

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of: Hsin-Tai Wu
U.S. Patent No.: 9,727,655 Attorney Docket No.: 52472-0009IP1
Issue Date: Aug. 8, 2017
Appl. Serial No.: 14/616,478
Filing Date: February 6, 2015
Title: SEARCHING SYSTEM, METHOD, AND P2P DEVICE FOR
P2P DEVICE COMMUNITY

Mail Stop Patent Board

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

**PETITION FOR *INTER PARTES* REVIEW OF UNITED STATES PATENT
NO. 9,727,655 PURSUANT TO 35 U.S.C. §§311–319, 37 C.F.R. §42**

TABLE OF CONTENTS

I.	REQUIREMENTS FOR IPR—37 C.F.R. §42.104.....	1
A.	Grounds for Standing—37 C.F.R. §42.104(a).....	1
B.	Challenge and Relief Requested—37 C.F.R. §42.104(b)	1
II.	SUMMARY OF THE '655 PATENT	2
A.	Brief Description.....	2
B.	Prosecution of the '655 Patent	4
III.	LEVEL OF ORDINARY SKILL	4
IV.	CLAIM CONSTRUCTION	5
V.	DETAILED EXPLANATION OF GROUNDS.....	6
A.	GROUND 1A: Schwan anticipates Claims 1-4, 7-9, 11-13, 16, 18, and 19	6
1.	<i>Schwan</i>	6
2.	<i>Application to Challenged Claims</i>	8
B.	GROUND 1B: – Schwan in view of Lee would have rendered obvious claim 1-4, 7-13, 16-19.....	28
1.	<i>Lee</i>	28
2.	<i>The Schwan-Lee Combination</i>	29
3.	<i>Application to Challenged Claims</i>	32
C.	GROUND 1C: Schwan, in view of Lee, and further in view of Strickland would have rendered obvious claims 5, 6, 14, 15	39
1.	<i>Strickland</i>	39
2.	<i>The Schwan-Lee-Strickland Combination</i>	40
3.	<i>Application to Challenged Claims</i>	42
D.	GROUND 2A: Li anticipates claims 1, 7-9, 11, 16, 18, 19.....	45
1.	<i>Li</i>	45
2.	<i>Application to Challenged Claims</i>	47
E.	GROUND 2B: Li in view of Rose would have rendered obvious claims 1, 7-11, and 16-19	64
1.	<i>Rose</i>	64
2.	<i>The Li-Rose Combination</i>	65
3.	<i>Application to Challenged Claims</i>	68
F.	GROUND 2C: Li, in view of Rose, and further in view of Schwan would have rendered obvious claims 2-4, 12, and 13	75
1.	<i>The Li-Rose-Schwan Combination</i>	75
2.	<i>Application to Challenged Claims</i>	77

G.	GROUND 2D: Li, in view of Rose, and further in view of Strickland would have rendered obvious claims 5, 6, 14, and 15.....	79
1.	<i>The Li-Rose-Strickland Combination</i>	80
2.	<i>Application to Challenged Claims</i>	80
VI.	PAYMENT OF FEES – 37 C.F.R. §42.103.....	80
VII.	MANDATORY NOTICES UNDER 37 C.F.R §42.8(a)(1).....	80
A.	Real Party-In-Interest Under 37 C.F.R. §42.8(b)(1).....	80
B.	Related Matters Under 37 C.F.R. §42.8(b)(2).....	81
C.	Lead And Back-Up Counsel Under 37 C.F.R. §42.8(b)(3).....	81
D.	Service Information	81

LIST OF EXHIBITS

EX1001	U.S. Patent No. 9,727,655 to Wu (“the ’655 Patent”)
EX1002	Prosecution History of the ’655 Patent (“the Prosecution History”)
EX1003	Declaration and Curriculum Vitae of David Lett
EX1004	EP Publication No. 2,086,206 (“Schwan”)
EX1005	U.S. Publication No. 2011/0087775 (“Lee”)
EX1006	U.S. Publication No. 2007/0226368 (“Strickland”)
EX1007	CN Publication No. 101018172 (“Li”)
EX1008	U.S. Publication No. 2005/0050028 (“Rose”)
EX1009	RESERVED
EX1010	An Extensible Set-Top Box Platform for Home Media Applications (2005), <i>available at</i> https://www.researchgate.net/profile/Trevor_Smith9/publication/3181282_An_extensible_set-top_box_platform_for_home_media_applications/links/54e7d6a40cf25ba91c7a2012.pdf
EX1011	Integration of Mobile Devices into Popular Peer-to-Peer Networks (2009), <i>available at</i> https://ieeexplore.ieee.org/document/5175763
EX1012	U.S. Publication No. 2010/0094877 (“Garbe”)
EX1013	U.S. Publication No. 2009/0100128 (“Czechowski”)
EX1014	U.S. Publication No. 2003/0208621 (“Bowman”)

EX1015	U.S. Patent No. 8,332,482 (“Alexander”)
EX1016	U.S. Publication No. 2006/0218274 (“Labio”)
EX1017	U.S. Publication No. 2013/0054634 (“Chakraborty”)
EX1018	Complaint, <i>ThroughTEK Co. Ltd. v. Reolink Innovation Inc.</i> , Case No. 1:24-cv-01027 (D. Del.), Sep. 12, 2024

LISTING OF CLAIMS

Claim Element	Language
[1pre]	A searching system for a P2P (peer-to-peer) device community, for receiving a search request sent from at least one user's device, the searching system comprising:
[1a]	at least one P2P device community, wherein each the P2P device community includes at least one P2P device, wherein each the P2P device respectively comprises at least one file data, and each the file data comprises a content data and an indexed data;
[1b]	at least one online server connecting to the P2P device community, each the online server includes a topology data table, the topology data table being recorded with a link data between each of the P2P devices of the P2P device community; and
[1c]	at least one content management server connecting to the P2P device community and the online server respectively, each the content management server includes a search engine, an index database, and a user's device management unit,
[1d]	wherein the index database records the indexed data of each the P2P device of the P2P device community, the user's device management unit receives the search request sent from the user's device, the search engine searches the index database to generate a search result based on the search request and then returning the search result to the user's device,
[1e]	wherein the user's device obtains the link data of the P2P device within the P2P device community according to the topology data table of the online server and the search result.
[2]	The searching system of claim 1, wherein each the P2P device includes a weighted data, and the content management server is provided with a sorting algorithm module creating the weighted data based on the link data associated with the P2P device, the search result being sorted by the order of the weighted data of the P2P device so as to generate a sorted search result.

Claim Element	Language
[3]	The searching system of claim 2, further comprising: a weight look-up table recorded with the weighted data of each the P2P device, wherein the weight look-up table is stored in the content management server, the online server, or a storage device.
[4]	The searching system of claim 2, wherein the sorting algorithm module performs a function operation based on the search result and the weighted data corresponding to the P2P device to generate a weighted search result, and the weighted search result is sorted by the order of the weighted search results so as to generate a weighted and sorted search result.
[5]	The searching system of claim 1, wherein the user's device management unit includes a user-charge module, and wherein after confirming that the user's device has been charged, the content management server provides the search result, or the online server provides at least one of the link data of the P2P device, to the user's device.
[6]	The searching system of claim 5, wherein the user charge module receives a user's identity data, a charging data, a charging-mode data, or a payment confirmation data, all of which being sent from the user's device.
[7]	The searching system of claim 1, wherein the indexed data can be created by the P2P devices themselves, or the indexed data can be created by the content management server gathering from the P2P devices via a network gathering module.
[8]	The searching system of claim 1, wherein the user's device connects to the P2P device according to the link data and downloads the file data based on the link data.
[9]	The searching system of claim 1, wherein the user's device connects to the P2P device based on the link data.
[10]	A P2P device applicable to the searching system of claim 1, comprising: a control unit controlling the file data being an open state or a hidden state.

Claim Element	Language
[11pre]	A searching method for a P2P device community, applicable to a searching system for a P2P device community, the searching system includes at least one P2P device, an online server, and a content management server, the searching method comprising the steps of:
[11a]	recording a link data of each the P2P device to generate a topology data table and storing the topology data table in the online server, wherein each the P2P devices respectively comprises at least one file data, and each the file data comprises a content data and an indexed data;
[11b]	recording the indexed data of each the P2P device and storing the indexed data in an index database of the content management server;
[11c]	receiving by the content management server a search request sent from a user's device and generating a search result by searching the index database based on the search request;
[11d]	returning the search result to the user's device; and
[11e]	obtaining the link data of the P2P device within the P2P device community by the user's device according to the topology data table of the online server and the search result.
[12]	The searching method of claim 11, further comprising the steps of: creating a weighted data associated with each the P2P device by the content management unit based on the link data of the P2P devices; performing a function operation based on the search result and the weighted data corresponding to the P2P device to generate a weighted search result; and sorting the weighted search result based on the order of the weighted search results to generate a weighed and sorted search result.
[13]	The searching method of claim 12, further comprising the steps of: recording the weighted data of the P2P devices to generate a weight look-up table; and storing the weight look-up table in the content management server, the online server, or a storage device.

Claim Element	Language
[14]	The searching method of claim 11, further comprising the steps of: providing the search result by the content management server, or providing at least one link data of the P2P device by the online server, to the user's device by the content management server confirming that the user's device has been charged.
[15]	The searching method of claim 14, wherein the content management server confirming that the user's device has been charged comprises the steps of: confirming a user's identity data of the user's device; receiving a charge-mode data sent from the user's device; confirming a payment confirmation data sent from the user's device; and recording a pricing data required for connection or down loading requested by the user's device.
[16]	The searching method of claim 11, wherein the indexed data can be created by the P2P devices themselves, or the indexed data can be created by the content management server gathering from the P2P devices via a network gathering module.
[17]	The searching method of claim 11, further comprising the step of: controlling a file data of each the P2P device being an open state or a hidden state.
[18]	The searching method of claim 17, further comprising the steps of: connecting to the P2P device and downloading the file data by the user's device based on the link data.
[19]	The searching method of claim 11, further comprising the steps of: connecting to the P2P device by the user's device based on the link data.

Reolink Innovation Inc. and Shenzhen Reolink Technology Co., Ltd.

(collectively, “Petitioners” or “Reolink”) petition for *Inter Partes* Review (“IPR”) of claims 1-19 of U.S. Patent 9,727,655 (“the ’655 Patent”).

I. REQUIREMENTS FOR IPR—37 C.F.R. §42.104

A. Grounds for Standing—37 C.F.R. §42.104(a)

Petitioners certify that the ’655 Patent is available for IPR, and Petitioners are not barred or estopped from requesting IPR.

B. Challenge and Relief Requested—37 C.F.R. §42.104(b)

Petitioners request IPR on grounds 1A-2D listed below. An expert declaration from David Lett supports each ground. *See* EX1003.

Ground	Claims	Basis for Rejection
1A	1-4, 7-9, 11-13, 16, 18, 19	§102 – Anticipated by Schwan (EX1004)
1B	1-4, 7-13, 16-19	§103 – Obvious based on Schwan in view of Lee (EX1005)
1C	5, 6, 14, 15	§103 – Obvious based on Schwan, in view of Lee, and further in view of Strickland (EX1006)
2A	1, 7-9, 11, 16, 18, 19	§102 – Anticipated by Li (EX1007)
2B	1, 7-11, 16-19	§103 – Obvious based on Li in view of Rose (EX1008)
2C	2-4, 12, 13	§103 – Obvious based on Li, in view of Rose, and further in view of Schwan
2D	5, 6, 14, 15	§103 – Obvious based on Li, in view of Rose, and further in view of Strickland

Grounds 1A-2D rely on publications that all qualify as prior art under 35 U.S.C. §102 as shown below:

Reference	Prior Art Status
Schwan	§102(a)(1) – published May 8, 2009
Lee	§102(a)(1) – published Apr. 14, 2011
Strickland	§102(a)(1) – published Sep. 27, 2007
Li	§102(a)(1) – published Aug. 15, 2007
Rose	§102(a)(1) – published Mar. 3, 2005

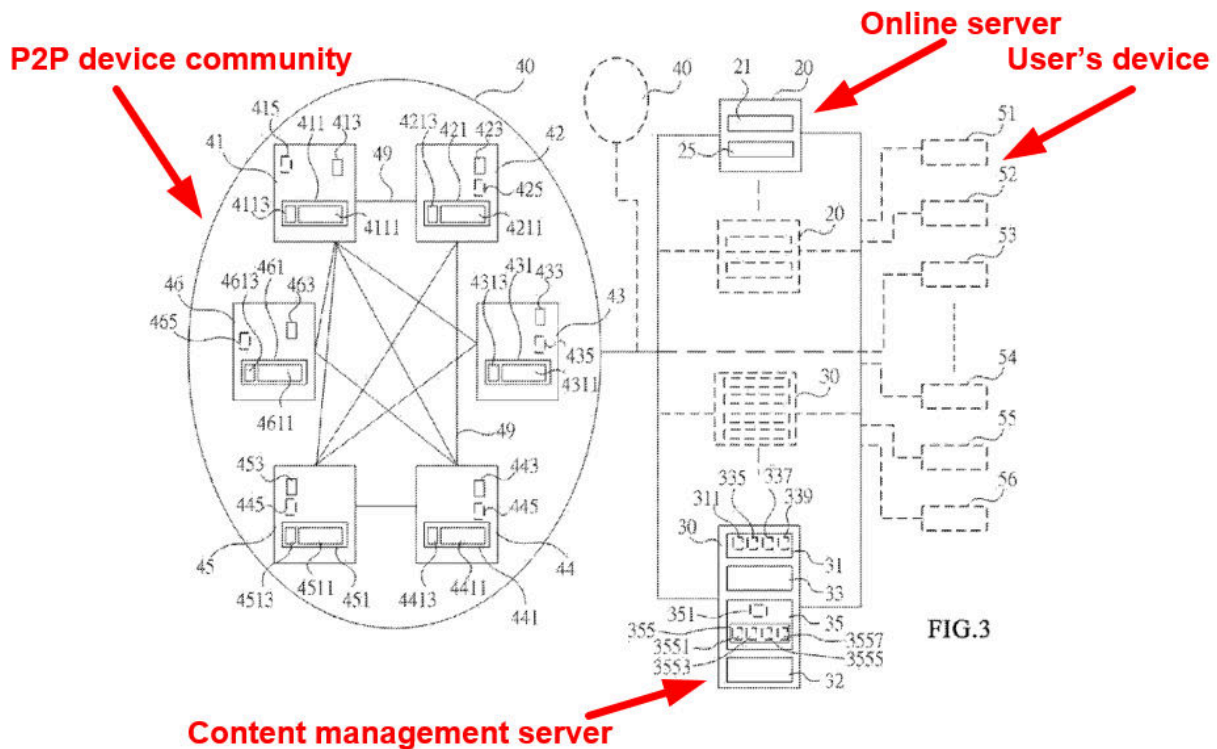
Here, all of the prior art publications in Grounds 1A-2D were published before the Oct. 3, 2014 filing date of the '655 Patent, thereby qualifying as prior art under §102(a)(1).

II. SUMMARY OF THE '655 PATENT

A. Brief Description

The '655 Patent describes a peer-to-peer (P2P) system for managing content downloads from a P2P device community. EX1001, 1:56-3:60, 4:26-6:40, FIG. 2, FIG. 3; EX1003, ¶37. The P2P device community includes a network of P2P devices connected to each other through a web of links. EX1001, 4:35-5:22. An online server 20 of the P2P system maintains a topology data table 21 that records link data indicating connection relationships between P2P devices. EX1001, 4:51-5:22, 7:11-27, 8:41-61, FIG. 3. The P2P devices can store content that is available for download by other devices. EX1001, 4:35-67, 5:38-6:16, FIG. 2, FIG. 3. This

content is indexed in an index database 33 of a content management server 30, which can be different from the online server 20. EX1001, 4:58-5:34, 6:17-40, FIG. 2, FIG. 3.



EX1001, FIG. 3 (annotated)

The '655 Patent explains that content management server 30 receives a search request from a user's device, the search request indicating content for download. EX1001, 1:60-2:13, 5:1-22, 6:17-40, FIG. 2, FIG. 3; EX1003, ¶38. The content management server 30 searches the index database 33 based on the search request to generate a search result indicating one or more P2P devices storing the requested content. EX1001, 5:50-6:52, FIG. 3. The content management server 30

can use the topology data table 21 stored by the online server 20, for example, to sort the entries of the search result. EX1001, 5:50-6:52, FIG. 3.

B. Prosecution of the '655 Patent

The Examiner issued just one office action during original examination of the '655 Patent. EX1002, 80-92; EX1003, ¶39. The applicant responded with: (i) an amendment clarifying that each P2P device includes “at least one file data” comprising “a content data and an indexed data,” and (ii) a recitation of intended use of “the user’s device” for interacting with the claimed “searching system.” EX1002, 114-133 (adding the recitation of intended use where the user’s device “obtains the link data of the P2P device...according to the topology data table of the online server and the search result”). From there, the Examiner allowed the claims. EX1002, 151-153. As described in further detail below (*Infra*, §V), the claims were improvidently granted because more pertinent prior art was not disclosed to the Examiner. Review of the '655 patent is proper and just.

III. LEVEL OF ORDINARY SKILL

A POSITA in the field of the '655 Patent would have had a bachelor’s degree in electrical engineering, computer engineering, computer science, or equivalent, and two years of work experience with network devices and communication technologies. EX1003, ¶¶17-19. Additional education might compensate for less experience, and vice versa. EX1003, ¶¶17-19.

IV. CLAIM CONSTRUCTION

All claim terms should be construed according to the *Phillips* standard.

Phillips v. AWH Corp., 415 F.3d 1303 (Fed. Cir. 2005); 37 C.F.R. §42.100. When the specification uses “lexicography” to give a special definition to a claim term that differs from the meaning it would otherwise possess, this lexicography governs. *Phillips* at 1316. Here, the ’655 Patent appears to provide such lexicography for the term “link data.” EX1001, 4:53-57 (“link data 49 refers to...”). As shown below, this lexicography for “link data” does not impact Grounds 1A-2D below because the analysis and prior art mapping therein align with the preferred embodiment of the ’655 patent and apply regardless of whether applicant’s lexicography for “link data” adopted. *Infra*, §V; EX1003, ¶¶34-36.

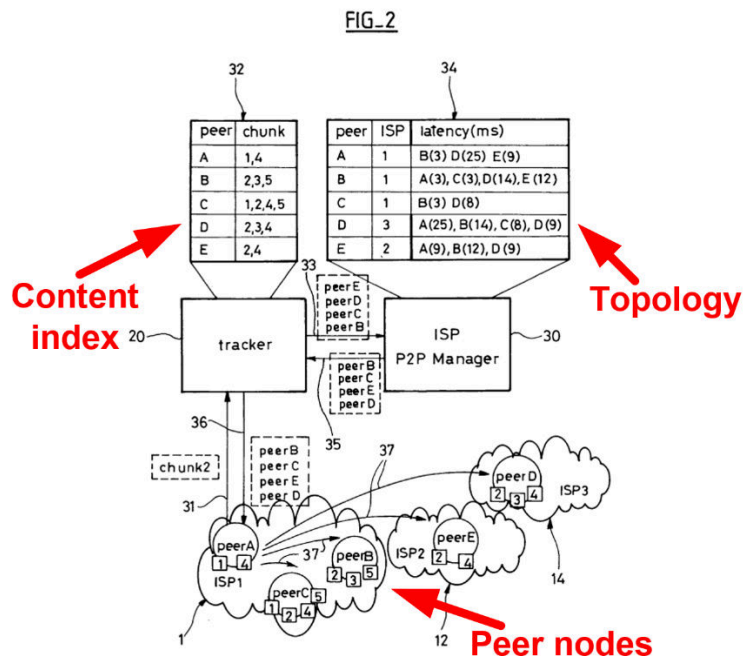
From there, given the noticeable overlap between the preferred embodiment of the ’655 Patent and the prior art/predictable combinations articulated in Grounds 1A-2D. Petitioners submit that no other claim elements need a formal construction for purposes of resolving controversies in this proceeding. *Wellman, Inc. v. Eastman Chem. Co.*, 642 F.3d 1355, 1361 (Fed. Cir. 2011); *Google Inc. v. Intellectual Ventures II LLC*, 701 Fed. Appx. 946, 956 (Fed. Cir. 2017) (“do not require construction” for purposes of “resolv[ing] the parties’ patentability arguments”); EX1003, ¶¶34-36.

