

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

UNIFIED PATENTS, LLC
Petitioner

v.

TOPIA TECHNOLOGY, INC.
Patent Owner

Case No. IPR2022-00782
U.S. Patent 10,067,942

**PETITION FOR *INTER PARTES* REVIEW OF
U.S. PATENT 10,067,942**

Table of Contents

Appendix of Exhibits	iii
I. Introduction.....	1
II. U.S. Patent 10,067,942	2
A. Patent Overview	2
B. Prosecution History	4
C. Priority	6
D. Level of Ordinary Skill	6
E. Claim Construction	6
1. [A] first electronic device . . . configured to . . . (Claims 1, 3, 6)	6
III. Requirements for <i>Inter Partes</i> Review.....	9
A. Certification of Standing	9
B. Identification of Challenge and Requested Relief	9
IV. Grounds of Unpatentability	11
A. Ground 1: Obviousness of Claims 1, 3, 4, 10, 12, and 13 over Strong and Brezak.....	11
1. Overview of Strong	11
2. Overview of Brezak.....	13
3. The Combination of Strong and Brezak.....	14
4. Claim 10	14
5. Claim 1	36
6. Claims 3 and 12	39
7. Claims 4 and 13	40

B.	Ground 2: Obviousness of Claims 5-6 and 14-15 over Strong, Brezak, and Hamidi	42
1.	Overview of Hamidi	42
2.	The Combination of Strong, Brezak, and Hamidi.....	46
3.	Claims 5 and 14	47
4.	Claims 6 and 15	52
C.	Grounds 3 & 4: Obviousness of Claims 1, 3, 5-6, 10, 12, and 14-15 over Hamidi and Ellman and Hamidi, Ellman, and Brezak	53
1.	Overview of Ellman	53
2.	The Combination of Hamidi and Ellman	54
3.	Claim 10	54
4.	Claim 1	73
5.	Claims 3 and 12	76
6.	Claims 5 and 14	77
7.	Claims 6 and 15	78
V.	Discretionary Considerations	79
VI.	Conclusion	81
	Mandatory Notices	82
A.	Real Parties-in-Interest	82
B.	Related Matters.....	82
C.	Lead and Back-Up Counsel.....	83
	Certification of Word Count.....	84
	Certificate of Service.....	85

APPENDIX OF EXHIBITS

Exhibit	Description
1001	U.S. Patent 10,067,942 (“’942 Patent”)
1002	File History of U.S. Patent 10,067,942 (“File History”)
1003	International Application WO 2009/062182 file wrapper, compiled via https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2009062182 on March 31, 2022
1004	Appendix of Challenged Claims
1005	Declaration of Darrell Long, Ph.D. (“Long Decl.”)
1006	U.S. Patent Publication 2006/0101064 to Strong <i>et al.</i> (“Strong”)
1007	U.S. Patent Publication 2006/0242206 to Brezak <i>et al.</i> (“Brezak”)
1008	U.S. Patent Publication 2004/0267697 to Hamidi <i>et al.</i> (“Hamidi”)
1009	Int’l Publication WO 2000/075781 to Ellman <i>et al.</i> (“Ellman”)
1010	File History of U.S. Patent 9,143,561
1011	Declaration of Kevin Jakel (“Jakel Decl.”)
1012	Microsoft Corporation, <i>Excerpts from Microsoft Computer Dictionary</i> , 5th Ed. Microsoft Press (2002) (“Microsoft”)
1013	Scott Lewandowski, <i>Frameworks for Component-Based Client/Server Computing</i> , ACM Computing Surveys 30:1 (Mar. 1998) (“Lewandowski”)
1014	Gaurav Banga and Peter Druschel, <i>Measuring the Capacity of a Web Server</i> , Proc. of the USENIX Symposium on Internet Technologies and Systems, Monterey, California, (Dec. 1997) (“Banga”)
1015	Elsevier B.V., <i>Three-Way Handshake</i> , https://www.sciencedirect.com/topics/computer-science/three-way-handshake , accessed March 18, 2022 (“ScienceDirect”)
1016	Vannevar Bush, <i>As We May Think</i> , The Atlantic Monthly (July 1945), reproduced by Denys Duchier (April 1994) (“Bush”)
1017	Barry M. Leiner <i>et al.</i> , <i>A Brief History of the Internet</i> , Internet Society

Exhibit	Description
	(1997) (“Leiner”)
1018	Andrew D. Birrell <i>et al.</i> , <i>Grapevine: An Exercise in Distributed Computing</i> , Comm. of the ACM 25:4 (Apr. 1982) (“Birrell”)
1019	E. James Whitehead, Jr. and Meredith Wiggins, <i>WebDAV: IETF Standard for Collaborative Authoring on the Web</i> , IEEE Internet Computing (Sep.-Oct., 1998) (“Whitehead ’98”)
1020	Jim Whitehead, <i>WebDAV: Versatile Collaboration Multiprotocol</i> , IEEE Internet Computing (Feb. 2005) (“Whitehead ’05”)
1021	Fredj Dridi and Gustaf Neumann, <i>How to Implement Web-based Groupware Systems based on WebDAV</i> , Proc. of WETICE 99, IEEE 8th Intl. Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, (June 1999) (“Dridi”)
1022	U.S. Patent Publication 2003/0061350 to Masuoka <i>et al.</i> (“Masuoka”)
1023	Int’l Publication WO 03/032126 to Owen <i>et al.</i> (“Owen”)
1024	David Rasch and Randal Burns, <i>In-Place Rsync: File Synchronization for Mobile and Wireless Devices</i> , Proc. of the FREENIX rack: 2003 USENIX Annual Technical Conf. (Jun. 2003) (“Rasch”)
1025	Utku Irmak <i>et al.</i> , <i>Improved Single-Round Protocols for Remote File Synchronization</i> , Proc. IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies pp. 1665-1676 (2005), available at https://ieeexplore.ieee.org/abstract/document/1498448 , accessed via http://users.eecs.northwestern.edu/~peters/references/FileSynchronizationSuel05.pdf on March 30, 2022

I. INTRODUCTION

Unified Patents, LLC (“Unified” or “Petitioner”) requests *inter partes review* of Claims 1, 3-6, 10, and 12-15 (the “Challenged Claims”) of U.S. Patent 10,067,942 (the “’942 Patent, **EX1001**). On information and belief, the ’942 Patent is owned by Topia Technology, Inc. (“Topia” or “Patent Owner”).

II. U.S. PATENT 10,067,942

A. Patent Overview

The '942 Patent relates to sharing files between electronic devices, such as a server and user devices like personal computers or workstations. *'942 Patent* (EX1001), 1:18-20. For example, user devices communicate with a server over a network. *Id.*, 7:26-32. The devices “cooperate in two-way communication with the server” for sending and retrieving information. *Id.*, 7:49-56.

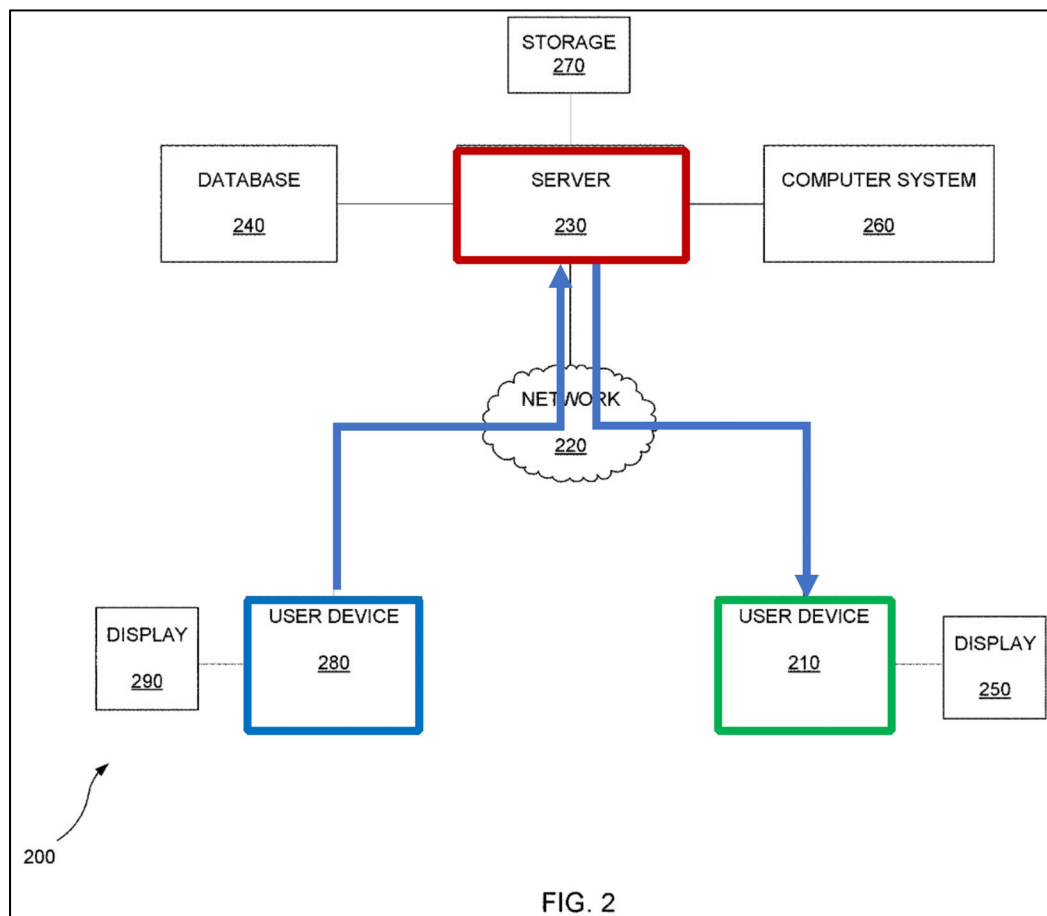
Applications on the user devices provide clients that permit “all files that the user desires to be synched or shared” to be uploaded to a server. *Id.*, 8:2-19. The server stores the files “and/or” transfers them to other clients. *Id.*, 8:19-23.

The server receives files from a client based on a triggering event, such as the user saving the file, the elapsing of predetermined amount of time, or the re-initiation of communication between the client and server. *Id.*, 8:37-44. The server polls other user devices and transfers the file if the given device in is communication with the server. *Id.*, 8:50-59. The '942 Patent describes that synchronization may apply in the “reverse direction,” such that multiple user devices can exchange files to one another through the server. *Id.*, 8:63-67.

The '942 Patent describes drawbacks in prior art systems including manual transfer requirements, redundancy, error-prone environments, and unnecessary complexity. *Id.*, 1:33-65. Additionally, the '942 Patent sought to improve systems

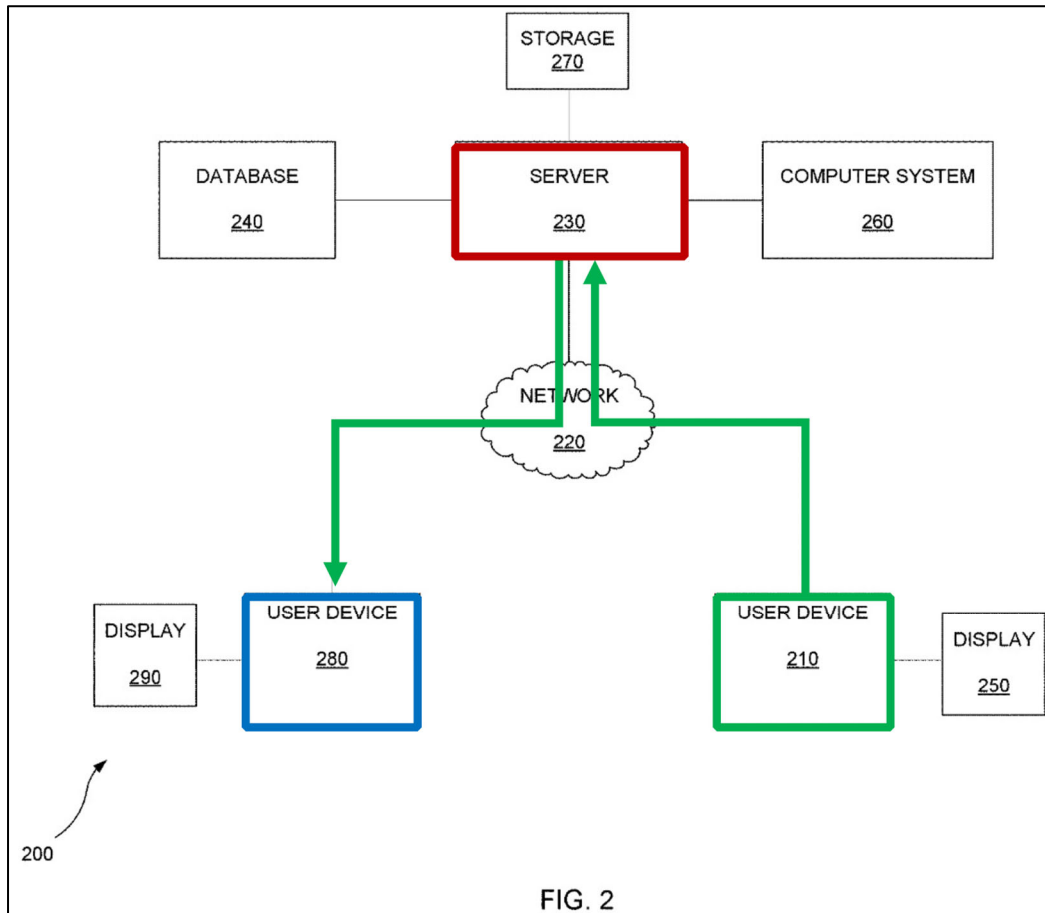
that required clients to be continuously connected for file sharing. *Id.*, 2:27-47.

An Appendix of Challenged Claims is provided as **Exhibit 1004**. There are two claim sets: method claims (Claim 10 and its dependents), and nearly identical system claims (Claim 1 and its dependents). Elements 10.1-10.3 (and elements 1.1.1-1.1.3) describe a **first device**, such as a server, receiving a modified file copy from a **second device**, determining whether the first device is in communication with a **third device**, and automatically sending the file to the third device upon receiving the file and determining that the first and third devices are in communication:



Id., Fig. 2 (annotated); *see also* Fig. 3. Elements 10.4-10.6 (and elements 1.1.4-

1.1.6) describe a second file copy going in the opposite direction, from the third device to the first device and then on to the second device:



Id., Fig. 2 (annotated); *see also* Fig. 3. In elements 10.7 and 10.8 (and elements 1.2-1.3), the first and second files, respectively, are caused to replace existing versions of those files.

B. Prosecution History

During prosecution (EX1002), the original claims were rejected for double patenting, claiming patent-ineligible subject matter, and as anticipated by a prior art reference, Mohanty. *File History* (EX1002), 20 (original claims), 107-21 (non-final

rejection).

The claims were amended to add two steps: (1) determining whether a first device is in communication with the third device and (2) automatically sending, via the first application, a modified first electronic file copy to a third application at a third device responsive to the determination that the first electronic device is in communication with the third electronic device and responsive to receiving the modified first electronic file copy from the second electronic device. *Id.*, 144-48.

An examiner's amendment was discussed during an examiner interview. *Id.*, 174, 184. The examiner's amendment recited the same concepts already claimed, but for a **second file copy** going from the third device to the second device through the first device. *Id.*, 175-181. The examiner identified these limitations (now limitations 1.1.5, 1.3, 10.5, and 10.8) as the reasons for allowance. *Id.*, 182.

Notably, an international preliminary report on patentability found claims reciting such bi-directionality of file synchronization not to be novel. *PCT Counterpart* (EX1003), 51, 56.¹ Further, the grounds herein demonstrate that automatic, bi-directional file sharing among a user's devices existed before 2007.

¹ It does not appear that this PCT analysis was provided during prosecution of the '942 Patent or its continuation parent. *See* EX1002, EX1010, generally.

C. Priority

The '942 Patent was filed September 21, 2015. The earliest possible priority date is November 9, 2007. Because the references cited were published over one year before this date, Petitioner does not challenge priority at this time.

D. Level of Ordinary Skill

A person of ordinary skill in the art (“POSA”) would have been one having, by November 9, 2007: (1) an undergraduate degree in computer science, computer engineering, or closely-related field, or similar advanced post-graduate education; and (2) two or more years of experience with file sharing systems, with more experience substituting for less education and vice versa. *Long Decl.* (EX1005), ¶¶ 34-36.

E. Claim Construction

Claims in IPR proceedings are construed using the standards applied by Article III courts. 37 C.F.R. § 42.100(b). Construction is unnecessary because the Challenged Claims are obvious under any reasonable construction; however, Petitioner addresses certain limitations below.

1. [A] first electronic device . . . configured to . . . (Claims 1, 3, 6)

Some limitations recite a first device that is “configured to” perform certain functions, such as receiving files, determining whether it is in communication with another device, sending files, and storing files. Since “means” is not used, there is a

rebuttable presumption against applying 35 U.S.C. § 112 ¶ 6 (“Paragraph 6”) to these limitations. *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1346 (Fed. Cir. 2015) (*en banc*). Unless Topia presents evidence showing that the claims lack sufficient structure, Paragraph 6 should not apply. *Dyfan, LLC v. Target Corp.*, 28 F.4th 1360 (Fed. Cir. 2022).

The Board may reach obviousness without deciding whether Paragraph 6 applies or if Paragraph 6 applies and the claims are indefinite. *Intel Corp. v. Qualcomm Inc.*, 21 F.4th 801, 813-14 (Fed. Cir. 2021) (obviousness determination of indefinite means-plus-function claims possible, as indefiniteness precludes a patentability finding “only when the indefiniteness renders it logically impossible” to assess obviousness); *see also Huawei Techs. Co. v. WSOU Invs., LLC*, IPR2021-00226, Paper 10, 22 (P.T.A.B., June 10, 2021) (construction of potential means-plus-function terms unnecessary at the institution stage).² And Paragraph 6 is inapplicable if the limitations have sufficient structure. Here, the claims recite an algorithm that comprises receiving data, sending data, determining if a device is

² Here, the prior art teachings are consistent with the specification’s disclosure for these claims, and the general public has an interest in the determination of patentability of the method claims, to which Paragraph 6 does not apply. Therefore, Petitioner requests IPR even if some claims are indefinite.

communication with others, and storing data—all features of generic computers. *See In re Katz Interactive Call Processing Patent Litig.*, 639 F.3d 1303, 1316 (Fed. Cir. 2011) (algorithms unnecessary for functions like receiving, storing, and processing).

III. REQUIREMENTS FOR *INTER PARTES REVIEW*

A. Certification of Standing

The '942 Patent is available for IPR, and Petitioner is not barred or estopped from seeking IPR of the Challenged Claims. 37 C.F.R. § 42.104(a).

B. Identification of Challenge and Requested Relief

In view of the prior art, evidence, and analysis discussed herein, Petitioner requests IPR of Claims 1, 3-6, 10, and 12-15 of the '942 Patent under pre-AIA 35 U.S.C. § 103. 37 C.F.R. §§ 42.104(b)(1) and 42.22.

