UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

DROPBOX, INC.,

Petitioner

v.

MOTION OFFENSE, LLC,

Patent Owner

IPR2024-00286
U.S. Patent No. 11,611,520

PATENT OWNER'S DEMONSTRATIVE EXHIBITS

Mail Stop PATENT BOARD
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

Dropbox Inc. v. Motion Offense LLC
IPR2024-00286
IPR2024-00287

Patent Owner's Hearing Demonstratives

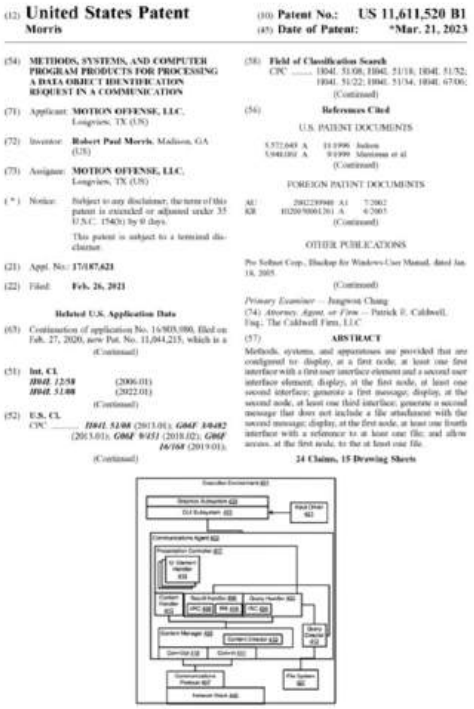
April 23, 2025

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview: 286 IPR

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview



(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,080, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a

(Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 16/182*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/164*** (2019.01);

(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 10200300013761 A 4/2003
(Continued)

OTHER PUBLICATIONS

Pre Software Corp., Hotdrop for Windows User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick F. Caldwell, Esq.; The Caldwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

24 Claims, 15 Drawing Sheets

17. A method, comprising:

[a] causing, at a first node, display of: a first user interface element, for collecting information associated with at least one folder,

[b] a second user interface element, for collecting at least one object associated with at least one email address, the at least one object associated with at least one email address being the at least one email address or an alias associated with the at least one email address, and

[c] a third user interface element, for detecting an indication of a selection thereof to cause an initiation of a sharing of the at least one folder;

[d] causing generation of at least one email, based on the information associated with the at least one folder, the at least one object associated with the at least one email address, and the detection of the indication of the selection of the third user interface element to cause the initiation of the sharing of the at least one folder, where the at least one email: identifies the information associated with the at least one folder, includes an Hypertext Transfer Protocol (HTTP) link, does not include a file attachment, for permitting avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected and the communication commences via at least one server that stores the at least one file, is at least partially pre-written, and is automatically caused to be received without requiring user involvement after the detection of the indication of the selection of the third user interface element to cause the initiation of the sharing of the at least one folder;

[e] receiving, from the second node and [] at least one server, a signal for causing creation of a first representation of the at least one folder, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

[f] causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first representation of the at least one folder that is stored at the at least one server; and

[g] causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second representation of the at least one folder, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

EX1001, Fig. 5)

‘520 Patent Overview

(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a (Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 1020030061261 A 6/2005
(Continued)

OTHER PUBLICATIONS

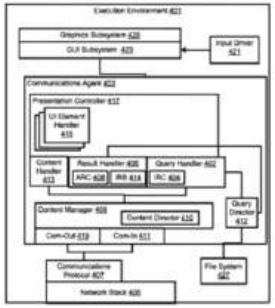
Pro Softnet Corp., IBackup for Windows-User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Caldwell, Esq.; The Caldwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

24 Claims, 15 Drawing Sheets



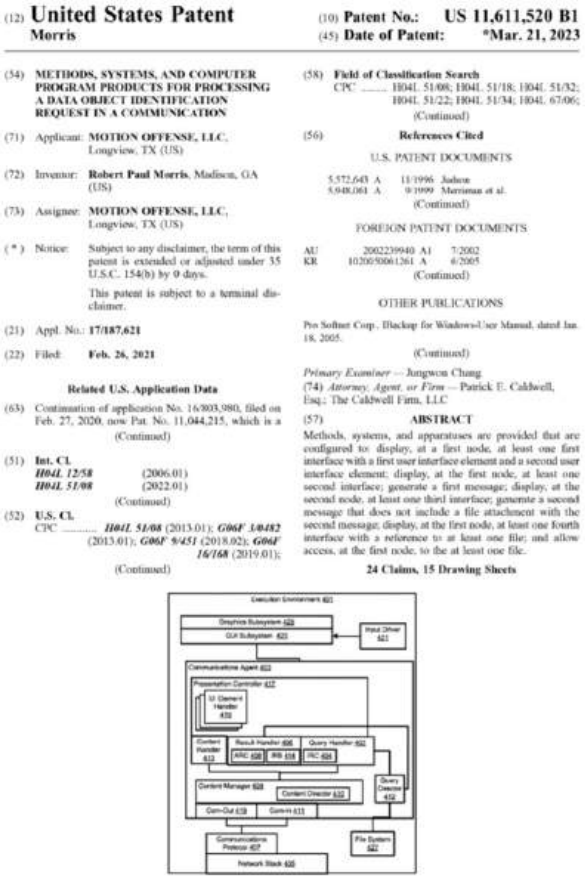
U.S. patent application Ser. No. 13/624,906 (published US 2014-0089419 A1) filed 2012 Sep. 22 and entitled “METHODS, SYSTEMS, AND PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION;” AND

U.S. patent application Ser. No. 13/626,635 (published under US 2014-0089421 A1) filed 2012 Sep. 25 entitled “METHODS, SYSTEMS, AND PROGRAM PRODUCTS FOR SHARING A DATA OBJECT IN A DATA STORE VIA A COMMUNICATION;” the entire contents of each are herein incorporated by reference for all purposes.

(Ex1001 at 1:52-62)

(POR at 28)

‘520 Patent Overview



(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a (Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 10200500611261 A 6/2005
(Continued)

OTHER PUBLICATIONS

Pro Softnet Corp., Backup for Windows-User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Cahlwell, Esq.; The Cahlwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

24 Claims, 15 Drawing Sheets

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

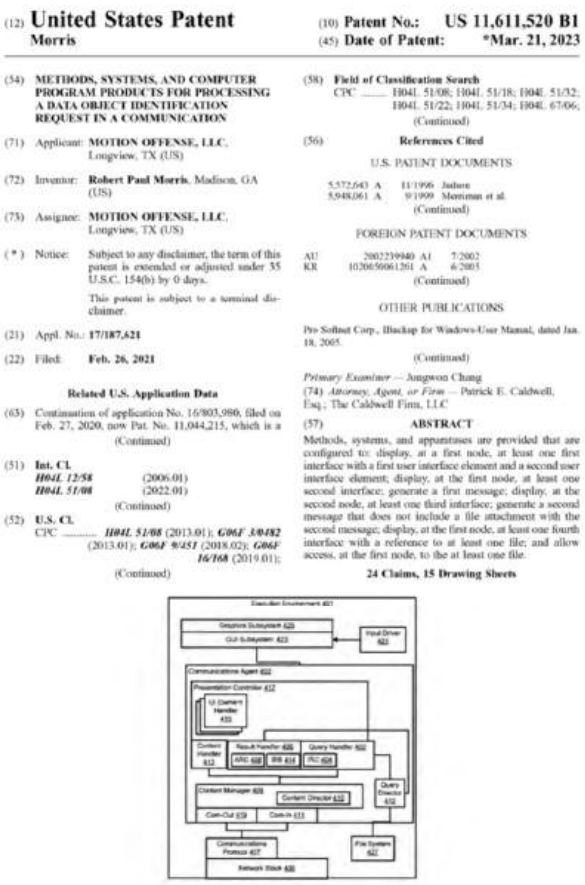
17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

(POR at 28)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview



(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a (Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 1020050061261 A 6/2005
(Continued)

OTHER PUBLICATIONS

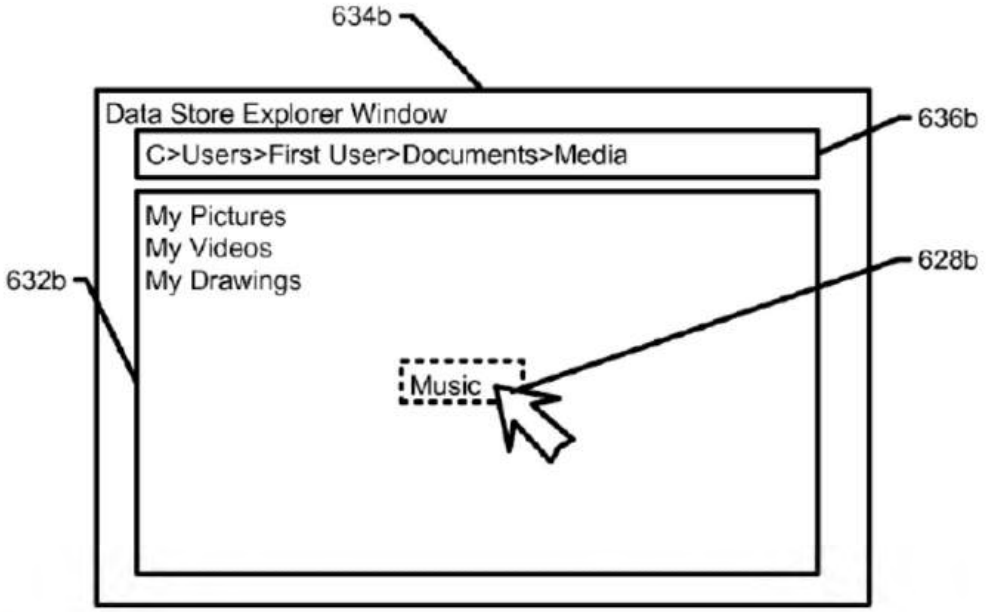
Pro Softnet Corp., Ilbackup for Windows-User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Caldwell, Esq.; The Caldwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

24 Claims, 15 Drawing Sheets

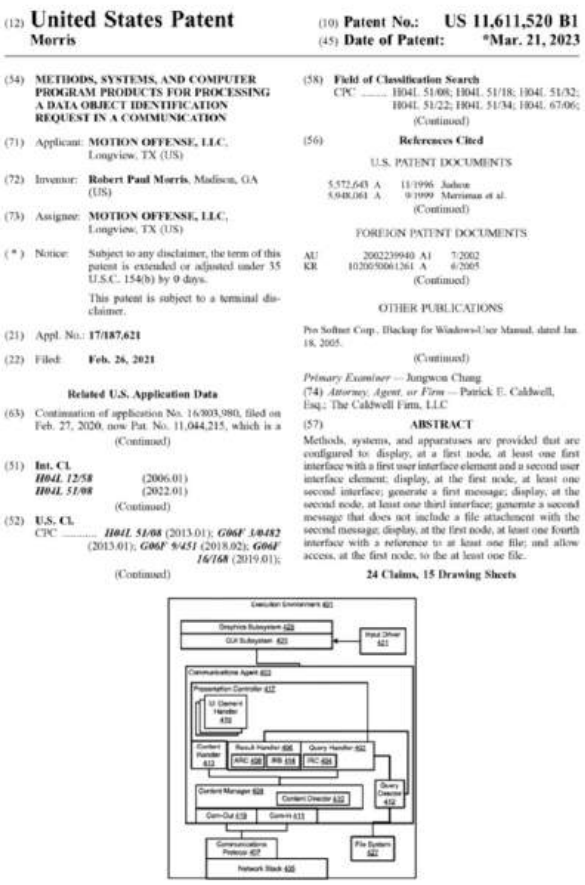


(EX2004 at Fig. 6B.)

(POR at 28)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview



(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a (Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 10200500611261 A 6/2005
(Continued)

OTHER PUBLICATIONS

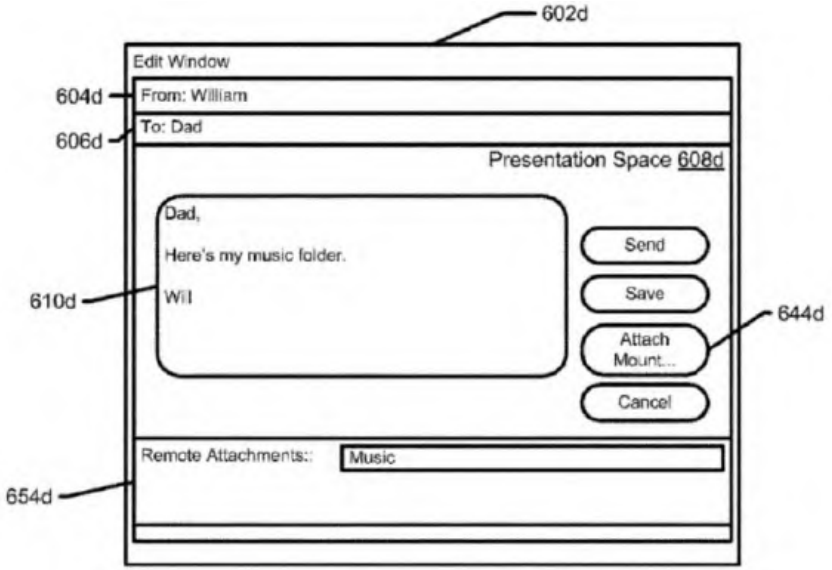
Pro Softnet Corp., Backup for Windows-User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Cahlwell, Esq.; The Cahlwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

24 Claims, 15 Drawing Sheets



(EX2004 at Fig. 6D; EX2051 ¶27.)

(POR at 28)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview

(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a (Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 1020050061261 A 6/2005
(Continued)

OTHER PUBLICATIONS

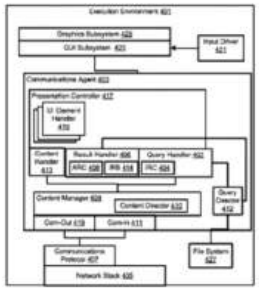
Pro Softnet Corp., Backup for Windows-User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Cahlwell, Esq.; The Cahlwell Firm, LLC

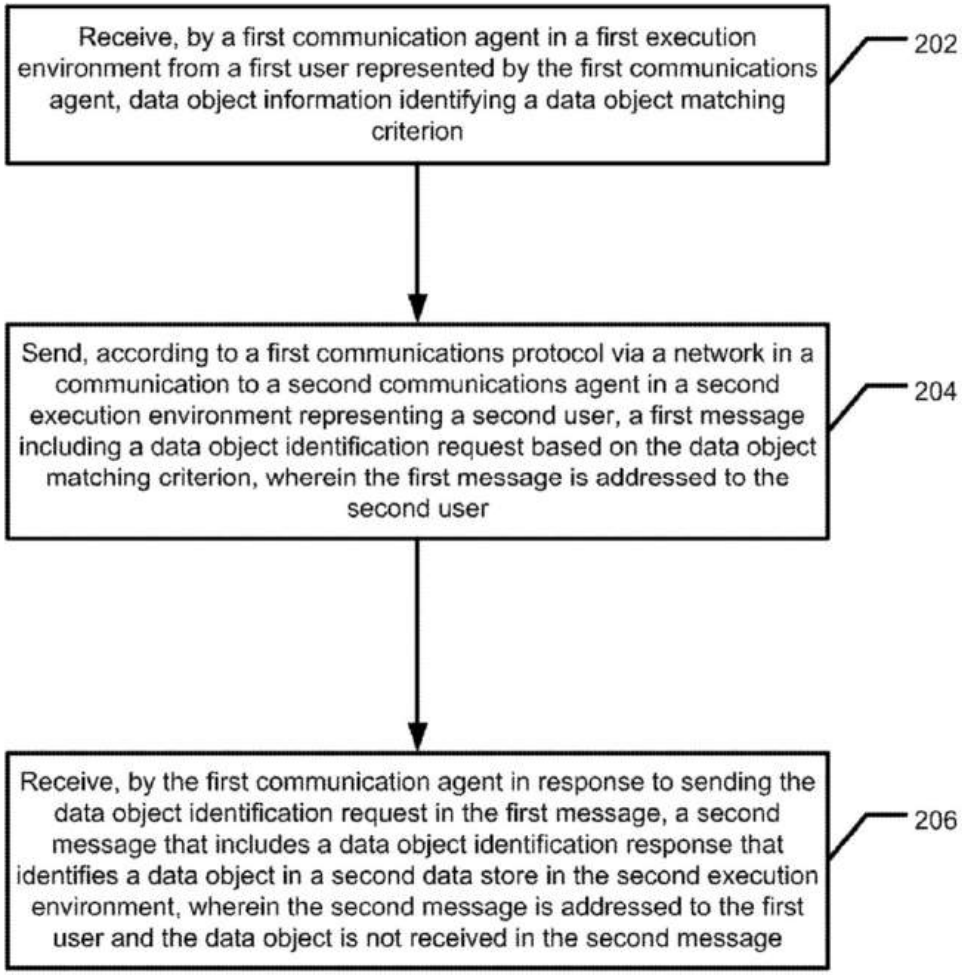
(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

24 Claims, 15 Drawing Sheets

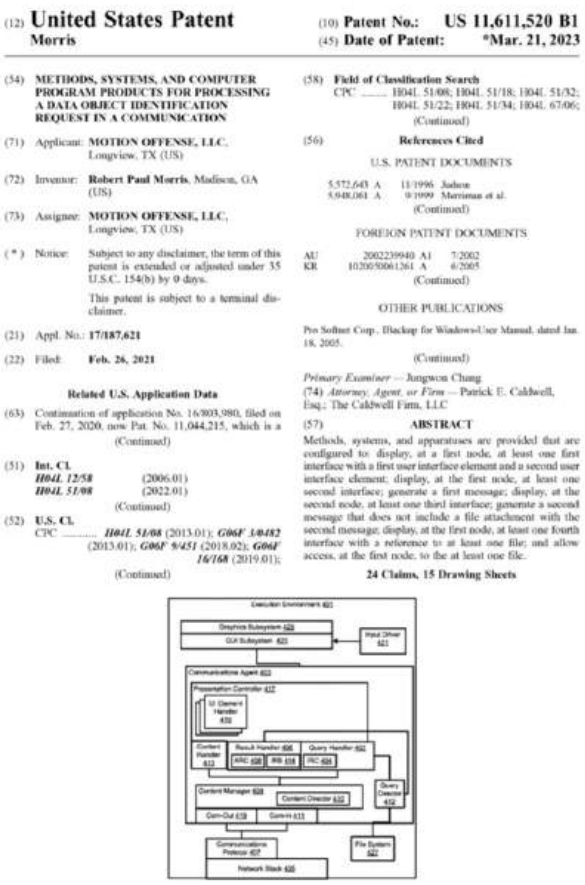


(EX1001, Fig. 2A)



DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview



(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a (Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 1020050061261 A 6/2005
(Continued)

OTHER PUBLICATIONS

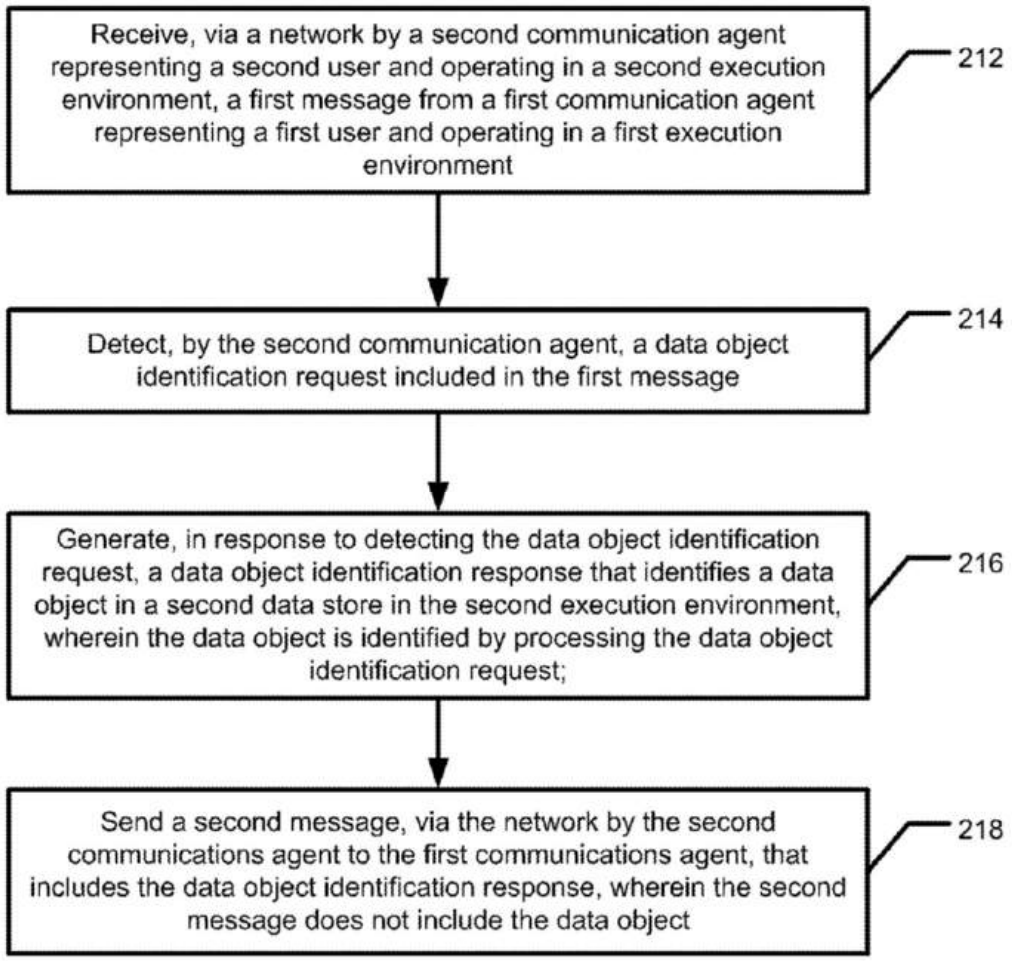
Pro Softnet Corp., Backup for Windows-User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Cahlwell, Esq.; The Cahlwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

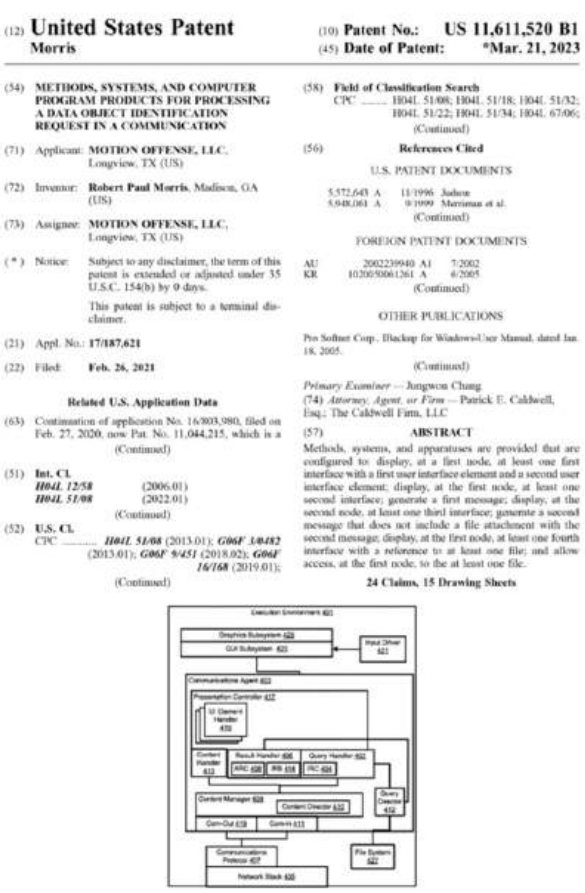
24 Claims, 15 Drawing Sheets



(EX1001, Fig. 2B)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview



(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a (Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUMENTS

AU 2002239940 A1 7/2002
KR 10200500611261 A 6/2005
(Continued)

OTHER PUBLICATIONS

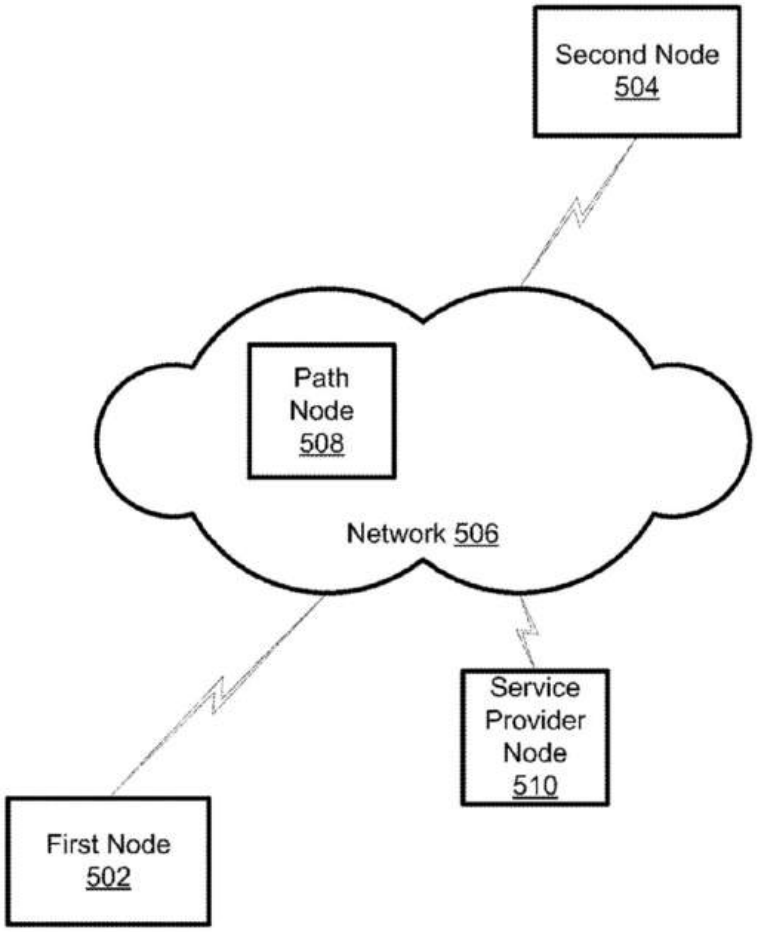
Pro Softnet Corp., Blackup for Windows-User Manual, dated Jan. 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Cahlwell, Esq.; The Cahlwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are provided that are configured to: display, at a first node, at least one first interface with a first user interface element and a second user interface element; display, at the first node, at least one second interface; generate a first message; display, at the second node, at least one third interface; generate a second message that does not include a file attachment with the second message; display, at the first node, at least one fourth interface with a reference to at least one file; and allow access, at the first node, to the at least one file.

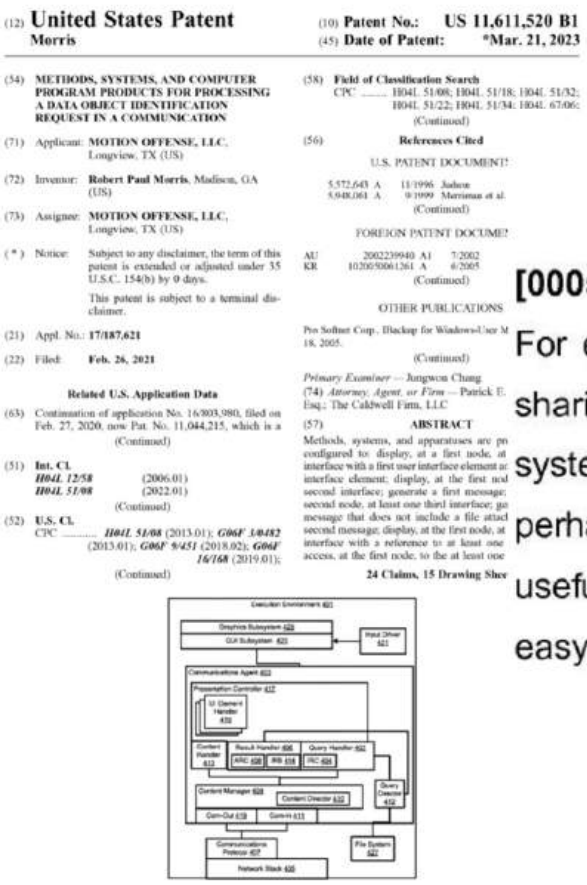
24 Claims, 15 Drawing Sheets



EX1001, Fig. 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview



(12) **United States Patent**
Morris

(10) **Patent No.: US 11,611,520 B1**
(45) **Date of Patent: *Mar. 21, 2023**

(54) **METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR PROCESSING A DATA OBJECT IDENTIFICATION REQUEST IN A COMMUNICATION**

(71) Applicant: **MOTION OFFENSE, LLC**, Longview, TX (US)

(72) Inventor: **Robert Paul Morris**, Madison, GA (US)

(73) Assignee: **MOTION OFFENSE, LLC**, Longview, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a

(Continued)

(51) **Int. Cl.**
H04L 12/58 (2006.01)
H04L 51/08 (2022.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 51/08*** (2013.01); ***G06F 3/0482*** (2013.01); ***G06F 9/451*** (2018.02); ***G06F 16/168*** (2019.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 51/08; H04L 51/18; H04L 51/32; H04L 51/22; H04L 51/34; H04L 67/06;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,643 A 11/1996 Judson
5,948,061 A 9/1999 Merriman et al.
(Continued)

FOREIGN PATENT DOCUME

AU 2002239940 A1 7/2002
KR 1020050061261 A 6/2005
(Continued)

OTHER PUBLICATIONS

Pro Softnet Corp., Backup for Windows-User M 18, 2005.
(Continued)

Primary Examiner — Jungwon Chang
(74) *Attorney, Agent, or Firm* — Patrick E. Esq.; The Caldwell Firm, LLC

(57) **ABSTRACT**

Methods, systems, and apparatuses are pr configured to: display, at a first node, at interface with a first user interface element a interface element; display, at the first nod second interface; generate a first message; second node, at least one third interface; ge message that does not include a file attach second message; display, at the first node, at interface with a reference to at least one access, at the first node, to the at least one

24 Claims, 15 Drawing Shee

BACKGROUND

[0005] A variety of means exist for sharing filed, folders, and other data via network. For each a user must know how to configure and/or setup the sharing. For example, sharing a file a Samba file system is different than sharing a via cloud based storage system, such as DROPBOX. Sharing via email is still different. Sharing copies of files is perhaps the best know means and most widely used means for sharing copies. It would useful if sharing files, folders, directory data, and other network accessible data were as easy as sending an attachment in an email.

(POR at 7-8 (citing EX2004 at [0005]))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘520 Patent Overview

These inventions represented significant improvements over existing systems at the time of invention, such as Dropbox, which was expressly mentioned in the ’635 application. (EX2004 ¶ 5; EX2051 ¶36.) Indeed, as *Houston* teaches, Dropbox and similar prior art systems would automatically download the folder and its contents at the second desktop computer when a folder was shared with it. (EX2051 ¶36.) In other words, when the folder was created at the second desktop computer, it would include all of the folder’s files locally. (*Id.*)

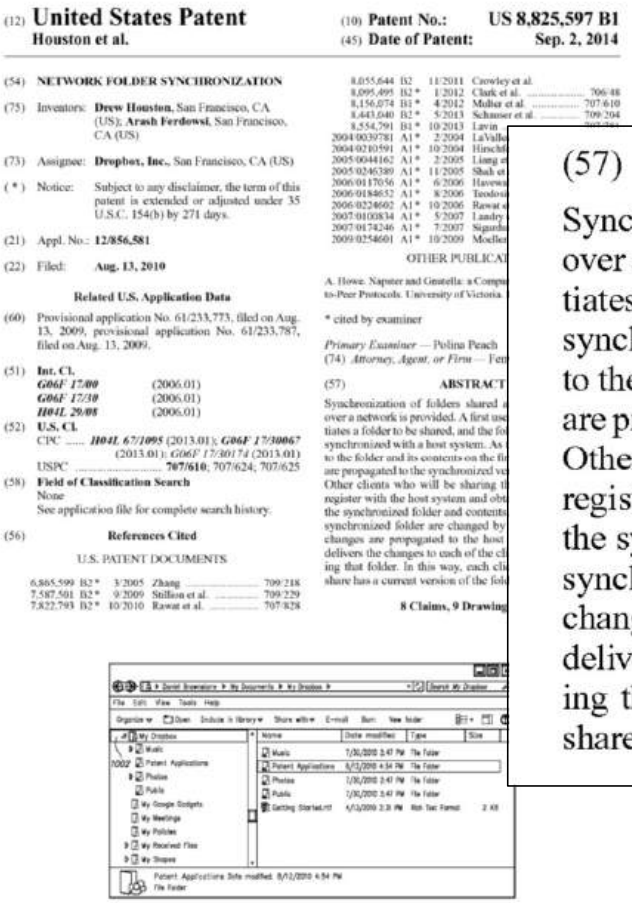
(POR at 8)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston (EX1003) Overview

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview



(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 271 days.

(21) Appl. No.: **12/856,581**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,865,599 B2 * 3/2005 Zhang 709/218
7,587,501 B2 * 9/2009 Stillion et al. 709/229
7,822,793 B2 * 10/2010 Rawat et al. 707/828
8,055,644 B2 11/2011 Crowley et al.
8,095,495 B2 * 1/2012 Clark et al. 706/48
8,156,074 B1 * 4/2012 Moller et al. 707/610
8,443,040 B2 * 5/2013 Schauser et al. 709/204
8,554,791 B1 * 10/2013 Lavin 707/741
2004/0039781 A1 * 2/2004 LaVallee et al. 709/223
2004/0210591 A1 * 10/2004 Hirschfeld et al. 707/100
2005/0044162 A1 * 2/2005 Liang et al. 709/206
2005/0246389 A1 * 11/2005 Shah et al. 707/200
2006/0117056 A1 * 6/2006 Havewala et al. 707/200
2006/0184652 A1 * 8/2006 Teodosiu et al. 709/219
2006/0224602 A1 * 10/2006 Rawat et al. 707/10
2007/0100834 A1 * 5/2007 Landry et al. 707/10
2007/0174246 A1 * 7/2007 Sigurdsson et al. 707/1
2009/0254601 A1 * 10/2009 Moeller et al. 709/203

OTHER PUBLICATIONS

A. Howe. Napster and Gnutella: a Comparison of Two Peer-to-Peer Protocols. University of Victoria.

* cited by examiner

Primary Examiner — Polina Peach
(74) *Attorney, Agent, or Firm* — Fenwick & West LLP

(57) **ABSTRACT**

Synchronization of folders shared among multiple clients over a network is provided. A first user of a first client instantiates a folder to be shared, and the folder and its contents are synchronized with a host system. As the user makes changes to the folder and its contents on the first client, those changes are propagated to the synchronized version on the host server. Other clients who will be sharing the synchronized folder register with the host system and obtain a current version of the synchronized folder and contents. As the contents of the synchronized folder are changed by any of the clients, the changes are propagated to the host system, which in turn delivers the changes to each of the clients registered as sharing that folder. In this way, each client participating in the share has a current version of the folder and its contents.

8 Claims, 9 Drawing Sheets

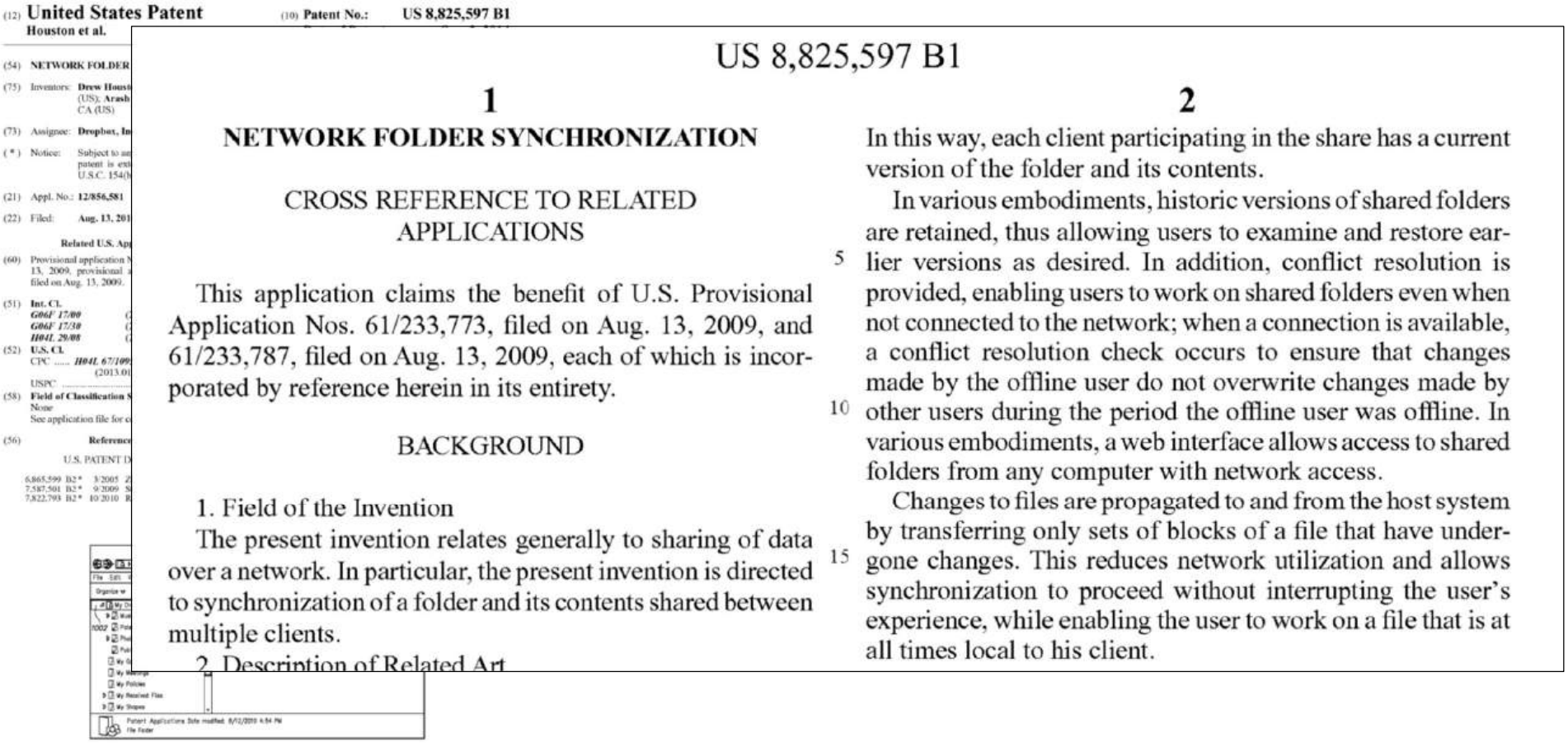
(57) **ABSTRACT**

Synchronization of folders shared among multiple clients over a network is provided. A first user of a first client instantiates a folder to be shared, and the folder and its contents are synchronized with a host system. As the user makes changes to the folder and its contents on the first client, those changes are propagated to the synchronized version on the host server. Other clients who will be sharing the synchronized folder register with the host system and obtain a current version of the synchronized folder and contents. As the contents of the synchronized folder are changed by any of the clients, the changes are propagated to the host system, which in turn delivers the changes to each of the clients registered as sharing that folder. In this way, each client participating in the share has a current version of the folder and its contents.

(POR at 17)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview



(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**

(54) NETWORK FOLDER

(75) Inventors: **Drew Hous**
(US); **Arash**
CA (US)

(73) Assignee: **Dropbox, In**

(*) Notice: Subject to a
patent is ex
U.S.C. 154(

(21) Appl. No.: **12/856,581**

(22) Filed: **Aug. 13, 201**

Related U.S. Ap

(60) Provisional application N
13, 2009, provisional
filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00
G06F 17/30
H04L 29/08

(52) **U.S. Cl.**
CPC ***H04L 67/109***
(2013.01)
USPC .....

(58) **Field of Classification S**
None
See application file for c

(56) **Referenc**

U.S. PATENT D

6,865,599 B2 * 3/2005
7,587,501 B2 * 9/2009
7,822,793 B2 * 10/2010

US 8,825,597 B1

1

NETWORK FOLDER SYNCHRONIZATION

CROSS REFERENCE TO RELATED APPLICATIONS

This application claims the benefit of U.S. Provisional Application Nos. 61/233,773, filed on Aug. 13, 2009, and 61/233,787, filed on Aug. 13, 2009, each of which is incorporated by reference herein in its entirety.

BACKGROUND

1. Field of the Invention

The present invention relates generally to sharing of data over a network. In particular, the present invention is directed to synchronization of a folder and its contents shared between multiple clients.

2. Description of Related Art

2

In this way, each client participating in the share has a current version of the folder and its contents.

In various embodiments, historic versions of shared folders are retained, thus allowing users to examine and restore earlier versions as desired. In addition, conflict resolution is provided, enabling users to work on shared folders even when not connected to the network; when a connection is available, a conflict resolution check occurs to ensure that changes made by the offline user do not overwrite changes made by other users during the period the offline user was offline. In various embodiments, a web interface allows access to shared folders from any computer with network access.

Changes to files are propagated to and from the host system by transferring only sets of blocks of a file that have undergone changes. This reduces network utilization and allows synchronization to proceed without interrupting the user's experience, while enabling the user to work on a file that is at all times local to his client.

(POR at 17-18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview

(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 271 days.

(21) Appl. No.: **12/856,581**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,865,599 B2 * 3/2005 Zhang 709/218
7,587,501 B2 * 9/2009 Stillion et al. 709/229
7,822,793 B2 * 10/2010 Rawat et al. 707/828
8,055,644 B2 11/2011 Crowley et al.
8,095,495 B2 * 1/2012 Clark et al. .....
8,156,074 B1 * 4/2012 Multer et al. .....
8,443,040 B2 * 5/2013 Schauser et al. .....
8,554,791 B1 * 10/2013 Lavin .....
2004/0039781 A1 * 2/2004 LaVallee et al. .....
2004/0210591 A1 * 10/2004 Hirschfeld et al. .....
2005/0044162 A1 * 2/2005 Liang et al. .....
2005/0246389 A1 * 11/2005 Shah et al. .....
2006/0117056 A1 * 6/2006 Havewala et al. .....
2006/0184652 A1 * 8/2006 Teodosiu et al. .....
2006/0224602 A1 * 10/2006 Rawat et al. .....
2007/0100834 A1 * 5/2007 Landry et al. .....
2007/0174246 A1 * 7/2007 Sigurdsson et al. .....
2009/0254601 A1 * 10/2009 Moeller et al. .....