V. DETAILED EXPLANATION OF GROUNDS

A. GROUND 1A: Schwan anticipates Claims 1-4, 7-9, 11-13, 16, 18, and 19

1. *Schwan*

Similar to the '655 Patent, Schwan describes a P2P system that tracks P2P devices within a P2P device community and facilitates content downloads from these P2P devices. EX1004, [0002]-[0006], [0023]-[0039], FIG. 1, FIG. 2; EX1003, ¶40. For example, Schwan's P2P system includes a plurality of "peer nodes A to E" that "form a peer-to-peer network." EX1004, [0024]-[0030], FIG. 1. Each of these peer nodes store content. EX1004, [0025], [0029], [0030], FIG. 1, FIG. 2. This content is indexed by a tracker node 20 of Schwan's P2P system in a table 32 that identifies content stored by each peer node.



EX1004, FIG. 2 (annotated).

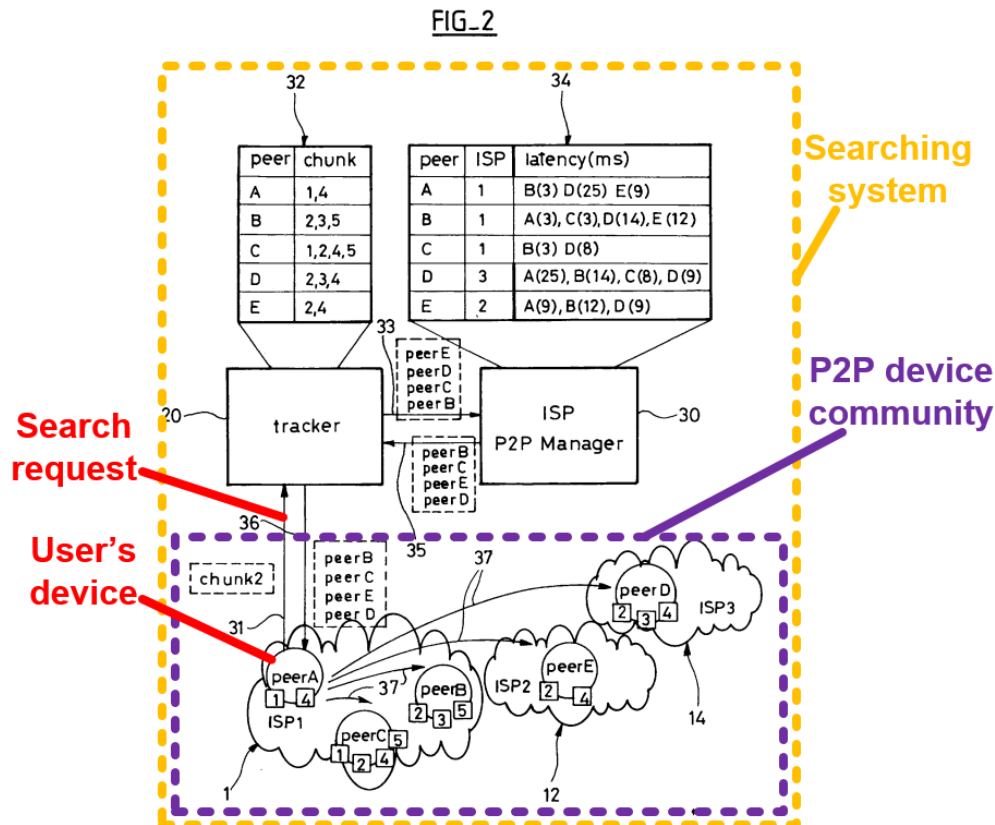
Schwan also includes a peer-to-peer manager 30 that stores a table 34 providing a topology of the P2P network. EX1004, [0034]; EX1003, ¶41. This table 34 indicates, for each of peer nodes A-E, a latency between the peer node and each other peer node and an internet service provider (ISP) corresponding to each peer node. *Id.* Together, tracker node 20 and peer-to-peer manager 30 process requests for resources stored by the peer nodes. EX1004, [0026], [0029], [0030], [0032], [0033], FIG. 2.

For example, Schwan explains that “peer node A sends a request to the tracker for a peer list of active peers holding the resource ‘chunck 2.’” EX1004, [0029]. Tracker node 20 receives this request, “generates an initial list of peer nodes that hold the requested resource” that is “stored in...a table 32,” and “sends this initial list to the peer-to-peer manager 30.” EX1004, [0029], [0030]. Peer-to-peer manager 30 “uses...topology information and specific rules to rank the peer nodes of the initial list” and “returns the ranked list to the tracker node 20.” *Id.* Tracker node 20 “returns the ranked peer list to the requesting peer node A.” EX1004, [0032]. When peer node A receives the ranked list of peer nodes, “[t]hen peer node A can contact the peer nodes accordingly.” EX1004, [0032], FIG. 2. Much like the ’655 Patent, therefore, Schwan discloses techniques for performing a content index search using an index database and a topology data table. EX1004, [0028]-[0032], FIG. 2; *cf.* EX1001, 5:38-7:64, FIG. 3; EX1003, ¶42.

2. Application to Challenged Claims

Element [1pre]

To the extent the preamble is limiting, Schwan discloses the claimed *searching system*. EX1003, ¶69. Schwan discloses a *P2P device community* including “peer nodes A to E” that “form a peer-to-peer network.” EX1004, [0024], [0025], *see also id.*, [0007]-[0021], [0023], [0026], [0028]-[0033], FIG. 1, FIG. 2. Schwan also discloses a *searching system* including a tracker node 20, a peer-to-peer manager 30, and the *P2P device community*. EX1004, [0027]-[0039], FIG. 2.



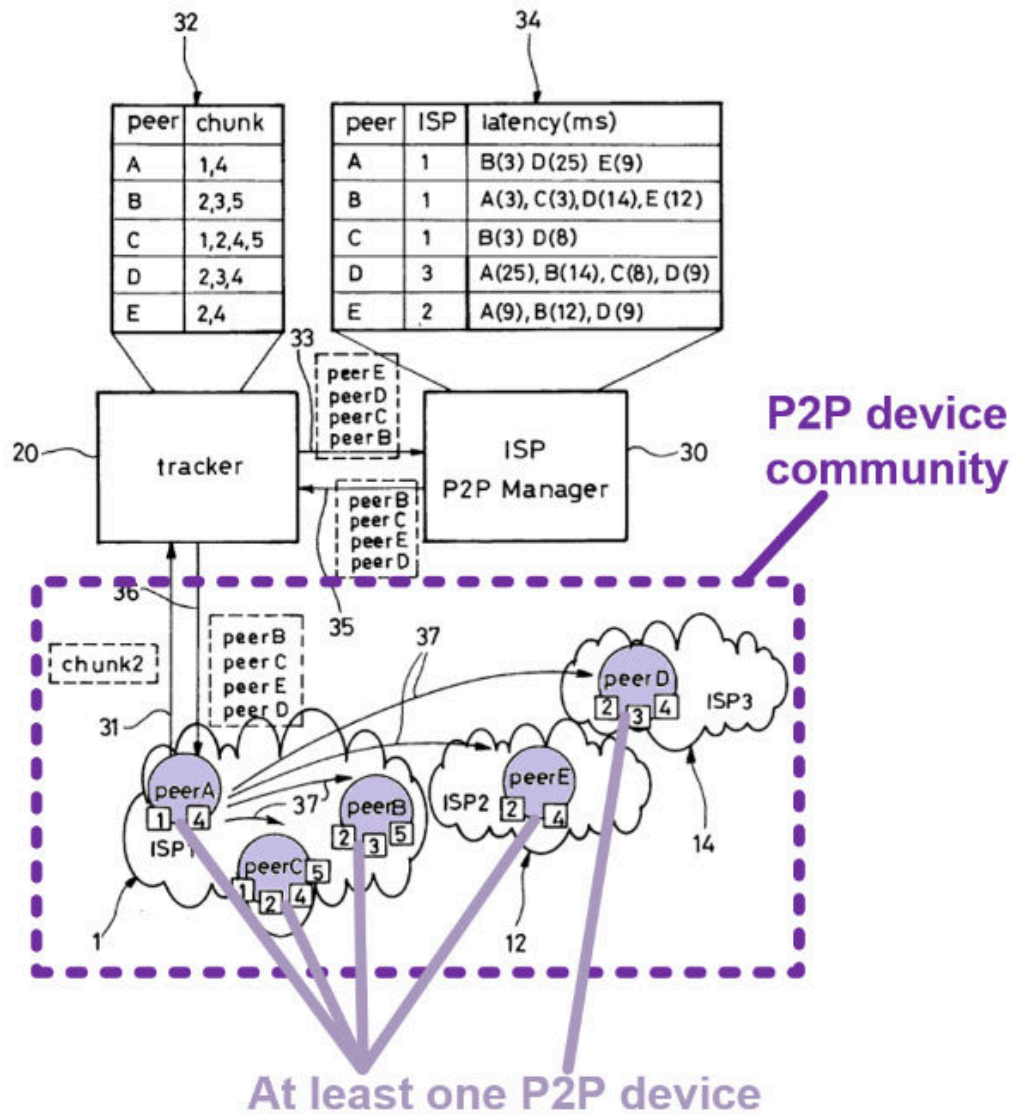
EX1004, FIG. 2 (annotated).

Schwan explains that the tracker node 20 receives a *search request* from peer node A. EX1004, [0029] (“peer node A sends a request to the tracker”), [0030] (“[u]pon receiving the request 31”). Peer node A represents *at least one user’s device*, because Schwan describes peer nodes as being user devices for downloading content. EX1004, [0004] (“[a] user that wants to download a file from a BitTorrent P2P network”), [0024] (describing network devices that host peer nodes as including “cellular radiotelephones, personal computers, personal digital assistants, set-top boxes, etc.”); EX1003, ¶70.

Element [1a]

Schwan describes a *P2P device community* including “peer nodes A to E” that “form a peer-to-peer network.” EX1004, [0026], *see also id.*, [0023]-[0025], [0029]-[0033], FIG. 1, FIG. 2; *supra*, Element [1pre]. Peer nodes A to E represent *at least one P2P device* because each node hosts content for sharing with other devices. EX1004, [0025], *see also id.*, [0024], [0026], [0028]-[0032], FIG. 1, FIG. 2. Furthermore, Schwan explains that peer nodes are common P2P devices such as “cellular radiotelephones, personal computers, personal digital assistants, set-top boxes.” EX1004, [0024]; EX1003, ¶71 (citing corroborating, EX1010, 1, EX1011).

FIG. 2



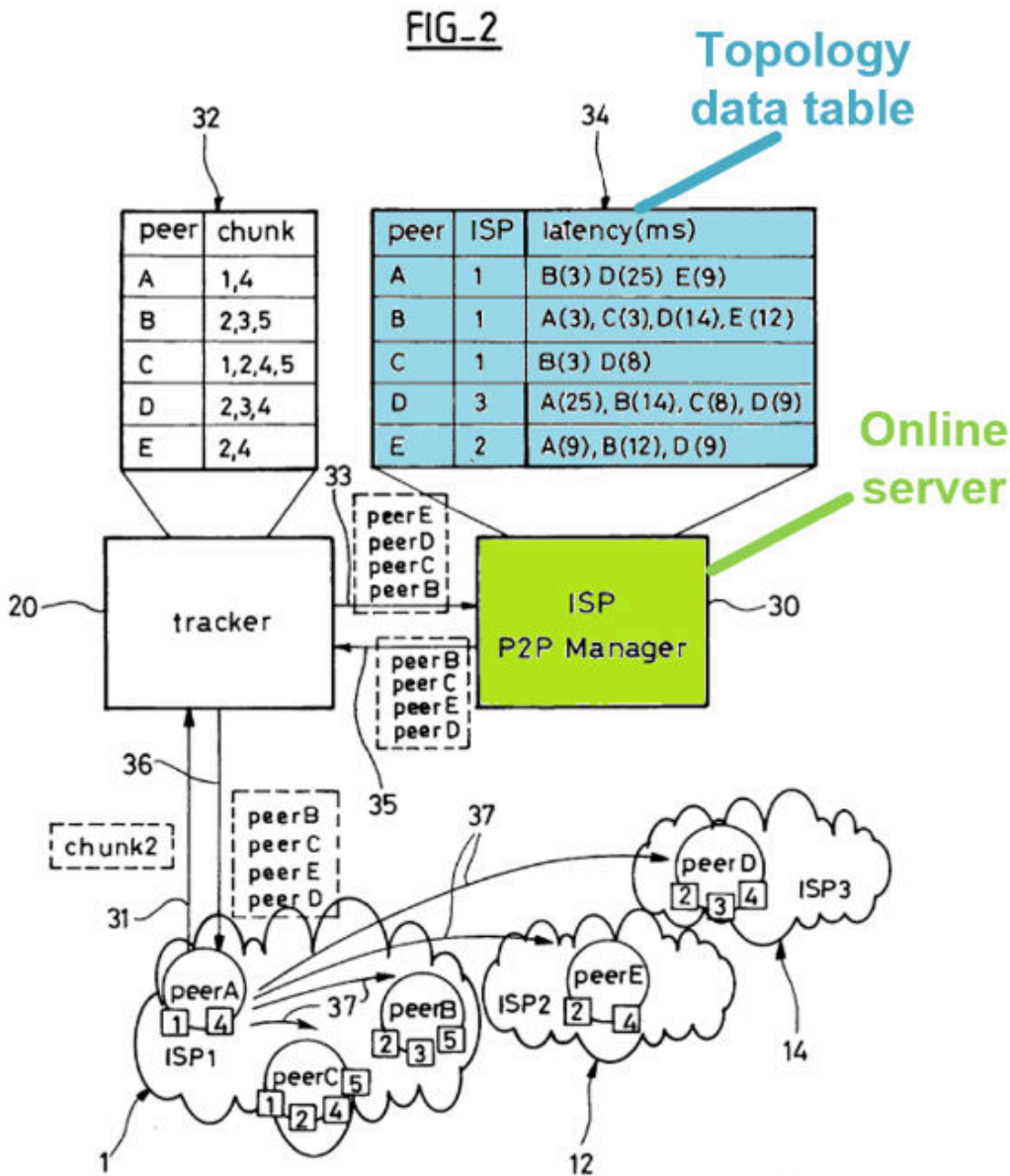
EX1004, FIG. 2 (annotated).

Because Schwan expressly discloses that “[e]ach of the peer nodes A to E may hold resources such as applications, services or files that another node may wish to use or retrieve,” the *at least one P2P device* stores *file data* including *content data*. EX1004, [0024], *see also id.*, [0012] (“peer nodes holding the given

resource”), [0014], [0017], [0019], [0029], [0030]. The **file data** stored by each **P2P device** also comprises **indexed data** such as a file name. EX1004, [0029] (“file called ‘chunck 2’”), *see also id.*, [0024]-[0028], [0030], FIG. 2; EX1003, ¶72.

Element [1b]

The P2P system of Schwan includes a peer-to-peer manager 30 that provides an **online server**. EX1004, [0027], [0029]-[0031], FIG. 2; EX1003, ¶73. Schwan’s peer-to-peer manager 30 is **online**, because manager 30 collects information, such as the latency between each pair of peer nodes over the P2P network. EX1004, [0029], FIG. 2. Because peer-to-peer manager 30 manages information relating to multiple different peer nodes spanning multiple different service providers, peer-to-peer manager 30 represents a **server**. EX1004, [0027], [0029]-[0031], FIG. 2. The **online server** represented by peer-to-peer manager 30 **connects** to the **P2P device community** when peer node A sends a request to tracker node 20 which forwards this request to peer-to-peer manager 30 and peer-to-peer manager 30 sends a sorted search result back to peer node A. EX1004, [0028]-[0032], FIG. 2. The **online server** represented by peer-to-peer manager 30 also **connects** to the **P2P device community** to collect information such as a latency between each pair of peer nodes. EX1004, [0029]-[0031], FIG. 2.



EX1004, FIG. 2 (annotated).

The *online server* represented by peer-to-peer manager 30 includes a table 34 that stores, for each of the peer nodes, “an identification of the access network to which the peer node is connected and a distance measurement with other peer nodes.” EX1004, [0031], FIG. 2; EX1003, ¶74. Schwan explains that “[t]he access

network is identified by the name of the ISP” corresponding to the peer node and “[t]he distance measurement between two peer nodes is expressed as the latency in milliseconds.” *Id.* Specifically, the table 34 indicates the internet service provider (ISP) corresponding to each peer node of the *P2P device community* and the latency, in milliseconds, between multiple different pairs of peer nodes. EX1004, [0031], FIG. 2. This information is referred to by Schwan as “*topology* information.” EX1004, [0030], *see also id.*, [0036], [0044].

34

Topology data table

Latency between peerA and other P2P devices

peer	ISP	latency(ms)
A	1	B(3) D(25) E(9)
B	1	A(3), C(3), D(14), E (12)
C	1	B(3) D(8)
D	3	A(25), B(14), C(8), D (9)
E	2	A(9), B(12), D (9)

ISP where each peer device is located

EX1004, FIG. 2 (annotated).

The *online server* represented by peer-to-peer manager 30 therefore stores a *table* 34 including *topology data*. EX1004, [0027]-[0031], FIG. 2. The ISP corresponding to each P2P device represents *link data* between *each of the P2P*

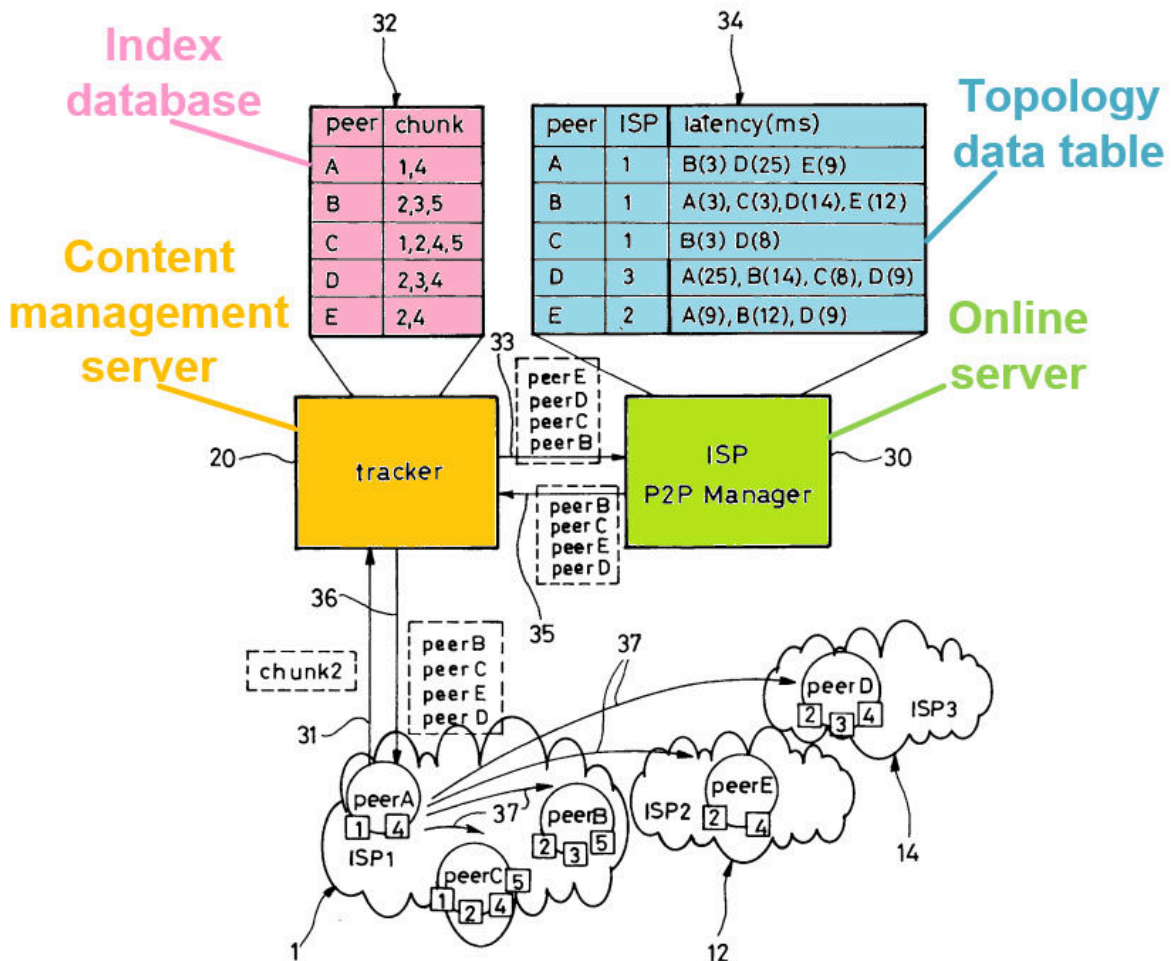
devices, because the ISP is where the P2P device connects to other P2P devices.

