is received from an Internet storage area network." EX1024, ¶¶32-33.

3. Claim 3

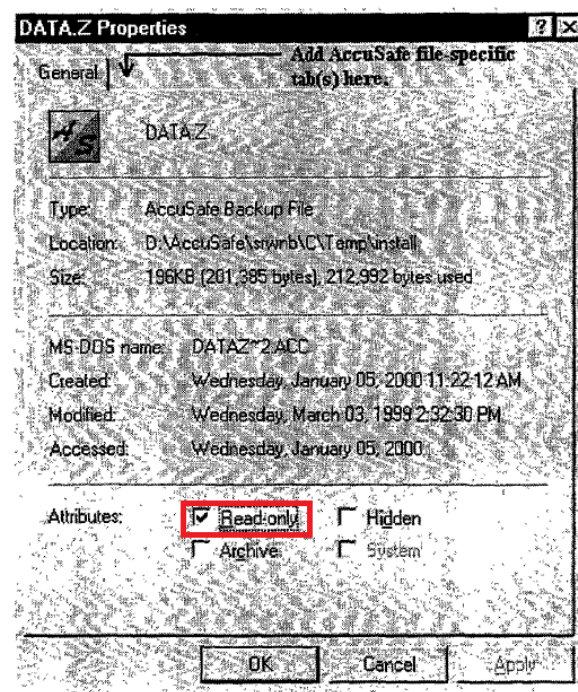
The POR asserts that "[t]he Petition's entire theory as to obviousness is that 'a POSITA would understand Schneider to, when previewing a file with an appropriate application, operate the application in read-only mode because the previewed file is stored in a temporary directory with its existence hidden from the user and is subsequently 'automatically discarded.'" POR, 32 (quoting Petition, 31-32). This is not, however, the Petition's "entire theory" as the POR contends. The POR ignores the Petition's explanation that previewing a file in a read-only mode would have been an obvious design choice because "read-only" is one of a limited number of known access permission modes for accessing files. Petition, 32. Moreover, read-only mode would be the most obvious choice of those options as it would prevent inadvertent file writes and is consistent with Schneider's purpose of only viewing the contents of a file without retrieving such a file. *See* EX1005, 60:25-27 ("This allows the user to not only know of the existence of an old version of a

file, but to view its contents without actually formally retrieving the file.”); EX1024, ¶34.

The POR attempts to distinguish a “read-only *file* or a read-only *directory*” from “launching an *application* in read-only mode.” POR, 33. This is a distinction without a difference. A POSITA would understand that an application accessing a read-only file would be doing so in a read-only mode because the application would be prevented from writing to or deleting data from such a file. *See* EX1014, 440 (“A read-only file or document can be displayed or printed but not altered in any way.”); EX1024, ¶35. Indeed, nowhere does the ’850 patent’s specification or the Provisional describe an application in a read-only mode. Instead, the Provisional presents files having a read-only property, as shown below.



EX1003, 42



EX1003, 160

Mr. Jawadi confirmed at deposition that an application operating on a read-only file would be prevented from writing to such a file.⁷ Therefore, the Provisional does not appear to distinguish between implementing a read-only file and an application using such a file in a read-only mode and a POSITA would likewise not recognize such a distinction. EX1024, ¶35.

B. Ground 2: Obviousness Over Schneider In View Of Crawford

For Ground 2, the POR relies only on its Ground 1 (Schneider) arguments regarding limitations 1.4, 10.4, and 18.5, and 19.3. POR, 34-35. The POR does not dispute that a POSITA would have applied Crawford’s teachings “to substitute the local area network ... disclosed in connection with Figure 47G of Schneider with the internet.” POR, 34-35 (citing Petition, 45, 48). For the same reasons explained above, the POR’s Ground 2 arguments also fail. *Supra* Section III.A.1; EX1024, ¶36.

C. Ground 3: Obviousness Over Sweat

1. Limitations 1.4, 10.4, 18.5, and 19.5

For claim limitations 1.4, 10.4, 18.5, and 19.5, the POR relies on its proposed claim constructions, arguing that Sweat does not satisfy the claims under such

⁷ Ex. 1027, 123:10-16 (“Q. What does the read only attribute indicate to a POSITA? A. Read only means the file ... cannot be written. If the file is read only if ... a user opens the file and attempts to save the file, the file will not be saved.”).

constructions. POR, 36-38 (claims 1, 10, 18), 38-39 (claim 19). For the reasons explained in Section II, *supra*, a POSITA would not interpret the claims as proposed by PO, and instead would give them their plain and ordinary meaning. The POR does not dispute that Sweat renders claims 1, 10, 18, and 19 obvious absent PO's strained proposed claim construction.