Grounds:

Ground	Claims	Challenge
1	1, 3-4, 10, 12-13	Obviousness over Strong ³ and Brezak ⁴
2	5-6, 14-15	Obviousness over Strong, Brezak, and Hamidi ⁵
3	1, 3, 5-6, 10, 12, 14-15	Obviousness over Hamidi and Ellman ⁶
4	1, 3, 5-6, 10, 12, 14-15	Obviousness over Hamidi, Ellman, and Brezak

³ U.S. Patent Publication 2006/0101064, **EX1006**.

⁴ U.S. Patent Publication 2006/0242206, **EX1007**.

⁵ U.S. Patent Publication 2004/0267697, **EX1008**.

⁶ Int'l Publication WO 2000/075781, **EX1009**.

Section II, *supra*, addresses claim construction. 37 C.F.R. § 42.104(b)(3).
Section IV, *infra*, identifies where each element of the Challenged Claims is found in the prior art, citing relevant evidence listed in the Appendix of Exhibits, including support from Petitioner's expert, Dr. Long (**EX1005**). 37 C.F.R. § 42.104(b)(4)-(5).

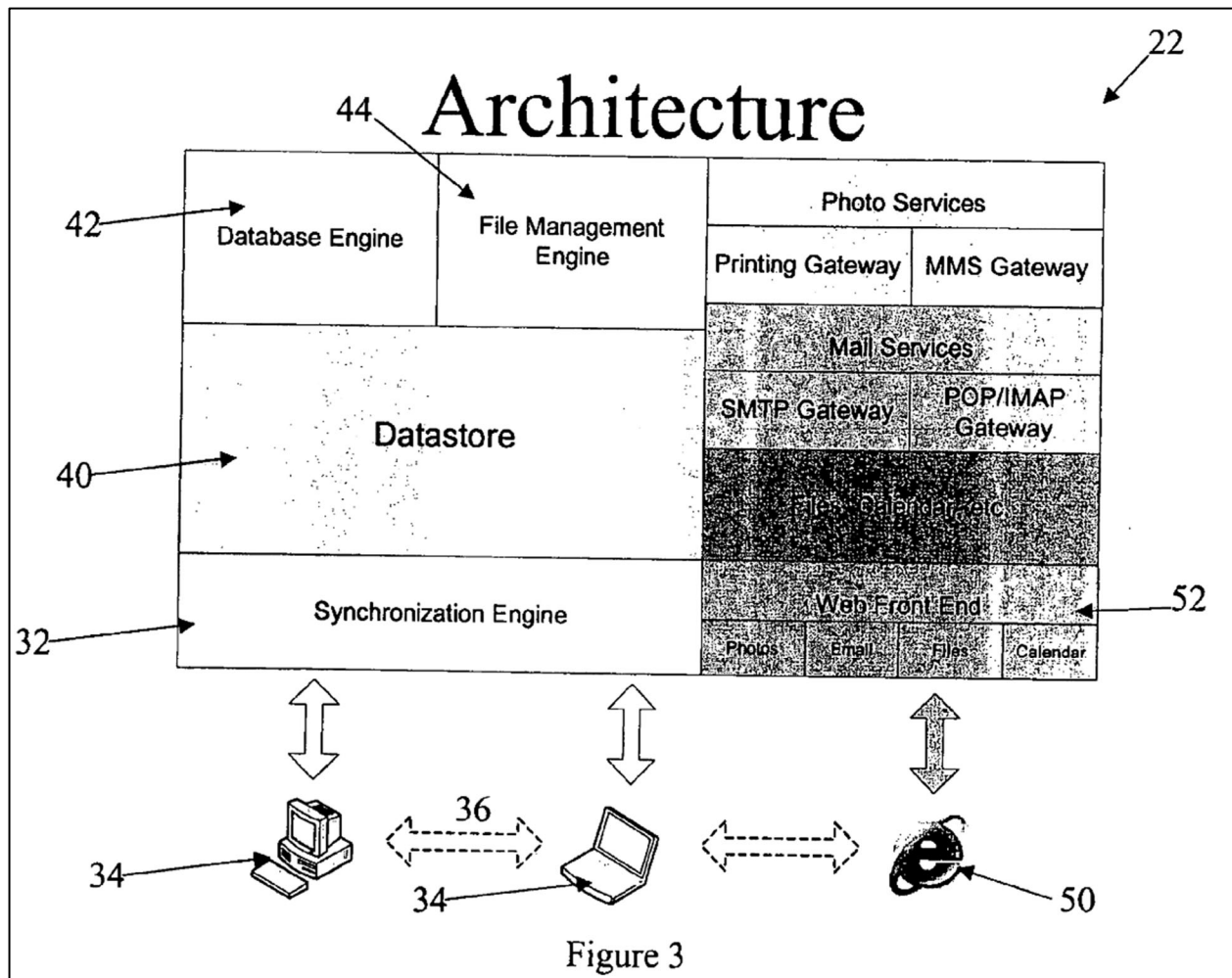
IV. GROUNDS OF UNPATENTABILITY

A. Ground 1: Obviousness of Claims 1, 3, 4, 10, 12, and 13 over Strong and Brezak

1. *Overview of Strong*

Strong was filed November 8, 2005, claims priority to a provisional application filed November 8, 2004, and was published May 11, 2006; therefore, Strong is prior art under 35 U.S.C. §§ 102(a), (b), and (e).

Strong describes a file synchronization system in which one or more servers maintains copies of a user's files and shares them between the user's devices. *Strong* (EX1006), Abstract; *see also* [0008]; *see also* [0023] (embodiments may use one server). A server synchronization engine sends real-time updates to client devices when one updates a file. *Id.*, Abstract.



Id., Fig. 3; *see also* Figs. 1-2. Changes may be synchronized using update messages or by sending the entire data object, such as a picture file or email message. *Id.*, [0027]; *see also* [0036] (disclosure not limited to photographs); claims 6-7.

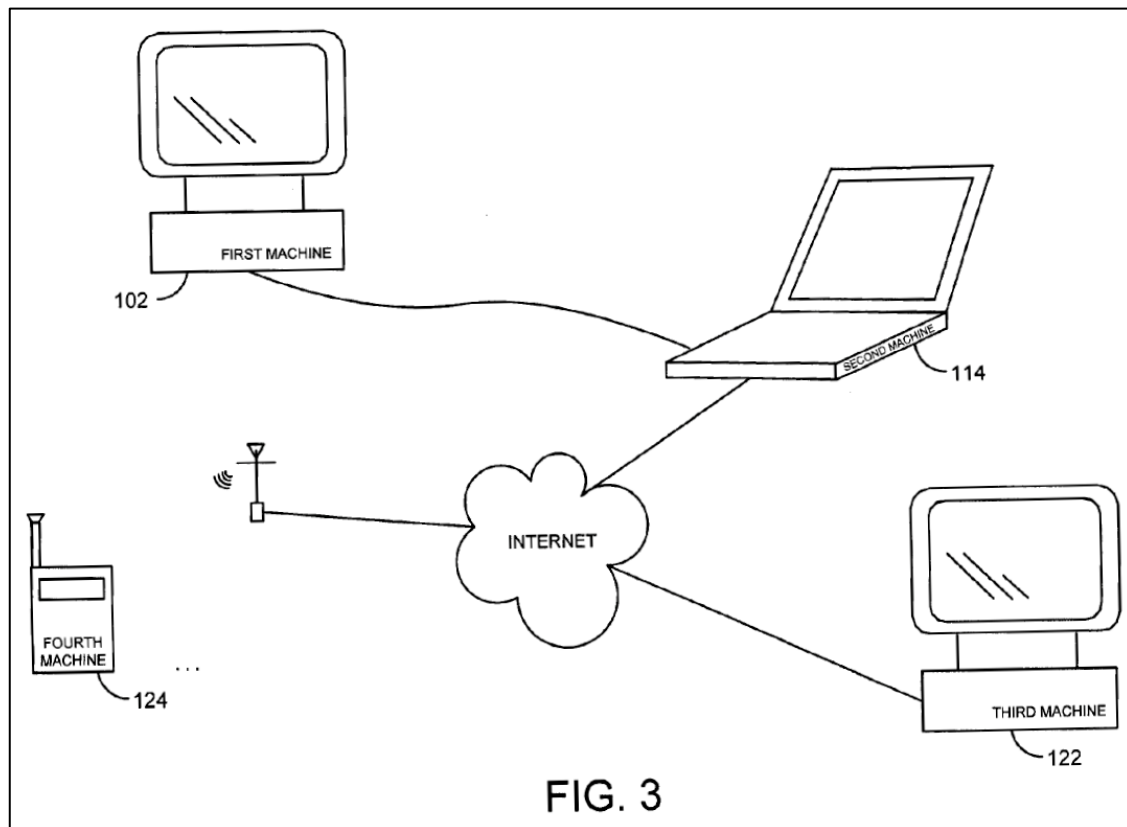
Strong is analogous art to the '942 Patent. *Long Decl.* (EX1005), ¶ 54. Strong is within the field of endeavor because it relates to the sharing of electronic files among electronic devices. *Strong* (EX1006), Abstract, [0002]-[0003]. Further, Strong is reasonably pertinent to multiple problems concerning the inventors, such as the efficiency of file sharing; *id.*, [0003]-[0004]; and effecting synchronization

automatically and without a continuous connection. *Id.*, [0004], [0032]-[0033].

2. *Overview of Brezak*

Brezak was filed April 22, 2005 and published October 26, 2006; therefore, Brezak is prior art under 35 U.S.C. §§ 102(a), (b), and (e).

Brezak relates to systems and methods for sharing content between a user's machines in a peer-to-peer environment. *Brezak* (EX1007), Abstract. The user's machines include a "sync engine" that allows users to designate files for sharing to selected machines "on an automatically consistent basis." *Id.* Machines are connected physically or wirelessly:



Id., Fig. 3; *see also* [0016].

Brezak is analogous art to the '942 Patent. *Long Decl.* (EX1005), ¶ 64. Brezak is within the field of endeavor because it relates to the sharing of electronic files. *Brezak* (EX1007), Abstract, [0003]. Further, Brezak addresses at least one problem concerning the inventors: the automatic sharing of up-to-date versions of files, even when devices are not continuously online. *Id.*, [0004], [0006]; *see also* Abstract, [0016].

3. *The Combination of Strong and Brezak*

Brezak is cited for its disclosure related to detecting the connectivity status of another machine before transmitting files. *Id.*, [0015], [0023]-[0024], Fig. 4. Combined with Brezak, Strong's system performs an additional step in which its server first determines whether it is in communication with a receiving client device before sending files to it. Sec. IV.A.4 (10.2-10.3, 10.5-10.6); Sec. IV.A.5 (1.1.2-1.1.3, 1.1.5-1.1.6), *infra*.

Below, Petitioner discusses the independent claims separately, beginning with the broader method claim, and then addresses like dependent claims together.

4. *Claim 10*

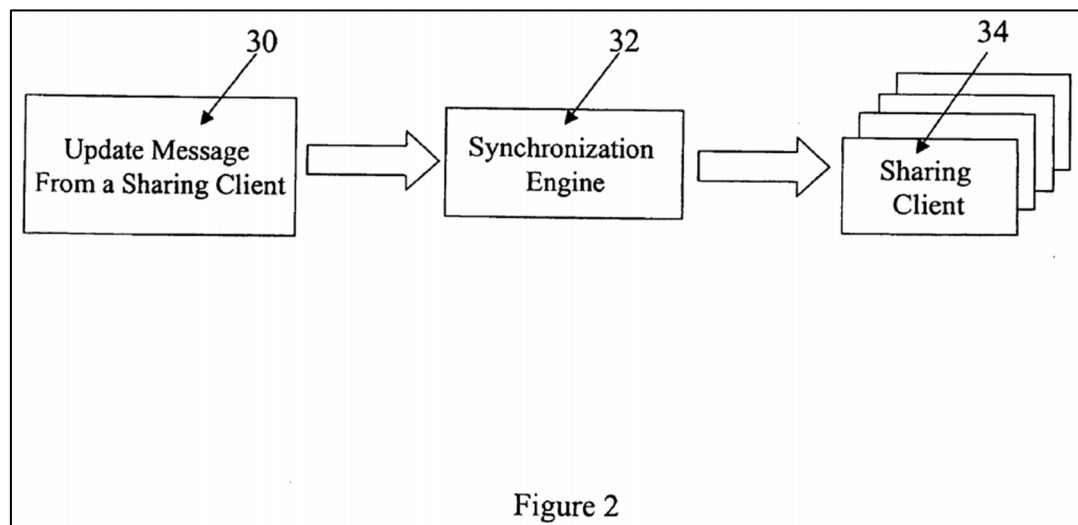
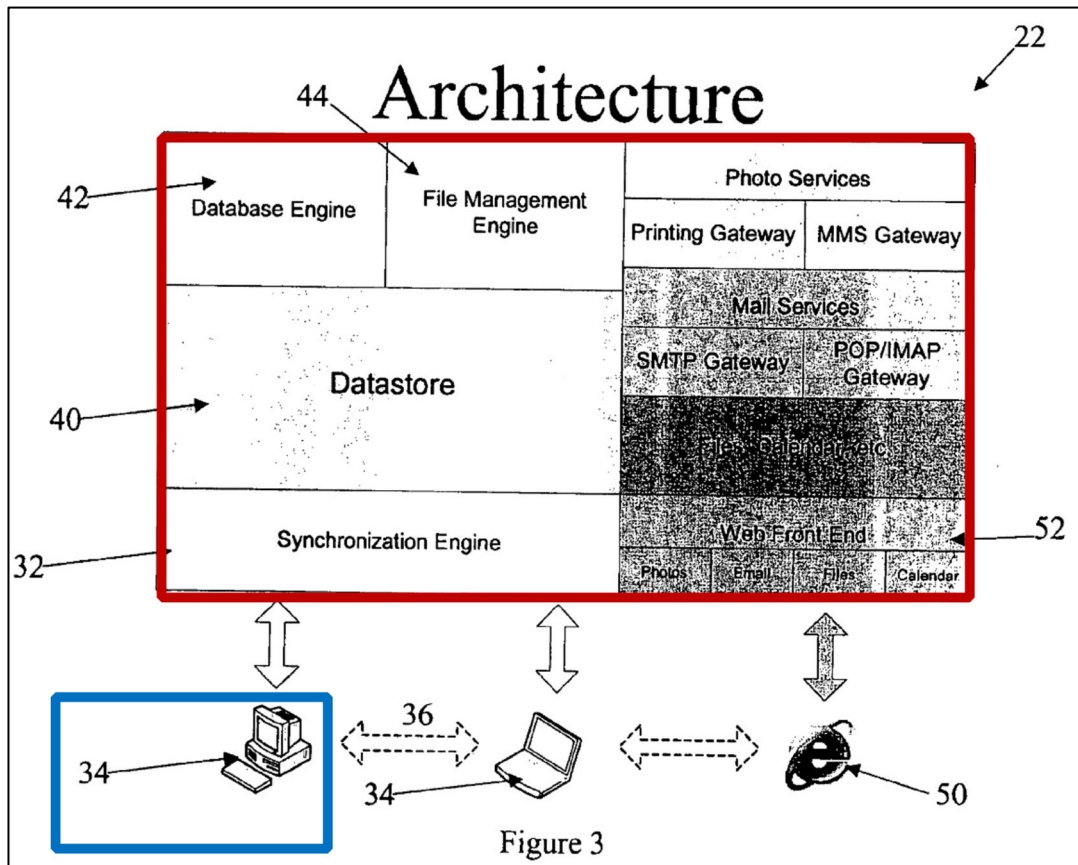
[10P]. A method comprising:

Strong discloses “**methods** of synchronizing data.” *Strong* (EX1006), [0002]; *see also* Abstract.

[10.1] receiving, via a first application at a first electronic device, a copy of a modified first electronic file from a second application at a second electronic device associated with a user, wherein the modified first electronic file copy is automatically received from the second application responsive to the user modifying a content of the first electronic file;

Strong discloses this limitation; alternatively, any differences would have been obvious variations to a POSA. Strong discloses *receiving,*⁷ *via a **synchronization engine at a server** (i.e., receiving, via a first application at a *first electronic device*) modified versions of data objects, such as picture files shared across a user's devices and on the server (i.e., a copy of a modified first electronic file) **from a sharing client running on a user's device, such as a desktop** (i.e., from a *second application at second electronic device* associated with a user), automatically, in real time upon some event, such as a user editing the or an autosave (i.e., wherein the modified first electronic file copy is automatically received from the second application responsive to the user modifying a content of the first electronic file):*

⁷ Unless otherwise indicated, *italics* signify claim language, and **bold** emphasis in quotations has been added.



Strong (EX1006), Fig. 3 (annotated), Fig. 2; see also [0026]-[0027].⁸

⁸ While the desktop computer is annotated here, any device running sharing client

Server 22 includes a synchronization engine (i.e., *a first application at a first electronic device*), which receives updated versions of data objects from an application running on one of the user's devices, individually and collectively referred to as sharing clients 34:

Referring to FIG. 2, **changes or updates 30 from a sharing client 34 will be sent to the synchronization engine 32** which will forward or update a copy of a data object on each of the clients listed as part of the plurality of sharing clients 34 in order to synchronize that data object with other sharing clients. **Each client preferably has the ability to communicate with a server and have the appropriate software running thereon.**

Id., [0026]; *see also* [0029] (“[T]he server platform may be at least one server 22 which has a synchronization engine 32 which communicates with sharing clients 34 running on devices which may be but are not limited to desktop computers and laptop computers.”); *see also* [0021] (a client is a program that runs on a user's device, such as a personal computer, personal digital assistant, or a workstation).

The entire or “actual data object in its modified form” may be shared; therefore, Strong discloses the server receiving *a copy of a modified first electronic file*. *Id.*, [0027] (“One way of representing a modification to an existing data object

34 may be mapped as the second or third electronic devices.

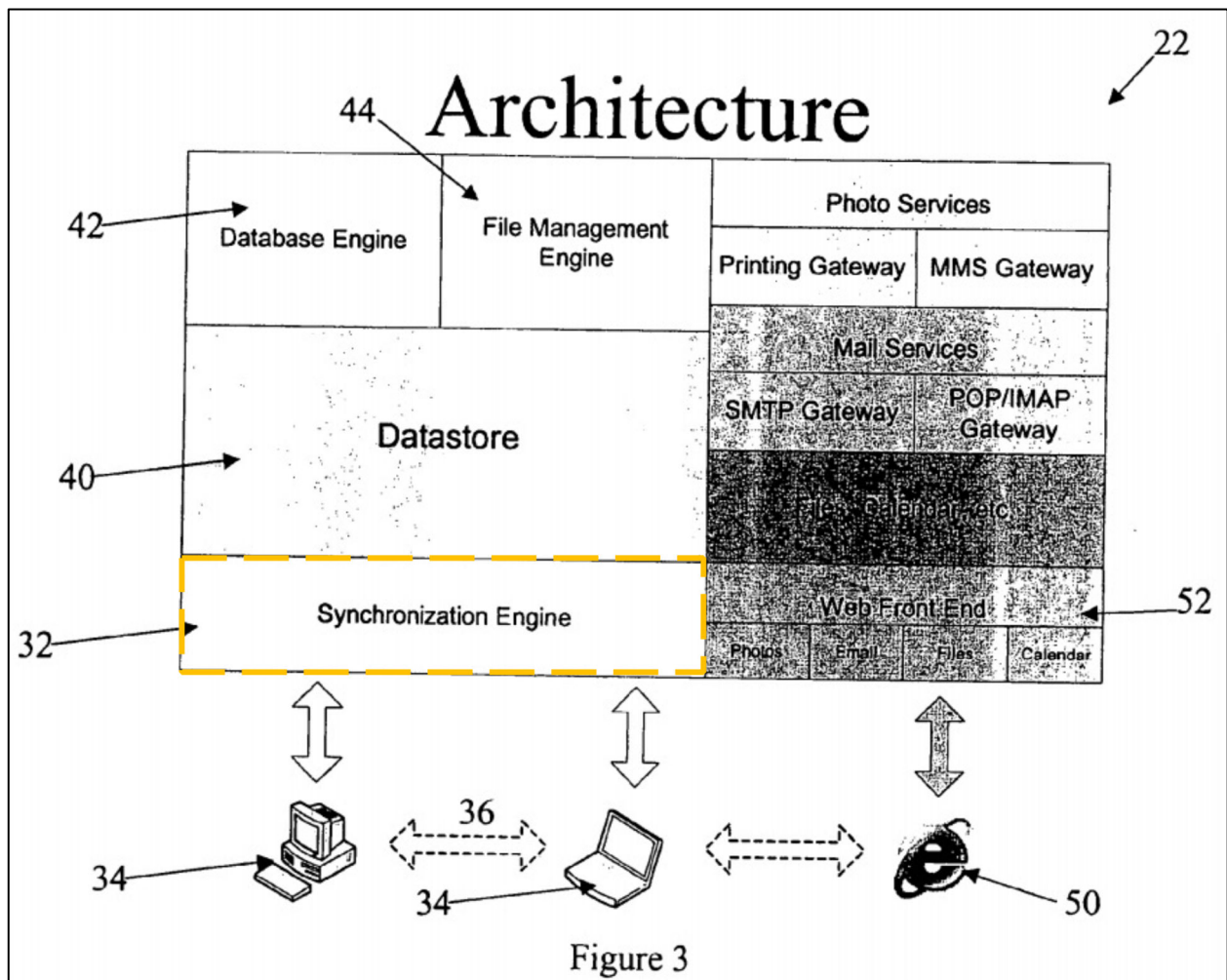
or the creation of a new data object is to include the actual data object in its modified form. Thus **in some embodiments, updates are represented by sending a new copy of the entire object.**”); *see also* [0028] (“This two way synchronization may occur in real-time which, for example and not limitation, may occur with the files being updated whenever an autosave occurs to the data or file being edited. **The autosaved version may be updated to the server having the synchronization engine** and then pushed out or cascaded to the sharing clients.”). For example, a data object may be a picture file or email message. *Id.*, claims 6-7; *see also* [0008], [0031] (describing different types of files managed in Strong’s system); *Long Decl.* (EX1005), ¶ 49.

Further, Strong discloses that *the modified first electronic file copy is automatically received from the second application responsive to the user modifying a content of the first electronic file.* Strong (EX1006), [0037] (“**When a user with a sharing client 34 makes changes to albums or files on a local computer, the changes are synchronized automatically across all devices.**”); Fig. 4 (“[c]hanges to albums **synchronized automatically** across all devices.”). In the ’942 Patent, transfers occur based on a trigger event, such as the user saving the file, the elapsing of a period of time, or the re-initiation of communication between a device and the server. ’942 Patent (EX1001), 8:35-44. Likewise, Strong discloses that the file is *automatically received by the server responsive to the user modifying a content of*

the first electronic file because it discloses that the synchronization occurs in “real time,” and “may occur with the . . . file being updated.” *Strong* (EX1006), [0028]; *see also* [0022] (“If the user makes a change on one computer to the database or a file, the change is synchronized to the server and synchronized in real-time to the user's other computers.”). As another example, *Strong* discloses that the file may be sent with every autosave of the file and that the application auto-saves changes as a user makes edits. *Id.*, [0028]; [0040].

The *second electronic device* is *associated with the user* in that it belongs to the user or is accessed by logging into an application running on a computer. *Id.*, [0022]; *see also* [0040] (“[C]hanges will be saved to the datastore and synchronized **to other computers belonging to the user who made the changes.**”); [0030]-[0031] (describing the sharing clients 34); Figs. 1-3 (illustrating user devices communicating with the server). Each sharing client 34 preferably runs as an *application* installed on the device. *Id.*, [0008]; *see also* [0043] (describing techniques for saving a user’s application settings); [0036] (referencing “the application of sharing client 34”); *see also* [0022], [0031] (a user may access the system via a web browser application).

A POSA would have appreciated that the synchronization engine is a *first application* because *Strong* depicts it as an independent module that is part of a collection of server applications:



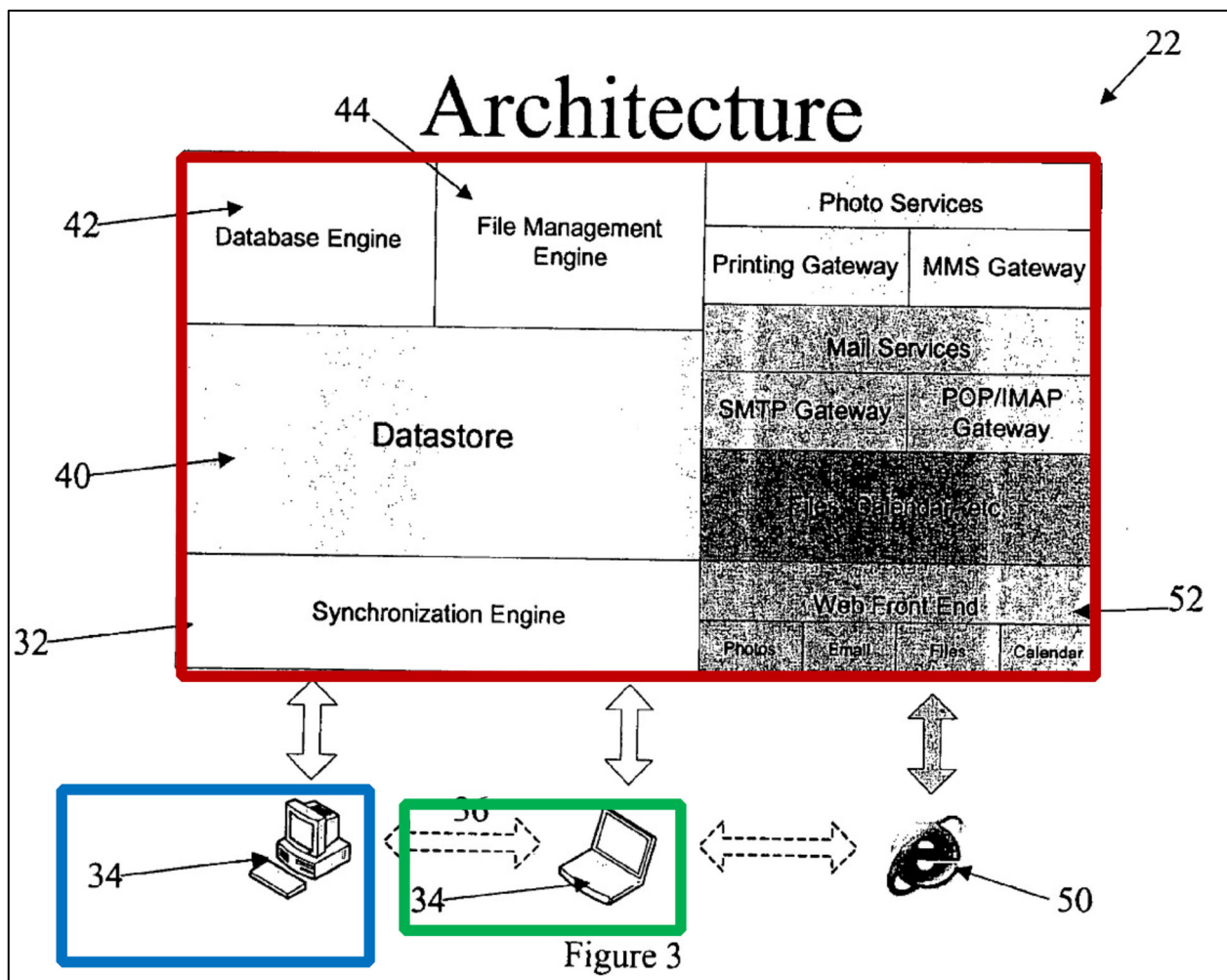
Id., Fig. 3 (annotated); *see also Long Decl.* (EX1005), ¶ 55. However, even if the synchronization engine were not a standalone application, it would be the component of a server program or application that includes synchronization functionality. *Strong* (EX1006), [0020] (describing server software); *see also Long Decl.* (EX1005), ¶ 55. Indeed, “engine” was a known software term with different acceptable uses by 2007. *Long Decl.* (EX1005), ¶¶ 55-56 (citing EX1012). A POSA would have recognized based on context, or at least have found obvious, that Strong’s synchronization engine would have been a standalone application or a part

of a server application with other functionality. *Id.*, ¶ 56 (citing *Strong* at [0025], Abstract). Thus, given Strong’s description of the synchronization engine as software and a POSA’s knowledge regarding server programs, it at least would have been obvious to implement the synchronization engine as *a first application* of server 22. *Id.*

[10.2] determining whether the first electronic device is in communication with a third electronic device;

This limitation is taught by the combination of Strong and Brezak. The ’942 Patent discloses an embodiment in which a server application polls a client device to determine whether it is in communication with a server. ’942 Patent (EX1001), 8:50-54. Strong discloses that certain operations in its system occur only if *the first electronic device (server 22) is in communication with a third electronic device* (another *device running sharing client 34*, such as a *laptop*). *Strong* (EX1006), [0032] (“When a continuous connection is not available, **then the next time the user is connected**, the changes will be synchronized.”); *see also* [0033] (describing that clients have an “on-line mode” in which they are configured to receive real-time updates, contrasted with an “off-line” mode); [0034] (describing potential ways of identifying files for updating “[w]hen communications are first established between the server 22” and the sharing client); [0023]-[0025] (clients can communicate with each other through the server by “hook[ing]” into the appropriate network).

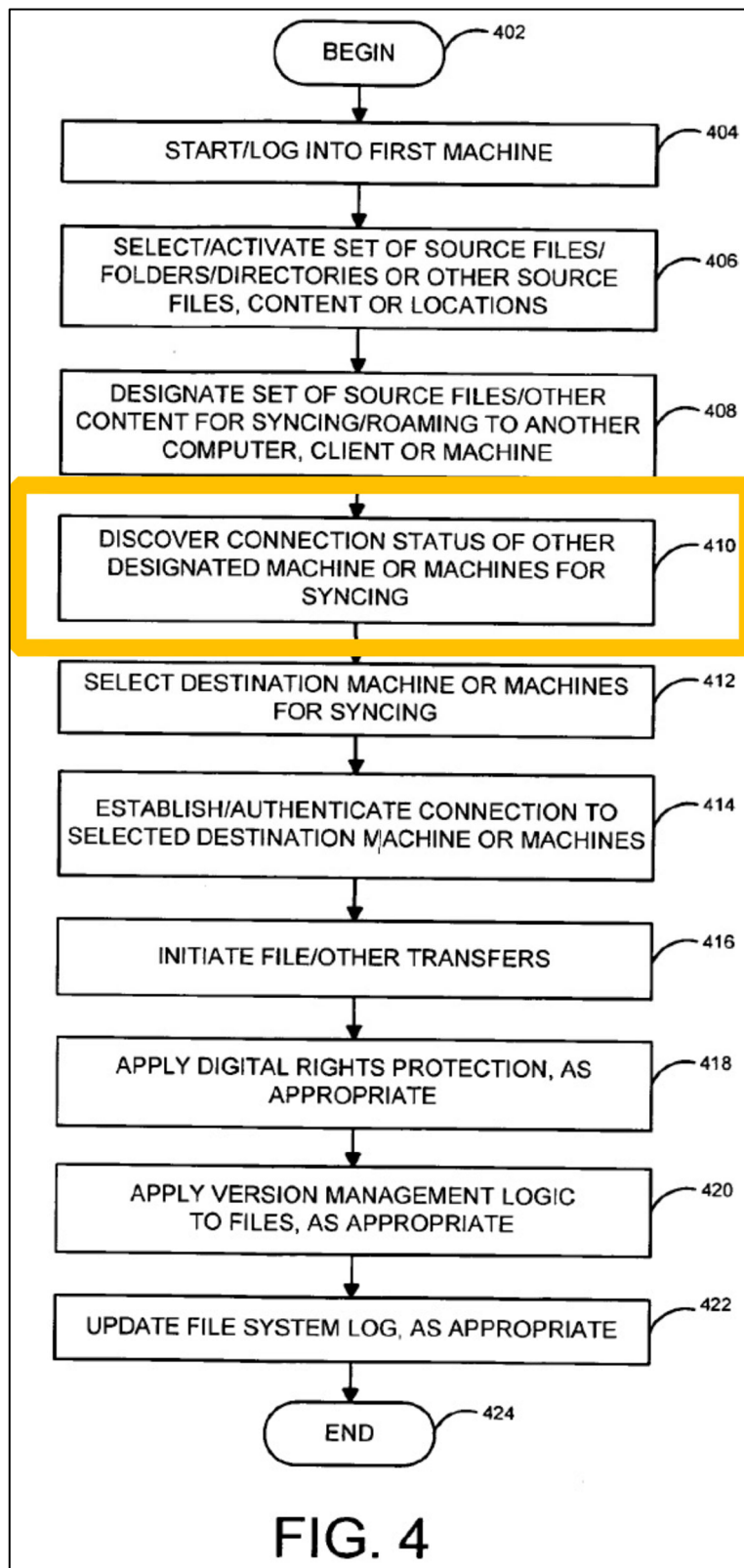
Therefore, Strong at least implies that this determination step occurs, as a POSA would have appreciated given the nature of server-client communications, in which servers “listened” for potential clients and accepted connections and serviced requests. *Long Decl.* (EX1005), ¶¶ 65-66; *see also* ¶¶ 40-41 (describing client-server relationships).



Strong (EX1006), Fig. 3 (annotating a *third electronic device*, green).

While Strong’s language implies this limitation, Brezak expressly teaches it in a similar context. Brezak describes the use of a “sync engine” to synchronize file

changes between different machines, where the sending machine first “**detect[s] the connection status**” of another to “**determine those occasions when those machines enjoy connectivity between each other or other participating hardware**” (i.e., *determining whether a first electronic device is in communication with another electronic device*), via wired connections or wireless communication channels, or by tunneling through firewall resources. *Brezak* (EX1007), [0015]; *see also* [0023] (“In step 410, the network connection status of those designated machine or machines may be automatically discovered, for instance by sync engine 108 or other logic.”).



Id., Fig. 4 (annotated).

The combination of Strong and Brezak would have taught *determining* (e.g., by detecting the connection status of another device) *whether the first electronic device (server 22) is in communication with a third electronic device* (e.g., one of the other devices running sharing client 34, such as a user's laptop computer). *Long Decl.* (EX1005), ¶ 67.

A POSA would have been motivated to modify Strong's server to detect the connection status of a receiving client machine, as taught by Brezak, for at least three reasons. First, Strong provides an express teaching, suggestion, or motivation to combine through its disclosure of having a server perform synchronization the next time a user is connected; Brezak simply provides a more specific teaching on how the server would determine the connection status between itself and a user's devices. *Id.*, ¶ 68.

Second, the combination would have been a matter of combining prior art elements according to known methods to yield predictable results, namely, combining Strong's disclosure of synchronizing the "next time the user is connected" with Brezak's teaching regarding *determining* the occasion when machines are connected. *Id.*, ¶ 69.

Third, incorporation of the determination step would have merely involved using a known technique (e.g., Brezak's determining the connection status of receiving machines) to improve Strong's similar synchronization system in the same

way, by assessing the connectivity status of sharing clients before attempting to push files to those devices, thereby avoiding unnecessary pushing attempts. *Id.*, ¶ 70.

Additionally, a POSA would have had a reasonable expectation of success in making the combination because Brezak and Strong describe similar systems (computing devices with synchronization engines) with similar goals (synchronizing files between devices), and servers were known to determine whether a connection with a client existed, such as by waiting for an acknowledgment, before providing a service. *Id.*, ¶¶ 71, 41. Further, because Strong already discloses that the server maintains a list of the user's clients, incorporating Brezak's teachings of the *determining* step into Strong's system would have required at most minor modifications in software and would have yielded predictable results consistent with polling techniques known well before 2007. *Id.*

Further, although Brezak describes peer-to-peer systems, a POSA would have appreciated that its teaching related to determining the connection status of a receiving device was applicable to client-server contexts, especially given Strong's own disclosure of performing synchronization when the designated sharing clients next becomes available. *Id.*, ¶ 72; *see also Strong* (EX1006), [0032]. Indeed, "peer-to-peer" refers to systems where devices normally acting as clients take turns acting as servers and clients to one another, so Brezak's teachings regarding determining a connectivity status were not limited to peer-to-peer arrangements; thus, modifying

Strong's server-client system to include this step would have been consistent with Brezak's teachings. *Long Decl.* (EX1005), ¶¶ 40, 72.

