

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

DROPBOX, INC. AND MICROSOFT CORPORATION,
Petitioners,

v.

DATANET LLC,
Patent Owner.

IPR2024-00080
U.S. Patent No.: 10,585,850
Issued: March 10, 2020
Application No.: 14/970,044
Filed: December 15, 2015

Title: Automatic Real-Time File Management Method And Apparatus

PETITION FOR
INTER PARTES REVIEW OF U.S. PATENT NO. 10,585,850

TABLE OF CONTENTS

	Page(s)
LIST OF EXHIBITS.....	vi
I. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8.....	viii
A. Real Party-In-Interest	viii
B. Related Matters.....	viii
C. Lead And Back-Up Counsel, And Service Information	viii
I. INTRODUCTION	1
II. GROUNDS FOR STANDING PER 37 C.F.R. § 42.104(a)	1
III. IDENTIFICATION OF CHALLENGE	2
A. Statement Of The Precise Relief Requested / Statutory Grounds.....	2
B. No Examiner Addressed These Unpatentability Grounds	3
C. The Board Should Institute Despite Parallel Litigation	4
IV. TECHNOLOGY OVERVIEW.....	4
A. File-Based Backup and Recovery Systems Were Known Since At Least 1978	4
B. From The Perspective Of The Backup/Restore Application, There Was No Distinction Between Local and Remote File Storage...	5
V. THE '850 PATENT	6
A. The '850 Patent's Specification	7
B. The Prosecution History	11
C. The Challenged Claims	12
VI. PERSON OF ORDINARY SKILL IN THE ART	13
VII. CLAIM CONSTRUCTION	14

VIII. GROUND 1: CLAIMS 1-21 ARE OBVIOUS OVER SCHNEIDER	14
A. Schneider.....	15
1. Claims 1 & 10	19
a) [1.1]/[10.1]	19
b) [1.2.i]/[10.2.i].....	20
c) [1.2.ii]/[10.2.ii]	21
d) [1.2.iii]/[10.2.iii]	26
e) [1.3]/[10.3]	27
f) [1.4]/[10.4].....	29
2. Claims 2 & 11	30
3. Claim 3	31
4. Claims 4 & 12	33
5. Claims 5 & 13	33
6. Claims 6 & 14	34
a) [6.1]/[14.1].....	34
b) [6.2]/[14.2].....	36
7. Claims 7 & 15	37
8. Claims 8 & 16	37
9. Claims 9 & 17	38
10. Claim 18	39
a) [18.1].....	39
b) [18.2].....	40
c) [18.3.i]-[18.3.ii], [18.4]-[18.5].....	41

d)	[18.3.iii]	41
11.	Claim 19.....	42
a)	[19.1].....	42
b)	[19.2].....	42
c)	[19.3].....	42
d)	[19.4].....	43
e)	[19.5].....	44
12.	Claims 20 & 21	45
IX.	GROUND 2: CLAIMS 1-21 ARE OBVIOUS OVER SCHNEIDER IN VIEW OF CRAWFORD	45
X.	GROUND 3: CLAIMS 1-8, 10-16, AND 18-21 ARE OBVIOUS OVER SWEAT.....	50
A.	Sweat	50
1.	Claims 1 & 10	55
a)	[1.1]/[10.1].....	55
b)	[1.2.i]/[10.2.i].....	56
c)	[1.2.ii]/[10.2.ii]	57
d)	[1.2.iii]/[10.2.iii]	59
e)	[1.3]/[10.3]	59
f)	[1.4]/[10.4]	61
2.	Claims 2 & 11	62
3.	Claim 3	63
4.	Claims 4 & 12	65
5.	Claims 5 & 13	65

6.	Claims 6 & 14	66
	a) [6.1]/[14.1]	66
	b) [6.2]/[14.2]	66
7.	Claims 7 & 15	67
8.	Claims 8 & 16	67
9.	Claim 18	68
	a) [18.1]	68
	b) [18.2]	68
	a) [18.3.i], [18.3.ii], [18.4], [18.5]	69
	b) [18.3.iii]	69
10.	Claim 19	69
	a) [19.1]	69
	b) [19.2]	69
	c) [19.3]	69
	d) [19.4]	70
	e) [19.5]	70
11.	Claims 20 & 21	71
XI.	GROUND 4: CLAIMS 9 AND 17 ARE OBVIOUS OVER SWEAT IN VIEW OF SCHNEIDER	72
XII.	NO OBJECTIVE INDICIA OF NON-OBVIOUSNESS	73
XIII.	CONCLUSION	73
	CERTIFICATE OF COMPLIANCE	74
	CERTIFICATE OF SERVICE	75

TABLE OF AUTHORITIES

Page(s)

Cases

In re Johnston,
435 F.3d 1381 (Fed. Cir. 2006) 44

Perfect Web Techs., Inc. v. InfoUSA, Inc.,
587 F.3d 1324 (Fed. Cir. 2009) 35

Phillips v. AWH Corp.,
415 F.3d 1303 (Fed. Cir. 2005) 14

Statutes

35 U.S.C. § 103 2

Other Authorities

USPTO Memorandum, Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation (June 21, 2022).....4

Rules

37 C.F.R. § 42.15 2

37 C.F.R. § 42.24 74

37 C.F.R. § 42.6 75

LIST OF 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

No.	Description
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. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8

A. Real Party-In-Interest

The real-parties-in-interest are Dropbox, Inc. and Microsoft Corporation.

B. Related Matters

The '850 patent (EX1001) has been asserted in the following litigations:

- *Datanet LLC v. Dropbox, Inc.*, No. 6:22-cv-01142 (W.D. Tex.), filed October 31, 2022; and
- *Datanet LLC v. Microsoft Corporation*, No. 2:22-cv-01545 (W.D. Wash.), filed October 31, 2022.

A patent that alleges priority to the same original non-provisional and provisional applications to which the '850 patent alleges priority is being challenged in *ex parte* Reexamination No. 90/015,216.

C. Lead And Back-Up Counsel, And Service Information

Lead Counsel	Back-up Counsel
Andrew M. Mason, Reg. No. 64,034 andrew.mason@klarquist.com	Cameron D. Clawson, Reg. No. 73,509 cameron.clawson@klarquist.com Sarah E. Jelsema, Reg. No. No. 70,804 sarah.jelsema@klarquist.com Frank Morton-Park, Reg. No. 80,750 frank.morton-park@klarquist.com
KLARQUIST SPARKMAN, LLP 121 SW Salmon Street, Suite 1600 Portland, Oregon, 97204	

503-595-5300 (phone) 503-595-5301 (fax)
--

Petitioners consent to service via email at the above email addresses.

Pursuant to 37 C.F.R. § 42.10 (b), concurrently filed with this Petition is a Power of Attorney executed by Petitioners and appointing the above counsel.

I. INTRODUCTION

Dropbox, Inc. and Microsoft Corporation (“Petitioners”) respectfully request *inter partes* review (“IPR”) of claims 1-21 of U.S. Patent No. 10,585,850 (“’850 patent”) (EX1001), allegedly assigned to Datanet LLC (“Patent Owner”). For the reasons set forth below, these claims should be found unpatentable and cancelled.

The ’850 patent claims methods of restoring a file on a computer to a previous version of the file. During prosecution, the alleged point of distinction over the prior art related to the presentation of a “preview” of the previous version without necessarily restoring it as the current local version of the file. Each primary reference presented in this petition discloses a backup and recovery system that allows users to preview a previous version of a file before restoring that previous version to the user’s local computer. Those references, Schneider (EX1005) and Sweat (EX1006), were not considered during prosecution and, individually and together with other references, render each challenged claim unpatentable.

II. GROUNDS FOR STANDING PER 37 C.F.R. § 42.104(a)

Petitioners certify that the ’850 patent is available for IPR and that Petitioners are not barred or estopped from requesting an IPR challenging the patent claims on the grounds identified in this petition.

III. IDENTIFICATION OF CHALLENGE

A. Statement Of The Precise Relief Requested / Statutory Grounds

Petitioners request *inter partes* review of claims 1-21 (the “Challenged Claims”) of the ’850 patent, on the following statutory 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

For each ground, in Sections VIII-XI below, the petition presents evidence of unpatentability and establishes a reasonable likelihood that Petitioners will prevail in establishing that each Challenged Claim is unpatentable.

With the filing of this petition an electronic payment of \$42,625 for the requisite fees is being charged to deposit account no. 02-4550. 37 C.F.R. § 42.15(a). Any fee adjustments may be debited/credited to the deposit account.

B. No Examiner Addressed These Unpatentability Grounds

At least because the primary prior art references were not previously considered by the Office, this petition should not be denied under Section 325(d).

Primary references Schneider (EX1005) and Sweat (EX1006) were neither cited nor discussed during prosecution, and they are not referenced in the '850 patent. Schneider and Sweat provide express teachings on saving, storing, previewing and recovering archived data files, including for multiple versions of such files. These teachings, which show that the challenged claims are unpatentable as obvious, were not considered by the Examiner.

During prosecution, the Examiner cited and discussed a continuation of secondary reference Crawford. EX1002, 142-44¹. But the Examiner did not consider the combination of Schneider with Crawford presented herein as Ground 2. Secondary reference Crawford expressly teaches backup file storage using an Internet storage area network, which Patent Owner did not dispute during prosecution.

Accordingly, this petition should not be denied under Section 325(d).

¹ Unless otherwise noted, file history cites are to the page numbers added by Petitioner.

C. The Board Should Institute Despite Parallel Litigation

Each Petitioner stipulates that, if the Board institutes IPR, it will not pursue in its parallel district court proceeding the grounds raised, or which reasonably could have been raised, in this petition. When a petitioner presents such a stipulation, “the PTAB will not discretionarily deny institution in view of parallel district court litigation.” USPTO Memorandum, Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation at 3, 7 (June 21, 2022).

IV. TECHNOLOGY OVERVIEW

A. File-Based Backup and Recovery Systems Were Known Since At Least 1978

Systems for providing redundant backup for computer data files, including storage of multiple versions of a given file and capability to recover prior file versions were well known in the art. For example, as early as 1978, Verhofstad wrote that “[r]ecover techniques can be used to restore data in a system to a usable state. Such techniques are widely used in filing systems and database systems in order to cope with failures.” EX1010, 167. Verhofstad describes a “Backup/current version” system for files, in which “[t]he files containing the present values of existing files form the current version of the database. Files containing previous values form a consistent, backup version of the database. Backup versions can be used to restore

files to previous values.” *Id.*, 172; *see also* EX1011, 18-19 (“Backup software can operate either on files or on physical disk blocks. **File-based backup** systems understand file structure and copy entire files and directories to backup devices. ... Then backup software writes each file contiguously to the backup medium; this facilitates fast individual file recovery.”) (emphasis original)². EX1004, ¶33.

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

It was also well known to access backup storage devices 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, 379-390 (showing first release); EX1009, 3:64-66 (“Drives on a remote system defined using NFS and RFS are available for mounting by other remote systems on the Internet.”). This technology meant that applications on the user computer did not need to know the actual storage 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 locally and storing it remotely. EX1004, ¶34.

² Emphases throughout are added, unless otherwise noted.

This lack of a meaningful distinction is reflected in the prior art. For example, the Sweat reference relied on herein disclosed a distributed network architecture connecting local client computers to remote file servers and file storage over the Internet. EX1006, 3:12-26. Such an architecture allows a user to “view or download files from anywhere in the world to work on them.” *Id.*, Abstract, 2:52-53. And Cabrera described that then-state of the art backup and archival “systems typically copy data from a local storage medium to a backup or archival storage medium, sometimes located at a remote location. The process of transferring data from a local storage medium to a backup or remote storage medium is much the same in either case.” EX1008, 1:38-42. Further, the “[a]rchive or backup storage may comprise any storage medium suitable for such a purpose. The location of such a storage medium may be local to the backup or archive system or may be remote from the backup or archive system.” *Id.*, 3:3-7. EX1004, ¶35.

V. THE '850 PATENT

The '850 patent issued March 10, 2020, from an application filed December 15, 2015 (the “'044 Application”) and alleges priority through several continuation applications to a non-provisional application filed September 21, 2001, and also claims the benefit of U.S. Provisional Application No. 60/234,221 (“'221 Provisional”), filed September 21, 2000. EX1004, ¶36.