Even under PO's improper constructions, the unclaimed step of storing a newly restored file version as the current version at the remote storage location is satisfied by Sweat. The POR admits that Sweat discloses an "Update File" step to "ensure that the version on the local computer and the remote server both correspond to each other and are the most recent versions in time." POR, 37-38. The POR argues, however, that this does not meet the disputed claim limitations because it allegedly does not occur "responsive to a selection to restore the selected previous version," but instead in response to a subsequent manual step. POR, 38. However, the disclosure from the Provisional that PO relies on for its proposed construction also includes a subsequent manual step by a user. Specifically, the POR quotes the Provisional as disclosing that "AccuSafe records the file properties and then **does nothing until the user either closes the file without change, or makes a change and saves the file**. When the save operation occurs, AccuSafe informs the

IntelliQueuer that the file will require copying to the AccuSafe drive.”⁸ POR, 9 (quoting EX1003, 140); EX2004, ¶53 (quoting same). PO’s expert argues that “[a]s a result of these operations, ... the restored version ... also becomes the most current version in the AccuSafe drive.” EX2004, ¶54. Accordingly, the Provisional-disclosed operation that PO relies on for its proposed construction also requires an intervening user-initiated step, namely, the user opens the file and then closes the file or makes a change and saves the file. *See* EX1003, 140. If a POSITA interpreted the claims as proposed by PO, the POSITA would not exclude Sweat’s “Update File” functionality from such interpretation. EX1024, ¶¶37-38.

2. Claims 2, 3, And 11

The POR asserts that Sweat does not satisfy claims 2 and 11’s requirement that “the presentable representation is presented using an application that created the file or the selected previous version.” POR, 39-40. The POR latches on to language of Sweat stating that “[t]o view project files including drawing files and markup files, a user must install the appropriate viewer to open and view drawing files.... [T]he drawing file formats that the user is able of viewing depends on the viewer

⁸ The PO omits the first portion of the quotation as it is presented in the Provisional, which states, “**Once a user opens a file (perhaps a Word document for example),** AccuSafe records the file properties” EX1003, 140 (emphasis added).

that is installed on the computer.” POR, 39-40 (quoting EX1006, 23:39-51). The POR attempts to distinguish a “viewer” application from “an ‘editor’ or a full program.” POR, 40.⁹ This, however, is a meaningless distinction and misunderstands how a POSITA would understand the system of Sweat to be used in practice. As Sweat discloses, “[t]he use of Computer Aided Design (CAD) application programs is well known in the art.” EX1006, 1:24-25. And the file collaboration system of Sweat “let[s] a user share CAD drawing files (DWG DWF, etc.), document files, or any other type of file.” *Id.*, 1:47-49. A POSITA understood that a program used to create a file can also be used as a “viewer” to view such a file, and this is recognized by Sweat. *See id.*, 23:29-33 (“The ability to markup a drawing and store the markup files are aspects of the **drawing/viewing software utilized by the viewer** (e.g., Volo View Express™, Volo View™, **AutoCAD®**, etc.).”) ¹⁰ (emphasis added); EX1024,

⁹ The POR cites to no expert testimony in support for its arguments for claims 2, 3, and 11 and presents no evidence that a “viewer” for a given file type as described by Sweat would not include a full program used to create the file.

¹⁰ AutoCAD® is a well-known and widely utilized software program for creating, editing, and viewing Computer Aided Design files. *See, e.g.*, EX1029, 19-23, 541-42 (describing creating drawings using AutoCAD and saving as “.dwg” file extension); EX1007, 101 (defining a “Drawing (Drawing File)” as “[a]n AutoCAD

¶39. Additionally, Sweat discloses its system’s ability to “create, view, or edit markup (DWF) files” (EX1006, 11:13-14) and that such files may be previewed using Sweat’s “View in Window” preview functionality (*id.*, 23:35-36). Thus, for purposes of viewing a file, a “viewer” and a “full program” are the same, and a POSITA would recognize that Sweat contemplates a computer installed with a program for creating a file would also utilize that program as the appropriate “viewer” for viewing a file of the same file type created by that program. EX1024, ¶39.

D. Ground 4: Obviousness Over Sweat In View Of Schneider

For Ground 4, PO relies on its same claim construction arguments presented for Ground 3 (POR, 40-41), and those arguments fail as to Ground 4 for the same reasons explained for Ground 3 in Section III.C.1, *supra*; EX1024, ¶40.

drawing in DWG or DWF format. AutoCAD files are stored as drawings (DWG files.”); EX1024, ¶39.

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 5539 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 1024-1029 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