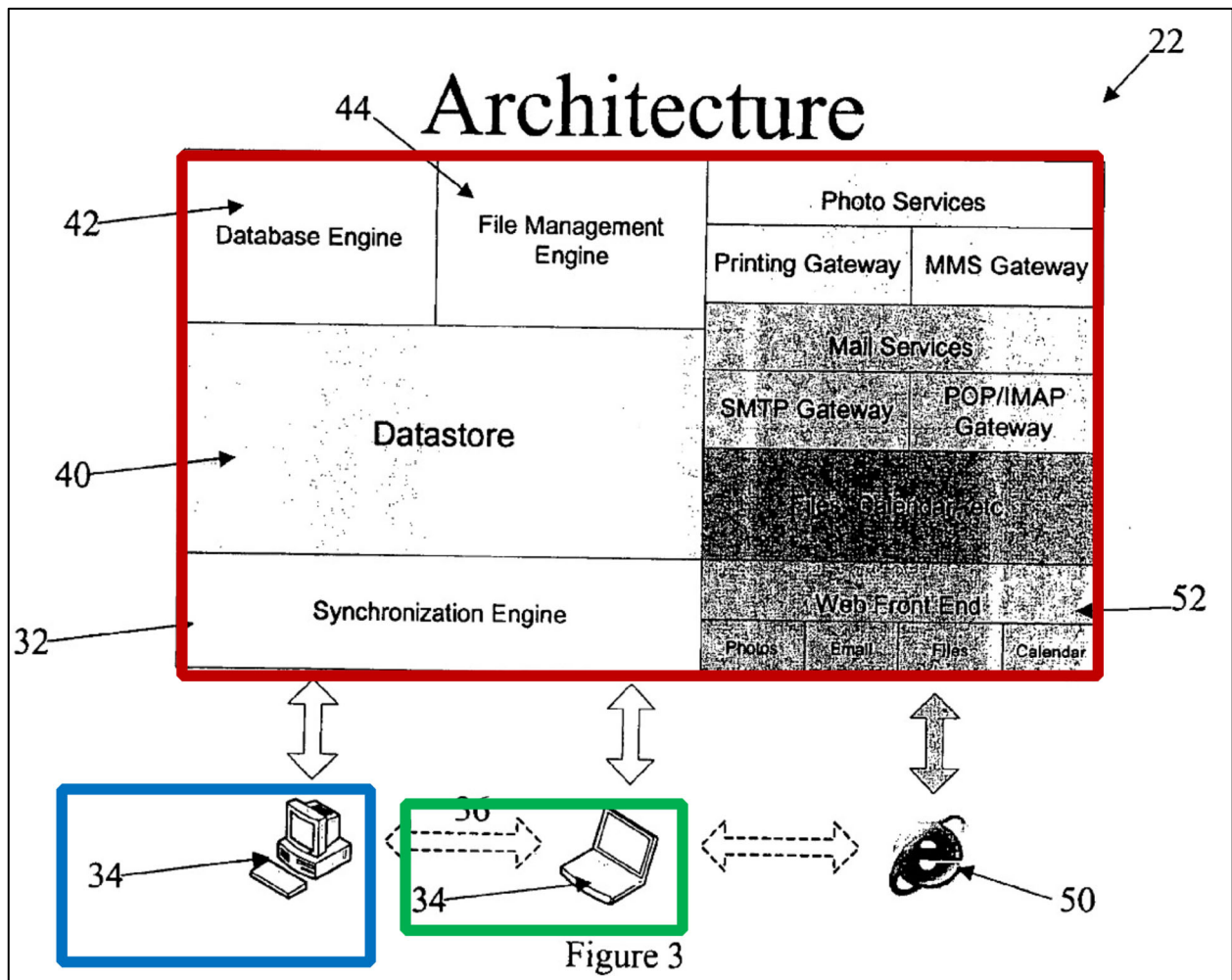
[10.3] automatically sending, via the first application, the modified first electronic file copy to a third application at the third electronic device responsive to the determination that the first electronic device is in communication with the third electronic device and responsive to receiving the modified first electronic file copy from the second electronic device;

This limitation is obvious over Strong in view of Brezak. Strong discloses that the **synchronization engine on server 22** automatically pushes the updated file copy (e.g., *automatically sending, via the first application, the modified first electronic file copy*) to another **sharing client 34 of another device, such as a laptop** (i.e., *to a third application at the third electronic device*) upon receiving the updated copy from the **first sharing client 34 of a device, such as a desktop** (i.e., *responsive to receiving the modified first electronic file copy from the second electronic device*):

[C]hanges or updates 30 from a sharing client 34 will be sent to the synchronization engine 32 **which will forward or update a copy of a data object on each of the clients listed as part of the plurality of sharing clients 34** in order to synchronize that data object with other sharing clients. Each client preferably has the ability to communicate with a server and have the appropriate software running thereon.

Strong (EX1006), [0026]; *see also* [0027] (update may comprise sending the entire data object), Claims 6-7 (files are data objects); [0029] (it is the “synchronization

engine 32 which communicates with sharing clients 34 running on devices”); [0031] (“[F]iles in datastore 40 are again pushed out to other sharing clients 34 so that the most recent versions are available to the users”); [0042] (“[T]he user can modify a picture . . . **Immediately, those changes will be synchronized across all of the user's multiple computers** or device[s].”); [0037], [0022], [0040].



Id., Fig. 3 (annotated).

As discussed regarding limitation 10.3, the combination of Strong and Brezak teaches that such sending would have been *responsive to the determination that*

server 22 is in communication with the third electronic device, such as a laptop because server 22 would detect the connectivity status of devices running sharing client 34 before pushing files to them. Sec. IV.A.4 (10.2-10.3), supra; see also Brezak (EX1007), Fig. 4, [0023]-[0024] (step 410, discovering the connection status, occurs before step 416, initiating file transfers).

[10.4] receiving, via the first application, a copy of a modified second electronic file from the third application at the third electronic device associated with the user, wherein the modified second electronic file copy is automatically received from the third application responsive to the user modifying a content of the second electronic file;

[10.5] determining whether the first electronic device is in communication with the second electronic device; and

[10.6]. automatically sending, via the first application, the modified second electronic file copy to the second application at the second electronic device responsive to the determination that the first electronic device is in communication with the second electronic device and responsive to receiving the modified second electronic file copy from the third electronic device,

Elements 10.4, 10.5, and 10.6 are like elements 10.1, 10.2, and 10.3, respectively, but for a second file copy going in the opposite direction. More precisely, 10.4 recites the first application of the *first electronic device* (e.g., synchronization engine 32 of **server 22**) receiving a *second* file copy from the *third electronic device* (e.g., **a device running sharing client 34**, such as the user's **laptop**), element 10.5 recites determining whether the first device is in communication with the *second electronic device* (e.g., **a different device running**

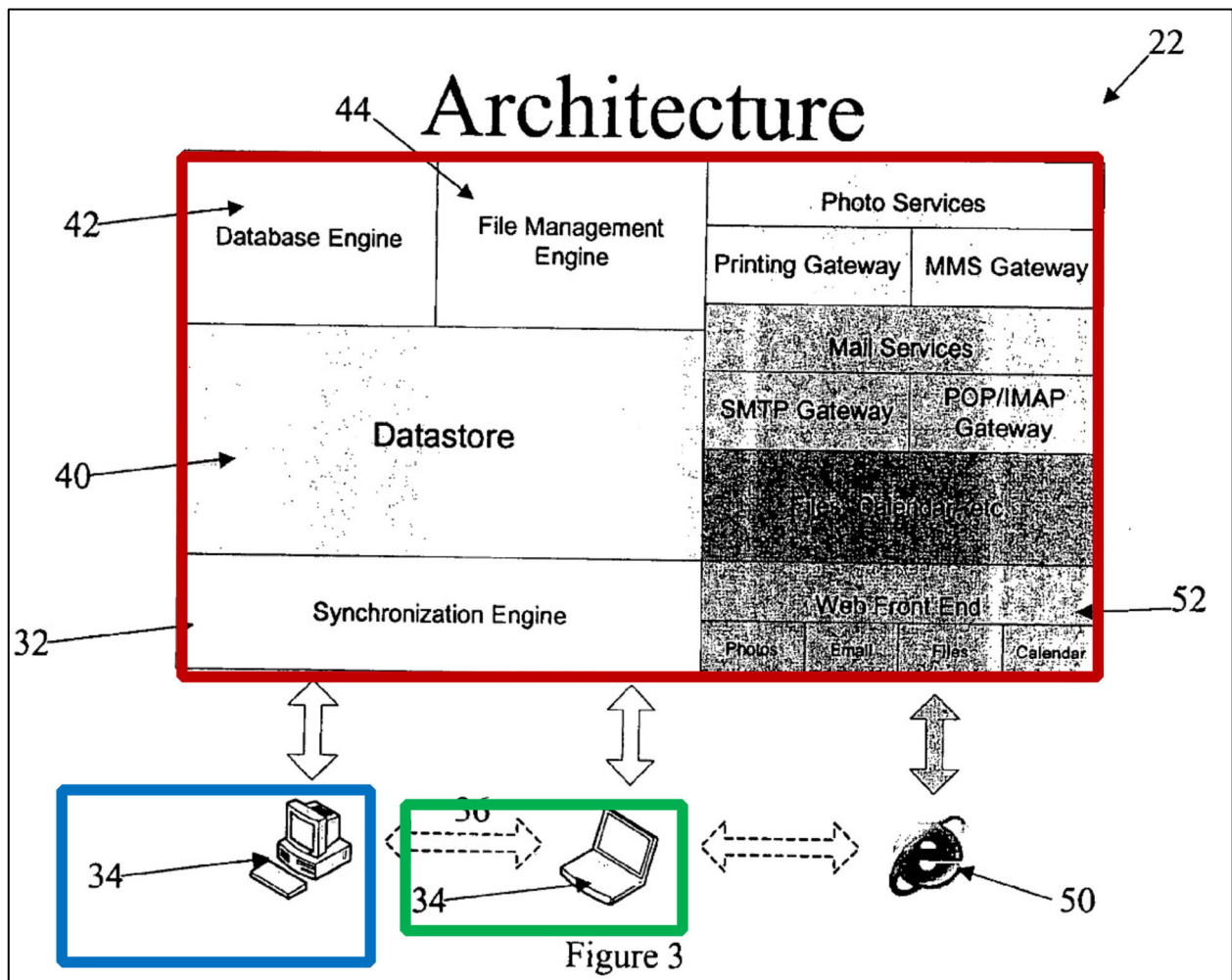
sharing client 34, such as a **desktop**), and element 10.6 recites the first application of the *first electronic device* automatically sending the second file copy to the *second electronic device*, responsive to the steps recited in limitations 10.4 and 10.5. Put another way, these limitations go to bi-directional file sharing.

For the same reasons discussed regarding elements 10.1 through 10.3, Element 10.4 is disclosed by Strong or at most an obvious variation of Strong's disclosure, and Elements 10.5 and 10.6 are obvious over Strong in view of Brezak. Strong discloses the bi-directional functionality claimed for a modified second file copy originating from the third device and sent to a second device. Strong describes its synchronization method as "two way" and uses the same reference number, 34, to refer to the different client applications and illustrates the reciprocal functionality of the client devices in Figure 3. *Strong* (EX1006), [0028]. Further, Strong explains and illustrates that the operations described are applicable for *each* of the devices running sharing clients 34:

[U]pdates 30 from **a sharing client 34** will be sent to the synchronization engine 32 which will forward or update a copy of a data object **on each of the clients listed as part of the plurality of sharing clients 34 . . . Each client preferably has the ability to communicate with a server and have the appropriate software running thereon.**

Id., [0026]; *see also* [0023]-[0025] (each client can communicate with any other

client through the server); [0006]-[0008] (the invention provides “easy sharing of files” and “file access from anywhere” where “[c]hanges to files . . . may be synchronized automatically across **a user’s computers**”); [0037] (“When a user with **a** sharing client 34 makes changes . . . the changes are synchronized automatically **across all devices**.”); [0030]-[0033] (describing the software employed by client devices to perform the invention); [0040], [0042] (describing a picture-sharing embodiment). This bi-directionality applies for multiple files *and* where the same file copy is modified. *See, e.g., id.* [0028], [0035] (describing conflict resolution procedures when a user edits the same file on different computers).



Id., Fig. 3 (annotated); *see also Long Decl.* (EX1005), ¶¶ 53, 57-59 (a POSA would have appreciated from Strong's figures and language that Strong's synchronization engine performs file transfers in both directions for any two sharing clients); ¶¶ 45-47 (describing known bi-directional synchronization technology).

[10.7] wherein, responsive to sending the modified first electronic file copy to the third electronic device, an older version of the first electronic file stored on the third electronic device is automatically caused to be replaced with the modified first electronic file copy such that the modified first electronic file copy is stored on the third electronic device in lieu of the older version of the first electronic file, and

Strong discloses element 10.7; alternatively, any differences would have been obvious variations to a POSA. Strong discloses that when files are sent from server 22 to sharing clients 34, such as a file sent from the user's **desktop** to **server 22** and then pushed to the user's **laptop** (*wherein, responsive to sending the modified first electronic file copy to the third electronic device*), the new file is saved and an older file copy is deleted from the receiving sharing client 34, such as a laptop, such that the modified copy is stored in lieu of an older version (*an older version of the first electronic file stored on the third electronic device is automatically caused to be replaced with the modified first electronic file copy such that the modified first electronic file copy is stored on the third electronic device in lieu of the older version of the first electronic file*). Strong (EX1006), [0027] (“[A]n update message 30 is sent from the particular sharing client to the synchronization engine 32. . . . **[I]n some embodiments, updates are represented by sending a new copy of the entire object.**”); *see also* [0028] (“The autosaved version may be . . . pushed out or cascaded to the sharing clients. **Some sharing clients may be configured to accept updates to files that are not currently being edited on at the local client, to**

prevent any conflicts.”); [0035] (in manually resolving conflicts, the user may select the server file so that it “replaces” the local version).

To the extent Strong is not express that the new version of the modified first electronic file copy *automatically* replaces and is stored *in lieu of* an older version, a POSA would have been motivated to, and would have reasonably expected success in implementing Strong’s system this way in view of Strong’s disclosure of how conflicts are manually resolved—by replacing or overwriting the version not selected by the user. *Strong* (EX1006), [0035]. Strong discloses that sharing clients may automatically accept changed files sent from the server to avoid conflicts. *Strong* (EX1006), [0028]. Because Strong provides express teachings of replacing old files with newer versions in another context, a POSA would have recognized that, at least in embodiments where updates are represented as the entire data object, applying this procedure to the automatic acceptance of new files would have been well within the scope of its system. *Long Decl.* (EX1005), ¶¶ 60-61; *see also Strong* (EX1006) [0051] (“[V]arious adaptations, changes, modifications, substitutions, deletions, or additions of procedures and protocols may be made without departing from the spirit and scope of the invention”). A POSA would have recognized that replacing the older files with the new to be one of the simplest ways from a programming perspective to implement the automatic acceptance of changes while avoiding conflicts. *Long Decl.* (EX1005), ¶ 61. Thus, this limitation is an obvious

variation over Strong’s combined teachings, if not expressly taught. *Game and Tech. Co., Ltd. v. Activision Blizzard Inc.*, 926 F.3d 1370, 1380-81 (Fed. Cir. 2019) (relying on separate disclosures from a single reference to find obviousness).

[10.8] wherein, responsive to sending the modified second electronic file copy to the second electronic device, an older version of the second electronic file stored on the second electronic device is automatically caused to be replaced with the modified second electronic file copy such that the modified second electronic file copy is stored on the second electronic device in lieu of the older version of the second electronic file.

Element 10.8 is like element 10.7, but refers to the second file, originating from the *third electronic device* (e.g., a **device running sharing client 34**, such as a user’s **laptop**) and sent from **server 22** to the *second electronic device* (e.g., another **device running sharing client 34** such as the user’s **desktop**). For the reasons discussed regarding element 10.7, Strong discloses the “replacing” step of element 10.8; alternatively, any differences would have been obvious variations to a POSA. As detailed above regarding elements 10.4-10.6, Strong’s synchronization techniques are applied across each of the sharing clients (*see, e.g., Strong*, [0026]), and a POSA would have appreciated that the reciprocal nature of the operation of the server and client applications would apply to the replacement of the second modified file copy. *Long Decl.* (EX1005), ¶ 59.

5. *Claim 1*

[1P]. A system comprising:

Strong discloses “an improved data management **system.**” *Strong* (EX1006), [0007]; *see also* Abstract.

[1.1]. a first electronic device, associated with a user, configured to:

Strong discloses this limitation; alternatively, any differences would have been obvious variations to a POSA. The '942 Patent specification does not use the phrase “associated with a user,” but it does disclose a server storing the user’s files and performing the steps recited in the following sub-elements of Claim 1. '942 *Patent* (EX1001), 7:41-56; 8:16-25. Likewise, Strong discloses a server (i.e., *a first electronic device*) on which a user’s files are stored, synchronized, and accessible via any of the user’s client devices (*associated with a user*). *See* Sec. IV.A.4 (10.1), *supra*.

[1.1.1] receive, via a first application at the first electronic device, a copy of a modified first electronic file from a second application at a second electronic device associated with the user, wherein the modified first electronic file copy is automatically received from the second application responsive to the user modifying a content of the first electronic file;

As discussed regarding element 10.1, Strong discloses this limitation; alternatively, any differences would have been obvious variations to a POSA. Sec. IV.A.4, *supra*.

[1.1.2] determine whether the first electronic device is in communication with a third electronic device;

As discussed regarding element 10.2, the combination of Strong and Brezak teaches this limitation. Sec. IV.A.4, *supra*.

[1.1.3] automatically send, via the first application, the modified first electronic file copy to a third application at the third electronic device responsive to the determination that the first electronic device is in communication with the third electronic device and responsive to receiving the modified first electronic file copy from the second electronic device;

As discussed regarding element 10.3, Strong discloses this limitation; alternatively, any differences would have been obvious variations to a POSA. Sec. IV.A.4, *supra*.

[1.1.4] receive, via the first application, a copy of a modified second electronic file from the third application at the third electronic device associated with the user, wherein the modified second electronic file copy is automatically received from the third application responsive to the user modifying a content of the second electronic file;

[1.1.5] determine whether the first electronic device is in communication with the second electronic device; and

[1.1.6] automatically send, via the first application, the modified second electronic file copy to the second application at the second electronic device responsive to the determination that the first electronic device is in communication with the second electronic device and responsive to receiving the modified second electronic file copy from the third electronic device;

Limitations 1.1.4 through 1.1.6 are nearly identical to limitations 10.4 through 10.6, reciting the transfer process from the previous three limitations in reverse. As discussed, Strong, alone (for element 1.1.4) or in view of Brezak (for elements 1.1.5

and 1.1.6) teaches these limitations; alternatively, any differences would have been obvious variations to a POSA. Sec. IV.A.4, *supra*.

[1.2]⁹ wherein, responsive to sending the modified first electronic file copy to the third electronic device, an older version of the first electronic file stored on the third electronic device is automatically caused to be replaced with the modified first electronic file copy such that the modified first electronic file copy is stored on the third electronic device in lieu of the older version of the first electronic file, and

As discussed regarding element 10.7, Strong discloses this limitation; alternatively, any differences would have been obvious variations to a POSA. Sec. IV.A.4, *supra*.

[1.3] wherein, responsive to sending the modified second electronic file copy to the second electronic device, an older version of the second electronic file stored on the second electronic device is automatically caused to be replaced with the modified second electronic file copy such that the modified second electronic file copy is stored on the second electronic device in lieu of the older version of the second electronic file.

As discussed regarding element 10.8, Strong discloses this limitation;

⁹ While elements 1.2 and 1.3 are indented past 1.1, treating them as sub-elements makes little sense given the use of a subordinating conjunction (*device configured to . . . wherein*) and passive voice, and the “and” concluding element 1.1.5, signaling that the next limitation is the last of the sub-elements of 1.1. Regardless, Strong discloses these limitations (or obvious variations) because the server’s transmission causes the older version to be replaced.

alternatively, any differences would have been obvious variations to a POSA. Sec. IV.A.4, *supra*.

6. Claims 3 and 12

3. The system of claim 1, wherein the first electronic device is configured to: store, via the first application, the modified first electronic file copy to a memory device associated with the first electronic device.

12. The method of claim 10, further comprising: storing, via the first application, the modified first electronic file copy to a memory device associated with the first electronic device.

Claims 3 and 12 depend from Claims 1 and 10, which, as discussed, are obvious over the combination of Strong and Brezak. Further, Strong discloses the additional limitations of Claims 3 and 12; alternatively, any differences would have been obvious variations to a POSA. Strong discloses that server 22 (i.e., *the first electronic device*) is configured to and does *store, via the first application* (e.g., synchronization engine 32), *the modified first electronic file copy to a memory device* (e.g., datastore 40) *associated with the first electronic device* (e.g., the datastore exists on server 22). Strong (EX1006), [0037] (“The synchronization engine 32 will have the datastore 40 updated with the latest file . . .”); *see also* [0030] (Datastore 40 “holds complete copies of the user files being shared.”); [0010] (“The present invention provides storage of a users [*sic*] files at a central server”). Fig. 3.

As discussed, a POSA would have appreciated that the synchronization engine 32 is a *first application*, or at least part of a *first application*, that runs on server 22.

Sec. IV.A.4 (10.1), *supra*.

Further, if datastore 40 were not a memory *device*, a POSA would have appreciated that a memory device is at least obvious over Strong's disclosure of datastore 40 given the nature of memory devices and Strong's disclosure regarding servers generally having greater memory capacity than clients. *Long Decl.* (EX1005), ¶ 49 n.14; *see also Strong* (EX1006), [0020]. In addition, it was well-known and very common at the time that the devices identified by Strong, such as clients and servers, stored data in memory devices. *Id.*

Thus, Claims 3 and 12 are obvious over the combination of Strong and Brezak.

7. *Claims 4 and 13*

4. The system of claim 3, wherein the modified first electronic file copy is stored on the memory device associated with the first electronic device in lieu of an older version of the first electronic file copy that was stored on the memory device associated with the first electronic device.

13. The method of claim 12, wherein the modified first electronic file copy is stored on the memory device associated with the first electronic device in lieu of an older version of the first electronic file copy that was stored on the memory device associated with the first electronic device.

Claims 4 and 13 depend from Claims 3 and 12, respectively, which are obvious over the combination of Strong and Brezak. Sec. IV.A.6, *supra*. Further, Strong discloses the additional limitations of Claims 4 and 13; alternatively, any differences would have been obvious variations to a POSA. Strong discloses *that*

the modified first electronic file copy is stored on the memory device (datastore 40) associated with the first electronic device (server 22) in lieu of an older version of the first electronic file copy that was stored on the memory device associated with the first electronic device (e.g., updates are either made to the copy of the data object on the server or are represented as a new copy of the entire object, and instructions in an update message may be to delete the existing data object):

[A]n update message 30 is sent from the particular sharing client to the synchronization engine 32. **The synchronization engine, in turn, will make the particular adjustment to the original or copy of the data object on the server . . . The update message 30 may include instructions, such as a delete command to remove a particular data object or a modify command for modifying a particular data object.** One way of representing a modification to an existing data object or the creation of a new data object is to include the actual data object in its modified form. Thus in **some embodiments, updates are represented by sending a new copy of the entire object.**

Strong (EX1006), [0027]; *see also* [0034] (server 22 maintains a log of changes, including identifiers for objects that have been “deleted in the server’s datastore”); Claim 6 (a data object may comprise an entire file); [0035] (conflicts between client and server versions of a file may be resolved by sending the client version to the server “**to overwrite the existing version**”).

To the extent Strong does not expressly disclose that the new version of the

file is stored *in lieu of* an older version in all of its embodiments, such would have been an obvious variation over Strong’s teachings as a whole. *Long Decl.* (EX1005), ¶¶ 60-61; *see also Strong* (EX1006), [0051] (“[V]arious adaptations, changes, modifications, substitutions, deletions, or additions of procedures and protocols may be made without departing from the spirit and scope of the invention.”). Replacing older versions of files with newer versions, disclosed by Strong in at least one context, would have been one of the simplest ways from a programming perspective to implement Strong’s methods while avoiding the risk of conflicts. *Long Decl.* (EX1005), ¶ 61.

Thus, Claims 4 and 13 are obvious over the combination of Strong and Brezak.

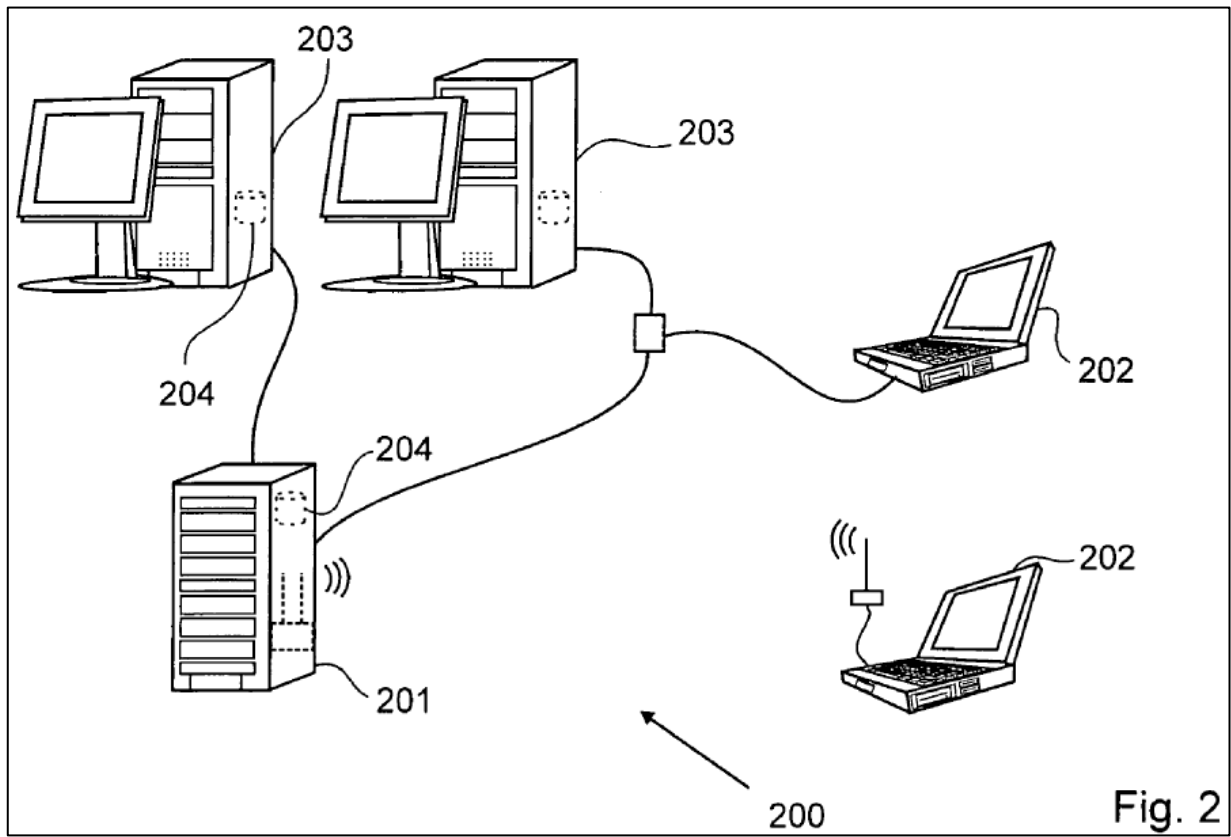
B. Ground 2: Obviousness of Claims 5-6 and 14-15 over Strong, Brezak, and Hamidi

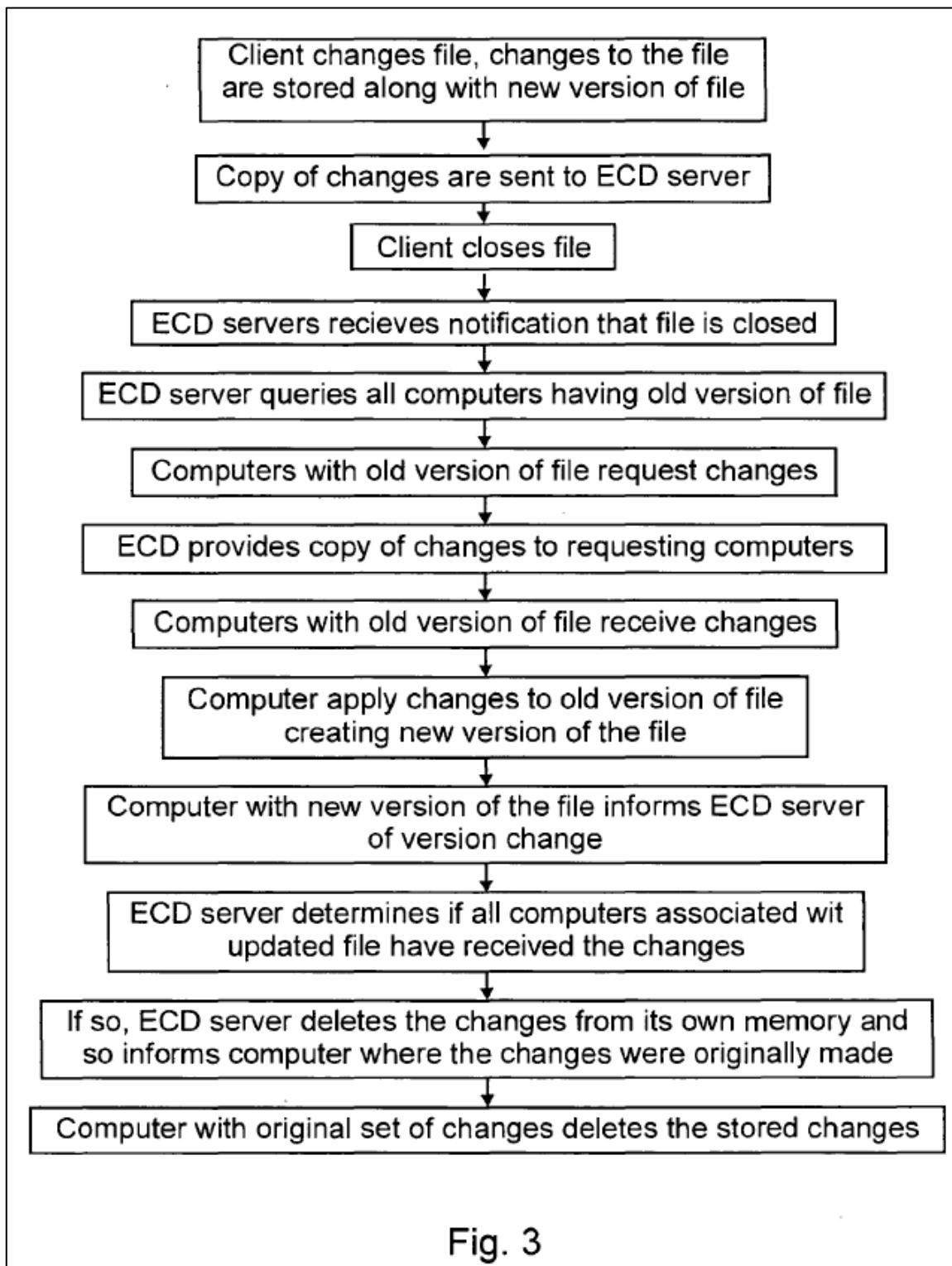
1. Overview of Hamidi

Hamidi claims priority to a provisional filed June 25, 2003, was filed June 25, 2004, and was published December 30, 2004; therefore, Hamidi is prior art under 35 U.S.C. §§ 102(a), (b), and (e).

Hamidi describes a file storage network in which an enterprise cloned drive (“ECD”) server receives updates from client devices, such as laptops and desktops, and transfers the updates to other clients authorized to maintain a copy of the

modified file. *Hamidi* (EX1008), [0007]-[0026] (outlining exemplary method and system according to Hamidi's disclosure). Client-side software generates "changes files" reflecting the updates. *Id.*, [0039]. The ECD server is notified whenever a client device is newly connected to the network, makes changes to a file, or has saved a file while not in communication with the server. *Id.*, [0038]. The ECD server queries the client for the changes file, and when it receives the copy, queries other client computers authorized to maintain a copy of the file. *Id.*, [0039]. The other client computers request a copy of the changes file so that they may modify their copy, thereby generating a new version. *Id.* If a client computer that normally maintains a copy is not accessible, the ECD server will maintain the changes file and query that device when it returns to the network. *Id.*; *see also* [0040]. Figure 2 illustrates an embodiment of Hamidi's network architecture, and Figure 3 outlines a file-transfer process in flow-chart form:





Id., Figs. 2-3; *see also* Figs. 4-5 (alternative distributed application embodiment).

Hamidi is analogous art to the '942 Patent. *Long Decl.* (EX1005), ¶ 77. Hamidi is within the field of endeavor because it describes an improved file distribution system for sharing files among devices. *Hamidi* (EX1008), Abstract. Further, Hamidi is reasonably pertinent to problems concerning the inventors. For example, Hamidi describes overcoming deficiencies in the prior art related depending on users to manually update files shared between devices and effecting file sharing even when devices are not continuously connected. *Id.*, [0003]-[0005].

2. *The Combination of Strong, Brezak, and Hamidi*

Hamidi is cited for the concept of storing a file on a memory device *responsive to* a determination that a receiving device is not in communication with the claimed first device. Combined with Hamidi and Brezak's disclosure, Strong's server 22 would be modified such that files would be stored temporarily if a determination is made that a receiving sharing client is offline; if each of the one or more sharing clients on the list of user devices for receiving a file are online, files would simply be pushed to the devices, not stored.

3. Claims 5 and 14

5. The system of claim 3, wherein the modified first electronic file copy is stored on the memory device associated with the first electronic device responsive to a determination that the first electronic device is not in communication with the third electronic device.

14. The method of claim 12, wherein the modified first electronic file copy is stored on the memory device associated with the first electronic device responsive to a determination that the first electronic device is not in communication with the third electronic device.

Claims 5 and 14 depend from Claims 3 and 12, respectively, which are obvious over the combination of Strong and Brezak. Sec. IV.A.6, *supra*. Claims 5 and 14 are obvious over the combination of Strong, Brezak, and Hamidi.

Strong discloses that clients may communicate with one another directly. *Strong* (EX1006), [0029]; *see also* [0024], Fig. 3 (including dashed arrows between clients). Strong also discloses that server 22 has a list of the clients that are “part of the plurality of sharing clients 34 in order to synchronize” data objects with other sharing clients. *Id.*, [0026]; *see also* [0008].

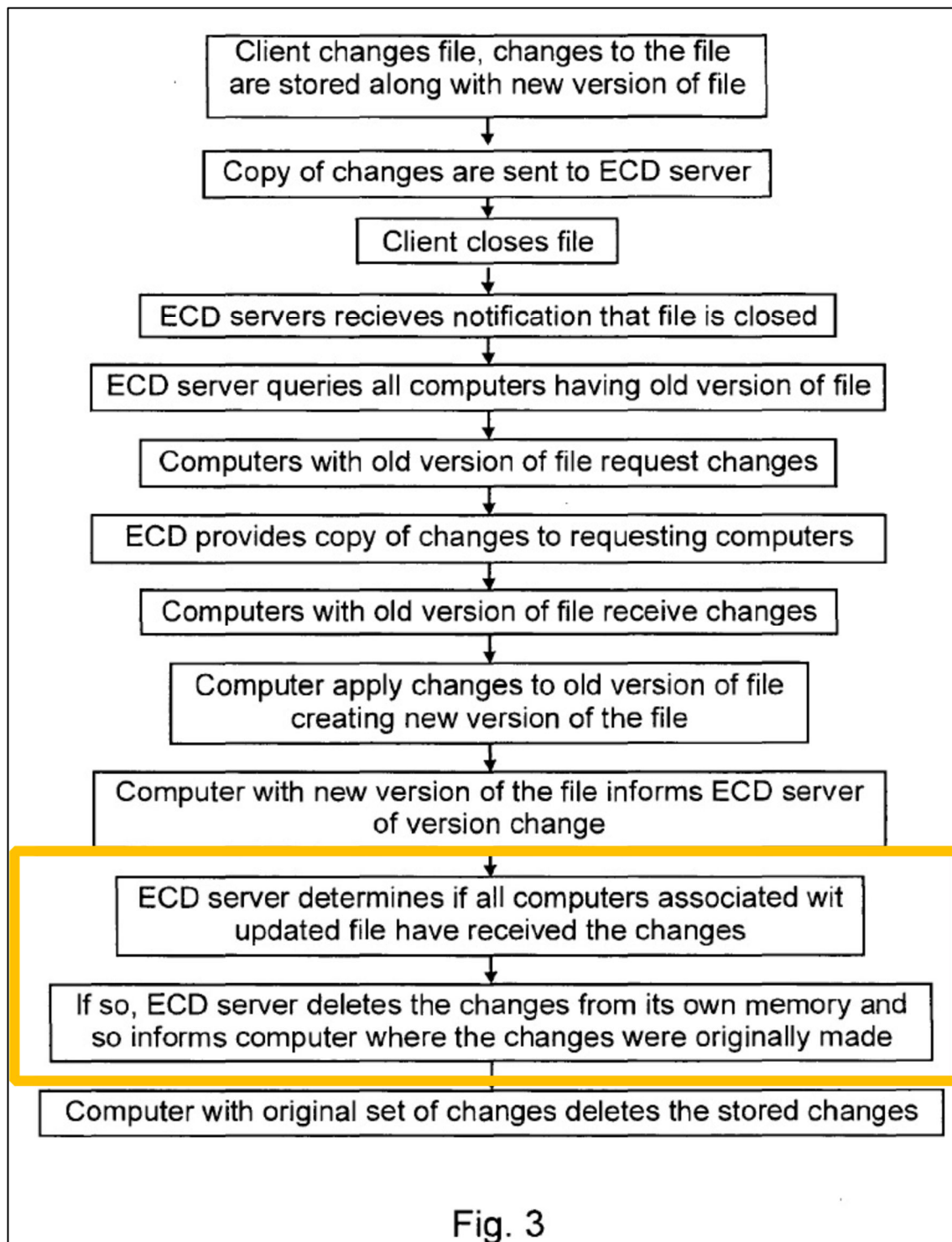
Strong in view of Brezak discloses a server determining whether it is in communication with the receiving client devices. Sec. IV.A.4 (10.2, 10.5), *supra*.

That the storage of the first file to the datastore is “responsive to” a determination that server 22 (i.e., *the first electronic device*) is *not in communication with* the receiving client 34 (i.e., *the third electronic device*) would have been obvious over the combination of Strong, Brezak, and Hamidi. Hamidi discloses that

a changes file is stored on the storage medium of ECD server 201 until the client computers that normally maintain a copy of the modified file return online:

If a client computer that normally maintains a copy of the network storage file is not accessible then the ECD server 201 maintains a copy of the changes file and queries the client computers authorized to have the network storage file when those client computers are once again in communication with the network. Thus, the ECD server 201 keeps a copy of the changes file until all client computers that normally maintain a copy of the network storage file receive the changes file.

Hamidi (EX1008), [0039].



Id., Fig. 3 (annotated). As modified by Hamidi and Brezak, Strong's system would store an updated file received from a first sharing client (i.e., *the modified first*

electronic file copy from the second electronic device) on datastore 40 of server 22 (i.e., the *memory device associated with the first electronic device*) only if the server 22 determined that the server is not in communication with a receiving sharing client (i.e., *responsive to a determination that the first electronic device is not in communication with the third electronic device*). *Long Decl.* (EX1005), ¶ 78.

A POSA would have been motivated to modify Strong, as combined with Brezak, to incorporate Hamidi's teaching of storing a file in a central location responsive to determining that it was not connected to a receiving client device for two reasons. First, Hamidi provides express motivations to combine: (1) increasing security against hackers who might gain access to the central server; *Hamidi* (EX1008), [0045], [0051]; and (2) conserving server resources; *id.*, [0006], [0035], [0051]. *Long Decl.* (EX1005), ¶ 79.