EX1004, [0030], [0031], FIG. 2. The amount of latency separating each pair of P2P devices also represents *link data* between *each of the P2P devices* because these latencies indicate the amount of time it takes for information to travel over a *link* between that pair of P2P devices. EX1004, [0027]-[0031], FIG. 2. The *link data* stored by Schwan's table 34 is very similar to the link data described by the '655 Patent, because in both cases the *link data* indicates connection locations and connection relationships. EX1004, [0027]-[0031], FIG. 2; *cf.* EX1001, 4:51-53; EX1003, ¶75.

Element [1c]

Schwan's peer node 20 represents a *content management server* connected to the *P2P device community* and the *online server*, because tracker node 20 stores a table 32 that indicates *content* stored by each *P2P device* (i.e., each peer node A-E) within the *P2P device community*. EX1004, [0029], [0030], FIG. ; EX1003, ¶77. Schwan expressly discloses that that tracker node 20 "maintains a list of the active peers in the peer-to-peer network and the respective resources that they hold." EX1004, [0029]. This information is stored as part of table 32 "in a data repository of the tracker node 20." *Id.*

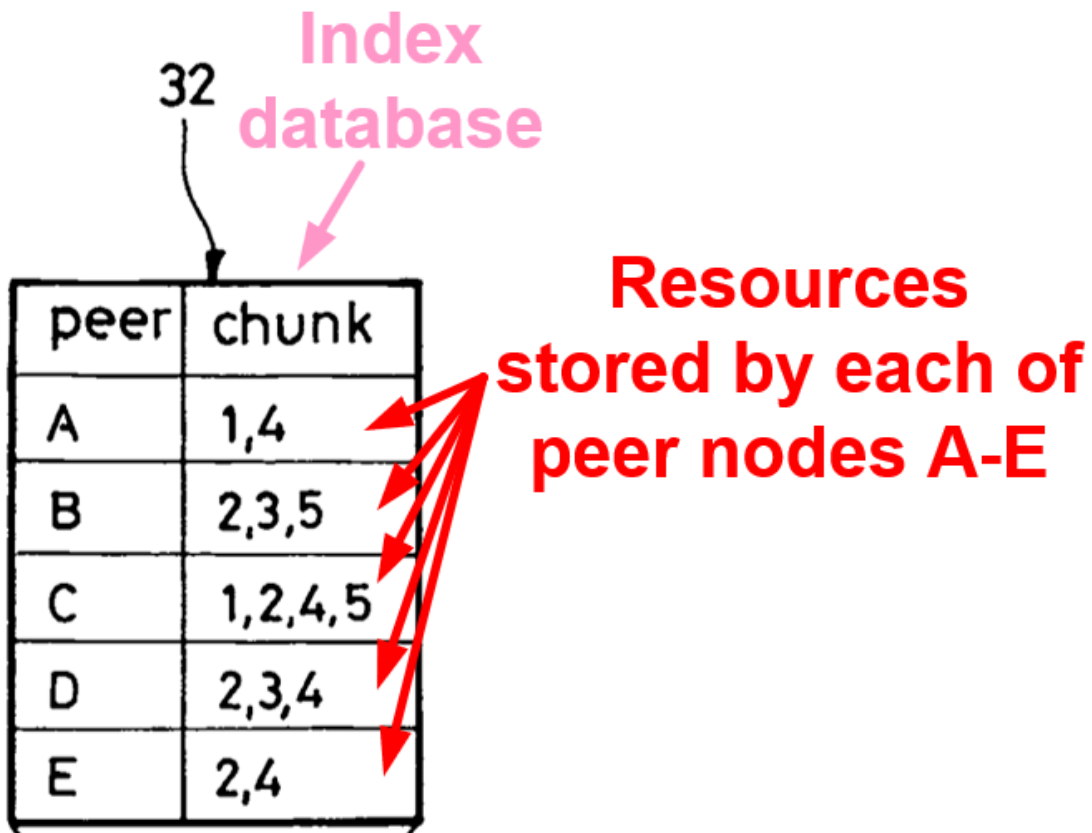
FIG. 2



EX1004, FIG. 2 (annotated).

Consequently, table 32 stored by Schwan's *content management server* represents an *index database* indicating resources stored by each of peer nodes A-E. EX1004, [0029], [0030], FIG. 2; EX1003, ¶78. Resources stored by peer nodes A-E are referred to by Schwan as "chuncks," each chunk being associated with an index number. EX1004, [0029], [0030], FIG. 2. For example, the *index database*

of Schwan's *content management server* indicates that peer nodes B-E each store "chunck2" and peer nodes B-C each store "chunck5." *Id.*



EX1004, FIG. 2 (annotated).

Schwan also explains that "peer node A sends a request to the tracker for a peer list of active peers holding the resource 'chunck 2'" and "[u]pon receiving the request...the tracker node 20 generates an initial list of peer nodes that hold the requested resource." EX1004, [0029], [0030] FIG. 2. To generate this initial list, the *content management server* represented by tracker node 20 searches the *index database* of table 32 which identifies the resources stored by each *P2P device* of

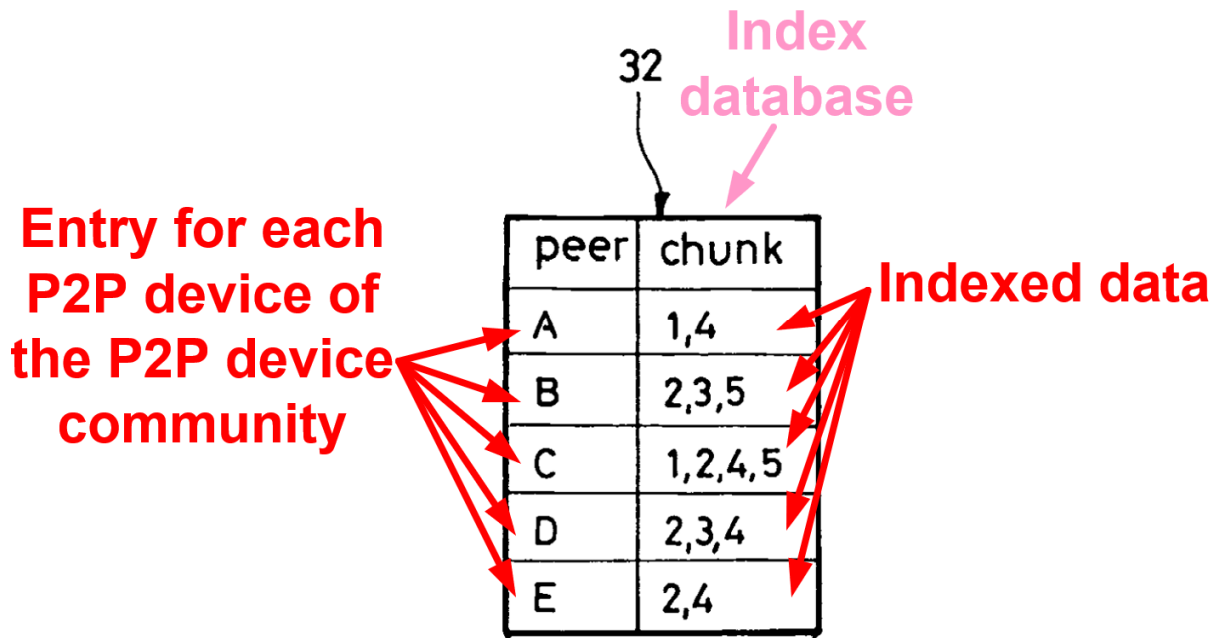
the *P2P device community*. *Id.* For example, Schwan explains that the “the initial list reads E, D, C, and B,” meaning that the initial list includes each P2P device that stores the “chunk2” resource requested by peer node A. EX1004, [0029], [0030] FIG. 2; EX1003, ¶79. This means that the *content management server* represented by tracker node 20 includes a *search engine* for searching the *index database* to identify every peer node that stores the “chunk2” resource. *Id.*

Because Schwan expressly discloses that “peer node A sends a request to the tracker for a peer list of active peers holding the resource ‘chunk 2’” and “tracker node 20 finally returns the ranked peer list to the requesting peer node A,” tracker node 20 includes a *user’s device management unit* for connecting to the user’s device represented by peer node A and managing the connection between peer node A and tracker node 20. EX1004, [0029], [0032], FIG. 2. In this aspect, tracker node 20 is strikingly similar to the content management server of the ’655 Patent, which includes a user’s device management unit that delivers search results based on a search request. EX1004, [0029], [0030], FIG. 2; *cf.* EX1001, 5:1-22, 5:50-6:5, 6:17-40; EX1003, ¶¶80-81.

Element [1d]

The *index database* stored by tracker node 20 records *indexed data* including identification numbers (e.g., “chunk 2”) for resources hosted by each

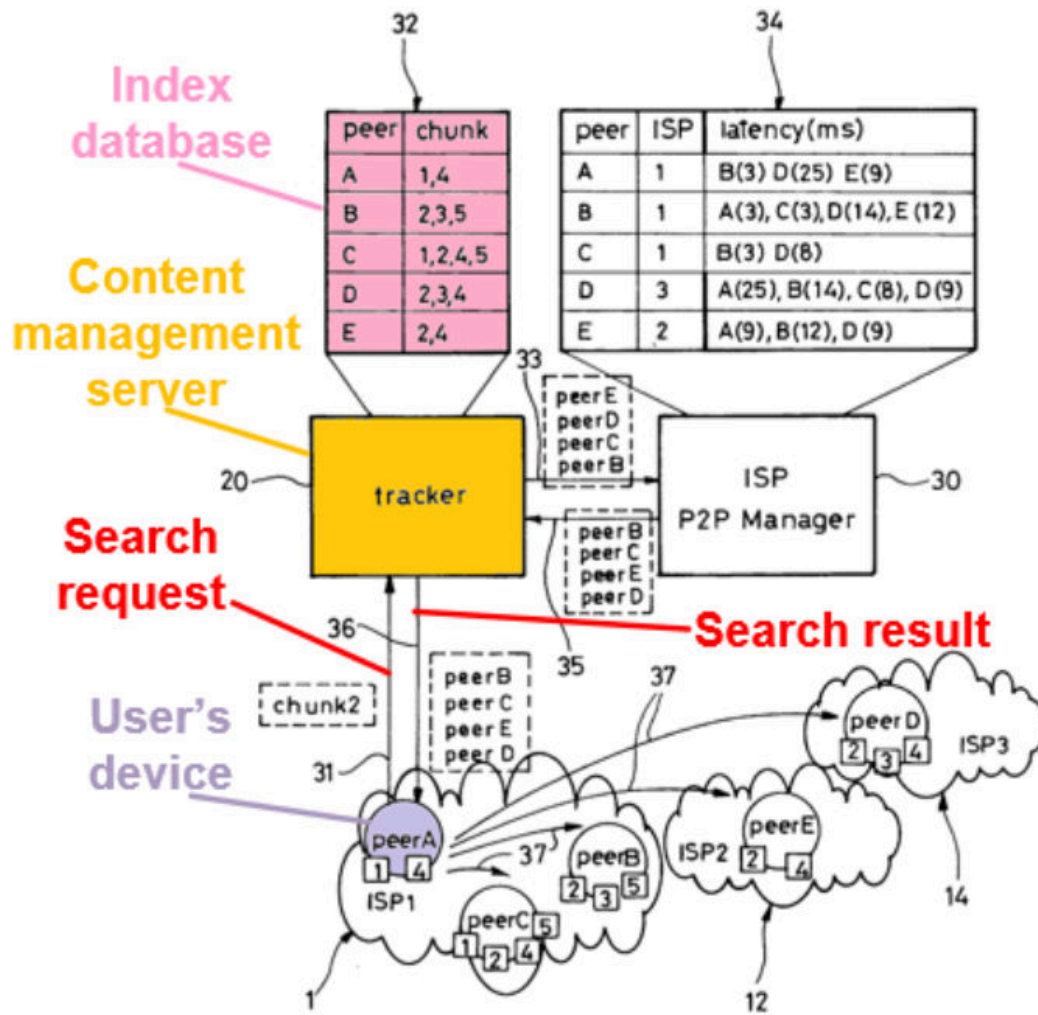
P2P device of the P2P device community including peer nodes A-E. EX1004, [0029], [0030], FIG. 2; EX1003, ¶82.



EX1004, FIG. 2 (annotated).

Schwan's tracker node 20 therefore includes a *user's device management unit* that receives a *search request* from peer node A. EX1004, [0029] ("peer node A sends a request to the tracker for a peer list of active peers holding the resource 'chunk 2'"), [0030] ("[u]pon receiving the request 31, the tracker node 20 generates an initial list of peer nodes that hold the requested resource"); *supra*, Element [1c]; EX1003, ¶¶83-84.

FIG. 2



EX1004, FIG. 2 (annotated).

A *search engine* of the *content management server* “generates an initial list of peer nodes that hold the requested resource” and “sends this initial list to the peer-to-peer manager 30.” EX1004, [0030], FIG. 2. According to one embodiment of Schwan, peer-to-peer manager 30 “returns [a] ranked list to the tracker node 20,” the ranked list representing a *search result*. *Id.* In another embodiment, the

“peer-to-peer manager 30 communicates the relevant topology information for the tracker node 20 to directly generate the optimized list in accordance with predefined rules.” EX1004, [0036], *see also id.*, [0029]-[0032], FIG. 2. In both of these embodiments, tracker node 20 “returns the ranked peer list to the requesting peer node A,” meaning that *search engine* of tracker node 20 returns the *search result* to the *user’s device*. EX1004, [0032], FIG. 2; EX1003, ¶85.

Element [1e]

To generate a *search result*, tracker node 20 generates an initial list of peer nodes by searching table 32 and sending the initial list to peer-to-peer manager 30 which ranks the initial list using “ranking criteria” that depend on the *link data* of the *topology data table* in some embodiments. EX1004, [0029]-[0032], FIG. 2; EX1003, ¶86. One of these ranking criteria involves “ranking the peer nodes as a function of the access network to which they are connected” so that peer nodes connected to the same ISP as the requesting peer node are ranked higher and peer nodes connected to other ISPs are ranked lower. EX1004, [0031], FIG. 2. Another ranking criterion involves using the latency measurements, with lower latencies resulting in a higher ranking and vice versa. *Id.* Schwan also discloses an embodiment where peer-to-peer manager 30 communicates “relevant topology information” to the tracker node 20 so that tracker node 20 generates the ranked list according to “predefined rules” including the ranking criteria. EX1004, [0036];

supra, Element [1a]. In another embodiment, the peer-to-peer manager 30 filters the initial list in accordance with one or more filtering criteria. For example, locality (ISP) can serve as the filtering criterion, meaning that peer nodes that do not connect to the same ISP as the requesting peer node are suppressed from the ranked list. EX1004, [0035].

After the peer-to-peer manager 30 and/or the peer node 20 ranks or filters the initial list of peer nodes, the ranked/filtered list is returned to peer node A—the *user's device*. EX1004, [0030]-[0035], FIG. 2. As depicted in FIG. 2, the ranked/filtered list of peer nodes includes peer node B as the first node, followed by peer nodes C, E, and D. EX1004, [0031], FIG. 2. When the *user's device* represented by peer node A receives this ranked/filtered list of peer nodes, peer node A obtains *link data* because the ranked/filtered list of peer nodes data indicates connections between peer node A and other peer nodes that can supply the requested resource. EX1004, [0029]-[0032], FIG. 2. Indeed, Schwan expressly discloses that when the *user's device* represented by peer node A receives the ranked/filtered list of peer nodes, “[t]hen peer node A can contact the peer nodes *accordingly*.” EX1004, [0032], FIG. 2. This indicates that by obtaining the ranked/filtered list of peer nodes, peer node A obtains *link data* for contacting other peer nodes to obtain the requested resource. EX1004, [0029]-[0032], FIG. 2; EX1003, ¶87. In embodiments where the peer-to-peer manager 30 suppresses peer

nodes that do not connect to the same ISP as peer node A from the filtered list, peer node A obtains **link data** because the filtered list of peer nodes provides a connection relationship indicating that all peer nodes in the filtered list connect to the same ISP. EX1004, [0035].

Peer node A obtains this **link data** according to the **search result** generated by the **search engine** of tracker node 20, because the **search result** includes the ranked/filtered list of peer nodes that prompts peer node A to “then...contact the peer nodes accordingly” using the **link data**. EX1004, [0032], FIG. 2. Peer node A also obtains the **link data** according to the **topology data table** of the online server, because either the peer-to-peer manager 30 or the tracker node 20 ranks/filters the initial list of peer nodes based on ranking criteria or filtering criteria that use information from the **topology data table**. EX1004, [0031], FIG. 2. For example, peer node B is at the top of the ranked peer list because peer node B and peer node A are connected to the same access network (ISP1), and there is a small latency (3 milliseconds) between peer nodes A and B, which means that peer node A obtains the **link data** according to the **topology data table** of the **online server**. EX1004, [0029]-[0032], FIG. 2; EX1003, ¶¶88-89. Furthermore, in cases where the filtered list includes only P2P nodes within the same ISP as peer node A, the filtered list includes **link data** indicating that all of the identified peer nodes are linked to the same ISP. EX1004, [0035].

Claim [2]

Schwan expressly discloses that tracker node 20 generates an initial list of peer nodes based on receiving a ***search request*** from peer node A by searching the ***index database***, and that this initial list of peer nodes is ***sorted*** according to “ranking criteria” that depend on the ***link data*** stored in the ***topology data table***. EX1004, [0029]-[0032], FIG. 2. The ***sorting*** techniques of Schwan ensure that the most ideal peer nodes for obtaining the requested resource appear at the top of the ranked peer list that is delivered to peer node A. EX1004, [0030] (“specific rules to rank the peer nodes of the initial list as a function of the quality of the connection with the requesting peer node A”), [0031], FIG. 2; EX1003, ¶¶90-91. This means that Schwan’s P2P system creates ***weighted data*** based on the ***link data*** stored in the ***topology data table*** and uses this ***weighted data*** to sort the initial peer list by the ***order*** of the ***weighted data***. EX1004, [0029]-[0032], FIG. 2.

Schwan explains that in one embodiment, tracker node 20 and peer-to-peer manager 20 “cooperate...to generate the...ranked list” of peer nodes. EX1004, [0036], FIG. 2. According to this embodiment, “peer-to-peer manager 30 communicates the relevant topology information for the tracker node 20 to directly *generate the optimized list* in accordance with *predefined rules*.” *Id.* This means that the ***content management server*** represented by tracker node 20 is provided with a ***sorting algorithm module*** that uses the ***link data*** from the ***topology data***

table for generating a *sorted search result* based on the initial peer list generated by tracker node 20. EX1004, [0029]-[0032], [0036], FIG. 2; EX1003, ¶92.

Claim [3]

Schwan expressly discloses that that in one embodiment, tracker node 20 can “cache the topology information from the P2P manager in order to have fast access to a nearly up-to-date list with all online peers and the connection ratings between them.” EX1004, [0036], FIG. 2; EX1003, ¶93. These “connection ratings” are generated based on “any information that is made available...about the physical topology,” and represent the *weighted data* of each *P2P device* of Schwan’s *P2P device community*. EX1004, [0030], FIG. 2. Schwan also explains that the “topology information” and “specific rules” are used to “rank the peer nodes of the initial list as a function of the quality of the connection with the requesting peer node A.” *Id.*

The *content management server* therefore includes a *weight look-up table* recorded with the *weighted data* corresponding to each *P2P device*. EX1004, [0030], [0031], [0035], [0036], FIG. 2; EX1003, ¶94. This *weight look-up table* is in addition to the *topology data table* stored by peer-to-peer manager 30. *Id.* To maintain the cache including the *weight look-up table*, tracker node 20 “invokes P2P Manager 30” to obtain the relevant information from the *topology data table*,

confirming that the *weight look-up table* is separate from the *topology data table*.