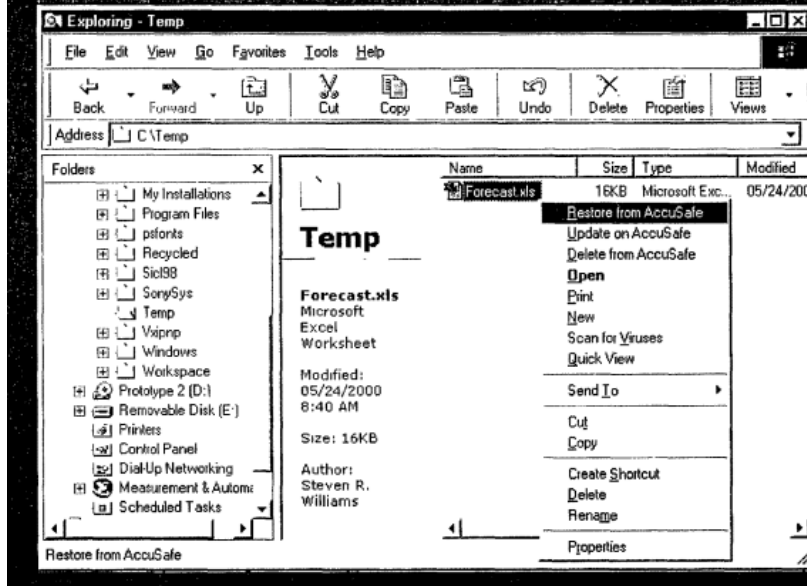
A. The '850 Patent's Specification

While the claims of the '850 patent relate to “restoring a file to a previous version” (EX1001, claims 1, 10, 18, 19), the written description of the '850 patent does not describe restoring a file to a previous version or any of the claimed “presenting” or “retrieving” steps. EX1004, ¶37.

The '850 patent purports to incorporate the '221 Provisional by reference. EX1001, 1:4-10. The '221 Provisional includes presentation slides for a prototype product called “AccuSafe™” (e.g., EX1003, 3-60)³ that supposedly allowed a user to “right-click on the file he’s been working with, then select AccuSafe’s restore function to retrieve the lost versions,” as shown in the slide below. *Id.*, 31 (excerpt). EX1004, ¶38.

³ Unless otherwise noted, '221 Provisional cites are to the page numbers added by Petitioner.

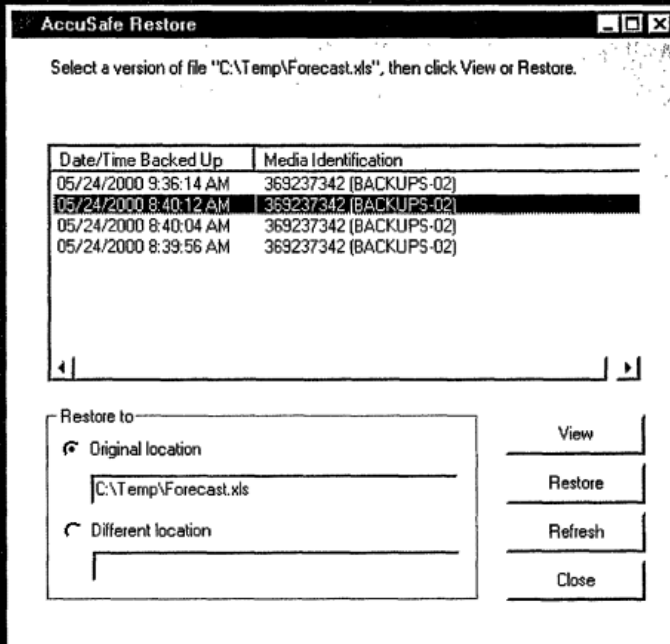
Example use-case scenario



The user realizes that he's overwritten his file. Via explorer (or any other file selection dialog), the user can right-click on the file he's been working with, then select AccuSafe's restore function to retrieve the lost versions.

As shown in the slides below, the '221 Provisional states that the AccuSafe product displays a list of captured revisions of the desired file and a user selects a previous file version to display that file using the same application that created the file. *Id.*, 32-34. The user may iteratively view multiple captured versions of the file until the user finds the correct prior version that the user wanted. *Id.* (excerpted below). EX1004, ¶39.

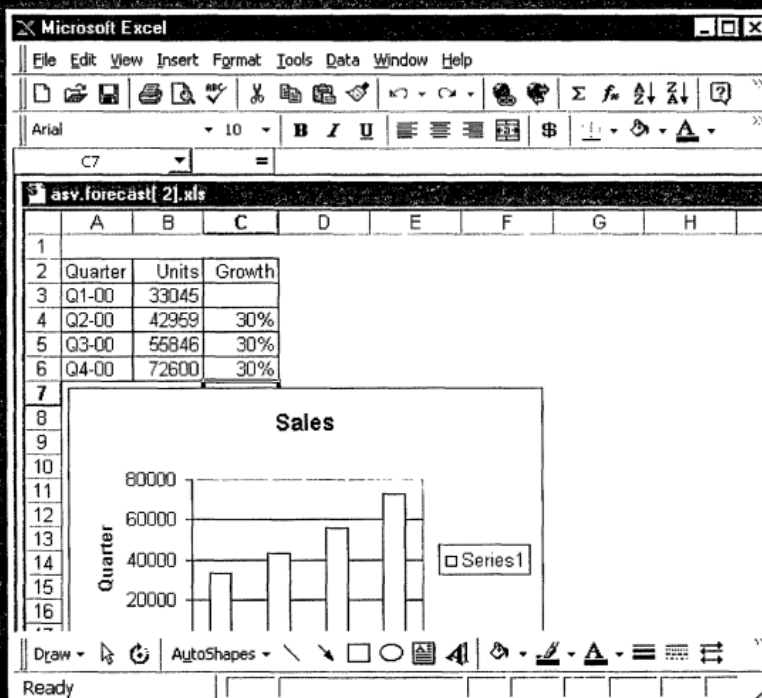
Example use-case scenario



AccuSafe displays the list of revisions it has captured.

The user selects a version from this list, then clicks the View button to determine if this is the version he wishes to restore...

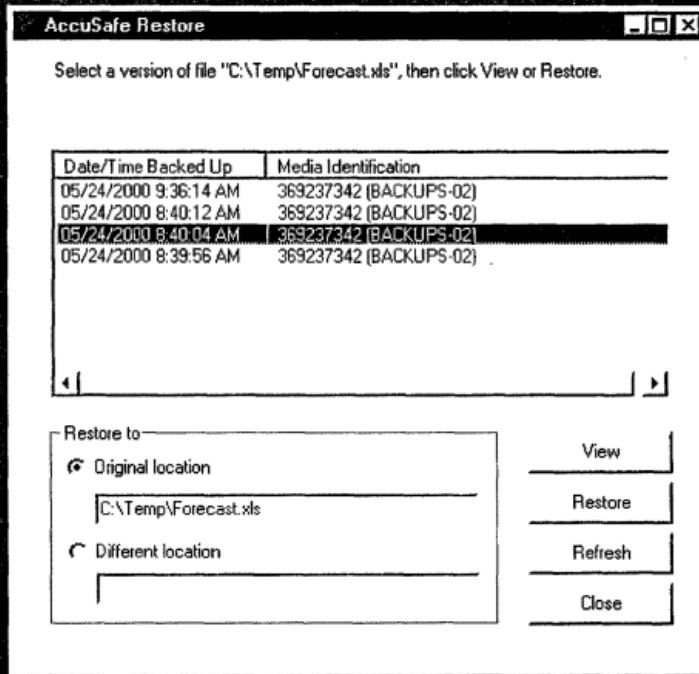
Example use-case scenario



AccuSafe displays the revision selected, using the application that created the file.

If this isn't the version that the user wanted...

Example use-case scenario



The user selects another revision to view...

Note: Most backup applications don't have such a preview capability, which significantly enhances ease-of-use.

Finally, as shown below, once a user verifies the prior file version to restore and clicks "Restore," the product overwrites the current version of the file with the prior version selected by the user, while saving a copy of the replaced current version. *Id.*, 36 (excerpt). EX1004, ¶40.

Example use-case scenario

AccuSafe Restore

Select a version of file "C:\Temp\Forecast.xls", then click View or Restore.

Date/Time Backed Up	Media Identification
05/24/2000 9:36:14 AM	369237342 (BACKUPS-02)
05/24/2000 8:40:12 AM	369237342 (BACKUPS-02)
05/24/2000 8:40:04 AM	369237342 (BACKUPS-02)
05/24/2000 8:39:56 AM	369237342 (BACKUPS-02)

Restore to:

☒ Original location

C:\Temp\Forecast.xls

☐ Different location

View

Restore

Refresh

Close

The user has verified this to be the proper revision to restore, so the user clicks Restore to retrieve his file.

B. The Prosecution History

After the December 15, 2015, application, a preliminary amendment on October 27, 2016, cancelled all claims and added new claims 52-72. EX1002, 59-66.

A first Office action issued January 31, 2019, rejecting claims 52-69 as obvious over Whiting (U.S. Patent No. 5,778,395) and rejecting 70-72 as obvious over Whiting in view of U.S. Patent No. 6,411,943 (a continuation of Crawford EX1009). *Id.*, 132-45.

In response, Applicant filed remarks traversing the examiners rejections on July 1, 2019. *Id.*, 223-29. Applicant argued that:

Whiting neither discloses the feature of “a selection to preview a selected previous version of the file” nor discloses the feature of “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.”

Id., 227.

A Notice of Allowance then issued for pending claims 52-72, which became claims 1-21 of the '850 patent. *Id.*, 230-36. The Examiner's reasons for allowance adopted Applicant's July 1, 2019, arguments regarding the disclosures of Whiting.

Id., 235.

C. The Challenged Claims

A full claims listing is provided at EX1022. The claims are also listed in EX1023, in a manner that better illustrates common language across certain claims. Exemplary claim 1 is listed here, with element⁴ numbers added in brackets:

[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:
-------	---

⁴ Petitioners use the terms “limitation” and “element” loosely throughout the Petition and, unless otherwise noted, without taking positions on whether the claim language is entitled to patentable weight or is merely part of the claim's environment.

[1.2]	[1.2.i] (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, [1.2.ii] wherein a restorable representation of each version, V, of the previous versions, is retrievable from a remote storage location, [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, the remote storage location being accessible through a network;
[1.3]	(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
[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.

VI. PERSON OF ORDINARY SKILL IN THE ART

The person of ordinary skill in the art in September 2000 (“POSITA”) would have held a bachelor’s degree in computer science, electrical engineering, computer engineering, or a closely related field, and one or more years of experience in the design and development of file management, restoration, and archival systems. More education, e.g., a Master’s or PhD could compensate for a lack of work experience and vice versa. There would be no meaningful distinction if the critical date were

1999 or 2001. *Id.* The technology overview provided above (*supra* Section IV) reflects some of the information within a POSITA's knowledge. EX1004, ¶¶29-32.

VII. CLAIM CONSTRUCTION

For *inter partes* review, claim terms are given the ordinary meaning that they would have to a POSITA on the earliest effective filing date, in view of the specification and file history. 83 Fed. Reg. 51340; *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312-13 (Fed. Cir. 2005) (*en banc*).

The Board need only construe terms to the extent necessary to resolve the disputed issues. Unless otherwise expressly discussed, Petitioners apply the plain and ordinary meaning of all claim terms below. Construction of certain claim terms may be relevant to issues in the parallel district court litigations⁵ Petitioners do not waive their right to raise additional issues of claim construction in any litigation, nor do they waive any argument in any litigation that claim terms are indefinite or otherwise invalid.

VIII. GROUND 1: CLAIMS 1-21 ARE OBVIOUS OVER SCHNEIDER

As explained below, claims 1-21 are obvious over Schneider.

⁵ Claim constructions positions exchanged and briefing filed to date in the parallel litigations are submitted herewith as EX1015-EX1021.

A. Schneider

Schneider (EX1005) published March 11, 1999, and thus qualifies as prior art under at least Sections 102(a-b). Schneider is analogous art to the '850 patent at least because it is from the same field of endeavor. The '850 patent describes its “Field of Invention” as “file preservation, file capture, file management and file integrity techniques in computing devices.” EX1001, 1:12-16. Schneider similarly relates to “the backup and recovery of data stored by a digital computer.” EX1005, 1:15-17. Thus, the '850 patent and Schneider are both within the same field of endeavor of computer file storage, management, and recovery. EX1004, ¶¶48-49.

With Schneider’s technology, “if the information on disk changes and it is determined the original state was desired, a user uses the backup to recover this original state.” *Id.*, 1:28-2:1. Schneider provides examples of such scenarios, including “a computer system ‘crashing’ during an update of a piece of information, thus leaving it in neither the original or ‘new’ state, the user changing information only later to desire to restore (or just reference) the original state, a computer virus altering information, or a file being deleted accidentally.” *Id.*, 2:6-10. EX1004, ¶50.