Second, a POSA would have recognized that the proposed modification would have simply been a matter of applying a known technique (storing data temporarily based on a negative determination related to the connection status of another device) to improve Strong's synchronization techniques (increasing server security and decreasing costs and resource demands for server 22) to yield predictable results consistent with Hamidi's teachings. *Id.*, ¶ 80.

Further, while the proposed modification may have had the potential disadvantage of slowing a user's access to their files via a web browser, a POSA

would have recognized that (1) such would have been offset by decreased storage costs and resource demands at the server and increased security and (2) by maintaining a list of files at Strong's server, ad hoc synchronization would still be possible. *Strong* (EX1006), [0009], [0034] (system synchronizes metadata, maintains a log of changes including object identifiers); *Long Decl.* (EX1005), ¶ 81; *Medichem, SA v. Rolabo, SL*, 437 F.3d 1151, 1165 (Fed. Cir. 2006) (“[A] given course of action often has simultaneous advantages and disadvantages, and this does not necessarily obviate motivation to combine.”). Further, web browser access is not a required feature of Strong's system. *Strong* (EX1006), Claim 5 (independent claim not reciting a web browser). Strong discloses that the “primary” method of accessing information is through native clients on the user's devices, and the web embodiments would still be applicable for sharing new files. *Strong* (EX1006), [0032]; *see also* Abstract, [0031] (a web interface “may” be provided”); [0051] (“various . . . deletions” may be made to Strong's procedures and protocols); *Long Decl.* (EX1005), ¶ 81.

A POSA would have had a reasonable expectation of success in making the above combination because Strong and Hamidi describe similar systems with similar goals, and Hamidi shows that conditionally storing files was possible in a similar context; further, such would have required at most minor modifications to Strong's server software logic and would have yielded predictable results

regarding the temporary storage of files. *Long Decl.* (EX1005), ¶ 82.

4. Claims 6 and 15

6. The system of claim 5, wherein the first electronic device is configured to: responsive to resuming communication with the third electronic device, automatically transfer, via the first application, the modified first electronic file copy to the third electronic device to cause the older version of the first electronic file stored on the third electronic device to be replaced with the modified first electronic file copy.

15. The method of claim 14, further comprising: responsive to resuming communication with the third electronic device, automatically transferring, via the first application, the modified first electronic file copy to the third electronic device to cause the older version of the first electronic file stored on the third electronic device to be replaced with the modified first electronic file copy.

Claims 6 and 15 depend from Claims 5 and 14, respectively, which are obvious over the combination of Strong, Brezak, and Hamidi. Sec. IV.B.3, *supra*. The combination of Strong, Brezak, and Hamidi likewise teaches this limitation. For example, as discussed regarding Claims 5 and 14, Hamidi discloses that, until a successful query, that is, when the client is “once again in communication with the network” (i.e., *resuming communication with the third electronic device*), and all applicable clients receive a changes file, the ECD server will maintain the updated copy. *Hamidi* (EX1008), [0039]. Further, for the reasons discussed regarding limitation 10.2 (determining whether a first device is in communication with a third device) and 10.7 (replacing an older version of a file), the combination of Strong and Brezak teaches the additional limitations of Claims 5 and 16; alternatively, any

differences would have been obvious variations to a POSA. Sec. IV.A.9. Therefore, Claims 6 and 15 are obvious over the combination of Strong, Brezak, and Hamidi.

C. Grounds 3 & 4: Obviousness of Claims 1, 3, 5-6, 10, 12, and 14-15 over Hamidi and Ellman and Hamidi, Ellman, and Brezak

1. Overview of *Ellman*

Ellman claims priority to a domestic patent application filed on June 7, 1999 and was itself published in English on December 14, 2000; therefore, Ellman is prior art under 35 U.S.C. §§ 102(a), (b), and (e).

Ellman describes software for synchronizing files between a user's home and office computers so that they may remotely work on files without manually copying them between computers. *Ellman* (EX1009), 6:2-20. A user designates the files for synchronization, and the software (pcTELECOMMUTE) ensures that the user has the updated version wherever they are. *Id.*, 6:14-20; *see also* 8:13-26, 2:8-16. Timestamps on different computer versions are compared, and the younger version is sent to the computer with the older version for overwriting. *Id.*, 13:10-14, 23:4-14, Claim 4, Abstract (57); *see also* 8:23-26 (overwrite may be automatic or upon user confirmation).

Ellman is analogous art to the '942 Patent. *Long Decl.* (EX1005), ¶ 87. Ellman is within the field of endeavor because it relates to the sharing of electronic files among a user's devices. *Ellman* (EX1009), Abstract, [0003]. Further, Ellman

addresses at least two problems concerning the inventors, namely, the automatic sharing of up-to-date files between a user's devices; *id.*, 1:14-25; and avoiding file redundancy by writing over older files with newer ones. *Id.*, 2:20-28.

2. *The Combination of Hamidi and Ellman*

Hamidi was introduced in Ground 2. Sec. IV.B.1. Combined with Ellman, Hamidi's system would employ the bi-directional transfer of entire file copies and, therefore, would implement updates by overwriting older files instead of having client software programs follow instructions in a changes file. *See, e.g.*, Sec. IV.C.3 (10.1, 10.7), *infra*.

3. *Claim 10*

[10P]. A method comprising:

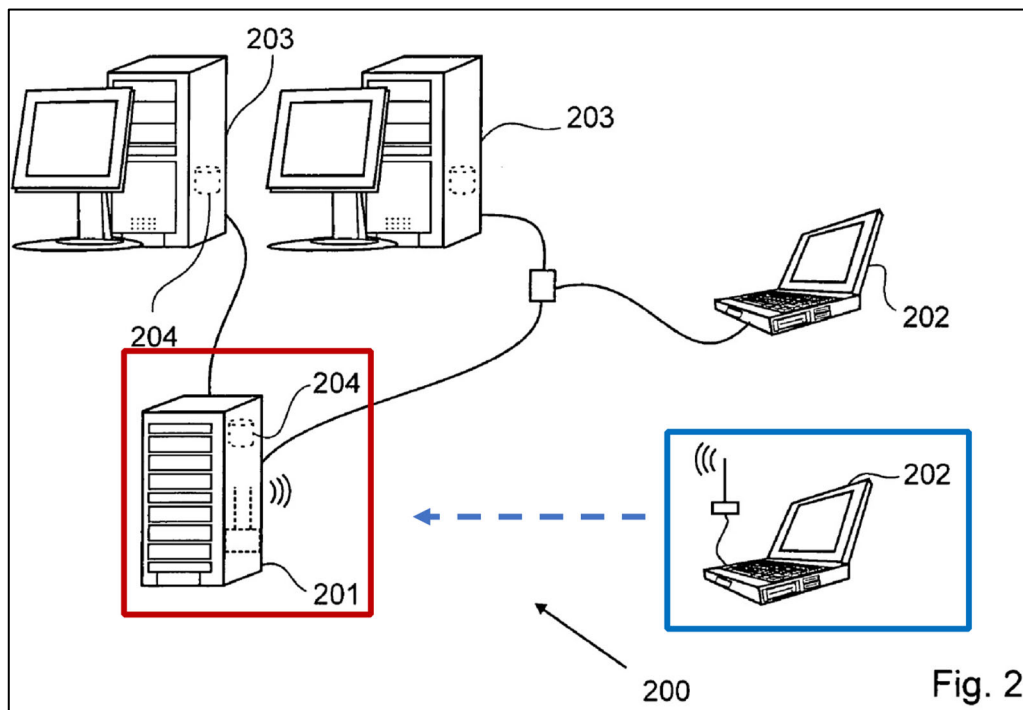
Hamidi discloses a method of storing electronic data files. *Hamidi* (EX1008), [0007].

[10.1] receiving, via a first application at a first electronic device, a copy of a modified first electronic file from a second application at a second electronic device associated with a user, wherein the modified first electronic file copy is automatically received from the second application responsive to the user modifying a content of the first electronic file;

This limitation is taught by the combination of Hamidi and Ellman. The '924 Patent discloses examples of a server automatically receiving a file from a client based on some triggering event, such as the user saving a file. '942 Patent (EX1001), 8:37-44. Likewise, Hamidi describes *receiving via* software at an

“enterprise cloned drive server,” **ECD server 201** (i.e., *an application at a first electronic device*), a changes file generated by software on a client computer, such as a user’s **laptop 202** (i.e., *a second application at a second electronic device associated with a user*), wherein the changes file is automatically copied from the client (i.e., *is automatically received from the second application*) upon a notification that the client has saved a file (i.e., *responsive to the user modifying a content of the first electronic file*):

When a network storage file is saved it is stored on the hard disc drive of the computer that saved it along with a changes file that stores all of the changes that have been made from the previous version of the network storage file The changes file is generated by client software on the client computers, both laptop computer 202 and desktop computer 203, associated with the ECD server 201. **The ECD server 201 receives a notification regarding the saving of the network storage file and queries the client computer to automatically copy the changes file to a storage medium of the ECD server 201.**

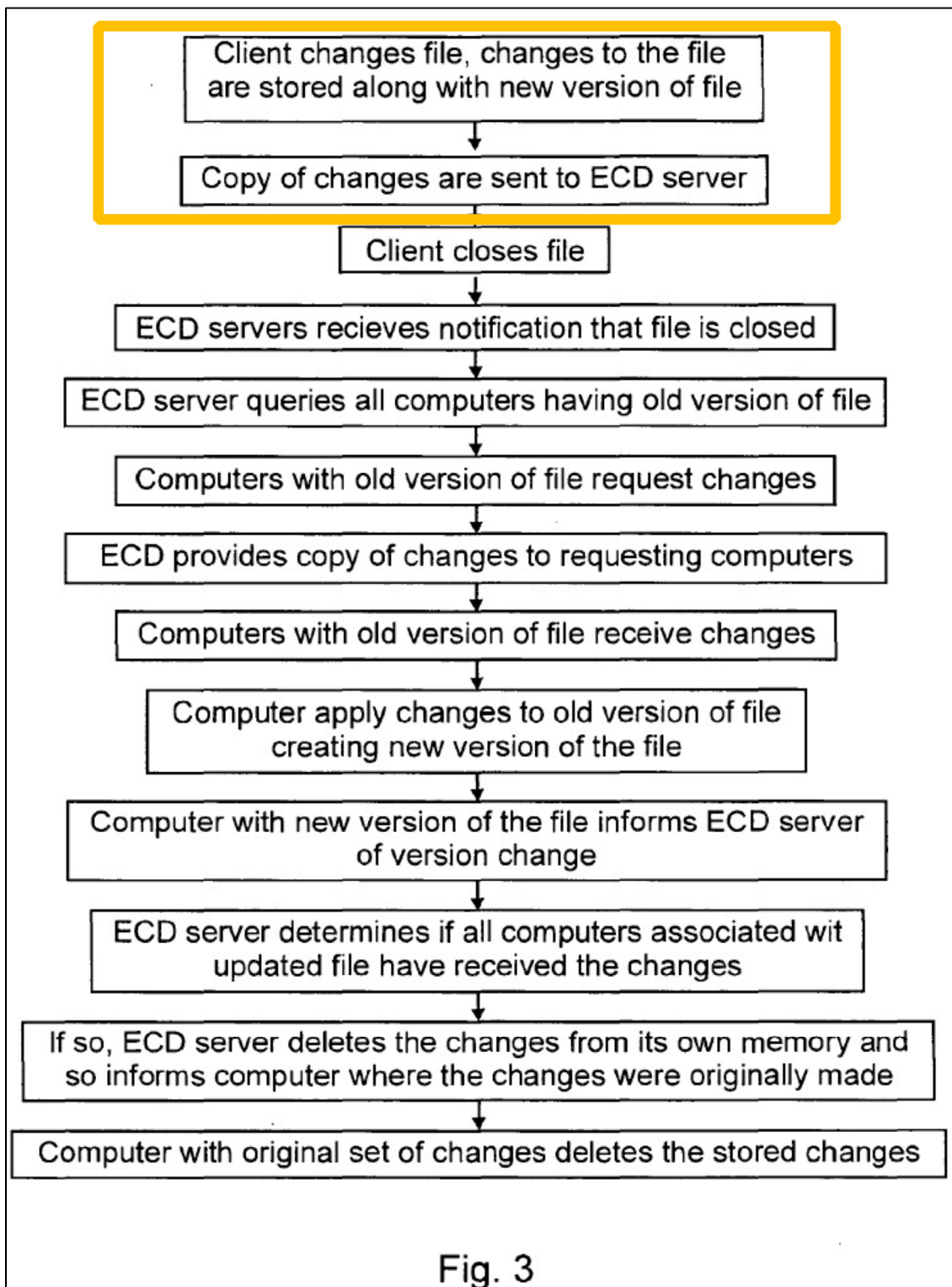


Hamidi (EX1008), [0039], Fig. 2 (annotated); *see also* [0045] (“**[F]ile updates . . . are handled automatically**, the user need not remember to update files . . .”); [0015]-[0017]; Fig. 4, [0046]-[0049] (describing a second embodiment using multiple ECD servers in a distributed fashion). The client may provide the changes file without a data transfer indicating that the server is ready to receive it. *Id.* [0039].

The ECD server runs “server software,” i.e., *a first application* to implement the functions of the ECD server. *Id.*, [0044]; *see also* [0050] (explaining that “any type of file system that provides generic files to **applications**” are useable to implement the invention); *see also Long Decl.* (EX1005), ¶ 75.

Hamidi discloses the “automatically received” concept in another way, where changes files are copied to a storage medium of the server from the client (i.e.,

automatically received from the second application) as they are generated (i.e., *responsive to the user modifying a content of the first electronic file*), and Hamidi emphasizes that updates should be made “promptly” when the server and clients enjoy a high bandwidth connection between them. *Hamidi* (EX1008), [0040] (“[E]ach set of changes to the network storage file made between each instance of saving the network storage file are optionally uploaded to the ECD server 201 as these changes files are generated.”).



Id., Fig. 3 (annotated).

It would have been obvious to modify Hamidi to transfer entire files, instead of just changes files, in light of Ellman's teachings. Like Hamidi, Ellman teaches a system in which files are synchronized between one or more computers; unlike Hamidi, copies of the entire file are used for synchronization:

With pcTELECOMMUTE's File Sync the user can synchronize individual files, entire folders or sub-folders on both the home and office PCs. The user can use File Sync to send files to, receive files from, and synchronize files with his office PC or network. . . .

When the user receives a file or folder from his office PC, pcTELECOMMUTE **copies the file or folder from his office PC to his home PC**. When the user synchronize files, **pcTELECOMMUTE copies the file from his office or home PC, depending on which is newer.**

Ellman (EX1009), 12:23-13:7; *see also* 2:7-28, 11:20-28, 12:4-12, 13:16-19, 23:4-14 (similar disclosures); 14:20-15:2 (users have the option between automatic synchronization or manual file transfer). Ellman's system uses timestamps to identify the most recent version, so tracking changes is not necessarily required. *Id.*, 8:12-26; *see also* 2:20-28, 13:4-9, 23:9-14.

Combined with Ellman, Hamidi's system would transfer entire files between devices instead of a changes file. A POSA would have been motivated to modify Hamidi in light of Ellman's disclosure for two reasons. First, Hamidi provides express motivation to combine, explaining that it would be "apparent" to a POSA

that a variation of its system “**supports the storage and distribution of entire files instead of change files.**” *Hamidi* (EX1008), [0055]; *see also Long Decl.* (EX1005),

¶ 88. Ellman’s disclosure simply confirms this and teaches specific methods of sharing entire files in a manner applicable to Hamidi’s system and methods. *Long Decl.* (EX1005), ¶ 88.

Second, the proposed combination would have simply been a matter of combining prior art elements (Hamidi’s central file distribution system and Ellman’s synchronization techniques) according to known methods (Ellman’s method of synchronization using file timestamps) to yield predictable results; indeed, Hamidi itself recognizes that the transferring and storage of entire files was known in the art, and instead proposes the use of change files as one improvement, among others. *Long Decl.* (EX1005), ¶ 89; *see also Hamidi* (EX1008), [0035]

A POSA would have appreciated that the proposed combination would have been consistent with the principal operation of Hamidi’s system, as transferring entire files would have been particularly appropriate where a user desired to synchronize devices for working remotely, an application contemplated by Hamidi, because one would rarely use different computers simultaneously. *Long Decl.* (EX1005), ¶ 90; *see also Hamidi* (EX1008), [0004]-[0005] (one object of Hamidi’s system is to allow a user to work remotely). Further, the proposed combination would circumvent the drawbacks of a “wasteful” mirror server (*Hamidi*, [0035])

because files would still only be stored temporarily, as discussed regarding limitations 10.7 and 10.8, *infra. Long Decl.* (EX1005), ¶ 91.

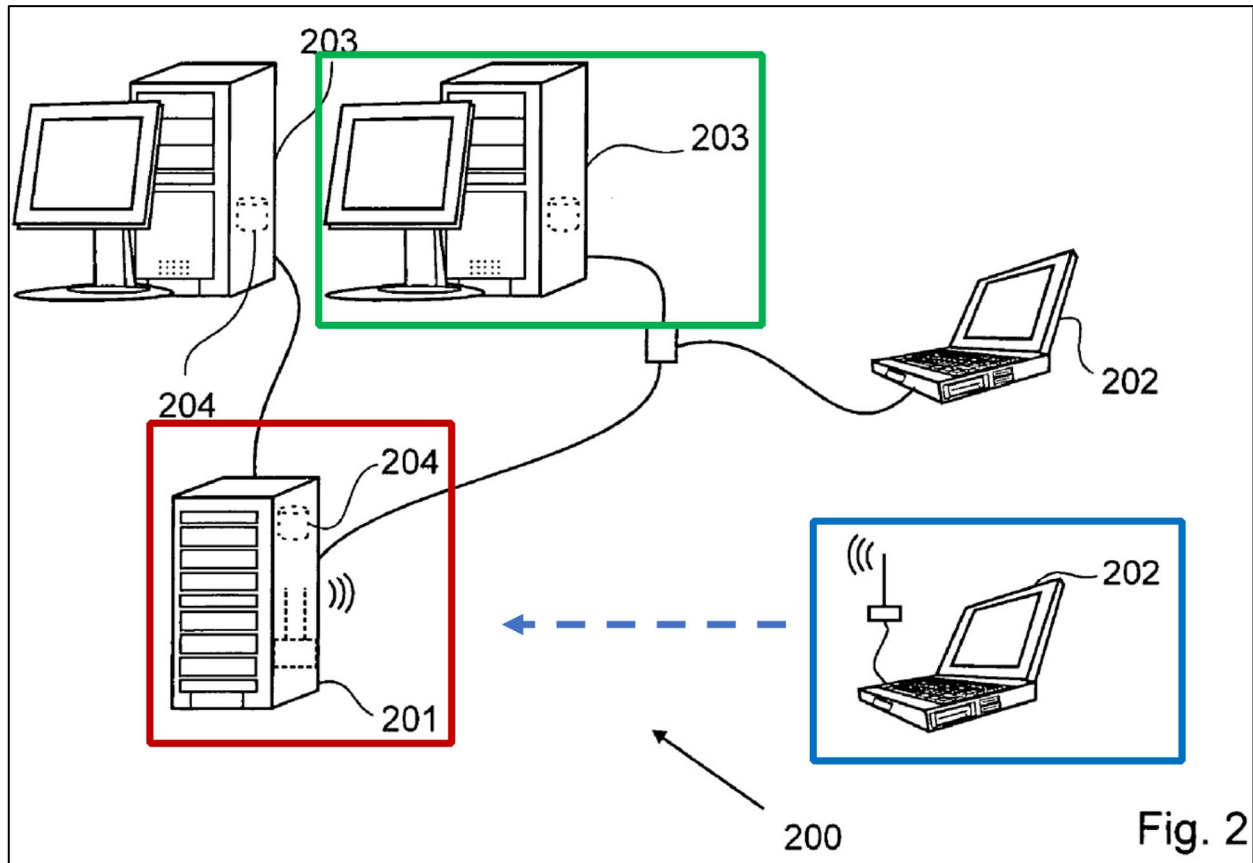
A POSA would have had a reasonable expectation of success in making the combination, which would have required the application of well-known techniques in file storage systems, as noted in Hamidi itself. *Hamidi* (EX1008), [0037]; *see also* [0035]; *Long Decl.* (EX1005), ¶ 92. Further, Ellman references transferring files through a network server, so a POSA would have expected success in implementing its teachings on a centralized system. *Ellman* (EX1009), 17:12-17; *Long Decl.* (EX1005), ¶ 92. Thus, the proposed combination teaches element 10.1.

[10.2] determining whether the first electronic device is in communication with a third electronic device;

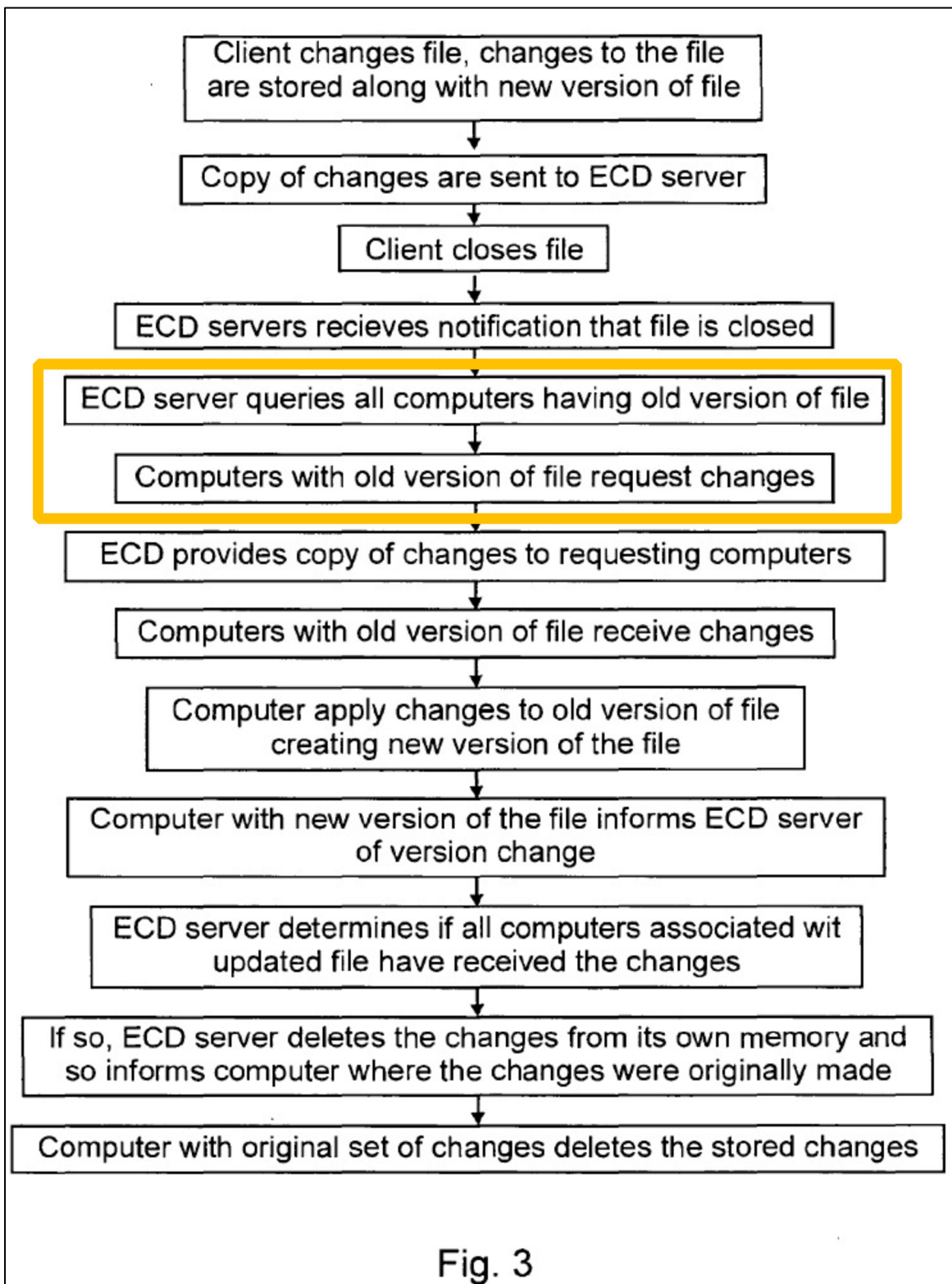
Hamidi discloses this limitation; alternatively, any differences would have been obvious variations to a POSA. Hamidi discloses that **ECD server 201** (i.e., *the first electronic device*) determines whether it is connected to another client device, such as **desktop computer 203**, over a network (*determining whether the is in communication with a third electronic device*), such as by querying the client computers.

Hamidi's ECD server 201 **"is notified when a client computer is connected to the network."** *Hamidi* (EX1008), [0038]; *see also* [0039] (the ECD server **"queries other client computers that are authorized to maintain a copy of the**

network file” and, if a given client is inaccessible, ECD server 201 maintains a copy and “queries the client computers . . . when those client computers are once again in communication with the network.”).



Id., Fig. 2 (annotated).



Id., Fig. 3 (annotated).

Further, to the extent Hamidi does not expressly disclose *determining* whether the different client devices, including desktop computer 203, are in communication with the ECD server, such would have been at most obvious over Hamidi's disclosure in light of the knowledge and ordinary skill of a POSA, because servers typically included sockets for "listening" for a client connection and would determine if a connection is complete before fulfilling requests, such as the requests for a file transfer. *Long Decl.* (EX1005), ¶¶ 100, 41 (a server waits for client acknowledgment before servicing requests in common network protocols).

Additionally, these claims are obvious over the combination of Hamidi, Ellman, and Brezak, which, as discussed in Ground 1, provides further teachings regarding the *determining* step. Sec. IV.A.4, IV.A.5, *supra* (10.2, 10.5, 1.1.2, 1.1.5). As with the combination of Strong and Brezak discussed above, a POSA would have been motivated to modify, and would have reasonably expected success in modifying, Hamidi's system to implement the server such that it would "determine" whether it is in communication with a client computer authorized to receive the network storage file because (1) such is suggested by Hamidi's disclosure; Brezak simply provides additional details regarding determining a connection status of a recipient consistent with Hamidi's teachings regarding the ECD server querying client computers and receiving file requests, and (2) the combination would have simply been a matter of combining prior art elements (server-client systems with a

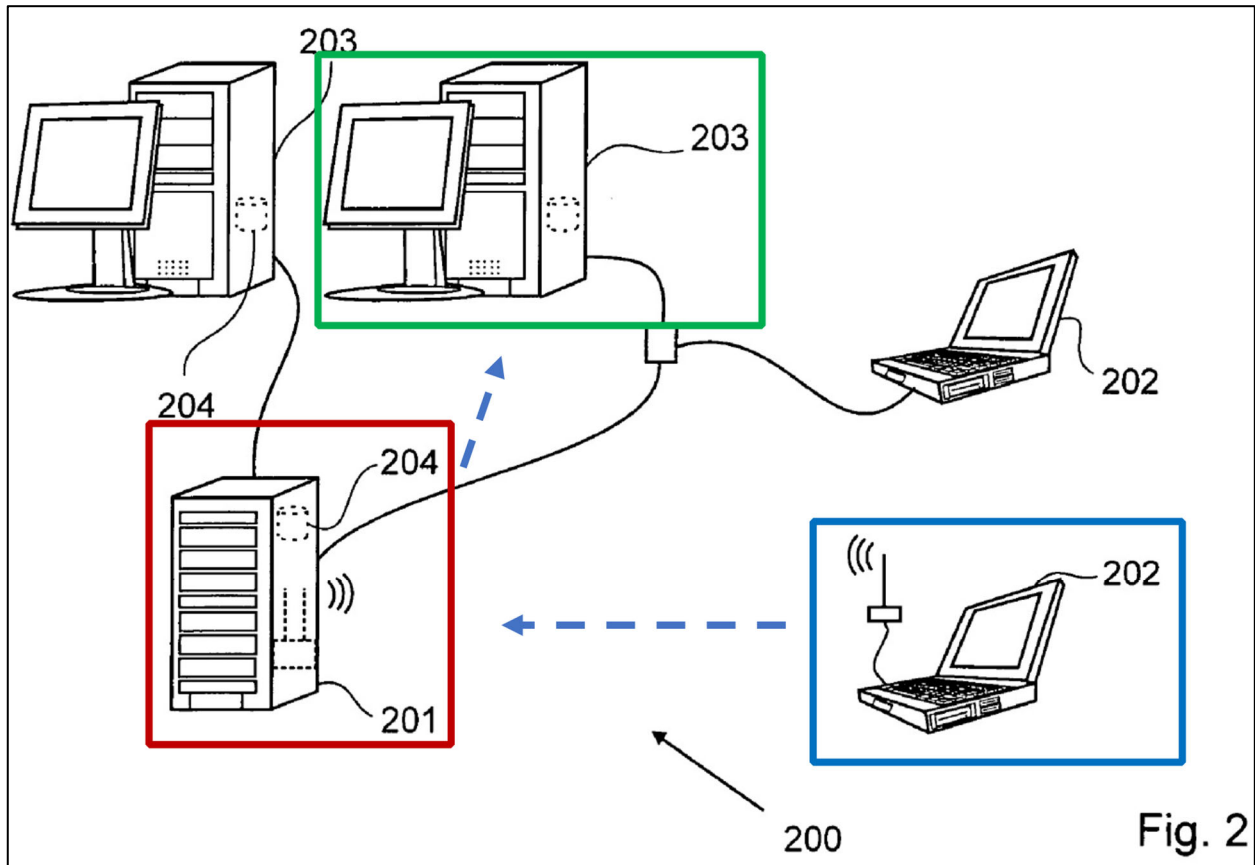
communication status check) according to known methods to yield predictable results (having a server determine whether it is in communication with a client before attempting to serve file requests). *Long Decl.* (EX1005), ¶¶ 99-100. Any changes to Hamidi's system would have involved at most minor modifications in software and would have yielded predictable results that would have been consistent with known aspects of server-client relationships. *Long Decl.* (EX1005), ¶ 101.

[10.3] automatically sending, via the first application, the modified first electronic file copy to a third application at the third electronic device responsive to the determination that the first electronic device is in communication with the third electronic device and responsive to receiving the modified first electronic file copy from the second electronic device;

This limitation is taught by the combination of Hamidi and Ellman. Hamidi discloses that the software running on the **ECD server 201** will automatically send (i.e., *automatically sending, via the first application*) a changes file to the software of a receiving client device, such as **desktop 203**, (i.e., *to a third application at the third electronic device*) upon a confirmation response to a query to the receiving client device (i.e., *responsive to the determination that the first electronic device is in communication with the third electronic device*) and upon receiving the changes file from the originating client device, such as **as laptop 202** (i.e., *responsive to receiving the changes file from the second electronic device*):

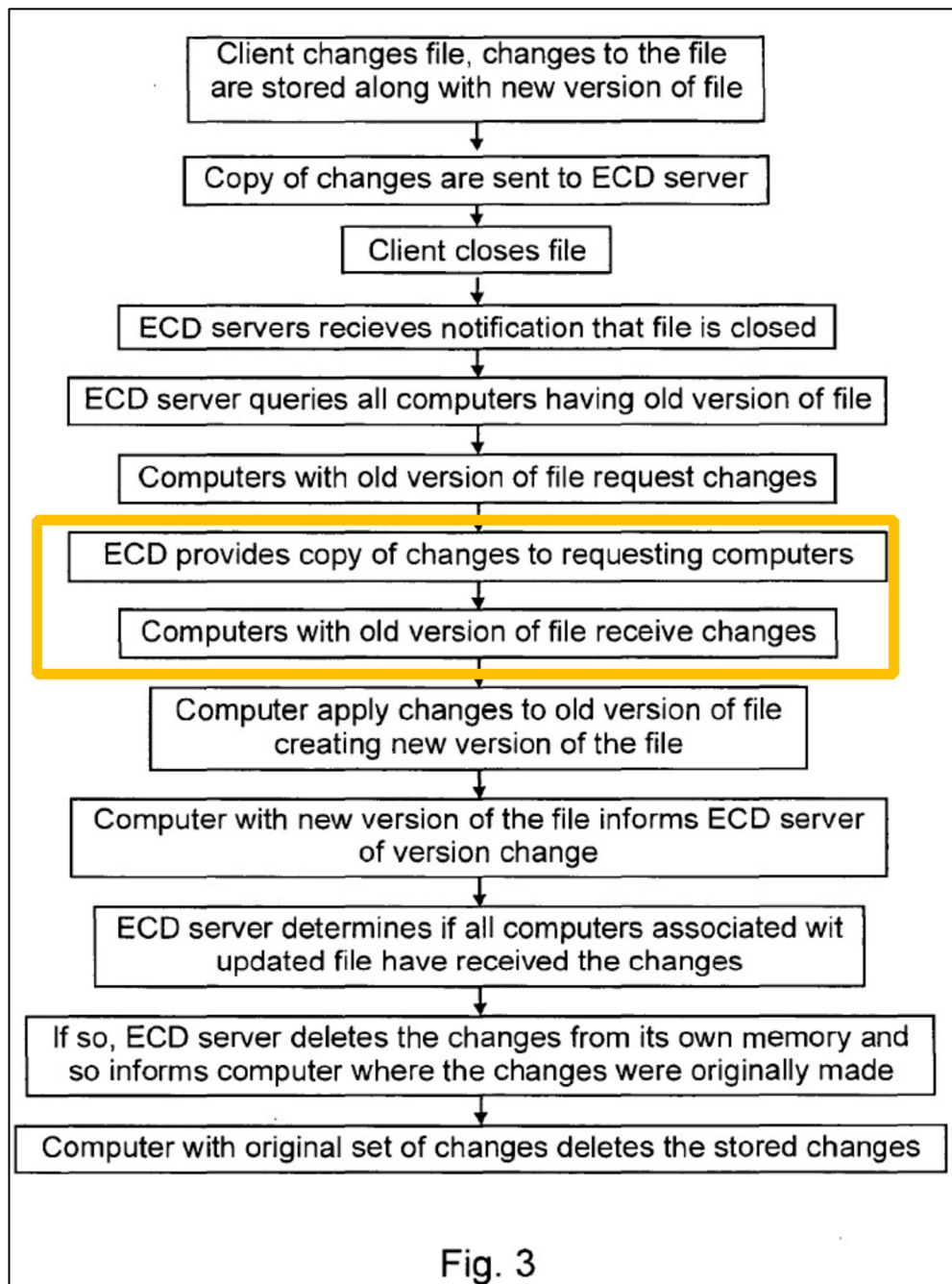
The ECD server . . . queries the client computer to automatically copy the changes file to a storage medium of the ECD server 201. **The ECD**

server 201 then queries other client computers that are authorized to maintain a copy of the network storage file. The authorized client computers that are in communication with the ECD server 201 then request a copy of the changes file. These client computers use the changes file to modify their copy



Hamidi (EX1008), [0039], Fig. 2 (annotated); *see also* [0017] (invention discloses a method comprising “**transferring** at least a change instruction from the first computing device to the second computing device **via the server.**”). This transfer happens “automatically.” *Id.*, [0045] (“[T]he system will act to save another copy of the file to another client computer **automatically.**”); *see also* [0050] (same);

[0040] (updates occur “promptly” as changes files are generated);



Id., Fig. 3.

Further, as discussed regarding element 10.1, a POSA would have been motivated to modify, and would have reasonably expected success in modifying,

Hamidi to have the ECD server receive and transfer entire files. Thus, the combination of Hamidi and Ellman render this limitation obvious.

[10.4] receiving, via the first application, a copy of a modified second electronic file from the third application at the third electronic device associated with the user, wherein the modified second electronic file copy is automatically received from the third application responsive to the user modifying a content of the second electronic file;

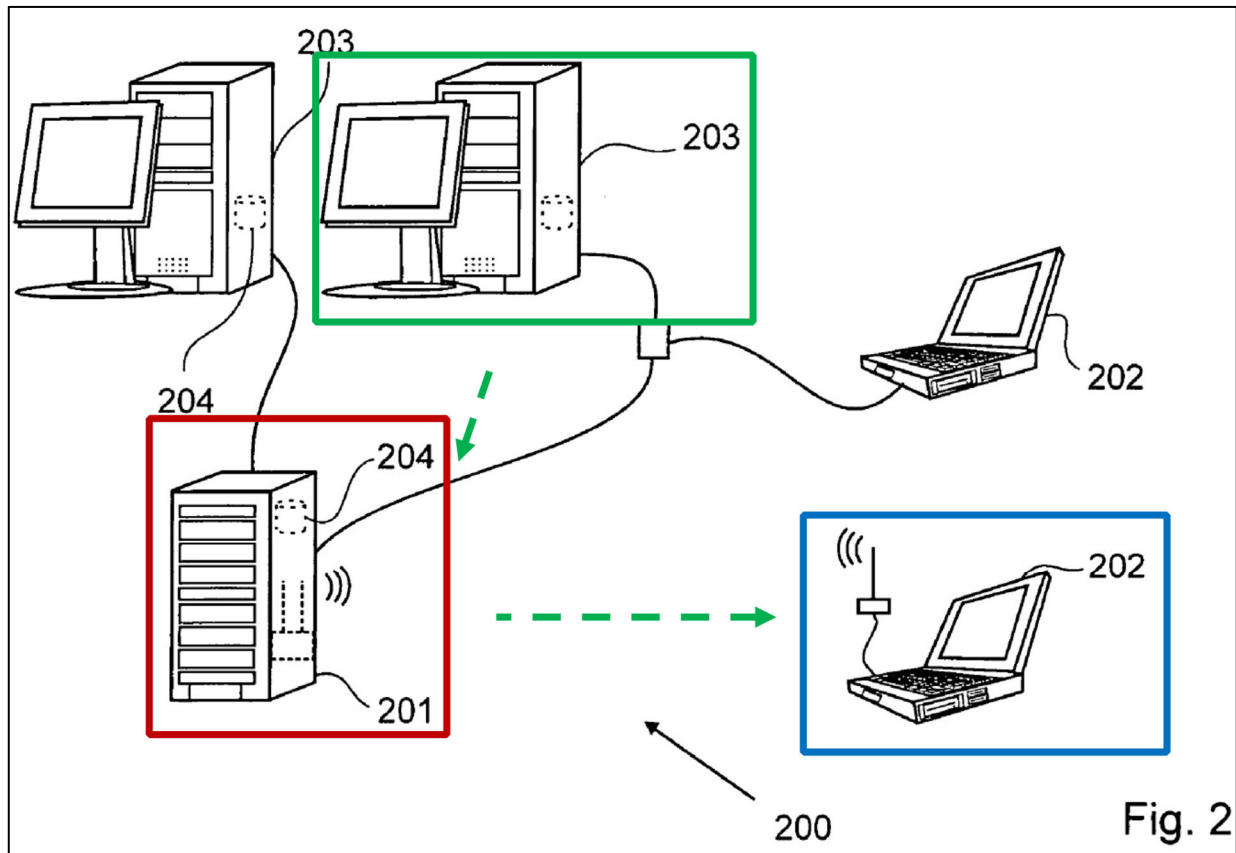
[10.5] determining whether the first electronic device is in communication with the second electronic device; and

[10.6]. automatically sending, via the first application, the modified second electronic file copy to the second application at the second electronic device responsive to the determination that the first electronic device is in communication with the second electronic device and responsive to receiving the modified second electronic file copy from the third electronic device,

Elements 10.4 through 10.6 reiterate the synchronization steps of elements 10.1 through 10.3, but for a second file in the reverse direction: from the third application at the *third electronic device* (e.g., client software at **desktop 203**) to the first application at the *first electronic device* (i.e., **ECD server 201**) and then on to the second application at the *second electronic device* (e.g., client software on **laptop 202**).

For the same reasons discussed regarding elements 10.1 through 10.3, these limitations are taught by the combination of Hamidi and Ellman. Hamidi discloses that **each** of the client devices are equipped with the software for sending and receiving updates to and from ECD server 201. *Hamidi* (EX1008), [0039]

(explaining that changes files are generated by software on the client computers, including “**both** laptop computer 202 and desktop computer 203”); *see also* [0038], [0040] (describing steps for “a” client file); Abstract; *see also Long Decl.* (EX1005), ¶ 94.



Hamidi (EX1008), Fig. 2 (annotated).

Likewise, Ellman’s system performs two-way synchronization; that is, changes updated on a user’s office computer are synced to their home computer, and vice versa, such that the user’s two computers may both send files to and receive files from one another. *See, e.g., Ellman* (EX1009), 2:20-28 (age of files is used to

determine the direction of syncing); *see also* 12:1-5.

To the extent Hamidi does not expressly disclose that its synchronization steps bi-directional, such would have been obvious over the combination of Hamidi and Ellman, which, as previously discussed, discloses bi-directional synchronization by selecting the most recent file of two computers. Sec. IV.C.1 (overview of Ellman), IV.C.4 (10.1), *supra*. A POSA would have been motivated to make such a combination at least because the prior art provides express motivation or suggestion to combine—Ellman explains that its system allows a user to synchronize files between multiple devices to work remotely, an application recognized by Hamidi. *Ellman* (EX1009), 1:14-2:28; *see also Hamidi* [0004]-[0005]; *Long Decl.* (EX1005), ¶¶ 95-96. Further, such a combination would have simply been a matter of combining prior art elements (Hamidi’s central file distribution system and Ellman’s bi-directional synchronization techniques) according to known methods (Ellman’s method of synchronization using file timestamps) to yield predictable results. *Long Decl.* (EX1005), ¶ 97.

Further, a POSA would have had a reasonable expectation of success in making the combination, at least because Hamidi already implies bi-directional file synchronization, and Ellman references transferring files through a network server. *Long Decl.* (EX1005), ¶ 98; *Ellman* (EX1009), 17:12-17.

[10.7] wherein, responsive to sending the modified first electronic file copy to the third electronic device, an older version of the first electronic file stored on the third electronic device is automatically caused to be replaced with the modified first electronic file copy such that the modified first electronic file copy is stored on the third electronic device in lieu of the older version of the first electronic file, and

This limitation is obvious over the combination of Hamidi and Ellman. Hamidi discloses that client computers that receive a changes file will “modify their copy of the network storage file thereby **generating the new version of the network storage file.**” *Hamidi* (EX1008), [0039].

As modified by Ellman, Hamidi’s system would transfer the entire file instead of a changes file, in which case, upon the new file being sent to the receiving client device (*responsive to sending the modified first electronic file copy to the third electronic device*), the new version would “overwrite” the old version of the file, thereby replacing it (i.e., *an older version of the first electronic file stored on the third electronic device is automatically caused to be replaced with the modified first electronic file copy*) such that the new version is stored instead of the older version (i.e., *such that the modified first electronic file copy is stored on the third electronic device in lieu of the older version of the first electronic file*):

When the user synchronize files, pcTELECOMMUTE copies the file from his office or home PC, depending on which is newer. pcTELECOMMUTE compares the date and time of both files **and overwrites the older file with the newer file.**

Ellman (EX1009), 13:4-9; *see also* 23:4-14 (similar); 8:20-26 (overwriting may be automatic); *see also Long Decl.* (EX1005), ¶ 86 n.15 (noting Ellman’s use of the term “overwrite,” observing it as one known technique to replace files prior to 2007).

For the reasons discussed regarding element 10.1, a POSA would have been motivated to modify, and would have reasonably expected success in modifying, Hamidi’s system to incorporate Ellman’s teachings of transferring an entire file and overwriting an older version. *Long Decl.* (EX1005), ¶¶ 88-90, 93. Such a modification would have required at most minor modifications in software and would have yielded predictable results, as overwriting would have been simpler to implement than applying update instructions from a programming perspective. *Id.*, 93.

[10.8] wherein, responsive to sending the modified second electronic file copy to the second electronic device, an older version of the second electronic file stored on the second electronic device is automatically caused to be replaced with the modified second electronic file copy such that the modified second electronic file copy is stored on the second electronic device in lieu of the older version of the second electronic file.

Element 10.8 is like element 10.7, but for the second file originating from the third device (desktop 203) and sent to the second device (laptop 202) via the first device (ECD server 201). For the reasons discussed regarding elements 10.4-10.7, this limitation is obvious over Hamidi in view of Ellman. As mentioned regarding elements 10.4-10.6, Hamidi and Ellman disclose features that apply to each of the

user's devices; thus, a POSA would have appreciated that the reciprocal nature of the combination would apply to the replacement of the *modified second file copy*. *Long Decl.* (EX1005), ¶¶ 96-98.

4. *Claim 1*

[1P]. A system comprising:

Hamidi discloses “file storage systems.” *Hamidi* (EX1008), [0002]; *see also* [0006], [0039]-[0041].

[1.1]. a first electronic device, associated with a user, configured to:

Hamidi discloses this limitation; alternatively, any differences would have been obvious variations to a POSA. As discussed regarding limitation 10.1, Hamidi discloses that the ECD server (i.e., *a first electronic device*) is where a user's files are temporarily stored and synchronized (*associated with a user*). Sec. IV.C.3 (10.1), *supra*.

[1.1.1] receive, via a first application at the first electronic device, a copy of a modified first electronic file from a second application at a second electronic device associated with the user, wherein the modified first electronic file copy is automatically received from the second application responsive to the user modifying a content of the first electronic file;

For the reasons discussed regarding element 10.1, this limitation is obvious over the combination of Hamidi and Ellman. Sec. IV.C.3 (10.1), *supra*.

[1.1.2] determine whether the first electronic device is in communication with a third electronic device;

For the reasons discussed regarding element 10.2, this limitation is obvious

over the combination of Hamidi and Ellman. Sec. IV.C.4 (10.2), *supra*.

[1.1.3] automatically send, via the first application, the modified first electronic file copy to a third application at the third electronic device responsive to the determination that the first electronic device is in communication with the third electronic device and responsive to receiving the modified first electronic file copy from the second electronic device;

For the reasons discussed regarding element 10.3, this limitation is obvious over the combination of Hamidi and Ellman. Sec. IV.C.3 (10.3), *supra*.

[1.1.4] receive, via the first application, a copy of a modified second electronic file from the third application at the third electronic device associated with the user, wherein the modified second electronic file copy is automatically received from the third application responsive to the user modifying a content of the second electronic file;

[1.1.5] determine whether the first electronic device is in communication with the second electronic device; and

[1.1.6] automatically send, via the first application, the modified second electronic file copy to the second application at the second electronic device responsive to the determination that the first electronic device is in communication with the second electronic device and responsive to receiving the modified second electronic file copy from the third electronic device;

For the reasons discussed regarding elements 10.4-10.6, these limitations are obvious over the combination of Hamidi and Ellman. Sec. IV.C.3 (10.4-10.6), *supra*.

[1.2]¹⁰ wherein, responsive to sending the modified first electronic file copy to the third electronic device, an older version of the first electronic file stored on the third electronic device is automatically caused to be replaced with the modified first electronic file copy such that the modified first electronic file copy is stored on the third electronic device in lieu of the older version of the first electronic file, and

For the reasons discussed regarding element 10.7, this limitation is obvious over the combination of Hamidi and Ellman. Sec. IV.C.3 (10.7), *supra*.

[1.3] wherein, responsive to sending the modified second electronic file copy to the second electronic device, an older version of the second electronic file stored on the second electronic device is automatically caused to be replaced with the modified second electronic file copy such that the modified second electronic file copy is stored on the second electronic device in lieu of the older version of the second electronic file.

For the reasons discussed regarding element 10.8, this limitation is obvious over the combination of Hamidi and Ellman. Sec. IV.C.3 (10.8), *supra*.

¹⁰ These limitations are obvious over the combination of Hamidi and Ellman even if they are sub-elements of limitation 1.1, as it is the ECD server's transmission of the file that causes a copy to be replaced on a receiving client.

5. *Claims 3 and 12*

3. The system of claim 1, wherein the first electronic device is configured to: store, via the first application, the modified first electronic file copy to a memory device associated with the first electronic device.

12. The method of claim 10, further comprising: storing, via the first application, the modified first electronic file copy to a memory device associated with the first electronic device.

Claims 3 and 12 depend from Claims 1 and 10, respectively, which are obvious over the combination of Hamidi and Ellman. Sec. IV.C.3, *supra*. Combined, Hamidi and Ellman teach the additional limitations of Claims 3 and 12.

Hamidi discloses storing (*storing, via the first application*) change files on a storage medium of ECD server 201 (i.e., *a memory device associated with the first electronic device*). Hamidi (EX1008), [0039] (the ECD server “queries the client computer **to automatically copy the changes file to a storage medium of the ECD server 201**” and “**keeps a copy of the changes file**” until all client computers that normally maintain a copy of the file receive it); *see also* [0042] (similar); [0047] (distributed application embodiment); [0050] (identifying hard disc drives, “tape drives, optical drives, RAM, and high capacity flash cards with any type of file system that provides generic files to applications” as appropriate storage media).

As discussed regarding element 10.1, combined with Ellman, Hamidi’s server would store entire files. Sec. IV.C.3 (10.1), *supra*. Thus, Claims 3 and 12 are obvious over the combination of Hamidi and Ellman.

6. *Claims 5 and 14*

5. The system of claim 3, wherein the modified first electronic file copy is stored on the memory device associated with the first electronic device responsive to a determination that the first electronic device is not in communication with the third electronic device.

14. The method of claim 12, wherein the modified first electronic file copy is stored on the memory device associated with the first electronic device responsive to a determination that the first electronic device is not in communication with the third electronic device.

Claims 5 and 14 depend from Claims 3 and 12, respectively, which are obvious over the combination of Hamidi and Ellman. Sec. IV.C.5, *supra*. Combined, Hamidi and Ellman teach the additional limitations of Claims 5 and 14.

As discussed in Ground 2, Hamidi discloses that a received changes file is stored on the storage medium of an ECD server 201, (*stored on the memory device associated with the first electronic device*) until the given client computer that normally maintains a copy of the file returns online and receives the changes (i.e., *responsive to a determination that the first electronic device is not in communication with the third electronic device*). Sec. IV.B.3. And as discussed regarding Claim 10, a POSA would have been motivated to modify, and would have reasonably expected success in modifying, Hamidi's operations to involve an entire file copy, rather than just a changes files, in light of Ellman's teachings. Sec. IV.C.3 (10.1), *supra*. Thus, Claims 5 and 14 are obvious over the combination of Hamidi and Ellman.

7. *Claims 6 and 15*

6. The system of claim 5, wherein the first electronic device is configured to: responsive to resuming communication with the third electronic device, automatically transfer, via the first application, the modified first electronic file copy to the third electronic device to cause the older version of the first electronic file stored on the third electronic device to be replaced with the modified first electronic file copy.

15. The method of claim 14, further comprising: responsive to resuming communication with the third electronic device, automatically transferring, via the first application, the modified first electronic file copy to the third electronic device to cause the older version of the first electronic file stored on the third electronic device to be replaced with the modified first electronic file copy.

Claims 6 and 15 depend from Claims 5 and 14, respectively, which are obvious over the combination of Hamidi and Ellman. Sec. IV.C.6, *supra*. Combined, Hamidi and Ellman teach the additional limitations of Claims 6 and 15.

Hamidi discloses that once the receiving client computer is again in communication with the network (i.e., *responsive to resuming communication with the third electronic device*) the software on the ECD server will automatically send a changes file to the authorized receiving client (*transferring, via the first application . . . to the third electronic device*), which updates its own version of the file (i.e., *to cause the older version of the first electronic file stored on the third electronic device to be replaced with the modified first electronic file copy*).

[T]he ECD server 201 maintains a copy of the changes file and queries the client computers authorized to have the network storage file when those client computers are once again in communication with

the network. Thus, the ECD server 201 keeps a copy of the changes file until all client computers that normally maintain a copy of the network storage file **receive the changes file.**

Hamidi (EX1008), [0039].

As discussed , a POSA would have been motivated to modify, and would have reasonably expected success in modifying, Hamidi's system to incorporate Ellman's teachings related to transferring entire files and overwriting older versions of files on receiving client devices. Sec. IV.C.4 (10.1, 10.7), *supra*. Thus, Claims 6 and 15 are obvious over the combination of Hamidi and Ellman.

V. DISCRETIONARY CONSIDERATIONS

Based on publicly available information, the '942 Patent is asserted in three cases, none in which Petitioner is named. The Board should not deny institution under § 314(a).

Factors 1-2. These factors favor institution or are neutral. All related matters are young; only initial pleadings have been filed by the parties, and it does not appear that a scheduling order has been entered in any case. Whether stays would be granted is uncertain, and no trial dates are set. Further, stays could be granted in proceedings filed after institution.

Factors 3-4, 5. These factors favor institution. No case is pending against Unified, and Unified is unaware of any overlap with related matters other than the

citation to the '942 Patent generally. Unified filed promptly, less than four months after the first known litigation. Unified spent resources preparing this Petition, while Topia appears not to have addressed validity issues yet. Unified does not know what claims are asserted nor what art is known to the parties.

Factor 6. This factor favors institution. The Petition has strong merits, citing art and corroborating evidence not before the examiner during prosecution that goes to the basis of allowance. Additionally, the WIPO International Search Authority's conclusion that bi-directional synchronization was non-inventive warrants further scrutiny of the Challenged Claims.

VI. CONCLUSION

For the foregoing reasons, Petitioner respectfully requests *inter partes* review of Claims 1, 3-6, 10, and 12-15 of U.S. Patent 10,067,942.

Respectfully,

Dated: April 15, 2022

By: /Michelle Aspen/
Michelle Aspen
Reg. No. 75,665

Roshan Mansinghani
Reg. No. 62,429

Counsel for Petitioner

MANDATORY NOTICES

A. Real Parties-in-Interest

Pursuant to 37 C.F.R. § 42.8(b)(1), Petitioner certifies that Unified Patents, LLC is the real party-in-interest, and further certifies that no other party exercised control or could exercise control over Unified's participation in this proceeding, the filing of this petition, or the conduct of any ensuing trial. In view of *Worlds Inc. v. Bungie, Inc.*, 903 F.3d 1237, 1242-44 (Fed. Cir. 2018), Petitioner has submitted voluntary discovery in support of its certification. *Jakel Decl.* (EX1011).

B. Related Matters

Pursuant to 37 C.F.R. § 42.8(b)(2), Petitioner identifies the following related proceedings involving the '942 Patent:

- *Topia Technology, Inc. v. Egnyte, Inc.*, No. 1:21-cv-01821 (D.Del. Dec. 27, 2021);
- *Topia Technology, Inc. v. Dropbox, Inc. et al.*, 6:21-cv-01373 (W.D. Tex. Dec. 29, 2021); and
- *Topia Technology, Inc. v. Box, Inc. et al.*, 6:21-cv-01372 (W.D. Tex. Dec. 29, 2021).

Petitioner identifies the following domestic patent applications that may claim priority to the filing date of the '942 Patent and are known to Petitioner: 16/017,348; 16/361,641; 16/750,399; 16/750,435; 17/306,548.

Petitioner has not previously challenged the '942 Patent, and Petitioner is

unaware of any other challenges involving the particular grounds and evidence discussed herein.

C. Lead and Back-Up Counsel

Pursuant to 37 C.F.R. § 42.8(b)(3)-(4), Petitioner identifies the following designation and service information for lead and back-up counsel. Michelle Aspen will serve as lead counsel. Roshan Mansinghani will serve as first back-up counsel. Petitioner consents to electronic service of documents. Please direct all correspondence regarding this proceeding to lead and back-up counsel at their respective email addresses listed below. 37 C.F.R. § 42.8(b)(4).

Lead Counsel	Back-Up Counsel
Michelle Aspen (Reg. No. 75,665) michelle@unifiedpatents.com Unified Patents, LLC 4445 Willard Ave, Suite 600 Chevy Chase, MD 20815 T: 559.214.3388	Roshan Mansinghani (Reg. No. 62,429) roshan@unifiedpatents.com Unified Patents, LLC 4445 Willard Ave, Suite 600 Chevy Chase, MD 20815 T: 214.953.6737

Dated: April 15, 2022

By: /Michelle Aspen/
Michelle Aspen
Reg. No. 75,665

Counsel for Petitioner

CERTIFICATION OF WORD COUNT

The undersigned certifies pursuant to 37 C.F.R. § 42.24 that the foregoing Petition for *Inter Partes* Review, excluding any table of contents, mandatory notices under 37 C.F.R. §42.8, certificates of service or word count, or appendix of exhibits, contains 13,893 words according to the word-processing program used to prepare this document (Microsoft Word).

Dated: April 15, 2022

By: /Michelle Aspen/
Michelle Aspen
Reg. No. 75,665

Counsel for Petitioner

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. §§ 42.6(e) and 42.105, the undersigned certifies that on April 15, 2022, a complete and entire copy of this Petition for *Inter Partes* Review, including all exhibits listed in the Appendix of Exhibits, as well as the accompanying Power of Attorney, was provided via Priority Mail Express, or by means at least as fast and reliable as Priority Mail Express, to the Patent Owner by serving the counsel of record for the '942 Patent as listed on PAIR:

SUGHRUE MION, PLLC
2000 PENNSYLVANIA AVENUE, N.W.
SUITE 9000
WASHINGTON, DC 20006

Dated: April 15, 2022

By: /Ashley Cheung/
Ashley Cheung

Paralegal for Petitioner