Id.

Claim [4]

Schwan expressly discloses that the P2P system uses “topology information and specific rules to rank the peer nodes of the initial list as a *function* of the quality of the connection with the requesting peer node A,” which represents a *function operation* to sort the initial peer list generated by tracker node 20.

EX1004, [0030], [0031], [0035], [0036], FIG. 2; *supra*, Claim [2]. The *topology information* used to sort the initial peer list includes *weighted data* including “connection ratings” for connections between peer node A and other peer nodes.

Id. Therefore, Schwan explains that the *sorting algorithm module* of the *content management server* represented by tracker node 20 performs a *function operation* to generate the sorted list of peer nodes based on the *weighted data* and the initial list of peer nodes from the *search result*. EX1004, [0030], [0031], [0035], [0036], FIG. 2; EX1003, ¶¶95-96. The ranked list of peer nodes represents a *weighted* and *sorted search result*, because the ranked list is *sorted* based by the order of the *weighted data* indicating the connection rating of each link. *Id.*

Claim [7]

Schwan expressly discloses that “tracker node 20 maintains a list of the active peers in the peer-to-peer network and the respective resources that they

hold,” which means that the *content management server* creates the *indexed data* including index numbers (e.g., “chunk 2”) for each resource. EX1004, [0029], *see also id.*, [0030] FIG. 2; EX1003, ¶97. To maintain this list of active peers and the resources they hold, the *content management server* represented by tracker node 20 therefore includes a *network gathering module* that gathers information from the peer nodes indicating the resources stored by each peer node. *Id.*

Claim [8]

Schwan expressly discloses that the “peer node A can contact the peer nodes accordingly” with the *link data*, meaning that the user’s device *connects* to one or more *P2P devices* based on the *link data*. EX1004, [0032], *see also id.*, [0028]-[0031], FIG. 2. Additionally, Schwan explains that peer node A sends a *search request* to tracker node 20 when “peer node A wants to *download a file*...from the...P2P network.” EX1004, [0029], *see also id.*, [0004], FIG. 2; EX1003, ¶98. Based on this search request, tracker node 20 returns a ranked list of peer nodes that host the requested file data, so that peer node A can *download the file data* based on the *link data* corresponding to the ranked search result. *Id.*

Claim [9]

Supra, Claim [8]; EX1003, ¶99.

Element [11pre]

Supra, Elements [1pre]-[1c]; EX1003, ¶100.

Element [11a]

Supra, Element [1a]-[1b]; EX1003, ¶101.

Element [11b]

Supra, Elements [1c]-[1d]; EX1003, ¶102.

Element [11c]

Supra, Element [1d]; EX1003, ¶103.

Element [11d]

Supra, Element [1d]; EX1003, ¶104.

Element [11e]

Supra, Element [1e]; EX1003, ¶105.

Claim [12]

Supra, Claims [2] and [4]; EX1003, ¶106.

Claim [13]

Supra, Claim [3]; EX1003, ¶107.

Claim [16]

Supra, Claim [7]; EX1003, ¶108.

Claim [18]

Supra, Claim [8]; EX1003, ¶109.

Claim [19]

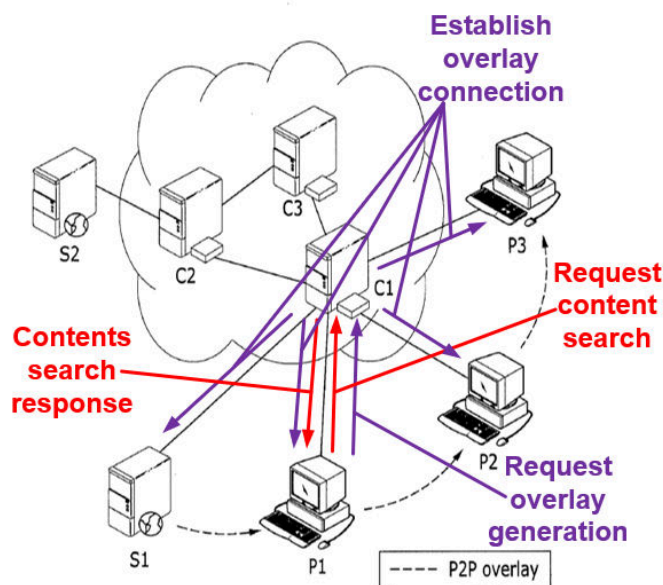
Supra, Claim [9]; EX1003, ¶110.

B. GROUND 1B: – Schwan in view of Lee would have rendered obvious claim 1-4, 7-13, 16-19

1. *Lee*

Lee describes a P2P network system including a P2P control device (C1) that manages content downloads by peers from contents servers that share data over the P2P network. EX1005, [0038]-[0094], FIGS. 2-8; EX1003, ¶43. For example, Lee explains that a peer (P1) outputs a request for a contents search to C1. EX1005, [0040], [0042], [0052], [0076], [0077], FIG. 2, FIG. 3, FIG. 6. Based on the request, C1 searches for contents selected by P1 and selects a contents server (S1) for providing the selected contents. EX1005, [0042], [0052], [0077], *see also id.*, FIG. 3.

FIG. 2



EX1005, FIG. 2 (annotated).

After searching for contents, any of the peers P1-P3 can output a request for C1 to generate an overlay over which the requesting peer can receive the contents. EX1005, [0034]-[0036], [0043], [0044], [0054], [0089]-[0094], FIG. 2, FIG. 3, FIG. 8; ; EX1003, ¶44. Specifically, Lee explains that C1 is configured to “generate a P2P overlay according to a request from the peer P1, and transfers information regarding a connection to parent and child nodes corresponding to the P2P overlay to the peer P1.” EX1005, [0054], *see also id.*, [0043], [0091]. The overlay is generated based on “information regarding the network” including “a network topology, a maximum bandwidth, a current network usage, a transmission delay time, and the like.” *Id.*

2. *The Schwan-Lee Combination*

To the extent that Schwan does not expressly disclose a user’s device that obtains link data for connecting to a P2P device based on a search result and a topology data table, Lee suggests this feature was commonly implemented to achieve several benefits. EX1005, [0038]-[0094], FIGS. 2-8; *supra*, §V.B.1; EX1003, ¶45. For example, Lee teaches that a peer (e.g., any of P1-P3) outputs a request for C1 to search for contents and receives a search result from C1 including a contents identifier. EX1005, [0042], [0052], [0077]. Using the contents identifier from this search result, the peer requests an overlay and C1 provides link data for the peer to receive the contents via the overlay. EX1005, [0043], [0054], [0091].

Multiple reasons would have prompted a POSITA to implement the P2P system of Schwan with Lee's suggestions for providing link data to generate an overlay for delivering contents requested by a user's device. EX1003, ¶54.

First, a POSITA would have understood that generating an overlay based on topology information and implementing the overlay by outputting link data would predictably provide a simple and secure way for a user's device to receive content through a P2P network. EX1003, ¶46. For example, Schwan acknowledges that when a P2P network overlay is generated without regard for network infrastructure, this can result in connections between P2P devices that span network borders and result in inefficient transmission. EX1004, [0002]-[0004]; EX1003, ¶46. Lee's solution for generating an overlay based on analyzing network information including network topology and transmission delay time and outputting link data to implement the overlay would address this problem of inefficient connections acknowledged by Schwan. EX1003, ¶46.

Second, a POSITA would have been prompted to implement Schwan's P2P system with Lee's suggestions for establishing overlays including multiple hops, because doing so would allow a P2P device to obtain resources from another P2P device even in cases where the P2P devices cannot connect over a direct link. EX1003, ¶47. Because Schwan explains that these multi-hop overlays can be generated to account for network topology, the resulting system would connect

devices over multiple hops in a way that does not result in unnecessary connections which slow transmission time or needlessly consume resources. EX1003, ¶47.

Third, because Lee describes techniques for changing an overlay in response to changing network characteristics, a POSITA would have been motivated to implement Schwan's P2P system with Lee's overlay suggestions to maintain efficient connections over time. EX1003, ¶48. For example, when network conditions such as topology, bandwidth, and transmission delay times change over time, the most optimal route for a P2P device to receive content can also change, meaning that it is beneficial to update the network overlay based on changing conditions as suggested by Lee. EX1003, ¶48.

Finally, it would have been obvious for a POSITA to implement Schwan's P2P system with the teachings of Lee for generating an overlay because doing so represents one of a finite number of predictable solutions for allowing P2P devices to receive content over a P2P network. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 417 ("When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp."). EX1003, ¶49. A POSITA would have reasonably expected success implementing the Schwan-Lee combination because P2P networks and P2P management servers

for searching content were well known before the Critical Date of the '655 Patent.

KSR at 401; EX1003, ¶49.

3. *Application to Challenged Claims*

Elements [1pre]-[1b]

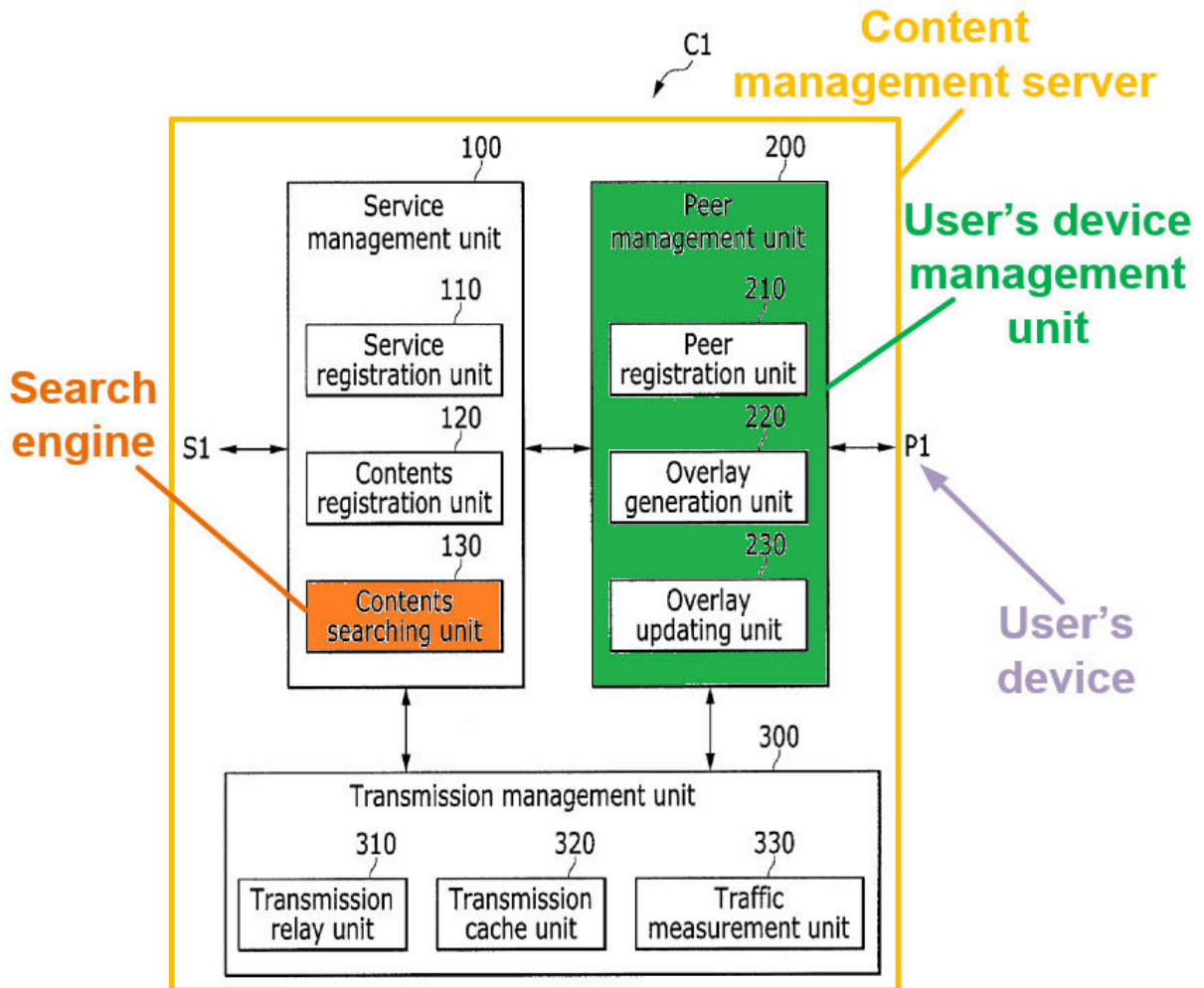
Supra, Ground 1A (§V.A.2) at Elements [1pre]-[1b]. EX1003, ¶111.

Elements [1c]-[1d]

The P2P control device C1 of Lee includes a service management unit 100 including a contents searching unit 130. EX1005, [0047], [0052], FIG. 3. This contents searching unit 130 represents a ***search engine*** that “searches for an appropriate version of contents fitting the characteristics of the peer P1 with respect to a corresponding type of contents selected by the peer P1 and transfers the information regarding the searched version of contents to the peer P1.”

EX1005, [0052], FIG. 3. This means that P2P control device C1 includes a ***content management server*** including a ***search engine*** that generates a search result based on receiving a ***search request*** from a peer. EX1005, [0045]-[0055], FIG. 3; EX1003, ¶¶112-113.

FIG. 3



EX1005, FIG. 3 (annotated).

Furthermore, the P2P control device C1 of Lee includes a peer *management unit* 200, that “registers the peer P1 according to a registration request from the peer P1 and manages information regarding the registered peer P1.” EX1005, [0053], *see also id.*, [0052], [0054], [0055], [0075]-[0082], FIG. 3, FIG. 6. As depicted in FIG. 3 of Lee, peer management unit 200 is connected to a *user’s device* represented by peer P1. EX1005, FIG. 3. This means that when peer P1

sends a *search request* to the P2P control device C1, the peer *management unit* 200 represents a *user's device management unit* that receives the *search request* sent from the *user's device*. EX1005, [0052]-[0055], [0075]-[0082], FIG. 3, FIG. 6; EX1003, ¶114.

Element [1e]

To the extent Schwan does not disclose that the *user's device* obtains *link data* for *P2P devices* indicated by the ranked peer list according to *topology data table* stored by the peer-to-peer manager 30 and the *search result* represented by the ranked peer list, such features would have been obvious and provided in the Schwan-Lee combination. EX1005, [0034], [0042]-[0044], [0053]-[0054], [0089]-[0094], FIG. 2, FIG. 3, FIG. 8; EX1003, ¶¶115-116. Lee discloses, for example, that a *user's device* represented by one of peers P1-P3 outputs a request for a contents search to P2P control device C1 which performs the contents search and returns a *search result* to the *user's device*. EX1005, [0052], [0076]-[0082], FIGs. 3, 6. This search result includes a contents identifier corresponding to the contents requested by the *user's device*. EX1005, [0077], *see also id.*, [0052]. The *user's device* subsequently uses the contents identifier to establish an overlay for downloading the resource. EX1005, [0053]-[0055], [0089]-[0094], FIG. 3, FIG. 8.

For example, Lee explains that “a contents searching unit” is “configured to receive a request for searching for [] desired contents from [a] requesting

peer...and provide a contents identifier of searched contents to the requesting peer.” EX1005, Cl. 1, Cl. 5; EX1003, ¶117. Lee also explains that “an overlay generation unit” is “configured to receive an overlay generation request of desired contents from [the] requesting peer,” and “the overlay generation request comprises the contents identifier.” *Id.* Lee therefore confirms that the *user’s device* represented by peer P1 requests an overlay *according to* the contents identifier included in a *search result*. *Id.*

To generate the overlay requested by the user’s device, Lee explains that “overlay generation unit 220 generates the P2P overlay by collectively analyzing and considering information regarding the corresponding application service, the peer, and the network.” EX1005, [0054], *see also id.*, [0092]; EX1003, ¶118. Lee expressly discloses that the information used to create the overlay includes “network *topology*, a maximum bandwidth, a current network usage, a transmission delay time, and the like.” *Id.* P2P control device C1 subsequently “transfers information regarding a connection to parent and child nodes corresponding to the P2P overlay to the peer P1,” and this information represents *link data*. *Id.*

The predictable Schwan-Lee combination therefore provides a *user’s device* that obtains *link data* of a *P2P device* within a P2P device community according to

a *topology data table* and a *search result*. EX1005, [0053], [0054], [0092], Cl. 1, Cl. 5; *supra*, §V.B.2; EX1003, ¶119.

Claims [2]-[4] and [7]-[9]

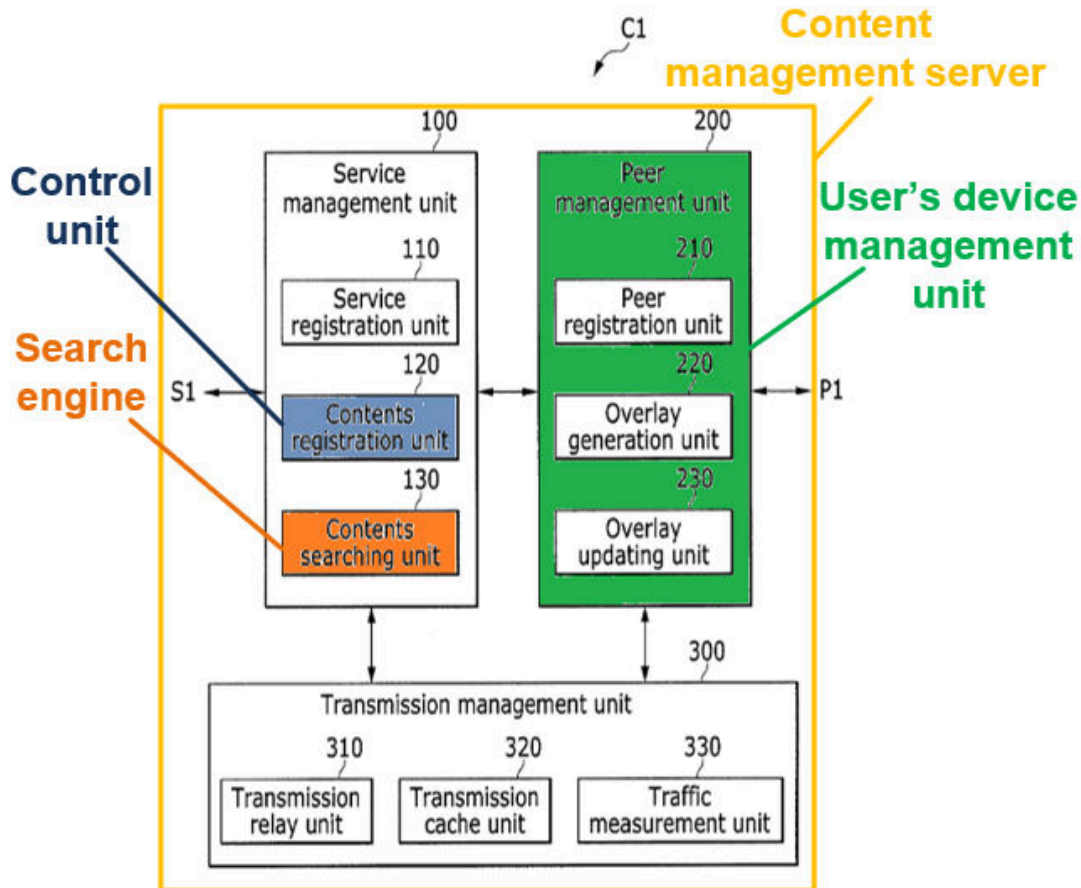
The predictable Schwan-Lee combination would provide the Elements recited here because Schwan's system discloses those features as detailed above.

Supra, Ground 1A, Claims [2]-[4] and [7]-[9]; EX1003, ¶120.

Claim [10]

Lee also suggests that the P2P control device C1 of includes a "contents registration unit 120" that receives a "contents registration request from the contents server S1 and registers information regarding contents providing an application service." EX1005, [0050], *see also id.*, [0069]-[0074], FIG. 3, FIG. 5; EX1003, ¶121. For example, Lee explains that "[t]he P2P control device C1 registers the contents information received from the contents server S1 and manages the contents information by assigning a contents identifier to the registered contents information." EX1005, [0071], FIG. 5. Lee also suggests that the P2P control device C1 should output the contents identifier to contents server S1 when the registration is complete. EX1005, [0074], FIG. 5.

FIG. 3



EX1005, FIG. 3 (annotated).

Server S1 therefore includes a control unit that registers contents with the contents registration unit 120 of the P2P control device C1. EX1005, [0051], [0071]. By registering contents, the *control unit* of server S1 controls *file data* of the contents to occupy an *open state* because registration results in a contents identifier associated with the *file data*. EX1005, [0038]-[0055], [0069]-[0074], FIG. 2, FIG. 3, FIG. 5. Lee explains that in the *open state*, peers establish an overlay connection for downloading the contents using the contents identifier.

EX1005, [0054], [0055], [0089]-[0094], FIG. 3, FIG. 8. When server S1 does not register contents with the contents registration unit 120, the contents remain in a ***hidden state*** because P2P control device C1 cannot provide a contents identifier for peers to access the ***file data***. *Id.* Accordingly, the predictable Schwan-Lee combination would have provided such a ***control unit*** (controlling the ***open*** or ***hidden state***, as suggested by Lee) to facilitate “active[] update[s]” of the participating peers and to “increas[e] content transmission quality.” EX1003, ¶122 (citing to EX1005, [0012], [0014]).

Element [11pre]

Supra, Ground 1A (§V.A.2) at Elements [1pre]-[1c]. EX1003, ¶123.

Element [11a]

Supra, Ground 1A (§V.A.2) at Elements [1a]-[1b]. EX1003, ¶124.

Elements [11b]-[11d]

Supra, Elements [1c]-[1d]; EX1003, ¶125.

Element [11e]

Supra, Element [1e]; EX1003, ¶126.

Claim [12]

Supra, Claims [2] and [4]; EX1003, ¶127.

Claim [13]

Supra, Claim [3]; EX1003, ¶128.

Claim [16]

Supra, Claim [7]; EX1003, ¶129.

Claim [17]

Supra, Claim [10]; EX1003, ¶130.

Claim [18]

Supra, Claim [8]; EX1003, ¶131.

Claim [19]

Supra, Claim [9]; EX1003, ¶132.

C. GROUND 1C: Schwan, in view of Lee, and further in view of Strickland would have rendered obvious claims 5, 6, 14, 15

1. *Strickland*

Strickland describes a system including a P2P service provider which facilitates monetary transactions for file sharing between P2P devices. EX1006, [0050]-[0094], [0098]-[0106], FIG. 1, FIG. 2; EX1003, ¶¶50-51. In the P2P system of Strickland, “user 1” can send a search query to a P2P server which searches a database for matches to the search query. EX1006, [0099]. When at least one match is identified, the P2P server sends a search result to the user 1 including the identified matches. *Id.* When the P2P server receives a selection of a particular file (i.e., File A hosted by “user 2”), the P2P server facilitates a transaction for user 1 to download the selected file. EX1006, [0099]-[0101], [0105], [0106], FIG. 2. In both direct monetary transactions and indirect credit transactions, the P2P server

ensures that the downloading user (e.g., user 1) is charged before the file download is authorized. EX1006, [0042], [0092], [0098]-[0106], FIG. 2.

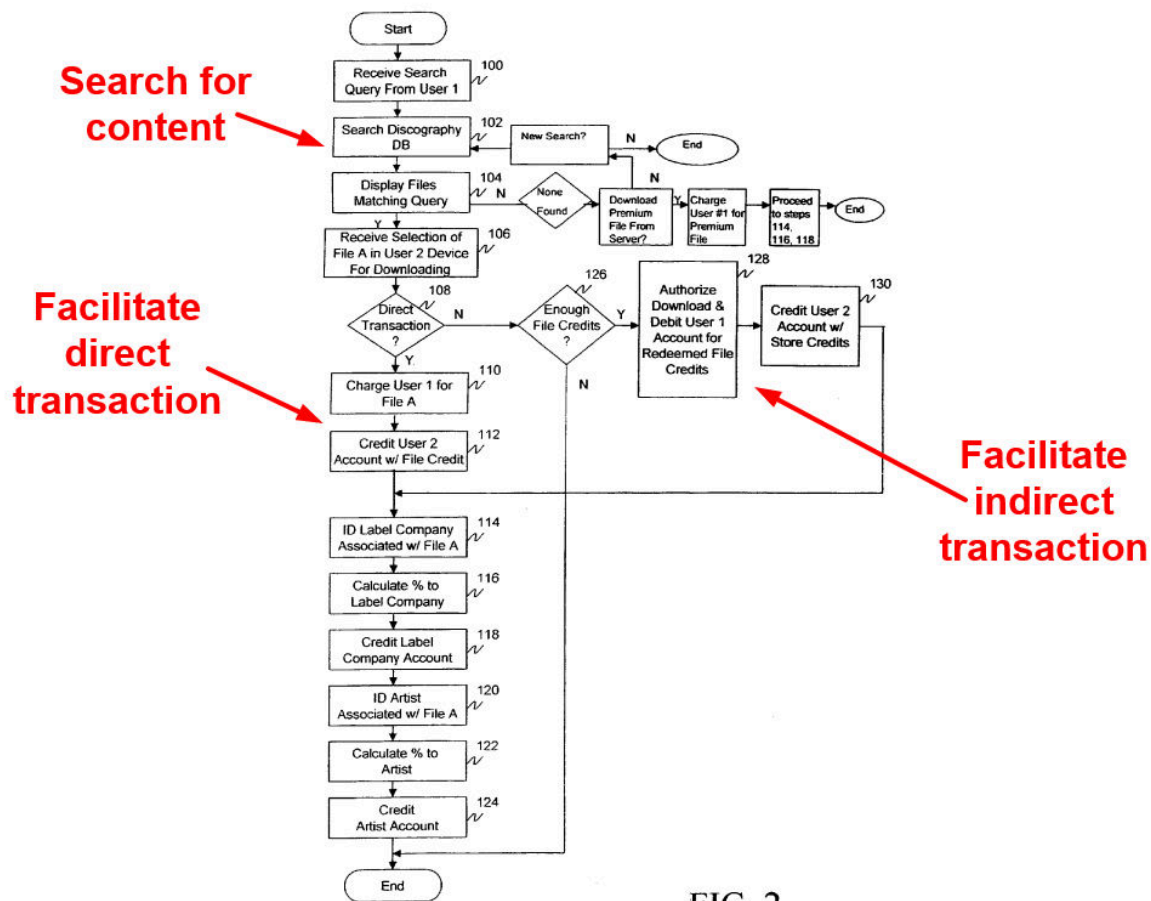


FIG. 2

EX1006, FIG. 2 (annotated).

2. The Schwan-Lee-Strickland Combination

The Schwan-Lee combination provides a P2P system including a content management server that receives a search request from a user's device, performs a content search based on an index database and a topology data table stored by an online server, and provides a search result to the user's device. EX1004, [0023]-[0039], FIG. 1, FIG. 2; EX1005, [0038]-[0055], FIG. 2, FIG. 3; *supra*, §V.B.2;

EX1003, ¶52. However, the Schwan-Lee combination does not describe any techniques for confirming that a requesting user's device has been charged before providing search results or link data. EX1003, ¶52. These features were nonetheless well known before the Critical Date of the '655 Patent as demonstrated by Strickland (*Supra*, §V.C.1). Multiple reasons would have prompted a POSITA to apply Strickland's suggestion for charging a requesting device before advancing a content download to the P2P system of the Schwan-Lee combination. EX1003, ¶52.

First, a POSITA would have been motivated to implement the P2P system of the Schwan-Lee combination with Strickland's suggestion for facilitating monetary transactions, because doing so would provide a convenient way to charge users for downloading content and reward users for providing content in the P2P network. EX1003, ¶53. **Second**, a POSITA would have recognized that facilitating at least two different kinds of transactions as suggested by Strickland including direct monetary transactions and indirect transactions involving pre-purchased tokens would result in a flexible system that allows users to acquire downloads conveniently as needed. EX1003, ¶54. **Third**, because media content is often subject to copyright ownership interests, a POSITA would have been motivated to implement the P2P system with Strickland's suggestion for rewarding copyright owners with a share of the download purchase price. EX1003, ¶55.

Finally, the Schwan-Lee combination and Strickland both involve P2P systems that facilitate content downloads in a P2P network. EX1003, ¶56. In this context, a POSITA would have possessed ample skill to successfully implement the combination. *KSR* at 417. Based upon the predictable electrical components and software involved in the combination and the state of the art at the time, a POSITA would have reasonably expected success achieving the combination before the Critical Date. *KSR* at 401; EX1003, ¶56.

3. *Application to Challenged Claims*

Claim [5]

Strickland describes a P2P server that receives a search query from “[u]ser 1 operating a first end user device,” meaning that Strickland describes a ***user’s device management unit*** that receives a ***search request*** from a ***user’s device***. EX1006, [0099], *see also id.*, [0052], [0059], [0060], [0092], [0093], FIG. 2; EX1003, ¶133.

For “direct” transactions, a P2P server charges User 1 “a monetary amount (Dollar, Euro, etc.) for the downloading of File A” from the second end user device as part of a direct monetary transaction. EX1006, [0100], *see also id.*, [0092], [0093], FIG. 2; EX1003, ¶134. Strickland also explains that “the server compensates User 2 with a credit...for providing File A to User 1.” *Id.* Therefore, Strickland describes a ***user-charge module*** that confirms that a ***user’s device***

represented by the first end user device has been charged before advancing a file download to provide a *search result* in response to the *search request*. *Id.*

For “indirect” transactions, Strickland explains that “a determination is made as to whether User 1 has accumulated enough file credits by examining User 1’s account” and “[i]f User 1 has accumulated enough file credits...the download is authorize...and User 1’s account is debited for the redeemed file credits.” EX1006, [0100], [0105], [0106], FIG. 2; EX1003, ¶135. By confirming that user 1 has enough credits for a download before authorizing the download, Strickland’s *user-charge module* therefore advances the download to provide a *search result after* confirming that the *user’s device* has been charged. EX1006, [0105], FIG. 2.

When the P2P system of the Schwan-Lee combination is implemented with Strickland’s suggestion for supporting monetary transactions, the *content management server* includes a *user-charge module* that provides a *search result* based on confirming that a *user’s device* has been charged for downloading a file. EX1006, [0098]-[0106], FIG. 2; EX1003, ¶136; *supra*, §V.C.2.

Claim [6]

Strickland expressly discloses that the *user-charge module* of the P2P server “may automatically run a purchasing routine where User 1 is asked a series of questions, including questions for effectuating payment for the downloading of the file, such as for example, credit card information, bank account information, or the

like.” EX1006, [0100], *see also id.*, [0078], [0094], FIG. 2; EX1003, ¶137.

Therefore, the *user-charge module* of the P2P server receives *charging data*, *charging-mode data*, and *payment confirmation data* from the *user’s device*. *Id.*

Claim [14]

Supra, Claim [5]; EX1003, ¶138.

Claim [15]

Supra, Claim [6]; EX1003, ¶139. Furthermore, Strickland expressly discloses that “[a]nonymous users may not be able to redeem credits for merchandise, however, because the user may have to be identified to receive shipments,” meaning that confirming that the user’s device has been charged comprises confirming *identity data* of the *user’s device*. EX1006, [0061], *see also id.*, [0034], [0237]; EX1003, ¶139. Because Strickland discloses that User 1 can make an indirect purchase using prepaid credits or a direct purchase using currency, Strickland confirms that the user has been charged by receiving charge-mode data sent from the user’s device. EX1006, [0100], [0100], [0105], [0106], FIG. 2; *cf.* EX1001, 7:34-37 (“charge-mode data...allowing the user’s devices...to select the payment plan which may be, for example, a monthly installment payment mode or pre-payment mode”).

Strickland also explains that for direct payments, the P2P server “proceeds...to charge User 1 a monetary...for the downloading of File A” and for

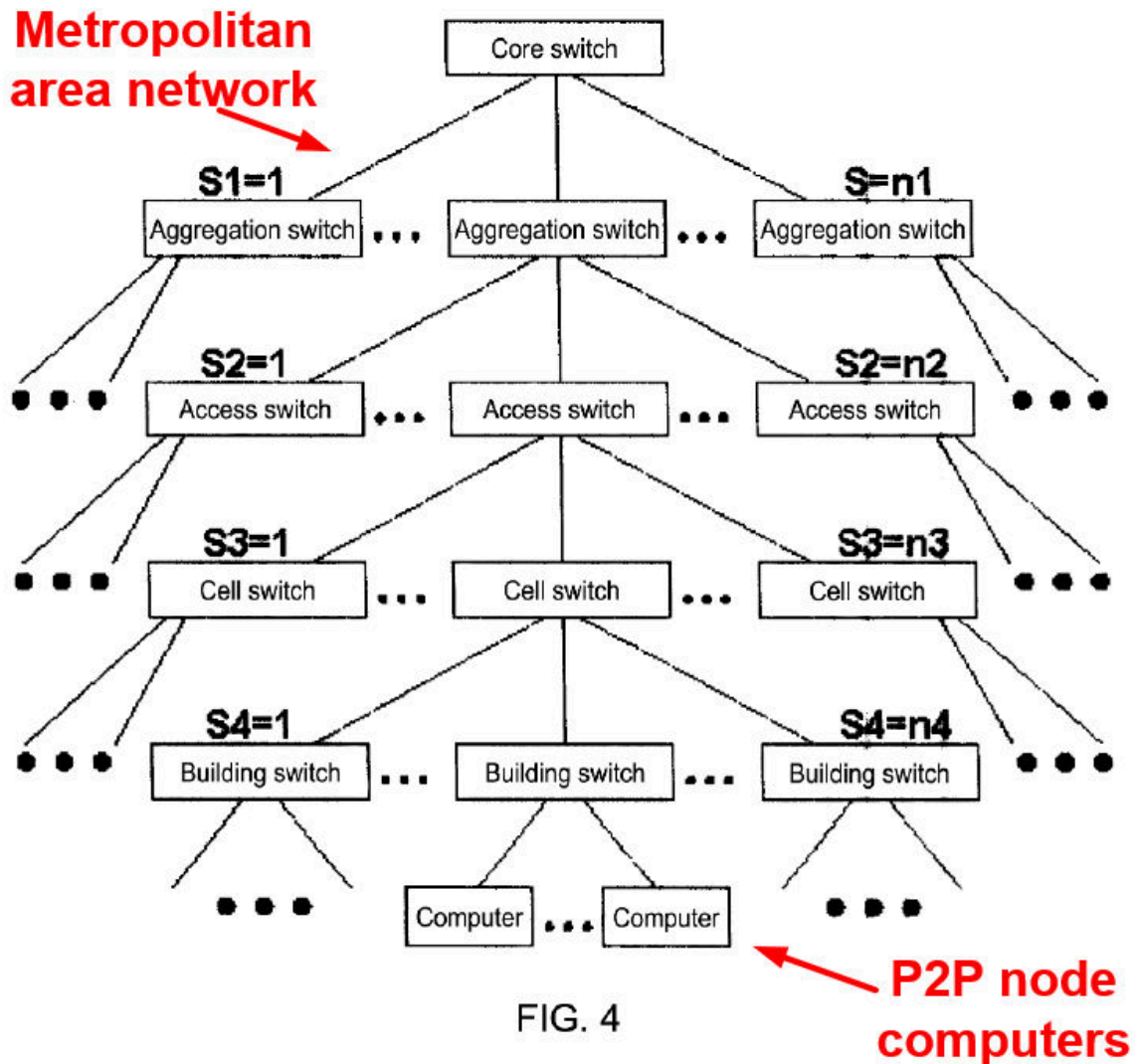
indirect payments “User 1’s account is debited for the redeemed file credits,” meaning that in both cases the P2P server confirms *payment confirmation data* to confirm that the user’s device has been charged. EX1006, [0100], [0105], FIG. 2. In direct transactions, the P2P server charges User 1 at a price point. EX1006, [0100] (“[b]y way of example, a price point of 50 cents may be chosen”), *see also id.*, [0085]-[0088], FIG. 2. In indirect transactions, credits have value. EX1006, [0089]-[0091]. Thus, Strickland explains that the P2P server records *pricing data* required for *downloading* the selected file. EX1003, ¶140.

D. GROUND 2A: Li anticipates claims 1, 7-9, 11, 16, 18, 19

1. *Li*

Li describes a P2P system that determines the most optimal path for a P2P node to receive a resource over a P2P network. EX1007, 4-11, 14-18; EX1003, ¶57. The P2P system of Li includes “aggregation switches, access switches, cell switches, and building switches” that are connected together to define a metropolitan area network. EX1007, 9, *see also id.*, 2-8 10-18, FIG. 1, FIG. 4. The metropolitan area network also includes “P2P node computers” that connect to the switches to access resources stored in other parts of the network. EX1007, 9, *see also id.*, 8-10, 17, FIG. 1, FIG. 4. Each node in the metropolitan area network is associated with a level depending on the type of switch (e.g., s1 for aggregation switches, s2 for access switches), and a number associated with the level. EX1007,

9, 14, FIG. 4. These levels and numbers can be used to identify a location of nodes within the metropolitan area network. EX1007, 9, 15, 16, FIG. 4.



EX1007, FIG. 2 (annotated).

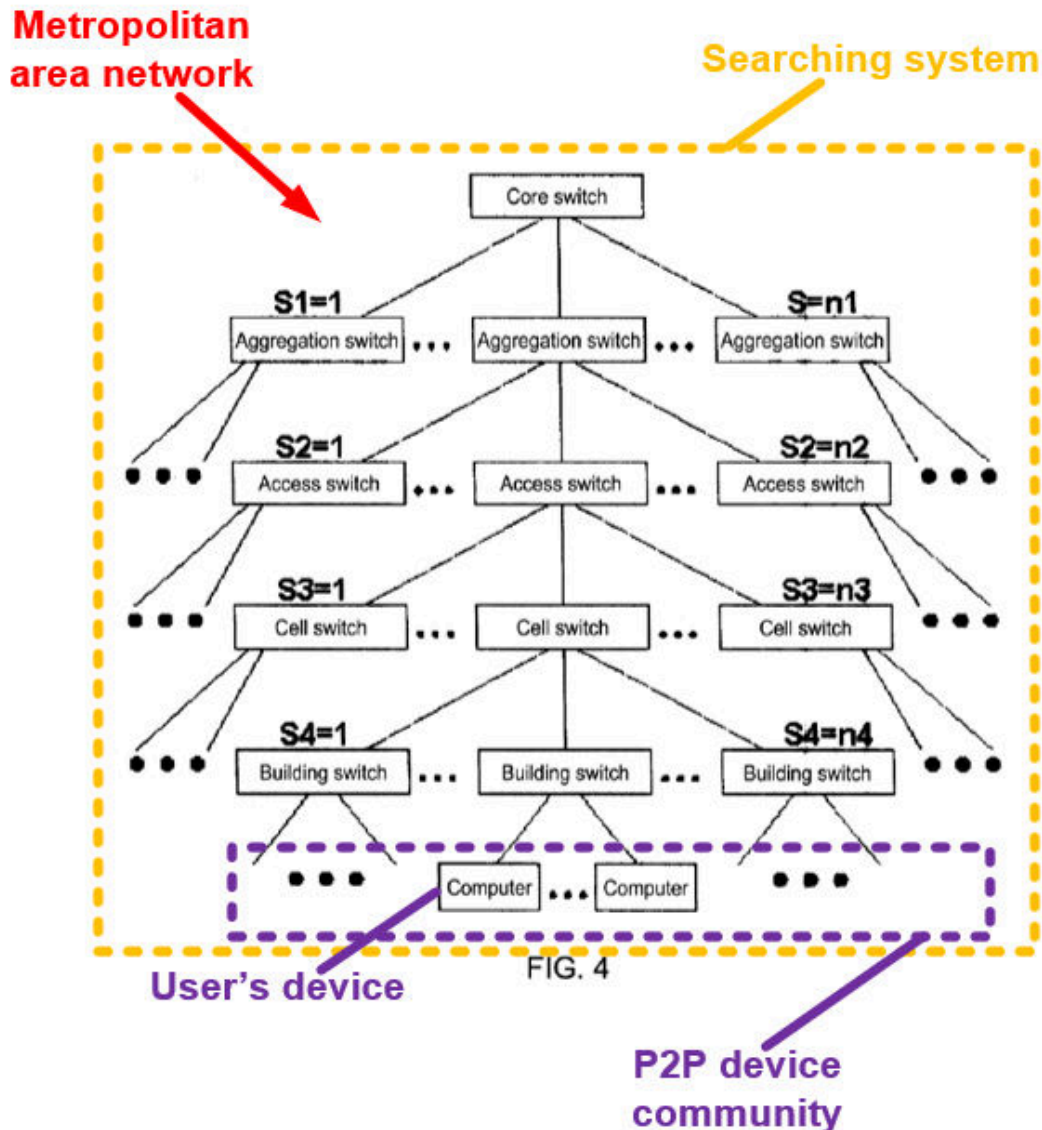
Li explains that “a topology server and a P2P index server are added in the metropolitan area network.” EX1007, 14, *see also id.*, 5, 7, 8, 10, 11, 16, 17, FIG. 4. In the metropolitan area network of Li, the P2P index server receives a “query

request” from a “requesting P2P node” including a request to download a “certain resource.” EX1007, 11, *see also id.*, 14, 15, 17; EX1003, ¶58. Subsequently, the P2P index server performs querying to identify “resource P2P nodes” that store the resource. *Id.* Based on this list of resource P2P nodes, Li’s system calculates “one or more nearest neighbor resource P2P nodes.” *Id.*

2. *Application to Challenged Claims*

Element [1pre]

To the extent the preamble is limiting, Li anticipates Element [1pre]. EX1003, ¶141. Li describes a plurality of P2P nodes—sometimes referred to as “P2P node computers”—that represent a ***P2P device community***. EX1007, 2, 3, 5-17. Li describes a metropolitan area network comprising a plurality of switches that are interconnected in a multi-level tree pattern. EX1007, 2, 3, 5, 6, 9, 11, 12-16; *supra*, §V.D.1; EX1003, ¶141. The P2P nodes of the ***P2P device community*** are connected to “building switches” of the multi-level tree pattern such that the P2P nodes can communicate with other devices that are connected to the metropolitan area network. EX1007, 2, 5-7, 9, 11-16, FIG. 4. Li expressly discloses that the metropolitan area network includes a ***searching system*** including a P2P index server and a topology server that handles “resource query requests” received from P2P nodes of the ***P2P device community***. EX1007, 5, 7, 8, 10, 11, 17.



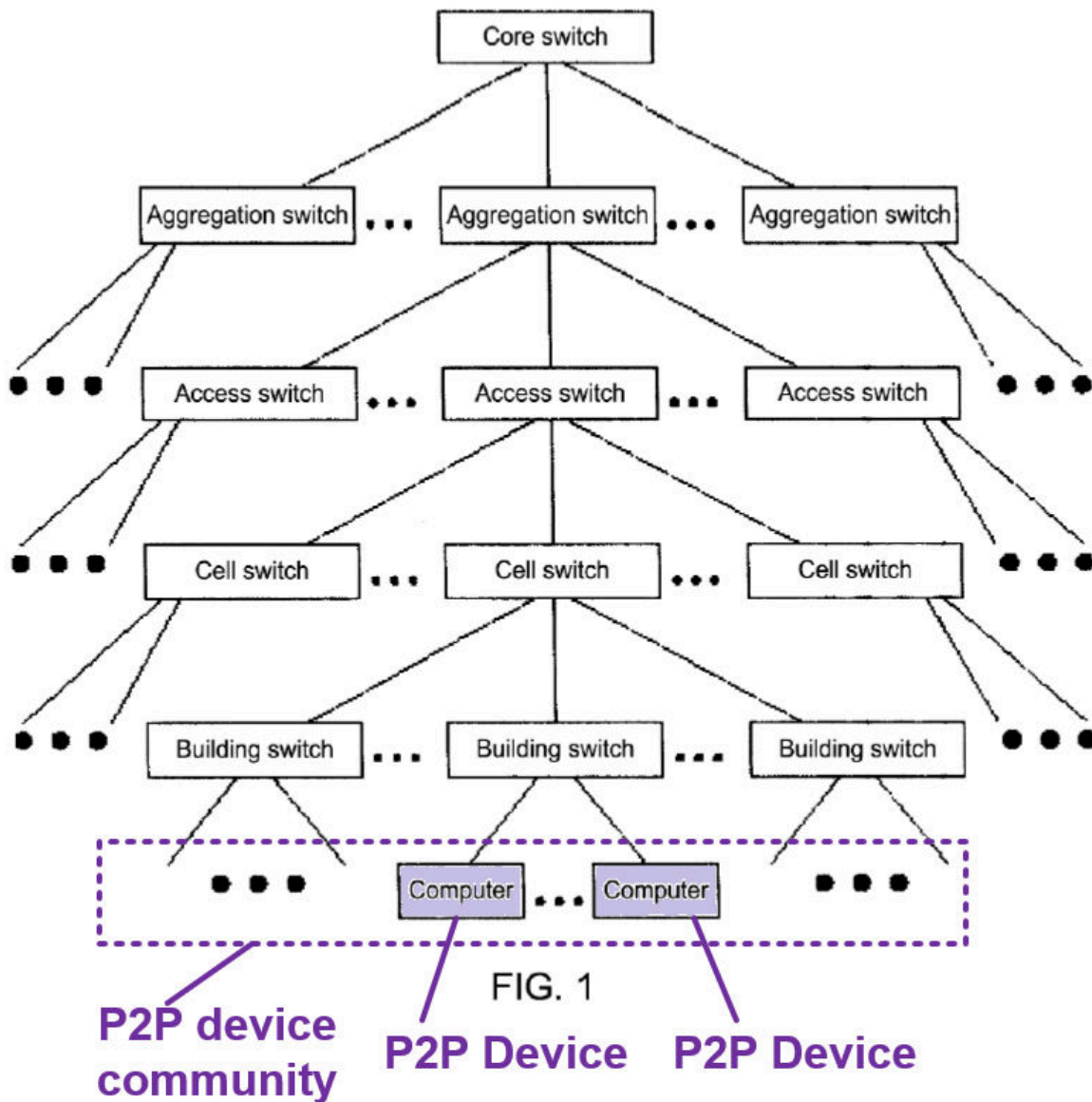
EX1007, FIG. 4 (annotated).

For example, Li explains that the P2P index server receives a “resource query request” from a “requesting P2P node” of the *P2P device community*. EX1007, 7, 9-11, 14-17, FIG. 4. This resource query request represents a *search request* “to download a certain resource.” EX1007, 14, *see also id.*, 5, 8, 10, 11, 18. The requesting P2P node that sends the search request represents a *user’s*

device, because Li expressly refers to P2P nodes connected to building switches as “user computers” and “user nodes.” EX1007, 1-3; EX1003, ¶142.

Element [1a]

Because Li expressly refers to P2P nodes as “P2P node computers” on multiple occasions, these P2P nodes represent *at least one P2P device* of a *P2P device community*. EX1007, 8-10, 17, 18. Furthermore, Li explains that in one embodiment, “[e]ach node computer is both a client that downloads from other nodes and a storage server that can share content stored thereby with other nodes.” EX1007, 1. Multiple sources confirm that a common feature of *P2P devices* is an ability to share files with other P2P devices and an ability to download files from other P2P devices. EX1003, ¶143 (citing corroborating, EX1012, [0008]-[0012], EX1013, [0047]).



EX1007, FIG. 1 (annotated).

The *one or more P2P devices* each store *file data* including *content data* because Li describes multiple examples of P2P nodes downloading files from other P2P nodes connected to the metropolitan area network. EX1007, 1-11, 14-18. P2P nodes that store *content data* for download by other P2P nodes are referred to by Li as “resource P2P nodes” and P2P nodes that request downloads from resource

P2P nodes are referred to by Li as “requesting P2P nodes.” EX1007, 2, 3, 5, 7-12, 14-17; EX1003, ¶144.

Because the P2P *index* server performs a query to identify a resource listed in an *index database*, this means that *content data* stored by each resource P2P node is associated with *indexed data* identifying the *content data*. EX1007, 7, 10, 11, 14, 17. For example, to “obtain a list of P2P nodes having the resource” as described by Li, the P2P index server identifies resource P2P nodes having the resource (*content data*) by identifying resource P2P nodes having *indexed data* corresponding to the resource. EX1007, 7, 10, 11, 14, 17. It was well known by the Critical Date of the '655 Patent for P2P devices to store *file data* including *content data* and *indexed data*. EX1003, ¶145 (citing corroborating, EX1012, [0012], [0052]).

Element [1b]

Li explains that the metropolitan area network includes a “topology server” that “stores the topological structure data of the entire metropolitan area network.” EX1007, 10, *see also id.*, 7, 8, 11, 14, 16, 17, FIGs. 1, 4, 5; EX1003, ¶146. The topological structure data stored by Li’s topology server includes “subnets and location vectors” for P2P nodes, and the location vectors are indicated “according to an IP address” of the P2P node. *Id.* Li’s topology server thus represents an

online server connecting to the *P2P device community* that includes a *topology data table. Id.*

Topology data table

Subnet IP address	Aggregation switch number (s1)	Access switch number (s2)	Cell switch number (s3)	Building switch number (s4)
...	...	...	...	...
10.30.11.65 / 26	2	3	5	4
10.30.11.129 / 26	2	3	5	5
...	...	...	...	...

Subnet IP address of each P2P node Link data

FIG. 5

EX1007, FIG. 5 (annotated).

Li explains that switch numbers “s1, s2, s3, s4” indicate an aggregation switch, an access switch, a cell switch, and a building switch that the P2P node is connected to. EX1007, 6, 9, 15, FIG. 4, FIG. 5. The location vectors for Li’s P2P nodes represent *link data* between each of the *P2P devices* of the *P2P device community* because the location vectors reveal when two or more *P2P devices* are connected to the same switch and are therefore *linked*. EX1007, 6-10, 15-17, FIG. 1, FIGs. 4, 5. For example, Li’s *topology data table* depicts location vectors for two different P2P nodes revealing that both nodes are connected to the same aggregation switch (s1=2), access switch (s2=3), and cell switch (s3=5), but are connected to different building switches (s4=4 and s4=5, respectively), revealing

that the two P2P nodes are *linked*. EX1007, FIG. 5; EX1003, ¶¶147-149. The *topology server* is connected to Li's *P2P device community* because a requesting P2P node sends a search result including the list of resource P2P nodes to the *topology server* in some embodiments, and the *topology server* is able to return one or more nearest neighbor resource P2P nodes to the requesting P2P node. EX1007, 8, 10, 11, 16, 17, 18.

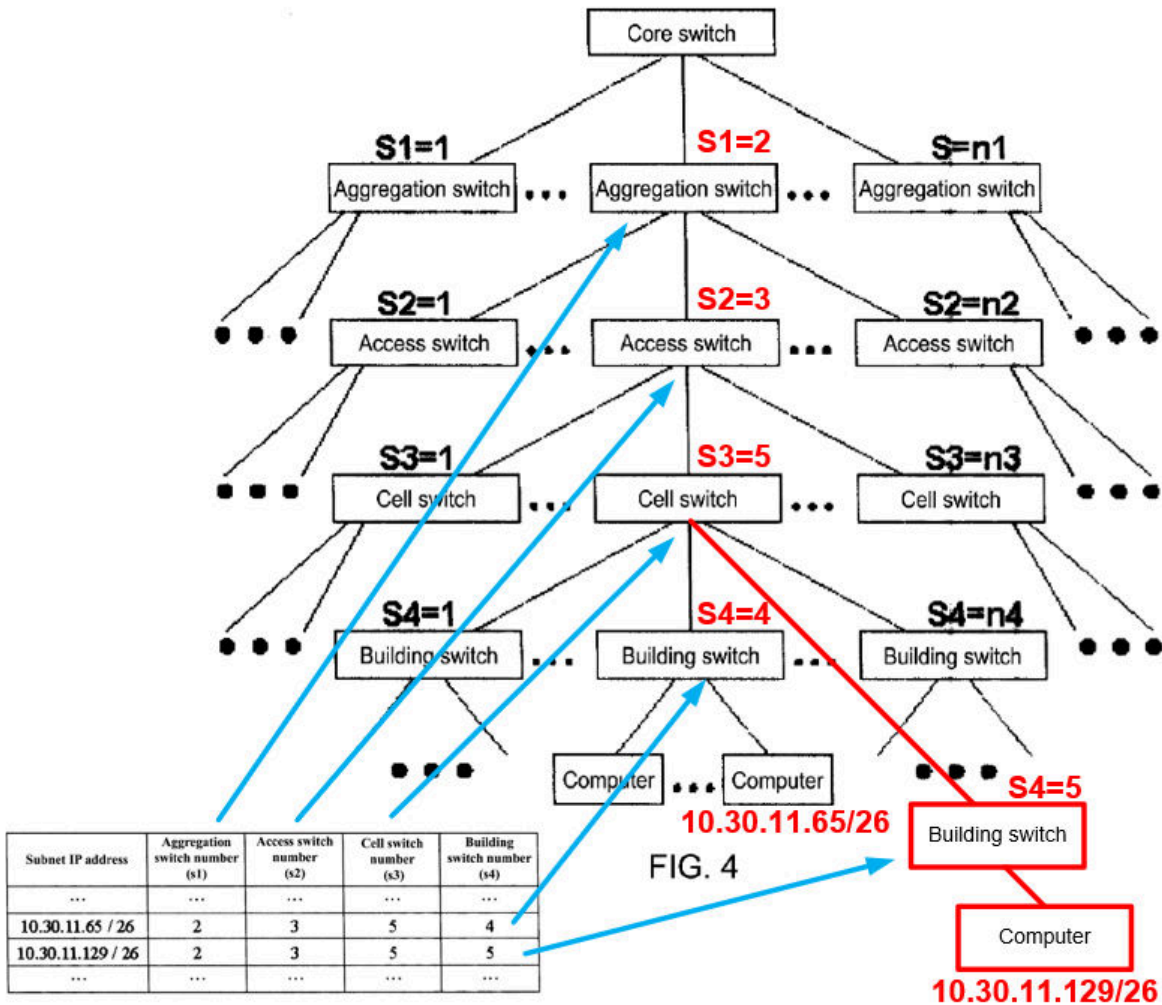


FIG. 5

EX1007, FIG. 4 and FIG. 5 (modified and annotated).

Element [1c]

Li explains that the metropolitan area network includes a “P2P index server” that receives queries from P2P nodes to download resources from other P2P nodes. EX1007, 11, *see also id.*, 5, 7, 14, 17; EX1003, ¶150. Because the P2P index server receives queries from the P2P devices for resources, the P2P index server represents a ***content management server*** connected to the ***P2P device community***. *Id.* The P2P index server also connected to Li’s topology server (***online server***) because the P2P index server can send identified resource P2P nodes to the topology server and receives “nearest neighbor” resource P2P nodes in return. *Id.*

The P2P index server performs “querying to obtain resource P2P nodes having the resource” based on receiving a ***search request***, which means that the P2P index server searches for P2P nodes having the requested resource. EX1007, 11, 14, 17; EX1003, ¶151. Li describes some embodiments where the P2P index server sends the list of resource P2P nodes to the topology server via the requesting P2P node, or directly sends the list of resource P2P nodes to the topology server. EX1007, 8, 10, 16, 17, 18. The topology server returns the one or more nearest neighbor resource P2P nodes to the requesting P2P node via the P2P index server, or directly returns the one or more nearest neighbor resource P2P nodes to the requesting P2P Node. *Id.* In other embodiments, the P2P index server sends the list of the resource P2P nodes to the requesting P2P node, and the requesting P2P node

identifies the one or more “nearest neighbor” resource P2P nodes locally. EX1007, 7, 8, 10, 11, 16.

Li’s P2P index server includes a *search engine* because the P2P index server receives a query request for a certain resource and performs “querying” to identify one or more P2P nodes that include this resource. EX1007, 5, 7, 11, 14, 17. This *search engine* identifies the list of P2P nodes that host the requested resource using information included in the *search request* identifying the requested resource. *Id.* Using a *search engine* of a *content management server* to search for P2P nodes hosting a requested resource was well known by the Critical Date of the ’655 Patent, as confirmed by multiple references that describe such search engines. EX1003, ¶152 (citing corroborating, EX1014, [0055], EX1015, 8:60-9:4).

Li’s P2P index server (*content management server*) also includes an *index database* for identifying P2P nodes storing a requested resource. EX1007, 5, 7, 10, 11, 14, 17. Li explains that the P2P index server performs “querying to obtain a list of nodes having the resource,” which means that the P2P index server includes an *index database* identifying the resources stored by each P2P node. *Id.* Multiple references confirm that it was well known by the Critical Date of the ’655 Patent for a *content management server* to include an *index database* indicating contents stored by one or more P2P devices. EX1003, ¶153 (citing corroborating, EX1016, [0039], EX1017, [0038]).

Additionally, the P2P index server includes a *user's device management unit* because the P2P index server manages P2P nodes that output query requests for resources. EX1007, 5, 7, 11, 14, 17; EX1003, ¶154. For example, the “requesting P2P node” of Li represents a user's device that sends a “resource query request” to the P2P index server. *Id.* Based on this resource query request, the P2P index server works with the topology server to generate a search result for output to the requesting P2P node, which means that the P2P index server *manages* the manner in which the user's device obtains resources. *Id.*

Element [1d]

The P2P index server includes an *index database* that records the *indexed data* corresponding to each P2P node that connects to the metropolitan area network because the P2P index server obtains “a list” of P2P nodes having a resource in response to receiving a resource query request identifying the resource. EX1007, 5, 7, 10, 11, 14, 17; EX1003, ¶155. To obtain this list of P2P nodes, the P2P index server differentiates between P2P nodes storing the resource and P2P nodes not storing the resource so that only P2P nodes storing the resource are included in the list. *Id.*

Li expressly discloses that the P2P index server (*content management server*) represented by the P2P index server *receives* a *search request* from a *user's device*. EX1007, 5, 7, 8, 10, 11, 14, 17; *supra*, Elements [1pre], [1d];

EX1003, ¶156. The *search engine* of Li's P2P index server generates a *search result* by searching the *index database* of the P2P index server to obtain "a list of P2P nodes having [a] resource" in response to receiving a "resource query request" identifying the requested resource. EX1007, 5, 7, 8, 10, 11, 14, 17. To obtain this list, the *search engine* consults the *index database* to differentiate between P2P nodes that host the requested resource and P2P nodes that do not host the requested resource by consulting the *index database*. *Id.*

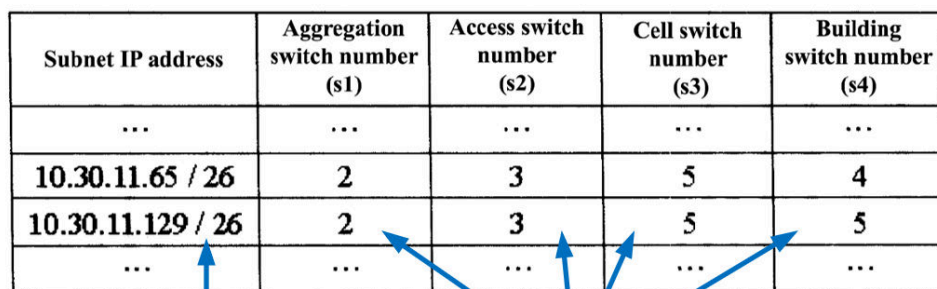
Li explains that in some embodiments, the P2P index server *returns* the *search result* including the list of resource P2P nodes to the requesting P2P node, and the requesting P2P node sends the *search result* including the list of resource P2P nodes to the topology server. EX1007, 7, 11, 17. Next, the topology server identifies "one or more nearest neighbor resource P2P nodes" and returns the one or more nearest neighbor resource P2P nodes to the requesting P2P node either directly or via the P2P index server. EX1007, 8, 10, 11, 16-18. In another embodiment, the P2P index server *returns* the *search result* including the list of resource P2P nodes to the requesting P2P device together with a location vector corresponding to each resource P2P node in the list of resource P2P nodes, and the requesting P2P device determines one or more nearest neighbor resource P2P nodes based on the location vectors. EX1007, 7, 8, 10, 11, 16. In each of these embodiments, a *search engine* of Li's P2P index server generates a *search result*

including a list of resource P2P nodes based on the *search request* and returns this *search result* to the requesting P2P node. EX1007, 7, 8, 10, 11, 16-18; EX1003, ¶¶157-158.