OTHER PUBLICATIONS

A. Howe. Napster and Gnutella: a Comparison of two Peer-to-Peer Protocols. University of Victoria. Feb. 28, 20

* cited by examiner

Primary Examiner — Polina Peach
(74) *Attorney, Agent, or Firm* — Fenwick & W

(57) **ABSTRACT**

Synchronization of folders shared among mu over a network is provided. A first user of a first tiates a folder to be shared, and the folder and it synchronized with a host system. As the user m to the folder and its contents on the first client, are propagated to the synchronized version on th Other clients who will be sharing the synchr register with the host system and obtain a curr the synchronized folder and contents. As the c synchronized folder are changed by any of th changes are propagated to the host system, delivers the changes to each of the clients regis ing that folder. In this way, each client partic share has a current version of the folder and its

8 Claims, 9 Drawing Sheets

2. Description of Related Art

People often use multiple computers on a regular basis. A typical user may have a first computer at the office and a second computer at home, for example. Sharing documents between these multiple computers generally requires transferring the document from one to the other—for example, a user may e-mail himself a copy of a document he is working on before leaving the office, so that he can resume working on it later from home. If the user forgets to e-mail or bring the document home with him, he must either go back to the office to retrieve it, or perhaps give up until the morning. Some methods exist to allow remote access to a computer, for example using a virtual private network (VPN) to access a corporate network from a remote location. However, if the user is accessing the document remotely and loses his connection, he may lose his changes, be unable to continue, and may end up with a corrupted document.

(POR at 17-18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview

(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 271 days.

(21) Appl. No.: **12/856,581**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

Synchronizing Across Multiple Clients

As described above, a client **108** with a synchronized folder informs host system **110** when a file in a synchronized folder has been added, modified or deleted. Other clients may also have versions of the same synchronized folder, which are updated via host system **110** as follows.

Referring again to FIG. **1**, assume that a user of a first client **108***a* has created a folder and invited a user of client **108***b* to share the folder. The folder is immediately synchronized on host system **110** as described above. In addition, both client **108***a* and client **108***b* are noted by notification server **106** as being associated with that folder. In one embodiment, each of the clients registers with notification server **106**; in alternative embodiments notification server is informed by metadata server **102** or by the originating client **108***a* of the sharing relationship.

When metadata server **102** receives and successfully executes a commit instruction, notification server **106** in one embodiment informs all clients subscribed to that folder that the folder contents have changed. In an alternative embodiment, the client that initiated the change is not informed of the change.

Upon receiving the notification, each client **108** sends a list request to metadata server **102**, and in response receives file metadata for all files in subscribed folders. The client then examines the block list for each file and identifies any listed blocks that the client does not already have in its database **502**. File transfer module **510** then asks block server **104** for a patch from the parent block the client is in possession of to the new block the client needs. Block server **104** creates the patch and provides it to client **108** in response to the request. Client **108** then applies the patch to the parent block to obtain the updated block, which it then stores. Client **108** repeats the process for each block that needs updating. At the conclusion of the process, the client's version of the file is synchronized with the updated version on host system **110**.

(POR at 17 (EX1003 at 5:37-6:3))

Houston Overview

(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 271 days.

(21) Appl. No.: **12/856,583**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,797, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

OTHER PUBLICATIONS

* cited by examiner

Primary Examiner — Pelisia Peach
(74) *Attorney, Agent, or Firm* — Fenwick & West LLP

(57) **ABSTRACT**

8 Claims, 9 Drawing Sheets

Assume now that the invited user has an existing namespace, "3:". Assuming the user accepts the invitation to share the folder, metadata server **102** creates a link in the 3: namespace, such that "3:/Patent Applications" points to namespace 2:. Metadata server **102** also adds the invited user's identifier to the list of users sharing the folder, and notification server **106** begins providing change notifications to the invited user's client. The invited user's client then obtains the latest version of the synchronized file according to the methods described above.

(POR at 18 (EX1003 at)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview

(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 251 days.

(21) Appl. No.: **12/856,501**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

OTHER PUBLICATIONS

(57) **ABSTRACT**

8 Claims, 9 Drawing Sheets

A single user may have multiple computers, each of which may or may not be running the same operating system. Using system **100**, the user can maintain documents and files in a synchronized folder, and have the contents of that folder available to him regardless of which of his computers and at which location he happens to be at the moment he needs them, without having to worry about which version is available on which computer.

(POR at 19 (EX1003 at 8:14-21))

Houston Overview

(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 251 days.

(21) Appl. No.: **12/856,501**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

OTHER PUBLICATIONS

(57) **ABSTRACT**

8 Claims, 9 Drawing Sheets

A single user may have multiple computers, each of which may or may not be running the same operating system. Using system **100**, the user can maintain documents and files in a synchronized folder, and have the contents of that folder available to him regardless of which of his computers and at which location he happens to be at the moment he needs them, without having to worry about which version is available on which computer.

(POR at 19 (EX1003 at 8:14-21))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview

(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 271 days.

(21) Appl. No.: **12/856,501**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,775, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

OTHER PUBLICATIONS

* cited by examiner

Primary Examiner — Pelitia Pinch
(74) *Attorney, Agent, or Firm* — Fenwick & West LLP

(57) **ABSTRACT**

Synchronization of folders shared among multiple clients over a network is provided. A first user of a first client instantiates a folder to be shared, and the folder and its contents are synchronized with a host system. As the user makes changes to the folder and its contents on the first client, those changes are propagated to the synchronized version on the host server. Other clients who will be sharing the synchronized folder register with the host system and obtain a current version of the synchronized folder and contents. As the contents of the synchronized folder are changed by any of the clients, the changes are propagated to the host system, which in turn delivers the changes to each of the clients registered as sharing that folder. In this way, each client participating in the share has a current version of the folder and its contents.

8 Claims, 9 Drawing Sheets

Assume now that the user indicates that he wishes to share the "Patent Applications" folder. In one embodiment, a user can indicate this through a user interface command, such as by right-clicking on the folder name and selecting a "Share This Folder . . . " option **1102**, as illustrated in FIG. **11**. Alternatively, the user can use a web interface to communicate the share instructions to host system **110**. In either event, the user also specifies the account identifier of the user(s) with whom the folder is to be shared.

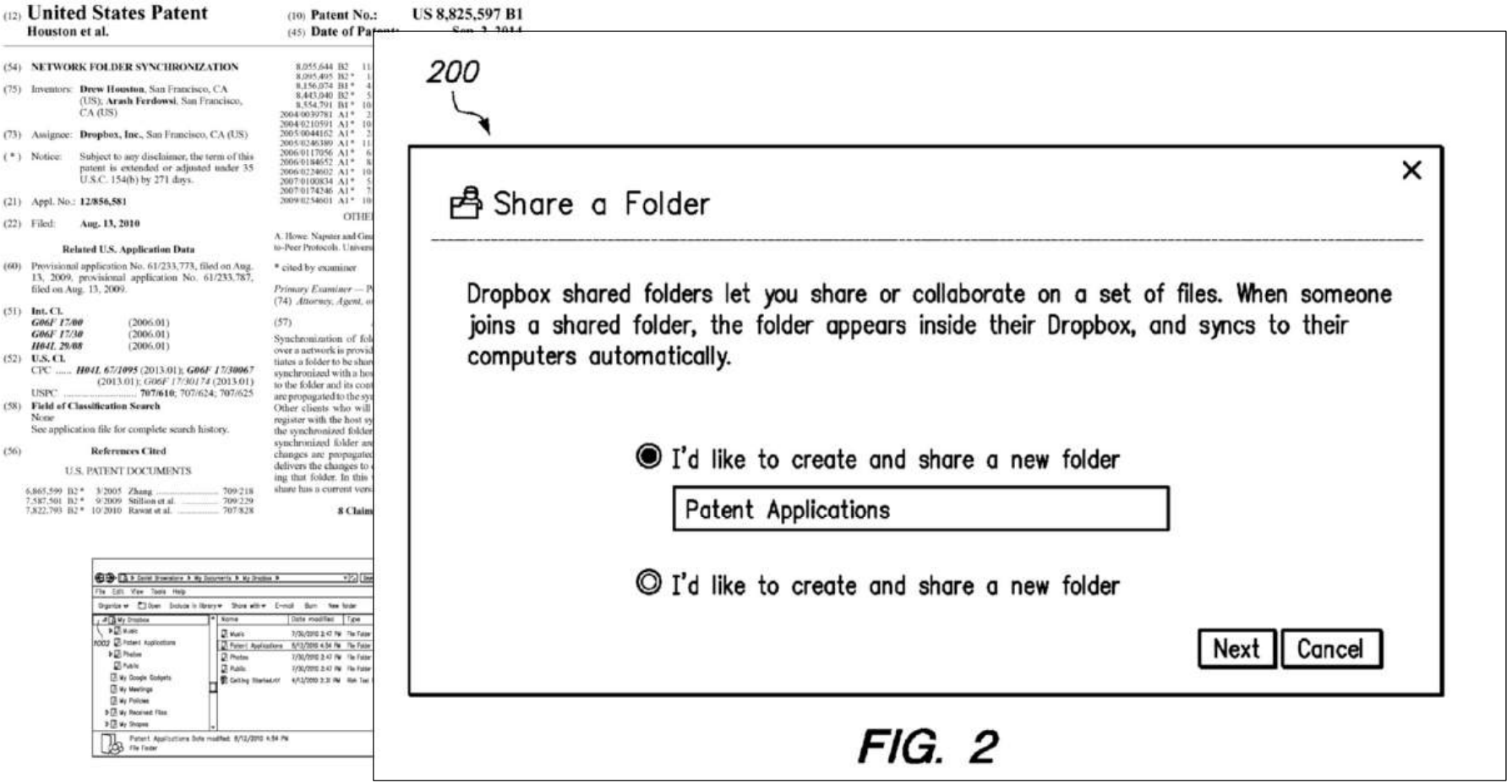
Metadata server **102** receives the share instruction and moves the subfolder "Patent Applications" from the path "1:/Patent Applications" to a new namespace, which we will call "2:". Metadata server **102** then creates a mapping from the namespace "1:/Patent Applications" to the namespace "2:", and instructs the client to do the same. Note that from the point of view of the user, nothing appears to have changed in the user interface.

Assume now that the invited user has an existing namespace, "3:". Assuming the user accepts the invitation to share the folder, metadata server **102** creates a link in the 3: namespace, such that "3:/Patent Applications" points to namespace 2:. Metadata server **102** also adds the invited user's identifier to the list of users sharing the folder, and notification server **106** begins providing change notifications to the invited user's client. The invited user's client then obtains the latest version of the synchronized file according to the methods described above.

(Sur-Reply at 8-9 (EX1003 at 7:38-58))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview



(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent:** Sep. 2, 2014

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 271 days.

(21) Appl. No.: **12/856,581**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,865,599 B2 * 3/2005 Zhang 709/218
7,587,501 B2 * 9/2009 Stillion et al. 709/229
7,822,793 B2 * 10/2010 Rawat et al. 707/828

8,055,644 B2 11
8,095,495 B2 * 1
8,156,074 B1 * 4
8,443,040 B2 * 5
8,554,791 B1 * 10
2004/0039781 A1 * 2
2004/0210591 A1 * 10
2005/0044162 A1 * 2
2005/0246389 A1 * 11
2006/0117056 A1 * 6
2006/0184652 A1 * 8
2006/0224602 A1 * 10
2007/0100834 A1 * 5
2007/0174246 A1 * 7
2009/0254601 A1 * 10

OTHE

A. Howe. Napster and Gnu to-Peer Protocols. Univers

* cited by examiner

Primary Examiner — P
(74) *Attorney, Agent,* o

(57)

Synchronization of fol over a network is provid tiates a folder to be shar synchronized with a ho to the folder and its cont are propagated to the sy Other clients who will register with the host sy the synchronized folder synchronized folder ar changes are propagate delivers the changes to ing that folder. In this share has a current vers

8 Claim

200

Share a Folder

Dropbox shared folders let you share or collaborate on a set of files. When someone joins a shared folder, the folder appears inside their Dropbox, and syncs to their computers automatically.

I'd like to create and share a new folder

Patent Applications

I'd like to create and share a new folder

Next Cancel

FIG. 2

(POR at 17-18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Overview

(12) **United States Patent**
Houston et al.

(10) **Patent No.: US 8,825,597 B1**
(45) **Date of Patent: Sep. 2, 2014**

(54) **NETWORK FOLDER SYNCHRONIZATION**

(75) Inventors: **Drew Houston**, San Francisco, CA (US); **Arash Ferdowsi**, San Francisco, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 271 days.

(21) Appl. No.: **12/856,581**

(22) Filed: **Aug. 13, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/233,773, filed on Aug. 13, 2009, provisional application No. 61/233,787, filed on Aug. 13, 2009.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 67/1095*** (2013.01); ***G06F 17/30067*** (2013.01); *G06F 17/30174* (2013.01)
USPC **707/610**; 707/624; 707/625

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,865,599 B2 * 3/2005 Zhang 709/218
7,587,501 B2 * 9/2009 Stillion et al. 709/229
7,822,793 B2 * 10/2010 Rawat et al. 707/828
8,055,644 B2 11/2011 Crowley et al.
8,095,495 B2 * 1/2012 Clark et al. 706/48
8,156,074 B1 * 4/2012 Multer et al. 707/610
8,443,040 B2 * 5/2013 Schauser et al. 709/204
8,554,791 B1 * 10/2013 Lavin 707/781
2004/0039781 A1 * 2/2004 LaValley et al. 709/205
2004/0210591 A1 * 10/2004 Hirschfeld et al. 707/100
2005/0044162 A1 * 2/2005 Liang et al. 7
2005/0246389 A1 * 11/2005 Shah et al. 7
2006/0117056 A1 * 6/2006 Havewala et al. 7
2006/0184652 A1 * 8/2006 Teodosiu et al. 7
2006/0224602 A1 * 10/2006 Rawat et al. 7
2007/0100834 A1 * 5/2007 Landry et al. 7
2007/0174246 A1 * 7/2007 Sigurdsson et al. 7
2009/0254601 A1 * 10/2009 Moeller et al. 7

OTHER PUBLICATIONS

A. Howe. Napster and Gnutella: a Comparison of two Popula to-Peer Protocols. University of Victoria. Feb. 28, 2002. pp.

* cited by examiner

Primary Examiner — Polina Peach
(74) *Attorney, Agent, or Firm* — Fenwick & West LL

(57) **ABSTRACT**

Synchronization of folders shared among multiple over a network is provided. A first user of a first client tiates a folder to be shared, and the folder and its conte synchronized with a host system. As the user makes cl to the folder and its contents on the first client, those cl are propagated to the synchronized version on the host Other clients who will be sharing the synchronized register with the host system and obtain a current ver the synchronized folder and contents. As the content synchronized folder are changed by any of the clien changes are propagated to the host system, which delivers the changes to each of the clients registered ing that folder. In this way, each client participating share has a current version of the folder and its conte

8 Claims, 9 Drawing Sheets

1

NETWORK FOLDER SYNCHRONIZATION

CROSS REFERENCE TO RELATED APPLICATIONS

This application claims the benefit of U.S. Provisional Application Nos. 61/233,773, filed on Aug. 13, 2009, and 61/233,787, filed on Aug. 13, 2009, each of which is incorporated by reference herein in its entirety.

(Sur-Reply at 9-10 (EX1003 at 1:6-9))

Garcia (EX1004) Overview

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pintus**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathius Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**
U.S. PATENT DOCUMENTS

5,008,853 A * 4/1991 Bly G06
5,862,346 A 1/1999 Kley et al.
6,348,932 B1 2/2002 Nishikawa et al.
6,407,750 B1 6/2002 Gioscia et al.
6,636,873 B1 10/2003 Carini et al.
6,636,888 B1 10/2003 Bookspan et al.
6,904,185 B1 6/2005 Wilkins et al.
7,039,723 B2 5/2006 Hu et al.
7,103,357 B2 9/2006 Kirani et al.
7,139,555 B2 11/2006 Apfel
7,319,536 B1 1/2008 Wilkins et al.
7,321,919 B2 1/2008 Jacobs et al.
7,369,161 B2 5/2008 Easwar et al.
7,372,485 B1 5/2008 Bodnar et al.
7,441,180 B1 10/2008 Kaczmarek et al.
(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are pr... enabling a user to synchronize, manage, and shar... folders across a plurality of client devices and a ... zation server. In a user interface, a first view d... folders, which a user has synchronized to the sy... tion server and a drop-target zone to which the u... to drag a folder from the local device to synch... folder with the synchronization server. A second ... plays all files and folders, which the user has s... other users. A third view displays a list of user ... where each contact is a drop-target zone on whic... able to drag a folder to share the folder with the ... fourth view is provided in which a user is able to ... related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

(POR at 19)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pinius**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

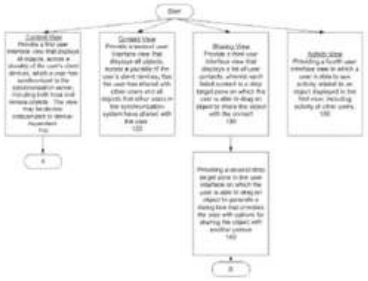
5,008,853 A * 4/1991 Bly G06F 17/30171 345/2.1
5,862,346 A 1/1999 Kley et al.
6,348,932 B1 2/2002 Nishikawa et al.
6,407,750 B1 6/2002 Gioscia et al.
6,636,873 B1 10/2003 Carini et al.
6,636,888 B1 10/2003 Bookspan et al.
6,904,185 B1 6/2005 Wilkins et al.
7,030,723 B2 5/2006 Ha et al.
7,103,357 B2 9/2006 Kirani et al.
7,130,555 B2 11/2006 Apfel
7,319,536 B1 1/2008 Wilkins et al.
7,321,919 B2 1/2008 Jacobs et al.
7,369,161 B2 5/2008 Easwar et al.
7,372,485 B1 5/2008 Bodnar et al.
7,441,180 B1 10/2008 Kaczmarek et al.
(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets



This invention relates generally to a synchronization system and, more particularly, to a system and method for enabling a user to synchronize, manage, and share folders across a plurality of client devices and a synchronization server.

2. Description of the Background Art

Many users have multiple computing devices, such as home computer, a work computer, and a mobile phone. Synchronization systems enable users to synchronize files on one device to another device. Some synchronization systems utilize a simple synchronization process, where a user adds any files he wishes to synchronize to a single folder, and the folder is synchronized to all of the user's devices. Other synchronization systems provide more enhanced capability and enable users to specify which folders/files will be synchronized to which devices.

The synchronization systems with the enhanced functionality have the advantage of enabling greater control and file management by the user. However, setting up and managing synchronization in these settings can also be more complex, especially for those new to the technology. Therefore, there is a need for an intuitive system and method for enabling a user to easily synchronize, manage, and share folders across a plurality of client devices in a synchronization system that provides enhanced synchronization functionality.

SUMMARY OF THE INVENTION

The present invention is directed to a system, method, and computer program for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE
(POR at 19 (EX1004 at 1:12-42))

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pintus**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,864**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 850, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,008,853 A * 4/1991 Bly G06F 17/30171 345/2.1
5,862,346 A 1/1999 Kley et al.
6,348,932 B1 2/2002 Nishikawa et al.
6,407,750 B1 6/2002 Gioscia et al.
6,636,873 B1 10/2003 Carini et al.
6,636,888 B1 10/2003 Bookspan et al.
6,904,185 B1 6/2005 Wilkins et al.
7,039,723 B2 5/2006 Hu et al.
7,103,357 B2 9/2006 Kirani et al.
7,139,555 B2 11/2006 Apfel
7,310,536 B1 12/2007 Wilkins et al.
7,321,919 B2 1/2008 Jacobs et al.
7,360,161 B2 5/2008 Tuovinen et al.
7,372,485 B1 5/2008 Bednar et al.
7,441,180 B1 10/2008 Kaczmarek et al.
(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets

FIGS. **20***a*-**20***b* illustrate a method, according to one embodiment, for enabling a user to access and edit, via a virtual drive, local and remote objects in a synchronization system, including objects synchronized to a plurality of synchronization clients. In the preferred embodiment, the objects are folders that may contain documents (e.g., EXCEL, POWERPOINT, or WORD documents), email/calendar items, video files, and/or photos, but those skilled in the art will appreciate that the method of the present invention may be applied to other objects.

Referring to FIG. **20***a*, a user is enabled to synchronize one or more folders on any synchronization client to the synchronization server and other synchronization clients (step **2005**). All folders, across a plurality of synchronization clients that a user has backed up or synchronized to the synchronization server, are identified, as well as any folders that the user shares with other users (step **2010**). Each of the synchronization clients stores metadata with information on all folders shared with or by the user and all folders that the user has backed up or synced to the synchronization server (including the contents of the folder). The synchronization client on the client device uses this metadata to generate the virtual drive and the user interface.

(POR at 20 (EX1004 at 5:14-26))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pinlus**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,008,853	A *	4/1991	Bly	G06F 17/30171 345/2.1
5,862,346	A	1/1999	Kley et al.	
6,348,932	B1	2/2002	Nishikawa et al.	
6,407,750	B1	6/2002	Gioscia et al.	
6,636,873	B1	10/2003	Carini et al.	
6,636,888	B1	10/2003	Bookspan et al.	
6,904,185	B1	6/2005	Wilkins et al.	
7,030,723	B2	5/2006	Ha et al.	
7,103,357	B2	9/2006	Kirani et al.	
7,130,555	B2	11/2006	Apfel	
7,319,536	B1	1/2008	Wilkins et al.	
7,321,919	B2	1/2008	Jacobs et al.	
7,369,161	B2	5/2008	Easwar et al.	
7,372,485	B1	5/2008	Bodnar et al.	
7,441,180	B1	10/2008	Kaczmarek et al.	

(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets

FIG. **9** illustrates the Web link tab **335**. The Web link tab **335** includes a list of folders or files that have been shared as a link with other users. In this example, a list of files is shown, which is sorted by the date the link was enabled 905. The list may be sorted using other criteria as well. Each file has an icon **910**, which indicates the number of times the file has been downloaded, and an icon **915**, which places the link onto the clipboard to be copied into an email, text, or other transmittal. One benefit of sending links to folders or files to other users is it allows other users to access the folders or files without requiring the other user to create an account in the synchronization system.

(POR at 20-21 (EX1004 at 10:32-43))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pinlus**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,008,853	A *	4/1991	Bly	G06F 17/30171 345/2.1
5,862,346	A	1/1999	Kley et al.	
6,348,932	B1	2/2002	Nishikawa et al.	
6,407,750	B1	6/2002	Gioscia et al.	
6,636,873	B1	10/2003	Carini et al.	
6,636,888	B1	10/2003	Bookspan et al.	
6,904,185	B1	6/2005	Wilkins et al.	
7,030,723	B2	5/2006	Ha et al.	
7,103,357	B2	9/2006	Kirani et al.	
7,130,555	B2	11/2006	Apfel	
7,319,536	B1	1/2008	Wilkins et al.	
7,321,919	B2	1/2008	Jacobs et al.	
7,369,161	B2	5/2008	Easwar et al.	
7,372,485	B1	5/2008	Bodnar et al.	
7,441,180	B1	10/2008	Kaczmarek et al.	

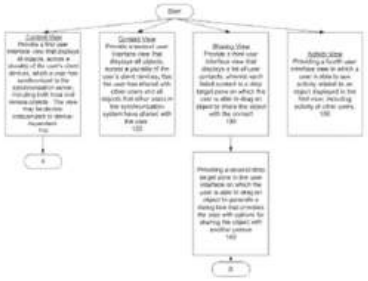
(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets



FIGS. **11-12** illustrate the functionality of the sharing view **310**. FIG. **11** has a list of contacts **1105**, each of which is a drop-target zone, and a separate dedicated drop-target zone **1110**. The list of contacts **1105** may be compiled from the user's MICROSOFT OUTLOOK contacts, mobile device contacts, or any other contacts. In one embodiment, when a user drags a folder to a particular contact, the folder or a link to the folder **1115** is automatically sent (via email, text, or other means) to the contact. In the preferred embodiment, when the user drags a folder **1115** to a particular contact in the contact list, a dialog box is automatically displayed that provides a user with options for sharing the folder with the contact. An example of such a dialog box is shown in FIG. **12**. In the illustrated example, sharing the folder **1115** may be by web link **1205**, by email **1210**, or by social networking **1215**. The user contact information is automatically populated into the "Recipients" field. When the user clicks on the send link button **1220**, a link to folder **1115** is sent to the contact. When the contact clicks on the link to folder **1115**, the contents of the folder are shared with the contact. The user can control the sharing level **1225** of the folder being sent to the contact (e.g., add and edit access or read-only access). The user may also create groups of contacts and share folders with a designated group(s).

In an alternate embodiment, if the user desires to share the folder **1115** with a contact not on the contact list **1105**, the user drags the folder **1115** to the separate dedicated drop-target zone **1110**, and a dialog box is generated that provides the user with share options. The user may then type in the contact's information (e.g., email address) in to the applicable field in the dialog box, and a link will be sent to the contact via web link **1205**, email **1210**, or by social networking **1215**.

(POR at 20-21 (EX1004 at 11:11-43))

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pinlus**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,008,853 A * 4/1991 Bly G06F 17/30171 345/2.1
5,862,346 A 1/1999 Kley et al.
6,348,932 B1 2/2002 Nishikawa et al.
6,407,750 B1 6/2002 Gioscia et al.
6,636,873 B1 10/2003 Carini et al.
6,636,888 B1 10/2003 Bookspan et al.
6,904,185 B1 6/2005 Wilkins et al.
7,030,723 B2 5/2006 Ha et al.
7,103,357 B2 9/2006 Kirani et al.
7,130,555 B2 11/2006 Apfel
7,319,536 B1 1/2008 Wilkins et al.
7,321,919 B2 1/2008 Jacobs et al.
7,369,161 B2 5/2008 Easwar et al.
7,372,485 B1 5/2008 Bodnar et al.
7,441,180 B1 10/2008 Kaczmarek et al.
(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets

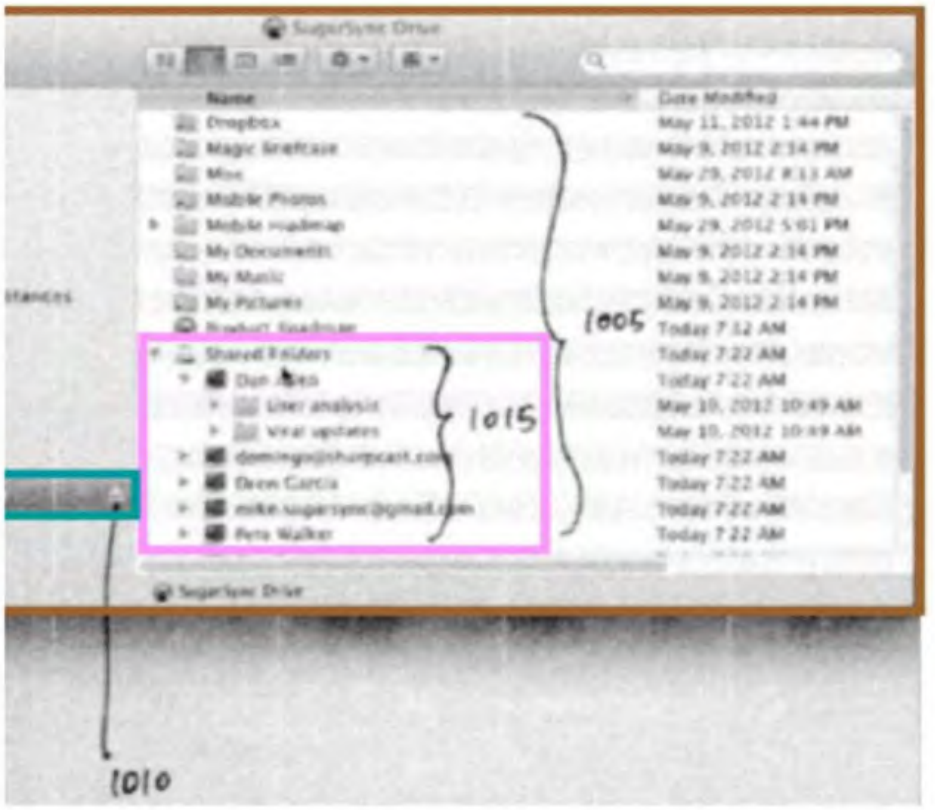


Exhibit 1004, Figure 10 (excerpted and annotated)

(POR at 20 (EX1004 at Fig. 10))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pinlou**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01; 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,008,853	A *	4/1991	Bly	G06F 17/30171 345/2.1
5,862,346	A	1/1999	Kley et al.	
6,348,932	B1	2/2002	Nishikawa et al.	
6,407,750	B1	6/2002	Gioscia et al.	
6,636,873	B1	10/2003	Carini et al.	
6,636,888	B1	10/2003	Bookspan et al.	
6,904,185	B1	6/2005	Wilkins et al.	
7,030,723	B2	5/2006	Ha et al.	
7,103,357	B2	9/2006	Kirani et al.	
7,130,555	B2	11/2006	Apfel	
7,319,536	B1	1/2008	Wilkins et al.	
7,321,919	B2	1/2008	Jacobs et al.	
7,369,161	B2	5/2008	Easwar et al.	
7,372,485	B1	5/2008	Bodnar et al.	
7,441,180	B1	10/2008	Kaczmarek et al.	

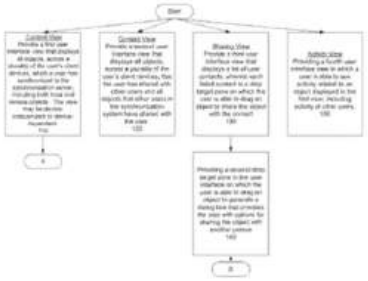
(Continued)

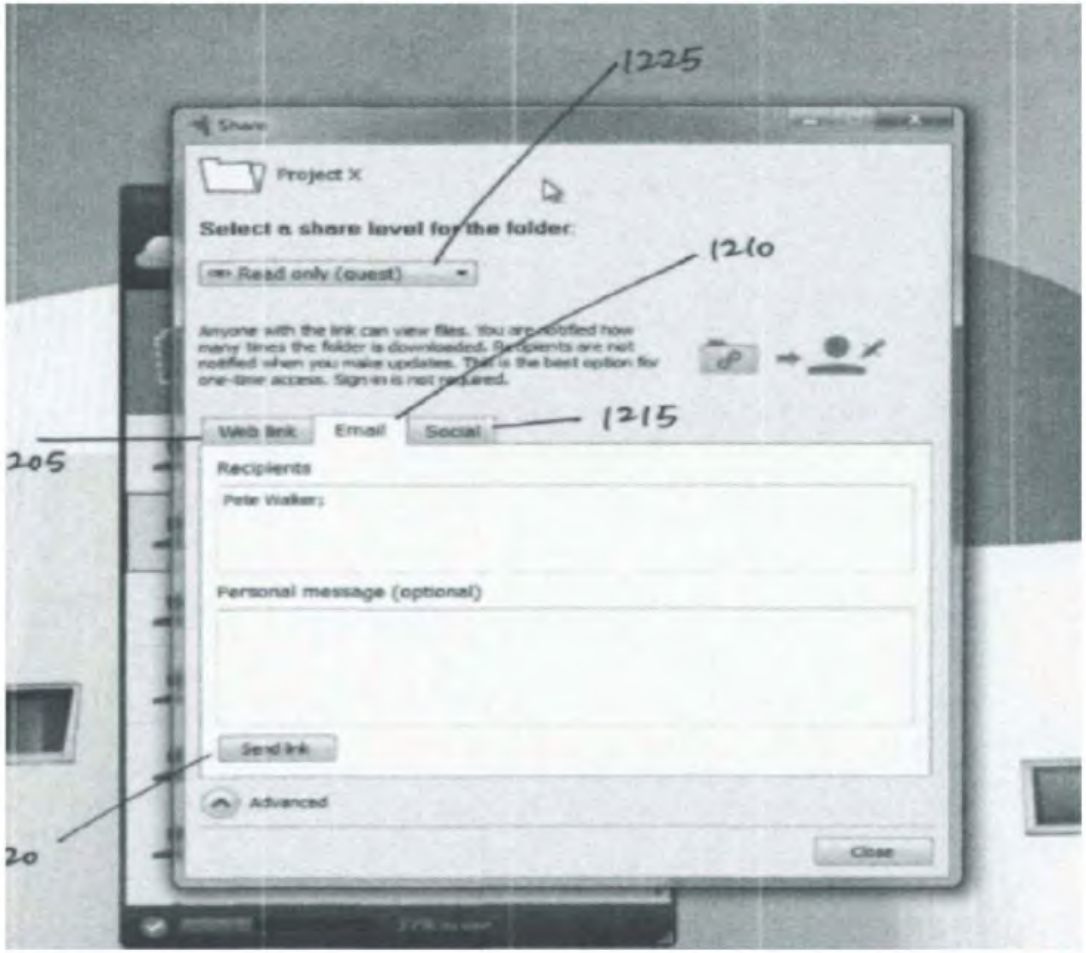
Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets





(POR at 22 (EX1004 at Fig. 12; *see also EX2052*))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Overview

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pinlus**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,008,853 A * 4/1991 Bly G06F 17/30345
5,862,346 A 1/1999 Kley et al.
6,348,932 B1 2/2002 Nishikawa et al.
6,407,750 B1 6/2002 Gioscia et al.
6,636,873 B1 10/2003 Carini et al.
6,636,888 B1 10/2003 Bookspan et al.
6,904,185 B1 6/2005 Wilkins et al.
7,030,723 B2 5/2006 Ha et al.
7,103,357 B2 9/2006 Kirani et al.
7,130,555 B2 11/2006 Apfel
7,319,536 B1 1/2008 Wilkins et al.
7,321,919 B2 1/2008 Jacobs et al.
7,369,161 B2 5/2008 Easwar et al.
7,372,485 B1 5/2008 Bodnar et al.
7,441,180 B1 10/2008 Kaczmarek et al.
(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided enabling a user to synchronize, manage, and share files folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize folder with the synchronization server. A second view displays all files and folders, which the user has shared other users. A third view displays a list of user contacts where each contact is a drop-target zone on which a user able to drag a folder to share the folder with the contact fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets

Houston in combination with *Garcia* does not disclose elements 17[d]-[g]. Indeed, Petitioner has admitted that *Garcia* discloses that sharing folders using links provides the content even when the recipient does not have an account. (Pet. 31 ("*Garcia* expressly disclosed a benefit of sending an email message with a link to a shared folder—'One benefit of sending links to folders or files to other users is it allows other users to access the folders or files without requiring the other user to create an account in the synchronization system.'")(quoting EX1004, 10:40–43);

(Sur-Reply at 14-15)

Garcia Originally Assigned to SugarSync

(12) **United States Patent**
Garcia et al.

(10) **Patent No.: US 9,633,125 B1**
(45) **Date of Patent: Apr. 25, 2017**

(54) **SYSTEM, METHOD, AND COMPUTER PROGRAM FOR ENABLING A USER TO SYNCHRONIZE, MANAGE, AND SHARE FOLDERS ACROSS A PLURALITY OF CLIENT DEVICES AND A SYNCHRONIZATION SERVER**

(75) Inventors: **Andrew Garcia**, San Francisco, CA (US); **Samir Bhupatrai Mehta**, Menlo Park, CA (US); **Athena Anagnostopoulos**, San Mateo, CA (US); **Fuad Rashid**, Foster City, CA (US); **Laura Susan Fried Yecies**, Woodside, CA (US); **Alice Viola Pindus**, San Francisco, CA (US); **Jason Edward Short**, San Francisco, CA (US); **Elie Mathias Monge**, San Francisco, CA (US); **James David Wong**, San Jose, CA (US)

(73) Assignee: **Dropbox, Inc.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 17 days.

(21) Appl. No.: **13/571,564**

(22) Filed: **Aug. 10, 2012**

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/30884*** (2013.01)

(58) **Field of Classification Search**
USPC 715/738, 733, 734, 748, 751, 234, 762, 715/810, 839, 853, 854; 707/620, 999.01, 707/E17.005, E17.032; 709/248, 203
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,008,853	A *	4/1991	Bly	G06F 17/30171 345/2.1
5,862,346	A	1/1999	Kley et al.	
6,348,932	B1	2/2002	Nishikawa et al.	
6,407,750	B1	6/2002	Gioscia et al.	
6,636,873	B1	10/2003	Carini et al.	
6,636,888	B1	10/2003	Bookspan et al.	
6,904,185	B1	6/2005	Wilkins et al.	
7,030,723	B2	5/2006	Ha et al.	
7,103,357	B2	9/2006	Kirani et al.	
7,130,555	B2	11/2006	Apfel	
7,319,536	B1	1/2008	Wilkins et al.	
7,321,919	B2	1/2008	Jacobs et al.	
7,369,161	B2	5/2008	Easwar et al.	
7,372,485	B1	5/2008	Bodnar et al.	
7,441,180	B1	10/2008	Kaczmarek et al.	

(Continued)

Primary Examiner — Linh K Pham
Assistant Examiner — Wendy Nicholas
(74) *Attorney, Agent, or Firm* — Arent Fox LLP

(57) **ABSTRACT**

A system, method, and computer program are provided for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server. In a user interface, a first view displays all folders, which a user has synchronized to the synchronization server and a drop-target zone to which the user is able to drag a folder from the local device to synchronize the folder with the synchronization server. A second view displays all files and folders, which the user has shared with other users. A third view displays a list of user contacts, where each contact is a drop-target zone on which a user is able to drag a folder to share the folder with the contact. A fourth view is provided in which a user is able to see activity related to a folder displayed in the first view.

19 Claims, 26 Drawing Sheets

(POR at 19; EX2009)

2. *Garcia*

U.S. Patent No. 9,633,125 to *Garcia* et al. ("*Garcia*," EX1004) is a patent originally assigned to SugarSync, Inc. that was assigned to Dropbox in 2014. (EX2008–2009.) *Garcia* purports to describe the operation of the then-existing SugarSync folder-sharing application. (EX1004 at Figs. 3–19; EX2051 ¶57.)

SUBMISSION TYPE:	NEW ASSIGNMENT
NATURE OF CONVEYANCE:	ASSIGNMENT

CONVEYING PARTY DATA

Name	Execution Date
SUGARSYNC, INC.	10/14/2014

RECEIVING PARTY DATA

Name:	DROPBOX, INC.
Street Address:	185 BERRY ST.
Internal Address:	4TH FLOOR
City:	SAN FRANCISCO
State/Country:	CALIFORNIA
Postal Code:	94107

Application Number:	13571564

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

SugarSync Documentation

Receiving a Shared Folder Invitation

If someone shares a SugarSync folder with you, you can access and download the files from the SugarSync Website, or you can sync the folder to your computer. After syncing to your computer, you can locate the shared folder in **Documents > SugarSync Shared Folders**.

NOTE

If you change the location of the SugarSync Shared Folders directory, or move any of its subfolders, SugarSync will not be able to update your shared files. To access your shared folders from a different location on your computer, you can create a shortcut (or an Alias for Mac) to the folder or any of the subfolders in any location. For more information, see Creating Shortcuts to Shared Folders.

When you send an invitation to a shared folder, your recipients receive an email with a link button.

If they don't have a SugarSync account, they can sign up for a free account.

After creating an account, SugarSync walks them through the SugarSync Manager installation. As soon as the installation completes, the folder syncs to the recipient's computer in a new folder created automatically by SugarSync: **Documents > SugarSync Shared Folders**.

After the folder is synced to the recipient's computer, SugarSync keeps the shared files up-to-date in everyone's shared folder no matter who makes changes to the files.

(EX2011 at 75–76 (annotated); EX2010 at 78; EX2051 ¶61.)

(POR at 23)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano (EX1005) Overview

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Overview

(19) **United States**
(12) **Patent Application Publication**
Manzano
(10) **Pub. No.: US 2010/0005138 A1**
(43) **Pub. Date: Jan. 7, 2010**

(54) **ELECTRONIC FILE SHARING**

(75) Inventor: **Michael R. Manzano**, Seattle, WA (US)

Correspondence Address:
BLACK LOWE & GRAHAM, PLLC
701 FIFTH AVENUE, SUITE 4800
SEATTLE, WA 98104 (US)

(73) Assignee: **Topia Technology**, Tacoma, WA (US)

(21) Appl. No.: **12/427,540**

(22) Filed: **Apr. 21, 2009**

Related U.S. Application Data

(60) Provisional application No. 61/046,725, filed on Apr. 21, 2008.

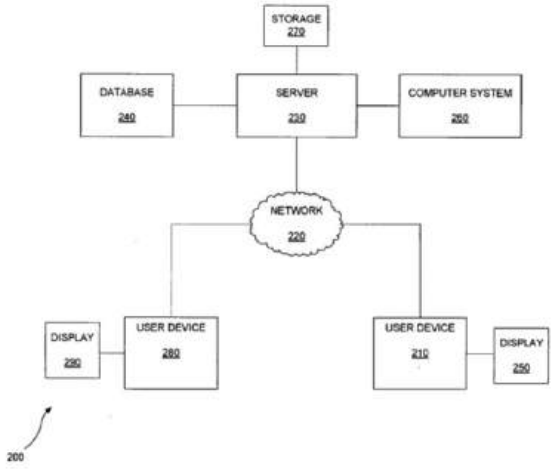
Publication Classification

(51) **Int. Cl.**
G06F 15/16 (2006.01)

(52) **U.S. Cl.** **709/203**; 709/219

(57) **ABSTRACT**

A system includes a server and a source client executable on a first electronic device in communication with the server. The source client is operable to make available to the server a set of files modifiable by a user of the first electronic device. The system further includes a first recipient client executable on a second electronic device in communication with the server. The first recipient client is configurable by a user of the second electronic device to automatically receive from the server only a subset of the files having a predetermined characteristic.



U.S. Pat. Appl. Pub. No. 2010/0005138 to Manzano (EX1005) ("*Manzano*") is titled "Electronic File Sharing" and purports to address problems involving "exchang[ing] large files using awkward prior-art methods such as e-mail attachments . . . ." (EX1005 ¶ [0003]; EX2051 ¶62.) *Manzano* teaches sharing files between devices using "mobile agents" or a "mobile agent object" that exists on a first computer system and can "transplant itself" to a second computer system, while preserving its current execution state. (EX1005 ¶ [0016]; EX2051 ¶62.) This is performed by the mobile object creating another mobile object that functions as a "proxy object" on the server. (EX1005 ¶ [0035]; EX2051 ¶62.)

BACKGROUND OF THE INVENTION

[0003] Computer-network users typically exchange large files using awkward prior-art methods such as e-mail attachments or file transfer protocol (FTP) sites. Most email systems limit the size of attachments, and, as a result, users are forced to send multiple e-mails with discrete files attached to each. FTP sites are typically only moderately secure and their use is often difficult to manage for inexperienced users.

