

Dropbox, Inc. and Microsoft Corp.

v.

Datanet, LLC

IPR2024-00080

Patent 10,585,850

Demonstratives of Petitioners Dropbox and Microsoft

February 13, 2025

Summary of Grounds

	Reference(s)	35 U.S.C. §	Claims
Ground 1	Schneider (EX1005)	103	1-21
Ground 2	Schneider and Crawford (EX1009)	103	1-21
Ground 3	Sweat (EX1006)	103	1-8, 10-16, 18-21
Ground 4	Sweat and Schneider	103	9, 17

Claim
Construction
Dispute

Petition (Paper 1), 2

Grounds 1-2

Grounds 1-2

Ground 1	Schneider (EX1005)	103	1-21
Ground 2	Schneider and Crawford (EX1009)	103	1-21

Petition (Paper 1), 2; *see also id.*, 14-49

Disputed elements:

- Element 1.4 – “responsive to a selection ... retrieve the restorable representation ... from the remote storage location”
POR (Paper 16), 17-31
- Element 19.3 – “previewing selected revision of said file ... received from an Internet storage area network”
POR (Paper 16), 31-32
- Claim 3 – “the application is in a read-only mode”

POR (Paper 16), 32-34

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Element 1.4

[1.4] “(C) 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 selected previous version available from the restorable representation of the selected previous version”

[1.2.iii] “the **restorable representation** having at least information required for recovering the version V by a computational machine with the current version being accessible”

EX1001, 9:21-28 & 9:5-8;

EX1022 (claims listing) (emphasis added)

Element 1.4

POR misreads the “retrieving” step:

Limitation 1.4 recites, *inter alia*, “**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 as the current version...**” In other words, **retrieval of the previous version of a file** from the remote storage location, and its storage as the current version, are both made in response to, and thus chronologically after, the selection to restore such previous version. The Petition has not established that any

POR (Paper 16), 17-18

POR on Grounds 1-2, Element 1.4

POR raises three arguments as to Element 1.4:

1. no “retrieving the restorable representation ... *from the remote storage location*”
POR (Paper 16), 18-22
2. no retrieving and storing “*responsive to a selection to restore*”
POR (Paper 16), 22-25
3. “reconstruction of the disk (or file) is performed locally, on the user’s computer,” in Schneider
POR (Paper 16), 26;
see generally id. at 25-31

Sur-Reply on Grounds 1-2, Element 1.4

In Sur-Reply, PO focuses on only its third argument:

“reconstruction of these previous file versions is performed locally,” in Schneider

Sur-Reply (Paper 21), 9;
see generally id. at 8-10

“this ‘simulated disk’ (or temporary folder), which is used to retrieve and store these reconstructed prior versions of files ... is local to the user’s computer”

Id. at 10

POR Argument #1 (pages 18-22)

file version in order to transfer the file to the user's computer.” EX1004 ¶ 71. No analysis or explanation is provided as to why the local history buffer recited in the relied-upon portions of Schneider can be understood to be from a remote storage location, and no attempt is made to explain why it would otherwise be obvious.

The Petition has thus failed to identify, with particularity, evidence to support the conclusion that limitations 1.4, 10.4, and 18.5 are obvious in view of Schneider.

POR (Paper 16), 22 (highlighting added)

Institution Decision (Element 1.4)

Patent Owner, referring to its earlier arguments regarding limitation 1.2.ii, contends Schneider does not teach or suggest “retrieving the restorable representation of the selected previous version from the remote storage location” because Schneider recovers files from local storage rather than from its external disk. Prelim. Resp. 35–36. However, as discussed above in § II.D.2.c., Petitioner has adequately shown that Schneider discloses or renders obvious storing data for individual file recovery on an external disk.

DI (Paper 8), 21 (highlighting added)
(cross-referencing DI, 11-17)

The Schneider implementation presented by the Petition

55. Based on these and other teachings 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. EX1005, 10:25-26, 24:1-2. 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). Schneider describes, and a POSITA would

EX1004, ¶ 55 (highlighting added) (cited by Petition (Paper 1), 18)