Element [1e]

Li explains that in some embodiments, a location vector includes four values, including an “s1” value corresponding to an aggregation switch, an “s2” value corresponding to an access switch, an “s3” value corresponding to a cell switch, and an “s4” value corresponding to a building switch. EX1007, 6, 9, 15, FIG. 4, FIG. 5; ; EX1003, ¶159. Because P2P nodes connect to building switches, the location vector $S = (s1, s2, s3, s4)$ corresponding to any two P2P nodes indicate *link data* for how those P2P nodes are connected to one another in the metropolitan area network. *Id.*

Topology data table



Subnet IP address	Aggregation switch number (s1)	Access switch number (s2)	Cell switch number (s3)	Building switch number (s4)
...	...	...	...	...
10.30.11.65 / 26	2	3	5	4
10.30.11.129 / 26	2	3	5	5
...	...	...	...	...

FIG. 5

**Subnet IP address
of each P2P node**

Link data

EX1007, FIG. 5 (annotated).

The *link data* recorded in the topology data table is used by the topology server or the requesting P2P node to identify “one or more nearest neighbor resource P2P nodes” based on the “list of resource P2P nodes” identified by the P2P index server as hosting the requested resource. EX1007, 7, 10, 11, 14, 16; EX1003, ¶160. To determine the one or more nearest neighbor resource P2P nodes, the topology server or the requesting P2P node uses a “calculation method” to compare the location vector of the requesting P2P node with the location vector of each identified resource P2P node. EX1007, 6, 9. For example, Li explains that the topology server or the requesting P2P node can calculate a distance vector (e.g., $D = (d1, d2, d3, d4)$) by subtracting the location vector of a resource P2P node (e.g., $S' = (s', s', s', s'4)$) from the location vector of the requesting P2P node (e.g., $S = (s1, s2, s3, s4)$). EX1007, 6, 8, 9, 10, 16. These distance vectors, in addition to the location vectors, represent *link data*. *Id.*

Determining the one or more nearest neighbor resource P2P nodes next involves comparing the distance vectors D with one another to determine the resource P2P nodes that are “nearest” the requesting P2P node within the metropolitan area network. EX1007, 6, 7, 9, 16; EX1003, ¶161. This list of one or more nearest neighbor resource P2P nodes also represents *link data*, because the list indicates one or more resource P2P nodes that are closest to the requesting P2P node within the metropolitan area network (e.g., the “nearest connected resource

P2P node”). EX1007, 6-11, 14-17. Li therefore explains that **link data** can include any one or combination of the location vectors corresponding to identified resource P2P nodes, distance vectors corresponding to links between resource P2P nodes, or a list of nearest neighbor resource P2P nodes indicating the closest nodes to the requesting P2P node. *Id.*

In some embodiments, the P2P index server generates a **search result** by performing “querying to obtain the resource P2P nodes having the resource,” returns a **search result** including a list of the identified resource P2P nodes to the requesting P2P device, and sends the **search result** including the list of resource P2P nodes to the topology server. EX1007, 7, 8, 11, 17; *supra*, Element [1d]. As described above, the topology server can identify one or more nearest neighbor P2P nodes based on the location vectors recorded in the **topology data table**, and return **link data** including the one or more nearest neighbor P2P nodes to the requesting P2P node. EX1007, 5, 6, 8-11, 16-18; EX1003, ¶162. The requesting P2P node therefore obtains **link data** comprising one or more nearest neighbor P2P nodes that are determined by the topology server according to the **search result** received from the P2P index server and the location vectors stored by the **topology data table**. *Id.*

In other embodiments, each P2P node reports resources stored on the node and a location vector of the node to the P2P index server. EX1007, 7, 10, 17. The

P2P index server thus can store a copy of the ***topology data table*** in some cases. *Id.* the P2P index server generates a ***search result*** by performing “querying to obtain a list of nodes having the resource,” and returns the ***search result*** to the requesting P2P node including “[t]he list of all resource P2P nodes obtained” that “*comprises a location vector of each resource node.*” EX1007, 8, *see also id.*, 10, 17, 18; *supra*, Element [1d]; EX1003, ¶163. The location vector of each resource P2P node represents ***link data*** determined according to the ***topology data table*** stored by the topology server. *Id.* Based on this ***search result*** indicating the list of resource P2P nodes and the ***link data*** indicating the location vector of each resource P2P node, the requesting P2P node performs the calculation method to determine the one or more nearest neighbor P2P nodes. *Id.* Because Li’s requesting P2P node (***user’s device***) receives a ***search result*** identifying one or more resource P2P nodes, obtains ***link data*** including the location vector of each of these resource P2P nodes, and further obtains ***link data*** by calculating the one or more nearest neighbor P2P nodes, the ***user’s device*** obtains the ***link data*** according to the ***topology data table*** and the ***search result***. *Id.*

Claim [7]

Li describes techniques for “reporting, by each P2P node computer, a location vector of the P2P node computer while reporting a resource owned by the P2P node computer to a P2P index server.” EX1007, 10. Li also explains that the

P2P index server performs “querying” to “obtain P2P nodes having [a] resource” in response to receiving “a resource query request for the resource” from a requesting P2P node. EX1007, 11, *see also id.*, 5, 14, 17; ; EX1003, ¶164. This means that the P2P index server searches an ***index database*** recording the ***indexed data*** for each P2P node so that the P2P index server can distinguish between P2P nodes having the resource and P2P nodes not having the resource. *Id.* The ***content management server*** represented by Li’s P2P index server includes a ***network gathering module*** for gathering ***indexed data*** from each P2P node indicating resources stored by that P2P node, so that the P2P index server can distinguish between the P2P nodes having the resource and P2P nodes not having the resource. *Id.*

Claim [8]

After obtaining ***link data***, the ***user’s device*** represented by Li’s requesting P2P node proceeds with “*connecting to and downloading from the nearest connected resource P2P node.*” EX1007, 11, *see also id.*, 5, 6, 8-10, 15-18, FIGs. 4, 5; *supra*, Element [1e]. This means that the requesting P2P node ***connects*** to the ***P2P device*** corresponding to the resource P2P node and ***downloads file data*** from the P2P device. EX1007, 11, *see also id.*, 5, 8-10 14, 17. One embodiment of Li involves the requesting P2P node obtaining ***link data*** including location vectors for each resource P2P node having a requested resource. EX1007, 8, 17, 18; *supra*,

Element [1e]. In this embodiment, the requesting P2P node determines one or more “nearest neighbor” resource P2P nodes based on the location vectors and proceeds to download the resource from these nodes. EX1007, 8, 17, 18. Li’s *user’s device* therefore connects to one or more resource P2P nodes and *downloads file data* based on *link data*. EX1007, 6, 8-10, 15-18; EX1003, ¶¶165-166.

Claim [9]

Supra, Claim [8]; EX1003, ¶167.

Element [11pre]

Supra, Elements [1pre]-[1c]; EX1003, ¶168.

Element [11a]

Supra, Element [1a]-[1b]; EX1003, ¶169.

Element [11b]

Supra, Elements [1c]-[1d]; EX1003, ¶170.

Element [11c]

Supra, Element [1d]; EX1003, ¶171.

Element [11d]

Supra, Element [1d]; EX1003, ¶172.

Element [11e]

Supra, Element [1e]; EX1003, ¶173.

Claim [16]

Supra, Claim [8]; EX1003, ¶174.

Claim [18]

Supra, Claim [8]; EX1003, ¶175.

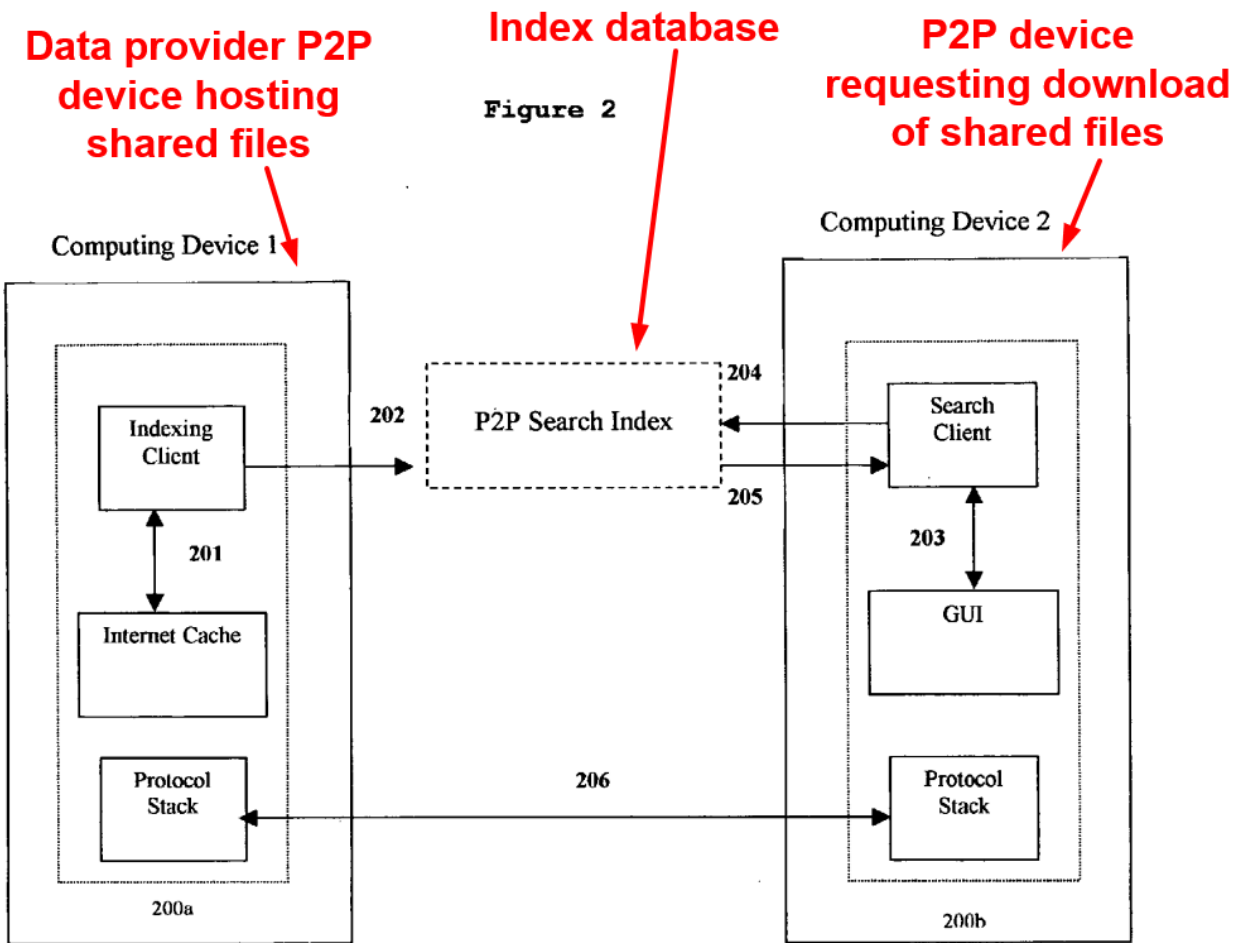
Claim [19]

Supra, Claim [9]; EX1003, ¶176.

E. GROUND 2B: Li in view of Rose would have rendered obvious claims 1, 7-11, and 16-19

1. *Rose*

Rose describes a system for “indexing and searching content” in a “distributed computing network” that represents a “P2P network.” EX1008, [0024], [0029], *see also id.*, [0003]-[0014], [0051]-[0061], FIGS. 1-4; EX1003, ¶59. Rose describes one embodiment where the P2P network includes a “computing device 1” that creates entries for a “comprehensive index of data” and a “computing device 2” that searches the index for a requested file. EX1008, [0038]-[0042], FIG. 2. The search index “searches the index for...links matching the search criteria” and “returns...the appropriate search response” to computing device 1. EX1008, [0040], FIG. 2. Based on this search response, computing device 2 communicates with computing device 1 download the requested file. EX1008, [0040], FIG. 2.



EX1008, FIG. 2 (annotated).

2. The Li-Rose Combination

Even if Li does not expressly disclose a content management server that includes a search engine and an index database recording indexed data of each P2P device of a P2P device community, Rose provides these features. EX1008, [0051]-[0071], FIGS. 3-5; EX1003, ¶60. For example, Rose discloses that a central server can generate an index database that is shared with a distributed search engine (DSE) client, the index database indicating shared files that are hosted by data

providers. *Id.* Multiple reasons would have prompted a POSITA to implement the P2P system of Li with Rose's suggestion for a content management server to maintain an index database that indicates files offered by each data provider P2P node. EX1003, ¶60.

First, a POSITA would have recognized that implementing Li's P2P index server to actively maintain an index database as suggested by Rose would result in the P2P index server accurately identifying P2P devices that store a resource in response to receiving a query identifying the resource. EX1003, ¶61. For example, Rose explains that a central server "maintains a downloadable data index, which is periodically updated" to reflect changes in the status of shared files over time. EX1008, [0052]; EX1003, ¶61. Causing Li's P2P index server to create and periodically update an index as described by Rose would therefore result in the P2P index server obtaining a list of P2P nodes that currently have the requested resource, accounting for any updates to shared files. EX1003, ¶61.

Second, implementing Li's P2P system to use more than one distributed search engine (DSE) client to handle search requests from P2P devices as suggested by Rose would prevent a single search engine from handling every single search request in the P2P network, therefore increasing a likelihood that the P2P system can handle a large number of search requests in a short amount of time. EX1003, ¶62. For example, Li acknowledges that "P2P technology

preemptively applies...network bandwidth...seriously affecting the normal operation of other services.” EX1007, 1; EX1003, ¶62. Using more than one distributed search engine (DSE) client to handle search requests would distribute the burden of handling these search requests, thus addressing the bandwidth issue acknowledged by Li. EX1003, ¶62.

Third, a POSITA would have been motivated to cause Li’s P2P index server to output a search result including an IP address of one or more P2P devices having a requested resource as suggested by Rose, because doing so would provide the requesting P2P node with information useful for connecting to the one or more P2P devices having the requested resource. EX1003, ¶63. For example, the requesting P2P node can use the IP address to connect to a resource P2P node over the network to conveniently download the resource. EX1003, ¶63.

Finally, it would have been obvious for a POSITA to implement Li’s P2P system with the teachings of Rose for maintaining an index database because doing so represents one of a finite number of predictable solutions for allowing P2P devices to receive content over a P2P network. *KSR at 417*; EX1003, ¶64. A POSITA would have reasonably expected success implementing the Li-Rose combination because P2P networks and content management servers for managing content search requests from P2P devices were well known before the Critical Date of the ’655 Patent. *KSR at 401*; EX1003, ¶64.

3. *Application to Challenged Claims*

Element [1pre]

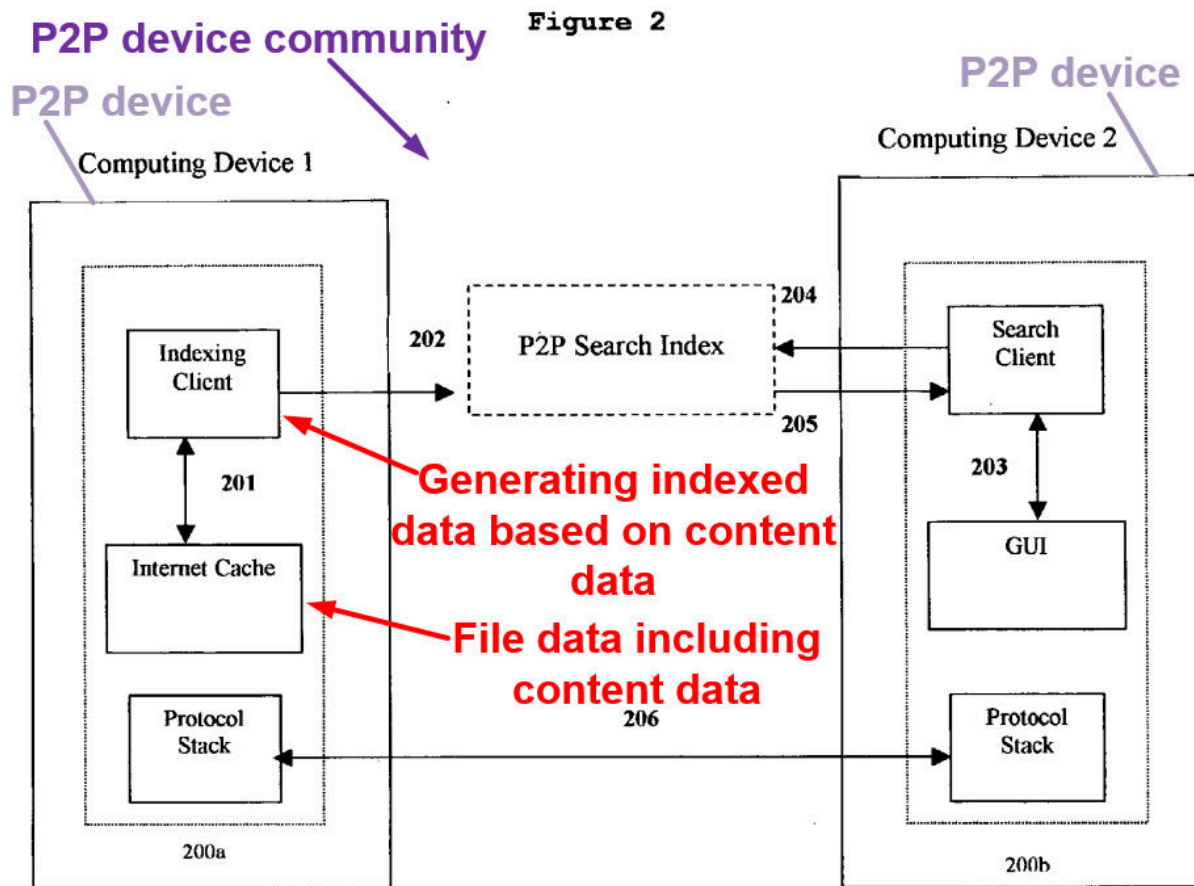
Supra, Ground 2A (§V.D.2) at Element [1pre]. EX1003, ¶177.

Element [1a]

Rose describes a ***P2P device community*** including one or more “computing devices” that serve as ***P2P devices***. EX1008, [0024]-[0028], [0038]-[0040], [0043], FIG. 1, FIG. 2. For example, Rose explains that the computing devices can “share files located on their own hard drives as well as to search and download files from other computers,” meaning that each computing device stores ***file data*** including ***content data*** downloaded by other computing devices. EX1008, [0030], *see also id.*, [0044]; EX1003, ¶¶178-179. Furthermore, Rose explains that each computing device includes an “indexing module” that can “periodically search a plurality of folder(s)” on the P2P device “to identify cached versions of data having network addresses associated thereto.” EX1008, [0027], [0028], *see also id.*, [0038].

Rose also explains that the indexing module “extracts identifying information from the files, extracts network address information from the files, and generates an index,” meaning that the computing device creates ***indexed data*** based on the ***content data*** of each file. EX1008, [0029], *see also id.*, [0038]; EX1003, ¶180. In the predictable Li-Rose combination, each ***P2P device*** in the

P2P device community therefore stores *file data* comprising *content data* and *indexed data*. EX1008, [0024]-[0030], [0038]-[0040], [0043], FIG. 1, FIG. 2; *supra*, §V.E.2.



EX1008, FIG. 2 (annotated).