(POR at 24-25)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Overview

EX2051 ¶63.) *Manzano* teaches that files may be shared with the second computer by placing them—such as by dragging and dropping them—in a designated folder. (EX1005 ¶¶ [0033], [0046]–[0049]; EX2051 ¶63.) The server transfers portions of

[0048] Referring to FIG. **8**, in an embodiment of the invention, the user may drag (as indicated by the dashed line **820**), using a pointer or other selection device, such as mouse **161**, or otherwise place into the interface **350**, or other user interface, such as an electronically generated graphical window, a first icon **810** displayable on a screen **700** of the display device **250** and representing a file to be transferred. In an embodiment, the icon **810** may represent a copy of the file, rather than, or in addition to, the file itself, thereby allowing transfer of the file copy rather than the file itself. The user may additionally drag or otherwise place (as indicated by the dashed line **840**) into the interface **350** a second icon **830** (e.g., an icon of a "v-card" including an email address) displayable on the display device **250** and representing a recipient. Dragging the first and second icons **810**, **830** into the interface **350** automatically causes the file associated with the first icon **810** to be transferred to the recipient associated with the second icon **830**. Similarly, dragging a third icon **850** (as indicated by the dashed line **860**) representing an additional recipient into the interface **350** automatically causes the file associated with the first icon **810** to be transferred to the recipient associated with the third icon **850**. Such file transfers described with reference to FIG. **8** may be effected using techniques described with reference to FIGS. **1-5** elsewhere herein.

[0049] Referring to FIG. **9**, in an embodiment of the invention, the user may drag or otherwise place (as indicated by the dashed line **920**) a first icon **910** on or partially coinciding with (e.g., superimposed on) a second icon **930** displayable on a screen **700** of the display device **250** and, perhaps, located in the interface **350**. The first icon **910** may represent a file to be transferred. In an embodiment, the first icon **910** may represent a copy of the file, rather than, or in addition to, the file itself, thereby allowing transfer of the file copy rather than the file itself. The second icon **930** may represent a recipient (i.e., may be an icon of a "v-card" including an email address). At least partially superimposing one of the first and second icons **910**, **930** with the other automatically causes the file associated with the first icon **910** to be transferred to the recipient associated with the second icon **930**. This transfer operation may similarly be performed with respect to a third icon **950** also representing a recipient. For example, the second and third icons **930**, **950** may be simultaneously selected, placed in close proximity to one another, or otherwise coupled using known graphical-user-interface techniques. Subsequently, at least partially superimposing the first icon **910** on the second and/or third icon **930**, **950** automatically causes the file associated with the first icon to be transferred to the recipients associated with the second and third icons. Such file transfers described with reference to FIG. **9** may be effected using techniques described with reference to FIGS. **1-5** elsewhere herein.

(POR at 24-25)

Manzano Overview

(19) **United States**
(12) **Patent Application Publication**
Manzano
(10) **Pub. No.: US 2010/0005138 A1**
(43) **Pub. Date: Jan. 7, 2010**

(54) **ELECTRONIC FILE SHARING**

(75) Inventor: **Michael R. Manzano**, Seattle, WA (US)

Correspondence Address:
BLACK LOWE & GRAHAM, PLLC
701 FIFTH AVENUE, SUITE 4800
SEATTLE, WA 98104 (US)

(73) Assignee: **Topia Technology**, Tacoma, WA (US)

(21) Appl. No.: **12/427,540**

(22) Filed: **Apr. 21, 2009**

Related U.S. Application Data

(60) Provisional application No. 61/046,725, filed on Apr. 21, 2008.

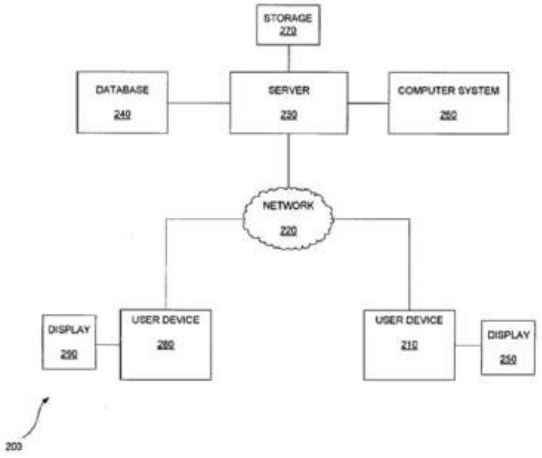
Publication Classification

(51) **Int. Cl.**
G06F 15/16 (2006.01)

(52) **U.S. Cl.** **709/203**; 709/219

(57) **ABSTRACT**

A system includes a server and a source client executable on a first electronic device in communication with the server. The source client is operable to make available to the server a set of files modifiable by a user of the first electronic device. The system further includes a first recipient client executable on a second electronic device in communication with the server. The first recipient client is configurable by a user of the second electronic device to automatically receive from the server only a subset of the files having a predetermined characteristic.



[0050] In an embodiment, file icons, such as the icon **600**, in a workspace displayed on display device **290** can visually indicate whether a corresponding file (such as a file, the transfer of which may be as is discussed herein with reference to FIG. **3**) has been downloaded to or is otherwise stored on (i.e., is local to), or has otherwise been made available (e.g., sent or in transit) to the user device **280**. For example, if the source client **310** has initiated, or has otherwise attempted to initiate, the sharing of a file with the recipient client **320**, the interface **360** associated with the user device **280** may display an icon, such as icon **600**, identifying or otherwise representing the file to be shared by source client **310**. If the corresponding file (i.e., file to be shared by source client **310**) is not local to user device **280**, the icon **600** may be ghosted (i.e., semi-transparent) or otherwise be visually configured to indicate the non-local status of the file. As such, the icon **600** can appear in the workspace in some manner whether or not the data associated with the icon **600** has actually been downloaded to, or otherwise received by, the user device **280**.

(POR at 24-25)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Overview

(19) **United States**
(12) **Patent Application Publication** (10) **Pub. No.: US 2010/0005138 A1**
Manzano (43) **Pub. Date: Jan. 7, 2010**

(54) **ELECTRONIC FILE SHARING**

(75) Inventor: **Michael R. Manzano**, Seattle, WA (US)

Correspondence Address:
BLACK LOWE & GRAHAM, PLLC
701 FIFTH AVENUE, SUITE 4800
SEATTLE, WA 98104 (US)

(73) Assignee: **Topia Technology**, Tacoma, WA (US)

(21) Appl. No.: **12/427,540**

(22) Filed: **Apr. 21, 2009**

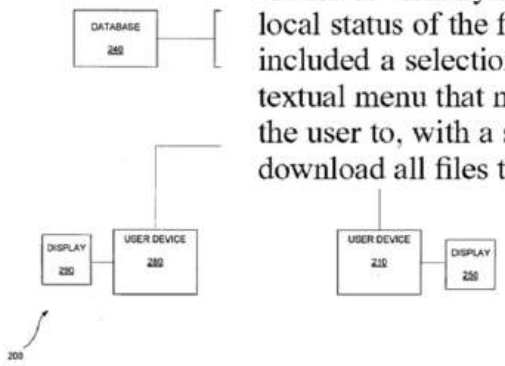
Related U.S. Application Data

(60) Provisional application No. 61/046,725, filed on Ap 21, 2008.

Publication Classification

(51) **Int. Cl.**
G06F 15/16 (2006.01)

[0051] In such a situation, and in an embodiment, a user may right-click or otherwise indicate the icon **600**, and from a resultant pop-up contextual menu (not shown), for example, that appears on the display **290**, make a selection from the menu to request receipt or download of the associated file, at which time the icon **600** will become unghosted or will otherwise be visually configured to indicate the pending-local or local status of the file. In an embodiment, there may also be included a selection feature, accessible from a pop-up contextual menu that may be generated to display **290**, to allow the user to, with a single selection from the menu, similarly download all files that are not local to user device **280**.



[0052] In an embodiment, files can be downloaded to a user device **280** according to a user-defined "subscription." With such a subscription, users can elect, on a per-workspace basis, to always download all files that have been made available to the user, or never download all such files. For purposes of this description, the "workspace" corresponds to a respective electronic folder assigned to a particular user group, as discussed in further detail above herein with reference to FIG. **3**. As such, access to each workspace is limited to a predefined and finite set of users on a given network. Files, when created or modified on, for example, user device **210** can be categorized according to particular characteristics of each file such as, by way of non-limiting example, by author, by source, or by subject matter. Users can also be able to specify a subset of the files (e.g., by author, by source, by subject matter, etc.) to always download. In this context, the term "always download" means that when someone remotely updates a file (e.g., modifies or creates a file), such file will be downloaded to a user-specified machine (e.g., user device **280**) without specific instruction by the recipient. As such, an embodiment provides a mechanism for specifying the manner in which files stored on a remote computing device are automatically shared or otherwise synchronized with a local computing device, and vice-versa.

(POR at 24-25)

Claim Construction:
“Representation of the at least one folder”

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

"Representation of the at least one folder" Is a Special Type of Folder

Term	PO's construction	Petitioner's Construction[2]
"representation of the at least one folder"	"a folder that can be displayed at the user's node, but whose file contents are stored remotely and are not communicated to the user till the user initiates the communication"	"some sign or symbol that stands in place of (i.e., represents) the 'at least one folder'" "a stand-in image or symbol that depicts, but that is not itself, a folder"

(Sur-Reply at 2; POR at 28)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Proposed Constructions Are Result-Driven and Based on Hindsight

First, no construction is needed. This straightforward 7-word claim term simply refers to some sign or symbol that stands in place of (i.e., represents) the "at least one folder" referred to in the claims. There is no reason this term requires construction, much less PO's proposed tortured 30-word construction. EX1034, ¶¶7-8.

• • •

inconsistent with the claim language. Elements 17[e]-[g] discuss "display[ing]" the "representation of the at least one folder" on a webpage or file explorer application, suggesting that, consistent with the claim term's plain meaning, the "representation" of the folder is a stand-in image or symbol that depicts, but that is not itself, a folder.

(Pet. Rep. at 2)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Claim Language Confirms a "Representation" Is Not Simply a "Sign," "Symbol," "Icon," or "Stand-in"

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

(POR at 29)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Claim Language Confirms a "Representation" Is Not Simply a "Sign," "Symbol," "Icon," or "Stand-in"

First, these claim elements show that a "representation of the at least one folder" is itself a folder because it can be stored "in a location among one or more folders," either at the "at least one server" (in 17[e] and [f]) or at the "second node" (in 17[g]). (EX2051 ¶77.)

(POR at 30)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Claim Language Confirms a "Representation" Is Not Simply a "Sign," "Symbol," "Icon," or "Stand-in"

Second, these claim elements show that there are two representations of a folder—a "first" and "second" representation—and both representations can be displayed at the user's node. (EX2051 ¶78.) The first representation is "displayable" at the user's node (i.e., the "second node") "via at least one web page" (in 17[e] and [f]), and the second representation is "displayable" at the user's node "via the file explorer interface of the client-based file explorer application" (in 17[g]). (*Id.*)

(POR at 30)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Claim Language Confirms a "Representation" Is Not Simply a "Sign," "Symbol," "Icon," or "Stand-in"

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

(POR at 31)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Claim Language Confirms a "Representation" Is Not Merely a "Sign," "Symbol," "Icon," or "Stand-in"

With this, claim element 17[d] provides two key pieces of information about the "representation[s] of the at least one folder": (1) the contents of the files in the representations of the folder (i.e., "**at least one file**") are "**store[d]**" on the "**at least one server**"; and (2) those contents are not "**communicated**" to "**the second node**," nor are they "**stored at the second node**," till the user at the second node "**initiat[es] [] the communication**." (*Id.*) Accordingly, Patent Owner incorporates this information in its proposed construction of "representation of the at least one folder": i.e., "a folder that can be displayed at the user's node, but whose file contents are **stored remotely** and are **not communicated to the user till the user initiates the communication**." (*Id.*)

(POR at 31-32)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

[0013] Further, a system for sharing a data object in a data store via a communication is described. The system includes a descriptor handler component for receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent. The system further includes a descriptor generator component for creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user. The system still further includes a content manager component for including, by the second communications agent, the mount descriptor in a first message, addressed to the first user. The system additionally includes a com-out component for, sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store. The system also includes a processor, wherein at least one of the descriptor handler component, the descriptor generator component, the content manager component, and the com-out component includes an instruction that is executed by the processor during operation of the system.

(EX2004 at [0013]; POR at 36)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

[0109] Returning to FIG. **2A**, a block **208** illustrates that the method yet further includes creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. Accordingly, a system for sharing a data object in a data store via a communication includes means for creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. For example, the arrangement in FIG. **3A**, includes mount service component **308** that is operable for creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. FIG. **4** illustrates mount service component **408** as an adaptation and/or analog of mount service component **308** in FIG. **3A**. One or more mount service components **408** operate in an execution environment **401**. The system for sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. In FIG. **4**, a mount service component **408** is illustrated as a component of data store navigator component **406**.

(EX2004 at [0109]; POR at 36)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

[0116] In yet another aspect, a representation of a data object a may include and/or may otherwise be based on a replica of at least a portion of the data object. The replica may be synchronized with other replicas and/or an original data object. A mount service **408** may interoperate with an XtreeemFS client to create a representation of a data object identified in a mount descriptor at location in a data store. XtreemFS replicates file data across multiple storage servers, which can be distributed worldwide. The replication algorithm is designed to handle problems and failure scenarios that can occur in distributed systems including message loss, network partitions, server crashes, and the like. When a replicated file is opened, XtreemFS automatically elects a primary replica for that file which processes all file operations. If the primary fails, one of the backup replicas will automatically take over after a short failover period. To an application accessing a file via XtreemFS a regular file and/or folder and a replicated representation behave the same or similarly.

(EX2004 at [0116]; POR at 37)

The Intrinsic Evidence Supports PO's Construction

[0116] In yet another aspect, a representation of a data object a may include and/or may otherwise be based on a replica of at least a portion of the data object. The replica may be synchronized with other replicas and/or an original data object. A mount service **408** may interoperate with an XtreeemFS client to create a representation of a data object identified in a mount descriptor at location in a data store. XtreemFS replicates file data across multiple storage servers, which can be distributed worldwide. The replication algorithm is designed to handle problems and failure scenarios that can occur in distributed systems including message loss, network partitions, server crashes, and the like. When a replicated file is opened, XtreemFS automatically elects a primary replica for that file which processes all file operations. If the primary fails, one of the backup replicas will automatically take over after a short failover period. To an application accessing a file via XtreemFS a regular file and/or folder and a replicated representation behave the same or similarly.

[0117] For read-only access, replicas of an original folder and/or file may be created and distributed to various replication servers to enhance performance. Through replication a caching infrastructure may be built in order to reduce latency and bandwidth consumption. Partial replicas may be supported, in addition to or instead of full replicas. File, folder, and/or other data object metadata may also be replicated.

[0118] Thus, as describes above, a representation of a remote data object may be created based on a mount descriptor that includes information for accessing the data object based on at least one of Samba, content management interoperability services (CMIS), Web Distributed Authoring and Versioning (WedDAV), Atom Publishing Protocol (AtomPub), file transfer protocol (FTP), RSS, SSH FTP, and an identified network file system.

[0119] Other distributed file systems that may be accessed to mount a representation of a data object identified in a mount descriptor at a location in a data store. The Andrew File System (AFS), the Distributed File System (DFS), and the Network File System (NFS). Atom Publishing Protocol (AtomPub), file transfer protocol (FTP), RSS, SSH FTP, and content management interoperability services (CMIS) are other means for mounting a data object identified in a mount descriptor.

(Sur-Reply at 5; EX2004 at [0116]-[0119]; *see also* POR at 36-38)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Specification Confirms a "Representation" Must be Accessible to Retrieve a Data Object

[0103] In an aspect, determining a location in a data store for creating a representation of a data object may include presenting a navigation UI element for navigating the data store. The location may be determined in response to detecting a user input identifying the location. View window UI element **602a** in FIG. **6A** includes a mount UI element **618a** for receiving user input to display a navigation window UI element **620a** for identifying a location in a data store of execution environment **401** of first node **502** to mount and/or otherwise attach the remote "Music" folder identified in the mount descriptor. In response to an input directed to mount UI element **618a**, a corresponding UI element handler **415** may interoperate with data store navigator component **406** to present navigation window UI element **620a** that includes a presentation space **622a** for presenting a user detectable representation of one or more locations in a data store of execution environment **401** of first node **502**.

(EX2004 at [0103]; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

[0110] FIG. **7** includes data flow message that illustrates an attach data flow **708** that may be internal to execution environment **401** of first node **502** and/or that may include a message received via network **506** by execution environment **401** of first node **502**. Attach data flow **708** illustrates a data flow to mount service component **408** operating in execution environment **401** of first node **502**. Mount service component **408** may be invoked automatically in response to detecting a mount descriptor in an aspect. In another aspect, mount service component **408** may be invoked in response to a user input. A representation of a data object identified in a mount descriptor may be created automatically in response to detecting the mount descriptor and/or detecting a user input associated with the message in which the mount descriptor was received.

(EX2004 at [0110]; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

[0115] As an alternative and/or in case execution environment **401** of first node **502** does not include a Samba client and/or a compatible analog, a mount descriptor may specify one or more alternatives in addition to or instead Samba. For example execution environments may be separated and/or otherwise blocked by one or more firewalls, Mount Service **408** may operate to mount the "Music" found via another identifier in mount descriptor **804a**. Web Distributed Authoring and Versioning (WebDAV) is an extension of the Hypertext Transfer Protocol (HTTP) that facilitates collaboration between users in editing and managing documents and files stored on World Wide Web servers. A WebDAV URL **810a** identifies HTTP as an access protocol and identifies a location in execution environment **401** of second node **504** for accessing the "Music" folder. Mount service component **408** in execution environment **401** of first node **502** may interoperate with a WebDAV client (not shown) included in and/or associated with file system **427**, which in an aspect may be and/or may be included in the data store of execution environment **401** of first node **502**. WebDAV URL **810a** may include user identification and authentication information to the WebDAV client if needed. Alternatively, the WebDAV client and/or mount service **408** may receive such information from the user and/or from a pre-stored profile. In an aspect, the "Music" folder may be mounted as a folder in, for example, a pre-existing Music folder in execution environment **401** of first node **502**. The two music folders may be presented as a merged entity. In another aspect, the "Music" folder identified in the mount descriptor may be mounted at a location in the pre-existing Music folder. The mounted folder may be renamed to distinguish it from a folder with the same name and/or to identify its source by user and/or by node, for example. Alternatives to WebDAV include the file transfer protocol FTP.

(EX2004 at [0115]; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

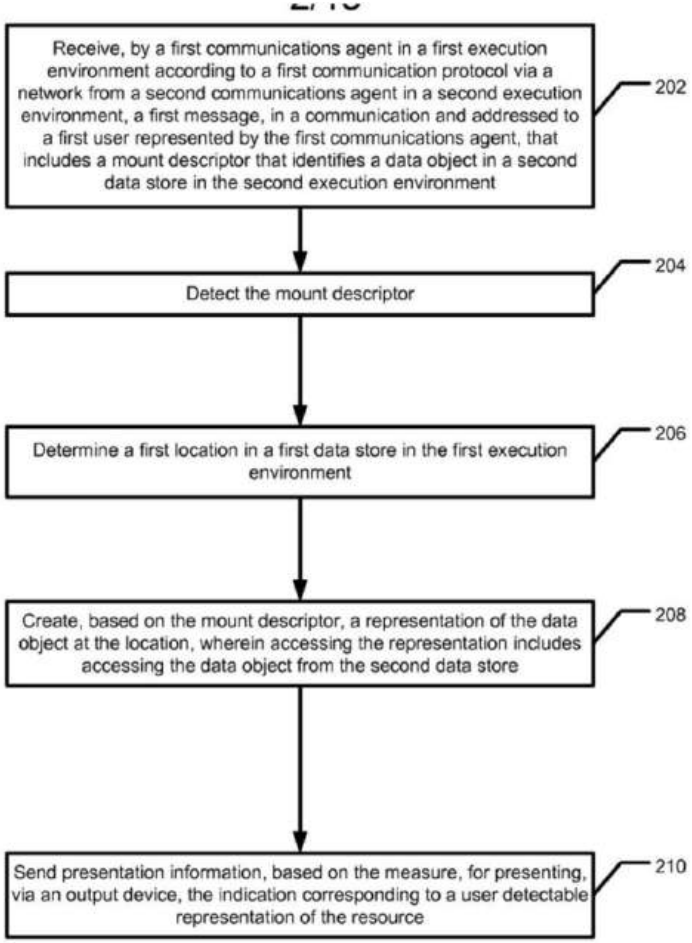
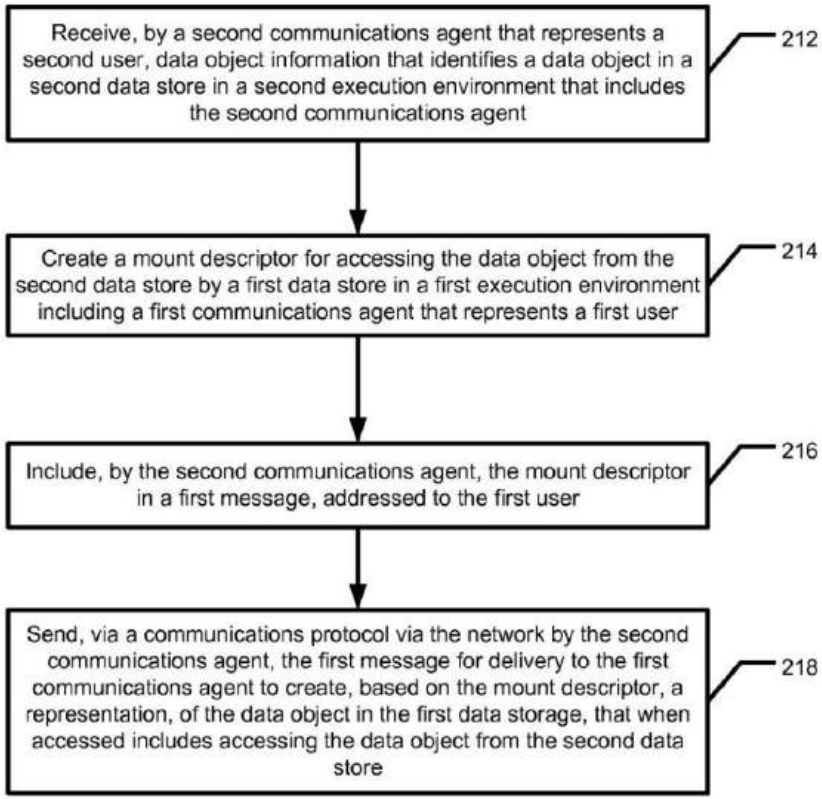


FIG. 2A



(EX2004, Figures 2A and 2B; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

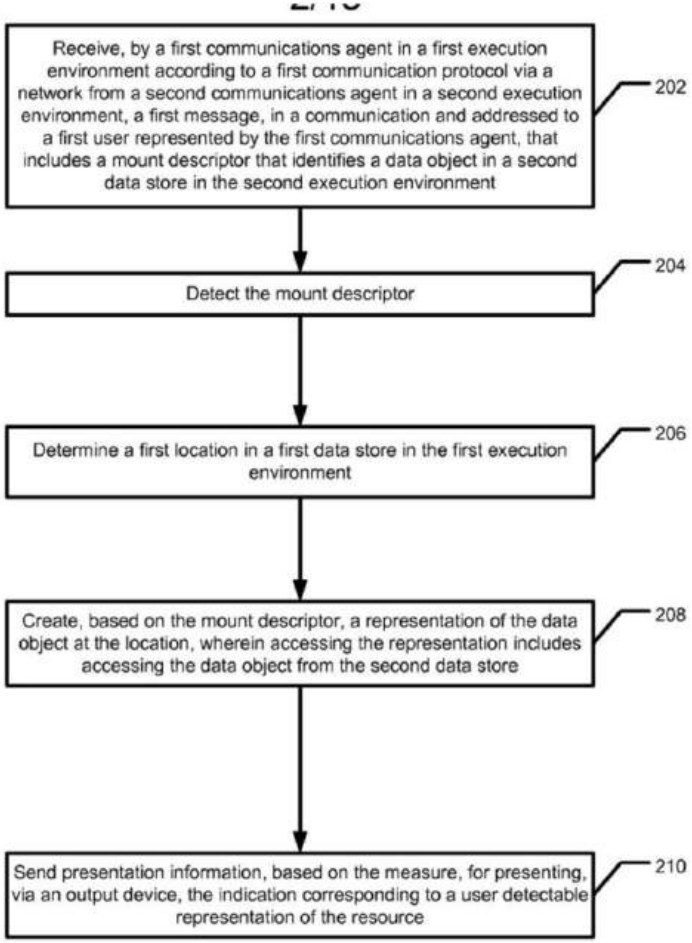
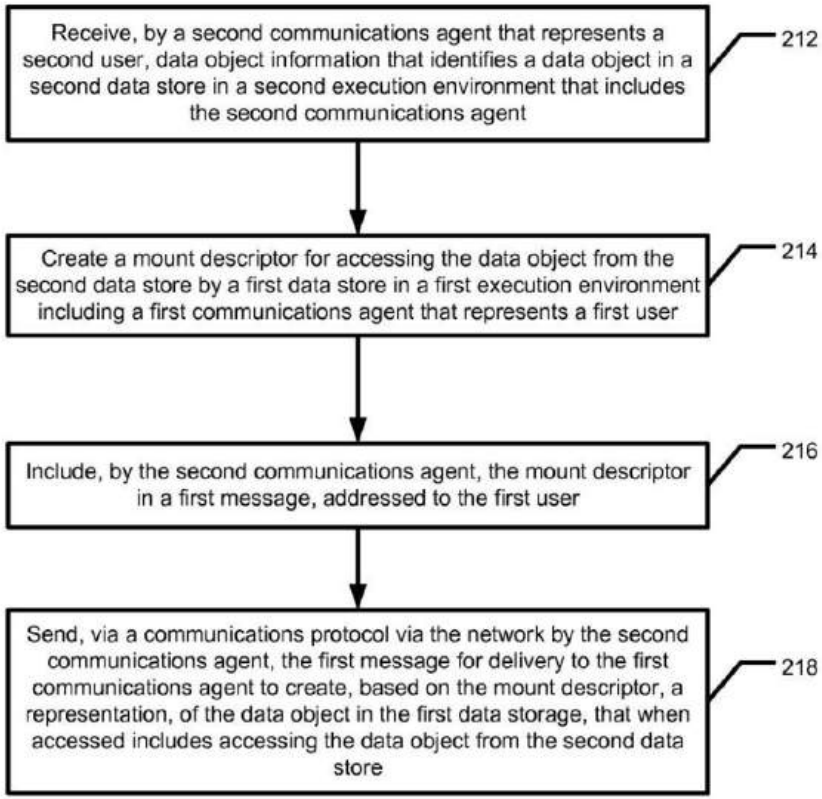


FIG. 2A



(EX2004, Figures 2A and 2B; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Confirms a "Representation" Must be Accessible to Retrieve a Data Object

1. A method for sharing a data object in a data store via a communication, the method comprising:

receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment;

detecting the mount descriptor;

determining a first location in a first data store in the first execution environment; and

creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store,

wherein performing at least one of the preceding actions comprising the method includes execution of an instruction by a processor.

11. A method for sharing a data object in a data store via a communication, the method comprising:

receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent;;

creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user;

including, by the second communications agent, the mount descriptor in a first message, addressed to the first user; and

sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store,

wherein performing at least one of the preceding actions comprising the method includes execution of an instruction by a processor.

(EX2004, claims 1 and 11; Sur-Reply at 5)

The Intrinsic Evidence Cited by the Board Does Not Contradict PO's Construction

[0049] The user-detectable outputs of a user interface are generically referred to herein as "user interface elements" or abbreviated as "UI elements". More specifically, visual outputs of a user interface are referred to herein as "visual interface elements". A visual interface element may be a visual output of a graphical user interface (GUI). Exemplary visual interface elements include windows, textboxes, sliders, list boxes, drop-down lists, spinners, various types of menus, toolbars, ribbons, combo boxes, tree views, grid views, navigation tabs, scrollbars, labels, tooltips, text in various fonts, balloons, dialog boxes, and various types of button controls including check boxes and radio buttons. An application interface may include one or more of the elements listed. Those skilled in the art will understand that this list is not exhaustive. The terms "visual representation", "visual output", and "visual interface element" are used interchangeably in this document. Other types of UI elements include audio outputs referred to as "audio interface elements", tactile outputs referred to as "tactile interface elements", and the like.

(Paper 12 at 6-7; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Cited by the Board Does Not Contradict PO's Construction

[0050] A visual output may be presented in a two-dimensional presentation where a location may be defined in a two-dimensional space having a vertical dimension and a horizontal dimension. A location in a horizontal dimension may be referenced according to an X-axis and a location in a vertical dimension may be referenced according to a Y-axis. In another aspect, a visual output may be presented in a three-dimensional presentation where a location may be defined in a three-dimensional space having a depth dimension in addition to a vertical dimension and a horizontal dimension. A location in a depth dimension may be identified according to a Z-axis. A visual output in a two-dimensional presentation may be presented as if a depth dimension existed allowing the visual output to overlie and/or underlie some or all of another visual output.

(Paper 12 at 6-7; Sur-Reply at 5)

The Intrinsic Evidence Cited by the Board Does Not Contradict PO's Construction

[0051] A "user interface (UI) element handler" component, as the term is used herein, includes a component of configured to send information representing a program entity for presenting a user-detectable representation of the program entity by an output device, such as a display. A "program entity" is an object included in and/or otherwise processed by an application or executable. The user-detectable representation is presented based on the sent information. Information that represents a program entity for presenting a user detectable representation of the program entity by an output device is referred to herein as "presentation information". Presentation information may include and/or may otherwise identify data in one or more formats. Exemplary formats include image formats such as raw pixel data, JPEG, video formats such as MP4, markup language data such as hypertext markup language (HTML) and other XML-based markup, a bit map, and/or instructions such as those defined by various script languages, byte code, and/or machine code. For example, a web page received by a user agent from a remote application provider may include HTML, ECMAScript, and/or byte code for presenting one or more UI elements included in a user interface of the remote application. Components configured to send information representing one or more program entities for presenting particular types of output by particular types of output devices include visual interface element handler components, audio interface element handler components, tactile interface element handler components, and the like.

(Paper 12 at 6-7; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Cited by the Board Does Not Contradict PO's Construction

[0052] A representation of a program entity may be stored and/or otherwise maintained in a presentation space. As used in this document, the term "presentation space" refers to a storage region allocated and/or otherwise provided for storing presentation information, which may include audio, visual, tactile, and/or other sensory data for presentation by and/or on an output device. For example, a buffer for storing an image and/or text string may be a presentation space as sensory information for a user. A presentation space may be physically and/or logically contiguous or non-contiguous. A presentation space may have a virtual as well as a physical representation. A presentation space may include a storage location in a processor memory, secondary storage, a memory of an output adapter device, and/or a storage medium of an output device. A screen of a display, for example, is a presentation space.

(Paper 12 at 6-7; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Intrinsic Evidence Cited by the Board Does not Contradict PO's Construction

[0069] The term "data object" as used herein refers to an entity, identifiable within a data store, for storing and/or accessing data. Exemplary data stores include file systems, directory service including DNS and LDAP, and data bases including relational databases and hierarchical databases, to name a few examples. Exemplary data objects include files and folders in a file system. Accessing a data object includes one or more of accessing data in the data object and accessing metadata, maintained by the data store, about the data object.

(Paper 12 at 6-7; Sur-Reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Only PO's Construction Is Consistent with Other Claims of the '520 Patent

1[h]: . . . based on the receipt of the indication of the at least one folder, the indicia associated with the at least one email address, and the indication to share the at least one folder via the at least one network;

cause, utilizing code configured to be stored on a storage at the second node and further configured to cooperate with a file explorer interface, creation of a representation of the at least one folder in a location among one or more folders on the file explorer interface,

where **the storage at the second node does not store the at least one file when the creation of the representation of the at least one folder is caused**;

1[i]: detect, at the second node, **an indication to open the at least one file in the at least one folder**; and

1[j]: **in response to detection of the indication to open the at least one file in the at least one folder, cause retrieval of the at least one file via the at least one network for permitting display of the at least one file at the second node.**

(POR at 33)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Only PO's Construction Is Consistent with Other Claims of the '520 Patent

Because claim terms are normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims.

Phillips v. AWH Corp., 415 F.3d 1303, 1314 (Fed. Cir. 2005)

(POR at 33)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Mischaracterizes PO's Construction

Second, PO's proposed construction is facially illogical. PO equates a "representation" of a folder with the "folder" itself. *See* POR, 30. This is inconsistent with the claim language. Elements 17[e]-[g] discuss "display[ing]" the "representation of the at least one folder" on a webpage or file explorer application, suggesting that, consistent with the claim term's plain meaning, the "representation" of the folder is a stand-in image or symbol that depicts, but that is not itself, a folder.

(Rep. at 2)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Remaining Arguments Lack Merit

Third, PO's construction improperly inserts language from other portions of the claim into this simple term. POR, 29-32. PO's construction appears to be based on language from 17[d]—"permitting avoidance of at least one file from being communicated to and stored at the second node **until an initiation of the communication** of the at least one file **by a user** of the second node is detected."[1] *Id.*, 31-32. But PO provides no justification for why the phrase "representation of the at least one folder" that appears in elements 17[e]-[g] needs to incorporate concepts from 17[d] and cannot speak for itself. *Id.* Indeed, 17[d] relates to a

(Rep. at 3)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Remaining Arguments Are Unavailing

Fourth. the error of PO's construction is further exemplified by the other claims whose language PO's construction would render language superfluous.

(Rep. at 4)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Remaining Arguments Are Without Merit

Finally, PO's reliance on the '635 application, which is an abandoned related application incorporated by reference, fails to support its incorrect construction.

. . .

36. No claim construction principle supports importing limitations from abandoned related applications into the claims of a granted patent. Moreover, the cited portions of the application (POR, 36) do not support PO's construction: they do not indicate what initiates the "accessing" of the data object, and even if the accessing were described as being user-initiated, nothing in the application precludes communication of the data object separate from that accessing.[3] EX1034, ¶¶14-16.

(Rep. at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Disclose 17[d]-17[g] and Teaches Away from the Claimed Inventions (Grounds 1-4)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

to retrieve it, or perhaps give up until the morning. Some methods exist to allow remote access to a computer, for example using a virtual private network (VPN) to access a corporate network from a remote location. However, if the user is accessing the document remotely and loses his connection, he may lose his changes, be unable to continue, and may end up with a corrupted document.

(EX1003 1:29-35)

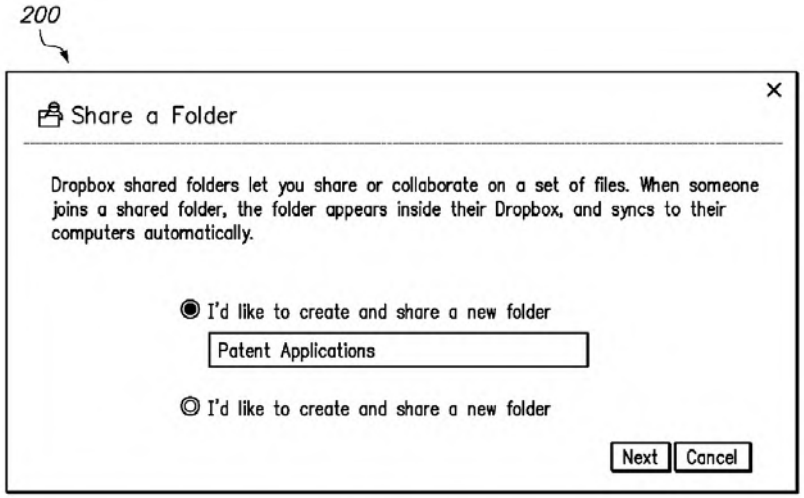


FIG. 2

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Metadata server **102** receives requests from clients to update the server's copy of synchronized folders and provides clients with a list of metadata for files being synchronized. Block server **104** receives, stores, and serves blocks of data constituting synchronized files. Notification server **106** provides updates to clients when a synchronized folder has been updated on the server, and provides those notifications to the clients. The operation of each of these components is described further below.

Note that in various embodiments, sharing occurs at the folder level—that is, a folder and any files in that folder are shared among clients, and kept synchronized by the clients and host system **110**. Throughout this description therefore, we refer to both folders and files as being synchronized and shared.

(EX1003 2:63-3:10)

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Metadata server **102** receives requests from clients to update the server's copy of synchronized folders and provides clients with a list of metadata for files being synchronized. Block server **104** receives, stores, and serves blocks of data constituting synchronized files. Notification server **106** provides updates to clients when a synchronized folder has been updated on the server, and provides those notifications to the clients. The operation of each of these components is described further below.

Note that in various embodiments, sharing occurs at the folder level—that is, a folder and any files in that folder are shared among clients, and kept synchronized by the clients and host system **110**. Throughout this description therefore, we refer to both folders and files as being synchronized and shared.

(EX1003 2:63-3:10)

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

As described above, a client **108** with a synchronized folder informs host system **110** when a file in a synchronized folder has been added, modified or deleted. Other clients may also have versions of the same synchronized folder, which are updated via host system **110** as follows.

. . .

Upon receiving the notification, each client **108** sends a list request to metadata server **102**, and in response receives file metadata for all files in subscribed folders. The client then examines the block list for each file and identifies any listed blocks that the client does not already have in its database **502**. File transfer module **510** then asks block server **104** for a patch from the parent block the client is in possession of to the new block the client needs. Block server **104** creates the patch and provides it to client **108** in response to the request. Client **108** then applies the patch to the parent block to obtain the updated block, which it then stores. Client **108** repeats the process for each block that needs updating. At the conclusion of the process, the client's version of the file is synchronized with the updated version on host system **110**.

(EX1003 5:37-6:5)

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

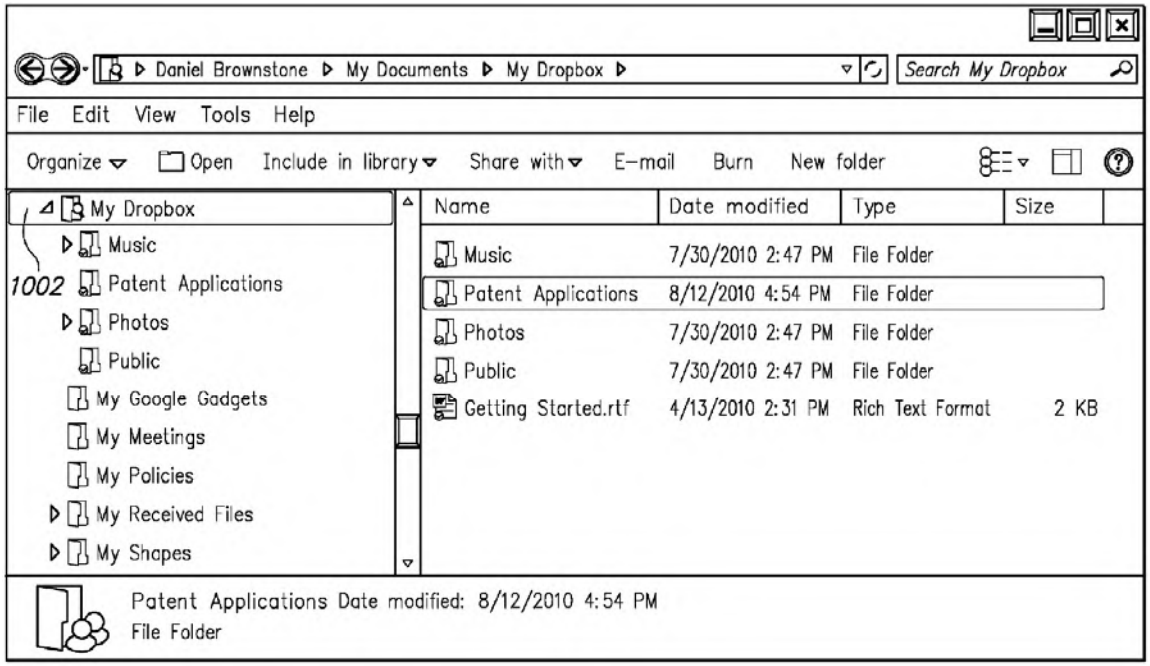
Houston Does Not Support Petitioner

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.



(EX1003)

FIG. 10

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

In one embodiment, host system **110** performs namespace mapping functions, allowing users of system **100** to interact seamlessly with shared folders through their operating system's standard user interface. Assume a first user has a folder that is synchronized with host system **110**. Referring to FIG. **10**, the "My Dropbox" folder **1002** is such a folder. In one embodiment, the synchronized folder **1002** exists in a first name space, which for purposes of example we will refer to as "1:". In the example of FIG. **10**, the folders Music, Patent Applications, Photos, and Public, and the document Getting Started.rtf are each stored also in namespace 1:. Note also that the "My Dropbox" folder **1002** in the illustrated embodiment is displayed next to other conventional folders such as "My Meetings", "My Received Files," and others.

. . .

Assume now that the invited user has an existing namespace, "3:". Assuming the user accepts the invitation to share the folder, metadata server **102** creates a link in the 3: namespace, such that "3:/Patent Applications" points to namespace 2:. Metadata server **102** also adds the invited user's identifier to the list of users sharing the folder, and notification server **106** begins providing change notifications to the invited user's client. The invited user's client then obtains the latest version of the synchronized file according to the methods described above.

(EX1003 7:18-58)

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42, 17-18)

The Incorporated *Houston* '787 Provisional Confirms PO's Position

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

55. Storing copies of files locally at each user is critical to *Houston*'s system. *Houston* repeatedly criticizes systems that store documents remotely—for example, *Houston* notes that "if the user is accessing the document remotely and loses his connection, he may lose his changes, be unable to continue, and may end up with a corrupted document." (EX1003, 1:29–35; EX2026 ¶¶ 8–10, 12–14.) And *Houston*'s parent provisional application, Provisional Patent App. No. 61/233,787 (*Houston II*), which is incorporated by reference, explains that the use of local copies of files at each client is a core feature of the system. (EX2026 ¶ 11.)