Schneider: storing historic states on external disk

10 To be more specific, each PC has an internal disk for which a portion represents OS visible data and the rest generally is historic (original states of overwritten OS visible data). The OS visible portion is typically bounded by the size of the PC's internal disk minus a minimum that is set aside for historic data (which could be zero). The server needs, for each PC, to have at least sufficient space for the OS visible portion of the PC's internal disk. The amount of additional disk
15 allocated on the server to a given PC is used to hold historic data. If one views the external disk as simply a second copy of the PCs internal disk which lags behind in being updated, the two disks should be the same size. However, there is no reason the external disk cannot have more or less additional storage used for historic states as compared to that reserved on the internal disk. This implies the external disk may
20 be able to reach further back in time in re-creating prior states, if it has more historic information, or not as far back if it has less.

internal disk

external disk

EX1005, 113:9-21 (highlighting added)
(cited by Petition (Paper 1), 16-17);
see also Petition, 22-23

The POR elsewhere articulated the Petition's theory

¶ 62. The Petition then concludes that “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.” Pet. 18.

POR (Paper 16), 26 (highlighting added);
see generally POR, 25-26 (citing Petition (Paper 1), 16-18);
see also Reply, (Paper 19), 12

The Reply

No dispute it was obvious to implement Schneider with external drive for file back-up:

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.

Reply (Paper 19), 12-13 (highlighting added) (citing POR (Paper 16), 26)

POR Argument #2 (pages 22-25)

Finally, EX1005-130:11-13 states that “[a] selection can be made from the list and the recovered file either replaces the current version or is copied to a new file.” This does not teach, however, that *in response to* such selection, to both (a) retrieve the restorable representation of the selected previous version from the remote storage location and (b) store the selected previous version as the current version on the local storage location, as recited in limitation 1.4.

POR (Paper 16), 25-26 (highlighting added)

Schneider: the “recovered file ... replaces the current version”

10 4. Hooking into or supplementing the directory and file presentation
means of an OS, and allowing the user to view a list of earlier versions of a file. A
selection can be made from the list and the recovered file either replaces the current
version or is copied to a new file. The list is generated from the OS's file activity
that is logged by the engine. For a given file, the engine constructs a list of a file's
15 available earlier versions by scanning its log, and following the path, for the selected
file, of its file modifications, file renames, and file moves (from one directory to
another).

EX1005, 130:10-17 (highlighting added)
(cited by Petition (Paper 1), 18; EX1004, ¶ 55)

Institution Decision

location in which an existing file may be overwritten.” *Id.* Schneider, however, expressly states that its “recovered file either *replaces the current version* or is copied to a new file.” Ex. 1005, 130:12–13 (emphasis added). The recovered version becomes the current version it replaces. As such, at

DI (Paper 8), 21 (highlighting added)

Schneider: file retrieval

15

stored in the general log (described in the prior section). 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. Thus, the user may come to access historic information based on a

EX1005, 59:12-17 (cited by Reply (Paper 19), 12)

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

EX1024, ¶ 23 (highlighting added)
(cited by Reply (Paper 19), 11-12)

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

...

activating) 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

EX1024, ¶ 23 (highlighting added) (quoting EX1005, 59:12-17 & 130:10-13) (cited by Reply (Paper 19), 11-12)

Sur-Reply

With this understanding, the Reply further admits that “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.” Reply 15 (*citing* EX1005, 59:8-17). According to the Reply, “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.” Reply 16

Sur-Reply (Paper 21), 9
(highlighting added) (quoting Reply
(Paper 19), 15-16 (bold added by
Sur-Reply))

POR Argument #3 (pages 25-31)

- POR focuses on where Schneider reconstruction occurs
- POR notes that reconstruction uses historic data retrieved from the history buffer

history buffer.)” EX1005-38:30-40:3. However, regardless of where the image and history buffer may be located, reconstruction of the disk (or file) is performed locally, on the user’s computer. EX2004 ¶¶ 88-95; EX1004 ¶ 52 (“history buffer allows a user to recover data of a Drive C by looking back to a prior reference time by *creating a simulated Drive D* that combines unchanged data from the current Drive C with archived data from the history buffer.”) (emphasis added); EX1005-

POR (Paper 16), 26 (highlighting added)

POR: underlying data for “prior history versions of files” “may be obtained remotely” in Schneider

However, as shown above in connection with limitation 1.4, all prior history versions of files in Schneider are generated *locally*, even if the underlying data used to generate the versions may be obtained remotely, and thus Schneider does not teach “previewing selected revision of said file, wherein said selected revision of said file is received from an Internet storage area network,” as recited in limitation 19.3. EX2004 ¶ 100.

POR (Paper 16), 32 (highlighting added)

Sur-Reply

The Reply appears willing to concede that reconstruction of these previous file versions is performed locally. *See* Reply 14 (“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.”).

Sur-Reply (Paper 21), 9 (highlighting added)

Schneider: *view* contents of old file versions, then discard

25

Note that it is useful to allow 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. (The contents of this temporary directory, when no longer in use, are eventually automatically purged.) 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, as the viewed file is automatically discarded. Therefore the viewed file's existence in terms of being retrieved is hidden from the user in that the user does not have to manage the viewed file on disk.

30

EX1005, 60:21-29 (highlighting added)
(cited by Petition (Paper 1), 18; EX1004, ¶ 55)

Schneider: create simulated disk in response to selection to retrieve

stored in the general log (described in the prior section). 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. Thus, the user may come to access historic information based on a selection other than directly choosing a time.

EX1005, 59:12-17 (highlighting added)
(cited by Reply (Paper 19), 12)

Dr. Maier: viewing in Schneider, followed by restore

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

EX1024, ¶28 (highlighting added)
(cited by Reply (Paper 19), 16)

Dr. Maier: simulated disk retrieves from the remote drive history buffer

“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.

...

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

EX1024, ¶22 (highlighting added)
(cited by Reply (Paper 19), 10)

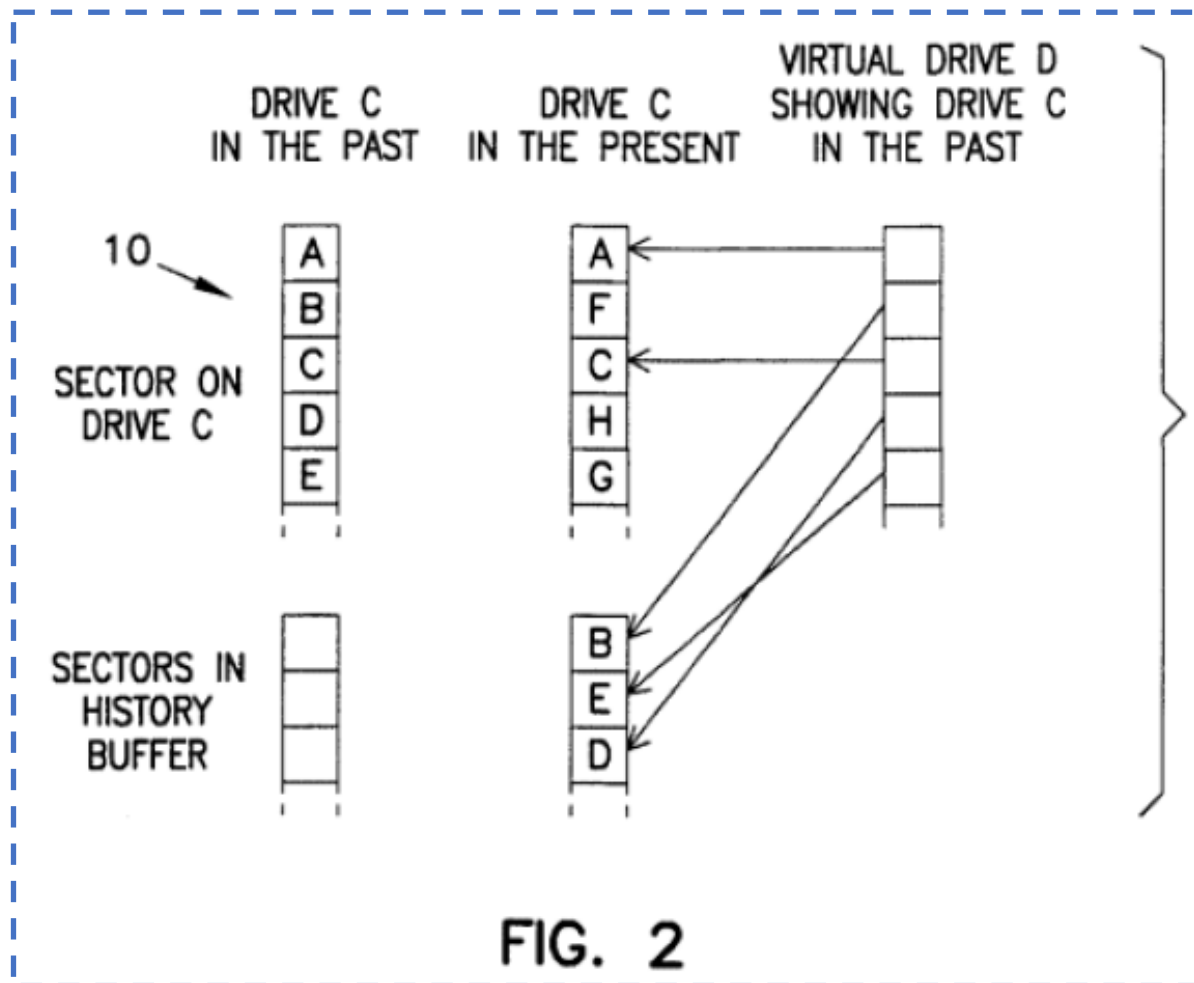
Schneider “simulated or virtual drive”

20 The Information Recovery Process

As illustrated in Figure 2, the present invention provides for two basic forms of recovery. In the first case, assuming the user is running on a drive C, they may desire to look back in time. The technology of the present invention assumes the system can support another drive D, which instead of being a real disk drive, is a simulated or virtual drive whose image is created by combining information on the original drive C and the history buffer (which is typically part of drive C). The process of looking back in time involves setting a reference time for drive D and then simply accessing it as if it really was another physical disk drive whose contents had been copied from drive C at the specified time.

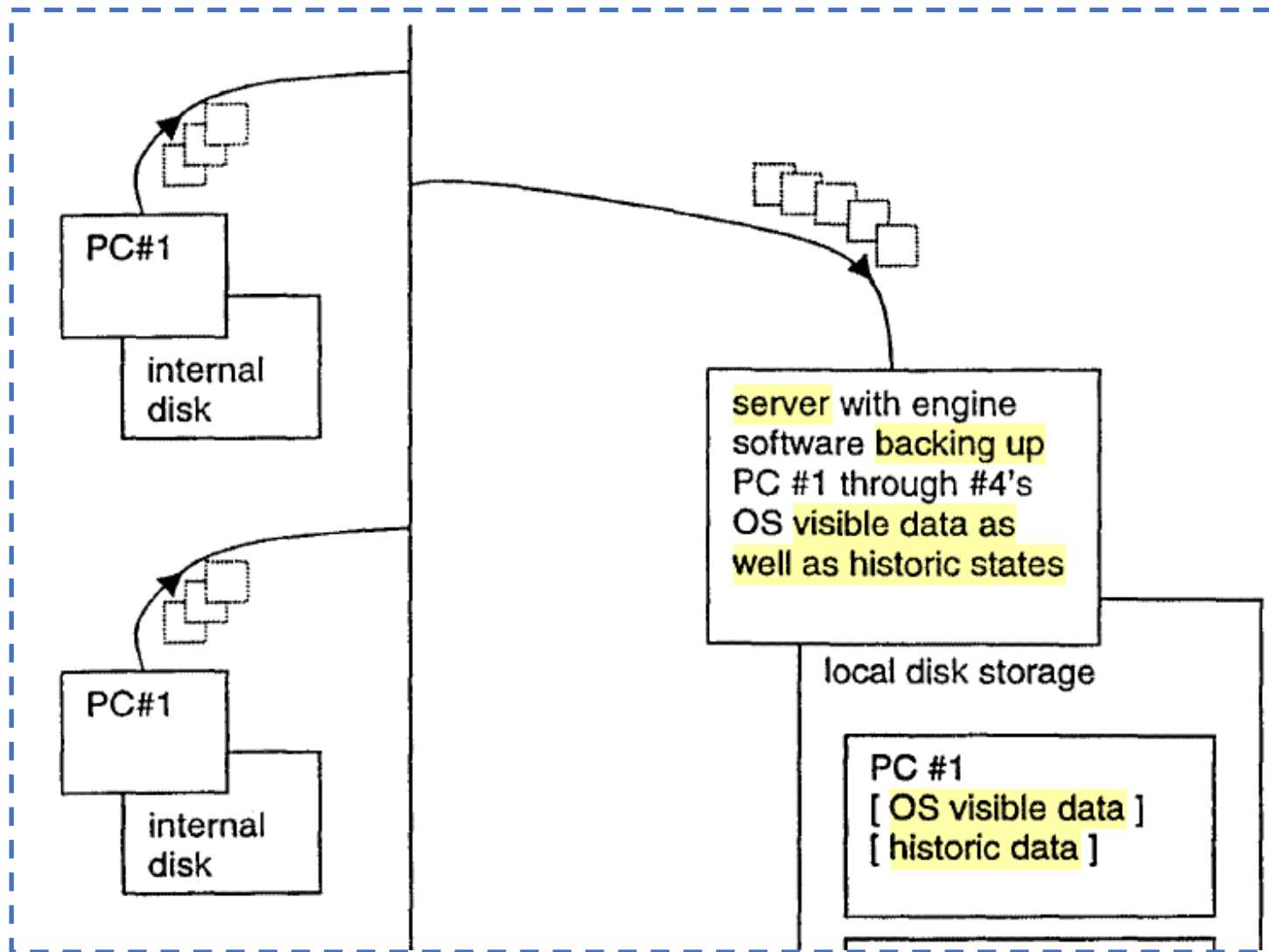
EX1005, 15:20-29 (cited by Petition (Paper 1), 30; *see also* POR (Paper 16), 29)

Schneider virtual drive



EX1005, Fig. 2 (cited by Petition (Paper 1), 30; EX1004, ¶¶ 52, 71)

Schneider: storing historic states on external disk



EX1005, FIG. 47G (cropped from bottom for readability, highlighting added)
(cited by Petition (Paper 1), 23-24;
EX1004, ¶ 64)

Claim 3

[3] “The method of claim 2, wherein the application is in a read-only mode.”

EX1001, 9:33-34; *see also* EX1022 (claims listing)

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-

Reply (Paper 19), 19

Claim Construction

Claim 1

1. A method of restoring a file to a previous version of the file, a current version of the file being available at a local storage location, comprising the steps of:
 - (A) presenting information for a collection of one or more previous versions of the file, the information for the collection including information indicative of at least one or more of previous versions of the file, wherein a restorable representation of each version, V, of the previous versions, is retrievable from a remote storage location, the restorable representation having at least information required for recovering the version V by a computational machine with the current version being accessible, the remote storage location being accessible through a network;
 - (B) responsive to a selection to preview a selected previous version of the file based on the presented information for the collection of (A), presenting a presentable representation of the selected previous version, the selected previous version being one of the previous versions of the file in the presented information for the collection, the presentable representation having at least information required for presenting at least a portion of the selected previous version by the computation machine with the current version being accessible; and
 - (C) 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 selected previous version available from the restorable representation of the selected previous version.

EX1001, 8:63-9:28
(highlighting added)

Element 1.4

(C) 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 selected previous version available from the restorable representation of the selected previous version.

EX1001, 9:21-28
(highlighting disputed language)

Claim Construction

Disputed term in Element 1.4

storing the selected previous version by the computational machine as the current version on the local storage location

PO's proposed construction?

storing the selected previous version by the computational machine as the current version on the local storage location **and automatically reflecting back the now-current local version in the remote storage location so that parity exists between the current version on the local storage location and the current version on the remote storage location**

See POR (Paper 16), 12-14 & 36; Sur-Reply (Paper 21), 2-6 & 13;
see also POR, 3.

20 Q. So are you opining that some
21 particular action leads to this remote
22 location having the Version 1 as the current
23 version?

24 A. Yes.

EX1027, 100:20-24 (highlighting added) (cited by Sur-Reply (Paper 21), 5)

Claim Construction

Disputed claim term in Element 19.5:

restoring selected another revision of said file following said previewing step

PO's proposed construction:

[T]he “selected another revision of said file” automatically becomes the most recent version in time on both the local machine and the “Internet storage area network”

See POR (Paper 16), 14-15;
Sur-Reply (Paper 21), 6-7

Grounds 3-4

Grounds 3-4

Ground 3: claims 1-8, 10-16, 18-21 obvious over Sweat (EX1006)

Ground 4: claims 9 and 17 obvious over Sweat and Schneider

Petition (Paper 1), 2 & 50-73

Elements in dispute:

- Elements 1.4 & 19.5 – turns entirely on claim construction issue POR (Paper 16), 36-38 & 38-39
- Claims 2, 3, 11 – “presented using an application that created the file or the selected previous version”
POR (Paper 16), 39-40

Grounds 3-4: independent claims obvious under the proper construction

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.

Reply (Paper 19), 22 (highlighting added)

Grounds 3-4: claim 2

Claim 2

The method of claim 1, wherein, for the presenting of (B), the presentable representation is presented using an application that created the file or the selected previous version.

EX1001, 9:29-32

Grounds 3-4: claim 2

Sweat teaches a viewer to view files

To view project files including drawing files and markup files, a user must install the appropriate viewer to open and view drawing files. Client computer 104 may use the same plug-ins and ActiveX controls that a browser 108 uses to display files. If a user can't see the drawing files (DWG,

EX1006, 23:38-41 (highlighting added) (quoted by POR (Paper 16), 40)
(cited by Petition (Paper 1), 62)

the world, but the drawing file formats that the user is able of viewing depends on the viewer that is installed on the computer. To find out which viewer is installed, in the View tab, right-click, and then choose About.

EX1006, 23:47-51 (highlighting added) (quoted by POR (Paper 16), 39-40)
(cited by Reply (Paper 19), 23-24)

Grounds 3-4: claim 2

Sweat identifies AutoCAD® as a “drawing/viewing software utilized by the viewer”

available depending on the viewer and software utilized. 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.). Markups or markup files may be stored

EX1006, 23:29-33 (highlighting added) (cited by Reply (Paper 19), 24)

Sweat describes using its software to “create, view, or edit” files

The markups tab 510 is used to create, view, or edit markup (DWF) files. A user can save markup files to a specified location either on the server 106 or local computer

EX1006, 11:13-14 (highlighting added) (cited by Reply (Paper 19), 25)

Dr. Maier

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

• • •

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

EX1024, ¶ 39 (highlighting added)
(cited by Reply (Paper 19), 25)

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.

EX1024, ¶ 39 (highlighting added) (cited by Reply (Paper 19), 25)

Appendix

State of the Art: no meaningful distinction between local / remote file backup

B. From The Perspective Of The Backup/Restore Application, There Was No Distinction Between Local and Remote File Storage

34. It was also well known to access storage devices, including those used for backup, over a network, allowing such storage devices to be located remotely and accessed by a local computer. By the 1990s, Network File Systems (such as NFS, first released by Sun Microsystems in 1984) erased the difference between how users accessed a local file and a remote file. EX1012, (showing first release);

...

location of the file, a property known as “location transparency.” Therefore, by 1999, with respect to backup applications, there was no significant technical barrier between storing a file or its backup version locally and storing it remotely.

EX1004, ¶ 34 (highlighting added);
see also id., ¶ 35
(cited by Petition (Paper 1), 5-6
(cross-referenced by Petition, 24-25))

Known benefits of backing up to a remote external disk

66. Given that it was a known alternative to LAN-accessible external disks, it would have been an obvious design choice to use a remote Internet-accessible disk when implementing Schneider's networked external disk. Indeed, the POSITA would have been motivated to do so, e.g., to reduce the chances of the backup information being destroyed. EX1013, 556 (“[T]he chances are pretty good that the

. . .

or at home.”). Moreover, an Internet-accessible drive would conserve local storage space and power and allow computers in different remote locations to access a single external storage disk. A POSITA would easily implement this known alternative, as

EX1004, ¶ 66 (highlighting added) (cited by Petition (Paper 1), 26); *see also* DI (Paper 8), 17 (“On this record, the external storage shortcomings do not nullify the benefits motivating external storage...[and] Petitioner has shown a reasonable likelihood that Schneider teaches or suggests [Element 1.2.ii].”))

Schneider: “invention” implemented at either file or sector level

The present invention can be implemented either at the sector (disk) level or as part of an operating system’s filing system at the file level. The concept of

EX1005, 24:1-2 (highlighting added) (cited by Petition (Paper 1), 16; DI (Paper 8), 14)

25 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). There are advantages

EX1005, 10:25-26 (highlighting added) (cited by Petition (Paper 1), 18; DI (Paper 8), 14)

Institution Decision (Element 1.2.ii)

file recovery embodiments. *See id.* at 108:15–114:7. Schneider does, however, state its “invention can be implemented to work either with managing data at the disk sector level or file level (or portions of a file)” (*id.* at 10:25–26) and it “extend[s] the backup services to utilize a second hard disk, and so provide[s] a degree of hardware redundancy” (*id.* at 34:4–5). Those statements, in conjunction with Schneider’s description that its external disk is a mirror of its main internal disk, at least suggest that Schneider discloses that its file recovery embodiments would recover files using data stored on an external disk. As such, Petitioner has shown a reasonable likelihood that this limitation is disclosed by Schneider.

DI (Paper 8), 14 (highlighting added)

Institution Decision (Element 1.2.ii)

local storage failure. Prelim Resp. 30. If local storage fails, it follows that remotely stored information can be accessed more easily than locally stored information and, so, would be beneficial to provide. And, even if it were the case that that local storage is a “better solution[]” than remote storage in some scenarios, “[i]t’s not necessary to show that a combination is ‘the best option, only that it be a suitable option.’” *Intel Corp. v. Qualcomm Inc.*, 21 F.4th 784, 800 (Fed. Cir. 2021) (citing *PAR Pharm., Inc. v. TWI Pharms., Inc.*, 773 F.3d 1186, 1197–98 (Fed. Cir. 2014)) (emphasis omitted). Remote storage is a suitable option for information recovery because Schneider itself describes remote storage used for information recovery.

DI (Paper 8), 16 (highlighting added)

Institution Decision (Element 1.2.ii)

we weigh those considerations against the benefits of external storage. Petitioner identifies, e.g., hardware redundancy, a longer history of information to restore from, conserving local storage space and power, and allowing multiple computers access to shared backup data (Pet. 22–26; see Ex. 1004 ¶ 62 n.4). On this record, the external storage shortcomings do not nullify the benefits motivating external storage. *Medichem S.A. v. Rolabo S.L.*, 437 F.3d 1157, 1165 (Fed. Cir. 2006) (“a given course of action often has simultaneous advantages and disadvantages, and this does not necessarily obviate motivation to combine”) (citing *Winner*, 202 F.3d at 1349 n.8). As such, Petitioner has shown a reasonable likelihood that *Schneider* teaches or suggests this limitation.

DI (Paper 8), 17
(highlighting added)

Schneider: history buffer to recover previous file versions

specialized access to the history buffer would be able to locate and recover versions only of desired files. . . . [F]rom the user's point of view, he or she could select a file and request the present invention's software to locate all available versions of that file, over time, that are still in the history buffer.

Ex. 1005, 31:19–24.

EX1005, 31:19-24 (highlighting added)
(as quoted by DI (Paper 8), 14) (cited by Petition (Paper 1), 19)

Schneider: file-based approach saves “original states of recent changes”

10 In a file level implementation of the present invention, with tight integration into the operating system, the method and understanding by the user of retrieving information may significantly differ, although is still based on the principal of saving original states of recent changes. For example, an operating system may

EX1005, 24:7-10 (highlighting added) (cited by Petition (Paper 1), 21)

external disk drive that, like the main disk, employs a method of saving original disk states. Thus, instead of creating a backup of a specific point in time, the information

EX1005, 109:7-8 (highlighting added) (cited by Petition (Paper 1), 21)

Schneider: saves either entire file state or portions

implementation could save an entire prior file state just before a file is opened and the intention to modify is clear, or it could attempt to only save prior states of the portion of the file being modified. That later starts to resemble the sector based method but differs in where in the computer system technology of the present

EX1005, 24:22-24 (highlighting added) (cited by Petition (Paper 1), 23)

Schneider teaches backing up entire files

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

Schneider:

"[T]he present invention at the file level only seeks to **allow access to the saved backup file copies and not simulate an entirely reverted disk.**"

EX1005, 25:4-6 (emphasis added)
(cited by Reply (Paper 19), 15)

EX1024, ¶27 (highlighting added)
(cited by Reply (Paper 19), 15)

Schneider: “take advantage” of more historic data on external disk

maintaining historic data on their respective PC's. Provisions in the PC's software would divert to and take advantage of an external disk that had more historic information than available on the internal disk, and whose access is desired.

EX1005, 113:26-28 (highlighting added) (cited by Petition (Paper 1), 17)

Schneider: view list of prior versions of a file

5 The present invention provides an extension to Explorer wherein the user can right click on a specific file and have the option to view a list of old versions of the file. This list is constructed by scanning the general log. However, the approach

EX1005, 60:3-5 (highlighting added)
(cited by Petition (Paper 1), 18; EX1004, ¶ 55)

221 Provisional

Describes features as “optional”:

AccuSafe provides the user with true background and real time backup. No scheduling on the part of the user is required. No file selection by the user is required. The features offered may be individually selected or not selected and controlled or not controlled by the user at their choosing.

EX1003, 139 (highlighting added) (cited by Reply (Paper 19), 3)

AccuSafe users must have the ability to control which files are included and excluded within the AccuSafe backup process. A control hierarchy will be employed to eliminate

EX1003, 156 (highlighting added) (cited by Reply (Paper 19), 3)

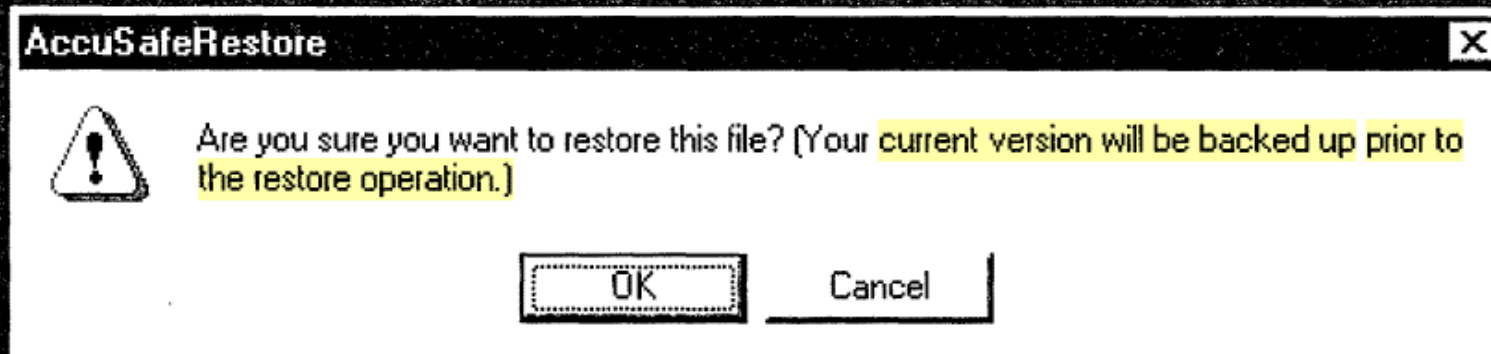
Note that other file-specific AccuSafe Behavior controls will be present on the “AccuSafe Behavior” tab. These include, but are not limited to file encryption on/off, file encryption password, file compression on/off/type options, backup type (automatic, schedule, modified-automatic, etc.), etc.

EX1003, 161 (highlighting added) (cited by Reply (Paper 19), 3-4)

221 Provisional

Describes backup *before* restoring:

Example use-case scenario



AccuSafe will save the existing file (if it needs to!) before overwriting it with the copy the user is about to restore.

EX1003, 37 (highlighting added)(cited by Reply (Paper 19), 6); *see also* EX1024, ¶ 19 (cited by Reply, 6)

221 Provisional

Describes backup *before* restoring:

prior to actually restoring the file, the restoration process should determine if the backed up file will overwrite a file of the same name on the users PC, and if so, if the version of the file on the users PC is on the top level of the AccuSafe revision “stack”. If such is not the case, AccuSafe *must* **first backup the users file to the AccuSafe Media Device**, then perform the restoration function.

This will assure that no version of a file is lost as a result of the restoration process.

Sur-Reply (Paper 21), 4-5 (highlighting added)
(quoting EX1003, 147) (bold and italics added by Sur-Reply)

221 Provisional

The user may select an item (file or directory) from the list presented in this user interface, then click the Restore button in order to restore the file from the AccuSafe Media Device. Depending on the setting of an appropriate user interface widget, the file will be restored to it's original location, or to an alternate location chosen by the user.

EX1003, 147 (highlighting added) (cited by Reply (Paper 19), 4; Sur-Reply (Paper 21), 4)

Sur-Reply Focuses On the Wrong Language

Because the term “restore” means that a file is to be overwritten, and because the resultant overwritten file inherently becomes the “current version” of such file, the term “as the current version” in limitation 1.4 must mean more than mere overwriting of a file – to interpret the claim otherwise would be to construe the

Sur-Reply (Paper 21), 3 (highlighting added)

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

The undersigned certifies that on February 6, 2025, **PETITIONERS'**
DEMONSTRATIVES were served on Patent Owner via electronic mail as
follows:

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

Ronald Abramson
M. Michael Lewis
Gina K. Kim
Liston Abramson LLP
405 Lexington Avenue, 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 Petitioners