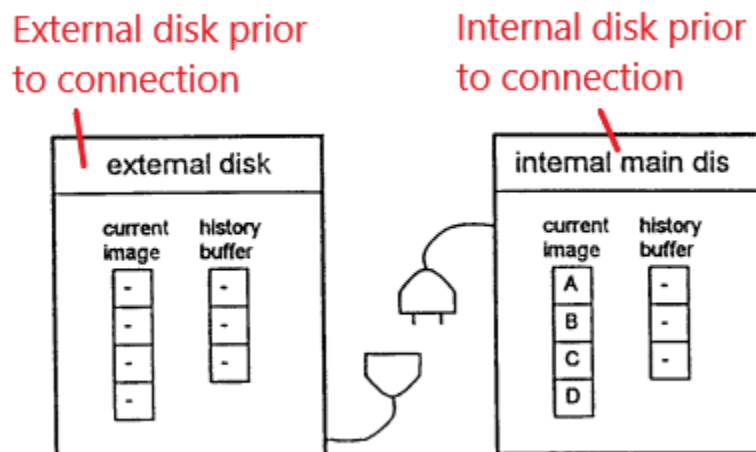
Among its “essential components,” Schneider discloses the ability to restore a prior state of information: “the ability to simulate a time reverted disk while at the same time allowing the user to continue using the current disk (thus, for example, allowing you to copy forward information from the past into the current).” *Id.*, 11:6-

17. The user may also “completely revert a disk to a prior state in time.” *Id.* EX1004, ¶51.

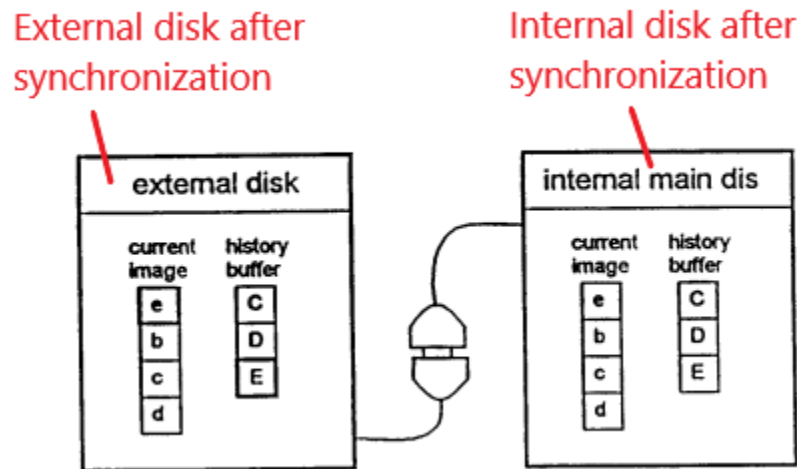
Schneider discloses that its techniques are applicable to file-level data recovery, allowing a user to select a prior version of a specific file and recover only the data for that file at a specific point in time. *Id.*, 23:29-24:6. This file-level implementation is “based on the principal of saving original states of recent changes” and allows the operating system to “simply present the user with a list of prior states for a given file, and allow any to be copied forward.” *Id.*, 24:7-12. For example, Schneider discloses “an extension to [Microsoft] 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.” EX1005, 60:3-7. It also discloses a process that “shows all available old versions of files and directories ... in the form of a hierarchy.” *Id.*, 60:15-20. Schneider also “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,” e.g., by copying the file to a temporary directory that is later purged. *Id.*, 60:21-29; *see also id.*, 130:10-17. EX1004, ¶¶52-53.

Finally, Schneider teaches maintaining the current version of a file on the user’s local disk, i.e., the “internal disk”, while maintaining historic data on a separate “external disk” remotely accessible via a network. EX1005, 112:26-114:7. For example, “each PC has an internal disk for which a portion represents OS visible

data” and the internal disk may also have “zero” or more space “set aside for historic data.” *Id.*, 113:9-13. It also teaches an external disk with “additional storage used for historic states.” *Id.*, 113:15-19. In this manner, compared to the internal disk, “the external disk may be able to reach further back in time in re-creating prior states.” *Id.*, 113:19-21. Schneider teaches that the user’s “PC’s software would divert to and take advantage of an external disk that had more historic information.” *Id.*, 113:26-28. An example of this storage of files to local and external disks is shown in Figures 47A and 47I, reproduced and annotated below. 47A depicts the internal and external disks before being connected, while 47I shows the state of the disks after they have been fully synchronized. *Id.*, 112:1-3, 112:24. As shown, over time the external disk mirrors the same data contained in the internal disk, thereby providing an effective backup of the disk. EX1004, ¶54.



Schneider FIG. 47A (annotated)



Schneider FIG. 47I (annotated)

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. EX1004, ¶37; 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). EX1004, ¶37. Schneider describes, and a POSITA would recognize, the compatibility of each of these features and the POSITA would have no issue implementing a file recovery system that included each of them. EX1004, ¶55.

1. Claims 1 & 10⁶

a) [1.1]⁷/[10.1]

[1.1]/[10.1] 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:

To the extent the preamble is limiting, Schneider discloses it.

Schneider discloses “allowing the user to view a list of earlier versions of a file,” select an earlier version, and then “replace[] the current version” with the recovered earlier version of the file. EX1005, 130:10-17; *see also* 23:29-24:12 (the system “present[s] the user with a list of prior states for a given file, and allow[s] any to be copied forward”); *id.*, 31:19-24 (discussing ability “to locate and recover versions only of desired files”). EX1004, ¶58.

⁶ The limitations of claims 1 and 10 are identical except that Claim 1 includes certain additional limitations not included in claim 10. EX1004, ¶¶42, 56. Accordingly, the Petition presents and analyzes the limitations of these claims together and identifies the additional limitations contained in claim 1 and not claim 10 [*bracketed and in italics*].

⁷ Reference numbers in the format of [claim#.limitation#] are added throughout for ease of reference.

Schneider teaches that a file's current version is available at a local storage location, such as the internal disk of the user's PC as shown in FIGS. 47A-G. *See id.*, 113:9-10 (“each PC has an internal disk for which a portion represents OS visible data”); *see also* 112:1-114:5 (describing state of current version on internal disk, over time); FIGS. 47A-47F, 47H-47I (showing “current image” on “internal main” disk); 110:13-14 (describing “changes that were made to the current image” on the main internal disk); FIG. 47G; 135:6-8, FIG. 63. Schneider explains that the internal disk contains non-historic system data, such as information visible to the operating system. *Id.*, 113:9-13; EX1004, ¶58.

b) [1.2.i]/[10.2.i]

[1.2.i]/[10.2.i] (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,

Schneider discloses presenting information indicative of previous versions of a file, e.g., using an extension to the “industry standard Windows95 Explorer utility” that allows the user to “right click on a specific file and have the option to view a list of old versions of the file.” EX1005, 59:27-60:5; *see also id.*, 58:20-23 (describing history log portion of history buffer), 58:24-27 (describing general log portion of history buffer), 130:10-17 (describing system “allowing the user to view a list of earlier versions of a file”), 31:19-24 (user may “select a file and request the present invention's software to locate all available versions of that file, over time”). The

presented list of old file versions is information indicative of at least one or more previous versions of the file. EX1004, ¶¶59-60.

c) [1.2.ii]/[10.2.ii]

[1.2.ii]/[10.2.ii] wherein a restorable representation of each version, V, of the previous versions, is retrievable from a remote storage location,

The file-level data recovery implementation taught by Schneider maintains a restorable representation of each version presented to the user. For example, a “selection can be made from the list [of earlier versions] and the recovered file either replaces the current version or is copied to a new file.” EX1005, 130:10-13; *see also id.*, 23:29-24:12 (the system “present[s] the user with a list of prior states for a given, file and allow[s] any to be copied forward”). 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; *see also id.*, 109:7-8 (describing “external disk drive that, like the main disk, employs a method of saving original disk states”). To provide redundant backup via an external disk, Schneider discloses a process whereby “[w]hen an external disk that has been set up is connected to the computer, the engine seeks to synchronize it

with the internal disk.” *Id.*, 111:11-12; *see also id.*, 111:3-112:24 (describing general process for transferring/storing historic data on the external disk), 112:26-114:7 (describing how to implement the external disk as “a disk interfaced to a target computer through a network”). Schneider teaches use of an external disk to ensure “a guaranteed usable backup image is available” even if the local drive crashes or otherwise becomes unavailable. EX1005, 109:19-29, EX1004, ¶62. Schneider teaches one implementation in which the external (remote) disk maintains the same historic data as the internal (local) disk. EX1005, 113:15-17; FIGS. 47A-F, 47H-I.⁸ Schneider also teaches, alternatively, implementing the external disk with more “additional storage used for historic states,” allowing the external disk “to reach

⁸ To the extent Patent Owner contends that Schneider’s disclosure of an external disk only applies to Schneider’s disk level implementation, a POSITA would have understood that such teachings were equally applicable to Schneider’s file level implementation because Schneider discloses that its “present invention can be implemented to work either with managing data at the disk sector level or the file level.” EX1005, 10:25-27; *see also id.*, 28:2-6 (explaining that in the file-level implementation, data pages in the history buffer “would correspond to file information instead of disk sectors”). EX1004, ¶62, fn. 6.

further back in time in re-creating prior states.” *Id.*, 113:17-21; *see also* FIG. 47G. EX1004, ¶62.

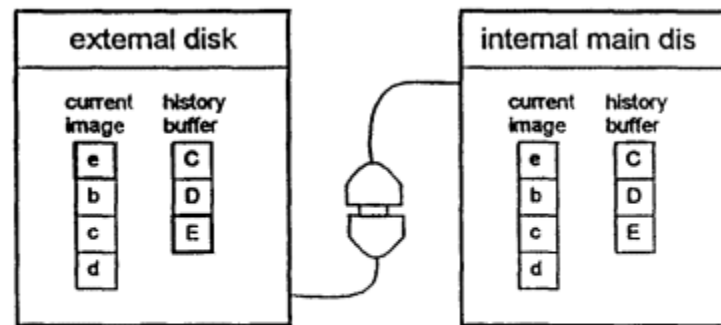


Figure 47I

EX1005 FIG. 47I

Each prior version of a file stored on the external disk is retrievable because Scheider teaches that the external disk’s historic information is used to “re-creat[e] prior states.” *Id.*, 113:19-21, 24:21-24 (describing saving a prior file state “just before a file is opened and the intention to modify is clear”), 112:29-113:2 (describing transfer “from” the external disk). EX1004, ¶63.

Schneider’s network-connected external disk is a remote⁹ storage location at least because it is not internal to the user’s PC, as shown in FIG. 47G, which depicts the historic data as backed up by a “server with engine software” that is connected to the user PCs via a LAN. EX1005, FIG. 47G; EX1004, ¶64.

⁹ The ’850 patent never uses the term “remote” outside the claims.

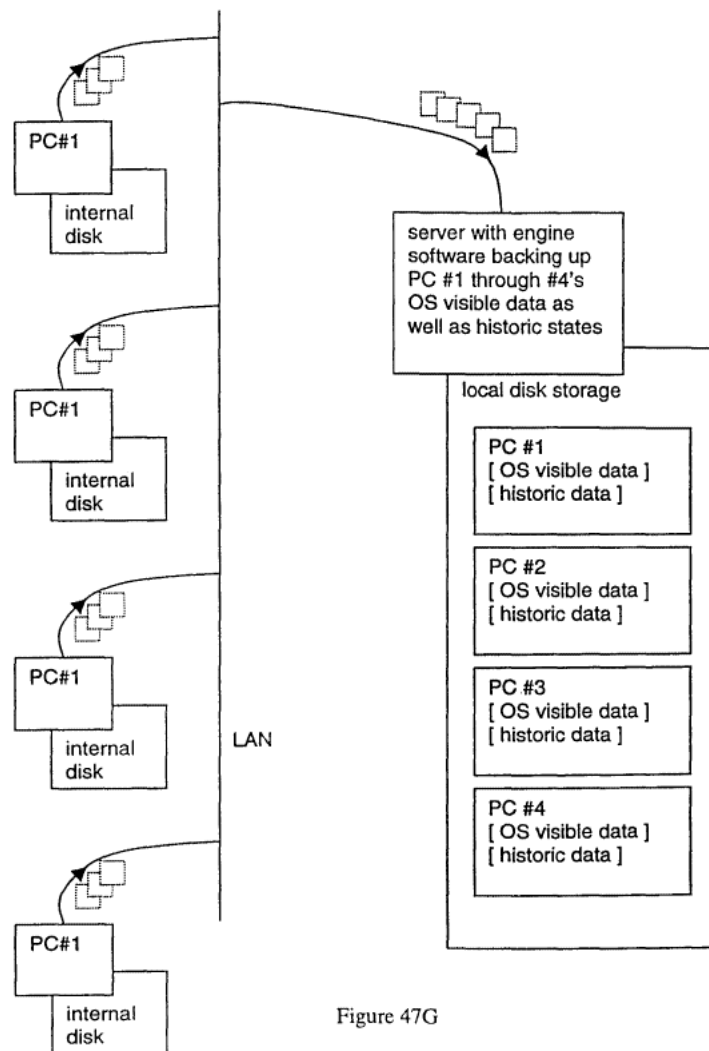


Figure 47G

EX1005 FIG. 47G

To the extent Patent Owner argues that Schneider’s network-connected external disk is not a “remote storage location,” e.g., because it is depicted as connected via a Local Area Network (“LAN”), it would have been obvious to connect to that external disk using a wide-area network such as the Internet instead of a LAN. *Supra* Section IV.B. As early as 1993, it was known that the Internet was “commonly used to transfer ... files between computers.” EX1009, 3:13-19

(background provided in patent issued from application filed November 4, 1993). Even then, the Internet was a known alternative to a LAN: “The Internet is similar to a LAN/WAN in that it was designed to allow many computers of differing types to interconnect and exchange information and programs.” *Id.*, 3:37-39; *see also* EX1008, 6:9-36 (1999 patent describing remote backup storage where “logical connections to one or more remote computers,” such as a “memory storage device,” included “commonplace” LAN and WAN connections, “such as the Internet”). It was also known to access external drives using the Internet, as “[d]rives on a remote system defined using NFS and RFS [we]re available for mounting by other remote systems on the Internet.” EX1009, 3:64-66; EX1008, 6:9-36; *see also supra* Section IV.B. By 2000, people had been “sharing data on the Internet for decades already” (EX1013, 544) and it was known that “[o]ne of the earliest Internet-based storage applications to arise since the emergence of the World Wide Web is system backup and recovery” (*id.*, 556). Thus, the POSITA knew how to implement Internet-connected drives, and that they were an established alternative to LAN-connected drives. EX1004, ¶65.

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 web site will not be destroyed by the same disaster that strikes a system in the office or at home.”); EX1004, ¶66. 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. EX1004, ¶66. A POSITA would easily implement this known alternative, as Schneider already explains how a network server “can communicate with more than one PC” and “can independently maintain the OS visible disk image and historic states for a set of PC’s.” EX1005, 113:3-5. The obviousness of this alternative is confirmed by the ’850 patent itself, which states that the storage device of the alleged invention “may be a personal attached storage device, a network attached storage device, an Internet storage area network, a peer-to-peer storage device, or other storage device,” without any explanation of how to implement any of these alternatives and treating them as interchangeable. EX1001, 5:35-40; EX1004, ¶66. And during prosecution, applicants did not dispute the Examiner’s finding that it would have been obvious to receive prior revisions of a file from an Internet storage area network. EX1002, 144. Accordingly, element [1.2.ii/10.2.ii] is obvious. EX1004, ¶66.

d) [1.2.iii]/[10.2.iii]

[1.2.iii]/[10.2.iii] the restorable representation having at least information required for recovering the version V <i>[by a computational machine with the</i>
--

current version being accessible], the remote storage location being accessible through a network;

As discussed for claim element [1.2.ii/10.2.ii] Schneider discloses that its historic data, e.g., as stored in a history buffer, is used to recover prior versions of files on the user's computer. *See also* EX1005, 24:12-14. When a specific file is recovered from the historic data, "the recovered file either replaces the current version or is copied to a new file." *Id.*, 130:10-13. As further explained (*supra* Section VIII.A.1.c), Schneider discloses accessing the remote storage location through a network and it would have been at least obvious for access to occur via the Internet to a geographically remote location. EX1004, ¶67.

The user's computer also provides access to the current version of a given file. For example, Schneider discloses that the current image is accessed via the user's computer and the current image constitutes "the non-historic view of the disk," i.e., the "data last written by the user." *E.g.*, EX1005, 35:26-36:3. Accordingly, a POSITA would understand that a current version of a given file is available and accessible to a user on the user's hard disk. EX1004, ¶68.

e) [1.3]/[10.3]

[1.3]/[10.3] (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]

As discussed for [1.1/10.1] and [1.2.i/10.2.i], Schneider discloses displaying a list of previous file versions by the computational machine (i.e., the user's computer). In response to user selection of a particular version, the Schneider system previews the version, e.g., by copying the previous file to a temporary directory and allowing the user to view that temporary copy:

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. ... 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.

EX1005, 60:21-29; *see also id.*, 60:3-5; EX1004, ¶69. Similarly, Schneider further teaches that its “invention” allows users to “simulate” historic versions “while at the same time allowing the user to continue using the current disk” and thus “copy forward information from the past into the current.” EX1005, 11:10-13; *see also id.*, 22:28-23:2; 130:10-13; 142:17-18, 160:27-28. At least by providing a temporary view of prior versions that can be viewed simultaneously with the current version of a file, the system taught by Schneider presents a presentable representation of the

selected previous version, the selected previous version being one of the previous versions of the file in the information presented to the user. EX1004, ¶69.

At least because Schneider discloses users viewing and copying from the prior versions, the presentable representation (i.e., the previewed file) includes “information required for presenting at least a portion of the selected previous version by the computation machine.” EX1004, ¶70.

As explained above for claim [1.2.iii/10.2.iii] (*supra* Section VI.A.1.d), Schneider discloses the current version of a file being accessible, and such current version would remain accessible during the preview process disclosed by Schneider. EX1004, ¶70.

f) [1.4]/[10.4]

[1.4]/[10.4] (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.

Schneider teaches that, from a “list of earlier versions of a file” displayed to the user, a “selection can be made from the list and **the recovered file ... replaces the current version.**” *Id.*, 130:10-13. A POSITA would have understood this to involve retrieving the restorable representations from the remote storage location and storing it locally because Schneider discloses restoring past files by replacing

the current file with the recovered file. EX1005, 130:11-13, 15:20-16:21, Fig. 2, EX1004, ¶71. Schneider discloses selecting a previous version of a file in order to restore it and store it as the current file version, e.g. by allowing a user to select the previous version from a displayed list of available stored previous versions. EX1005, 130:10-13. As explained above for element [1.1/10.1], Schneider discloses storing the current version by the local computer on the computer's local drive, i.e., a local storage location. EX1004, ¶71.

2. Claims 2 & 11¹⁰

[2/11]. The method of claim [1/10], wherein, for the presenting of (B), the presentable representation is presented using an application that created the file or the selected previous version.

Schneider teaches this limitation. Schneider teaches that when a previous file version is previewed, Schneider “automatically launch[es] the appropriate application to view the file.” EX1005, 60:21-23. A POSITA knew that the most appropriate application to view a specific file was the application that created the file, making that application at least an obvious design choice. EX1004, ¶73. A

¹⁰ Dependent claims 2, 4, 5, 6, 7, 8, and 9 (depending from claim 1) are substantively identical to dependent claims 11, 12, 13, 14, 15, 16, and 17 (depending from claim 10). Accordingly, this Petition presents and analyzes each such corresponding pair of dependent claims together. EX1004, ¶¶43, 72.

POSITA understood that computer files are typically viewable by the same software application that created the file. EX1004, ¶73. As just one example, in 2000 Microsoft Word was the most widely used word progressing program. EX1004, ¶73. Microsoft Word saved files created using the software using a file format specific to that application, using the “.doc” extensions, and documents stored using that extensions are viewable using Microsoft Word. It would be evident to a POSITA that when a user created a specific previous version of a file using a software application on her computer, with the application creating the file according to the application’s specific file format, that same software application on the user’s computer would be used to view that file. EX1004, ¶73.

3. Claim 3

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

Schneider teaches this limitation. Schneider teaches that when a user selects a listed previous version of a file to preview, the file may be viewed using an appropriate application by copying the file to a temporary directory. EX1005, 60:21-29. The contents of this temporary directory, “when no longer in use, are eventually automatically purged.” *Id.*, 60:24-25. Thereby, 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.” *Id.*, 60:28-29. Accordingly, 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” (*id.*, 60:27-28). EX1004, ¶74.

It would have also at least been an obvious design choice to implement Scheider such that the preview of a given previous version was opened in read-only mode. EX1004, ¶75. A POSITA knew that applications had three access permission modes for opening or viewing files: “read-only” (i.e., enabling only viewing of the file), “read/write” (i.e., enabling viewing and editing the file) and read, write, and delete (i.e., enabling viewing, editing, and file deletion). EX1014, 398, 440; EX1004, ¶75. The most obvious design choice for implementing Schneider’s temporary preview feature was for the application to open the file in read-only mode. Further, the POSITA would have been additionally motivated to open the file in read-only mode so as to avoid inadvertent overwriting of the contents of the previous file version. Read-only mode would thus maintain the underlying purpose of Schneider’s file recovery system to retain previous versions for later viewing. EX1004, ¶75. The POSITA had a reasonable expectation of success with such an implementation because read-only is one of the three fundamental permission modes for accessing a file. EX1004, ¶75.

4. Claims 4 & 12

[4/12]. The method of claim [1/10], wherein the information includes at least two previous versions of the file.

Schneider discloses this limitation. Schneider presents information for multiple (at least two) previous versions of a selected, e.g., as part of its displayed “list of earlier versions of a file.” EX1005, 130:10-13. *See also* 31:21-24 (“[The user] 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.”). EX1004, ¶76.

5. Claims 5 & 13

[5/13]. The method of claim [1/10], wherein the information for the collection includes information indicative of at least one or more of a timestamp, an original file location, and a comment of the previous versions or the file.

Schneider teaches this limitation. As explained, Schneider discloses storing previous file versions for an archived file in a history buffer. EX1005, 24:12-14, 31:12-24 Schneider further discloses that its history buffer is “organized in[to] blocks” with each block having a header containing a table of mapping entries. EX1005, 26:19-24. Each table entry contains: “**1) a type, 2) a time stamp, and 3) an original disk location.**” *Id.*, 26:25-27. And Schneider teaches implementing this organization scheme “at the file level, where the data pages would correspond to file information [and o]ther details would also change as are obvious.” *Id.*, 28:3-6. As explained above, Schneider also teaches “show[ing] all available old versions of

files.” *Id.*, 59:27-60:20. Accordingly, a POSITA would have implemented Schneider to present information that included a timestamp, an original file location, and a comment indicating a file type, for each archived previous version of a file.. EX1004, ¶77.

6. Claims 6 & 14

a) [6.1]/[14.1]

[6.1/14.1] The method of claim [1/10], further comprising the steps of: (B2) after the presenting of (B) and responsive to a selection to not restoring the previous version and responsive to a second selection of a second previous version of the previous versions, version presenting a second presentable representation of the selected second previous version, the second presentable representation having at least information required for presenting at least a portion of the selected previous version; and

In the Parallel Litigation, Patent Owner suggests that this limitation recites nothing more than a user to repeat the claim steps recited in claims [1.2.i-iii/10.2.i-iii], and Schneider renders the limitation obviousness under such an interpretation.¹¹

¹¹ In the Parallel Litigation, Petitioner Dropbox contends that the term “responsive to a selection to not restoring the previous version” is indefinite as being incompatible with independent claim 1’s limitation “responsive to a selection to restore the selected previous version.” EX1017, 2-5; *see also*, EX1021, 3. Patent owner contends that these claims are not inconsistent because a “user can (pre)view

It would be common sense to a POSITA that a user would repeat the steps of claims [1.2.i-iii/1.10.i-iii] to iterate between previewing and rejecting multiple previous file versions until the user finds the correct file version the user wishes to restore. EX1004, ¶78; *Perfect Web Techs., Inc. v. InfoUSA, Inc.*, 587 F.3d 1324 (Fed. Cir. 2009) (affirming District Court obviousness determination that it would be common sense to perform repetition of a known procedure until success is achieved).

As explained, Schneider presents to a user a list of previous file versions and allows a user to select a file to preview and subsequently select to retrieve and restore that file to save it as the current version. *See, e.g.*, EX1005, 60:3-5. A POSITA would understand that a user would implement Schneider's file retrieval system iteratively. That is, a POSITA would have understood the purpose of Schneider's file preview functionality of allowing a user to view the contents of a previously stored version, without having to actually retrieve and restore that file version. EX1004, ¶79; 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."). A user searching for a specific previous file version may not know which

and restore multiple times across various versions of files" and could therefore first select to restore a first file version and later decide not to restore that version and restore a different version. EX1018, 7-8; *see also*, EX1021, 3.

listed previous version is the one she is looking for without first previewing the content of the file. EX1004, ¶79. Thus, a user might go through multiple iterations of selecting a first previous file version to preview, determining that the previewed version is not the correct version, selecting to not restore that version (e.g., by closing the preview display or selecting another version in the presented list of file versions)¹², and selecting to preview the second version, until the user finds the correct file version that she is searching for. EX1004, ¶79.

b) [6.2]/[14.2]

[6.2/14.2] (C2) responsive to a selection to restore the second selected previous version, retrieving the restorable representation of the second selected previous version from the remote storage location and storing the second selected previous version as the current version on the local storage location, the second selected previous version available from the restorable representation of the second selected previous version.