(EX2051, paragraph 55)

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42, 17-18)

The Incorporated *Houston* '787 Provisional Confirms PO's Position

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

¶9 ("the innovative technology presented herein eliminates the need to have a person maintaining a central server for a business. The technology makes local copies of data . . . . The technology allows for a superior user experience, since multiple users may be offline or online and still use this technology."); ¶10 ("The technology provides local input/output speed (which equates to less speed time in comparison to traditional systems) because the files are local, and thus listing and accessing of the files appear much more quickly to users than if they had used traditional systems."); ¶12 ("Also, most applications of a traditional file server architecture are

(PO Sur-reply at 10)

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42, 17-18)

The Incorporated *Houston* '787 Provisional Confirms PO's Position

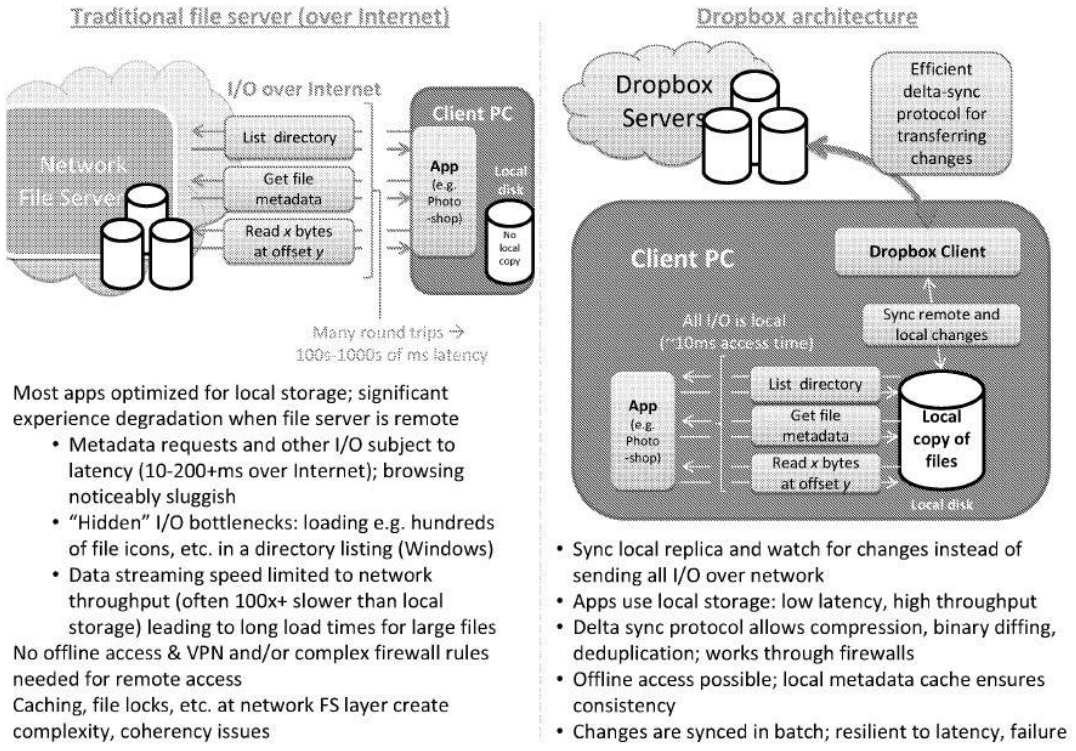
17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Traditional File Server Architecture vs. Dropbox FIG. 1



(PO Sur-reply at 11)

Houston's system "automatically" and "immediately" synchronizes folders and their contents among multiple clients by downloading local copies of shared documents to each user's client.

(POR at 42, 17-18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

1001, 55:52–56. With respect to Figure 10 in Houston, the I explained that the "brief absence of checkmark overlays indicates that the folders and file had not yet been synchronized with the host system," and "[t]hus, at the very moment these icons were created . . . the client would not have stored any file within the folders,

(EX1034 ¶ 30)

31. Dr. Smith is wrong because my explanation is supported by evidence in the form of (1) Houston's disclosure and (2) a supplemental document called SugarSync. Exhibit 1011 ("SugarSync[7]"). ***First***, I explained that Houston's Figure 10 explicitly discloses the use of checkmarks for some folders but not others, and therefore teaches that the items with checkmarks have been synchronized with the host system and are stored locally. Exhibit 1002, ¶99[8] ("Houston's Figure 10 reveals check mark overlays on the icons for the folders and one file (i.e., the check marks enclosed in circles on the lower left corner of the icons and annotated in green) in the 'My Dropbox' folder 1002, but not on the conventional, local folders (e.g., 'My Meetings')"). ***Second***, SugarSync explicitly explains that "Files or folders that in sync with the cloud are represented by with green check mark . . .". SugarSync, 1. Thus, my explanation is supported by evidence in the form of Houston's disclosure and SugarSync.

(EX1034 ¶ 31)

Petitioner's "checkmark overlay" argument is misleading, at best

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

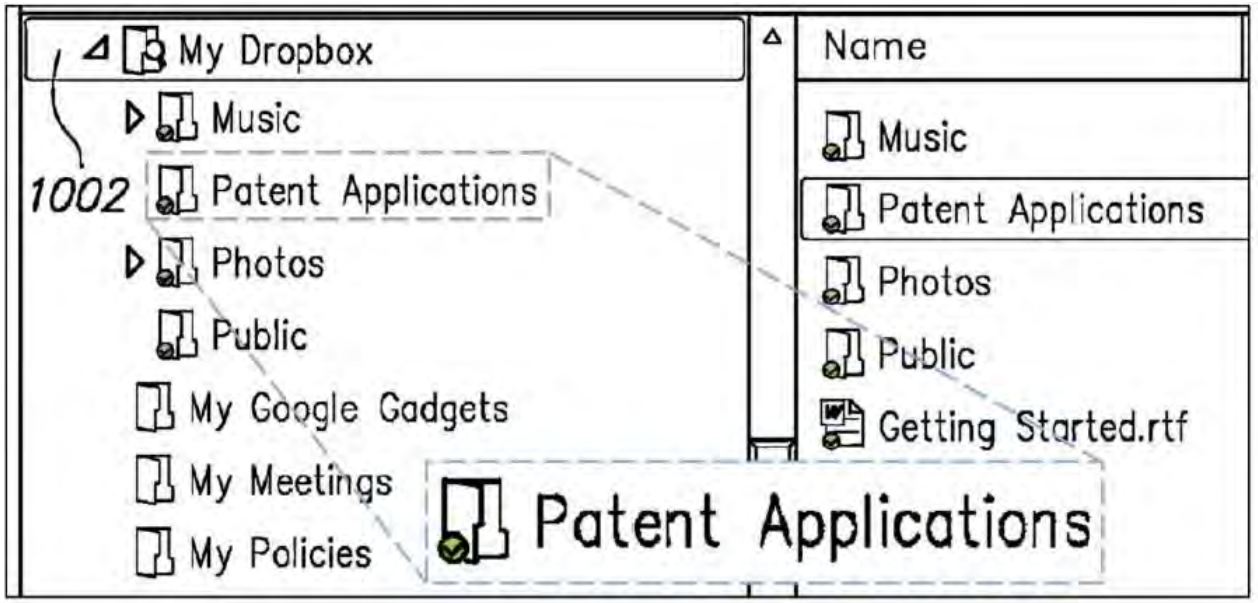


Exhibit 1003, Figure 10 (excerpted and annotated)

(EX1002 ¶ 99)

Petitioner's "checkmark overlay" argument is misleading, at best

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Does Not Support Petitioner

including the "Patent Applications" folder[.]" (Pet. at 36.) But *Houston* does not teach that any "overlays" are briefly absent, or that "at the very moment [they] were created . . . the client would not have stored any file within the folders."

(POR at 43)

least one server that stores the at least one file." (EX2051 ¶102.) Regardless of any alleged "brief absence," *Houston* does not wait for "an initiation of the communication of the at least one file" by a user before it stores a file at the user's node; instead, it downloads local copies to the user's node automatically. (*Id.*)

(POR at 45)

Petitioner's "checkmark overlay" argument is misleading, at best

(POR at 43)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's "Checkmark Overlay" Argument Fails

Petitioner also relies on EX1011—a document describing a different system, "SugarSync"—to assert that "Files or folders that in sync with the cloud are represented by with green check mark[.]" (Pet. at 35–36.) But EX1011 is not related to *Houston* and does not otherwise describe it. (EX2051 ¶101.) Petitioner

(POR at 44)

anyway. And EX1011 does not even show the "brief absence" of checkmark overlays; instead, EX1011 shows that a folder that automatically downloads local copies would have displayed a gold "syncing" overlay instead:

(*See* EX1011 ("objects that are being uploaded or downloaded are represented with a gold syncing icon"); EX2051 ¶101.)

(POR at 44)

Petitioner's "checkmark overlay" argument is misleading, at best

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's "Checkmark Overlay" Argument Fails

fn. 7.) That should speak for itself. Dropbox has access to all relevant information concerning the icon overlays in *Houston*, yet ***it did not provide any Dropbox materials for support***. Instead, it appears to have ***withheld*** such information because it is inconsistent. A Dropbox guide from 2009 states that the checkmark overlay refers to syncing files to Dropbox but makes no reference to downloads. It indicates that Dropbox and SugarSync had different icon overlay functionality. (*Cf.* EX1002,

(PO Sur-reply at 12)

Petitioner's "checkmark overlay" argument is misleading, at best

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Teaches Away from the Challenged Claims

For these reasons, *Houston* teaches away from Petitioner's proposed combinations. Modifying *Houston* from a system intended to download the file contents of shared folders automatically to one that functions as a remote "virtual drive" would have frustrated the heart of the teachings of *Houston*. (EX1003, 2:17–18; Fig. 2; 5:37–6:3; 7:49-58; EX2051 ¶98.) *Houston* specifically criticizes such systems and describes numerous benefits using synchronized, locally stored copies. EX1003, 1:6-9 (*Houston II*, ¶¶1, 12-13-14).) The proposed combination of *Houston* with *Garcia* and/or *Manzano* would result in the loss of the fundamental purpose of *Houston*, namely, making shared folders and their contents locally available to all of a user's devices. (EX2051 ¶105.) This demonstrates that there would not have been

(PO Sur-reply at 13)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Does Not Seriously Dispute the Operation of *Houston*

interpret the Houston reference—or any reference in this IPR for that matter. But there is also nothing inconsistent with having locally available folders and modifying Houston so that the recipient first needs to click on a newly shared folder before its content is downloaded. *See, e.g.*, Pet. Reply at 7, 9-11.

(Pet. Mtn. to Exclude, Paper 40, at 3)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Properly Construed, the Board Agrees with PO's Characterization of *Houston*

Thus, Patent Owner's argument, that Houston teaches away from combining with Garcia to delay storing a file when a folder is shared, is not applicable to Petitioner's challenges of claim 17–21. The fact that the Houston system ensures the availability of synchronized local copies of files at each user and criticizes systems that store documents remotely, such as in Garcia, is irrelevant to those claims. On this record, nothing about that particular difference in the approaches of Houston and Garcia would have discouraged one of ordinary skill in the art from adopting Houston to take advantage of the teachings in Garcia regarding use of email and folder interfaces.

(Paper 12 at 36)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Properly Construed, the Board Agrees with PO's Characterization of *Houston*

In arguing that Petitioner has not identified, in the Houston/Garcia combination, any creation of a "representation" of a shared folder at the receiving second node as required by the claims, Patent Owner argues that the claimed creation of a "representation" means that:

> [W]hen a folder is shared with a second node, a ***special version*** of the shared folder is created at the second node that appears to the user to exist locally at the second node, but whose contents (and the contents of files within the folder) exist on the cloud and are not downloaded till the user wishes to interact with them.

Prelim. Resp. 33–34 (emphasis added). However, there is no such requirement of a "special version of the shared folder" in claim 17. Patent Owner's assertions otherwise are based on unsupported attorney argument that has no basis in the record.

(Paper 12 at 42)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Properly Construed, the Board Agrees with PO's Characterization of *Houston*

There is no explicit definition of "representation" in the '520 patent. One example of a representation of a folder is described in the incorporated by reference '635 application as a window or other visual interface element displayed on a screen of a display presenting information representing a program entity such as a folder. Ex. 2004 ¶¶ 49–52, 69. Another example, also in the '635 application, states that "a representation of a data object may include and/or may otherwise be based on a replica of at least a portion of the data object" — the opposite of Patent Owner's asserted "special version" definition. It is also true that disclaimed claim 1 includes the requirement that, when a representation of a folder is created, such creation "does not store the at least one file when the creation of the representation of the at least one folder is caused." Ex. 1001, 49:44–49. But this specific requirement for claim 1 does not apply to claim 17, or support the sweeping interpretation of "representation" that Patent Owner asserts.

(Paper 12 at 42-43)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Does Not Cure Houston's Deficiencies (Grounds 1-4)

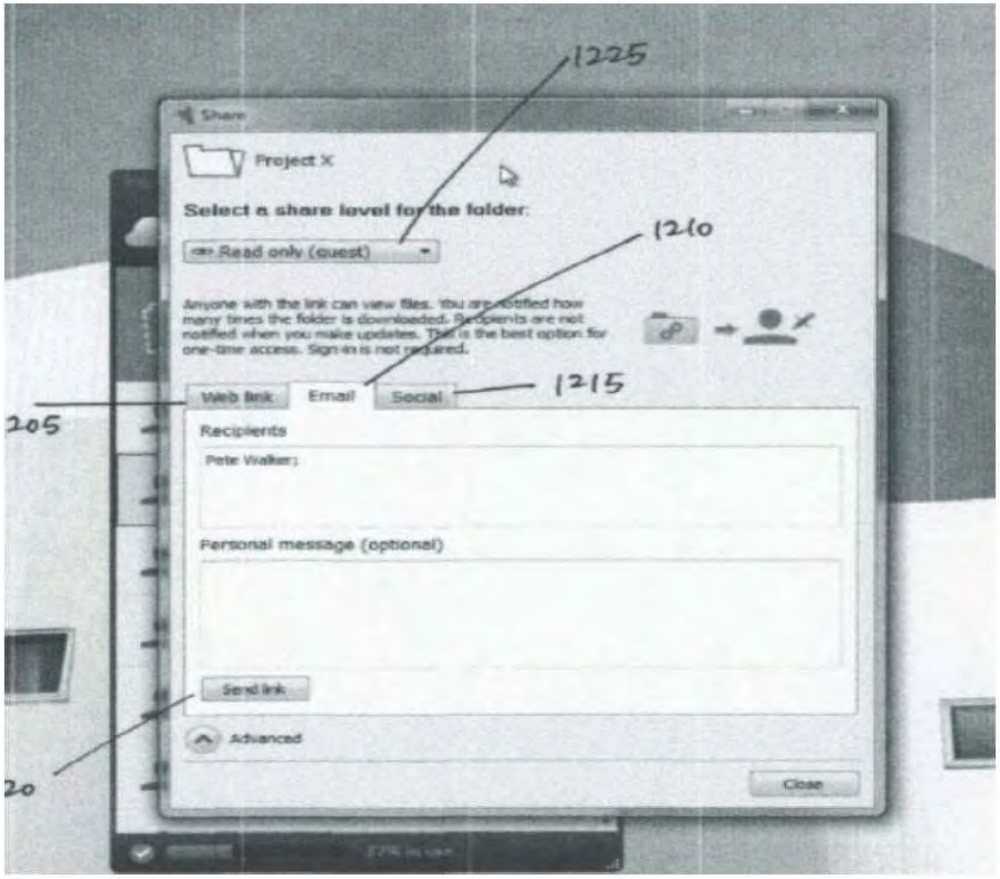
Garcia Does Not Teach Creating a Representation at the Second Node

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.



(EX2052; see also EX1004 Fig.12)

Garcia downloads the contents of the shared folder at the receiving node

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

(POR at 47-48)

Garcia Does Not Teach Creating a Representation at the Second Node

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Referring to FIG. **20***a*, a user is enabled to synchronize one or more folders on any synchronization client to the synchronization server and other synchronization clients (step **2005**). All folders, across a plurality of synchronization clients that a user has backed up or synchronized to the synchronization server, are identified, as well as any folders that the user shares with other users (step **2010**). Each of the synchronization clients stores metadata with information on all folders shared with or by the user and all folders that the user has backed up or synced to the synchronization server (including the contents of the folder). The synchronization client on the client device uses this metadata to generate the virtual drive and the user interface.

(EX1004 at 5:14-26)

In response to detecting a change in a folder in the virtual drive, the folder is updated on the synchronization server (step **2025**). All remote clients of the user in the synchronization system are identified, as well as all remote synchronization clients, if any, of other users whom have access to the changed folder (step **2030**). In one embodiment, all synchronization clients that have an interest in updates to the folder "register" for notices of changes to the folder. The identified clients are notified of the change to the folder (step **2035**). In response to the change being a change to content within the folder, the content of the folder is automatically updated on any identified remote clients that have a corresponding folder. In one embodiment, the synchronization server first sends metadata to all the identified clients, notifying them of the change. The synchronization clients that have their own copy of the changed folder will then request the applicable file data from the server and the server will download the requested file data to the requesting synchronization clients.

(EX1004 at 5:40-62)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

(POR at 45-46)

Petitioner Mischaracterizes *Garcia*

Moreover, nothing in Garcia suggests that if the second user clicks on the email link, the second user receives an immediate download of files without using the virtual interfaces and synchronization features in Garcia. Indeed, a POSITA would have known the second user still needs to access the folder by going through a virtual interface. And a POSITA would have understood that clicking on a link, such as the link in the email of Garcia, will typically take a user to a webpage. Indeed, Garcia expressly envisions accessing the folder through ***either*** a file explorer interface (as shown in Garcia's Figure 10) or a web interface (as shown in Figure 13). *See, e.g.*, EX1004, 10:52-53 ("FIG. 10 illustrates one way in which a user may access items in objects displayed in the client sync application."), 11:44-45 ("FIGS. 13-14 illustrate the corresponding web interface 1305 for the synchronization system."). EX1034, ¶38.

(Rep. at 17)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

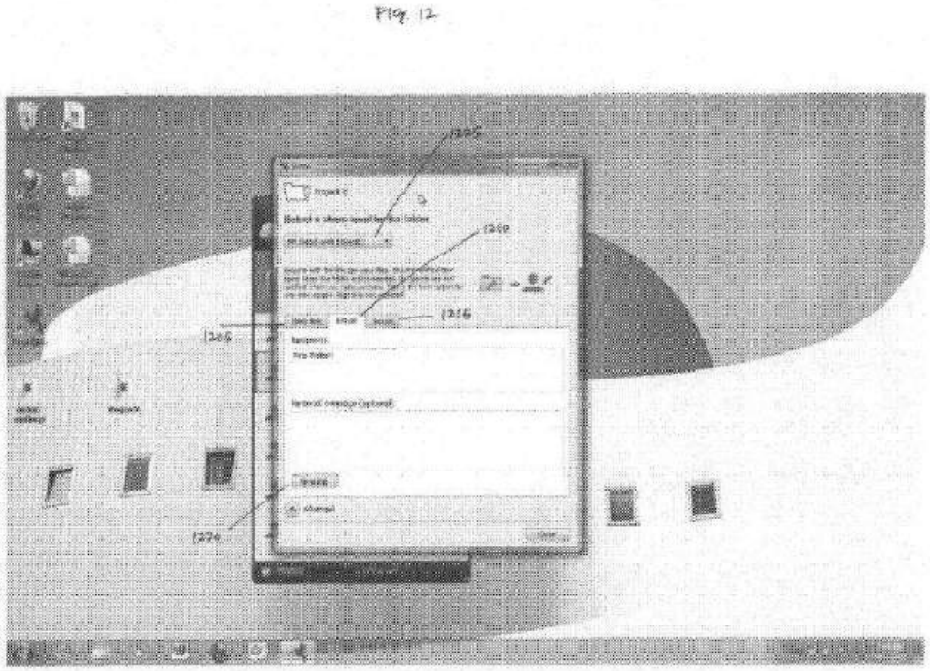
Petitioner Mischaracterizes *Garcia*

Second, PO's suggestion that only ***other devices*** (i.e., devices other than the device that receives the email) would show representations is an unsupported overly narrow interpretation of Garcia. POR, 47. Garcia explicitly discloses that "a user" double clicks on a file in a folder of the virtual drive to download a copy and does not limit this to the "user's other devices (e.g., different nodes)." *See* Pet., 9, 40, 41, 42 (citing EX1004, 5:35-39 ("when a user double clicks on a file or item in a remote folder in the virtual drive, the server downloads its copy of the file/item to the local synchronization software client (i.e., the client currently being used by the user)."). Thus, regardless of user device, Garcia teaches that a folder or file is downloaded by clicking on a representation of the file or folder in a virtual drive. EX1034, ¶39.

(Rep. at 17-18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Mischaracterizes *Garcia*

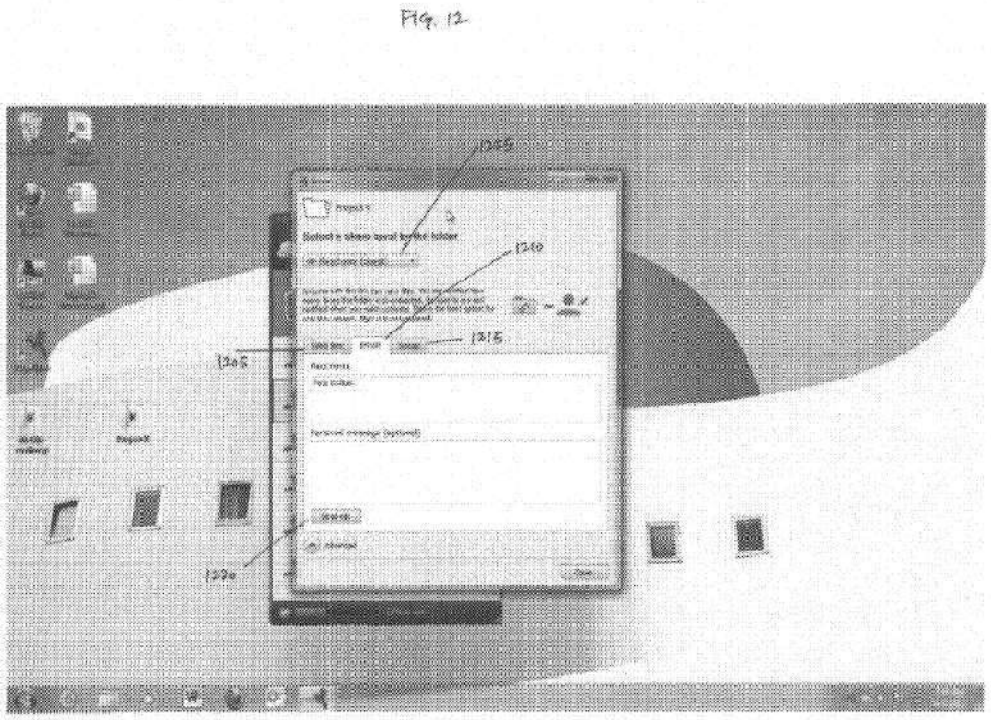


(EX1004 Fig. 12)

EX1002, ¶92.) But without an account, they would not have a virtual drive. Therefore, this supports PO's argument that the links provide access to the shared folder and its content to any recipient. That is consistent with *Garcia's* other disclosures that "[w]hen the user clicks on the send link button 1220, a link to the folder 1115 is sent to the contact. When the contact clicks on the link to folder 1115, the contents of the folder are shared with the contact." (EX1004, 11:26-31; *see also* 11:16-19; Fig. 12.) It is also consistent with contemporaneous evidence that PO provided regarding SugarSync. (EX2051, ¶61; EX2011 at 75-76; EX2010 at 78; POR, 22-24.)[12]

(PO Sur-reply at 15)

Petitioner Mischaracterizes *Garcia*



(EX1004 Fig. 12)

Moreover, a link to a shared folder is not a "representation" of the folder created at the recipient's node. Dr. Mowry's conclusory testimony seeks to gap-fill by claiming a POSITA would have understood that a second user still needs to access the folder through the virtual interface. (EX1034, ¶38.) But *Garcia* does ***not*** state that; indeed, PO's position is that *Garcia* discloses that when the link is accessed the folder contents are downloaded locally at the device being used and at most the folder and contents would show up as remotely stored objects if the user accessed its virtual drive from a ***different device.*** Petitioner relies heavily on *Garcia*'s virtual drive but it does not connect any disclosure in *Garcia* involving the receipt of the link sharing the folder and the creation of a "***representation***" of the shared folder at the device being used when the link is accessed.

(PO Sur-reply at 15-16)

Petitioner Mischaracterizes *Garcia*

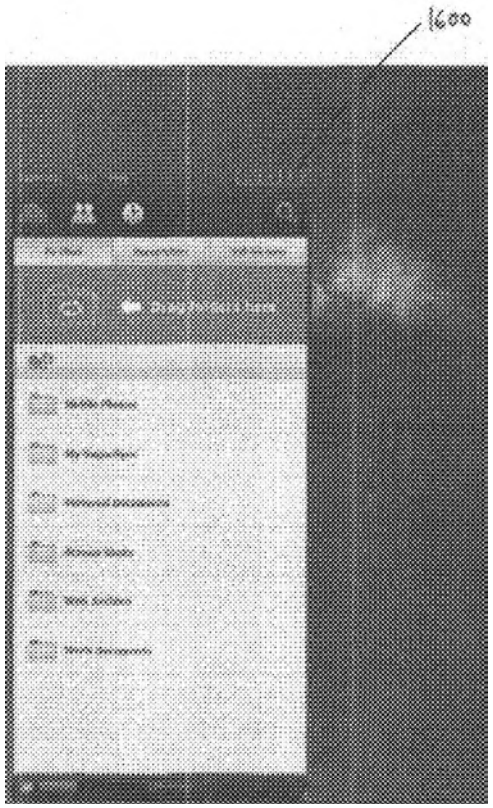
FIG. **9** illustrates the Web link tab **335**. The Web link tab **335** includes a list of folders or files that have been shared as a link with other users. In this example, a list of files is shown, which is sorted by the date the link was enabled 905. The list may be sorted using other criteria as well. Each file has an icon **910**, which indicates the number of times the file has been downloaded, and an icon **915**, which places the link onto the clipboard to be copied into an email, text, or other transmittal. One benefit of sending links to folders or files to other users is it allows other users to access the folders or files without requiring the other user to create an account in the synchronization system.

(EX1004 10:32-44)

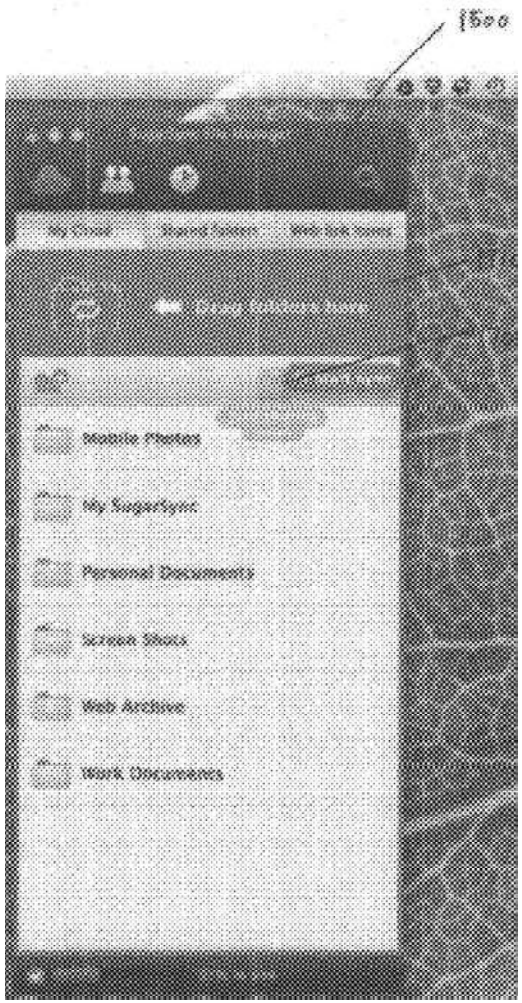
Moreover, a link to a shared folder is not a "representation" of the folder created at the recipient's node. Dr. Mowry's conclusory testimony seeks to gap-fill by claiming a POSITA would have understood that a second user still needs to access the folder through the virtual interface. (EX1034, ¶38.) But *Garcia* does ***not*** state that; indeed, PO's position is that *Garcia* discloses that when the link is accessed the folder contents are downloaded locally at the device being used and at most the folder and contents would show up as remotely stored objects if the user accessed its virtual drive from a ***different device.*** Petitioner relies heavily on *Garcia*'s virtual drive but it does not connect any disclosure in *Garcia* involving the receipt of the link sharing the folder and the creation of a "***representation***" of the shared folder at the device being used when the link is accessed.

(PO Sur-reply at 15-16)

Petitioner Mischaracterizes *Garcia*



(EX1004 Fig. 16 (excerpt))



(EX1004 Fig. 17 (excerpt))

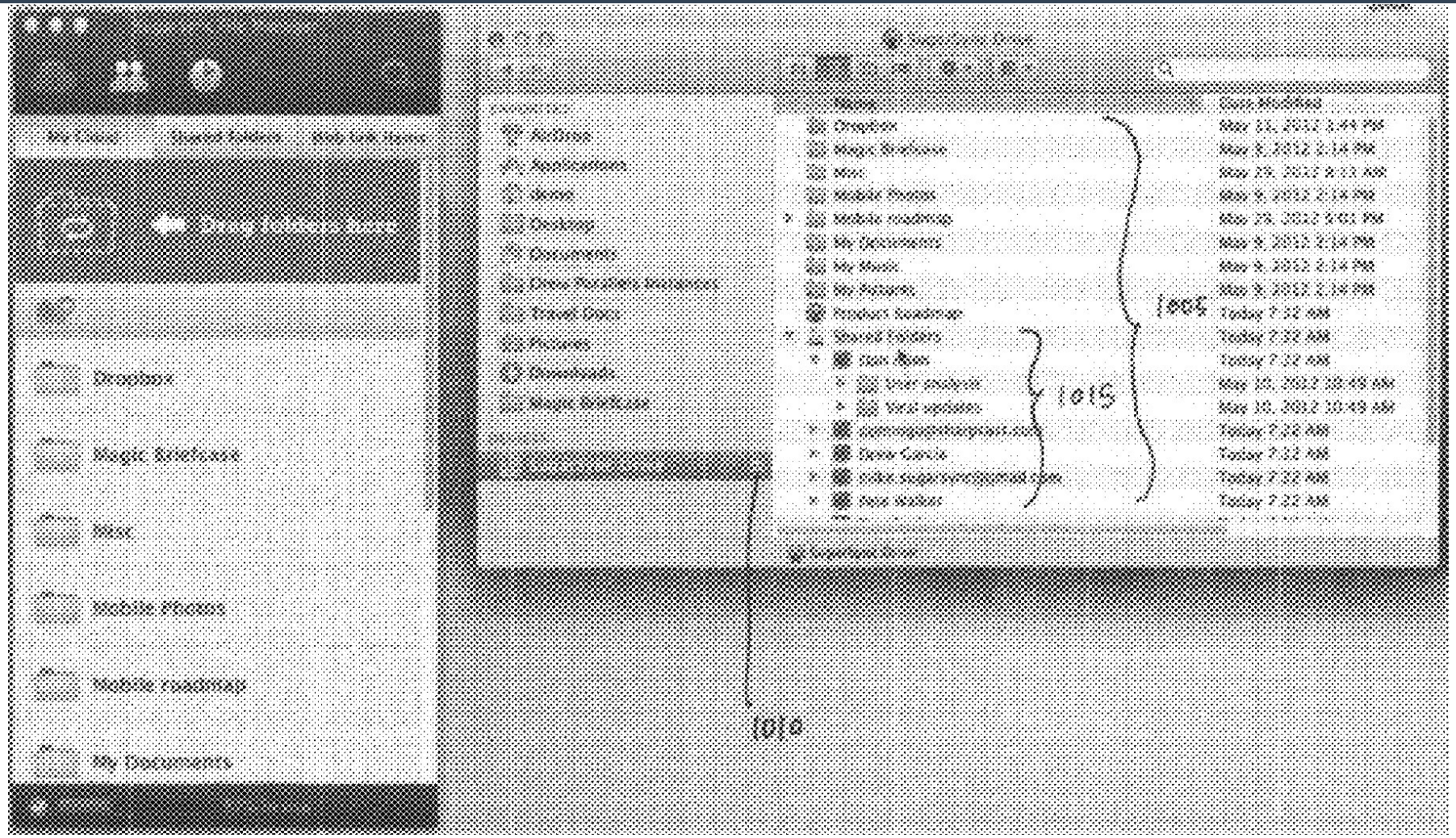
puter). FIG. **16** shows a user interface **1600** on the desktop of a second computer (e.g., a WINDOW-based laptop computer), which has one or more folders synchronized to the first computer. As seen in FIGS. **17-18**, when the user of the first computer drags a folder labeled "June Meeting Notes" **1705** to the drop-target zone **1710** of the user interface **1500**, the folder **1705** is uploaded to the synchronization server **220** and added to the list of folders **1805** of the user interface **1500** of the first computer. The folder **1705** is also added to the user interface **1600** of the second computer.

(EX1004, 12:1-10)

(PO Sur-reply at 15-16)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

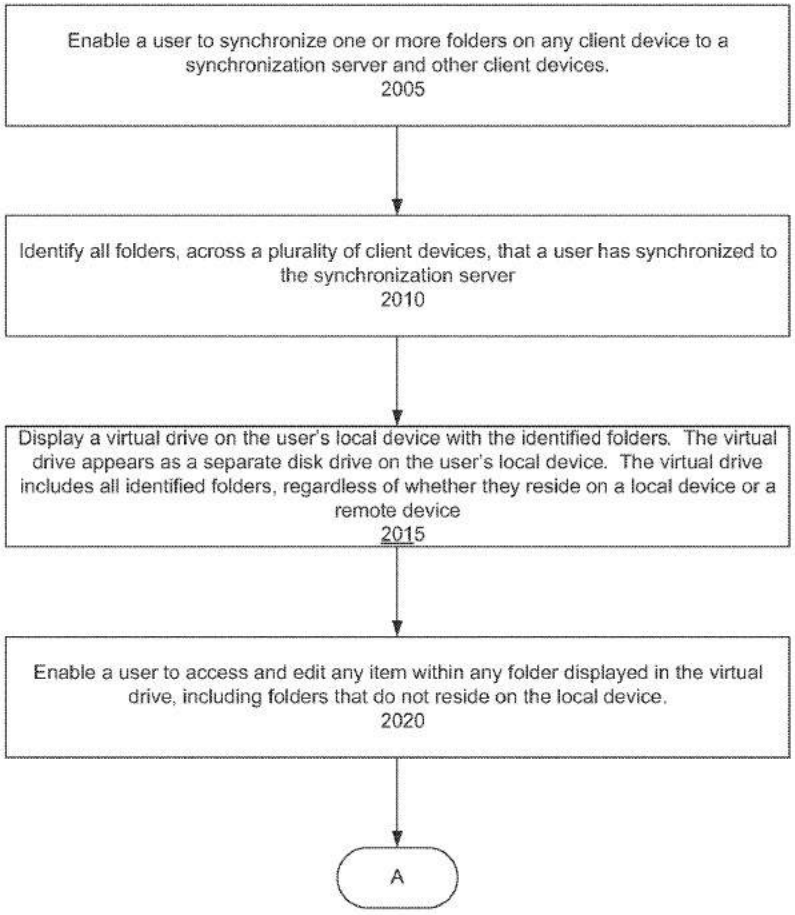
Petitioner Mischaracterizes *Garcia*



(EX1004 Fig. 10)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Mischaracterizes the Teachings of *Garcia*



(EX1004 Fig. 20a)

Moreover, a link to a shared folder is not a "representation" of the folder created at the recipient's node. Dr. Mowry's conclusory testimony seeks to gap-fill by claiming a POSITA would have understood that a second user still needs to access the folder through the virtual interface. (EX1034, ¶38.) But *Garcia* does ***not*** state that; indeed, PO's position is that *Garcia* discloses that when the link is accessed the folder contents are downloaded locally at the device being used and at most the folder and contents would show up as remotely stored objects if the user accessed its virtual drive from a ***different device.*** Petitioner relies heavily on *Garcia*'s virtual drive but it does not connect any disclosure in *Garcia* involving the receipt of the link sharing the folder and the creation of a "***representation***" of the shared folder at the device being used when the link is accessed.

(PO Sur-reply at 15-16)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Teaches Away from the Proposed *Houston-Garcia* Combination

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Moreover, modifying *Houston* from a system intended to download the contents of files automatically and immediately to one that merely functions as a "virtual drive" would have frustrated the heart of the inventions of *Houston* and would render it inoperable for its intended purposes, namely, a system designed to allow users to access a shared folder across a plurality of client devices, such that "each client participating in [a shared folder]" would be provided with "a current version of the folder and its contents." (EX1003, Abstract; EX2051 ¶105.) The proposed combination of *Houston* with *Garcia* would therefore result in the loss of "key functions" and "benefit" of *Houston*, namely, in making a shared folder and its up-to-date contents available to all client devices of the users the folder has been shared with. (EX2051 ¶105.) Such a "loss of key functions" supports a

(POR at 46)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Teaches Away from the Proposed *Houston-Garcia* Combination

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Moreover, modifying *Houston* from a system intended to download the contents of files automatically and immediately to one that merely functions as a "virtual drive" would have frustrated the heart of the inventions of *Houston* and would render it inoperable for its intended purposes, namely, a system designed to allow users to access a shared folder across a plurality of client devices, such that "each client participating in [a shared folder]" would be provided with "a current version of the folder and its contents." (EX1003, Abstract; EX2051 ¶105.) The proposed combination of *Houston* with *Garcia* would therefore result in the loss of "key functions" and "benefit" of *Houston*, namely, in making a shared folder and its up-to-date contents available to all client devices of the users the folder has been shared with. (EX2051 ¶105.) Such a "loss of key functions" supports a

(POR at 46)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Teaches Away from the Proposed *Houston-Garcia* Combination

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Fifth, a POSITA would have been motivated to combine the teachings of Houston and Garcia because Garcia expressly disclosed a benefit of sending an email message with a link to a shared folder—"One benefit of sending links to folders or files to other users is it allows other users to access the folders or files without requiring the other user to create an account in the synchronization system." EX1004, 10:40–43. A POSITA would have recognized other benefits—*e.g.*,

92. ***Fifth***, a POSITA would have been motivated to combine the teachings of Houston and Garcia because Garcia expressly disclosed a benefit of sending an email message with a link to a shared folder—"One benefit of sending links to folders or files to other users is it allows other users to access the folders or files without requiring the other user to create an account in the synchronization system." Exhibit 1004, 10:40–43. A POSITA would have recognized other benefits—*e.g.*,

(Pet at 31; Ex1002, Para. 92; Sur-Reply at 14-16)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Teaches Away from the Proposed *Houston-Garcia* Combination

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Houston in combination with *Garcia* does not disclose elements 17[d]-[g]. Indeed, Petitioner has admitted that *Garcia* discloses that sharing folders using links provides the content even when the recipient does not have an account. (Pet. 31 ("*Garcia* expressly disclosed a benefit of sending an email message with a link to a shared folder—'One benefit of sending links to folders or files to other users is it allows other users to access the folders or files without requiring the other user to create an account in the synchronization system.'")(quoting EX1004, 10:40–43); EX1002, ¶92.) But without an account, they would not have a virtual drive. Therefore, this supports PO's argument that the links provide access to the shared folder and its content to any recipient. That is consistent with *Garcia's* other disclosures that "[w]hen the user clicks on the send link button 1220, a link to the folder 1115 is sent to the contact. When the contact clicks on the link to folder 1115, the contents of the folder are shared with the contact." (EX1004, 11:26-31; *see also* 11:16-19; Fig. 12.) It is also consistent with contemporaneous evidence that PO provided regarding SugarSync. (EX2051, ¶61; EX2011 at 75-76; EX2010 at 78; POR, 22-24.)[12]

(Sur-Reply at 14-15)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Teaches Away from Proposed *Houston-Garcia-*Manzano Combination

17[d]: . . . for permitting **avoidance of at least one file from being communicated to and stored at the second node until an initiation of the communication of the at least one file by a user of the second node is detected** and the communication commences via **at least one server that stores the at least one file**[.]

17[e]: receiving, from the second node and at [] least one server, a signal for causing creation of a first **representation of the at least one folder**, in a location among one or more folders, that is stored at the at least one server and that is displayable via at least one web page;

17[f]: causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first **representation of the at least one folder** that is stored at the at least one server; and

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

In view of the above, Petitioner's argument that "[n]othing in *Garcia's* specification indicates clicking on the link causes downloading to occur" (Reply, 16), and "a POSITA would have known the second user still needs to access the folder by going through the virtual interface" (*id.*, 17) are demonstrably wrong. For the same reasons, Petitioner's assertion that PO's arguments are "contrary to the intent of *Garcia*" are baseless. (Reply, 18.) These statements also contradict Petitioner's reliance on *Garcia*'s disclosure that recipients ***without accounts*** receive the shared folder and its content by accessing the link. (Pet., 31; EX1002, ¶92; EX1004, 10:40-44.) If anything, it would be "contrary to the intent of *Garcia*" to eliminate one of the express benefits noted in the specification, which is the ability to share folders and content with people that do not have an account and virtual drive. (*Id.*)

For these reasons, the proposed combination of *Houston* with *Garcia* and/or *Manzano* would result in the loss of a key benefit disclosed by *Garcia*, which Petitioner ***itself*** relies on in its Petition. This supports the patentability of the challenged claims. *AstraZeneca*, 19 F.4th at 1337; *In re Urbanski*, 809 F.3d at 1244; *Samsung*, No. IPR2023-01100, Paper 11.

(Sur-Reply at 16-17)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Does Not Cure Deficiencies of *Houston* and *Garcia* and No Motivation to Combine (Grounds 2 and 4)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Relies on Manzano for the "A second representation of the at least one Folder"

renders obvious claims 1–8 and 17–21. However, to the extent PO contends that this combination does not disclose, teach, suggest, or render obvious (1) "*the storage at the second node does not store the at least one file when the creation of the representation of the at least one folder is caused*" as recited by element 1[h], (2) "*an additional representation on the file explorer interface is caused to be created that represents the at least one file*" as recited by claim 3, (3) "*the storage at the second node does not store any file in the at least one data object when the creation of the representation of the at least one data object is caused*" as recited by element 7[f], and (4) "*a second representation of the at least one folder*" as recited by element 17[g] and to the extent PO further contends that a representation of a folder in a file explorer interface implies that at least one file in the folder is not downloaded when the representation of the folder is caused[10], the combination of Houston, Garcia, and Manzano renders these features obvious. EX1002, ¶166.

(Petition at 64-65; Sur-reply at 17-18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Directed to Purported Problems Transferring Files Using Email

U.S. Pat. Appl. Pub. No. 2010/0005138 to Manzano (EX1005) ("*Manzano*") is titled "Electronic File Sharing" and purports to address problems involving "exchang[ing] large files using awkward prior-art methods such as e-mail attachments . . . ." (EX1005 ¶ [0003]; EX2051 ¶62.) *Manzano* teaches sharing files between devices using "mobile agents" or a "mobile agent object" that exists on a first computer system and can "transplant itself" to a second computer system, while preserving its current execution state. (EX1005 ¶ [0016]; EX2051 ¶62.) This is performed by the mobile object creating another mobile object that functions as a "proxy object" on the server. (EX1005 ¶ [0035]; EX2051 ¶62.)