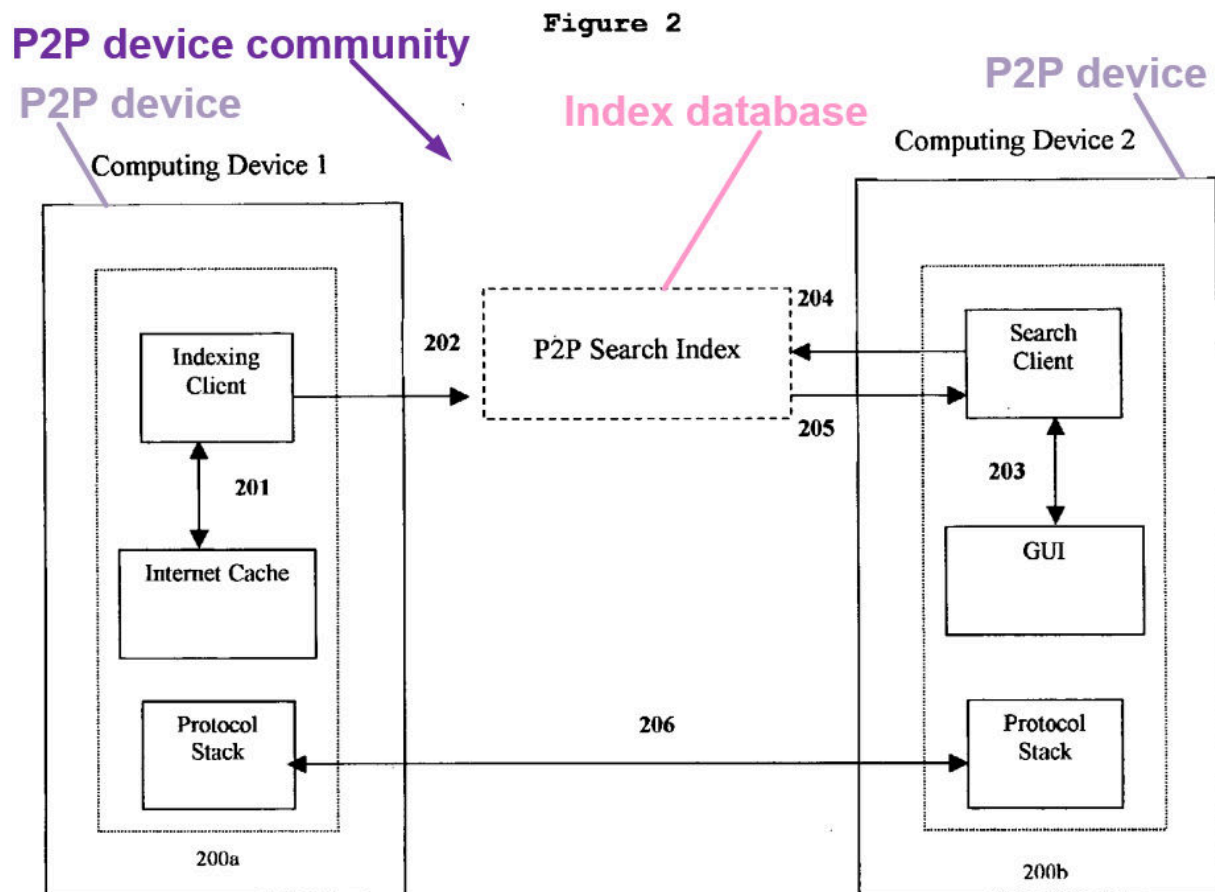
Element [1b]

Supra, Ground 2A (§V.D.2) at Element [1b]. EX1003, ¶181.

Element [1c]

Rose describes one embodiment where each computing device representing a *P2P device* in a *P2P device community* generates *indexed data* for one or more

files that are available for downloading by other computing devices. EX1008, [0024]-[0030], [0038]-[0040], [0043], FIG. 1, FIG. 2; *supra*, Element [1b]. Rose expressly discloses that the computing device outputs this ***indexed data*** to a “search index of a P2P network...to integrate the indexed cached data into a larger local or regional P2P index.” EX1008, [0029], *see also id.*, [0006], [0007], [0038]; EX1003, ¶¶182-183. Rose explains that the search index can be stored on a “regional node” that is separate from the computing device. EX1008, [0006], [0007], [0038], FIG. 2.



EX1008, FIG. 2 (annotated).

Because the “larger local or regional P2P index” of Rose is stored separately from the *P2P devices* in the *P2P device community* and incorporates indexed data from multiple computing devices, this index represents an *index database* that can be stored by a *content management server* connected to the *P2P device community*. EX1008, [0006], [0007], [0026]-[0029], [0038]-[0041], FIG. 1, FIG. 2. Rose also describes the “search index” as including “an associated software application that searches the index,” meaning that the regional node also includes a *search engine*. EX1008, [0040], *see also id.*, [0029], [0030]. The *content management server* of the Li-Rose combination therefore includes a an *index database* and a *search engine*. EX1008, [0006], [0007], [0026]-[0029], [0038]-[0041], FIGs. 1, 2; *supra*, §V.E.2; EX1003, ¶¶184-185.

Element [1d]

The *indexed data* recorded by the indexed database includes information identifying one or more cached files stored by each computing device of Rose’s *P2P device network*. EX1008, [0006], [0007], [0026]-[0029], [0038]-[0041], FIG. 1, FIG. 2. For example, Rose explains that a P2P device includes an “indexing module” that can “periodically search a plurality of folder(s)” on the P2P device “to identify cached versions of data having network addresses associated thereto.” EX1008, [0027], [0028], *see also id.*, [0038]. Rose also explains that the indexing module “extracts identifying information from the files, extracts network address

information from the files, and generates an index.” EX1008, [0029], *see also id.*, [0038]. In the predictable Li-Rose combination, the ***indexed database*** therefore records indexed data of each ***P2P device*** of the ***P2P device community***. *Supra*, §V.E.2; EX1003, ¶186.

Element [1e]

Supra, Ground 2A (§V.D.2) at Element [1e]. EX1003, ¶187.

Claim [7]

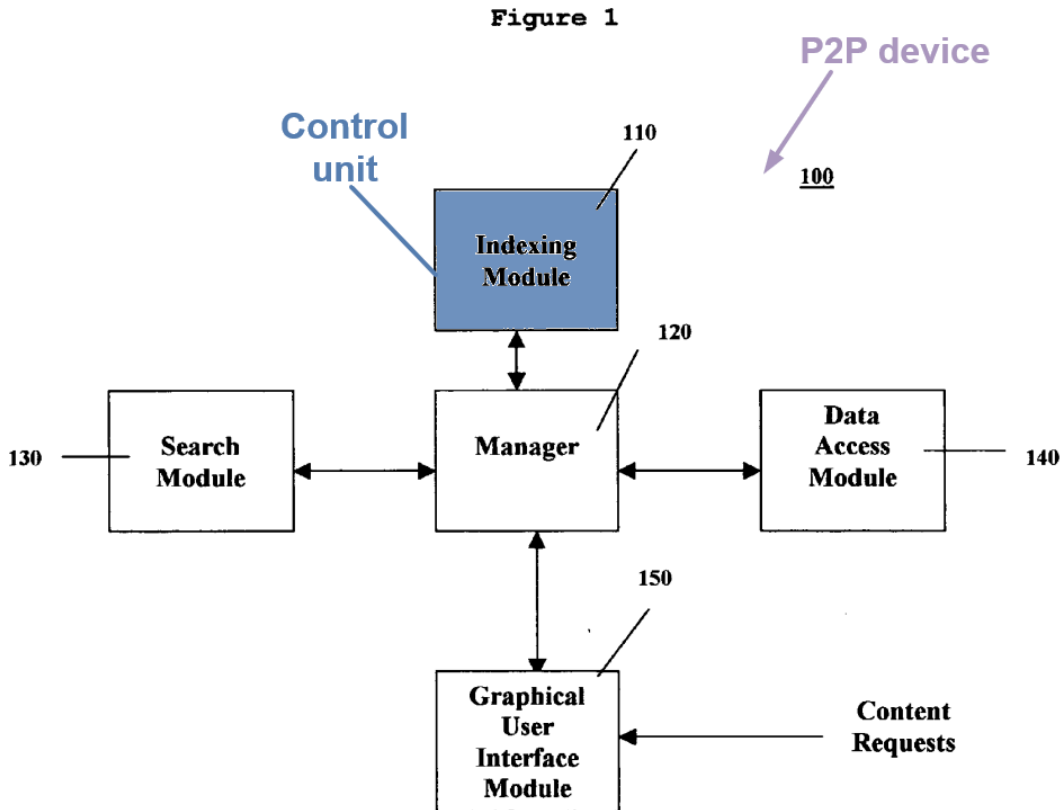
To the extent that Li does not expressly disclose that the ***P2P devices*** or the ***content management server*** generates the ***indexed data***, such features would have been obvious and provided in the Li-Rose combination. EX1008, [0026]-[0029], [0038], [0039], FIG. 1, FIG. 2; *supra*, §V.E.2; EX1003, ¶188. Because Rose expressly discloses ***P2P devices*** within a P2P device community each include an “indexing module” that generates ***indexed data*** for files stored on the P2P device by extracting “information in the file, including words, images, links, source, or other information,” Rose explains that the ***indexed data*** can be created by the ***P2P devices*** themselves. EX1008, [0029], *see also id.*, [0027], [0028], [0038], [0039].

Claims [8]-[9]

The predictable Li-Rose combination would provide the Elements recited here because Li’s system discloses those features as detailed above. *Supra*, Ground 2A, Claims [8]-[9]; EX1003, ¶189.

Claim [10]

Rose explains that each **P2P device** includes an “indexing module” that generates the **indexed data** for cached data files stored by the computing device. EX1008, [0027]-[0029], [0038], [0039]. Because the indexing module manages the indexing of the cached data files, the indexing module represents a **control unit** that controls the **file data** for the **P2P device** corresponding to the computing device. EX1008, [0027]-[0029], [0038], [0039]; EX1003, ¶190.



EX1008, FIG. 1 (annotated).

Rose explains that each computing device “provides access to *unrestricted* sources of content and, if desired, *closed* or *restricted* access as well,” meaning

that the ***control unit*** controls ***file data*** to occupy an ***open state*** or a ***hidden state***.

EX1008, [0048], *see also id.*, [0013], [0049], [0050]; EX1003, ¶191. Rose expressly discloses that “the method of indexing cached network files includes the step of ignoring personal identification information of a user or deleting personal identification information of a user,” which means that the P2P device’s ***control unit*** controls ***file data*** revealing personal information of the user to be in the ***hidden state***. EX1008, [0013]. Rose also explains that the ***control unit*** causes personal information to occupy the ***hidden state*** “by not allowing such information to be indexed and thereby become a part of the global distributed index,” which is very similar to the manner in which the ’655 Patent controls file data to be in the hidden state. EX1008, [0049]; *cf.* EX1001, 7:11-27.

Element [11pre]

Supra, Elements [1pre]-[1c]; EX1003, ¶192.

Element [11a]

Supra, Element [1a]-[1b]; EX1003, ¶193.

Element [11b]

Supra, Elements [1c]-[1d]; EX1003, ¶194.

Element [11c]

Supra, Element [1d]; EX1003, ¶195.

Element [11d]

Supra, Element [1d]; EX1003, ¶196.

Element [11e]

Supra, Element [1e]; EX1003, ¶197.

Claim [16]

Supra, Claim [7]; EX1003, ¶198.

Claim [17]

Supra, Claim [10]; EX1003, ¶199.

Claim [18]

Supra, Claim [8]; EX1003, ¶200.

Claim [19]

Supra, Claim [9]; EX1003, ¶201.

F. GROUND 2C: Li, in view of Rose, and further in view of Schwan would have rendered obvious claims 2-4, 12, and 13

1. *The Li-Rose-Schwan Combination*

Li describes “P2P index server” that obtains a list of P2P nodes having a requested resource and a “topology server” that determines a “nearest neighbor” P2P nodes based on a topology data table. EX1007, 4-11, 14-17, FIG. 1, FIG. 4, FIG. 5; EX1003, ¶65. Even if Li does not expressly disclose that the P2P index server includes a sorting algorithm module for creating weighted data based on link data stored in a topology data table of the P2P index server, such features would have been obvious and provided in the Li-Rose-Schwan combination.

EX1004, [0024]-[0036], FIG. 1, FIG. 2. Multiple reasons would have prompted a POSITA to implement the P2P system of the Li-Rose combination with Schwan's suggestion to use a content management server to sort search results based on a topology data table stored by an online server. EX1003, ¶65.

First, a POSITA would have been motivated to implement the P2P system of the Li-Rose combination so that the P2P index server can determine the nearest neighbor P2P nodes in addition to the topology server, because doing so would provide a server that conveniently searches for content and refines the search based on topology data stored on another server. EX1003, ¶66. *Second*, implementing the nearest neighbor analysis of the predictable Li-Rose combination with Schwan's suggestion to sort search results based on latency would provide another way to prioritize resource P2P nodes that are close to the requesting P2P node within the P2P network. EX1003, ¶66. *Third*, because Schwan discloses techniques for managing a P2P device community spread across multiple different ISPs, a POSITA would have been motivated to apply Schwan's content management techniques to the P2P system of the Li-Rose combination to increase a number of P2P devices that can connect. EX1003, ¶66.

Finally, it would have been obvious for a POSITA to implement the P2P system of the Li-Rose combination with the teachings of Schwan for sorting a search result using a content management server because doing so represents one

of a finite number of predictable solutions for managing search requests in a P2P network. *KSR* at 417. EX1003, ¶67. A POSITA would have reasonably expected success implementing the Li-Rose-Schwan combination because P2P management servers storing indexed databases and online servers storing topology data tables were well known before the Critical Date of the '655 Patent. *KSR* at 401; EX1003, ¶67.

2. *Application to Challenged Claims*

Claim [2]

Li explains one way to determine the “nearest neighbor” resource P2P nodes is to calculate a “distance vector $D = (d1, d2, d3, d4)$ ” for each resource P2P node by subtracting the “location vector $S' = (s'1, s'2, s'3, s'4)$ ” of the resource P2P node from the “location vector $S = (s1, s2, s3, s4)$ ” of the requesting P2P node. EX1007, 9, *see also id.*, 6, 16; EX1003, ¶202. The distance vector between each resource P2P node and the requesting P2P node represents ***weighted data*** for that resource P2P node, because these distance vectors indicate how close each resource P2P node is to the requesting P2P node within the metropolitan area network. *Id.* Because the distance vectors are calculated using ***link data*** including the location vector of the requesting P2P node and the location vectors of each resource P2P node identified by the P2P index server, Li describes a ***sorting algorithm module*** that creates the ***weighed data*** based on ***link data***. *Id.*

To the extent Li's *content management server* represented by the "P2P index server" does not include a *sorting algorithm module* for creating the distance vectors, it would have been obvious to implement the P2P index server with this feature based on the suggestions of Schwan. EX1007, [0028]-[0039], FIG. 2; *supra*, § V.F.1; EX1003, ¶203. Because Schwan describes a *content management server* represented by tracker node 20 and an *online server* represented by P2P manager 30 and explains that "peer-to-peer manager 30 communicates [] relevant topology information for the tracker node 20 to directly generate [a] optimized list in accordance with predefined rules," a POSITA would have been motivated to implement Li's P2P index server with a *content management server* for generating the distance vectors. EX1007, [0036]; *supra*, § V.F.1.

Claim [3]

Because Li describes a "calculation method for searching for the nearest neighbor resource P2P node" that involves calculating a distance vector for each resource P2P node and a "size comparison method" for comparing distance vectors to determine the nearest neighbors, Li describes a *weight look-up table* including the information for completing the size comparison method including a table entry corresponding to each distance vector. EX1007, 6, 7, 9, 10, 16, 17; EX1003, ¶204. A POSITA would have been motivated to implement Li's P2P index server with a

weight look-up table recorded with the *weighted data* based on Schwan's suggestion for a *content management server* to generate an optimized list of P2P devices. EX1007, [0036]; *supra*, § V.F.1.

Claim [4]

The “calculation method” of Li represents a *function operation* that is performed to *sort* a list of resource P2P nodes identified by the P2P index server as including a requested resource. EX1007, 6, 7, 9, 10, 16, 17; EX1003, ¶205. This *function operation* involves comparing the distance vectors identified by the *sorting algorithm module*. *Id.* Specifically, the function operation includes a “size comparison method” for determining the larger of a pair of distance vectors “ $D = (d_1, d_2, d_3, d_4)$ and $D' = (d'_1, d'_2, d'_3, d'_4)$.” *Id.* Li explains that this *function operation* can be used to determine a *weighted and sorted search result* including the nearest neighbor P2P nodes sorted by size so that the smallest distance vectors are the “nearest.” *Id.*

Claim [12]

Supra, Claims [2] and [4]; EX1003, ¶206.

Claim [13]

Supra, Claim [3]; EX1003, ¶207.

G. GROUND 2D: Li, in view of Rose, and further in view of Strickland would have rendered obvious claims 5, 6, 14, and 15

1. *The Li-Rose-Strickland Combination*

A POSITA would have found it obvious and would have been motivated to further modify the content management server of the Li-Rose combination based on Strickland's suggestion to include a user charge module to achieve the first, second, and third known benefits previously articulated above in Ground 1C. *Supra*, § V.C.2; EX1003, ¶68.

2. *Application to Challenged Claims*

Claims [5], [6], [14], [15]

Strickland discloses the elements of Claims [5], [6], [14], and [15] for the reasons described above in connection with the analysis of Claims [5], [6], [14], and [15] in Ground 1C. *Supra*, § V.C.2; EX1003, ¶208.

VI. PAYMENT OF FEES – 37 C.F.R. §42.103

Reolink authorizes the Patent and Trademark Office to charge Deposit Account No. 06-1050.

VII. MANDATORY NOTICES UNDER 37 C.F.R §42.8(a)(1)

A. Real Party-In-Interest Under 37 C.F.R. §42.8(b)(1)

Reolink Innovation Inc. and Shenzhen Reolink Technology Co., Ltd. (“Reolink”) are the Petitioners and real parties-in-interest. No other party had access or control over this Petition, and no other party provided funding for this Petition.

B. Related Matters Under 37 C.F.R. §42.8(b)(2)

Petitioners are not aware of any disclaimers, reexamination certificates, or other petitions for IPR of the '655 Patent. The '655 Patent is the subject of civil action captioned *ThroughTEK Co. Ltd. v. Reolink Innovation Inc. et al.*, Case No. 1:24-cv-01027 (D. Del.), which remains in the pleadings stage and without the start of discovery or a scheduled jury trial. See EX1018 (complaint filed Sep. 12, 2024).

C. Lead And Back-Up Counsel Under 37 C.F.R. §42.8(b)(3)

Reolink provides the following designation of counsel.

Lead Counsel	Backup Counsel
Michael Hawkins, Reg. No. 57,867 Fish & Richardson P.C. 60 South Sixth Street, Suite 3200 Minneapolis, MN 55402 Tel: 612-337-2569 Fax: 877-769-7945 Email: IPR52472-0009IP1@fr.com	Nicholas W. Stephens, Reg. No. 74,320 Cameron Ubel, Reg. No. 77,081 Joseph Bauer, Reg. No. 81,218 Fish & Richardson P.C. 60 South Sixth Street, Suite 3200 Minneapolis, MN 55402 Tel: 612-335-5070 Fax: 877-769-7945 Email: IPR52472-0009IP1@fr.com

D. Service Information

Please address all correspondence and service to the address listed above.

Petitioner consents to electronic service by email at IPR52472-0009IP1@fr.com.

Respectfully submitted,

Dated: July 18, 2025

/Michael T. Hawkins/
Michael Hawkins, Reg. No. 57,867
Nicholas W. Stephens, Reg. No. 74,320
Cameron Ubel, Reg. No. 77,081
Joseph Bauer, Reg. No. 81,218
Fish & Richardson P.C.
60 South Sixth Street
Minneapolis, MN 55402
T: 612-335-5070
F: 877-769-7945

(Control No. IPR2025-01327)

Attorneys for Petitioners

CERTIFICATION UNDER 37 CFR §42.24

Under the provisions of 37 CFR §42.24(d), the undersigned hereby certifies that the word count for the foregoing Petition for *Inter partes* Review totals 13,883 words, which is less than the 14,000 allowed under 37 CFR §42.24.

Dated: July 18, 2025

/Michael T. Hawkins/
Michael Hawkins, Reg. No. 57,867
Nicholas W. Stephens, Reg. No. 74,320
Cameron Ubel, Reg. No. 77,081
Joseph Bauer, Reg. No. 81,218
Fish & Richardson P.C.
60 South Sixth Street
Minneapolis, MN 55402
T: 202-783-5070
F: 877-769-7945

Attorneys for Petitioner

CERTIFICATE OF SERVICE

Pursuant to 37 CFR §§42.6(e)(4)(i) *et seq.* and 42.105(b), the undersigned certifies that on July 18, 2025, a complete and entire copy of this Petition for *Inter partes* Review, Power of Attorney, and all supporting exhibits were provided via Federal Express, to the Patent Owner by serving the correspondence address of record as follows:

Litron Patent and Trademark Office
11F, No.287, Sec. 3, Nanjing E. Rd.
Songshan District
Taipei City - 10595
TAIWAN

/Crena Pacheco/

Crena Pacheco
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
pacheco@fr.com