Schneider teaches this limitation.

As explained above for claim [1.4/10.4] (*supra* Section VIII.A.1.f), Schneider discloses a process of responsive to a selection to restore a selected previous version, retrieving the restorable representation of the selected version from the remote

¹² In the Parallel Litigation, Patent Owner contends that a selection to not restore a previous version may be satisfied by “clicking on another version . . . or interacting with some other input/element.” EX1018, 7.

storage location and storing that version as the current version on the local storage location, with the selected previous version available from the restorable representation of that version. A POSITA understood that a user, after iteratively previewing multiple previous versions, would implement that same procedure to select, retrieve, and restore a second selected previous version (i.e., where the user determines that the second selected previous version is the version the user is searching for). EX1004, ¶80.

7. Claims 7 & 15

[7/15]. The method of claim [1/10], wherein the restorable representation of the selected previous version comprises one or more of a compressed and encrypted version of the selected previous version.

Schneider teaches this limitation. Schneider states that it “is anticipated that data compression techniques could be applied to the history buffer to reduce the space it requires.” EX1005, 26:2-4. Accordingly, a POSITA would implement data compression to the previous file versions stored in the history buffer to reduce the space required to store such data. EX1004, ¶81.

8. Claims 8 & 16

[8/16]. The method of claim [1/10], wherein the restorable representation of the selected previous version comprises an incremental change to the file.

Schneider discloses this limitation. As explained, Schneider discloses performing automatic backup in response to an operation such as a write operation.

EX1005, 12:1-2, 12:5-8, 24:7-10, 24:12-14, 24:21-23, 36:12-17. Therefore, a POSITA would understand that the list of previous file versions presented by Schneider would contain incremental changes introduced by successive write operations to the relevant file. EX1004, ¶82.

9. Claims 9 & 17

[9/17]. The method of claim [1/10], wherein the method has a substantially imperceptible impact on system performance from a user's point of view by delaying one or more operations to prevent a perceptible impact on system performance from a user's point of view.

Although there is uncertainty as to what constitutes a “substantially imperceptible impact ... from a user's point of view,” Schneider discloses this limitation.¹³ The ‘850 patent indicates that this step can be performed during a data backup process and is not limited to the stage of retrieving stored data. The patent discloses a method of automatically storing a backup version of a file at the time the file is opened by a user in response to an instruction to open the file. “In step 210, file capture block 10 detects the instruction and captures the operating file.... In step

¹³ In the Parallel Litigation, Petitioner Dropbox contends that the term is indefinite for failing to inform a POSITA with reasonable certainty about the scope of the claimed invention. EX1017, 9-11; *see also*, EX1021, 3. Regardless, a POSITA would recognize the disclosures of Schneider to be within the bounds of the term.

215 the resident program opens the operating file.... Step 210 is performed by momentarily delaying the execution of step 215 in such a manner as to have little or no perceptible impact on system performance from the users [sic] point of view.” EX1001, 7:9-28. EX1004, ¶83.

Similarly, Schneider discloses that when files are written to archive, “[i]n order to least impact the user’s use of a computer, a goal is implementing the present invention to minimize the performance impact on reading data and rely on writes being a background activity. If writes are done in the background then overhead can be added without impacting response to the user.” EX1005, 32:9-12. Accordingly, Schneider teaches delaying a write operation such that its data backup operations have a substantially imperceptible impact on system performance from user’s point of view. *See also id.*, 37:22-29 (“Caching writes in memory and committing them to disk during free time can reduce or eliminate the impact on the user, even though there is a tripling of time in the actual writing of the data.... [I]n theory the user had been free to continue working and so should not notice the performance degradation.”). EX1004, ¶84.

10. Claim 18

a) [18.1]

[18.1] 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:

Schneider discloses this limitation for the same reasons explained for identical preamble element [1.1]. EX1004, ¶¶ 46, 85.

b) **[18.2]**

[18.2] (A) iteratively transmitting a representation of at least a portion of a current version of a file to a remote storage location for archiving the current version responsive to a determination of an availability of the remote storage location for accepting a transmission of the representation, the current version being accessible at a computational machine, the remote storage location being accessible through a network;

Schneider discloses this limitation. As explained, Schneider discloses performing automatic backup in response to an operation such as a write operation. EX1005, 12:1-2, 12:5-8, 24:7-10, 24:12-14, 24:21-23, 36:12-17. Accordingly, a POSITA would understand that Schneider iteratively transmits such backed-up file versions to Schneider's remote storage location, because Schneider would store and transmit the current version of the file each time the file is backed up responsive to a write operation. EX1004, ¶86. Schneider teaches to a POSITA a determination of whether a remote storage location is available in that it discloses that the external storage disk can be removable and that transfer of information does not begin until the external storage is reattached. EX1005, 110:24-28. It would further be obvious to a POSITA to make a determination of the availability of the remote storage location prior to transmitting file back-up versions. EX1004, ¶86 A POSITA would understand that if the remote storage location is unavailable, for example if the

remote storage is faulty, without power, or disconnected from the network, transmission of file back-ups would be unsuccessful and the user should be notified of the problem so that saved files may be backed up using a separate method or and so the user can fix the problem with the remote backup location. EX1004, ¶86.

Schneider teaches a current file version being accessible at a computational machine (i.e., a local computer) and a remote storage location being accessible through a network for the same reasons explained for elements [1.2.i-ii/10.2.i-ii] (*supra* Sections VIII.A.1.b-c). EX1004, ¶87.

c) [18.3.i]-[18.3.ii], [18.4]-[18.5]

Claim elements [18.3.i]-[18.3.ii] and [18.4]-[18.5] are textually identical to elements [1.2.i]-[1.2.ii] and [1.3]-[1.4], respectively, except for internal headers (A)-(D).. EX1004, ¶¶44, 88. Schneider discloses each of elements [18.3.i]-[18.3.ii] and [18.4]-[18.5] for the same reasons explained for respective elements [1.2.i]-[1.2.ii] and [1.3]-[1.4] (*supra* Sections VIII.A.1.b-c, e-f).

d) [18.3.iii]

[18.3.iii] the restorable representation having at least information required for recovering the version V by the computational machine, the remote storage location being accessible through a network;
--

Schneider discloses this limitation for the same reasons explained for limitation [1.2.iii] (*supra* Section VIII.A.1.d). Limitations [18.B.iii] and [1.2.iii] are

textually identical except that [1.2.iii] requires the additional language of “with the current version being accessible.” EX1004, ¶¶44, 89.

11. Claim 19

a) [19.1]

[19.1] 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:

Schneider discloses this limitation for the same reasons explained for identical preamble element [1.1]. EX1004, ¶¶ 46, 90.

b) [19.2]

[19.2] (a) receiving a request to display a list of captured revisions for a file; (b) displaying a list of captured revisions for said file;

Schneider discloses this limitation for the same reasons explained for element [1.2.i/10.2.i] (*supra* Section VIII.A.1.b). Schneider discloses allowing the user to “right click on a specific file and have the option to view a list of old versions of the file.” EX1005, 59:27-60:5; *see also id.*, 58:20-23, 58:24-27, 130:10-17, 31:19-24. EX1004, ¶91.

c) [19.3]

[19.3] (c) receiving a selection from said list indicating a revision of said file to restore; (d) previewing selected revision of said file, wherein said selected revision of said file is received from an Internet storage area network;

Schneider teaches this limitation for the same reasons explained for element [1.3/10.3] (*supra* Section VIII.A.1.e). The '850 patent defines “Internet storage area network” as “any storage area (device, collection of devices, etc.) that can be accessed by the computing device when the computing device is temporarily or permanently connected to the Internet.” EX1001, 4:45-48. EX1004, ¶92.

In response to user selection of a particular version, the Schneider system previews the version by allowing the user to view a temporary copy. EX1005, 60:21-29; *see also id.*, 60:3-5. EX1004, ¶92.

d) [19.4]

[19.4] (e) receiving another selection from said list indicating another revision of said file to restore; (f) previewing selected another revision of said file wherein said selected another revision of said file is received from a network attached storage location; and

Schneider teaches this limitation for the same reasons as explained for element [6.1/14.1] (*supra* Section VIII.A.6.a). That is, a POSITA would understand that a user attempting to restore a particular file version would utilize Schneider’s file preview functionality iteratively to select and preview successive previous versions of a file until the user finds the correct version the user was searching for. EX1004, ¶93.

e) [19.5]

[19.5] (g) restoring selected another revision of said file following said previewing step in (f), optionally, decrypting said selected revision of said file prior to step (d) and decrypting said selected another revision of said file prior to step (f).

Schneider teaches this limitation for the same reasons as explained for element [1.4/10.4] and [6.2/14.2] (*supra* Section VIII.A.1.f, VIII.A.6.b). A user seeking to restore a particular file version, after finding the correct version through iteratively previewing multiple versions would then select to restore the file. Schneider discloses that after displaying its list of prior file versions “a selection can be made from the list and **the recovered file ... replaces the current version.**” *Id.*, 130:10-13. EX1004, ¶94.

All parts of this claim limitation following the word “optionally” should be treated as optional and therefore can be omitted and require no additional disclosure in the prior art. *See In re Johnston*, 435 F.3d 1381, 1384 (Fed. Cir. 2006) (“[O]ptional elements do not narrow the claim because they can always be omitted.”). A POSITA would understand the entirety of the limitation following the word “optionally” to be modified by the word and therefore be an optional element, because the remainder of the limitation following “optional” is a single clause undivided by any punctuation. EX1004, ¶95.

12. Claims 20 & 21¹⁴

[20/21]. The method of claim 19, [wherein said method comprises/further comprising the step of] decompressing said selected revision of said file prior to [step] (d) and decompressing said selected another revision of said file prior to [step] (f).

Schneider discloses this limitation. As explained above for claims 7/15 (*supra* Section VIII.A.7), Schneider discloses performing data compression for files stored in its history buffer. A POSITA would understand that compression of archived data would naturally also require decompressing an archived file version in order to display the file. EX1004, ¶97.

**IX. GROUND 2: CLAIMS 1-21 ARE OBVIOUS
OVER SCHNEIDER IN VIEW OF CRAWFORD**

To the extent Patent Owner argues that Schneider and the knowledge of a POSITA alone does not render obvious a remote storage location accessible via a network (independent claims 1, 10, 18) or an internet storage area network and network attached storage location (independent claim 19), such storage was obvious over Crawford (EX1009). Crawford published on June 23, 1998, and is thus prior art under at least Sections 102(a) and 102(b). EX1004, ¶98. Crawford is analogous

¹⁴ Claims 20 & 21 are substantively identical, with the differences between the claims shown in brackets. Accordingly, this Petition presents and analyzes these claims together. EX1004, ¶¶45, 96.

art to the '850 patent because it is from the same field of endeavor. Crawford discloses an “Internet online backup system provid[ing] remote storage.” EX1009, Title. Crawford thereby allows computer data recovery, when recovery is necessary. *Id.*, 8:13-22. Thus, the '850 patent and Crawford are both within the same field of endeavor of computer file storage, management, and recovery. EX1004, ¶98.

The invention of Crawford allows a local customer computer to connect to a remote server over the internet “and obtain additional processing and storage resources for the customer’s computer. The resources can take the form of virtual storage and processing capabilities.” EX1009, Abstract. EX1004, ¶99.

Crawford discloses a preferred embodiment as depicted in Figure 1, reproduced below. In this embodiment, a computer 50 is connected to an on-line service system 100 over a data link 150. *Id.*, 12:40-45, 14:18-21, Fig. 1. This data link “may comprise a dial up telephone line or other similarly convenient telecommunications link that allows customer computer 50 to be located remotely to the on-line service system 100.” *Id.*, 14:22-25. Both the computer 50 and on-line service system 100 include a modem to facilitate the data link. *Id.*, 16:3-9. EX1004, ¶100.

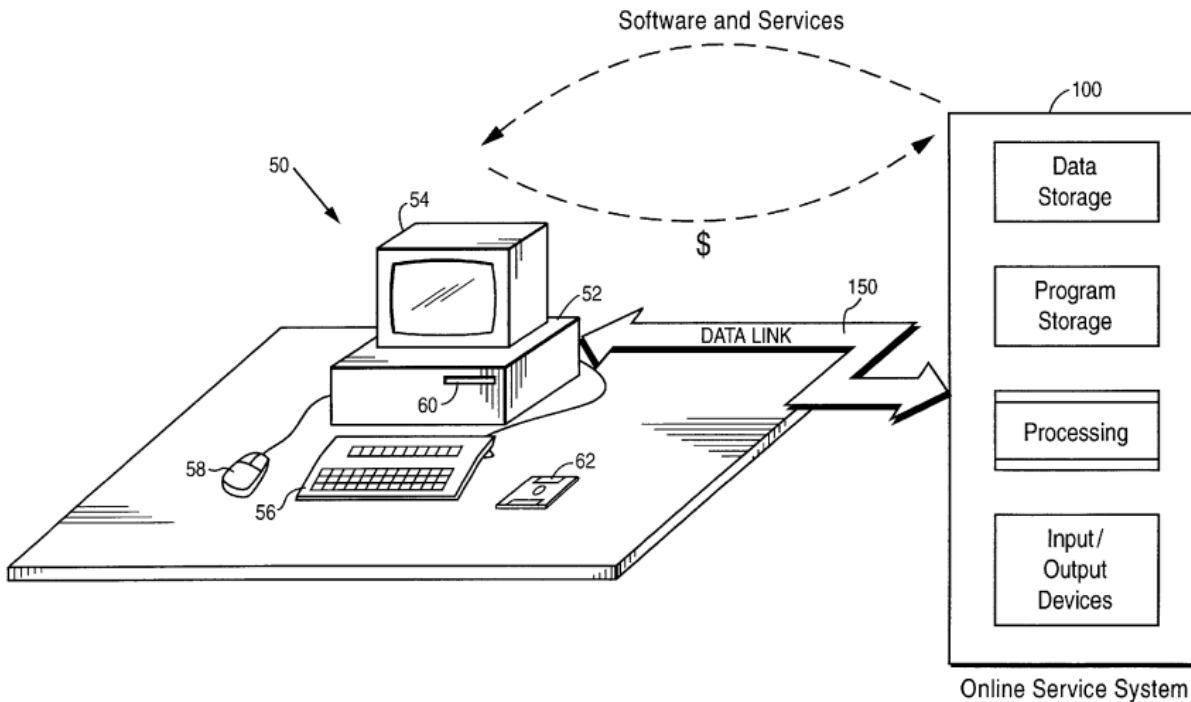


Fig. 1

Via this data link, the system of Crawford provides offsite archival services whereby files on the local customer computer “are automatically copied to on-line service virtual disks for offsite archiving.” *Id.*, 14:45-48. When needed, this on-line virtual disk archive is used to restore files on the local customer computer. *Id.*, 14:50-53 (“Virtual disks inactive for a specified period can be copied to on-line service tape for long term offsite archival.”). Crawford therefore discloses computer file backup and restoration with a remote storage location over an Internet network connection. *See id.*, Title (“Internet Online Backup System Provides Remote Storage”), 61:20-24 (claim 12) (“allowing the customer to restore data from the online backup service provider storage over the Internet connection”). EX1004, ¶101

A POSITA would have been motivated to modify the file back-up and restore system of Schneider to include an Internet data link to a remote storage device as taught by Crawford. A POSITA would have done so to achieve the benefits explained for claim [1.2.ii/10.2.ii] for Ground 1 above (*supra* Section VIII.A.1.c). Crawford explains additional benefits for its Internet based storage system that a POSITA would want to achieve. EX1004, ¶102. Crawford explains that a user's local computer may run out of space, thereby limiting the amount of data that can be stored. EX1009, 13:28-31. Crawford solves this problem by connecting the local computer to additional processing and memory resources over the internet, thereby providing a user with "additional disk space, or access to a more powerful processor." *Id.*, Abstract. A POSITA would have therefore recognized that the teachings Crawford could improve the computing system of Schneider by providing it with additional computing resources including additional memory storage and processing power. EX1004, ¶102. A POSITA would have had a reasonable expectation of success of implementing remote storage over an Internet link as taught by Crawford, because Crawford implements such connecting using a conventional modem (*id.*, 16:3-7) and teaches that communications over the modem are performed using "conventional data packet transmission techniques" (*id.*, 57:44-48). EX1004, ¶102.

Further, to the extent Patent Owner argues that the term “optionally, decrypting said selected revision of said file prior to step (d) and decrypting said selected another revision of said file prior to step (f)” (claim [19.8]) is not merely an optional limitation, such limitation is also rendered obvious via the proposed combination of Schneider and Crawford. Crawford discloses encrypting data transmitted by the local customer computer to the remote online backup system. EX1009, 60:42-46 (claim 1) (“(1) encrypting backup data provided by the customer’s computer, (2) transmitting the encrypted backup data to the online backup service provider”). Because the system of Crawford encrypts data to send from the customer computer to remote backup, a POSITA would understand that such encrypted data would be decrypted when transmitted in the other direction from the online backup to the local computer. EX1004, ¶103. A POSITA combining the file backup system of Schneider with the remote Internet data link between local and remote storage as taught by Crawford would also be motivated to incorporate Crawford’s teachings of encrypting and decrypting data transmitted between the local computer and remote storage in order to give a user the option to encrypt files backed up to remote storage and decrypt such files when they are retrieved in order to ensure security of data in transmission and while it is being stored. *Id.*, ¶103.

**X. GROUND 3: CLAIMS 1-8, 10-16,
AND 18-21 ARE OBVIOUS OVER SWEAT**

As explained below, claims 1-8, 10-16, and 18-21 are obvious over Sweat.

A. Sweat

Sweat (EX1006) issued June 13, 2006, from an application filed March 24, 2000, and is prior art under at least Section 102(e). EX1004, ¶105.

Sweat claims priority to and is supported by the disclosures of provisional application No. 60/126,170 (EX1007, the “Sweat provisional”), filed March 25, 1999. EX1004, ¶¶106, 112, 113. Should Patent Owner introduce additional evidence and attempt to argue that the challenged claims are entitled to a priority date earlier than the September 21, 2000, provisional application filing date and prior to Sweat’s filing date, Petitioner reserves the right to show that Sweat is entitled to its provisional filing date as the appropriate 102(e) date.

Sweat is analogous art to the ’850 patent because it is from the same field of endeavor. Sweat discloses “a project hosting service” allowing users to view and download prior versions of files. EX1006, Abstract. Sweat allows a user to recover prior file versions from a server to the user’s local computer. *Id.*, 10:51-55. Thus, the ’850 patent and Sweat are both within the same field of endeavor of computer file storage, management, and recovery. EX1004, ¶107.

Sweat relates to “a method, apparatus, and article of manufacture for a project hosting service that a user can access with a web browser. With it, a user can communicate and collaborate with the members of a design team or construction project. Embodiments of the invention let a user share CAD drawing files (DWG DWF, etc.), document files, or any other type of file.” EX1006, 1:43-49. The hardware and software environment of Sweat is depicted in Figure 1, reproduced below, which shows a user’s client computer 100 connected to a server computer 106 via a network 102. *Id.*, Fig. 1, 3:2-6. The server 106 is connected, among other things, to a database 116 and a file server 120 that accesses file storage 124. *Id.*, 3:18-26. EX1004, ¶108.

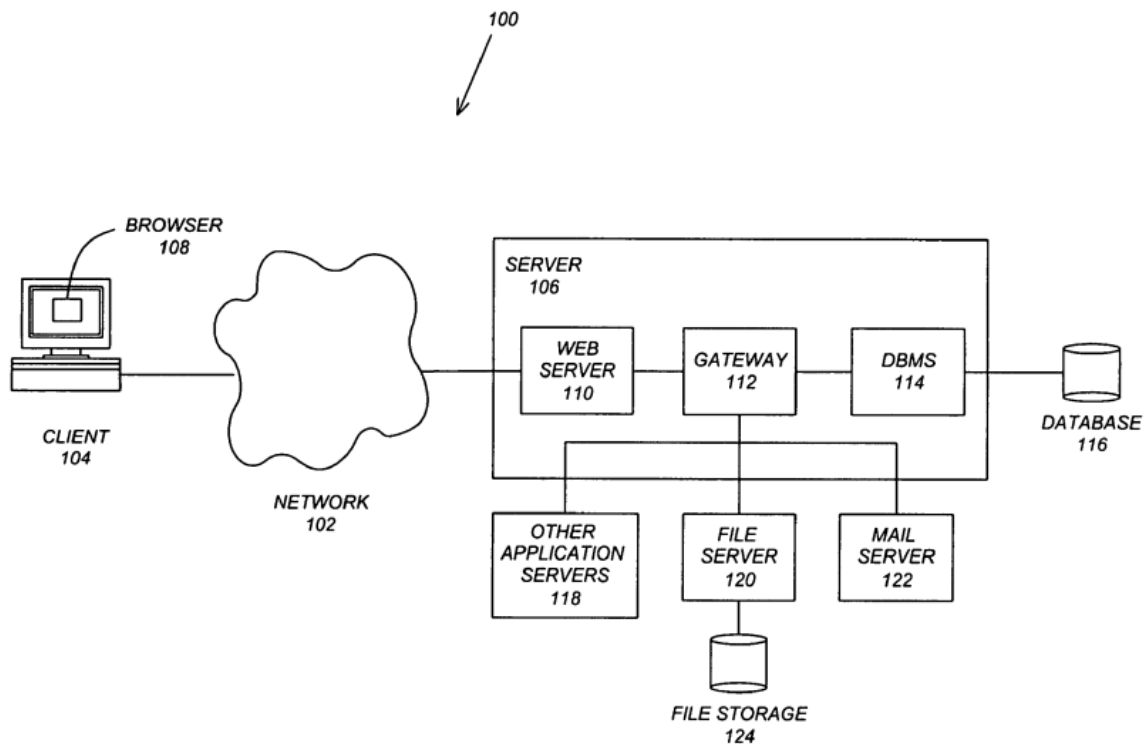


FIG. 1

Sweat’s distributed file management system manages projects that can be accessed by multiple users connected via the network. A project is managed in the form of a project folder, which “may be used to organize and store folders, project files, text files, other files, drawings, links, notes, etc.” *Id.*, 16:40-44; *see also id.*, 17:5-8 (“A user can create and use standard folders and sub-folders to organize and store project information (text files, drawings, links, and notes) to the best advantage for the user’s work environment.”). Sweat implements incremental backup of files from the client to the server as users make changes to files. “When a member saves changes or adds new information to files in the client 104... the new data is uploaded

to the server 106.” *Id.*, 17:16-19. Sweat allows multiple versions of a file to be saved, including allowing for an unlimited number of prior versions. *Id.*, 9:19-25, Fig. 4. As Sweat explains, “[f]ile versions allow the opportunity to create multiple versions of a single document wherein prior versions are available for viewing. When a user creates a new file version, an updated version of a file that already exists on the server 106 is updated,” creating a new version of the file. *Id.*, 19:42-52. EX1004, ¶109.

Sweat provides an interface to a user for viewing, previewing, and downloading previous versions of a file. Figure 5, reproduced below, shows an interface window displayed to a user when the user selects a specific file using Sweat’s file management system on the user’s client computer. *Id.*, Fig. 5, 2:16-18. EX1004, ¶110.

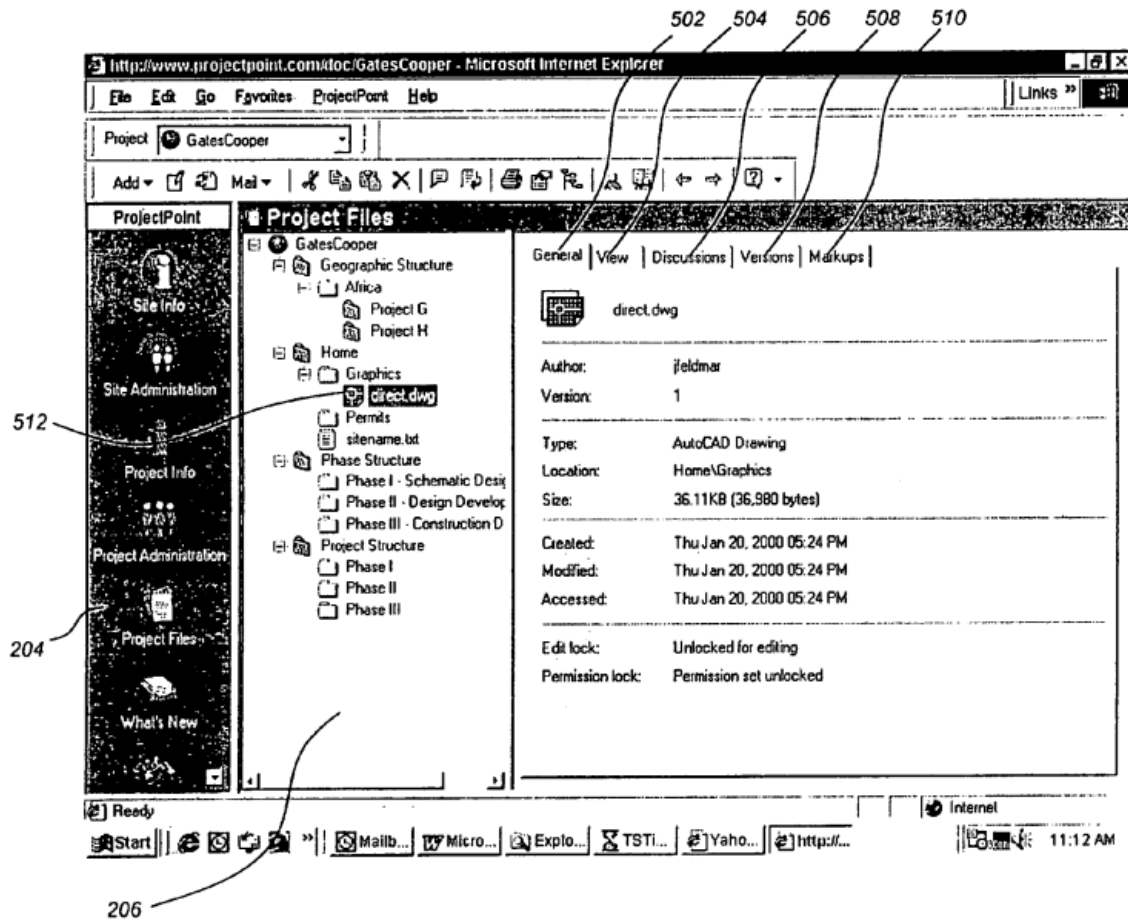


FIG. 5

A user can select the “versions tab 508,” which “provides a list of all of the currently selected files” and “is used to open and view any file version [and] download a selected file to a local computer 104.” *Id.*, 10:51-55. EX1004, ¶111.

1. Claims 1 & 10

a) [1.1]/[10.1]¹⁵

To the extent the preamble is limiting, Sweat teaches this limitation.

As explained with respect to the remainder of claim 1 below, Sweat discloses restoring a file to a previous version of the file.

Sweat teaches making a current version of the file available at a local storage location. For example, Sweat discloses that its versions tab 508 “is used to open and **view any file version [and] download a selected file to a local computer.**” EX1006, 10:51-55; EX1007, 9, 31¹⁶. Any file version would also include the most recent, i.e., the current version of the file. Sweat enables the user to download any file version, including this current version from the server to their local computer by clicking a “Download” button. EX1006, 10:56-61; EX1007, 9, 31. Accordingly, a POSITA would recognize a user would have a current version of a file stored to their local client computer, for example to work on and edit the file. EX1004, ¶116.

¹⁵ The full claim language is included in mapping the claims for Ground 1 and in the Listing of Claims (EX1022), and Petitioner hereinafter does not reproduce the claim language but instead references it by element/claim number.

¹⁶ Unless otherwise noted, Sweat Provisional cites are to the page numbers added by Petitioner.

Further, Sweat discloses that the user can use an “Update” command to designate a current file version at the user’s local computer as the newest file version at the server. Sweat discloses that “[i]f the user elects to update a file (e.g., by selecting the ‘Update’ option from the right-click menu), the user is prompted to select the file on the local computer 104 that corresponds to the file 512 to be updated on server 106.” EX1006, 11:5-8; EX1007, 7, 63. This further indicates that a user would have the current version available at their local computer and the user can also upload that current version to the server if desired. EX1004, ¶116-117.

b) [1.2.i]/[10.2.i]

Sweat discloses that its file management system backs up previous file versions to a remote storage server. EX1006, 17:16-19 (“When a member saves changes or adds new information to files in the client 104... the new data is uploaded to the server 106.”), 5:56-58 (“Use [the Update] command to select and upload a new version of a file to the server.”); EX1007, 63, 104. The server can store any number of previous versions of a file. EX1006, 9:19-25, Fig. 4; EX1007, 21, 25, 30; EX1004, ¶118.

Sweat discloses presenting this collection of previous file versions to a user. Specifically, Sweat discloses a “a list displayed in versions tab” that “is used to open and view any file version, download a selected file to a local computer 104, and review or edit comments that are attached to the selected file version.” EX1006,

10:52-57; EX1007, 31, 9. Sweat also describes an alternative “view tab” to view the previous versions EX1006, 10:57-58, 10:59-11:2; EX1007, 21, 25, 30. Accordingly Sweat discloses presenting information for a collection of one or more previous versions of a file (i.e., by presenting a list of those collected files), with the information for the collection including information indicative of at least one or more previous versions (i.e., by presenting each previous version in the list). EX1004, ¶119.

c) **[1.2.ii]/[10.2.ii]**

Sweat discloses this limitation. Sweat makes each previous version displayed in the list of versions tab 508 restorable from the remote storage location of server 106. As Sweat shows in Fig. 1, reproduced below, server 106 is accessible by a user’s local computer over network 102. EX1006, Fig. 1, 3:2-6; EX1007, 3, 53.

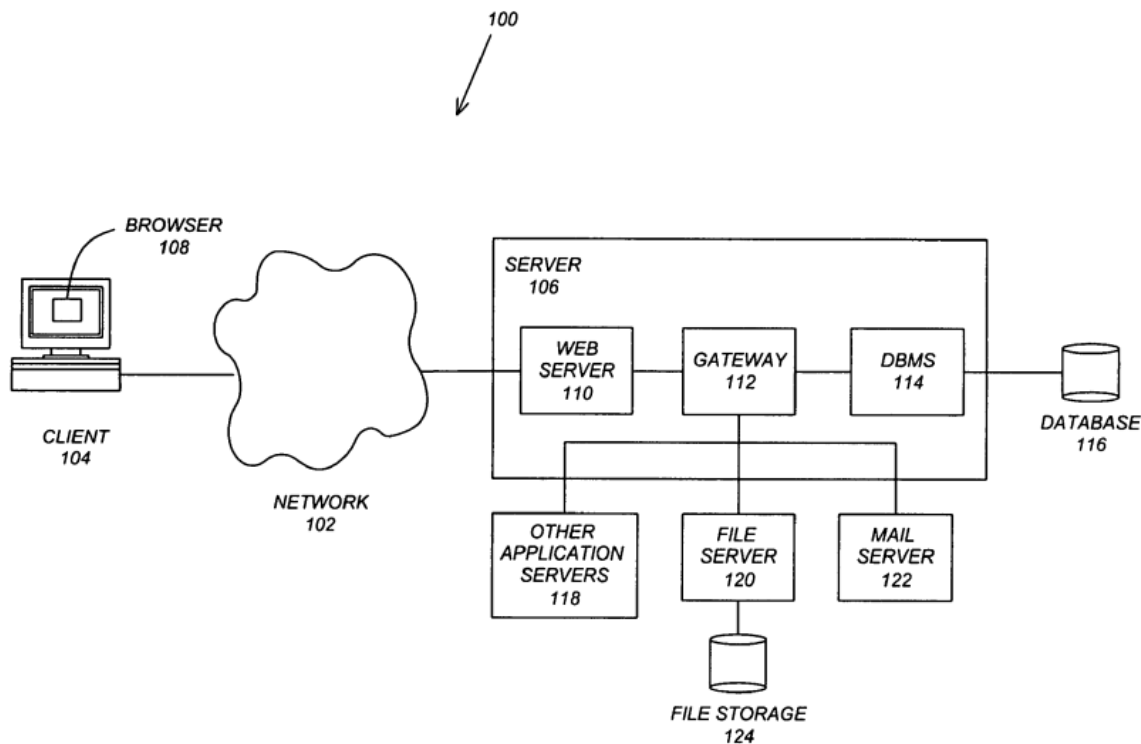


FIG. 1

Sweat specifically describes the network as “Internet 102.” EX1006, 3:12-14; EX1007, 3, 103.A POSITA would therefore recognize Sweat’s server 106 to be remotely connected to the local computer over the Internet, and server 106 (and storage connected thereto such as file storage 124) constitutes a remote storage location. EX1004, ¶120.

A user can restore a previous version by downloading a prior stored version from server 106 to the user’s local computer. “To download a version of the file from the server 106 to the local computer 104, the user selects the desired version and clicks a ‘Download’ button. Thereafter, the user identifies/selects the target

location to save the selected file. The file is then copied to the location specified.” EX1006, 10:58-63; EX1007, 31, 9. When a user clicks the “Download” button, the system “[c]opies the file or folder to the user’s local computer” and the user can “overwrite files that exist on the user’s local computer, as desired.” EX1006, 5:52-55; EX1007, 33, 79. A POSITA would therefore recognize that a user would store the retrieved file version as the current operative version of the file by using the “Download” command to overwrite the existing current version on the user’s computer with the desired previous version of the file. EX1004, ¶121.

d) [1.2.iii]/[10.2.iii]

Sweat discloses this limitation.

As explained for claim [1.2.ii/10.2.ii], each listed previous file version in the versions tab 508 of Sweat includes information required for recovering that version because it allows a user to select the version to download it from the server 106 to the user’s local computer 104. The remote storage location of server 106 is accessible through network 102, which may comprise the Internet. EX1006, Fig. 1, 3:2-8; EX1007, 3, 53; EX1004, ¶122

e) [1.3]/[10.3]

Sweat teaches this limitation.

Sweat further discloses allowing a selection to preview a previous file version in the presented list of versions and then presenting a presentable representation of

that version. As explained, Sweat presents a list of previous file versions to a user. For example, “[t]o open a version, the user selects the version from a list displayed in versions tab 508 and selects a ‘View in Window’ button.” EX1006, 10:55-58; EX1007, 31, 9. When a user selects the “View in Window” button, “the selected file appears in a separate window” such that the “user sees the contents of the file.” EX1006, 24:13-19; *see also id.*, 24:6-11; EX1007, 31, 9. A POSITA would understand that the presentable representation of each file (i.e., the file presented in the list of Sweat’s versions tab) include information required for presenting the selected previous version by the user’s computer, because the user selects the specific previous version from the list to view the content of that version. EX1004, ¶123.

It would be obvious to a POSITA that a user would implement the teachings of Sweat so as to perform the file preview functionality of Sweat with the current version being accessible. As explained for claim [1.1] (*supra* Section X.A.1.a) a user would have a current version stored in their local computer and it would therefore be accessible. That is, Sweat discloses that a user would download the newest version of a file from the remote server to their local computer, or, alternatively, use the update command to set a file already stored to the computer as the most current version of the file. Such a user with the current version accessible on their computer would desire to restore a previous version of the file to replace the current version,

for example, in the case where the newest version of the file has become corrupted or where desired data has been deleted from the newest version. EX1004, ¶124; EX1005, 4:26-29 (explaining the long standing practice of providing backup files to recover a file that has been corrupted or lost). A POSITA would recognize that such a user would utilize Sweat's previous file version preview functionality (e.g., through selecting the "View in Window" button in the versions tab 508) to determine whether a given previous version is the correct version the user is seeking prior to restoring the file in order to conserve time, computer memory, and processing resources. EX1004. ¶124.

f) [1.4]/[10.4]

Sweat teaches this limitation. Specifically, Sweat allows a user to restore a selected previous version of a file by presenting a selection to the user to download the previous version from the restorable representation of that version (i.e., the previous version displayed in the list of previous versions to the user via versions tab 508). Sweat discloses that "[t]o download a version of the file from the server 106 to the local computer 104, the user selects the desired version and clicks a 'Download' button. Thereafter, the user identifies/selects the target location to save the selected file. The file is then copied to the location specified." EX1006, 10:58-63; EX1007, 31, 9. Sweat explains that when a user clicks the "Download" button, the system "[c]opies the file or folder to the user's local computer" and that the user

can “overwrite files that exist on the user’s local computer, as desired.” EX1006, 5:52-55; EX1007, 3, 103. A POSITA would therefore find it obvious that a user implementing the system of Sweat would use the “Download” button to restore a previous version of a file by copying the file to the user’s local computer and overwriting the current version of the file on the user’s local computer, so as to save the restored file as the new current version of that file. EX1004, ¶¶125-126.

2. Claims 2 & 11

Sweat teaches this limitation. A POSITA would understand that an appropriate application to view a specific file includes an application that created the file. EX1004, ¶128. A POSITA understood that computer files are typically viewable by the same software application that created the file. *Id.* As Sweat explains, “[t]o view project files including drawing files and markup files, a user must install the appropriate viewer to open and view drawing files.” EX1006, 23:38-41; *see also id.* 5:46-48 (opening files “using the application registered to that file type on your computer”), 1:63-67; EX1007, 59, 73. Sweat relates specifically to the management of CAD drawing files and acknowledges that “[t]he use of Computer Aided Design (CAD) application programs is well known in the art.” EX1006, 1:24-26; *see also id.*, Abstract. Because Sweat discloses project members using CAD applications on their local computers to create and edit project files, it would be obvious to a

POSITA that the same CAD application would be used to view previewed files of the same file type generated by that CAD application. EX1004, ¶128.

3. Claim 3

Sweat teaches this limitation. As explained, Sweat discloses using an appropriate application to view a previous file version in order to provide a preview of that version to a user. Sweat discloses that members of a project using its system can be granted different levels of permission by a system administrator. EX1006, 14:43-52; EX1007, 35-36, 38. One such permission setting is “[v]iew permission,” which only “provides the ability to view project files without the ability to add, edit, or delete them.” EX1006, 14:60-61, 13:18-21; EX1007, 35-36, 38; EX1004. Therefore, a POSITA would understand that where a user has been granted view permission by a system administrator and where the user selects to preview a given file, the corresponding application would display the selected file in read-only mode, so as to effect the view permission and prevent the ability to add, edit, or delete viewed files. EX1004, ¶129.

Further, A POSITA understood that a computer system had only three potential access permission modes of operation in which the computer could view data within a file: A POSITA knew that applications had three access permission modes for opening or viewing files: “read-only” (i.e., enabling only viewing of the file), “read/write” (i.e., enabling viewing and editing the file) and “read, write, and

delete” (i.e., enabling viewing, editing, and file deletion). EX1014, 398, 440; EX1004, ¶130. Accordingly, it would be obvious to a POSITA to implement the teachings of Sweat by operating the application used to preview a previous file version in read-only mode because a POSITA understood that there was a finite number of alternative ways a computer system could read data from a file, and the POSITA would have had a reasonable expectation of success of using an appropriate application in read only mode to view the contents of a previous file version. *Id.*

Further, a POSITA would be motivated to operate the viewing application for the previewing step of Sweat in read-only mode to maintain the underlying purpose of Sweat’s file recovery system. Sweat specifically presents a list of previous versions of files to a user and allows a user to select a previous version and to select a previous version to preview. A POSITA would understand that if the viewing application is operated in a mode other than read-only (e.g., read-write mode), this would allow the possibility that the application overwrites data in the previous file version. EX1004, ¶131. The POSITA would understand that allowing writes to a previous file version would defeat the purpose of Sweat’s file recovery system, because writing new data to a previous file version would turn that file into a new file with new data and would thereby prevent a user from accessing and restoring the data of the previous file vile. *Id.* A POSITA would therefore implement Sweat

to preview previous file versions in read-only mode in order to ensure the data integrity of stored previous versions of files. *Id.*

4. Claims 4 & 12

Sweat stores multiple file versions at its server. EX1006, 19:42-46. Sweat presents a list of previous file versions and allows a user to open and view any file version. EX1006, 10:52-57; EX1007, 31; EX1004, ¶132.

5. Claims 5 & 13

Sweat discloses this limitation. Sweat explains that project files include discussion threads containing comments that are attached to a project file. EX1006, 10:34-36; EX1007, 29. Sweats version tab 508 provides a user the ability to edit comments for a given file using the “Edit Comment” button and to review such comments. EX1006, 10:52-55, 10:63-66; EX1007, 31, 9. Further, Sweat discloses a “general tab 502” that “displays information about the selected project file 512” including the location, date the file was created, last modified and accessed. EX1006, 10:10-14; EX1007, 30, 32. This general tab 502 is part of the information for the collection of previous versions because the general tab provides information for the specific file that the listed previous versions relate to. EX1004, ¶133.

6. Claims 6 & 14

a) [6.1]/[14.1]

Sweat teaches this limitation. As explained, Sweat presents to a user a list of previous file versions and allows a user to select a file to preview and subsequently select to retrieve and restore that file to save it as the current version. And as explained above for claim [6.1] for the Schneider reference in Ground 1 (*supra* Section VIII.A.6.a), a POSITA would understand that a user would implement preview steps of a file retrieval system iteratively. Accordingly, a POSITA searching for a specific previous version of a file using the system of Sweat would undergo the process of selecting a previous file version to preview, determining that the previewed version is not the desired version, selecting to close the previewed version, and then iteratively previewing additional previous file versions in the same manner until the user determines that the previewed file version is indeed the file version the user is seeking. EX1004, ¶¶78-79, 134.

b) [6.2]/[14.2]

Sweat teaches this limitation.

As explained above for claim [1.4/10.4] (*supra* Section X.A.1.f), Sweat discloses a process of responsive to a selection to restore a selected previous version, retrieving the restorable representation of the selected version from the remote storage location and storing that version as the current version on the local storage

location, with the selected previous version available from the restorable representation of that version. A POSITA understood that a user, after iteratively previewing multiple previous versions, would implement that same procedure to select, retrieve, and restore a second selected previous version (i.e., where the user determines that the second selected previous version is the version the user is searching for). EX1004, ¶135.

7. Claims 7 & 15

Sweat discloses this limitation. Specifically, Sweat explains that “[f]iles may be automatically compressed when they are being transported to the local computer 104 or the server 106.” EX1006, 19:67-20:2. Accordingly, a POSITA would understand restorable previous file versions stored at the server to be compressed to reduce space of such files in storage. EX1004, ¶136.

8. Claims 8 & 16

Sweat discloses this limitation. As explained, Sweat saves numerous file versions, with new data uploaded to the server upon a user selection to “update” an edited file to the server. EX1006, 5:54-58, 10:4-10, 17:16-21; EX1007, 63, 104. Accordingly, a POSITA would understand that successive versions of a particular file would reflect incremental changes made to that file by a user between such versions. EX1004, ¶137.

9. **Claim 18**

a) **[18.1]**

Sweat discloses this limitation for the same reasons explained for identical preamble element [1.1]. EX1004, ¶¶ 46, 138.

b) **[18.2]**

Sweat discloses this limitation. As explained, Schneider discloses performing incremental backup in response to a user saving or editing a project file. EX1006, 5:56-58, 17:16-19; EX1007, 31. Accordingly, a POSITA would understand that Sweat iteratively transmits such backed-up file versions to Sweat's remote storage location. EX1004, ¶139. It would further be obvious to a POSITA to have the system of Sweat make a determination of the availability of the remote storage location prior to transmitting file back-up versions. *Id.* A POSITA would understand that if the remote storage location is unavailable, for example if the remote storage is faulty, without power, or disconnected from the network, transmission of file back-ups would be unsuccessful and the user should be notified of the problem so that saved files may be backed up using a separate method and so the user can fix the problem with the remote backup location. *Id.*

a) [18.3.i], [18.3.ii], [18.4], [18.5]

Sweat discloses each of limitations [18.3.i], [18.3.ii], [18.4], and [18.5] for the same reasons explained for respective limitations [1.2.i], [1.2.ii], [1.3], and [1.4] (*supra* Sections XA.1.b-c, e-f). EX1004, ¶¶44, 140.

b) [18.3.iii]

Sweat discloses this limitation for the same reasons explained for limitation [1.2.iii]. (*supra* Section X.A.1.d). EX1004, ¶¶44, 141.

10. Claim 19

a) [19.1]

Sweat discloses this limitation for the same reasons explained for identical preamble element [1.1]. EX1004, ¶¶ 46, 142.

b) [19.2]

Sweat discloses this limitation for the same reasons explained for element [1.2.i/10.2.i] (*supra* Section X.A.1.b). Sweat discloses receiving a selection to view a “versions tab” and presenting a list of previous file versions to a user. EX1006, 10:52-57, 24:13-19; EX1007, 31, 9; EX1004, ¶143.

c) [19.3]

Sweat discloses this limitation for the same reasons explained for element [1.3/10.3] (*supra* Section X.A.1.e). Sweat discloses a user selection to preview a

previous file version from the presented list of previous versions by selecting the “View in Window” button. EX1006, 10:55-58; EX1007, 31, 9; EX1004, ¶144

d) [19.4]

Sweat teaches this limitation for the same reasons as explained for element [6.1/14.1] (*supra* Section X.A.6.a). That is, a POSITA would understand that a user attempting to restore a particular file version would utilize Sweat’s file preview functionality iteratively to select and preview successive previous versions of a file until the user finds the correct version the user was searching for. EX1004, ¶145

e) [19.5]

Sweat teaches this limitation for the same reasons as explained for element [1.4/10.4] and [6.2/14.2] (*supra* Sections X.A.1.f, X.A.6.b). A user seeking to restore a particular file version, after finding the correct version through iteratively previewing multiple versions would then select to restore the file. Sweat discloses that after displaying its list of prior file versions, a user selects the “Download” button to save a selected file from the server to copy to the user’s local computer, including overwriting a current version of that file on the user’s computer. EX1006, 5:52-55, 10:58-63; EX1007, 3, 9, 31 103; EX1004, ¶146.

Sweat also discloses optionally decrypting the selected and second selected revision of the files prior to steps (d) and (f), respectively. Specifically, for all communications between the client and the server, Sweat provides the user the

option to perform such data transfer in secure or non-secure mode by specifying whether the data is transferred using the HTTP or HTTPS protocol. EX1006, 8:10-16. A POSITA understood that HTTPS is an encrypted Internet communication protocol and that therefore files transferred between the client computer and remote server using this protocol would be encrypted for data transfer and then decrypted for use of the files (e.g., for viewing files). EX1014, 259-260 (describing HTTPS as a Web protocol “that provides for encryption and transmission through a secure port”). Accordingly, Sweat teaches providing an option to a user to encrypt and decrypt files prior to the file previewing steps of claim 19 steps (d) and (f). EX1004, ¶147.

Sweat also discloses this limitation to the extent it is found that the parts of this claim limitation following the word “optionally” are optional limitations and may be omitted from the claim. EX1004, ¶148.

11. Claims 20 & 21

Sweat discloses this limitation. As explained above for claims 7/15 (*supra* Section X.A.7), Sweat discloses automatically compressing files being transported to the local computer server. EX1006, 19:67-20:2. A POSITA would understand that compression of archived data during transmission would naturally also require decompressing an archived file version in order to display the file. EX1004, ¶150.

**XI. GROUND 4: CLAIMS 9 AND 17 ARE
OBVIOUS OVER SWEAT IN VIEW OF SCHNEIDER**

As explained above, Sweat discloses all limitations of claims 1 and 10 (*supra* Sections X.A.1). Claims 9 and 17 are obvious in further view of the teachings of Schneider. Specifically, Schneider discloses delaying a write operation such that its data backup operations have a substantially imperceptible impact on system performance from user's point of view for the same reasons explained for claims [9/17] in Ground 1 (*supra* Section VIII.A.9.) EX1004, ¶152.

As explained, Sweat performs iterative backup of file versions by transferring saved versions to its remote storage. It would have been obvious to a POSITA to combine such file backup transmissions with Schneider's procedure of delaying transmissions to remote storage and transmitting only during free time in order to reduce or eliminate the impact on the user. EX1004, ¶153. A POSITA would have been motivated to make such a combination because Sweat and Schneider are directed to the same technology, mainly file storage, management, and retrieval systems. *Id.* Further, a POSITA would make such a combination in order to provide the file backup system of Sweat with the benefit explicitly described by Schneider of making its file transmissions impact-free to the user such that the user does not notice any performance degradation. *Id.* A POSITA would have had a reasonable expectation of success in combining the teachings of Sweat and Schneider as it

would be simple for a POSITA to program the system of Sweat to perform transmissions of data to remote storage as a background operation and to delay such transmissions to a time when such operation would go unnoticed by the user. *Id.*

XII. NO OBJECTIVE INDICIA OF NON-OBVIOUSNESS

Petitioner is unaware of any objective indicia of non-obviousness as to any challenged claim.

XIII. CONCLUSION

Petitioners request institution for claims 1-21, on Grounds 1-4 specified in this petition.

Respectfully submitted,

Dated: October 30, 2023

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

**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 13,903 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 2013 in a 14-point Times New Roman font.

Dated: October 30, 2023

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

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

The undersigned certifies that on October 30, 2023, a complete copy of the **Petition for *Inter Partes* Review of U.S. Patent No. 10,585,850 and Exhibits 1001 – 1023** were served via **Express Mail** on the Patent Owner at the following address of record as listed on the USPTO Patent Center:

DATANET, LLC
c/o Messner Reeves LLP
7250 N. 16th Street, Suite 410
Phoenix, AZ 85020

A copy of the Petition for *Inter Partes Review* of U.S. Patent No. 10,585,850 only was served on litigation counsel for Patent Owner via email to:

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