Specifically, *Manzano*'s Figure 3 describes that a user may share files between a first computer and a second computer using a "file transfer application" with "file transfer clients" running at each computer device. (EX1005 ¶ [0032]; EX2051 ¶63.) *Manzano* teaches that files may be shared with the second computer by placing them—such as by dragging and dropping them—in a designated folder. (EX1005 ¶¶ [0033], [0046]–[0049]; EX2051 ¶63.) The server transfers portions of the shared file to the second computer by creating a separate mobile object that informs the recipient client of the file transfer. (EX1005 ¶¶ [0036], [0039];

FIELD OF THE INVENTION

[0002] This invention relates generally to computer-implemented processes and, more specifically, to sharing of electronic files among electronic devices.

BACKGROUND OF THE INVENTION

[0003] Computer-network users typically exchange large files using awkward prior-art methods such as e-mail attachments or file transfer protocol (FTP) sites. Most email systems limit the size of attachments, and, as a result, users are forced to send multiple e-mails with discrete files attached to each. FTP sites are typically only moderately secure and their use is often difficult to manage for inexperienced users.

(POR at 24; EX2051 at 62-63)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Directed to Purported Problems Transferring Files Using Email

U.S. Pat. Appl. Pub. No. 2010/0005138 to Manzano (EX1005) ("*Manzano*") is titled "Electronic File Sharing" and purports to address problems involving "exchang[ing] large files using awkward prior-art methods such as e-mail attachments . . . ." (EX1005 ¶ [0003]; EX2051 ¶62.) *Manzano* teaches sharing files between devices using "mobile agents" or a "mobile agent object" that exists on a first computer system and can "transplant itself" to a second computer system, while preserving its current execution state. (EX1005 ¶ [0016]; EX2051 ¶62.) This is performed by the mobile object creating another mobile object that functions as a "proxy object" on the server. (EX1005 ¶ [0035]; EX2051 ¶62.)

Specifically, *Manzano*'s Figure 3 describes that a user may share files between a first computer and a second computer using a "file transfer application" with "file transfer clients" running at each computer device. (EX1005 ¶ [0032]; EX2051 ¶63.) *Manzano* teaches that files may be shared with the second computer by placing them—such as by dragging and dropping them—in a designated folder. (EX1005 ¶¶ [0033], [0046]–[0049]; EX2051 ¶63.) The server transfers portions of the shared file to the second computer by creating a separate mobile object that informs the recipient client of the file transfer. (EX1005 ¶¶ [0036], [0039];

[0048] Referring to FIG. **8**, in an embodiment of the invention, the user may drag (as indicated by the dashed line **820**), using a pointer or other selection device, such as mouse **161**, or otherwise place into the interface **350**, or other user interface, such as an electronically generated graphical window, a first icon **810** displayable on a screen **700** of the display device **250** and representing a file to be transferred. In an embodiment, the icon **810** may represent a copy of the file, rather than, or in addition to, the file itself, thereby allowing transfer of the file copy rather than the file itself. The user may additionally drag or otherwise place (as indicated by the dashed line **840**) into the interface **350** a second icon **830** (e.g., an icon of a "v-card" including an email address) displayable on the display device **250** and representing a recipient. Dragging the first and second icons **810**, **830** into the interface **350** automatically causes the file associated with the first icon **810** to be transferred to the recipient associated with the second icon **830**. Similarly, dragging a third icon **850** (as indicated by the dashed line **860**) representing an additional recipient into the interface **350** automatically causes the file associated with the first icon **810** to be transferred to the recipient associated with the third icon **850**. Such file transfers described with reference to FIG. **8** may be effected using techniques described with reference to FIGS. **1-5** elsewhere herein.

(POR at 24; EX2051 at 62-63)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Directed to Purported Problems Transferring Files Using Email

U.S. Pat. Appl. Pub. No. 2010/0005138 to Manzano (EX1005) ("*Manzano*") is titled "Electronic File Sharing" and purports to address problems involving "exchang[ing] large files using awkward prior-art methods such as e-mail attachments . . . ." (EX1005 ¶ [0003]; EX2051 ¶62.) *Manzano* teaches sharing files between devices using "mobile agents" or a "mobile agent object" that exists on a first computer system and can "transplant itself" to a second computer system, while preserving its current execution state. (EX1005 ¶ [0016]; EX2051 ¶62.) This is performed by the mobile object creating another mobile object that functions as a "proxy object" on the server. (EX1005 ¶ [0035]; EX2051 ¶62.)

Specifically, *Manzano*'s Figure 3 describes that a user may share files between a first computer and a second computer using a "file transfer application" with "file transfer clients" running at each computer device. (EX1005 ¶ [0032]; EX2051 ¶63.) *Manzano* teaches that files may be shared with the second computer by placing them—such as by dragging and dropping them—in a designated folder. (EX1005 ¶¶ [0033], [0046]–[0049]; EX2051 ¶63.) The server transfers portions of the shared file to the second computer by creating a separate mobile object that informs the recipient client of the file transfer. (EX1005 ¶¶ [0036], [0039];

[0049] Referring to FIG. **9**, in an embodiment of the invention, the user may drag or otherwise place (as indicated by the dashed line **920**) a first icon **910** on or partially coinciding with (e.g., superimposed on) a second icon **930** displayable on a screen **700** of the display device **250** and, perhaps, located in the interface **350**. The first icon **910** may represent a file to be transferred. In an embodiment, the first icon **910** may represent a copy of the file, rather than, or in addition to, the file itself, thereby allowing transfer of the file copy rather than the file itself. The second icon **930** may represent a recipient (i.e., may be an icon of a "v-card" including an email address). At least partially superimposing one of the first and second icons **910**, **930** with the other automatically causes the file associated with the first icon **910** to be transferred to the recipient associated with the second icon **930**. This transfer operation may similarly be performed with respect to a third icon **950** also representing a recipient. For example, the second and third icons **930**, **950** may be simultaneously selected, placed in close proximity to one another, or otherwise coupled using known graphical-user-interface techniques. Subsequently, at least partially superimposing the first icon **910** on the second and/or third icon **930**, **950** automatically causes the file associated with the first icon to be transferred to the recipients associated with the second and third icons. Such file transfers described with reference to FIG. **9** may be effected using techniques described with reference to FIGS. **1-5** elsewhere herein.

(POR at 24; EX2051 at 62-63)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Directed to Transferring Files, Not Folders, Using Mobile Agents Without Email

Moreover, *Manzano* is not "from the same field of endeavor" as the '520 patent, nor from *Garcia*'s. (Pet. at 69; EX2051 ¶116.) Both the '520 patent and *Garcia*—other differences aside—are directed primarily to sharing folders between users. (EX2004 ¶ 5; EX2051 ¶116.) But, as noted above, *Manzano* does not teach sharing folders with another user at all. (EX2051 ¶116.) Instead, *Manzano* is directed to allowing a user to transfer their own files from a first computer to a second computer, and its file-transfer method is far removed from the '520 patent's methods. (*See, e.g.*, EX1005 ¶¶ [0032]–[0035] (describing "mobile agents" and "mobile objects") [0033]–[0043] (describing "proxy objects"), [0050] (describing "ghosted" icons).) Nothing in *Manzano* describes sharing entire folders with another user by email without attachments, nor creating representations of those shared folders, nor downloading contents of those folders as recited in claim 17. (EX2051 ¶116.) Therefore, a POSITA would not have been motivated to combine

[0032] Referring now to FIG. **3**, illustrated is functionality of an embodiment of the invention allowing a user (not shown) of the user device **210** to transfer a file to the user device **280**. In an embodiment, an administrator (not shown) of the server **230** or other appropriate electronic device transfers a file-transfer application to the user devices **210**, **280** for installation thereon. Once installed on the user devices **210**, **280**, the file-transfer application provides file-transfer clients **310**, **320** executable by the user devices **210**, **280**, respectively. Each of the file-transfer clients **310**, **320** includes a respective mobile-agent runtime environment **330**, **340**. The mobile-agent runtime environment **330**, **340** include portions of memory of the user devices **210**, **280** dedicated to allowing a mobile object the ability to perform operations that the mobile object is programmed to carry out. Also included in the file-transfer application are user interfaces **350**, **360** that are displayable on the displays **250**, **290**, respectively. In an embodiment, the interfaces **350**, **360** correspond to respective electronic folders or windows into which files may be placed. Each such electronic folder may be assigned a user group defining users having permission to receive and/or send files via a particular folder.

(POR at 52-53)

Manzano Directed to Transferring Files, Not Folders, Using Mobile Agents Without Email

Moreover, *Manzano* is not "from the same field of endeavor" as the '520 patent, nor from *Garcia*'s. (Pet. at 69; EX2051 ¶116.) Both the '520 patent and *Garcia*—other differences aside—are directed primarily to sharing folders between users. (EX2004 ¶ 5; EX2051 ¶116.) But, as noted above, *Manzano* does not teach sharing folders with another user at all. (EX2051 ¶116.) Instead, *Manzano* is directed to allowing a user to transfer their own files from a first computer to a second computer, and its file-transfer method is far removed from the '520 patent's methods. (*See, e.g.*, EX1005 ¶¶ [0032]–[0035] (describing "mobile agents" and "mobile objects") [0033]–[0043] (describing "proxy objects"), [0050] (describing "ghosted" icons).) Nothing in *Manzano* describes sharing entire folders with another user by email without attachments, nor creating representations of those shared folders, nor downloading contents of those folders as recited in claim 17. (EX2051 ¶116.) Therefore, a POSITA would not have been motivated to combine

[0033] To initiate sharing of a file with a user of the user device **280**, the user of the user device **210** may place a file into the folder corresponding to the interface **350** and with which the user device **280** has been associated as a recipient. The user may do so through conventional actions such as, for example, "dragging and dropping" an icon representing the file into the interface **350** or opening the file using commands associated with the interface **350**. The source client **310** monitors the folder corresponding to the interface **350**. Upon noting the placement of the file in the interface **350**, the source client **310** polls the server **230** to determine the recipient group including, in this case, the user device **280** and places a request with the server to share the file.

[0034] In response to the file-sharing request, the server **230** returns a transfer token (not shown) the source client **310**. In an embodiment, the transfer token includes a data string that functions as a transaction identifier corresponding to the file. Upon receiving the transfer token, the source client **310** creates a transfer mobile object (TMO) **370** that includes the token and information characterizing the file (e.g., recipient identification, sender identification, file name, file workspace (i.e., name of folder corresponding to the interface **350** and into which the file was placed for sharing), file size, etc.) (collectively, "file-transfer information").

[0035] The TMO **370** migrates to a mobile-agent runtime environment **380** executing on the server **230** and creates a proxy object **390** that functions as the TMO's server-side agent for the file transfer. In an embodiment, the proxy **390** is, itself, a mobile object. After creating the proxy **390**, the TMO **370** returns to the source client **310**.

(POR at 52-53)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Acknowledges *Manzano* Is Inapposite to Creating a Representation of a Shared Folder at the Second Node

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

Even under PO's proposed claim interpretation, PO's arguments fail. PO argues Manzano does not disclose a representation "displayable via the file explorer interface" (POR, 51-52) but Petitioner did not rely on Manzano for that feature of 17[g]. The Petition relied on Manzano for teaching that the second node does not store a shared file until the user initiates the communication—a feature that PO acknowledges Manzano discloses. *See* POR, 52 ("Manzano's system

(Rep. at 23)

7[f], and (4) "*a second representation of the at least one folder*" as recited by element 17[g] and to the extent PO further contends that a representation of a folder in a file explorer interface implies that at least one file in the folder is not downloaded when the representation of the folder is caused[10], the combination of Houston, Garcia, and Manzano renders these features obvious. EX1002, ¶166.

(Pet. at 64-65

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Does Not Teach Creating a Representation of a Shared Folder at the Second Node

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

¶114.) *Manzano* teaches that a recipient client is instructed "to set up an electronic file folder into which the transferred file will be place." (EX1005, ¶ [0041].) It does not teach a representation of a shared folder, much less a representation of a shared folder being created in a location among one or more folders" "that is displayable via the file explorer interface of the client-based file explorer application" as recited in claim element 17[g]. (EX2051 ¶114.) Indeed, *Manzano* only teaches transferring ***files*** rather than entire folders in the manner recited in claim 17. (*Id.*)

(POR at 51-52)

POR, 51-52.) *Manzano* relates to transferring ***files***, not folders, and its "ghosted icon" refers only to files. (*Id.*; EX1005, ¶50.). It also discloses that a client creates "an electronic file folder into which the transferred file will be place." (EX1005, ¶41.) It ***does not*** teach a representation of a shared folder, much less a representation of a shared folder being created in a location among one or more folders" "that is displayable via the file explorer interface of the client-based file explorer application" as recited in claim element 17[g]. (EX2051 ¶114.)

(PO Sur-reply at 18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Does Not Teach Creating a Representation of a Shared Folder at the Second Node

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

sharing folders with another user at all. (EX2051 ¶116.) Instead, *Manzano* is directed to allowing a user to transfer their own files from a first computer to a second computer, and its file-transfer method is far removed from the '520 patent's methods. (*See, e.g.*, EX1005 ¶¶ [0032]–[0035] (describing "mobile agents" and "mobile objects") [0033]–[0043] (describing "proxy objects"), [0050] (describing "ghosted" icons).) Nothing in *Manzano* describes sharing entire folders with another user by email without attachments, nor creating representations of those shared folders, nor downloading contents of those folders as recited in claim 17.

(POR at 53; EX2051 at 116)

Manzano Does Not Teach Creating a Representation of a Shared Folder at the Second Node

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

[0041] Once the RTMO **315** returns to the user device **280**, it establishes a network connection with the proxy **325**. Additionally, the RTMO **315** instructs the recipient client **320** to set up an electronic file folder into which the transferred file will be placed.

[0050] In an embodiment, file icons, such as the icon **600**, in a workspace displayed on display device **290** can visually indicate whether a corresponding file (such as a file, the transfer of which may be as is discussed herein with reference to FIG. **3**) has been downloaded to or is otherwise stored on (i.e., is local to), or has otherwise been made available (e.g., sent or in transit) to the user device **280**. For example, if the source client **310** has initiated, or has otherwise attempted to initiate, the sharing of a file with the recipient client **320**, the interface **360** associated with the user device **280** may display an icon, such as icon **600**, identifying or otherwise representing the file to be shared by source client **310**. If the corresponding file (i.e., file to be shared by source client **310**) is not local to user device **280**, the icon **600** may be ghosted (i.e., semi-transparent) or otherwise be visually configured to indicate the non-local status of the file. As such, the icon **600** can appear in the workspace in some manner whether or not the data associated with the icon **600** has actually been downloaded to, or otherwise received by, the user device **280**.

(EX1005 at [0041], [0050])

(POR at 51-53; Sur-reply at 18)

Manzano Does Not Teach Creating a Representation of a Shared Folder at the Second Node

17[g]: causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: cause creation of a second **representation of the at least one folder**, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.

[0048] Referring to FIG. **8**, in an embodiment of the invention, the user may drag (as indicated by the dashed line **820**), using a pointer or other selection device, such as mouse **161**, or otherwise place into the interface **350**, or other user interface, such as an electronically generated graphical window, a first icon **810** displayable on a screen **700** of the display device **250** and representing a file to be transferred. In an embodiment, the icon **810** may represent a copy of the file, rather than, or in addition to, the file itself, thereby allowing transfer of the file copy rather than the file itself. The user may additionally drag or otherwise place (as indicated by the dashed line **840**) into the interface **350** a second icon **830** (e.g., an icon of a "v-card" including an email address) displayable on the display device **250** and representing a recipient. Dragging the first and second icons **810**, **830** into the interface **350** automatically causes the file associated with the first icon **810** to be transferred to the recipient associated with the second icon **830**. Similarly, dragging a third icon **850** (as indicated by the dashed line **860**) representing an additional recipient into the interface **350** automatically causes the file associated with the first icon **810** to be transferred to the recipient associated with the third icon **850**. Such file transfers described with reference to FIG. **8** may be effected using techniques described with reference to FIGS. **1-5** elsewhere herein.

(EX1005 at [0041], [0050])

(POR at 51-53; Sur-reply at 18)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Motivation to Combine and Teaching Away

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Motivation to Combine and Teaching Away

2011) (internal citation omitted). Rather, Petitioner also has the burden to show there would have been some motivation to modify or combine the asserted prior art, and that the proposed modifications or combinations would render the patented claims obvious. *In re Magnum Oil Tools Int'l.*, 829 F.3d at 1376. It does not suffice that an element is simply known in a prior art reference; "a reason for combining must exist." *Virtek Vision Int'l ULC v. Assembly Guidance Sys., Inc.*, 97 F.4th 882 (Fed. Cir. 2024).

(POR at 39)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Motivation to Combine and Teaching Away

A reference can "teach away" from combining prior art references when a POSITA, "upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant." *Ricoh Co., Ltd. v. Quanta Computer Inc.*, 550 F.3d 1325, 1332 (Fed. Cir. 2008) (quoting *In re Kahn*, 441 F.3d 977, 990 (Fed. Cir. 2006)).

(POR at 39-40)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Motivation to Combine and Teaching Away

Motivation and expectation of success are generally not present where the proposed modification or combination contradicts a reference's basic teachings or is inoperable. *AstraZeneca AB v. Mylan Pharms. Inc.*, 19 F.4th 1325, 1337 (Fed. Cir. 2021); *In re Urbanski*, 809 F.3d 1237, 1244 (Fed. Cir. 2016) (same).

(POR at 39-40)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Motivation to Combine and Teaching Away

Moreover, modifying *Houston* from a system intended to download the contents of files automatically and immediately to one that merely functions as a "virtual drive" would have frustrated the heart of the inventions of *Houston* and would render it inoperable for its intended purposes, namely, a system designed to allow users to access a shared folder across a plurality of client devices, such that "each client participating in [a shared folder]" would be provided with "a current version of the folder and its contents." (EX1003, Abstract; EX2051 ¶105.) The proposed combination of *Houston* with *Garcia* would therefore result in the loss of "key functions" and "benefit" of *Houston*, namely, in making a shared folder and its up-to-date contents available to all client devices of the users the folder has been shared with. (EX2051 ¶105.) Such a "loss of key functions" supports a finding of no motivation to combine the references. *Cook Grp., Inc.*, 809 F. App'x at 1000); *see also Samsung Elecs. Co. v. Mojo Mobility, Inc.*, No. IPR2023-01100, Paper 11, 2024 Pat. App. LEXIS 574, *14–15 (PTAB Feb. 12, 2024).

(POR at 46)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Motivation to Combine and Teaching Away

Further, the proposed combination, on this record, raises a concern over the operability of the proposed combination. *In re Urbanski*, 809 F.3d 1237 (Fed. Cir. 2016) ("if references taken in combination would produce a seemingly inoperative device such references teach away from the combination and thus cannot serve as predicates for a prima facie case of obviousness.") (cleaned up); *Plas-Pak Industries, Inc. v. Sulzer Mixpac AG*, 600 F. App'x. 755, 758 (Fed. Cir. 2015) ("combinations that change the basic principles under which the prior art was designed to operate, or that render the prior art inoperable for its intended purpose, may fail to support a conclusion of obviousness.") (cleaned up). . . . . It is unclear to us that such a device would function. . . . *Cook Grp. Inc. v. Bos. Sci. Scimed, Inc.*, 809 F. App'x 990, 1001 (Fed. Cir. 2020) ("the Board properly weighed the loss of key functions as a relevant factor in the motivation to combine analysis"). Petitioner has not discussed these changes in its analysis. As a whole, the combination raises questions about its operability, further lending to the view that Petitioner has not established a reasonable likelihood of success.

Samsung Elecs. Co. v. Mojo Mobility, Inc., No. IPR2023-01100, Paper 11, 2024 Pat. App. LEXIS 574, *14–15 (PTAB Feb. 12, 2024).

(POR at 46)

Objective Indicia of Non-Obviousness Support PO

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Dropbox Did Not Introduce Representations of Shared Folders Until 2016

Published on January 30, 2017

- Dropbox introduces Smart Sync—the industry's first cross-platform, on-demand cloud storage solution
- Dropbox Paper, a flexible workspace for teams, officially launches in 21 languages worldwide
- All new Dropbox web interface brings modern collaboration to files
- Three new business packages allow companies to tailor Dropbox to their needs

Bring your file server to the cloud with Smart Sync

Smart Sync is a Dropbox feature that makes all the content in a user's Dropbox account seamlessly accessible from their desktop file system — and the content takes up virtually no local disk space until it's needed. Smart Sync works across Windows and Mac computers, and is fully backwards compatible to Windows 7 and Mac OS X 10.9. This is the industry's first cross-platform, on-demand cloud storage solution.

(EX2012)

(POR at 54)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Dropbox Did Not Introduce Representations of Shared Folders Until 2016

Published on April 26, 2016

These days, modern companies and truly productive teams often span offices, regions, and operating systems. Sadly, the tools that many rely on to access their data—traditional ones like corporate file servers (shared network drives) and newer, browser-based solutions—aren't suited to the way modern teams work. Today, at Dropbox Open London, we showcased a new technology preview that will tackle this problem head on: Project Infinite.

it's stored locally or in the cloud, will show up in Dropbox on your desktop. If it's synced locally, you'll see the familiar green checkmark, while everything else will have a new cloud icon. https://youtu.be/iT4rAaV9MhU Through Project Infinite, teams will gain:

- **Visibility in context.** Every file you've been given access to—even ones that aren't stored locally—will appear in Windows File Explorer and Mac OS X Finder. You can quickly drill down through folders to find what you need, without the lag of a network drive or the inconvenience of a web app. Plus, you can view key info like file size, and creation and modification dates through your desktop file system, no downloading required.

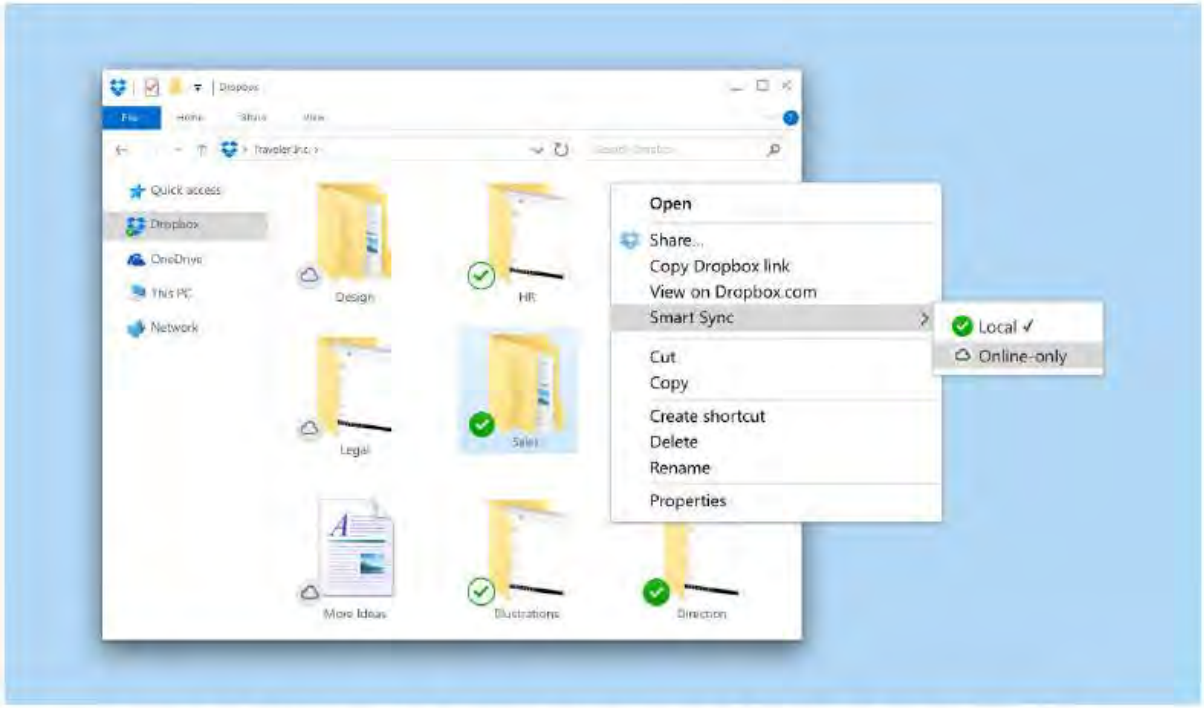
(EX2013)

(POR at 54)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Dropbox Did Not Introduce Representations of Shared Folders Until 2016

Published on January 30, 2017



Screenshot of Windows File Explorer showing right-click menu for Dropbox Smart Sync

Users working with just 128 GB of hard drive space can easily comb through terabytes of files to find exactly what they need—right from Windows File Explorer or macOS Finder. Now, they won't need to take extra steps—like switching to a web browser—just to view files. And whenever they need to access files stored in the cloud, users can download them with a quick double click. Smart Sync also provides the controls admins need to manage their

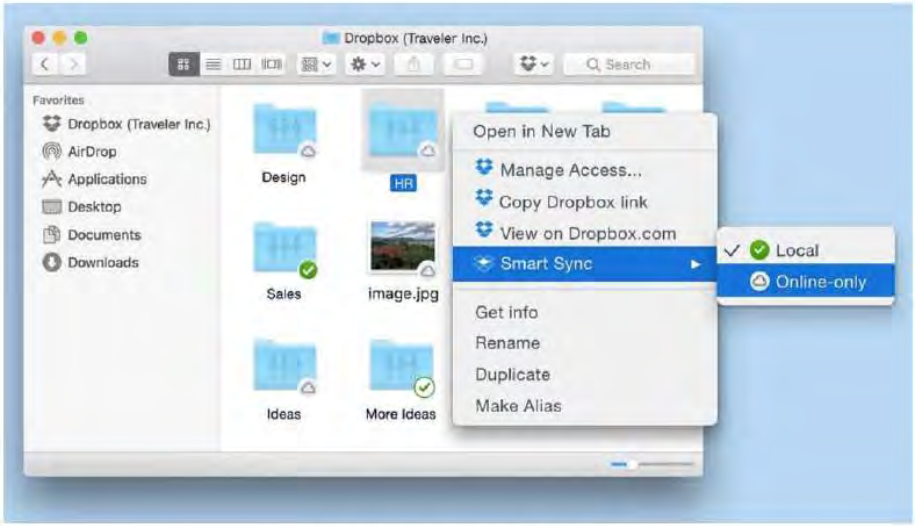
(POR at 54)

(EX2013)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Dropbox Did Not Introduce Representations of Shared Folders Until 2016

Jan. 30, 2017



Dropbox Smart Sync. Via Dropbox.

Cloud storage vendor Dropbox has spent the last several years building out products for its pivot to the enterprise software space -- and now two of those products are making an official debut.

At a media event in San Francisco, Calif., on Monday, Dropbox announced the release of Smart Sync, the formal name for what was previously dubbed Dropbox Project Infinite, as well as Paper, the company's note-taking and collaboration tool.

Smart Sync manages cloud files as if they were local, allowing users to access all files and folders in context on the desktop, but without eating up a ton of storage space on local hard drives. Smart Sync is cross platform and backwards compatible, meaning it works with all devices and operating systems -- something Dropbox claims is an industry first.

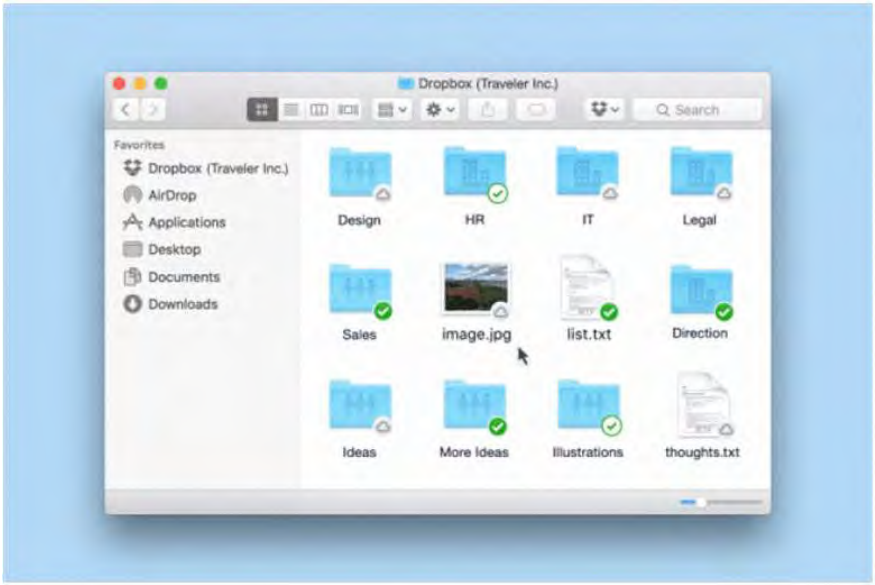
(EX2019)

(POR at 55-56)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Dropbox Did Not Introduce Representations of Shared Folders Until 2016

Published on May 09, 2017



Here's how Smart Sync can help your team get more out of Dropbox by letting you put more into it.

2. Make the knowledge base accessible to everyone

Gathering all your information in one location gives you more than the satisfaction of being better organized. It gives you a central hub of information where all your collaborators can access a single source of truth, a library of knowledge with contributions from everyone on your team. Of course, the sum of all your company's knowledge represents a significant amount of data. So you might be wondering how this impacts disk space. That's why we created Smart Sync. It gives everyone on your team access to every file and folder that's been shared in the team folder, right from their desktop.

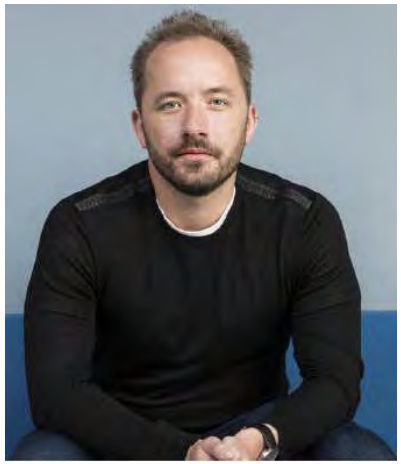
3. Sync global, act local

When you make a new file, it's local by default. But if you have a collection of images, videos, and other large files you want to get off your hard drive, Smart Sync can help. It lets you keep that info and its metadata on your computer, while copying the core space-consuming parts of the file to Dropbox where they're stored until you recall them. All you have to do is right-click the files you want to move from your hard drive to the cloud, then choose Smart Sync "Online Only." When you need to bring them back, just double-click, and Dropbox will automatically download and open them for you. To find out how Smart Sync can help prepare your growing team for large-scale collaboration, check out dropbox.com/smartsync.

(EX2020)

(POR at 55-56)

Dropbox Did Not Introduce Representations of Shared Folders Until 2016



January 30, 2017

For Drew Houston, Dropbox's cofounder and CEO, Paper and the other new initiatives are also the foundation for a reinvention of his company. In its decade of life, cloud storage—even done as slickly as Dropbox does it—has come to be seen widely as a commodity service. With Paper and the other new team-oriented changes, Houston hopes people will stop thinking of Dropbox as just "a home for your stuff."

It will be more like a group home for you and everyone you work with. And it will be a crowded one, because everyone's invited to share. And sync.

That's where Smart Sync comes in. It routes around these problems by making all the files in your group Dropbox, no matter how gigantic it gets, accessible through your computer's file-system interface. It doesn't matter if you don't have enough space on your local drive. The upstream files look like any other item in your Dropbox, but a little cloud icon indicates that they're "on-demand" files: click on one and it will quickly download and open, as if it were there on your disk the whole time.

Ben Newhouse, who leads the engineering team at Dropbox that built Smart Sync, got the idea when he joined the company in 2014 and started working on a Macbook Air. It had a measly 128-gigabyte solid-state drive that couldn't hold his Dropbox files. At first, Newhouse thought with some alarm that if he wanted to create desktop shortcuts for all those remote Dropbox files, he'd have to write a whole new file system from scratch — a task to make any sane engineer quail. Then he figured out he could borrow some less onerous tricks from the world of antivirus tools. By the end of a summer hack week, he had the core of the thing up and running.

Smart Sync is very much in the spirit of the original Dropbox: It lets users keep their old habits and skills while breaking down boundaries between local and network, what's "here" and what's "in the cloud." It may not *feel* like a revolution. But it does make you want to start using it right now.

(POR at 54-55)

(EX2015)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Is Wrong to Suggest that Smart Sync Is Irrelevant to the Claims

First, PO does not, and cannot, meet its burden of showing a nexus exists between Dropbox's Smart Sync feature and claims of the '520 patent. *See, e.g.*, *Fox Factory, Inc. v. SRAM, LLC*, 944 F.3d 1366, 1373 (Fed. Cir. 2019). PO has not shown, and will be unable to show, either in district court or in this proceeding that Dropbox practices the challenged claims. The district court litigation is in its earliest stages—PO served insufficient preliminary infringement contentions; no discovery has occurred; and Dropbox has not yet served non-infringement interrogatory responses or expert reports.

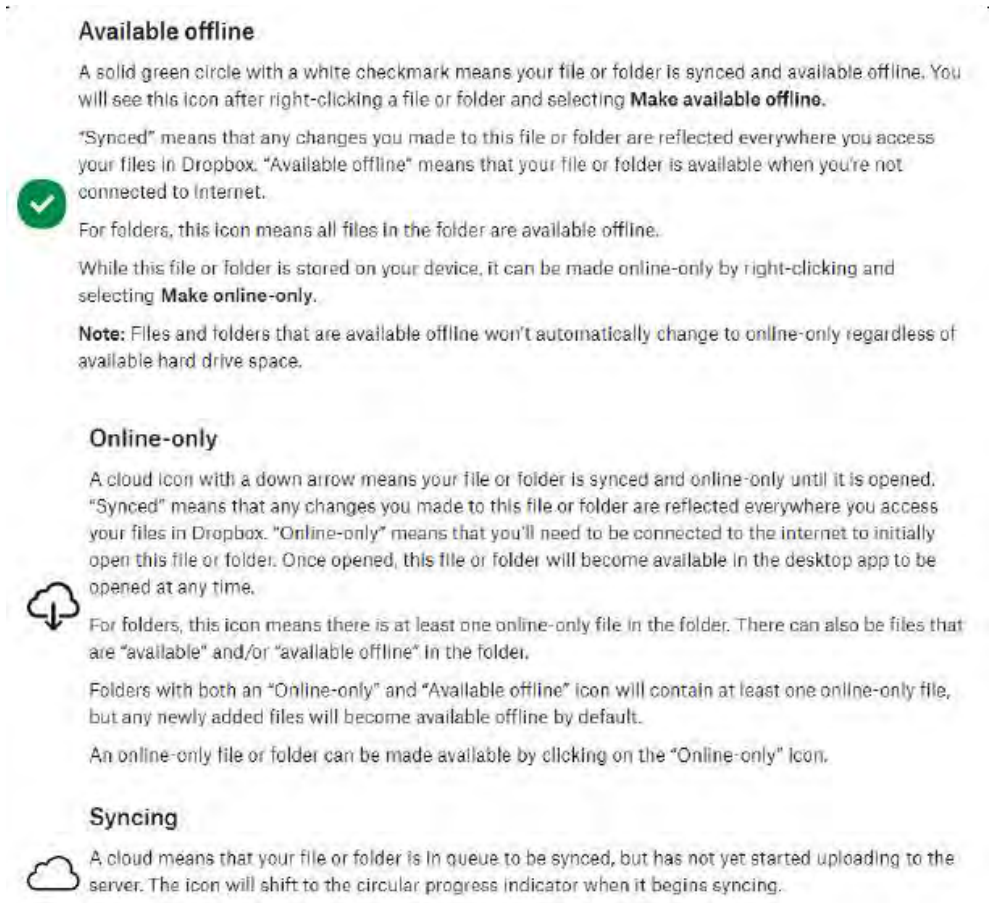
(Rep. at 25)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Is Wrong to Suggest that Smart Sync Is Irrelevant to the Claims

causing, at the second node, receipt of code for storage at the second node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to:

cause creation of a second representation of the at least one folder, in a location among one or more folders, that is stored at the second node and that is displayable via the file explorer interface of the client-based file explorer application.



Available offline

A solid green circle with a white checkmark means your file or folder is synced and available offline. You will see this icon after right-clicking a file or folder and selecting **Make available offline.**

"Synced" means that any changes you made to this file or folder are reflected everywhere you access your files in Dropbox. "Available offline" means that your file or folder is available when you're not connected to Internet.

For folders, this icon means all files in the folder are available offline.

While this file or folder is stored on your device, it can be made online-only by right-clicking and selecting **Make online-only.**

Note: Files and folders that are available offline won't automatically change to online-only regardless of available hard drive space.

Online-only

A cloud icon with a down arrow means your file or folder is synced and online-only until it is opened. "Synced" means that any changes you made to this file or folder are reflected everywhere you access your files in Dropbox. "Online-only" means that you'll need to be connected to the internet to initially open this file or folder. Once opened, this file or folder will become available in the desktop app to be opened at any time.

For folders, this icon means there is at least one online-only file in the folder. There can also be files that are "available" and/or "available offline" in the folder.

Folders with both an "Online-only" and "Available offline" icon will contain at least one online-only file, but any newly added files will become available offline by default.

An online-only file or folder can be made available by clicking on the "Online-only" icon.

Syncing

A cloud means that your file or folder is in queue to be synced, but has not yet started uploading to the server. The icon will shift to the circular progress indicator when it begins syncing.

(PO Sur-reply at 19) (EX2049 at 134-40) DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Is Wrong to Suggest that Smart Sync Is Irrelevant to the Claims

Second, contrary to PO's arguments, the verdict of non-infringement in the related case, *Motion Offense, LLC v. Dropbox, Inc.*, 6:20-cv-251 (W.D. Tex.), does not support a finding of nexus. POR, 59; *see also id.*, 14-17. PO incorrectly argues the jury's non-infringement finding was based only on one claim element—"HTML-equipped code"—that is not recited in any of the claims challenged here. POR, 59. Not so. Dropbox also presented evidence that Dropbox does not make, use, offer to sell, or sell hardware and/or software required by the asserted apparatus claims and that some of the steps required by the asserted method claims were performed by third-parties. EX1035, 2-5.

(Rep. at 25-26)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Is Wrong to Suggest that Smart Sync Is Irrelevant to the Claims

Now, as you know, and as Mr. Devlin mentioned, we didn't go through every element of every claim during the course of this trial that Dropbox doesn't meet because one of the very first claim limitations of every asserted claim is not met. ***And as His Honor has told you, that's enough. One element's not met, you're done***. There's no infringement. And every single asserted claim in this case requires that certain interfaces be displayed using hypertext markup language-equipped code or HTML code.

. . .

Dropbox doesn't use that claim limitation. There is no infringement. That's the end of the story.

(EX2048 at 1375:16-1381:24 (emphasis added))

(PO Sur-reply at 19-20)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Objective Indicia of Non-Obviousness and Nexus

F.3d at 1378. The Federal Circuit has held that "if the marketed product embodies the claimed features, and is coextensive with them, then a nexus is presumed and the burden shifts to the party asserting obviousness to present evidence to rebut the presumed nexus." *Brown & Williamson Tobacco Corp. v. Philip Morris Inc.*, 229 F.3d 1120, 1130 (Fed. Cir. 2000).

(Paper 12 at 19)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Objective Indicia of Non-Obviousness and Nexus

51. In other words, Dropbox's only non-infringement argument at trial hinged on a claim limitation that is absent here, and this suggests a nexus between Dropbox's accused products from the Dropbox Litigation (including Smart Sync) and the challenged claims. Otherwise, if Dropbox could have argued that Smart Sync did not meet the other claim limitations that are "substantially similar" to those in the challenged claims here, it likely would have done so.

52. Similarly, Dropbox's invalidity arguments in the Dropbox Litigation were based primarily on source code that corresponded to prior art systems, which are not available as prior art in an *inter partes* review and do not apply here. *See* 35 U.S.C. § 311(b). One of those systems was "Dropbox 2017," a Dropbox system that included Smart Sync and was used as a prior art reference against the asserted patents, even though it post-dated the asserted patents' priority date, because Dropbox had also successfully challenged those patents' priority date at trial. But here, Petitioner does not dispute that the priority date of the '520 patent is September 22, 2012 (Pet. at 5 n.1)—i.e., five years prior to the Dropbox 2017 reference—so the jury's invalidity verdict does not preclude any issue here. And if anything, to the extent that the jury's invalidity verdict was based on Dropbox 2017, it would suggest a nexus between Dropbox 2017's Smart Sync and any "substantially similar" challenged claim of the '520 patent.

(EX2051, paragraphs 51-52)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Objective Indicia of Non-Obviousness and Nexus

IV. OBJECTIVE INDICIA OF NON-OBVIOUSNESS

Petitioner' reply attacks nexus. (Reply, 25-26.) Its arguments fail. First, PO provided a detailed claim chart demonstrating that Dropbox practices the challenged claims. (EX2049.) Petitioner responds with attorney argument and does not offer any substantive response. (Reply, 25-26.) PO's evidence is unrebutted. Moreover,

(Sur-reply at 19; EX2049 (*see 125-140 for entire claim 17; 133-140 for 17[g]*)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Objective Indicia of Non-Obviousness and Nexus

causing, at the second node, receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first representation of the at least one folder that is stored at the at least one server; and	The Accused Instrumentalities, when in operation, comprise a method for *causing, at the second node* (e.g., second user's device, etc.), *receipt of the at least one web page, that results in display, at the second node and via the at least one web page, the first representation of the at least one folder that is stored at the at least one server;* For example, the following web page is caused to be received (at the second node) by a Dropbox server.

(EX2051, paragraphs 51-52)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston ‘787 Provisional and Motion to Exclude

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

PO Correctly Identified the Houston '787 Provisional

2026	U.S. Provisional Pat. App. No. 61/233,787 (*Houston* II)

POR at vii

Storing copies of files locally at each user is critical to *Houston*'s system. (EX2051 ¶55.) *Houston* repeatedly criticizes systems that store documents remotely—for example, *Houston* notes that "if the user is accessing the document remotely and loses his connection, he may lose his changes, be unable to continue, and may end up with a corrupted document." (EX1003, 1:29–35; EX2026 ¶¶ 8–10, 12–14.) And *Houston*'s parent provisional application, Provisional Patent App. No. 61/233,787 (*Houston II*), which is incorporated by reference, explains that the use of local copies of files at each client is a core feature of the system. (EX2026 ¶ 11; EX2051 ¶55.)

