

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

REPLY DECLARATION OF DAVID MAIER

DROPBOX EXHIBIT 1024
Dropbox v. Datamet
IPR2024-00080

TABLE OF CONTENTS

	Page
EXHIBIT LIST	4
I. INTRODUCTION AND ENGAGEMENT	7
II. BACKGROUND AND QUALIFICATIONS	7
III. INFORMATION CONSIDERED	7
IV. RELEVANT LEGAL STANDARDS	8
V. SUMMARY OF OPINIONS	8
VI. PATENT OWNER’S PROPOSED CLAIM CONSTRUCTIONS ARE INCORRECT	10
VII. PATENT OWNER’S ARGUMENTS AGAINST GROUNDS 1-4 FAIL ...	19
A. Ground 1: Obviousness Over Schneider	19
1. Limitations 1.4, 10.4, And 18.5	19
a) Schneider Discloses Or Teaches “Retrieving The Restorable Representation Of The Selected Previous Version From The Remote Storage Location”	19
b) Schneider Discloses Or Teaches “Retrieving” And “Storing” Responsive To “A Selection To Restore”	23
c) Schneider’s Use Of A Simulated Disk Or Temporary Directory Is Irrelevant To The Proposed Grounds	24
2. Claim 19	31
3. Claim 3	32
B. Ground 2: Obviousness Over Schneider In View Of Crawford	34
C. Ground 3: Obviousness Over Sweat	34

1.	Limitations 1.4, 10.4, And 18.5	34
2.	Claim 19.....	35
3.	Claims 2, 3, And 11	36
D.	Ground 4: Obviousness Over Sweat In View Of Schneider	37

EXHIBIT LIST

LIST OF NEWLY FILED EXHIBITS

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

No.	Description
1024	N/A
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 ”)

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

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, David Maier, do hereby declare as follows:

I. INTRODUCTION AND ENGAGEMENT

1. I have been retained as an independent expert on behalf of Dropbox Inc. and Microsoft Corporation in connection with the above-captioned Petition for *Inter Partes* Review (“IPR”) to provide my analyses and opinions on certain technical issues related to U.S. Patent No. 10,585,850 (hereinafter “the ’850 patent”). To that end, I prepared a Declaration (EX1004) (“First Declaration”) in support of the Petition.

2. I have prepared this Reply Declaration in connection with Petitioners’ Reply in support of its IPR of the ’850 patent, which I understand is to be filed concurrently with this declaration. I respond herein to certain points raised in the Declaration of Zaydoon Jawadi (EX2004) filed by Patent Owner Datanet, LLC.

II. BACKGROUND AND QUALIFICATIONS

3. My background, qualifications, and compensation for this matter are set forth in my First Declaration which I incorporate herein by reference.

III. INFORMATION CONSIDERED

4. In preparation for this declaration, I have considered the materials discussed in this declaration, including, for example, the ’850 Patent, the prosecution history of the ’850 Patent and applications from which it derives, various background articles and materials referenced in this declaration, and the prior art

references identified in this declaration. I have also reviewed the declaration of Mr. Zaydoon Jawadi (“Jawadi”) and all the cited exhibits contained therein, submitted on behalf of the Patent Owner. I have also considered technical arguments in Datanet’s Patent Owner Response as referenced herein. I have also reviewed the transcript of the deposition of Mr. Jawadi, and all the exhibits used therein. I have also considered the Board’s decision instituting this proceeding. In addition, my opinions are further based on my education, training, experience, and knowledge in the relevant field.

5. I reserve the right to supplement or amend this declaration if additional facts and information that affect my opinions become available

IV. RELEVANT LEGAL STANDARDS

6. I am not an attorney and offer no legal opinions. For the purposes of this declaration, I have been informed about certain aspects of the law that are relevant to my analysis. My understanding of the law is summarized in my initial declaration.

V. SUMMARY OF OPINIONS

7. A POSITA would not construe claims 1, 10, 18, and 19 as proposed by Jawadi. A POSITA would give those terms their plain and ordinary meaning.

8. A POSITA would find it obvious to implement Schneider to perform the step of “responsive to a selection to restore the selected previous version,

retrieving the restorable representation of the selected previous version from the remote storage location and storing the selected previous version by the computational machine as the current version on the local storage location....” The Petition explains how Schneider teaches such a step and identifies substantial evidence from Schneider in support of such a teaching.

9. A POSITA would find it obvious that Schneider teaches the step of “previewing selected revision of said file, wherein said selected revision of said file is received from an Internet storage area network.” The Petition explains how Schneider teaches such a step and identifies substantial evidence from Schneider in support of such a teaching.

10. A POSITA would find it obvious to implement Schneider to perform the step of previewing a previous file version using an application in a read-only mode.

11. A POSITA would not apply Jawadi’s proposed claim constructions to Sweat. Even if a POSITA construed the claims as Jawadi proposes, a POSITA would understand Sweat to meet the claims under such constructions.

12. A POSITA would find it obvious to implement Sweat to perform the step of previewing a previous file version using an application that created the file or the selected previous version.

**VI. PATENT OWNER'S PROPOSED
CLAIM CONSTRUCTIONS ARE INCORRECT**

13. Jawadi presents two related proposed claim constructions. For limitations 1.4, 10.4, and 18.5, Jawadi proposes that the term “storing the selected previous version by the computational machine as the current version on the local storage location” (claims 1, 10, 18) should be construed to mean that “the version of a file now present on the user’s PC also has a corresponding version on the remote storage location, each of them corresponding to the other and being the most recent version in time of such file.” EX2004, ¶61. For limitation 19.5, Jawadi proposes that the term “restoring selected another revision of said file following said previewing step” be construed “to mean that the ‘selected another revision of said file’ is now most recent version in time on the local machine and the ‘Internet storage area network’.” EX2004, ¶65. Both constructions read a similar requirement into the claims, namely that previous version of a file selected to be restored becomes the “current version” of the file at both a local storage location and a remote storage location (claims 1, 10, 18) or Internet storage area network (claim 19). *Id.*, 13-15.

14. In advancing its proposed constructions, Jawadi apparently relies on disclosures in U.S. Provisional Application No. 60/234,221 (the “Provisional” or “’221 Provisional”) regarding operation of the AccuSafe system described therein. Specifically, Jawadi relies on the Provisional disclosures regarding AccuSafe’s implementation of automatic file backup. EX2004, ¶¶ 51-55 (citing EX1003, 139,

140, 147). Jawadi's apparent theory is that, following restoration of a backed-up file version to a user's PC, AccuSafe's disclosed automatic backup operation effects a back-up of the newly restored file version such the newly restored version becomes the most recently stored version at both the user's PC and the archival storage. *Id.*, ¶ 54. And according to Jawadi, a POSITA would read such unclaimed functionality into the '850 patent's independent claims. EX2004, ¶¶ 59-65. For the reasons explained below a POSITA would not consider the relied-on AccuSafe automatic backup functionality as relevant to interpreting the claims of the '850 patent.

15. First, the claims themselves suggest that the automatic backup functionality relied on by Jawadi was not part of the claimed invention. For example, where the '850 patent claims intend for a previous file version to be stored as the current file version at a given storage location, the claims *explicitly* identify such location. See claims 1, 10, 18 ("storing the selected previous version ... as the current version *on the local storage location*") (emphasis added). It is apparent that where the '850 patent applicant wished a particular file version to be a current version at a particular storage location, the applicant knew how to claim such a feature and explicitly did so, but only with respect to a local storage location. Thus, the claims identify one specific location (the local storage location) where a previous version is to be stored as the current version, and a POSITA would not also read into the claims from the Provisional subsequently storing the selected previous version as

the current version at a second, unclaimed location (a remote storage location). Indeed, the claims' preambles reflect that they are directed to "restoring a file," while the applicant's other related patents are directed to "archiving files." *Compare, e.g.,* EX1001, 8:63 ("A method of restoring a file...."), *with* EX1025, 8:40 ("[A] method for archiving files....").]).

16. Second, the automatic backup functionality of the AccuSafe application that Jawadi relies on is disclosed as optional in Provisional. 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 allowing a user to set the "backup type" used for files, including changing from "automatic" backup to "schedule" backup (*id.*, 161)¹. The Provisional even describes that AccuSafe will in certain circumstances change the backup methodology for files from "automatic" backup to a different type. *Id.*, 176, 178 ("AccuSafe ... may adjust the file backup mode from 'automatic' to 'interval' or 'scheduled'."). Accordingly, a POSITA would

¹ The '850 patent explains "schedule based backup systems" as one of several previously developed types of file-backup methodologies. EX1001, 1:63-67, 2:20-34. Such backup system saves and backs up files according to a set schedule.

not seek to read a purely optional functionality disclosed in the Provisional into '850 patent claims.

17. Third, a POSITA reading the '221 Provisional would understand the application's disclosed features and functionality related to selection, previewing, and restoring archived files (which are reflected in the claims of the '850 patent) are distinct from unclaimed features related to automatic file backup. The Provisional lists over twenty distinct features of the AccuSafe application. *Id.*, 139-179. One such feature is described in section "6.7 File restoration & revision control" (*id.*, 144), which section generally describes steps of 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) reflected in the claims of the '850 patent. Likewise, elsewhere in the Provisional, the functionality of "performing the real-time backup of a user's file without action or intervention" (which the Provisional labels the "File Change" functionality) and the functionality of "the restoration process of a file" (labeled as the "Restore File" functionality) are described as different "use cases". *Id.*, 199-200, 205. That these "use cases" are considered to be separate and distinct functionalities is illustrated in the figure reproduced below.

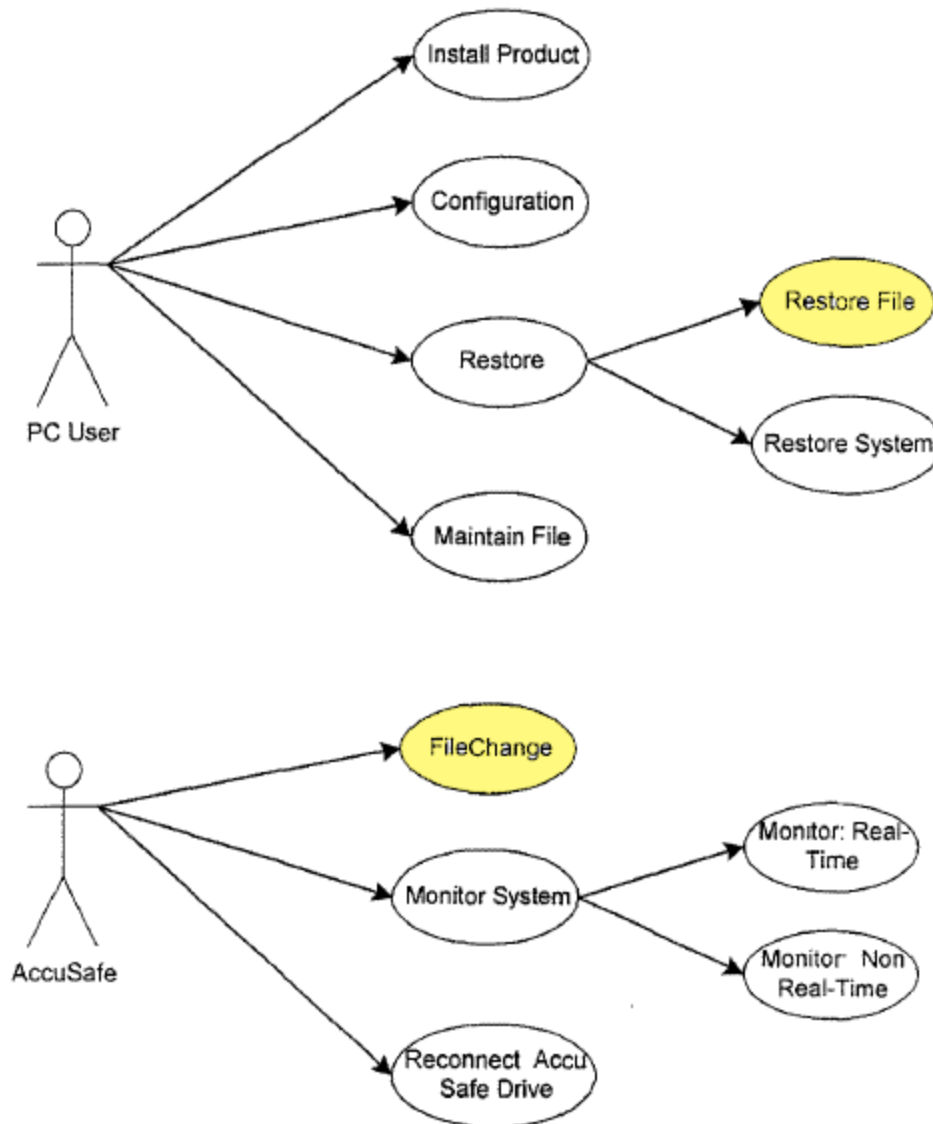


Figure 2 System Level Use Case Diagram

EX1003, 199 (highlights added). No part of the '221 Provisional discussion of AccuSafe's file restoration & revision control functionality describes reflecting a restored file version as a current version at both local storage and remote storage locations in response to a restoration procedure. The automatic backup functionality relied on by Jawadi, on the other hand, is described in an entirely separate section dedicated thereto, "6.2 File change monitoring and copying." *Id.*, 139-41 (cited in

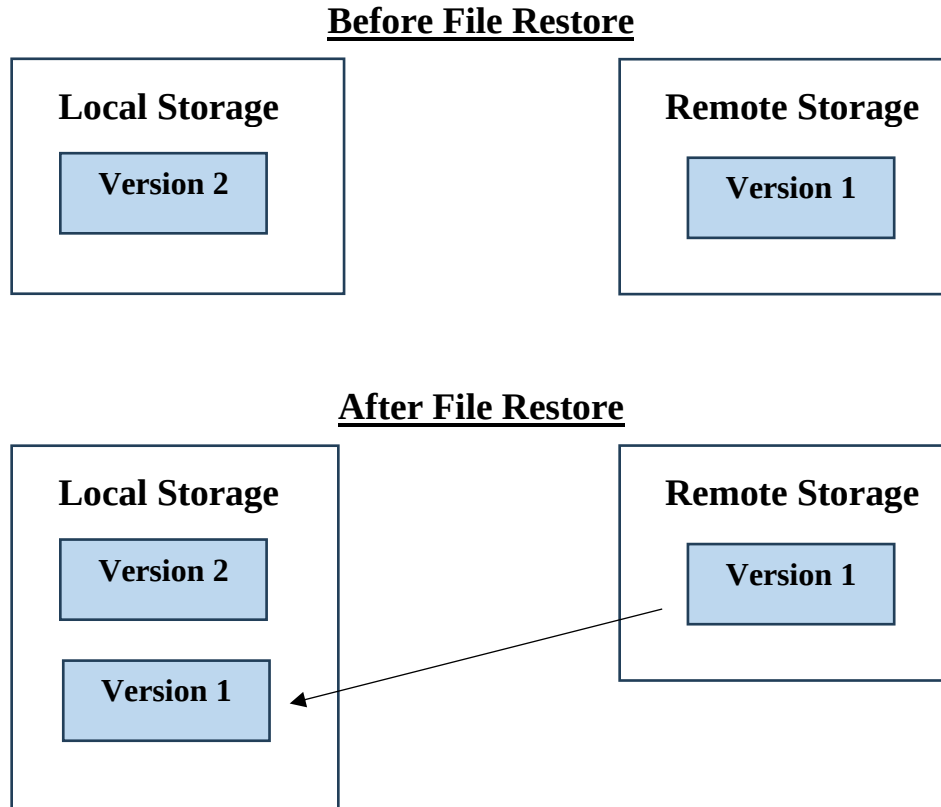
EX2004, ¶¶ 51, 53). Accordingly, a POSITA would not read a separate feature unrelated to the claimed functionality of selecting, previewing, and restoring archived file versions from the Provisional into the claims. Moreover, Jawadi’s relied-on functionality of automatically backing up files in response to a change to the file is captured in the claims of separate patents owned by the Patent Owner (U.S. Patent Nos. 8,473,478 & 9,218,348), further indicating to a POSITA that such functionality was not intended to be covered by the ’850 patent claims. *See* EX1025, 8:40-48 (claiming “creating an archive file” responsive to “detecting an instruction ... to perform an operation on an operating file”); EX1026, 8:47-54.²

18. Jawadi argues that his construction gives 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.” EX2004, ¶62. 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, having a previous version

² At deposition, Jawadi confirmed his belief that such same functionality explicitly claimed in the ’478 and ’348 patents should be read into the ’850 patent through claim construction. EX1027, 275:4-277:12.

“1” backed up remotely and a current version “2” stored locally.³ Such a system could retrieve and store to local storage the previous version “1” without changing its designation number or replacing the current version “2”. Such file restoration functionality is illustrated in the figure below, showing the files stored at a local storage location and remote storage location before and after a file restoration process to restore the previous version “1”.

³ The '221 Provisional describes such a “version control” implementation, where subsequent versions of a file are differentiated using a sequential version number. EX1003, 200, 205, 226 (“If version control is active, each version of the file will be given the same name on the AccuSafe Media Device with the exception of a version number will be placed on the file. E.g., OriginalFileName → OriginalFileName(1) → OriginalFileName(2).”)



To satisfy the term “as the current version,” however, the restored version “1” would need to be de-designated as the current version, which the above-described file restoration system does not do because version “2” at the local storage location remains the current version (i.e., it has the highest version number according to the version control system). Therefore, that term “as the current version” meaningfully narrows the claims without need of Jawadi’s constructions because making a restored version the current version locally in a file restoration system is not necessary or automatic.

19. Further, the ’221 Provisional’s *only* use of the term “current version” as applied to a file confirms that parity of a “current version” between local and

remote storage is not necessary. That is, the Provisional describes that when a current version of a file is overwritten as part of a file restore operation, such “current version will be backed up prior to the restore operation.” *Id.*, 37. This statement indicates that a “current version” of a file may not have a corresponding version backed-up at the time a restore operation is initiated, rebutting Jawadi’s suggestion that the ’850 patent requires constant parity between a local and remote storage location. This statement further contradicts Jawadi’s position as it expressly discloses that a version different from the restored version is backed up to the remote storage but makes no mention of backing up the newly restored version. The ’221 Provisional therefore does not support the constructions proposed by Jawadi.

20. Finally, Jawadi’s construction for claim 19 fails for another reason in addition to those above. The act of “restoring” a file had an accustomed 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. That is, restoring a backed-up file simply means to copy the file from backup storage to e.g., a local storage location where a user can access the file. The accustomed meaning of “restore” does not include changing the status of the file to a “current version” and nothing in the Provisional indicates that the applicant intended to change the scope of that accustomed meaning.

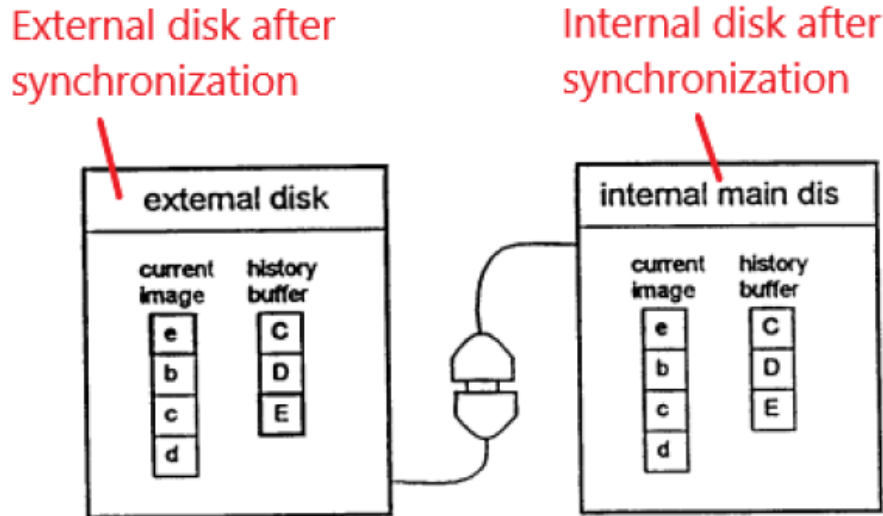
VII. PATENT OWNER’S ARGUMENTS AGAINST GROUNDS 1-4 FAIL

A. Ground 1: Obviousness Over Schneider

1. Limitations 1.4, 10.4, And 18.5

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

21. The Petition and my First Declaration presented the following position:
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. Specifically, the Petition at pages 16-18 explains Schneider’s methodology for maintaining historic data on an external disk. Schneider does so by synchronizing the internal and external disks, including a **current image and 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, the 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 as follows:

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.

22. Finally, the Petition cites to Schneider’s disclosure 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, including history buffer data, stored at the external disk. And, where a previous file version is stored to an external disk, that external drive would be utilized to restore that file by retrieving the previous file version from the external disk where it is stored. Figure 2 and

corresponding text of Schneider is cited in the Petition and my First Declaration to demonstrate how Schneider understands using a simulated drive in a recovery operation. See Petition 29-30 (citing EX1005 Fig. 2, 15:20-16:21). While Figure 2 and the corresponding text apply specifically to Schneider's disclosed implementation of full-disk recovery, a POSITA would understand how such teaching would be applicable to Schneider's expressly disclosed file-level implementation. For example, Figure 2 depicts the "virtual drive D" accessing drive "sectors in history buffer." EX1005, Fig. 2. Schneider explicitly states that in its file-level implementation, files are saved instead of disk sectors. EX1005, 140:20-23 ("What is saved. This can be either disk sectors, entire files, or portions of files. Depending on what is saved either a sector (disk) level implementation of the present invention is used, or a file (operating system) level implementation."). Accordingly, a POSITA would understand that in Schneider's file-level implementation a simulated disk as shown in Figure 2 of Schneider would be used to retrieve historical file versions (instead of disk sectors) from the applicable history buffer. In the application of Schneider presented in the Petition and my initial declaration, that applicable history buffer is located on a remote drive. Schneider therefore teaches to a POSITA the step of retrieving a restorable representation of a previous file version from a remote storage location.

**b) Schneider Discloses
Or Teaches “Retrieving” And “Storing”
Responsive To “A Selection To Restore”**

23. As discussed above, the Petition explains how Schneider teaches to a POSITA storing a file at a remote storage location. *Supra* VI.A.1.a. That is, “[t]he 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) It is apparent that the Petition’s reliance on Schneider’s disclosure that a “selection can be made from the list and the recovered file ... replaces the current version” (Petition, 29 (quoting EX1005, 130:10-13)) applies to that same stored data for individual file recovery on an external disk. That is, the prior sections of the Petition (e.g., pages 16-18, 21-22) describe how a POSITA would implement Schneider to back up restorable representations for prior file versions to an external disk for subsequent file restoration. Schneider’s additional disclosure of selecting an earlier version of a file from a list to recover the file (EX1005, 130:10-13) clearly demonstrates how such an implementation retrieves a restorable representation of a selected file from a list in response to such a selection (i.e., “a selection can be made from the list” to recover an earlier file version). A POSITA would understand that such a list would display backed up previous file versions available at the remote disk that and selecting a previous file version from that list would initiate the process

of retrieving the selected file version. 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 ("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 (deactivating) the simulated disk."). 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 relied-on implementation of Schneider stores the previous version as current version on the local storage location in response to the same selection. That is, by replacing the current version at the local storage location, the restored prior file version becomes the current version locally.

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

24. Jawadi argues that Schneider's restoration of a file from a remote storage location does not meet the language of limitation 1.4. EX2004, ¶¶ 85-99. Jawadi relies, for example, on Schneider's disclosure that its file recovery 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.” EX2004, ¶96 (quoting EX1005, 60:21-24). Therefore, according to Jawadi, reconstruction of the file occurs locally. EX2004, ¶¶ 90-96.

25. Schneider does not disclose using a temporary directory for restoring a file. 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.

26. Even if reconstruction of a file is performed locally at a user’s PC the claim language is satisfied by the Petition’s application of Schneider. The claims merely require “retrieving a *restorable representation*” of a selected previous version, not a fully restored file, from 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. That a restorable representation of a previous file version is not a restored file is further indicated by dependent claims 7 and 8, which respectively claim that a restorable representation “comprises one or more of a compressed and encrypted version” and “comprises an incremental change to the file.” These show that the ’850 patent recognizes that further processing of the restorable representation of a remotely stored previous file version (such as decompression, decryption, or reconstruction) may be necessary to effect a

restoration of the previous file version. Thus, even if recovery of a previous file version from remote storage includes reconstruction of the file locally, such reconstruction is performed by retrieving the archived file data required for such reconstruction (i.e., a restorable representation) from a remote storage location.

27. Moreover, while Schneider discloses recovering “*a prior state of the disk*” by reconstructing such a disk “by combining a current image with appropriate data in the history buffer” (EX1005, 39:30-40:3), Schneider discloses, and the Petition relies on, other implementations of Schneider. Specifically, 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 specifically discloses that its file-level implementation is applicable to whole files. 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).”). Thus, where file storage and backup are implemented at the file level, there would be no need for any reconstruction of historic data with current data as the system would back up and store to its external storage device and subsequently restore whole files. *See* EX1005, 25:4-6, (“A limited implementation of the present invention at the file level only seeks to **allow access to the saved backup file copies and not simulate an entirely reverted disk.**”).

28. A POSITA would understand that in the Petition’s application of Schneider to recover prior file versions from a remote drive, the disclosures of Schneider show that retrieval of a selected prior file version to be restored occurs directly in response to such a selection, even where a user previously selects to view the file without formally restoring it (according to the file preview functionality disclosed 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. EX1005, 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 preview functionality is used to view a previous file version without formally retrieving such version (*see* Petition 28-29), Schneider does not disclose different functionality for subsequently restoring a desired previous file version and a POSITA would therefore understand Schneider’s disclosure of creating the appropriate simulated disk upon a user’s selection to retrieve the file to be applicable to a subsequent request to restore after viewing the file. Further, this understanding makes sense in the context of a file version preview and recovery system such as Schneider’s. For example, as explained in the Petition (35-36), a POSITA understood that such a system would be used iteratively to view multiple different prior file versions, without restoring such versions, to determine the correct version the user is searching for. Schneider discloses that after a file is previewed, “the viewed file is automatically discarded” EX1005, 60:25-28. A POSITA therefore would naturally implement Schneider’s system such that, where a user iteratively previews multiple prior file versions and then makes a selection to restore a previously viewed file version that is no longer available locally (the viewed file having been discarded), the system would 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. Therefore, regardless of Schneider’s use of a simulated disk, limitation 1.4 is 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.

29. Jawadi 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. EX2004, ¶90 (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." EX1005, 24:21-23. Indeed, a patent to which the '850 patent claims priority and with which it shares a written description and figures (U.S. Patent No. 8,473,478) claims precisely such operating system implementation of storing and archiving a file. EX1025, 8:40-44 ("In a computing device, a method for archiving files comprising: **detecting an instruction by an operating system** to perform an operation on an operating file; creating an archive file from the operating file...."); see also EX1003, 42 ("AccuSafe is fused to the operating system the user is already familiar with, for maximum ease of use and functionality."). A POSITA would

therefore not have considered operating system integration to be incompatible with file-level remote storage and restoration.

30. 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. As explained in the Petition and my First Declaration, 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; EX1004, ¶34. 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.

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

2. Claim 19

32. For limitation 19.3, Jawadi relies on largely the same argument presented for limitation 1.4. EX2004, ¶100. However, as discussed in Section VI.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. 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 from there 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.”

33. Jawadi fails to explain why local implementation of a simulated disk or temporary directory to view a previous file version received from an Internet storage area does not satisfy claim 19.3. As the Board recognized in instituting this proceeding, “the ’850 patent seems to suggest that the preview mechanism for its file restoration feature creates a temporary directory, similar to Schneider. The ’221 Provisional describes a ‘temporary directory used by the restore operation,’ e.g.,

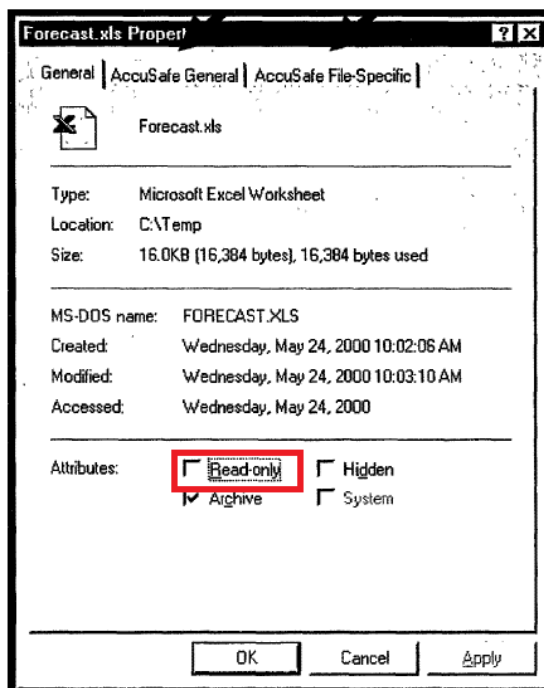
“TempDirForRestore”=“C:\\Program Files\\AccuSafe\\Temp\\ Restore.”” Paper 8, 19-20 (citing EX1003, 78). Therefore, Schneider’s utilizing a simulated disk or temporary directory to preview a historic file version retrieved from remote backup is consistent with the system of the Provisional.

3. Claim 3

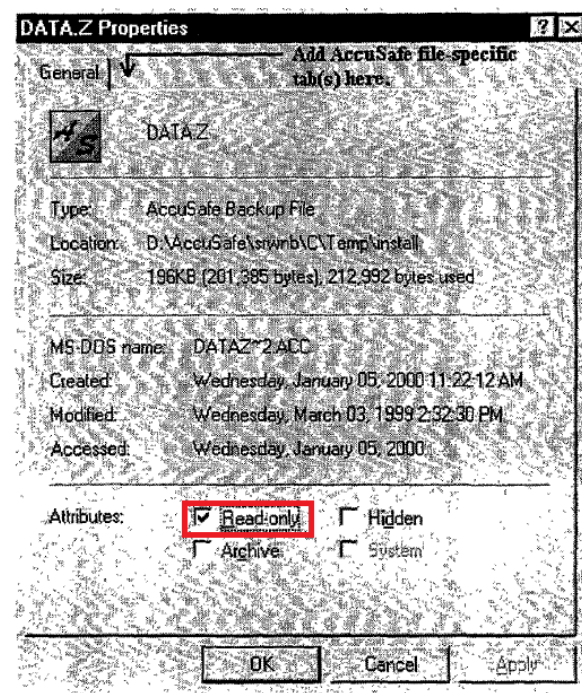
34. As explained in the Petition, previewing a file in a read-only mode would have been an obvious design choice because “read-only” is one of a limited number of known access permission modes for accessing files. Petition, 32. Moreover, read-only mode would be the most obvious choice of those options 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.”).

35. Jawadi attempts to distinguish a “read-only *file* or a read-only *directory*” from “launching an *application* in read-only mode.” EX2004, ¶ 105. 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.”). That is, a POSITA would understand that where an application

operates in such a way that it is prevented from performing write operations on a file, such an application is operating in a read-only mode. Further, 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

Jawadi confirmed at deposition that an application operating on a read-only file would be prevented from writing to such a file.⁴ Therefore, the Provisional does not

⁴ EX1027, 123:10-16 (“Q. What does the read only attributed 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.”).

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.

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

36. For Ground 2, Jawadi relies only on his Ground 1 arguments regarding limitations 1.4, 10.4, and 18.5, and 19.3. EX2004, ¶¶ 112-114. For the same reasons explained above, Jawadi’s Ground 2 arguments also fail. *Supra* Section VI.A.1.

C. Ground 3: Obviousness Over Sweat

1. Limitations 1.4, 10.4, And 18.5

37. Jawadi 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.” EX2004, ¶¶ 119-120. Jawadi argues, however, that function 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. *Id.*, ¶ 120. However, the disclosure from the Provisional that Jawadi relies on for its proposed construction similarly relies on a subsequent manual step by a user. Specifically, the Jawadi 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.”⁵ EX2004, ¶53 (emphasis added). Jawadi argues that “[a]s a result of these operations, ... the restored version, which becomes the current version on the user’s PC, also becomes the most current version in the AccuSafe drive.” *Id.*, ¶54. Accordingly, the Provisional-disclosed operation that Jawadi relies on for his proposed construction 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. The functionality of Sweat is therefore commensurate with the Provisional disclosure relied on by Jawadi for his proposed construction and a POSITA would not interpret the claim language to exclude such functionality.

2. Claim 19

38. As explained in Section VI.C.1, *supra*, Sweat’s disclosure of its “Download” function is commensurate to the Provisional’s disclosure of the same functionality that Jawadi relies upon for its proposed construction. Therefore, a POSITA would understand claim 19 to be satisfied by Sweat even under Jawadi’s proposed construction.

⁵ Jawadi 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.

3. Claims 2, 3, And 11

39. I understand that 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's argument relies on 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 that is installed on the computer." POR 39-40 (quoting EX1006, 23:39-51). The POR thereby attempts to distinguish a "viewer" application from "an 'editor' or a full program." POR, 40. This, however, is a meaningless distinction in the context of Sweat and misunderstands how a POSITA would understand the system of Sweat to be used in practice, because a POSITA understood that a full program that created a file is also a viewer for such a file. 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 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.).”) (emphasis added). AutoCAD®, for example, is a well-known and widely utilized software for creating Computer Aided Design Files. *See, e.g.*, EX1029, 19 (explaining that CAD software “is used by drafters, designers, and engineers to develop designs and drawings, and to plot them on paper or film”), 20-23, 541-542 (describing creating drawings using AutoCAD and saving as “.dwg” file extension); EX1007, 101 (defining a “Drawing (Drawing File) as “An AutoCAD drawing in DWG or DWF format. AutoCAD files are stored as drawings (DWG) files.”). 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 user’s 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.

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

40. For Ground 4, Jawadi relies on his same claim construction arguments presented for Ground 3 for limitations 1.4, 10.4, and 18.5. As discussed above in Section VI.C.1, *supra*, these limitations are satisfied by Sweat and the combination of Sweat and Schneider therefore equally satisfies those limitations.

I, David Maier, do hereby declare and state, that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: October 24, 2024

A handwritten signature in cursive script that reads "David Maier". The signature is written in dark ink and is positioned above a horizontal line.

David Maier