POR at 18

Petitioner's Willful Blindness Cannot Create Prejudice Nor Warrant Exclusion

[5] PO also cites Houston's provisional application. *See* POR, 18-19 (citing EX2026 at ¶¶8-14), 42 46. PO does not quote it. Nothing in the provisional teaches away from a system like Garcia that allows a user to control when the initial download of a file in a newly shared folder occurs. The provisional, like Houston, is focused on changes to shared folders that have already been downloaded. EX2026 ¶9 ("A file system is monitored for changes which takes place to a file. …, synchronization ('syncing') is accomplished based on the identification of the changes.").

Reply at 11-12

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Houston Includes the '787 Provisional Disclosure

Toys, Inc., 122 F. 3d 1396, 1407 (Fed. Cir. 1997). Houston II is considered part of the disclosure of Houston as a matter of law. *Netflix, Inc. v. Avago Techs. Int'l*, Nos. 2024 U.S. App. LEXIS 17998, at *21 (Fed. Cir. July 23, 2024) ("By clearly identifying Appendix J, detailing the subject matter it addresses, and pointing to where it is found, Masters makes clear that Appendix J is part and parcel to Masters itself. We therefore conclude that the Board erred when it declined to consider Appendix J for purposes of its analysis."); *see also Finjan LLC v. ESET, LLC*, 51 F.4th 1377, 1382 (Fed. Cir. 2022); *Trs. of Columbia Univ. v. Symantec Corp.*, 811F.3d 1359, 1365-66 (Fed. Cir. 2016). *Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve, Inc.*, 796 F.2d 443, 448-49 (Fed. Cir. 1986); *Recor Med. Inc. v. Medtronic Ireland Mfg. Unlimited Ltd.*, No. 23-2251 slip op.at 5 (Fed Cir. Mar. 27, 2025) ("A reference must be considered for everything it teaches by way of technology and is not limited to the particular invention it is describing and attempting to protect . . . We agree with Recor that the Board violated this principle here.") (cleaned up).

Paper 41 at 8

PO Correctly Identified the Houston '787 Provisional

Storing copies of files locally at each user is critical to *Houston*'s system. *Houston* repeatedly criticizes systems that store documents remotely, for example noting "if the user is accessing the document remotely and loses his connection, he may lose his changes, be unable to continue, and may end up with a corrupted document." (EX1003, 1:29-35; EX2026, ¶¶ 8-10, 12-14.) *Houston*'s parent provisional application, Provisional Pat. App. No. 61/233,787 (*Houston* II), which is incorporated by reference, explains in detail how the *Houston* system's use of local copies of files at each client is a core feature of the disclosed system:

> Under the traditional file server architecture, the client PC does not have a local copy in a local disk. In contrast, with the Dropbox architecture, the client does the input/output locally. According to various embodiments, the Dropbox software is watching the shared folder and doing efficient synchronization through input/output batches. That is, the Dropbox synchronizes local replica, and the server of the Drop box architecture watches for changes, instead of sending all input/output over the network. As shown in FIG. 1, ***the Dropbox architecture includes a local copy of the files in a local disk of the client personal computer.*** The Drop box client synchs remote and local changes from the local copy of the files with the Dropbox servers.

(EX2026, ¶ 11.)[2]

POPR at 18

PO Correctly Identified the Houston '787 Provisional

on which computer." (EX1003, 8:14-21.) Through the incorporated *Houston* II, *Houston* goes into even greater detail regarding the benefits of its system using local files at each client:

- "Various embodiments of the innovative technology presented herein eliminates the need to have a person maintaining a central server for a business. The technology makes local copies of data which are then synchronized using delta-synch protocol . . . The technology allows for a superior user experience, since multiple users may be offline or online and still use this technology. Also, the technology allows for edits of files in a shared folder to be quickly seen on the desktops of some or all the users who have access to the shared folder. The technology does not require a separate computer to be handling the sharing of folders. Instead, the technology makes use of a computing cloud, and also makes copies of data instead of storing them all in one area. In other words, the technology provides all the benefits of a traditional system for sharing files, without the hassle of a traditional infrastructure having a centralized server." (EX2026, ¶9)

POPR at 19

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

PO Correctly Identified the Houston '787 Provisional

- "Also, various embodiments of the technology provides local input/output speed (which equates to less speed time in comparison to traditional systems) because the files are local, and thus listing and accessing of the files appear much more quickly to users than if they had used traditional systems. Through this technology, the user may not be required to view the transfer of files that occur in the background. Further, users may work online or offline on a file in the shared folder, and there is no need to stream again the entire edited version of a file when changes may be made to the same file. Additionally, the technology allows for some or all edited versions of a file to be stored, such that if a user is working on version 5 of a file and then wishes to restore version 3, version 3 may be restored or "undeleted" by the user, while also storing the version 5 of the edited file as well. Also, if two users are working on a same file, the users may open multiple versions (such as versions 3 and 5) of the file, and both users' edits of versions 3 and 5 of the file will be stored. Thus, built-in reversion control is provided. The technology may also be used across a variety of networks." (*id.*, ¶10.)

POPR at 19-20

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

PO Correctly Identified the Houston '787 Provisional

- "Also, most applications of a traditional file server architecture are optimized for local storage, which means there may be a significant experience degradation when the file server is remote. Such degradation means that there may be hundreds to thousands of millisecond latency. There may be hidden input/output bottlenecks which may occur when loading data into a directory listing. Further, data streaming speed is oftentimes limited to local throughput which is much slower than local storage, leading to long load times for large files.

  In stark contrast, according to various embodiments, the applications in the Dropbox architecture use local storage with low latency and high throughput. Since all input/output is local, access time is approximately 10 milliseconds." (*id.*, ¶¶12-13.)

POPR at 19-20

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

PO Correctly Identified the Houston '787 Provisional

- "Also, in comparing traditional architecture with the Dropbox architecture, traditional file server architecture does not allow for offline access, and VPN and/or complex firewall rules may be needed for remote access. Also, caching, file locks, and the like at the network file system layer may create complexity and coherency issues. Unlike traditional file server architecture, according to various embodiments, the Dropbox architecture allows for offline access, since local metadata cache ensures consistency. Also, changes are synced in batches utilizing the Dropbox architecture, which increases efficiency and makes the system resilient to latency and failure." (*id.*, ¶14.)

POPR at 20

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Houston '787 Provisional Is Part of Houston

application. Petitioner's attempt to carve out one of the provisional applications that is incorporated by reference from the disclosure of Houston is contrary to Federal Circuit precedent: the incorporated provisional applications are "part and parcel" of Houston and all must be considered together as a whole.[4] A party is not allowed to pick and choose portions of a reference while excluding others for purposes of obviousness.[5] Excluding reliance on the Houston '787 provisional would preclude reliance on the remainder of Houston under these circumstances.[6]

[4] *Netflix, Inc. v. Avago Techs. Int'l*, Nos. 2024 U.S. App. LEXIS 17998, at *21 (Fed. Cir. July 23, 2024) ("By clearly identifying Appendix J, detailing the subject matter it addresses, and pointing to where it is found, Masters makes clear that Appendix J is part and parcel to Masters itself. We therefore conclude that the Board erred when it declined to consider Appendix J for purposes of its analysis."); *see also Finjan LLC v. ESET, LLC*, 51 F.4th 1377, 1382 (Fed. Cir. 2022); *Trs. of Columbia Univ. v. Symantec Corp.*, 811F.3d 1359, 1365-66 (Fed. Cir. 2016).

Paper 41 at 2

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Houston '787 Provisional Is Part of Houston

application. Petitioner's attempt to carve out one of the provisional applications that is incorporated by reference from the disclosure of Houston is contrary to Federal Circuit precedent: the incorporated provisional applications are "part and parcel" of Houston and all must be considered together as a whole.[4] A party is not allowed to pick and choose portions of a reference while excluding others for purposes of obviousness.[5] Excluding reliance on the Houston '787 provisional would preclude reliance on the remainder of Houston under these circumstances.[6]

[5] *Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve, Inc.*, 796 F.2d 443, 448-49 (Fed. Cir. 1986); *Recor Med. Inc. v. Medtronic Ireland Mfg. Unlimited Ltd.*, No. 23-2251 slip op.at 5 (Fed Cir. Mar. 27, 2025) ("A reference must be considered for everything it teaches by way of technology and is not limited to the particular invention it is describing and attempting to protect . . . We agree with Recor that the Board violated this principle here.") (cleaned up).

Paper 41 at 2

The Houston '787 Provisional Is Part of Houston

application. Petitioner's attempt to carve out one of the provisional applications that is incorporated by reference from the disclosure of Houston is contrary to Federal Circuit precedent: the incorporated provisional applications are "part and parcel" of Houston and all must be considered together as a whole.[4] A party is not allowed to pick and choose portions of a reference while excluding others for purposes of obviousness.[5] Excluding reliance on the Houston '787 provisional would preclude reliance on the remainder of Houston under these circumstances.[6]

[6] That is consistent with 35 U.S.C. § 312, which requires Petitioners provide Patent Owners with copies of the patents and printed publications relied upon in a Petition as a threshold condition for filing. *See AMC Multi-Cinema, Inc. v. Fall Line Patents, LLC*, No. 2021-1051, 2021 U.S. App. LEXIS 29547, at *15 (Fed. Cir. Sep. 30, 2021).

Paper 41 at 2

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

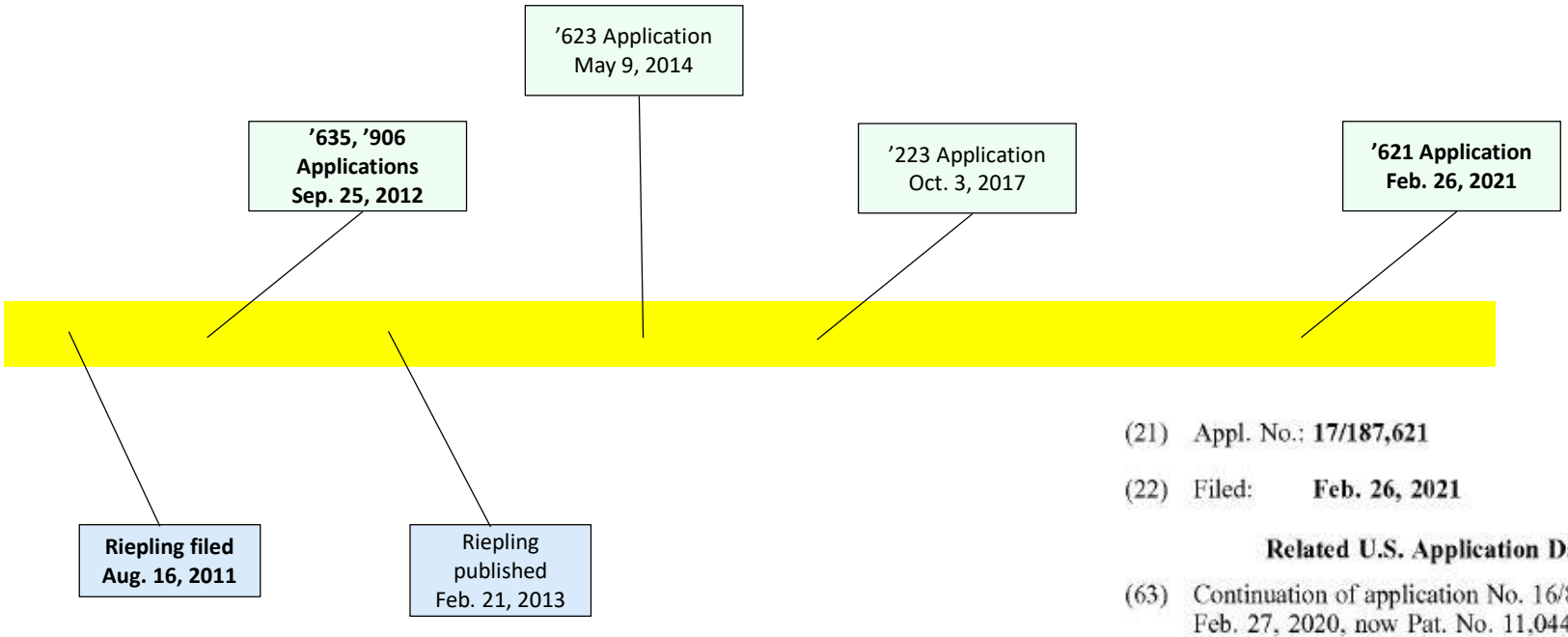
IPR2024-00287

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Is Not a Prior Art Printed Publication

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling is Not a Prior Art Printed Publication



(21) Appl. No.: **17/187,621**

(22) Filed: **Feb. 26, 2021**

Related U.S. Application Data

(63) Continuation of application No. 16/803,980, filed on Feb. 27, 2020, now Pat. No. 11,044,215, which is a continuation of application No. 15/984,406, filed on May 20, 2018, now Pat. No. 10,587,548, which is a continuation of application No. 15/724,223, filed on Oct. 3, 2017, now Pat. No. 10,021,052, which is a continuation-in-part of application No. 14/274,623, filed on May 9, 2014, now abandoned, which is a continuation-in-part of application No. 13/626,635, filed on Sep. 25, 2012, now abandoned, and a continuation-in-part of application No. 13/624,906, filed on Sep. 22, 2012, now abandoned.

(POR at 30-31; EX1101)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Is Not a Prior Art Printed Publication

(54) **METHOD AND SYSTEM FOR SENDING A DIGITAL INVITATION REQUESTING A DATA UPLOAD**

(75) Inventors: **Eric Casten Riepling**, Soquel, CA (US); **Reza Brian Vaziri**, Corona, CA (US)

(73) Assignee: **SMITH MICRO SOFTWARE, INC.**, ALISO VIEJO (CA)

(21) Appl. No.: **13/211,146**

(22) Filed: **Aug. 16, 2011**

1. Under § 311(b), Abandoned Patent Applications—Unlike Patents—Are Available as Prior Art Only as of Their Publication Date[2]

Section 311(b) provides:

> A petitioner in an inter partes review may request to cancel as unpatentable 1 or more claims of a patent only on a ground that could be raised under section 102 or 103 and only on the basis of prior art consisting of patents or printed publications.

35 U.S.C. § 311(b).

(POR at 31-35; EX1107)

Riepling Is Not a Prior Art Printed Publication

(54) **METHOD AND SYSTEM FOR SENDING A DIGITAL INVITATION REQUESTING A DATA UPLOAD**

(75) Inventors: **Eric Casten Riepling**, Soquel, CA (US); **Reza Brian Vaziri**, Corona, CA (US)

(73) Assignee: **SMITH MICRO SOFTWARE, INC.**, ALISO VIEJO (CA)

(21) Appl. No.: **13/211,146**

(22) Filed: **Aug. 16, 2011**

In sum, § 311(b) distinguishes "patents" from "printed publications," but does not list "patent applications" at all. So, unlike for patents—which a petitioner can use as prior art under § 311(b) as of the patent's earliest priority date, regardless of public accessibility—there is no reason to treat patent applications differently than any other printed publication under § 311(b). Either a patent application becomes a patent and is treated as a "patent[]" under § 311(b), or it does not and is treated like any other printed publication under § 311(b). Congress could have treated patent applications differently, and it did in many other patent statutes, but not in § 311(b).

(POR at 31-34; EX1107)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Is Not a Prior Art Printed Publication

(10) **Pub. No.: US 2013/0046833 A1**
(43) **Pub. Date: Feb. 21, 2013**

(54) **METHOD AND SYSTEM FOR SENDING A DIGITAL INVITATION REQUESTING A DATA UPLOAD**

(75) Inventors: **Eric Casten Riepling**, Soquel, CA (US); **Reza Brian Vaziri**, Corona, CA (US)

(73) Assignee: **SMITH MICRO SOFTWARE, INC.**, ALISO VIEJO (CA)

(21) Appl. No.: **13/211,146**

(22) Filed: **Aug. 16, 2011**

3. *Riepling* Is Not a Prior Art "Printed Publication" Available Under § 311(b) Because It Was Not Publicly Accessible Before the Challenged Claims' Priority Date

. . .

Here, the PTO did not publish *Riepling* until 2013, after the challenged claims' priority date in 2012. On and before the challenged claims' priority date, the public could not access *Riepling* because patent applications are confidential until they are published. *See* 35 U.S.C. § 122(a). Therefore, *Riepling* is not a prior art "printed publication" under § 311(b).

(POR at 34-36; EX1107)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Is Not a Prior Art Printed Publication

B. *Riepling* Does Not Qualify As a Prior Art Patent Application Under 35 U.S.C. § 311(b), So All Grounds Fail

PO acknowledges the decision in *Lynk Labs Inc. v. Samsung Electronics Co.*, *No.* 2023-2346, 2025 WL 85559, at *8 (Fed. Cir. Jan. 14, 2025) and reiterates its arguments (POR, 31-36) to preserve the issue for appeal, understanding that the Board is bound by the Federal Circuit's decision. To properly preserve the issue, PO respectfully requests that the Board render a decision on the issue of the claims' entitlement to the September 25, 2012 filing date of the '635 application. PO believes its position is warranted by its nonfrivolous arguments for reversal of existing law and notes that the issue is still pending in *VLSI Tech. LLC v. Patent Quality Assurance LLC*, No. 23-2298 (Fed. Cir.) although the Federal Circuit may need to issue an *en banc* decision given the prior panel decision.

(PO Sur-reply at 9)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The Claims Are Entitled to at Least a September 25, 2012 Priority Date

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Has Not Shown that PO Is Precluded

First, PO is collaterally estopped from arguing the '520 patent is entitled to a September 25, 2012 priority date. *SimpleAir, Inc. v. Google LLC*, 884 F.3d 1160, 1165 (Fed. Cir. 2018) ("Under the doctrine of claim preclusion, a judgment on the merits in a prior suit involving the same parties or their privies bars a second suit based on the same cause of action."). PO's summary of MO-DB-1 ***omits*** that although the District Court denied Petitioner's motion for summary judgment, the Court's final judgment found that PO was not entitled to the September 25, 2012 priority date for the related patents litigated in that case. EX1129 at 2. The '520

(Rep. at 6-7)

Reply at

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Preclusion Is Irrelevant and Inadequately Presented

1. Preclusion Does Not Apply

First, Petitioner is wrong that claim preclusion[4] bars PO from arguing for a September 25, 2012 priority date. (Rep. at 6.) The parties have submitted post-trial briefing and no appeal has been taken yet unlike in *SimpleAir* where the patent owner had appealed and lost the prior litigation after the Federal Circuit remanded with instructions to enter judgment. 884 F.3d at 1163. While Petitioner does not address the factors for either test, issue preclusion does not apply here either.

(Sur Reply at 3)

Because Petitioner has failed to analyze or even address the specific factors under either claim or issue preclusion, it has failed to demonstrate that either apply.

(Sur-Reply at 4)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

"Claim Preclusion" Is Irrelevant

The test for claim preclusion in the Fifth Circuit has four elements:
(1) the parties in the later action are identical to, or in privity with, the parties in the earlier action;
(2) the judgment in the earlier case was rendered by a court with proper jurisdiction;
(3) there has been a final judgment on the merits; and
(4) the earlier case and later case involve the same cause of action.

SimpleAir, Inc. v. Google LLC, 884 F.3d 1160, 1165 (Fed. Cir. 2018)

(Rep. at 6-7; PO Sur-reply at 3)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Issue Preclusion Is Irrelevant

Dropbox presented invalidity arguments that involved references prior to September 25, 2012, and were therefore independent of the challenge to the priority dates of the asserted claims in MO-DB-1. As such, the jury's priority date determination might not even be essential to any potential factual explanation for judgment (which would be improper to include anyway in a Rule 58 judgment). The jury's factual answers on priority date have no legally viable place in a final judgment and are therefore not "essential to a final judgment in the first action," as required under this factor. Moreover, Dr. Mowry's comparison of the specifications is irrelevant and ignores the specific claim limitations at issue both here and in the trial. (EX1127, ¶¶23-32.)

(PO Sur-reply at 3-4)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Relevant Disclosures in Priority Applications

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Argument Turns on Purported "External Messaging Node"

A. Generation of First Message

33. The challenged claims all include limitations Patent Owner contends are performed by external messaging node functionality that generates, based information received from a first node, a first message or email to send to a second node.

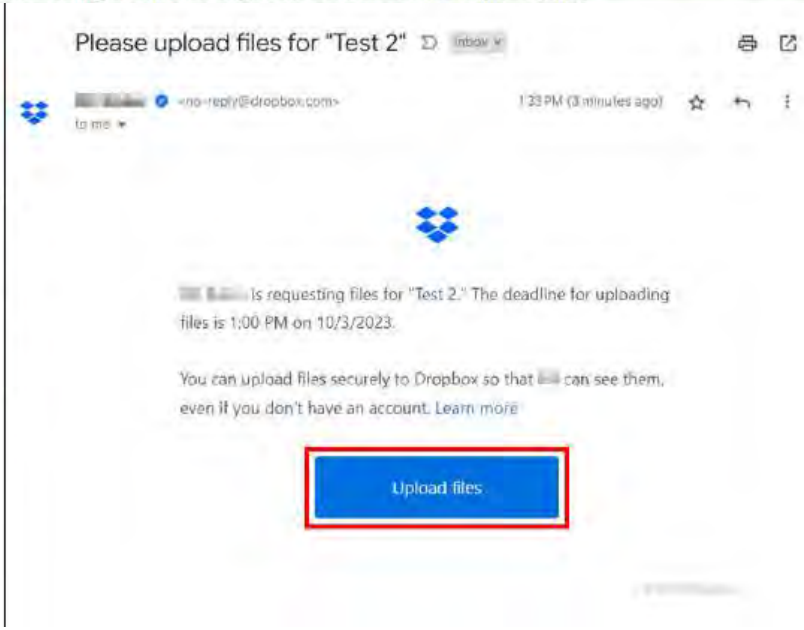
B. Generation of Second Message

68. The challenged claims all include limitations Patent Owner contends are performed by external messaging node functionality that generates a second message or email to send to the first node.

(EX1127 ¶¶ 33, 68)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Argument Turns on Purported "External Messaging Node"

in response to the receipt of the indication of the selection of the fourth user interface element and based on the file description, the indication of the at least one of the	The Accused Instrumentalities are designed to, *in response to the receipt of the indication of the selection of the fourth user interface element and based on the file description, the indication of the at least one of the date or the time, and the indicia associated with the at least one email address; generate a first message addressed to the at least one email address and including the file request, the first message indicating the at least one of the date or the time on which the file request is based, and further including a fifth user interface element and at least one sentence including the file description describing the at least one file in connection with the file request;*
date or the time, and the indicia associated with the at least one email address; generate a first message addressed to the at least one email address and including the file request, the first message indicating the at least one of the date or the time on which the file request is based, and further including a fifth user interface element and at least one sentence including the file description describing the at least one file in connection with the file request;	

(EX1127 ¶ 34, citing EX1132 at 71-72)

Petitioner's Expert Downplays Key Disclosures in Priority Applications

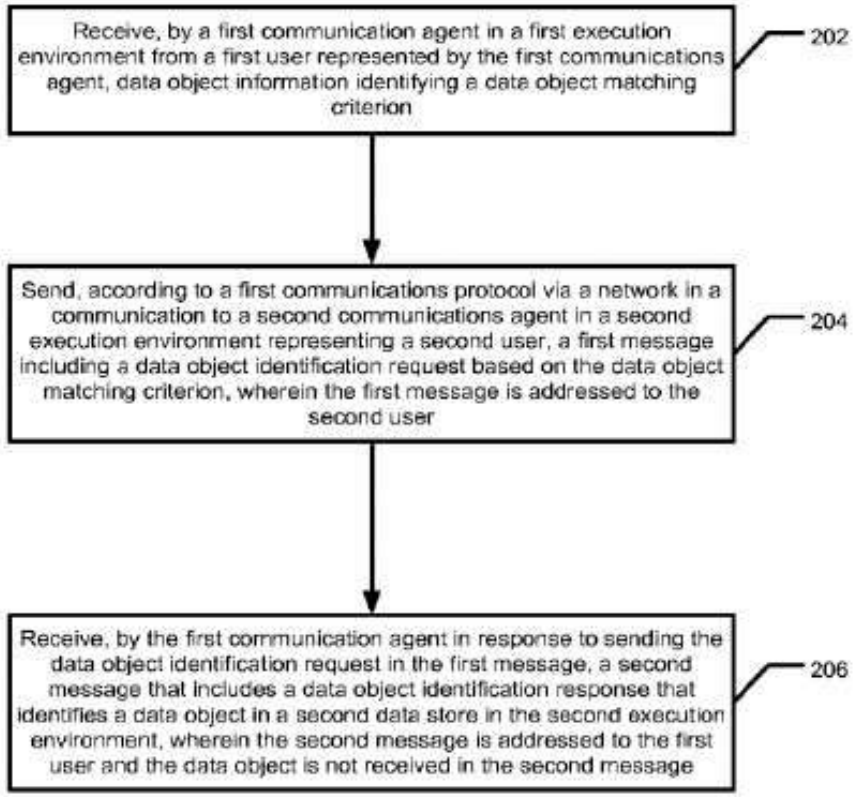
49. To the extent Dr. Smith is intending to suggest the "web service provider node" in paragraph 124 provides written description support for the generate/generation of a first message by an external messaging node, which Patent Owner appears to contend is encompassed in limitations of all challenged claims, I disagree. Paragraph 124 is describing that a user can perform a search by providing data object information to "a web service provider node" with a "query handler component." Exhibit 2004, ¶ 124 ("For example, a user may identify data object information to a browser operating in a client node. The browser may send the data object information via a network communicatively coupling the client node to a web service provider node. The web service provider node may include a query handler component that may operate to receive the data object information identified in the message from the browser."). The web service provider node and the query handler component do not generate messages.

(EX1127 ¶ 49)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures of External Messaging Nodes in Priority Applications

25. The '906 application describes techniques for causing the sending of "a first message including a data object identification request … wherein the first message is addressed to the second user." ('906 application, EX2004, at [0127]). In response to such first message, the '906 application discloses "a second message that includes a data object identification response that identifies a data object …, wherein the second message is addressed to the first user and the data object is not received in the second message." ('906 application, EX2004, at [0137] and Fig. 2A, reproduced below.)



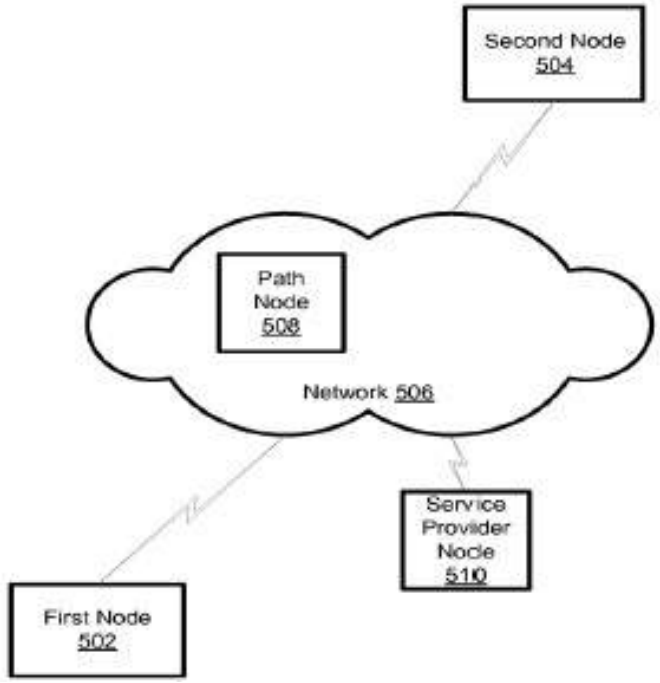
(EX2005 ¶ 25; EX2004 Fig. 2A; EX2005 App'x 1, at 34-37, 60-69; POR at 3; Sur-Reply at)

Petitioner Overlooks Key Disclosures in Priority Applications

26. This apparatus comprises at least one non-transitory memory storing instructions and at least one processor that can execute these instructions. ('906 application, EX2004, at [0035], [0038], [0052]–[0053], [0209]–[0210].) In embodiments, the software is stored in remote storage accessible via a network. ('906 application, EX2004 at [0039], [0050] , [0102] , [0124], [0162] , [0171].).

27. In some embodiments, a web services provider node may interoperate with users' client nodes, for example, through users' browsers. ('906 application, EX2004 at [0123]–[0124]) This is generally shown in Figure 5 of the '906 application, reproduced below.

28. When the apparatus such as the web services node executes these instructions, it causes, at a first node, the display of an interface for use in creating a file request for at least one file. ('906 application, EX2004, at [0100], [0111], [0115]–[0117].) An example of this user interface is included in FIG. 6A of the '906 application, copied below:



(EX2005 ¶¶ 26-28; EX2004 Fig. 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

26. This apparatus comprises at least one non-transitory memory storing instructions and at least one processor that can execute these instructions. ('906 application, EX2004, at [0035], [0038], [0052]–[0053], [0209]–[0210].) In embodiments, the software is stored in remote storage accessible via a network. ('906 application, EX2004 at [0039], [0050] , [0102] , [0124], [0162] , [0171].).

27. In some embodiments, a web services provider node may interoperate with users' client nodes, for example, through users' browsers. ('906 application, EX2004 at [0123]–[0124]) This is generally shown in Figure 5 of the '906 application, reproduced below.

28. When the apparatus such as the web services node executes these instructions, it causes, at a first node, the display of an interface for use in creating a file request for at least one file. ('906 application, EX2004, at [0100], [0111], [0115]–[0117].) An example of this user interface is included in FIG. 6A of the '906 application, copied below:

[0102] As stated, the various adaptations of the arrangement in FIG. **3A** as well as the various adaptations of the arrangement in FIG. **3B** illustrated and described herein are not exhaustive. For example, those skilled in the art will see based on the description herein that arrangements of components for performing the method illustrated in FIG. **2A** and the method illustrated in FIG. **2B** may each be distributed across more than one node and/or execution environment. For example, such an arrangements may operate at least partially in a browser in a one node and at least partially in a server in another node interoperating via a network.

[0123] FIG. **7** includes dataflow message **702** that illustrates a data flow internal to execution environment **401** of first node **502** and/or a message received via network **506** by execution environment **401** of first node **502**. Dataflow message **702** illustrates a data object information dataflow to query handler component **402** operating in execution environment **401** of first node **502**. Data object information is identified to and/or otherwise received by query handler component **402**.

[0124] In various aspects, a query handler component **402** may receive data object information from a user via one or more UI element handler components **415**. The data object information may be received, in response to a user input detected by an input device of an execution environment **401** that includes the query handler component **402**. Alternatively or additionally, data object information may be received by a query handler component in a message received via a network. For example, a user may identify data object information to a browser operating in a client node. The browser may send the data object information via a network communicatively coupling the client node to a web service provider node. The web service provider node may include a query handler component that may operate to receive the data object information identified in the message from the browser.

(EX2005 ¶¶ 26-28; EX2004 at [0102], [0123]-[0124])

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

But even assuming, *arguendo*, this "external messaging node" theory, Dr. Smith identified numerous disclosures from the Original Applications showing such functionality. Specifically, the Original Applications describe the folder sharing and file request systems of the '520 patent implemented over the web. (*E.g.*, EX2002, ¶¶5, 69, 102-116, 130, 143, 373, Fig.6A; EX2004, ¶¶101-102, 122-24, 141-144, 180-204; Fig.5.) Dr. Smith provided detailed testimony that the '520 patent, going back to the Original Applications, provides §112 support for claims 9-12, 14-16 and 22-24.[5] (EX2005, ¶¶24-38, 83-85, Appendix1.) This includes embodiments using a "web service provider node," (EX2004, ¶124), interoperating with first and second nodes and functioning as the purported external messaging node that Petitioner and Dr. Mowry allege is lacking:

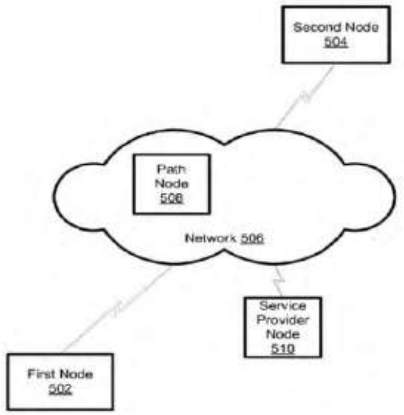


FIG. 5

(EX2005, ¶27 (citing EX2004, Fig. 5; EX2004, ¶¶123-124); EX1101, Fig. 5.) This includes file requests according to the methods of FIG. 2A and FIG.2B and the dataflow in Figure 7. (EX2004, ¶¶14-15, 27, 101-102, 122-124; *see also* ¶185.) Fig. 7 and Fig. 2A are reproduced below:

(PO Sur-reply at 5-6)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

Fig. 7 discusses the dataflow for file requests and response. (EX2004, ¶¶123, 128, 134, 138, 144, 149, 154, 158-160, 162, 171, 177 (noting messages 702, 704, 716, 718 may include interoperation with another node, 710 may be a message sent to a remote device, and messages 708 and 712 may include interoperation with another node).) Fig. 2A describes steps associated with embodiments of the file requests. (EX2004, Fig. 2A, *see also* Fig.2B; EX2005, ¶25.) The Original Applications specifically disclose that the data object information may be received from a browser by a web server and that "a data object identification request," "a data object identification response, an access request, and a response to an access request" "may be generated, detected and/or otherwise processed according to a schema," (EX2004, ¶186), which Dr. Mowry stated included a date, (EX2015, 134:22-135:7), that is identified by a node on a network in communication with communications agents hosted on one more network nodes. (EX2004, ¶¶185-186, *see also id.* ¶¶101-102, 204.) A representation of the requested data object may be presented to the

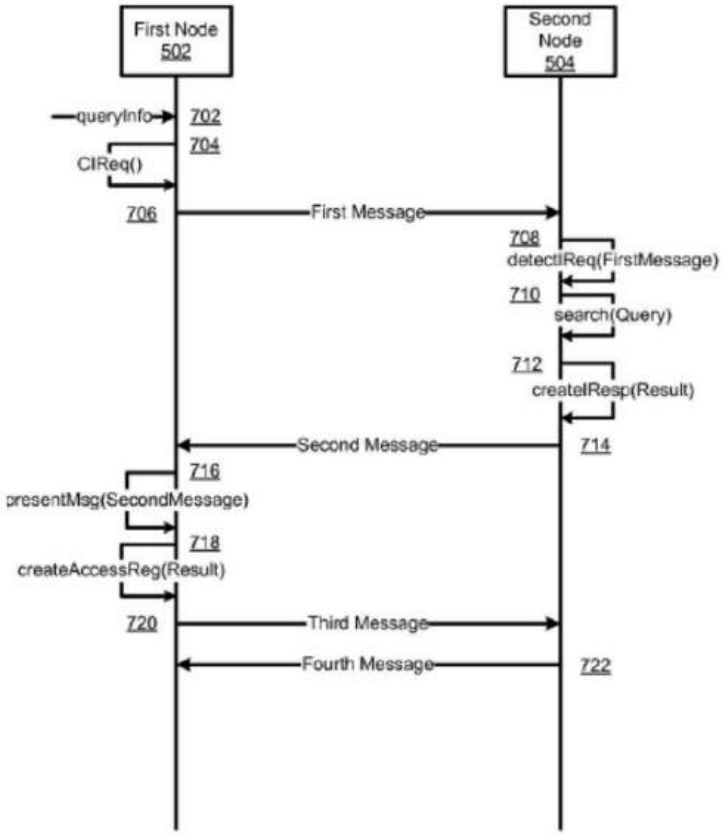


FIG. 7

(PO Sur-reply at 7-8)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

Fig. 7 discusses the dataflow for file requests and response. (EX2004, ¶¶123, 128, 134, 138, 144, 149, 154, 158-160, 162, 171, 177 (noting messages 702, 704, 716, 718 may include interoperation with another node, 710 may be a message sent to a remote device, and messages 708 and 712 may include interoperation with another node).) Fig. 2A describes steps associated with embodiments of the file requests. (EX2004, Fig. 2A, *see also* Fig.2B; EX2005, ¶25.) The Original Applications specifically disclose that the data object information may be received from a browser by a web server and that "a data object identification request," "a data object identification response, an access request, and a response to an access request" "may be generated, detected and/or otherwise processed according to a schema," (EX2004, ¶186), which Dr. Mowry stated included a date, (EX2015, 134:22-135:7), that is identified by a node on a network in communication with communications agents hosted on one more network nodes. (EX2004, ¶¶185-186, *see also id.* ¶¶101-102, 204.) A representation of the requested data object may be presented to the

[0185] As described above, receiving data object information may include receiving, via network from a node that received the data object information from the user, a message that identifies the data object information. For example, data object information may be received from a browser by a web server.

[0186] Data object information and/or a data object identification request may be generated, detected, and/or otherwise processed according to a schema that identifies at least one of a rule and a vocabulary that defines a valid data object identification request. Similarly, a data object identification response, an access request, and a response to an access request may be generated, detected, and/or otherwise processed according to a schema. One or more schemas may alone or together define one or more of valid data object information, a data object identification request, data object identification response, an access request, and/or an access response. A schema, as just described, may be identified based on data received and/or otherwise identified by a node hosting a communications agent via the network from another node hosting a communications agent and/or from a path node included in a network path in the network communicatively coupling the node to one or more nodes that respectively host communications agents.

(PO Sur-reply at 7-8; EX2004, [0185]-[0186])

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

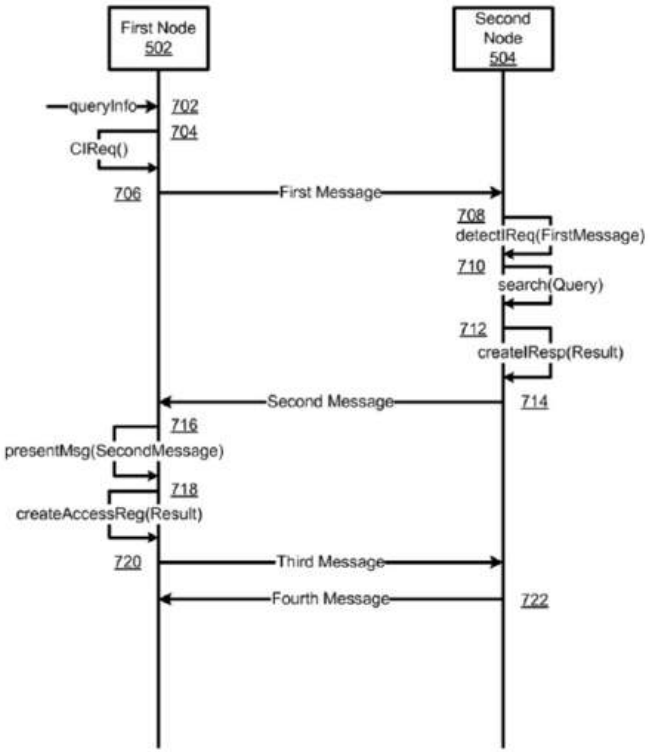


FIG. 7

[0123] ("FIG. 7 includes dataflow message 702 that illustrates a data flow internal to execution environment 401 of first node 502 and/or a message received via network 506 by execution environment 401 of first node 502. Dataflow message 702 illustrates a data object information dataflow to query handler component 402 operating in execution environment 401 of first node 502. Data object information is identified to and/or otherwise received by query handler component 402.")

[0124] ("In various aspects, a query handler component 402 may receive data object information from a user via one or more UI element handler components 415. The data object information may be received, in response to a user input detected by an input device of an execution environment 401 that includes the query handler component 402. Alternatively or additionally, data object information may be received by a query handler component in a message received via a network. For example, a user may identify data object information to a browser operating in a client node. The browser may send the data object information via a network communicatively coupling the client node to a web service provider node. The web service provider node may include a query handler component that may operate to receive the data object information identified in the message from the browser.")

(EX2005 at pp. 25-26 of Appendix 1; Sur-Reply at 6-8)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

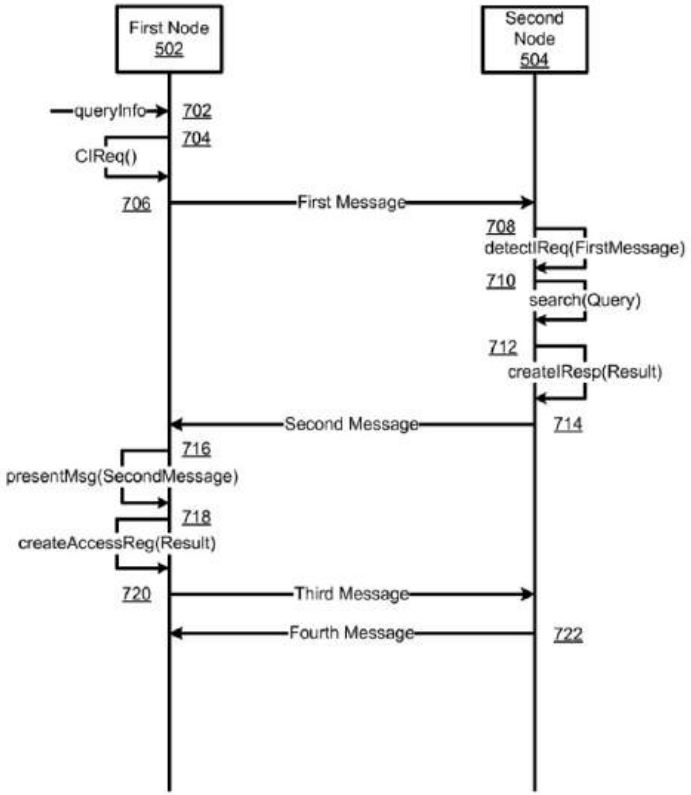


FIG. 7

[0128] FIG. **7** includes dataflow message **704** that illustrates a data flow internal to execution environment **401** of first node **502** and/or message received via network **506** by execution environment **401** of first node **502**. Dataflow message **704** illustrates a create data object identification request (CIReq) dataflow to an IRC component **404** operating in execution environment **401** of first node **502**. Data object information may be identified to and/or otherwise provided to IRC component **404**. A query handler component **402**, operating as described with respect to block **202** of FIG. **2A**, may identify and/or otherwise provide data object information to an IRC component **404**.

[0129] A query handler component may be a type of content handler component that operates to process data object information to create a data object identification request that conforms to a schema for creating and/or otherwise constructing a valid data object identification request. The IRC component **404** may construct and/or otherwise create a data object identification request based on a schema that defines and/or otherwise identifies a valid data object identification request and/or a valid data object identification response for a particular type of communication supported by a communications agent. The data object identification response is constructed and/or otherwise created, based on the data object information, by the IRC component **404**. IRC component **404** in execution environment **401** of first node **502** may provide the data object identification request to content manager component **409** to include and/or otherwise identify the data object identification request in a communication with execution environment **401** of second node **504**, such as email and/or instant message. Content manager component **409** may interoperate with other types of content handler components **413** to create and/or otherwise construct a message that includes the valid data object identification request.

(Sur-Reply at 6-8; EX2004 at [0128]-[0129], Figure 7; Exhibit 2005, Appendix 1 at 36, 132, 215)

Petitioner Overlooks Key Disclosures in Priority Applications

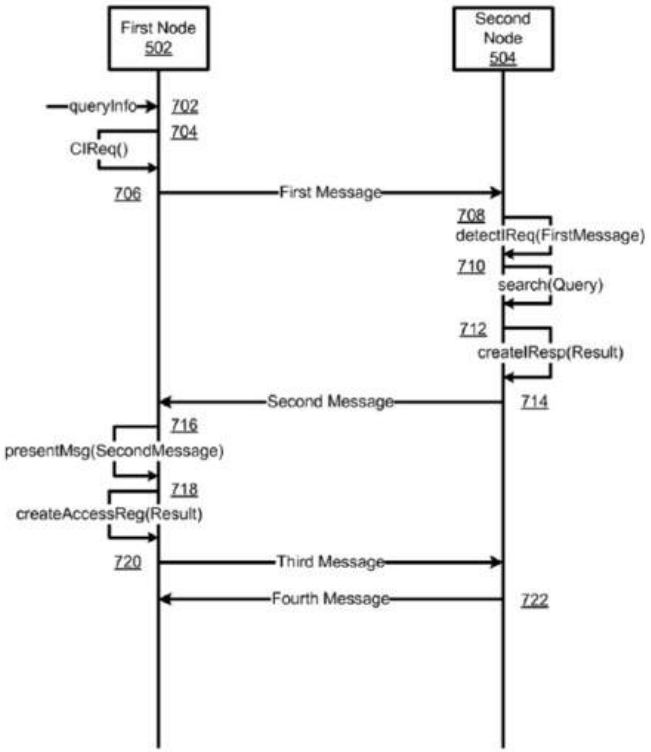


FIG. 7

[0144] FIG. **7** includes dataflow message **716** that illustrates a present message that may be internal to execution environment **401** of first node **502** and/or may be a message sent to a remote device, such as a node including a browser, to send presentation information to present, via an output device, a representation of the data object identification response received in second message **714**. FIG. **7** also includes dataflow message **718** that illustrates a create access request message that may be a data flow internal to execution environment **401** of first node **502** and/or a message received via network **506** by execution environment **401** of first node **502**. In the dataflow message **718** an ARC component **408** operating in execution environment **401** of first node **502** may be invoked to operate based on result information. Result information may be identified to and/or otherwise by ARC component **408** as part of and/or in response to being invoked.

((Sur-Reply at 6-8; EX2004, Fig. 7, [0144]))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

[3] Dr. Smith also cites, but does not excerpt, paragraphs 129-130, which provide additional details about the "query handler component" and the IRC component discussed in paragraph 128 and also discuss the "content manager 409." Exhibit 2005, Appendix 1 at 37; *see also id.* at 133, 216. I disagree these paragraphs provide written description support for the generate/generation of a first message by an external messaging node, which Patent Owner appears to contend is encompassed in limitations of all challenged claims. As shown in Figure 4, the query handler, IRC component and content manager are all within the execution environment 401. Exhibit 2004, Figure 4. And, paragraph 128 discusses "execution environment 401 of first node 502."

(EX1127 at paragraph 46)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Overlooks Key Disclosures in Priority Applications

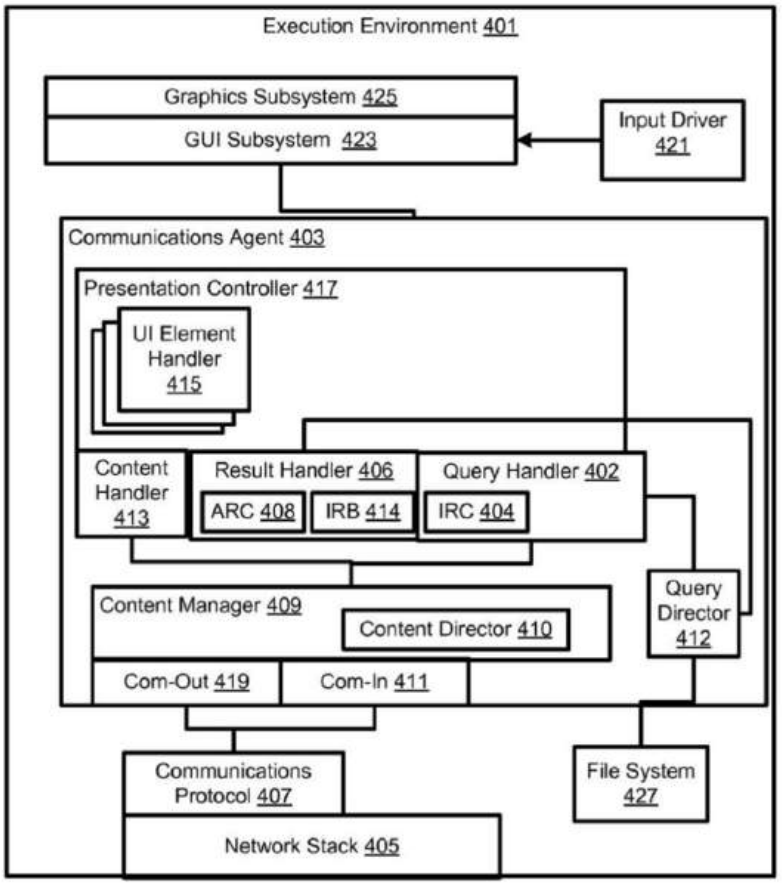


FIG. 4

[0128] FIG. **7** includes dataflow message **704** that illustrates a data flow internal to execution environment **401** of first node **502** and/or message received via network **506** by execution environment **401** of first node **502**. Dataflow message **704** illustrates a create data object identification request (CIReq) dataflow to an IRC component **404** operating in execution environment **401** of first node **502**. Data object information may be identified to and/or otherwise provided to IRC component **404**. A query handler component **402**, operating as described with respect to block **202** of FIG. **2A**, may identify and/or otherwise provide data object information to an IRC component **404**.

[0129] A query handler component may be a type of content handler component that operates to process data object information to create a data object identification request that conforms to a schema for creating and/or otherwise constructing a valid data object identification request. The IRC component **404** may construct and/or otherwise create a data object identification request based on a schema that defines and/or otherwise identifies a valid data object identification request and/or a valid data object identification response for a particular type of communication supported by a communications agent. The data object identification response is constructed and/or otherwise created, based on the data object information, by the IRC component **404**. IRC component **404** in execution environment **401** of first node **502** may provide the data object identification request to content manager component **409** to include and/or otherwise identify the data object identification request in a communication with execution environment **401** of second node **504**, such as email and/or instant message. Content manager component **409** may interoperate with other types of content handler components **413** to create and/or otherwise construct a message that includes the valid data object identification request.

(Sur-Reply at 6-8; Exhibit 2005, Appendix 1 at 40 (citing Fig. 4 and [0128]-[0129]; *see also* POR at 30-31)

Petitioner's Expert Downplays Key Disclosures in Priority Applications

Q. A "web service provider node." What does that mean to a person of skill in the art?
Strike that.
What would a person of skill in the art understand a web service provider node to be?
A. That's too vague of a term for someone to know what that means without additional context.

(EX2015 99:15-23)

Q. Sure. What does the phrase "web service provider node" --
Strike that.
What do you understand the reference to "node" to be in Figure 5 --
Strike that.
So what do you understand a "web service provider node" to refer to?
A. Well, in that column, let me go back to this. I mean, a web service is something that you access usually through a browser. And so in column 27, that's the only discussion I see of "web service provider node," and it's something that the browser is interacting with to view information in the form of a web page.

(EX2015 101:9-23)

(Sur-Reply at 5, 8; EX1127 ¶ 49)

Petitioner's Expert Downplays Key Disclosures in Priority Applications

Q. And the question is -- with all of that lead-up -- what would a person of skill in the art understand a "web service provider node" to be referring to in the 2012 time frame?

A. Well, in the '520 Patent in column 27, the -- starting at around -- so column 27, starting at around line 7, it says, (As read): The browser may send the data object information via a network to a web -- and then there's some more text -- to a web service provider. The web service provider node may include a query handler component to receive the data object information.

So to paraphrase this, it is responding to a request from the web browser, and as I discuss in paragraph 49 of Exhibit 7, I have a whole paragraph where I talk about this.

And what I say is that, for example, on the last sentence of paragraph 49, the web service provider node in the query handler component, these things do not generate messages.

(EX2015 105:8-106:5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Expert Downplays Key Disclosures in Priority Applications

Q. I'm just asking if you have -- what do you understand that to be depicting? And it can be in the context of the '520 Patent, as well.

I think you said there wasn't much disclosure in the '520 Patent, and so I would presume that you would interpret it according to its plain and ordinary meaning under the circumstances.

A. Just from the figure, the only clear thing is that this thing is connected to the network. That's really the only thing that's obvious from the figure.

But to understand, really, what service provider or node 510 is, we need to look to the spec, because those are fairly generic terms. "Service provider" and "node" in column 27, where we were just -- that we were referencing a minute ago, it talks about a "web service provider node." So assuming that's what -- assuming that corresponds to what we are looking at here in the figure, it does discuss that it is something that the web browser interacts with.

(EX2015 109:12-110:9))

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Expert Downplays Key Disclosures in Priority Applications

As stated, the various adaptations of the arrangement in FIG. **3**A as well as the various adaptations of the arrangement in FIG. **3**B illustrated and described herein are not exhaustive. For example, those skilled in the art will see based on the description herein that arrangements of components for performing the method illustrated in FIG. **2**A and the method illustrated in FIG. **2**B may each be distributed across more than one node and/or execution environment. For example, such an arrangements may operate at least partially in a browser in a one node and at least partially in a server in another node interoperating via a network.

FIG. **5** illustrates a first node **502** and an execution environment **401** of second node **504** as exemplary devices that each may be included in and/or otherwise adapted for providing an instance, adaptation, and/or analog of execution environment **401** in FIG. **4**. As illustrated in FIG. **5**, first node **502** and an execution environment **401** of second node **504** are operatively coupled to a network **506** via respective network interface components enabling first node **502** and execution environment **401** of second node **504** to exchange data in a communication on behalf of communicants represented by the respective nodes.

Q. So, I don't -- I'm sorry. I apologize.

I just really want you to answer the question, because I am asking what the spec disclosed, and this is written description report.

So I want to know what your understanding is of that paragraph. Is it agnostic -- or maybe that's too strong a word, but isn't it disclosing that you can use various arrangements of these nodes, including in a network, with kind of like a client server arrangement as part of the invention?

MR. ESRAM: Objection to form.

A. It says just exactly what it says here. It's the sentence is relatively narrow.

It's talking about the method in Figure 2A and 2B regarding -- your question seems to suggest that there's lots and lots of flexibility with regard to the arrangement of various things.

But you have to start with the claim language and what it requires, and these sentences are fairly specific, and they just say what they say.

(EX1101 22:21-42; EX2015 114:6-118:15)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Expert Downplays Key Disclosures in Priority Applications

Q. The third, this external node that you seem to say is always required.

So I want to -- I tried to understand, and I am not getting it.

So -- you said we need to focus on the claim language. We are looking at the claim. We have first and second node. Got it. You state in your dec that there has to be a third, a third node.

Okay. Well, we have first, second node in column 22. We have a server. We have the web service provider node in Figure 5.

What's missing?

MR. ESRAM: Objection to form.

BY MR. DAHLGREN:

Q. What's missing in those disclosures?

MR. ESRAM: Objection to form again.

A. Sorry. Can you point me to a specific thing? This line of questioning has gotten so abstract and vague. I just can't follow what the actual question is.

Can we just go back to the part of my declaration that you are referencing?

(EX1101 22:21-42; EX2015 114:6-118:15)

Petitioner's Expert Downplays Key Disclosures in Priority Applications

understanding he applied in his discussion of written description support. For example, when asked what a person of skill in the art would understand a web service provider node to be, he responded "That's too vague of a term for someone to know what that means without additional context." (EX2015, 99:23-102:23.) He eventually conceded that "presumably, it provides a service." (*Id.* 104:19-25; *see also* 106:15-107:6; 122:19-128:19.) He also would not answer questions about web servers, web service, proxy servers and like. (*See generally* EX2015, 86:3-94:22; 115:16-116:16; 214:6-228:22.) Accordingly, his opinions should be given no weight.

(PO Sur-reply at 5)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Expert Downplays Key Disclosures in Priority Applications

Dr. Mowry fails to consider these disclosures in the context of the specification as a whole. For example, when questioned about EX2004 Fig. 5, Dr. Mowry limited his answers to its description in the specification. (EX2015, 101:9-102:6, 105:8-106:4, 109:12-110:9; EX1127 ¶49, citing EX2004 [0124].) However, Dr. Mowry did not have any opinion as to whether other paragraphs of the specification, including, for example, '520 patent 22:21-31 (i.e., EX2002 [0373] and EX2004 [0102]), disproved the presence of an external server. (EX2015, 116:17-117:15, 123:9-124:4.) Importantly, Dr. Mowry was unable to offer any actual evidence to support his opinion that the specification of the Original Applications does not describe the use of an external server or external "node" to send messages. Dr. Smith's testimony—that the specification of the Original Applications as a whole shows the presence of an external node to send messages—is therefore unrebutted. (EX2005, ¶¶24-38, 83-85, Appendix 1.)

(PO Sur-reply at 8-9)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Even were *Riepling* eligible Prior Art, It Does Not Render the Challenged Claims Obvious (All Grounds)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Is Directed to a Different Problem from the Challenged Claims

[0004] Internet-based content sharing services provide users with the ability to upload and share data such as documents, media files, and other content. By using these content sharing services, registered users may easily share data with other registered users on the same content sharing service. To protect users of a content sharing service from unsolicited messages, malware, and other threats, guest access from unregistered accounts may be limited or denied, preventing unregistered users from sending data uploads to registered users.

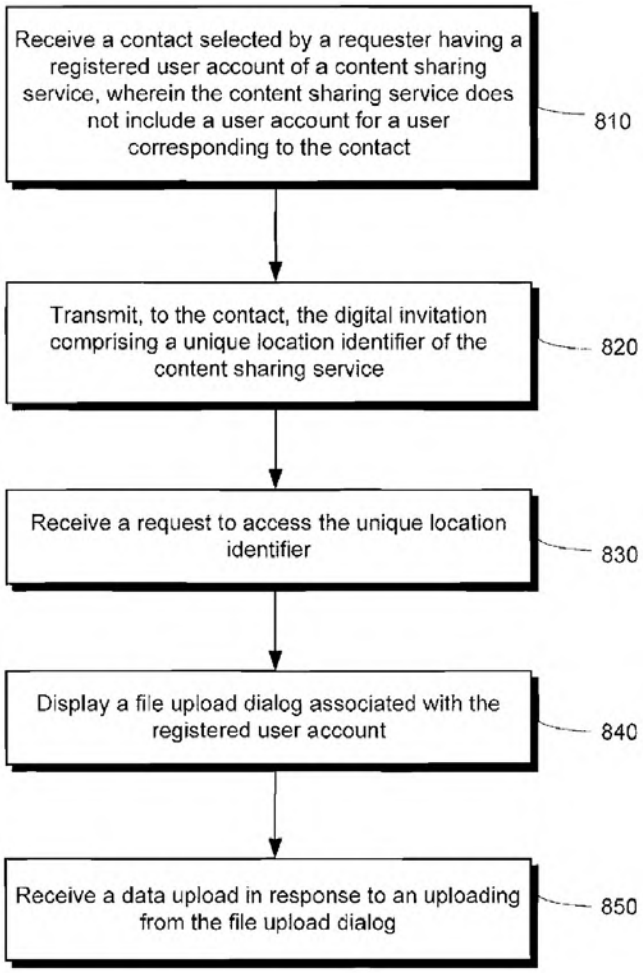
[0005] Thus, an unregistered user desiring to send a file through a content sharing service may be required to register a new user account, often a cumbersome and time consuming process. This may result in file upload delays, which may be problematic for time sensitive materials. Furthermore, unregistered users may have no interest in long-term usage of the content sharing service, thereby reducing the value and convenience proposition of user registration.

[0007] Accordingly, there is a need to overcome the drawbacks and deficiencies in the art by providing a way for a registered user of a content sharing service to receive content from an unregistered user in a convenient and secure manner.

(EX1107 [0004], [0005], [0007]; PO Sur-reply at 10)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Is Directed to a Different Problem from the Challenged Claims



[0051] Referring to step **810** of flowchart **800** in FIG. **800** and referring to system **100** of FIG. **1**, diagram **200** of FIG. **2** and diagram **400** of FIG. **4**, step **810** comprises receiving a contact selected by requester **111** having a registered user account of content sharing service **105**, wherein content sharing service **105** does not include a user account for a user corresponding to the contact. Requester **111**, using request device **110**, may access request interface **102** from the registered user account of the requester **111** in order to select a contact to receive digital invitation **410**. Sender **117** may be a user corresponding to the contact. After selecting a contact from list of contact information **230**, requester **111** may proceed to send, to the contact, digital invitation **410** by using grant access button **235**.

[0052] Proceeding to step **820** of flowchart **800** in FIG. **800** and referring to system **100** of FIG. **1**, diagram **300** of FIG. **3** and diagram **400** of FIG. **4**, step **820** comprises transmitting, to a contact, digital invitation **410** comprising a unique location identifier of the content sharing service **105**. After selection of a contact in step **810** is performed, content server **105**, using processor **102**, may further present upload parameter dialog **301** to allow requester **111** to specify and choose one or more upload parameters, which restrict the data selected for uploading based on content composition. As explained previously, the one or more upload parameters may also place restrictions on digital invitation **410**. After completing selection of the one or more upload parameters, requester **111** may use send button **330** to send digital invitation **410**, conforming to the one or more upload parameters, to the contact chosen in **810**. The digital invitation **400** may be sent using e-mail or another means of communication.

(EX1107 Fig. 8, [0051]-[0052]; PO Sur-reply at 11)

Riepling Is Directed to a Different Problem from the Challenged Claims

[0055] Proceeding to step **850** of flowchart **800** in FIG. **800** and referring to system **100** of FIG. **1**, diagram **500** of FIG. **5** and diagram **700** of FIG. **7**, step **850** comprises receiving a data upload in response to an uploading from file upload dialog **501**. From file upload dialog **501** presented in step **840**, sender **117** may elect one or more data files for uploading into the registered user account of requester **111**. When a data upload may be performed, content sharing service **105**, using processor **102**, may receive the data upload in response to an uploading from file upload dialog **501**. Once a data file has been successfully uploaded into the registered user account, the owner of the registered user account may access the data file through file entry link **703** displayed within data availability interface **702**. File entry link **703** may function like a conventional clickable data link.

(EX1107 [0055])

Riepling is directed to securely receiving unknown files from unregistered users of a content sharing service, not requesting files by file description

(PO Sur-reply at 10-11)

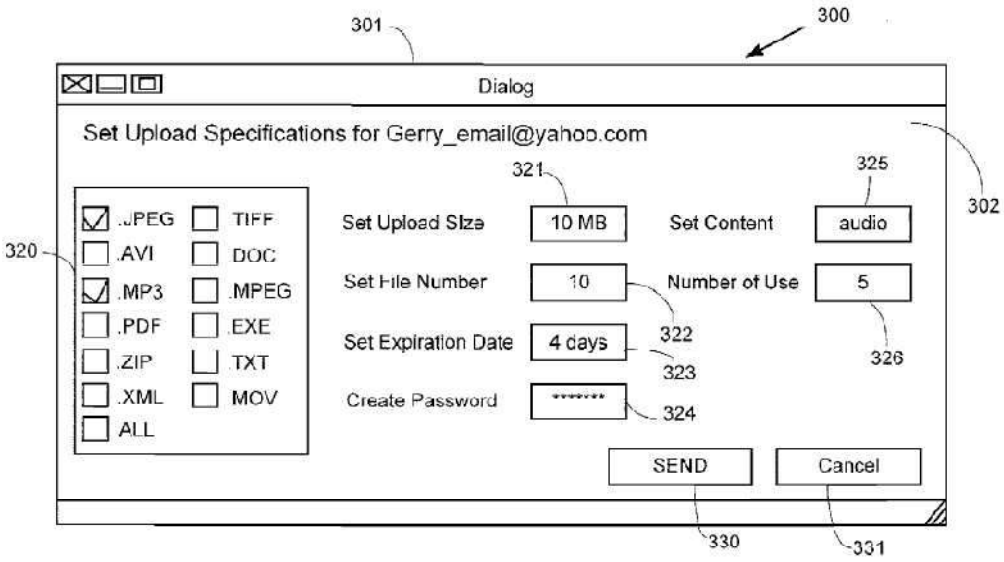
DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Does Not Teach or Suggest Element 9[b] and 9[g]

9[b]: receive, from the first node via at least one network, a file description via the first user interface element that includes a textbox, utilizing the first interface, the file description describing the at least one file in connection with the file request;

9[g]: in response to the receipt of the indication of the selection of the fourth user interface element and based on the file description, the indication of the at least one of the date or the time, and the indicia associated with the at least one email address; generate a first message addressed to the at least one email address and including the file request, the first message indicating the at least one of the date or the time on which the file request is based, and further including a fifth user interface element and at least one sentence including the file description describing the at least one file in connection with the file request;

Fig. 3



(EX1107 Fig. 3)

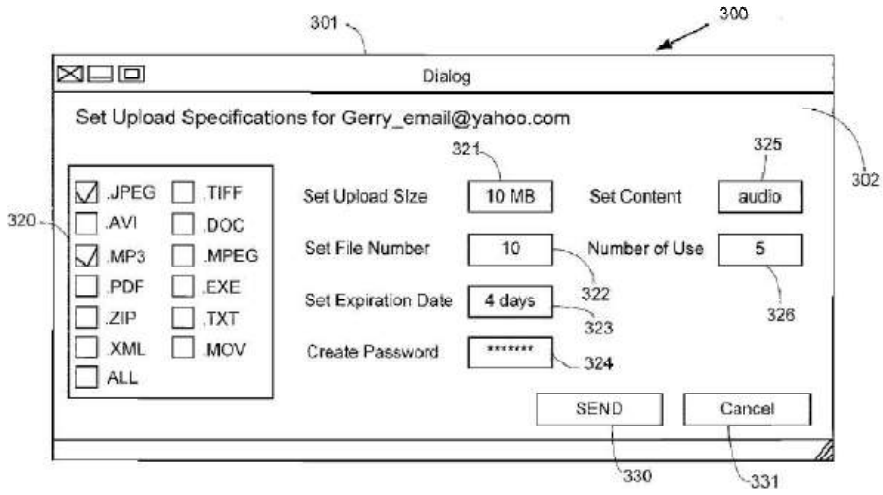
DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Mischaracterizes the Teachings of *Riepling*

First, PO argues Riepling does not teach that content field 325 "is a file description" because Riepling "does not describe" content field 325 outside of Figure 3. POR, 37-38. But the function of content field 325 is clear. Riepling explains Figure 3 "presents a diagram of an upload parameter dialog [301] for specifying upload parameters"—allowing the requester to specify parameters relating to the requested file, such as the file format ("file format check boxes 320") and the maximum file size ("size limit field 321"). In this context, box 325—labeled "Set Content" and containing the word "audio"—is logically understood to be a box that the requester can use to specify the content of the requested file—*i.e.*, "a file description." As Dr. Mowry explained, a POSITA "would have understood that the content field 325 is a text field for describing the content being requested [(*e.g.*, 'audio')]." EX1102, ¶70; *see also* DI, 29 ("the 'Set Content' entry is "audio," which is a file description"). Indeed, for Figure 3, where "audio" is identified in content

(Rep. at 10)

Fig. 3



(EX1107 Fig. 3)

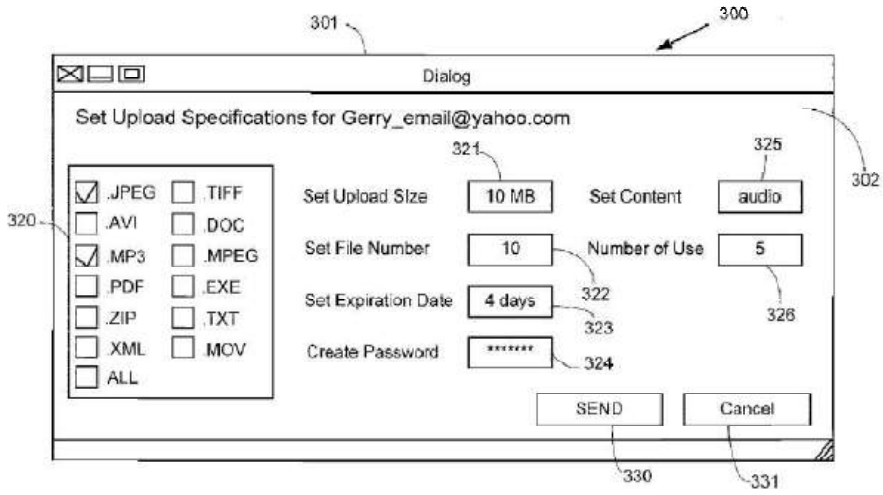
Fig. 3 of *Riepling* Does Not Include a Text Box Containing a Description of the Requested File

In its Reply, Petitioner continues to rely on "content field" 325 from Fig. 3 of *Riepling* as meeting element 9[b]. (Rep. at 9-11.) Notably, *Riepling* does not define it as a "field." And Petitioner fails to address how this so-called "content field"[6] includes a "textbox" and "describ[es] the at least one file in connection with the file request" as required by the claims.

First, Petitioner's argument around field 325 being a "textbox" is speculative and unsupported. Petitioner asserts that "box 325—labeled 'Set Content' and containing the word 'audio'—is ***logically*** understood to be a box that the requester can use to specify the content of the requested file," (Rep. at 10), but offers no basis for this "logical" inference. Its expert similarly characterizes field 325 as a "text field," without explaining why. (*Id.* at 10, citing EX1102 ¶ 70.) The Board in turn noted that "the fields of Figure 3 are generally described as ***text fields*** . . . ." (Paper 11 at 29.) However, none of these statements demonstrate that field 325 is a text ***box*** with text that can be entered by a file requester.

[6] Petitioner also mislabels field 325 in Fig. 3 as a "content field." Box 325 is a field labeled "Set Content."

Fig. 3



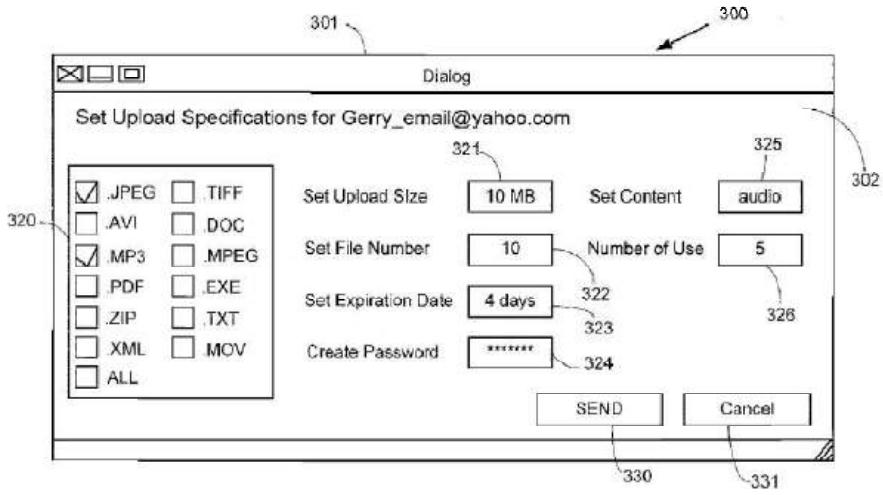
(EX1107 Fig. 3)

(PO Sur-reply at 11-12)

Fig. 3 of *Riepling* Does Not Include a Text Box Containing a File Description of the Requested File

Instead, "Set Content" field 325 is used to set further restrictions on the type of data uploaded by the recipient (here, "Gerry_email@yahoo.com") beyond file type checkboxes 320. Even though the requesting user may expect to receive a plurality of file types (such as JPEG and MP3 files), only a subset of such files (specifically, those containing "audio") is permitted to be received from the recipient uploader. (EX1107 ¶ [0032].) The word "audio" in field 325 is therefore not a textual entry arbitrarily entered by the requester—in contrast to the textual description in the '520 patent's own example ("Please send whatever documents you have about your Uncle Joe.") (EX1101, Fig. 6D)—but a predetermined (or automatically generated) content type corresponding to file type checkboxes 320. Dr. Mowry agrees. (*See* EX1127, ¶ 102 ("*Riepling* explains that '[r]equester 111 may select from file format check boxes 320 any number of file extensions, thereby ***generating*** a file format limitation . . . .")

Fig. 3



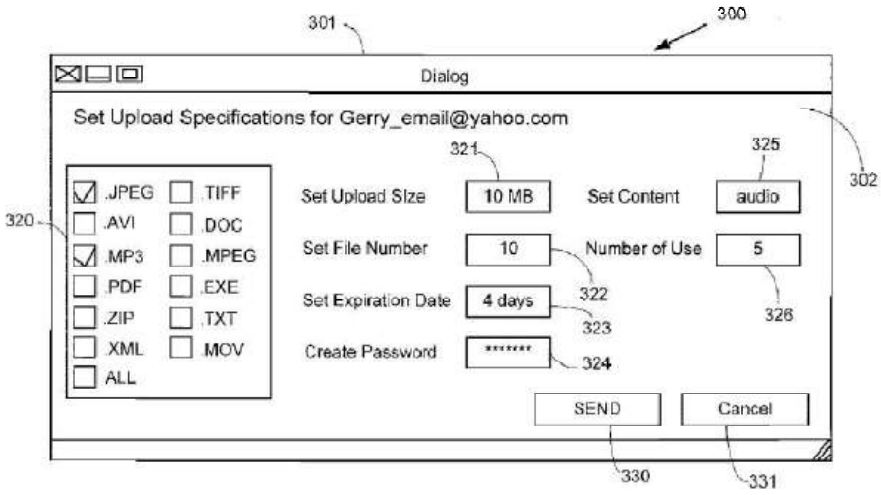
(EX1107 Fig. 3)

(PO Sur-reply at 12-13)

Fig. 3 of *Riepling* Does Not Include a Text Box Containing a Description of the Requested File

Second, to the extent field 325 is a "textbox" (which PO disagrees with), it does not describe "the at least one file in connection with the file request." Instead, Figure 3 shows an "upload parameter dialog for specifying upload parameters." (EX1107, [0028].) *Riepling* explains that the requesting user may request a "data upload" from an unregistered user simply by providing the recipient with a "unique location identifier of the content sharing service." (EX1107, Abstract, Fig. 8; [0050]-[0055].) At the time the request for data upload is sent to the recipient, the requester does not have any information about any specific file(s) that may be uploaded by the recipient. (*Cf.* EX1001, 2:1-10.) Any "upload parameters" in Figure 3 ***cannot*** "describe the at least one file in connection with the file request," because no such "description" is known to the requester *a priori*. *Riepling* therefore protects the requester from the unintended consequences of receiving unknown files from unregistered users—namely, "unsolicited messages, malware, and other threats" (EX1107, [0004])—by allowing the requester to specify upload parameters to limit the type of data that may be uploaded to the requester's account. (EX1107, [0029].)

Fig. 3



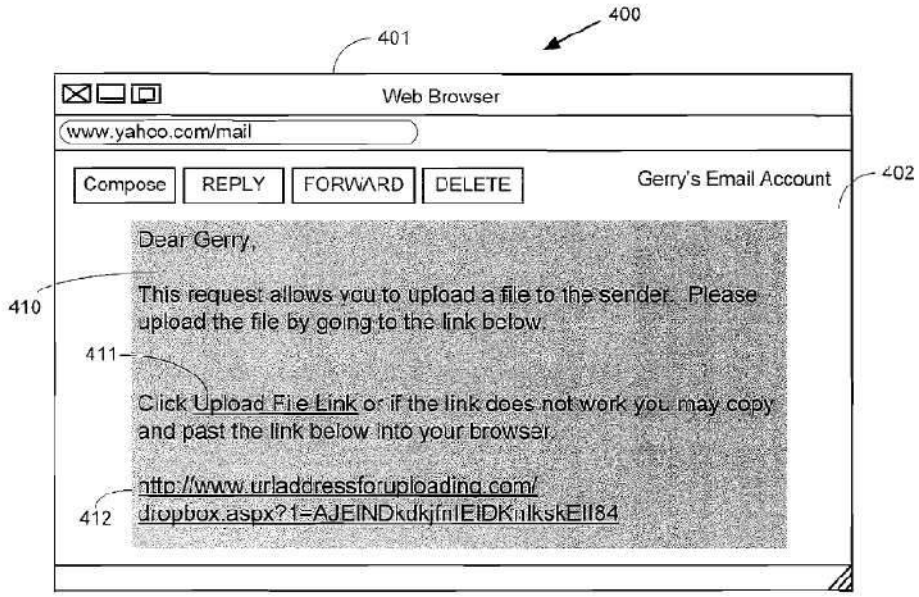
(EX1107 Fig. 3)

(PO Sur-reply at 13-14)

Riepling Does Not Teach or Suggest Element 9[g]

9[g]: in response to the receipt of the indication of the selection of the fourth user interface element and based on the file description, the indication of the at least one of the date or the time, and the indicia associated with the at least one email address; generate a first message addressed to the at least one email address and including the file request, the first message indicating the at least one of the date or the time on which the file request is based, and further including a fifth user interface element and at least one sentence including the file description describing the at least one file in connection with the file request;

Fig. 4



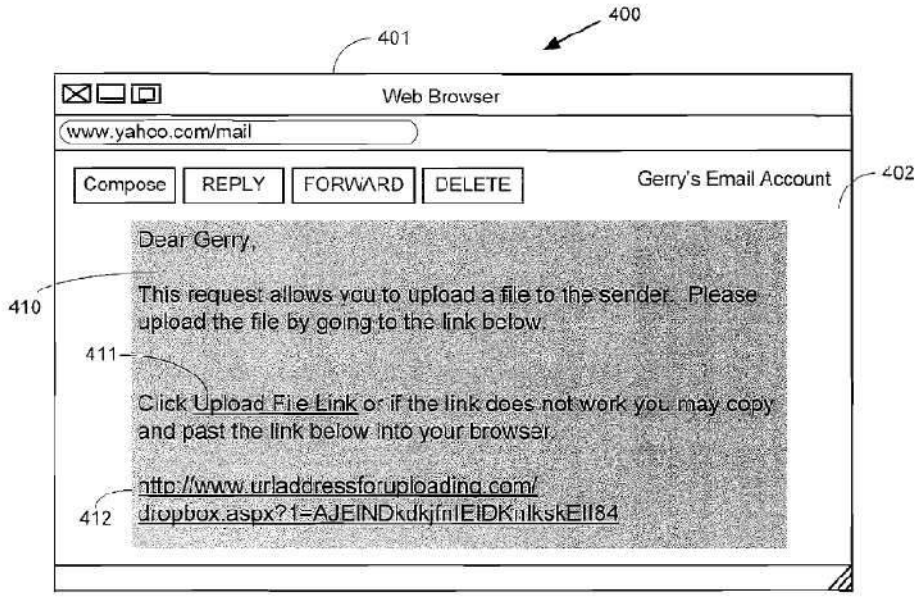
(EX1107 Fig. 4)

Petitioner's Obviousness Theories Are Inconsistent with its Expert's Opinions

First, PO incorrectly argues Petitioner "fails to describe any motivation to modify Riepling in such a way or why such a design choice would be obvious." POR, 46. Not so. Petitioner expressly explained "[a] POSITA would have found it an obvious design choice to include these pieces of information in the digital invitation ***in order to inform sender 117 of the kind of file requester 111 is requesting and by when requester 111 would like to receive the file.***" Pet., 31

(Rep. at 14)

Fig. 4



(EX1107 Fig. 4)

Petitioner's Obviousness Theories Are Inconsistent with its Expert's Opinions

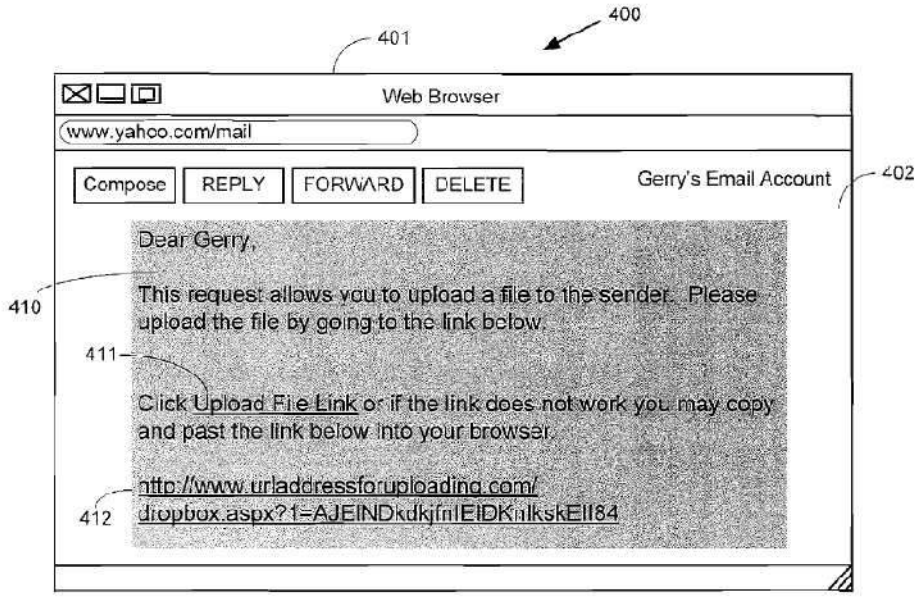
Q. I think that Figure 4 of Riepling's is a depiction of something that's sent to the party that the file be requested from.

A. Well, that figure is showing an example of something, and it appears to be chopped off.

You only see a portion of the email message at the top. The portion that includes just the link, but a person of ordinary skill in the art reading the patent would realize that it would be likely that the chopped off part would include that information in box with 325.

(EX2015 at 239:21-240:7)

Fig. 4



(EX1107 Fig. 4)

(PO Sur-reply at 15-17)

Petitioner's Obviousness Theories Are Inconsistent with its Expert's Opinions

A. I think that, as I said, a person of ordinary skill in the art reading Riepling would find it obvious to include the information from Box in 325 in the email message, in the invitation, which is shown in Figure 4 -- well, Figure 4 showing a portion of it.

BY MR. DAHLGREN:

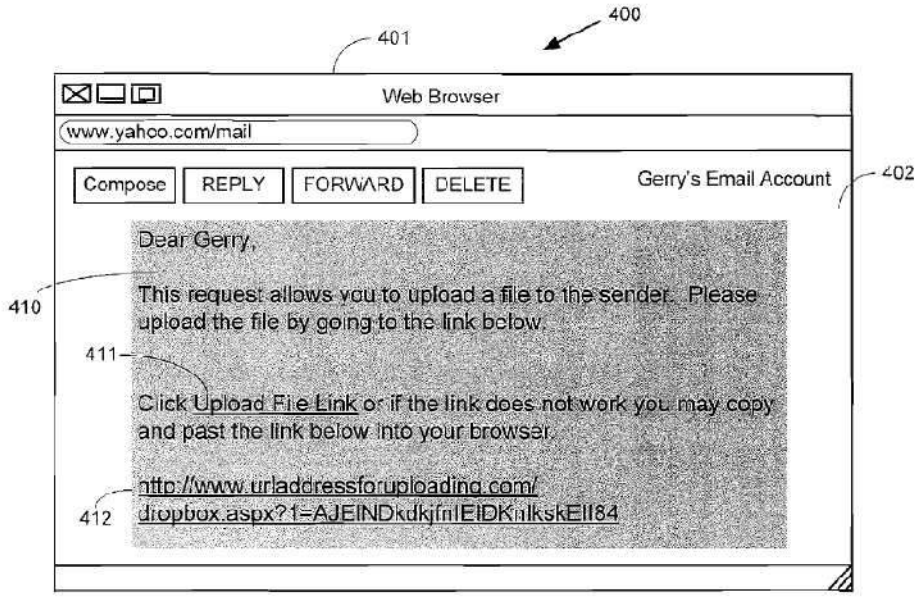
Q. Is there anything in Riepling that you can point to that you believe would provide some type of motivation or reason or any rationale for a person of ordinary skill in the art to modify Riepling in that way?

MR. ESRAM: Objection to form.

A. That's not modifying Riepling. That's what Riepling would do.

(EX2015 at 243:2-16)

Fig. 4



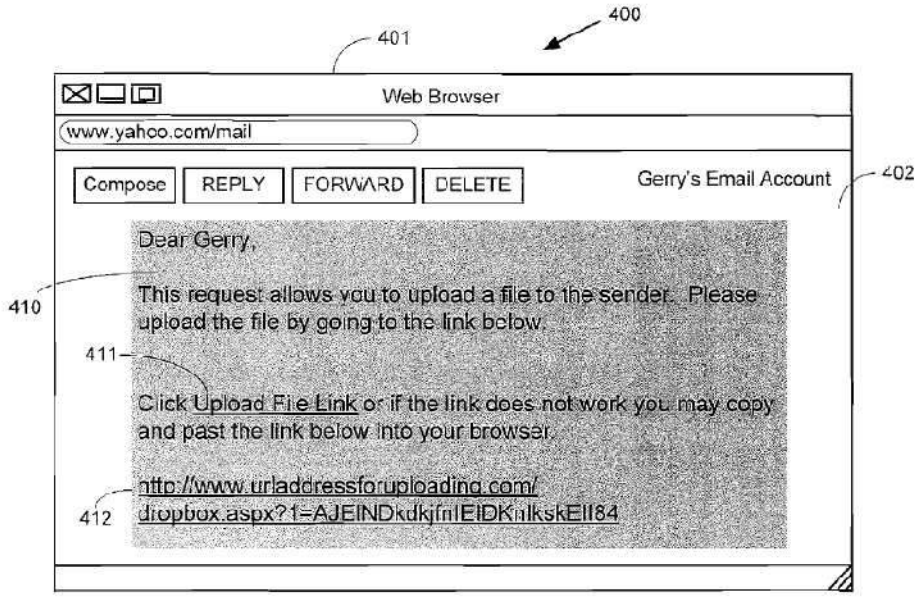
(EX1107 Fig. 4)

(PO Sur-reply at 15-17)

Petitioner and its Expert Mischaracterize the Teachings of *Riepling*

itself. As set forth above, *Riepling* consistently describes that the digital invitation sent to an uploader recipient includes a "unique location identifier of the content sharing service." (EX1107, Abstract, Fig. 8, [0034], [0035], [0038], [0052], [0053]; cls. 1, 11.) Nothing more. And in any event, this statement contradicts Petitioner's (and Dr. Mowry's own) assertions that including the claimed information would have required modifications to the digital invitation of *Riepling*, which are in any event unsupported by any factual evidence.

Fig. 4



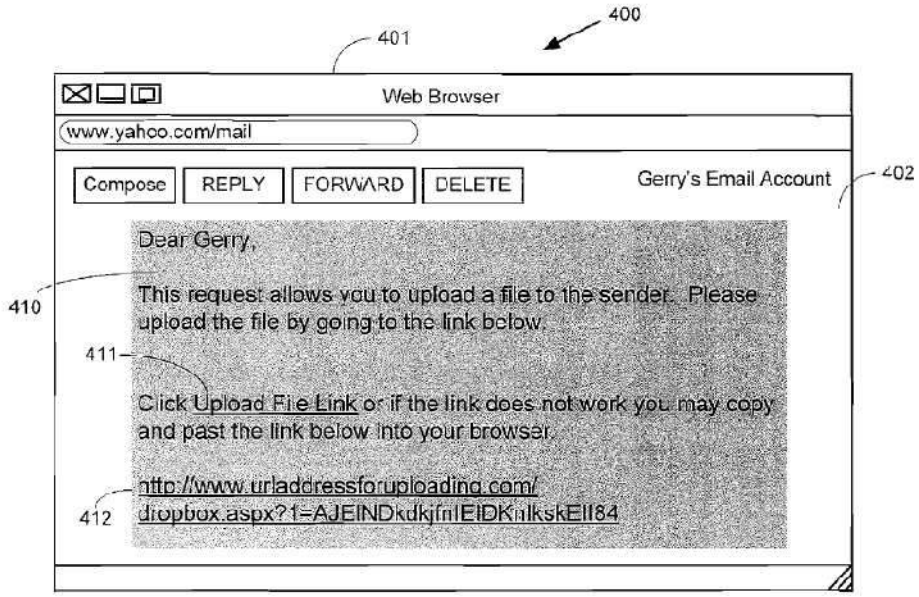
(EX1107 Fig. 4)

(PO Sur-reply at 18)

Petitioner and its Expert Mischaracterize the Teachings of *Riepling*

Indeed, *Riepling* describes that at least one type of information in the "upload parameters" interface of Figure 3, namely, the password, is conveyed to the recipient ***separately*** from the digital invitation, "in advance." (EX1107, [0044].) It is just as likely that file types, expiration date, or other upload parameters are conveyed to the recipient separately from digital invitation 410, if at all. Neither Petitioner nor Dr. Mowry explain why they must be included in the digital invitation of Figure 4:

Fig. 4

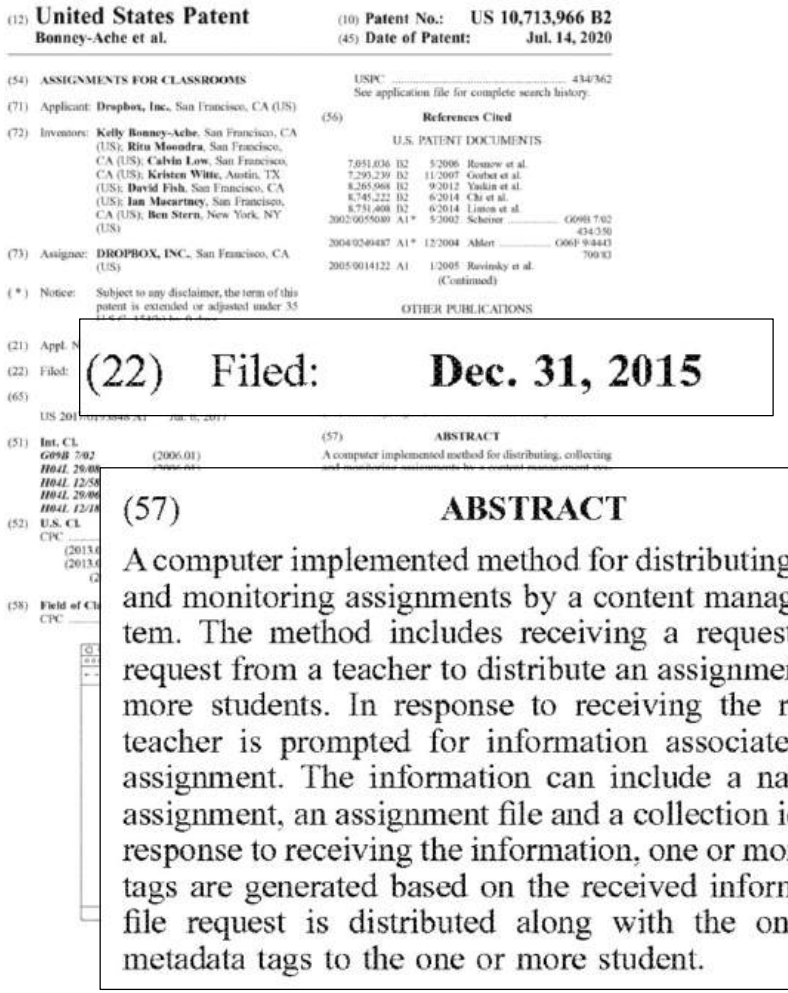


(EX1107 Fig. 4)

(PO Sur-reply at 18)

Petitioner Pursued Patents on File Request Features Years Later

In sum, *Riepling* does not teach that the "date or time on which the file request is based" or "file description describing the at least one file in connection with the file request" could or would be included in **digital invitation 410**, as required by the claim element 9[g]. (EX2005, ¶100.) Instead, Petitioner uses the '520 patent as a roadmap to support its argument, which is improper hindsight. (*Id.*; EX1001, 2:1–10) Notably, Petitioner filed patent applications on file-request functionality years later, which suggests that the file-request inventions of the '520 patent were not obvious. (*E.g.*, EX2003; EX2005, ¶ 100.)



(12) **United States Patent**
Bonney-Ache et al.

(10) **Patent No.: US 10,713,966 B2**
(45) **Date of Patent: Jul. 14, 2020**

(54) **ASSIGNMENTS FOR CLASSROOMS**

(71) Applicant: **Dropbox, Inc.**, San Francisco, CA (US)

(72) Inventors: **Kelly Bonney-Ache**, San Francisco, CA (US); **Ritu Moondra**, San Francisco, CA (US); **Calvin Low**, San Francisco, CA (US); **Kristen Witte**, Austin, TX (US); **David Fish**, San Francisco, CA (US); **Ian Macartney**, San Francisco, CA (US); **Ben Stern**, New York, NY (US)

(73) Assignee: **DROPBOX, INC.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35

(21) Appl. N

(22) Filed:

(65)

US 201

(51) **Int. Cl.**
G09B 7/02 (2006.01)
H04L 29/08
H04L 12/58
H04L 29/06
H04L 12/18

(52) **U.S. Cl.**
CPC
(2013.
(2013.
(

(58) **Field of Cl**
CPC

USPC 434/362
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

7,051,036	B2	5/2006	Rosnow et al.	
7,293,239	B2	11/2007	Gorbet et al.	
8,265,968	B2	9/2012	Yadkin et al.	
8,745,222	B2	6/2014	Chi et al.	
8,751,608	B2	6/2014	Limon et al.	
2002/0055089	A1*	5/2002	Scheirer	G09B 7/02 434/350
2004/0249487	A1*	12/2004	Ablett	G06F 9/4443 700/83
2005/0014122	A1	1/2005	Ravinsky et al.	

(Continued)

OTHER PUBLICATIONS

(22) Filed: **Dec. 31, 2015**

(57) **ABSTRACT**

A computer implemented method for distributing, collecting and monitoring assignments by a content management system. The method includes receiving a request for a file request from a teacher to distribute an assignment to one or more students. In response to receiving the request, the teacher is prompted for information associated with the assignment. The information can include a name for the assignment, an assignment file and a collection identifier. In response to receiving the information, one or more metadata tags are generated based on the received information. The file request is distributed along with the one or more metadata tags to the one or more student.

(POR at 47 (citing EX2003); *see also* EX2005 at 100)

No Obviousness Based on *Meisels* (Grounds 3-4, 6)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Meisels Does Not Teach or Suggest Elements 9[b] and 9[g]

In any event, Petitioner further fails to show that *Meisels* cures *Riepling*'s deficiencies as to elements 9[b] and 9[g]. Therefore, for reasons set forth above, Ground 3 fails.

(Sur-Reply at 21; POR at 53-55)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Meisels Does Not Teach or Suggest Elements 9[l] and 9[o]

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

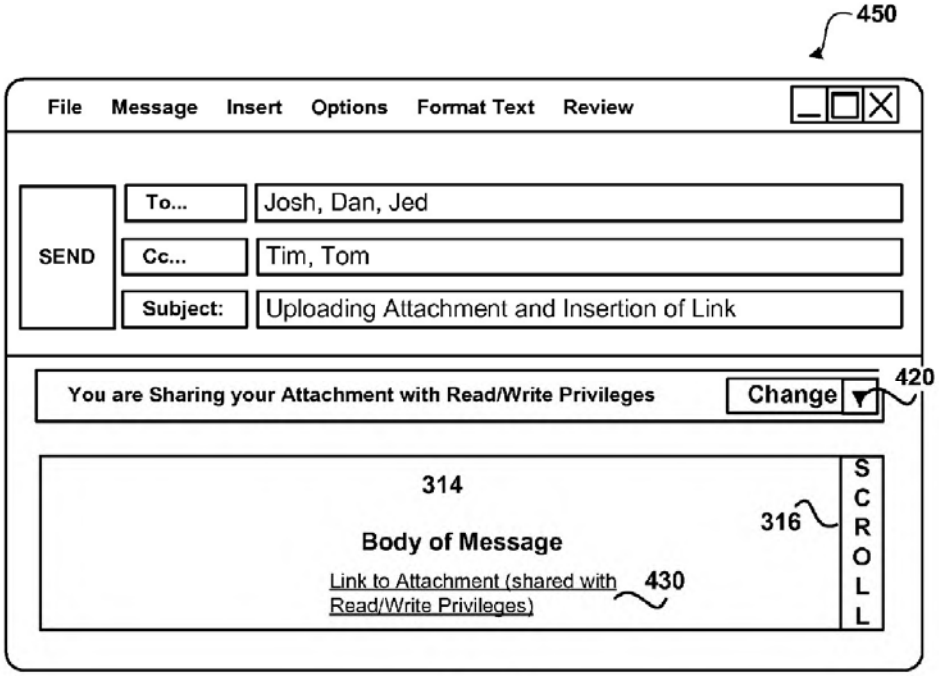


FIG.4

(EX1108 Fig. 4)

(POR at 54)

Meisels Does Not Teach or Suggest Elements 9[l] and 9[o]

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

teachings with respect to Ground 1, above. *Meisels* "discloses an electronic message with an automatically inserted [] link to a document in place of the document attachment itself." (Paper 11 at 21.) But *Meisels*'s process is divorced entirely from any other interfaces or software processes. (EX2005, ¶116.) A user must manually create an email, manually confirm to upload an attachment to insert a link to the attachment, and manually send the email to its recipient. (*Id.*; EX1108 ¶¶3, 37, 59) Then, selecting the link merely retrieves the attachment. (EX1108 at ¶ 23 ("Using the link generally retrieves the content that is pointed to by the link."); EX2005, ¶116.)

(POR at 53-54)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Meisels Does Not Teach or Suggest Elements 9[l] and 9[o]

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

In contrast, the claimed "second message" of claim element 9[l] is generated "***in response to the receipt of the indication of the selection of the seventh user interface element***" and "includes an eighth user interface element" such that the claimed apparatus will, "in response to the receipt of the indication of the selection of the eighth user interface element of the second message, ***cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node***." (claim elements 9[l] and 9[o].) Likewise, in claim element 14[j], the "claimed second message" of claim 14

(POR at 54)

Meisels Does Not Teach or Suggest Elements 9[l] and 9[o]

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

[0003] An option is presented when composing an electronic message to automatically upload and permission an attachment to the electronic message. A single click may be received to upload the attachment to a shared location (e.g. network share, online service, and the like). Permissions may be automatically set for the message recipients of the electronic message. For example, a default set of permissions and/or custom permissions may be set for each of the recipients listed on the To:/Cc:/Bcc: lines. A link to the shared location may also be automatically inserted into the electronic message along with or in place of the attachment.

(EX1108 ¶ 3)

(POR at 54)

Meisels Does Not Teach or Suggest Elements 9[l] and 9[o]

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

[0037] According to an embodiment, the attachments are uploaded to the shared location before the electronic message is sent. The attachment(s) may be uploaded upon selection of the upload option or at a point before the message is delivered to the recipients. For example, the attachment(s) designated to be uploaded may be uploaded in response to the send option being selected. In this way, if a user changes their mind in uploading an attachment and/or sending the message, the attachment will not be uploaded to the shared location until the message is instructed to be sent.

[0059] Flowing to operation **840**, a determination is made as to when to upload the attachments) to the shared location (s). According to an embodiment, the attachments are uploaded to the network share before the electronic message is sent. The attachment(s) may be uploaded upon selection of the upload option or at a point before the message is delivered to the recipients. For example, the attachment(s) designated to be uploaded may be uploaded in response to the send option being selected. In this way, if a user changes their mind in uploading an attachment and/or sending the message, the attachment will not be uploaded to the network share until the message is instructed to be sent. The attachment is uploaded at the determined time.

(EX1108 ¶¶ 37, 59)

(POR at 54)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Admits *Meisels* Does Not Teach Generation of a Message (Element 9[l])

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

14[j]). POR, 53-54. But Petitioner does not rely on Meisels for the process of creating the second message. Petitioner argues only that it would have been obvious to add a link (*i.e.*, "an eighth[/sixth] user interface element") as taught by Meisels into Riepling's notification message. Pet., 65-66. There is no dispute that Riepling discloses the required process of generating and sending of a notification message. (Rep. at 22)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

The *Riepling+Meisels* Combination Does Not Teach a "Fourth Interface" (Element 9[o])

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

to that attachment" is misplaced. POR, 55. This argument, at best, relates to element 9[o], for which Petitioner does ***not*** rely on Meisels. Regardless, a POSITA would have understood that, in Meisels, clicking on the link in the email would open an interface that displays and provides access to the uploaded file, similar to Riepling's data availability interface 702 with file entry link 703. *See* EX1107, ¶49, Figure 7. EX1127, ¶¶111-113.

(Rep. at 23)

The *Riepling+Meisels* Combination Does Not Teach a "Fourth Interface" (Element 9[o])

9[l]: in response to the receipt of the indication of the selection of the seventh user interface element, generate a second message that includes an eighth user interface element, and that does not include a file attachment with the second message, the second message indicating that the file request has been responded to;

9[o]: in response to the receipt of the indication of the selection of the eighth user interface element of the second message, cause, at the first node, display of at least one fourth interface with a reference to the at least one file for allowing access to the at least one file at the first node.

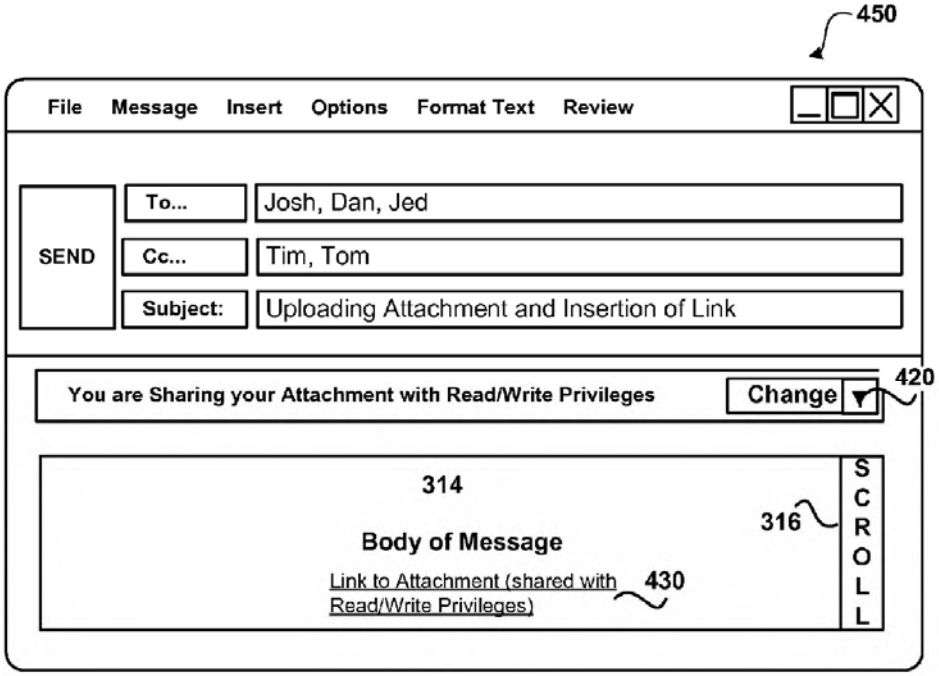


FIG.4

(EX1108 Fig. 4)

(PO Sur-reply at 21 n.8)

No Obviousness Based on *Manzano* (Grounds 2, 4-6)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Does Not Cure Deficiencies of *Riepling*

In Grounds 2, 4, 5, and 6, Petitioner continues to rely on *Riepling* alone for elements 9[b] and 9[g]. For the reasons set forth above and in the Patent Owner Response, Petitioner's arguments as to those elements fall short. Therefore, Grounds 2, 4, 5, and 6 also fail.

(PO Sur-reply at 21)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

A POSITA Would Not Have Been Motivated to Combine *Manzano* with *Riepling*

Manzano teaches a system that, among other things, allows a user to download files stored remotely. (EX1105 [0041], [0050]-[0051].) Specifically, *Manzano* is directed to allowing a user to transfer their own files, in their entirety, from a first computer to a second computer. (EX1005 ¶¶[0032]-[0035] (describing "mobile agents" and "mobile objects") [0033]–[0043] (describing "proxy objects"), [0050] (describing "ghosted" icons).) Petitioner does not dispute these aspects of *Manzano*. (Rep., 25.)

Riepling, in contrast, is directed to offering improved security for registered users of a content sharing service when receiving data uploaded by unregistered users. (EX1007, Abstract, [0004]-[0005], [0007].) Petitioner fails to address how the problem solved by *Riepling* is compatible with a system directed to allowing a user to transfer their own files, in their entirety, from a first computer to a second computer, as taught by *Manzano*, or why a POSITA would have been motivated to modify *Riepling*, with its heightened concerns about security, to store anything automatically on the requestor's local device. That is contrary to the overarching concerns underlying *Riepling*'s teachings.

(PO Sur-reply at 22)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Manzano Utilizes Mobile Agents Rather than Email

2. *Manzano*

Petitioner relies on U.S. Patent App. Pub. No. 2010/0005138 to Manzano et al. ("*Manzano*") (EX1105), "published on January 7, 2010, as § 102(b) prior art." (Pet. at 5.) *Manzano* is titled "Electronic File Sharing" and purports to address problems involving "exchang[ing] large files using awkward prior-art methods such as e-mail attachments . . . ." (EX1105 ¶[0003]; EX2005, ¶46.) *Manzano* teaches sharing files between devices using "mobile agents" or a "mobile agent object" that exists on a first computer system and can "transplant itself" to a second computer system, while preserving its current execution state. (EX1105 ¶[0016]; EX2005, ¶46.) This is performed by the mobile object creating another mobile object that functions as a "proxy object" on the server. (EX1105 ¶[0035]; EX2005, ¶46.)

BACKGROUND OF THE INVENTION

[0003] Computer-network users typically exchange large files using awkward prior-art methods such as e-mail attachments or file transfer protocol (FTP) sites. Most email systems limit the size of attachments, and, as a result, users are forced to send multiple e-mails with discrete files attached to each. FTP sites are typically only moderately secure and their use is often difficult to manage for inexperienced users.

(POR at 8-9; EX1105 at [0003]; Sur-reply at 22)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Riepling Directed to Interactions with Unregistered Users

Riepling is titled "Method and System for Sending a Digital Invitation Requesting a Data Upload" and purports to address a way for registered users of a content sharing service to receive content from unregistered users "in a convenient and secure manner" through the use of invitations. (EX1107 at Abstr., ¶¶[0002], [0007]-[0008]; EX2005, ¶45.) It purports to do so while protecting registered users from security concerns such as, among other things, unsolicited messages from unregistered accounts. (EX1107 ¶[0004]; EX2005, ¶45.)

(POR at 8; PO Sur-reply at 22)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

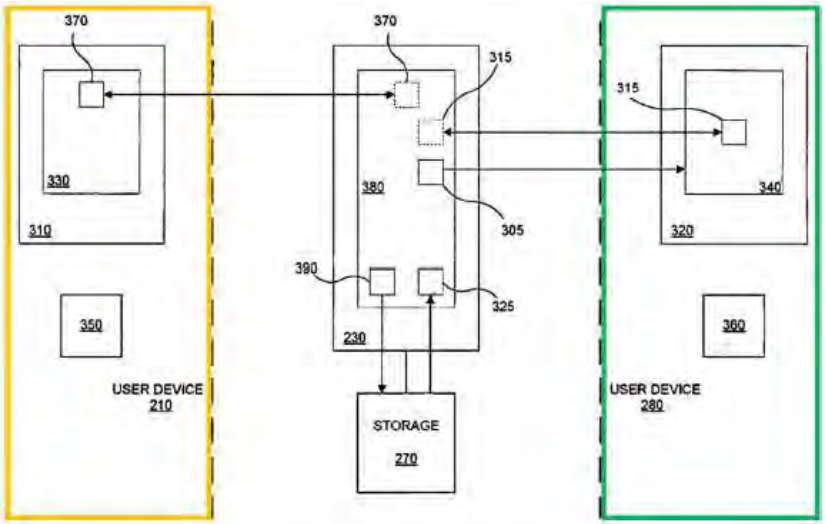
Manzano Does Not Teach Creating a Representation of an Uploaded File at the First Node

22[k]: causing, at the first node, receipt of code for storage at the first node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: based on the detection of the indication of the selection of the seventh user interface element, cause creation of a second representation of the at least one file, in a location among one or more folders, that is stored at the first node and that is displayable via the file explorer interface of the client-based file explorer application.

Manzano. (*Id.*) *Manzano* discloses that a recipient client is instructed "to set up an electronic file folder into which the transferred file will be place." (*Id.*; EX1105 ¶ [0041].) It does not disclose a representation of a file being created in a location among one or more folders on a file explorer interface as recited in claims 11–13, 16 and 22–24. (EX2005, ¶126.) Indeed, *Manzano* only discloses transferring ***files*** into a specific folder that is created for the files. (*Id.*; *E.g.*, EX1105, ¶¶[0041], [0052].) It does not disclose the creation of a representation in a location among one or more folders on the client's client-based file explorer application. (EX2005, ¶126; *E.g.*, EX1105, ¶¶[0041], [0052].)

(POR at 59)

Petitioner Does Not Assert that *Manzano* Teaches Creating a Representation of an Uploaded File at the First Node



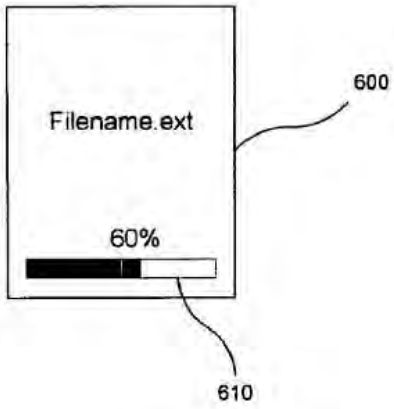
EX1105, FIG. 3

Third, similar to how a user at **receive device 115** (*i.e.*, "*the second node*") uploads a file into the registered user account in Riepling (elements 9[j]–9[k]), Manzano discloses that a user of **user device 210** (*i.e.*, "*the second node*") may share a file with **user device 280** (*i.e.*, "*the first node*") by placing a file into the folder "with which the user device 280 has been associated as a recipient." *Id.*, ¶33. In response, the file gets transferred to and stored at server 230. *See id.*, ¶¶34–38. Server 230 communicates with **user device 280** and "instructs the recipient client 320 to set up an electronic file folder into which the transferred file will be placed." *Id.*, ¶41. EX1102, ¶151.

(Pet. at 51)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Does Not Assert that *Manzano* Teaches Creating a Representation of an Uploaded File at the First Node



EX1105, FIG. 6

Manzano further discloses that "file icons, such as the icon 600, in a workspace displayed on display device 290 can visually indicate whether a corresponding file (such as a file, the transfer of which may be as is discussed herein with reference to FIG. 3) has been downloaded to or is otherwise stored on (i.e., is local to)… **user device 280**." EX1105, ¶50, Figure 6; *see also id.*, ¶52 ("'workspace' corresponds to a respective electronic folder"). If a shared file "is not local to **user device 280**, the icon 600 may be ghosted (i.e., semi-transparent)… to indicate the non-local status of the file." *Id.*, ¶50. Manzano's ghosted icon 600 corresponds to "*a representation of the at least one file*." Moreover, a POSITA would have understood that icon 600 would be displayed "*among one or more folders*" in Manzano's file-transfer application. EX1102, ¶152.

(Pet. at 52)

No Obviousness Based on *Garcia* (Grounds 5-6)

Garcia Does Not Cure Deficiencies of *Riepling*

In Grounds 2, 4, 5, and 6, Petitioner continues to rely on *Riepling* alone for elements 9[b] and 9[g]. For the reasons set forth above and in the Patent Owner Response, Petitioner's arguments as to those elements fall short. Therefore, Grounds 2, 4, 5, and 6 also fail.

(PO Sur-reply at 21)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Does Not Teach Creating a Representation of an Uploaded File at the First Node

22[k]: causing, at the first node, receipt of code for storage at the first node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: based on the detection of the indication of the selection of the seventh user interface element, cause creation of a second representation of the at least one file, in a location among one or more folders, that is stored at the first node and that is displayable via the file explorer interface of the client-based file explorer application.

(POR at 59)

by *Garcia*. (EX2005, ¶127.) *Garcia* discloses that the recipient of a shared folder ***does download*** the ***contents*** of the shared folder at the node where the folder share invitation is received. (*Id.*) That the user's ***other devices*** (e.g., different nodes) may show representations of the shared folder is beside the point. (*Id.*) *Garcia* does not disclose the creation of a representation at the node where the invitation to the shared folder is received. (*Id.*) *Garcia* discloses "in one embodiment, when a user drags a folder to a particular contact, ***the folder*** or ***link to the folder*** is automatically sent (via email, text, or other means) to the contact," (EX1104, 11:16–19), and "when the contact clicks on the link to folder 115, the ***contents*** of the folder are shared with the contact," (EX1104, 11:29–31.) (EX2005, ¶127; *See* (POR at 59-60)

Petitioner Does Not Assert that *Garcia* Teaches Creating a Representation of an Uploaded File at the First Node

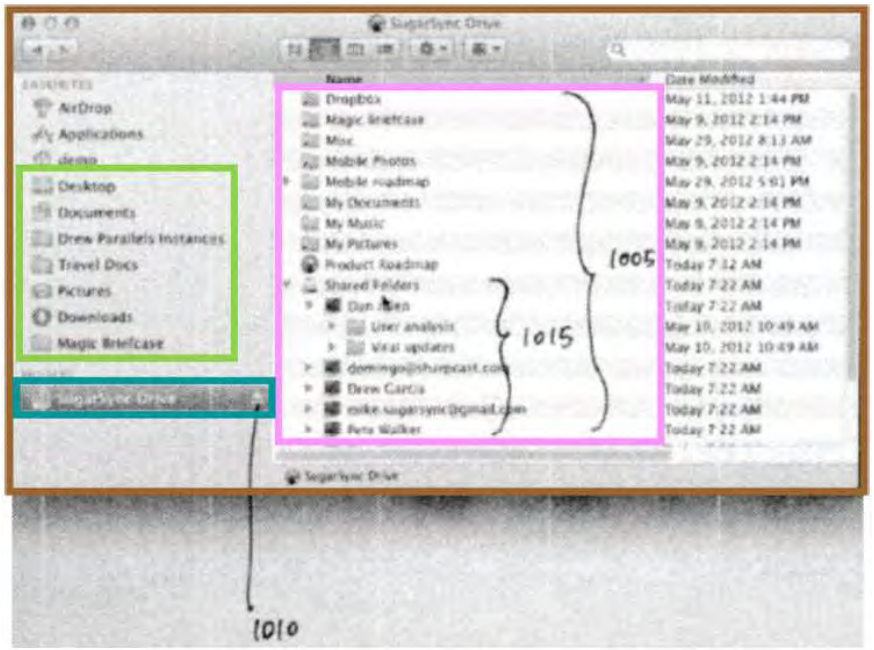
22[k]: causing, at the first node, receipt of code for storage at the first node and cooperation with a file explorer interface of a client-based file explorer application, for being utilized to: based on the detection of the indication of the selection of the seventh user interface element, cause creation of a second representation of the at least one file, in a location among one or more folders, that is stored at the first node and that is displayable via the file explorer interface of the client-based file explorer application.

Second, Garcia discloses a synchronization system 2100 that includes a synchronization server 2120 in communication with a client device 2130 via an internet/network 2110. *See* EX1104, 6:36–67, Figure 21. Garcia describes that "all synchronized objects between a user's client devices and between users are stored on server 2120 and are accessible from any client device having access to the server 2120." *Id.*, 6:45–48; *see also id.*, 2:37–41 ("access, via user interface, a virtual drive on the local device, where… a user is able to access items (e.g., files) in any object (e.g., folders), including objects located only on a remote device."). Garcia explains that "[w]ithin client device 2130 is local synchronization client software 2140." *Id.*, 7:1–2. The local synchronization client software 2140 "functions to synchronize client device 2130 with the synchronization server 2120." *Id.*, 7:2–3. "The local synchronization software 2140 is operatively coupled to a file system abstraction module 2150, within operating system 2155" of the client device 2130. *Id.*, 7:22–24. The operating system 2155 is "connected to a **user interface 2170** (e.g., MAC FINDER or WINDOWS EXPLORER) which displays the physical drive user interface 2180 and the **virtual drive user interface 2190.**" *Id.*, 7:25–29. The **user interface 2170** thus corresponds to "*a file explorer interface of the client-based file explorer application*." EX1102, ¶205.

(Pet. at 74-75)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Does Not Assert that *Garcia* Teaches Creating a Representation of an Uploaded File at the First Node



EX1104, FIG. 10 (excerpted)

Garcia explains that "any file in folders 1005 in the synchronization system may be accessed through [the] virtual drive 1010, including folders located only on a remote device (e.g., work computer 240) or folders located only on the synchronization server 220." *Id.*, 10:60–64. Folders 1005, including Shared Folders 1015, correspond to "*the one or more folders that is stored at the at least one server*." Moreover, Garcia illustrates, in the interface of Figure 10, folders 1005 among other conventional folders (*e.g.*, "Desktop," "Documents," "Downloads," etc.; *i.e.*, "*one or more folders that is stored at the first node*"), which are local to the client device. A POSITA would have understood that, at a moment in time, representations of older synchronized folders among the folders 1005, as opposed to a folder or a file synchronized at that moment, would have already existed among the other conventional folders. EX1102, ¶207.

(Pet. at 77-78)

Garcia Does Not Teach Creating a Representation of an Uploaded File at the First Node

Referring to FIG. **20***a*, a user is enabled to synchronize one or more folders on any synchronization client to the synchronization server and other synchronization clients (step **2005**). All folders, across a plurality of synchronization clients that a user has backed up or synchronized to the synchronization server, are identified, as well as any folders that the user shares with other users (step **2010**). Each of the synchronization clients stores metadata with information on all folders shared with or by the user and all folders that the user has backed up or synced to the synchronization server (including the contents of the folder). The synchronization client on the client device uses this metadata to generate the virtual drive and the user interface.

(EX1104 at 5:14-26)

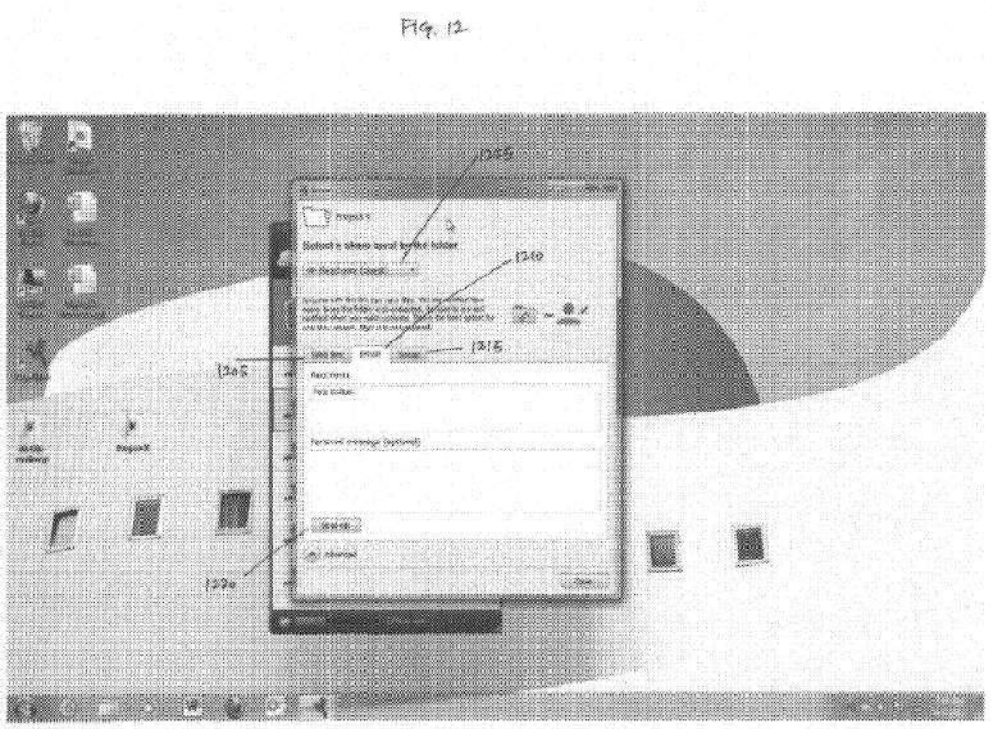
In response to detecting a change in a folder in the virtual drive, the folder is updated on the synchronization server (step **2025**). All remote clients of the user in the synchronization system are identified, as well as all remote synchronization clients, if any, of other users whom have access to the changed folder (step **2030**). In one embodiment, all synchronization clients that have an interest in updates to the folder "register" for notices of changes to the folder. The identified clients are notified of the change to the folder (step **2035**). In response to the change being a change to content within the folder, the content of the folder is automatically updated on any identified remote clients that have a corresponding folder. In one embodiment, the synchronization server first sends metadata to all the identified clients, notifying them of the change. The synchronization clients that have their own copy of the changed folder will then request the applicable file data from the server and the server will download the requested file data to the requesting synchronization clients.

(EX1104 at 5:40-62)

Garcia teaches that folders are synchronized with other clients, not at the first node where a file requester expects to receive a file uploaded in response to a file request

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Does Not Teach Creating a Representation of an Uploaded File at the First Node



(EX1104 Fig. 12)

(EX2006; EX2005, ¶127.) As shown above, the window shown in *Garcia* Figure 12 has "email" selected and states "Anyone with the link can view files. You are notified how many times the folder is downloaded." (EX2006; EX2005, ¶127.) At the bottom, there is a button stating "Send Link." (EX2006; EX2005, ¶127.) Thus, *Garcia* discloses that the recipient of the shared folder receives a link and downloads the ***contents*** of the folder (*i.e.*, the files) when the link is accessed. (EX2005, ¶127.) It does ***not*** disclose the creation of a representation of the folder or any representation of the files at the receiving node. (*Id.*) For this reason,

(POR at 60-61)

Garcia Does Not Teach Creating a Representation of an Uploaded File at the First Node

b. *Garcia* Does Not Teach the Claimed Creation of a Representation

Contrary to Petitioner's statement, PO does dispute Petitioner's characterization of *Garcia* for the grounds in this proceeding. Petitioner has also contradicted itself in the co-pending -286 IPR. There, it admitted *Garcia* uses links to share folders that allow recipients to receive file contents without having an account (or virtual drive). (IPR2024-00286, Pet., 31; EX1002, ¶92.) That is consistent with PO's arguments in the POR and *Garcia's* other disclosures that "[w]hen the user clicks on the send link button 1220, a link to the folder 1115 is sent to the contact. When the contact clicks on the link to folder 1115, the contents of the folder are shared with the contact." (EX1104, 11:26-31; *see also* 11:16-19; Fig. 12.) It is also consistent with contemporaneous evidence that PO provided regarding

Sur-Reply at 23

Garcia relates to "a system, method, and computer program . . . for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server." (EX1104, Abstract, 1:12–15, 1:39–42; EX2005, ¶51.) *Garcia* further describes that a client can share a folder with another client by generating and sending an email with a link to and identifying the shared folder, instead of an attachment. (EX1104, 10:32–34, 10:38–40, 11:39–43, Figs. 11–12; EX2005, ¶51.) While *Garcia* does not describe the process of a recipient receiving such an email or accessing a shared folder in much detail, it does state that "in one embodiment, when a user drags a folder to a particular contact, ***the folder*** or ***link to the folder*** is automatically sent (via email, text, or other means) to the contact," (EX1104, 11:16–19)[1], and "when the contact clicks on the link to folder 115, the ***contents*** of the folder are shared with the contact," (EX1104, 11:29–31). (*See also* EX1104, 10:32–43 (web link to download);

POR at 11

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Does Not Teach Creating a Representation of an Uploaded File at the First Node

Garcia relates to "a system, method, and computer program . . . for enabling a user to synchronize, manage, and share files and folders across a plurality of client devices and a synchronization server." (EX1104, Abstract, 1:12–15, 1:39–42; EX2005, ¶51.) *Garcia* further describes that a client can share a folder with another client by generating and sending an email with a link to and identifying the shared folder, instead of an attachment. (EX1104, 10:32–34, 10:38–40, 11:39–43, Figs. 11–12; EX2005, ¶51.) While *Garcia* does not describe the process of a recipient receiving such an email or accessing a shared folder in much detail, it does state that "in one embodiment, when a user drags a folder to a particular contact, ***the folder*** or ***link to the folder*** is automatically sent (via email, text, or other means) to the contact," (EX1104, 11:16–19)[1], and "when the contact clicks on the link to folder 115, the ***contents*** of the folder are shared with the contact," (EX1104, 11:29–31). (*See also* EX1104, 10:32–43 (web link to download);

POR at 11

FIG. **9** illustrates the Web link tab **335**. The Web link tab **335** includes a list of folders or files that have been shared as a link with other users. In this example, a list of files is shown, which is sorted by the date the link was enabled 905. The list may be sorted using other criteria as well. Each file has an icon **910**, which indicates the number of times the file has been downloaded, and an icon **915**, which places the link onto the clipboard to be copied into an email, text, or other transmittal. One benefit of sending links to folders or files to other users is it allows other users to access the folders or files without requiring the other user to create an account in the synchronization system.

EX1104 at 10:32-43

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Garcia Does Not Teach Creating a Representation of an Uploaded File at the First Node

Petitioner's only argument is that "*Garcia* explicitly discloses that, 'when a user double clicks on a file or item in a remote folder in the virtual drive, the server downloads its copy of the file/item to the local synchronization software client (i.e., the client currently being used by the user).'" (Rep., 26 (citing EX1104, 5:35-39; see also Pet., 10; EX1102, ¶61. EX1127, ¶¶117-118). But that *general* statement, where a user already has a file in their virtual drive, does not say anything about what occurs for it to get there in the first place. Nor does it show the creation of a representation of the uploaded file at the first node, as the claims all require.

Indeed, *Garcia* does not state that a recipient must navigate through a virtual drive, or that a recipient would be able to view a file in a virtual drive without accessing the link. *Garcia* discloses that when the link is accessed contents are downloaded locally at the device being used, and at most, contents would show up as remotely stored objects if the user accessed its virtual drive from a ***different device.*** Petitioner relies heavily on *Garcia*'s virtual drive but it does not connect a disclosure in *Garcia* involving the receipt of the link to the content and the creation of a "representation" at the device being used when the link is accessed. Instead, *Garcia* indicates the contents are downloaded locally at the device used to access the link.

Sur-Reply at 23-24

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Claims 11-13, 16, and 22-24

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Grounds 2, 4, 5 and 6 Fail For Same Reasons as Grounds 1 and 3

a. Claims 11–13 and 16

Petitioner purports to rely on *Manzano* only for elements of dependent claims 11–13 and 16, which rely on either independent claim 9 (claims 11–13) or independent claim 14 (claim 16) (Pet. at 49.) Likewise, Petitioner purports to rely on *Garcia* only for claim elements of dependent claims 11 and 16. (Pet. at 71–72.) Petitioner does not rely on either *Manzano* or *Garcia* for independent claims 9 or 14, which, as set forth above, are not obvious over *Riepling* alone or in combination with *Meisels*. (EX2005, ¶124.) Therefore, for the reasons set forth above with respect to Grounds 1 and 3, Petitioner has not demonstrated that claims 11–13 and 16 are rendered obvious over *Riepling* alone or in combination with *Meisels* in further view of *Manzano*, or both *Manzano* and *Garcia*. (*Id.*)

b. Claims 22–24

Petitioner purports to rely on *Manzano* and *Garcia* only for elements 22[h], 22[j], and 22[k]. (Pet. at 59–64, 71–72.) Otherwise, Petitioner relies on *Riepling* and Petitioner's arguments regarding independent claim 9 for independent claim 22. (Pet. at 59–64.) For example, with respect to claim element 22[a], Petitioner relies on its arguments for claim elements 9[a] and 9[b]. (Pet. at 59.) With respect to claim element 22[e], Petitioner relies on its arguments for claim elements 9[g] and 9[h]. (Pet. at 60.) With respect to claim element 22[i], Petitioner relies on its arguments for claim elements 9[l], 9[n], and other elements that also depend on the arguments for claim elements 9[l] and 9[n]. (Pet. at 63.) For at least the reasons described above with respect to Grounds 1 and 3, *Riepling* does not teach or suggest all of these elements of claim 9, including for example, the "file description" aspects for element 9[b], 9[g], and/or 9[l] described above and neither *Meisels*, *Manzano* nor *Garcia* cure those deficiencies. (EX2005, ¶125.) Therefore, Grounds 2, 4, 5 and 6 fail for the same reasons. (*Id.*)

POR at 57-58

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

End of Slides

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE