

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMERICAN AIRLINES, INC. AND SOUTHWEST AIRLINES CO.,

Petitioners

v.

INTELLECTUAL VENTURES I LLC,

Patent Owner.

IPR2025-00987
U.S. Patent No. 8,407,722

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 8,407,722**

TABLE OF CONTENTS

TABLE OF AUTHORITIES	iv
EXHIBIT LIST	v
I. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8.....	1
A. Real Parties-in-Interest Under § 42.8(b)(1)	1
B. Related Matters Under § 42.8(b)(2)	1
C. Lead and Backup Counsel Under § 42.8(b)(3)	3
D. Service Information Under § 42.8(b)(4)	4
II. GROUNDS FOR STANDING.....	5
III. IDENTIFICATION OF CHALLENGES AND RELIEF REQUESTED	5
IV. THE '722 PATENT.....	6
A. Overview	6
B. Level of Ordinary Skill	10
V. CLAIM CONSTRUCTION	10
A. Claims 17-19, 23-25, 29-31, 35-37: “activation module”.....	10
VI. OVERVIEW OF THE ART	12
A. Lawson (Ex.1005)	13
B. Tsutsumitake (Ex.1006)	15
C. Choquier (Ex.1007)	16
D. Bird (Ex.1008).....	19
E. Herz (Ex.1009)	21
VII. GROUND 1: CLAIMS 1-37 ARE OBVIOUS OVER LAWSON, TSUTSUMITAKE, AND CHOQUIER	22

A.	Motivations to Combine.....	22
B.	Lawson, Tsutsumitake, and Choquier meet all claim elements.....	24
1.	Independent Claims 1, 7, 14, 20, 26, and 32.....	24
2.	Claims 2 and 8.....	49
3.	Claims 3 and 9.....	51
4.	Claims 4 and 10.....	52
5.	Claims 5 and 11.....	53
6.	Claim 6.....	53
7.	Claim 12.....	55
8.	Claim 13.....	56
9.	Claims 15, 21, 27, and 33.....	56
10.	Claims 16, 22, 28, and 34.....	57
11.	Claims 17, 23, 29, and 35.....	58
12.	Claims 18, 24, 30, and 36.....	60
13.	Claims 19, 25, 31, and 37.....	62
VIII.	GROUND 2: CLAIMS 3, 5, 9, 11, 12, 15, 21, 27, AND 33 ARE OBVIOUS OVER LAWSON, TSUTSUMITAKE, CHOQUIER, AND HERZ.....	63
A.	Motivations to Combine.....	63
B.	Lawson, Tsutsumitake, Choquier, and Herz meet all claim elements.....	64
IX.	GROUND 3: CLAIMS 1-37 ARE OBVIOUS OVER CHOQUIER, BIRD, AND TSUTSUMITAKE.....	65
A.	Motivations to Combine.....	65

B.	Choquier, Bird, and Tsutsumitake meet all claim elements.	68
1.	Independent Claims 1, 7, 14, 20, 26, and 32.....	68
2.	Claims 2 and 8.....	85
3.	Claims 3 and 9.....	86
4.	Claims 4 and 10.....	87
5.	Claims 5 and 11.....	87
6.	Claim 6.....	88
7.	Claim 12.....	89
8.	Claim 13.....	90
9.	Claims 15, 21, 27, and 33	91
10.	Claims 16, 22, 28, and 34	91
11.	Claims 17, 23, 29, and 35	92
12.	Claims 18, 24, 30, and 36	93
13.	Claims 19, 25, 31, and 37	94
X.	GROUND 4: CLAIMS 3, 5, 9, 11, 12, 15, 21, 27, AND 33 ARE OBVIOUS OVER CHOQUIER, BIRD, TSUTSUMITAKE, AND HERZ	96
A.	Motivations to Combine.....	96
B.	Choquier, Bird, Tsutsumitake, and Herz meet all claim elements.....	97
XI.	THERE ARE NO DISCRETIONARY FACTORS FAVORING DENIAL OF INSTITUTION	98
A.	The <i>Fintiv</i> factors either favor institution or are neutral.....	98
B.	An analysis under 35 U.S.C. § 325(d) also favors institution.....	100
XII.	CONCLUSION.....	101

TABLE OF AUTHORITIES

Cases

<i>Ericsson Inc. v. XR Commc'ns LLC</i> , IPR2024-00613, Paper 9 (PTAB Oct. 9, 2024)	99
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005).....	10
<i>Sand Revolution II, LLC v. Continental Intermodal Grp.-Trucking LLC</i> , IPR2019-01393, Paper 24 (PTAB. June 16, 2020) (informative)	98
<i>Sotera Wireless, Inc. v. Masimo Corp.</i> , IPR2020-01019, Paper 12 (PTAB Dec. 1, 2020) (precedential)	99
<i>Toyota Motor Corp. v. Cellport Systems, Inc.</i> , IPR2015-00633, Paper 11 at 16 (Aug. 14, 2015)	10
<i>Wellman, Inc. v. Eastman Chemical Co.</i> , 642 F.3d 1355 (Fed. Cir. 2011).....	10

Statutes

35 U.S.C. § 325(d)	100
--------------------------	-----

Rules

37 C.F.R. § 42.104(a).....	5
37 C.F.R. §42.100	10

EXHIBIT LIST

Exhibit No.	Description of Document
1001	U.S. Patent No. 8,407,722
1002	Prosecution File History of U.S. Patent No. 8,407,722 (Application No. 11/396,251)
1003	Expert Declaration of Dr. William C. Easttom II (Chuck Easttom), Ph.D., D.Sc.
1004	Curriculum Vitae of Dr. William C. Easttom, Ph.D., D.Sc.
1005	U.S. Patent No. 5,721,825, entitled “System and method for global event notification and delivery in a distributed computing environment,” filed 10/3/1996 and issued 2/24/1998, issued to Lawson et al. (“Lawson”)
1006	U.S. Patent No. 6,480,883, entitled “Real-Time Information Transmission System,” filed 6/29/1999, issued 11/12/2002, issued to Tsutsumitake (“Tsutsumitake”)
1007	U.S. Patent No. 5,951,694, entitled “Method of redirecting a client service session to a second application server without interrupting the session by forwarding service-specific information to the second server,” filed 2/3/1997, issued 9/14/1999 to Choquier et al. (“Choquier”)
1008	European Patent Application Publication No. EP1043671, entitled “Message broker providing a publish/subscribe service and method of processing messages in a publish/subscribe environment,” filed 3/17/2000, published 10/11/2000 to Bird et al. (“Bird”)
1009	U.S. Patent No. 5,754,939, entitled “System for generation of user profiles for a system for customized electronic identification of desirable objects,” filed 10/31/1995, issued 5/19/1998 to Herz et al. (“Herz”)
1010	File History for U.S. Patent Application No. 60/256,613
1011	File History for U.S. Patent Application No. 60/276,847

Petition for *Inter Partes* Review of
U.S. Patent No. 8,407,722

Exhibit No.	Description of Document
1012	File History for U.S. Patent Application No. 60/278,303
1013	File History for U.S. Patent Application No. 60/279,608
1014	File History for U.S. Patent Application No. 60/280,627
1015	File History for U.S. Patent Application No. 10/017,182
1016	File History for U.S. Patent Application No. 10/105,018
1017	Notice of Intent to Issue Reexamination Certificate, dated March 11, 2025, from Reexamination of U.S. Patent No. 08,407,722 (Control No. 90/019,662)
1018	U.S. Patent No. 6,230,173
1019	U.S. Patent No. 5,907,837
1020	U.S. Patent No. 6,199,082
1021	Claim Listing
1022	Complaint, <i>Intellectual Ventures I LLC et al. v. Southwest Airlines Co.</i> , No. 7:24-cv-00277 (W.D. Tex. Nov. 2, 2024)
1023	Complaint, <i>Intellectual Ventures I LLC et al. v. American Airlines, Inc.</i> , No. 4:24-cv-980-ALM (E.D. Tex. Nov. 2, 2024)
1024	Claim Chart Exhibit to Complaint, <i>Intellectual Ventures I LLC et al. v. Southwest Airlines Co.</i> , No. 7:24-cv-277 (W.D. Tex. Nov. 2, 2024)
1025	Cornell University, Nodes; Parallel Programming Concepts and High Performance Computing (https://cvw.cac.cornell.edu/parallel/hpc/nodes)
1026	Northeastern, ECE 333: Introduction Communication Networks Fall 2002 (http://users.ece.northwestern.edu/~rberry/ECE333/Lectures/lec1.pdf)
1027	Zen and the Art of the Internet, Glossary (https://www.ou.edu/research/electron/internet/zen-glos.htm)

Exhibit No.	Description of Document
1028	Microsoft Computer Dictionary, 5th Edition, 2002
1029	Internetworking Technologies Handbook, Routing Basics, Chapter 5 (http://magrawal.myweb.usf.edu/dcom/Ch9_Routing-Basics.pdf)
1030	Juniper Networks, Inc., Routing Protocols Overview (https://www.juniper.net/documentation/us/en/software/junos/routing-overview/routing-overview.pdf)
1031	CloudFlare, What is Routing? IP routing https://www.cloudflare.com/learning/network-layer/what-is-routing/)
1032	Barrons, Dictionary of Computer and Internet Terms, 10th Edition
1033	NYU Networks at ITP, Proxies (https://itp.nyu.edu/networks/explanations/proxies/)

Petitioners American Airlines, Inc. (“American”) and Southwest Airlines Co. (“Southwest”) (collectively “Petitioners”) petition for *inter partes* review (“IPR”) of claims 1-37 of U.S. Patent No. 8,407,722 (“’722 Patent”) owned by Intellectual Ventures I LLC (“Patent Owner”).

I. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8

A. Real Parties-in-Interest Under § 42.8(b)(1)

American is a real party in interest to this petition. American is a wholly owned subsidiary of American Airlines Group, Inc., which is not a real party in interest to this petition. No unnamed entity is funding, controlling, or otherwise has an opportunity to control or direct this petition or American’s participation in any resulting IPR.

Southwest is also a real party in interest to this petition. No unnamed entity is funding, controlling, or otherwise has an opportunity to control or direct this petition or Southwest’s participation in any resulting IPR.

B. Related Matters Under § 42.8(b)(2)

To the best of Petitioners’ knowledge, the ’722 Patent has been involved in the following related matters:

- *Intellectual Ventures I LLC et al. v. Liberty Mutual Holding Company Inc., et al.*, No. 23-cv-525 (E.D. Tex. filed November 15, 2023)

- *Intellectual Ventures I LLC et al. v. Comerica Incorporated*, No. 23-cv-524 (E.D. Tex. filed November 15, 2023)
- *Intellectual Ventures I LLC et al. v. JP Morgan Chase Bank, National Association et al.*, No. 23-cv-523 (E.D. Tex. filed November 15, 2023)

The actions against Liberty, Comerica, and JP Morgan has been dismissed with prejudice.

The '722 Patent was also at issue in the following IPR proceeding:

- IPR2025-00200 – terminated/settled prior to institution

The '722 Patent is also asserted in the following cases:

- *Intellectual Ventures I LLC et al. v. American Airlines, Inc.*, No. 24-cv-980 (E.D. Tex. filed November 2, 2024)
- *Intellectual Ventures I LLC et al. v. Southwest Airlines, Co.*, No. 24-cv-277 (W.D. Tex. filed November 2, 2024)

The '722 Patent was also the subject of an *ex parte* reexamination request filed by Unified Patents, LLC on September 16, 2024 (Control No. 90/019,662). Reexamination was ordered on October 16, 2024. The Patent Office filed a Notice of Intent to Issue a Reexamination Certificate on March 11, 2025 and issued a Reexamination Certificate on April 4, 2025. The claims were not amended and no claims were canceled in the reexamination.

Petitioners are not aware of any other civil actions or proceedings filed in connection with the '722 Patent.

C. Lead and Backup Counsel Under § 42.8(b)(3)

Lead Counsel	Backup Counsel
Keith D. Harden (Reg. No. 74,472) kharden@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616	Daniel E. Venglarik (Reg. No. 39,409) dvenglarik@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616 S. Wallace Dunwoody (Texas Bar No. 24040838 - admission <i>pro hac vice</i> to be requested) wdunwoody@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616 Michael C. Wilson (Texas Bar No. 21704590 - admission <i>pro hac vice</i> to be requested) mwilson@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616
	John B. Campbell (Reg. No. 54,665) jcampbell@McKoolSmith.com McKool SMITH, P.C. 303 Colorado Street, Suite 2100

Petition for *Inter Partes* Review of
U.S. Patent No. 8,407,722

Lead Counsel	Backup Counsel
	Austin, TX 78701 Tel: (512) 692-8730 Fax: (512) 692-8744 Casey Shomaker (Reg. No. 77,998) cshomaker@mckoolsmith.com McKool Smith, P.C. 300 Crescent Court, Suite 1200 Dallas, TX 75201 Tel: (214) 978-4218 Fax: (214) 978-4044 Alan Block (Reg No. 35,450) ablock@mckoolsmith.com McKool Smith, P.C. 300 South Grand Avenue, Suite 2900 Los Angeles, California 90071 Tel: (213) 694-1200 Fax: (213) 694-1234

An executed power of attorney under 37 C.F.R. § 42.10(b) accompanies this Petition.

D. Service Information Under § 42.8(b)(4)

Petitioners consent to electronic service at the above designated Counsel email addresses and at the following email address:

SWA-IV@munckwilson.com.

Service of any documents via post or hand-delivery may be made at: Munck Wilson Mandala LLP, 2000 McKinney Ave., Ste. 1900, Dallas, TX 75201; Telephone: (972) 628-3600; Facsimile: (972) 628-3616.

II. GROUNDS FOR STANDING

Pursuant to 37 C.F.R. § 42.104(a), Petitioners certify the '722 Patent is available for IPR, and Petitioners are not barred or estopped from requesting IPR challenging the claims on the grounds identified in this Petition. This Petition is filed within one year of service of the Complaints. *See* Exs. 1022, 1023.

III. IDENTIFICATION OF CHALLENGES AND RELIEF REQUESTED

Petitioners respectfully request review and cancellation of claims 1-37 on the following prior art and statutory grounds:

Prior Art	
U.S. 5,721,825 (Lawson), issued 2/24/1998 (Ex.1005)	Pre-AIA 35 U.S.C. §§ 102(a), (b), and/or (e)
U.S. 6,480,883 (Tsutsumitake), filed 6/29/1999, issued 11/12/2002 (Ex.1006);	Pre-AIA 35 U.S.C. § 102(e)
U.S. 5,951,694 (Choquier), issued 9/14/1999 (Ex.1007);	Pre-AIA 35 U.S.C. § 102(a), (b), and/or (e)
EP1043671 (Bird), published 10/11/2000 (Ex.1008)	Pre-AIA 35 U.S.C. § 102(a)
U.S. 5,754,939 (Herz), issued 5/19/1998 (Ex.1009)	Pre-AIA 35 U.S.C. §§ 102(a), (b), and/or (e)

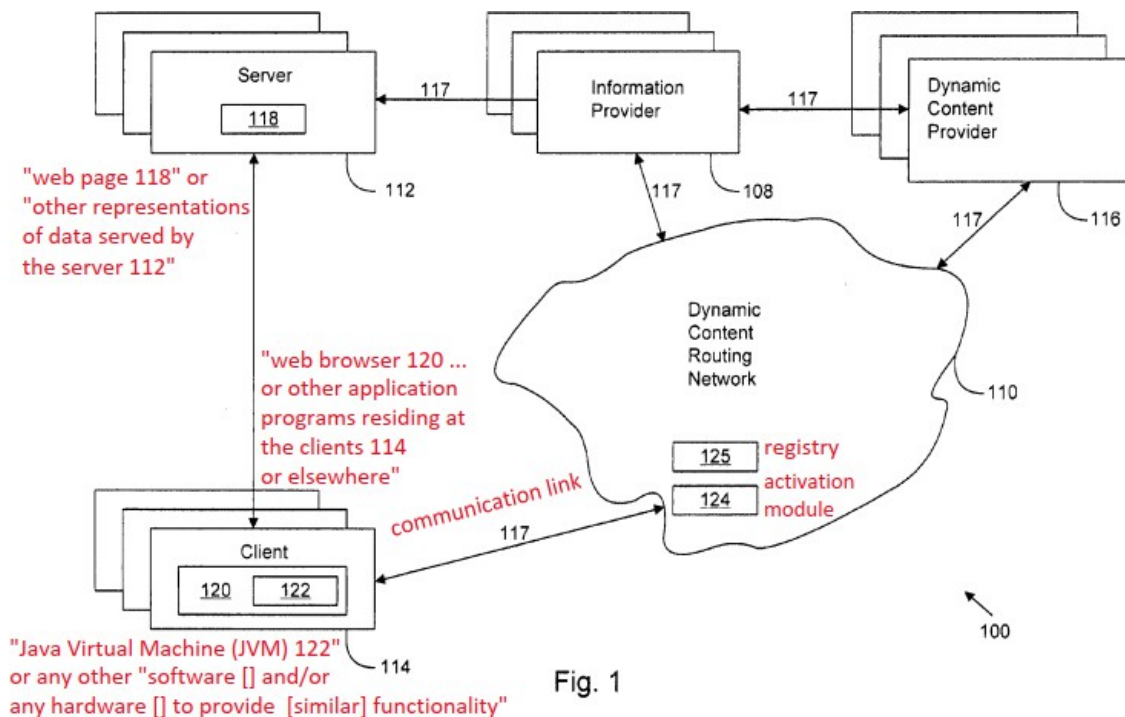
Ground	Challenged Claims	Basis
1	1-37	§ 103: Lawson, Tsutsumitake, and Choquier
2	3, 5, 9, 11, 12, 15, 21, 27, 33	§ 103: Lawson, Tsutsumitake, Choquier, and Herz
3	1-37	§ 103: Choquier, Bird, and Tsutsumitake
4	3, 5, 9, 11, 12, 15, 21, 27, 33	§ 103: Choquier, Bird, Tsutsumitake, and Herz

IV. THE '722 PATENT

A. Overview

The '722 Patent “pertains...to transferring information for remotely updating content at client devices through...digital networks.” Ex.1001, 1:24-27.

Fig. 1 (annotated below) illustrates an input source (e.g., dynamic content provider 116, information provider 108, server 112 alone or combined), client device 114 (e.g., PC, phone, PDA, web-browsing appliance, other devices), routing network 110, and registry 125 identifying clients who have registered for live objects, and a “web page” (which can be any text, data, and/or programs) to clients 114 over communication links 117 (e.g., network, Internet). *Id.* at 4:11-12, 4:55-5:9, 6:19-30, 6:53-60, 10:14-23.



The “routing network...routes messages [from the input source (e.g., information/content provider or its server)] containing data for updating properties of live objects to clients displaying web pages or other representations of data.” *Id.* at 3:9- 12, 6:22-29. The “‘live objects,’ which are designated to be real-time dynamically-updateable objects[, are] identified by an object identifier, or object ID.” 5:28-31. The “objectID...is preferably a string” (simple textual and/or numeric identifier”). *Id.* at 8:34-37, 8:62-65.

The “clients 114 contact the routing network 110 and register for update messages for the objectIDs” and “[t]he routing network 110, in turn, preferably maintains a registry indicating which clients have registered.” Ex.1001, 5:32-36, 6:58-59. Input sources send update messages “any time the[y] want[] to update a property of a live object,” “the routing network 110 routes the update message to the appropriate clients” and “the clients 114 update the specified property of the live object.” *Id.* at 5:41-48, 9:59-62. The updateable live object can include “stock prices and/or sports scores” or any other data at the client which “may be a simple numerical or textual value, e.g., ‘4’”). *Id.* at 7:23-24, 8:57-63, 9:53-56. Fig. 4 shows example live objects:

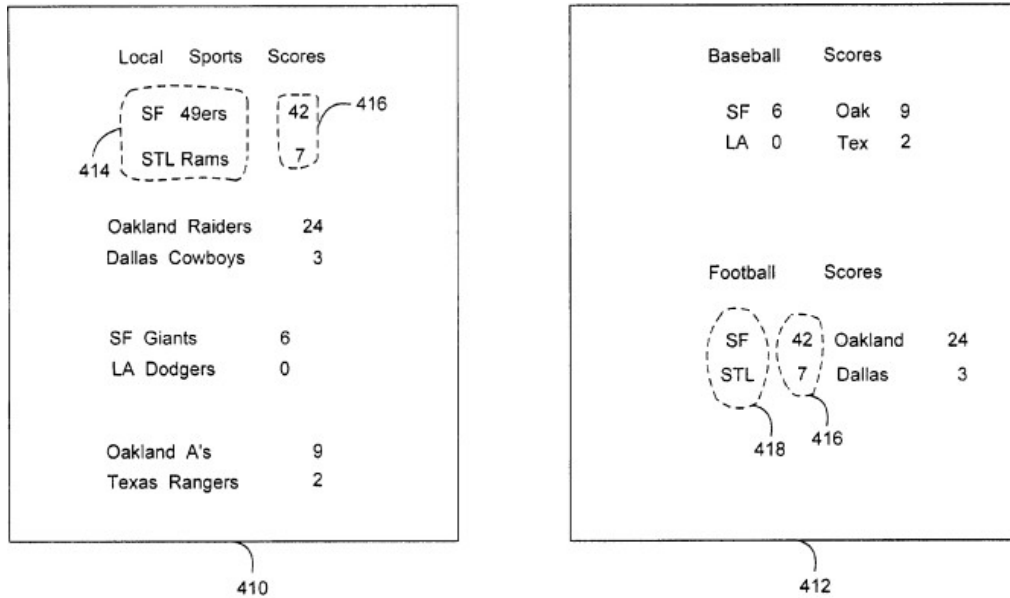


Fig. 4

When an update message is sent to the routing network by the input source “[t]he routing network 110 routes the update message to the appropriate clients 114, and the clients update the appropriate score.” *Id.* at 13:18-25.

“The routing network is adapted to selectively send messages to the nodes in the network” via a gateway. Ex.1001, 3:21-27.

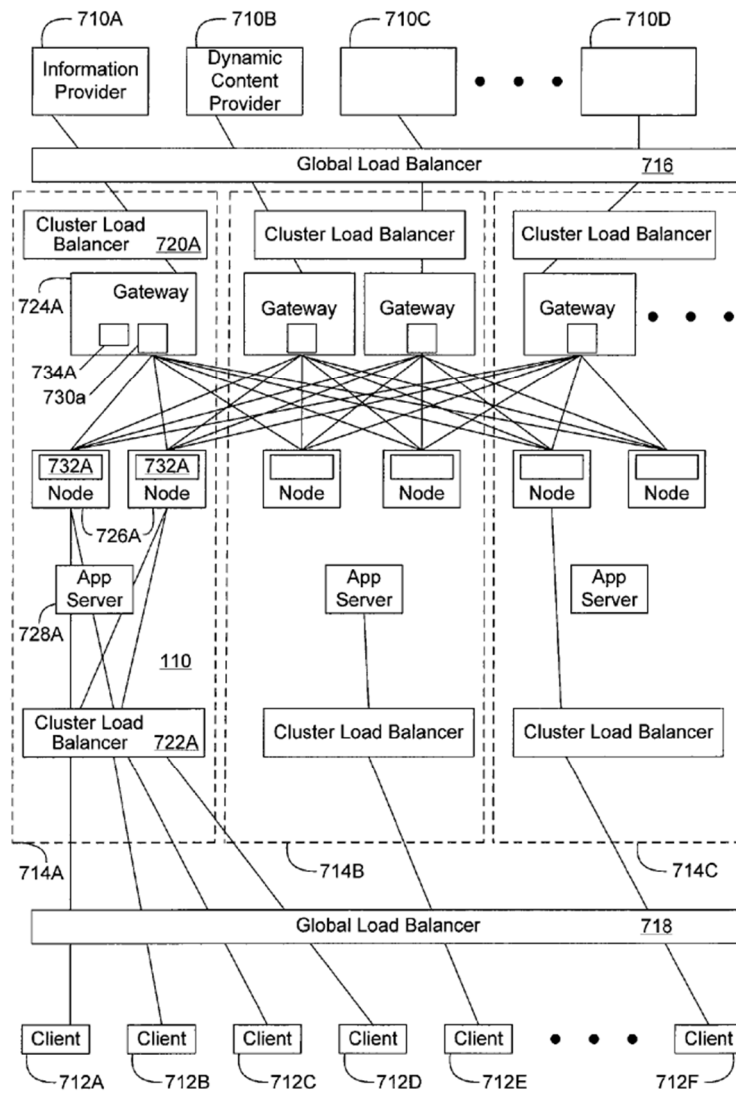


Fig. 7

Updates “from an input source are assigned to one or more of N categories” and “the nodes are assigned to one or more of M types,” mappings are created between message categories and node types,” and “[e]ach gateway keeps track of these mappings” (e.g., category 1 data sent to type 1 nodes). *Id.* at 3:39-46; *see also* 18:24-51, Fig. 9. Ex.1003, ¶¶24-28.

B. Level of Ordinary Skill

A person of ordinary skill in the art as of the '722 Patent's earliest priority claim would have a bachelor's degree in an academic area emphasizing computer networking (i.e. computer science, computer information systems, computer engineering, etc.) and (2) at least two years of work experience with Internet routing with additional education substituting for less experience and vice versa. Ex.1003, ¶¶22-23.

V. CLAIM CONSTRUCTION

Claim terms are construed according to the *Phillips* standard in an IPR. *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005); 37 C.F.R. §42.100. However, claim construction is only required where necessary to resolve a dispute. *See, e.g., Toyota Motor Corp. v. Cellport Systems, Inc.*, IPR2015-00633, Paper 11 at 16 (Aug. 14, 2015) (citing *Wellman, Inc. v. Eastman Chemical Co.*, 642 F.3d 1355, 1361 (Fed. Cir. 2011)). Petitioners propose the following claim constructions.

A. Claims 17-19, 23-25, 29-31, 35-37: “activation module”

Claims 17-19, 23-25, 29-31, 35-37 recite an “activation module” for performing certain functions. “Module” is a nonce word that substitutes for “means”

for purposes of § 112, ¶6. *See Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1350 (Fed. Cir. 2015) (en banc).¹

Claims	Function	Structure in '722 Specification
17, 23, 29, 35	“executing instructions by the client for registering the live object with the routing network”	“activation module 124[: (1)] parses 610 the web page 118 received from the server 112 and identifies the objectIDs of the live objects,” (2) “opens 612 a connection between the client 114 and the routing network 110,” and (3) “uses the connection ... to register 614 the object IDs by communicating to the routing network 116 [] the identified object IDs [(e.g., as a vector, list, or array)]” and equivalents thereof. Ex.1001, 14:33-15:1, Fig. 6; <i>see also</i> 11:1-8.
18, 24, 30, 36	“determining a node type that handles registration and causing the client device to register with the node corresponding to the node type”	activation module 124: (1) determines a node type in the routing network that handles registration, (2) makes a connection to that node, and (3) “registers the appropriate object IDs with the appropriate node[]” and equivalents thereof. Ex.1001, 21:43-52.

¹ In the related litigation, Patent Owner asserts claims 17 and 18 and reads the “activation module” very broadly against open-source software known as Kafka. *See* Ex.1024.

19, 25, 31, 37	“determining a message category of the data representation and causing the client device to register with a node having a node type corresponding to the message category”	activation module 124: (1) “determines which categories of messages it needs and to which nodes types it should connect,” (2) “makes a connection to one node of each relevant type,” and (3) “registers the appropriate object IDs with the appropriate nodes” and equivalents thereof. Ex.1001, 21:43-52.
----------------	--	---

Ex.1003, ¶¶35-36.

No express construction of the other challenged claims is necessary in this IPR because the prior art renders these claims obvious under any reasonable construction. Petitioners reserve the right to propose constructions and respond to constructions proposed by others.

VI. OVERVIEW OF THE ART

A POSITA would understand a node to be any computing device connected to a network, such as “a standalone physical computer unit with a network connection that typically runs its own copy of the operating system” like “a workstation in your office or your laptop computer” (Ex.1025, 1-2), “computers, switches, routers, servers and other devices” (Ex.1026, 4), and “[a] computer that is attached to a network” (Ex.1027, 4), which are consistent with the Microsoft 2002 Computer Dictionary definition: “In networking, a device, such as a client computer,

a server, or a shared printer, that is connected to the network and is capable of communicating with other network devices” (Ex.1028, 3). Ex.1003, ¶¶57-63.

A POSITA would understand routing to be the moving of a packet/message from one node to another in a network or between networks, e.g., “[t]he process of forwarding packets between networks from source to destination” (Ex.1028, 3), “the act of moving information across an internetwork from a source to a destination” (Ex.1029, 1), “the transmission of packets from a source to a destination address” (Ex.1030, 7), and/or “the process of selecting a path across one or more networks.” (Ex.1031, 1). Ex.1003, ¶¶64-65.

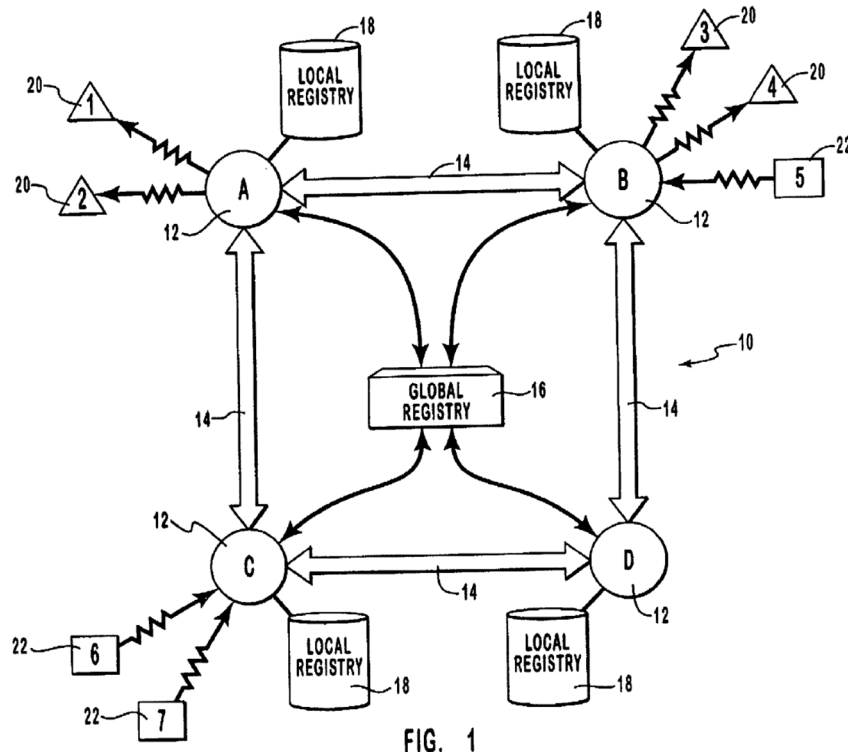
Proxies were also well-known. A proxy is “an item that represents something else” and a proxy server is “a computer that saves information acquired from elsewhere on the INTERNET and makes it available to other computers in its immediate area.” Ex.1032, 3. A proxy or proxy server “acts as an intermediary between a client requesting data and the server providing that data.” Ex.1033, 1. Ex.1003, ¶¶66-67.

A. Lawson (Ex.1005)

Lawson, not considered during prosecution, discloses “[a] system [and] method for global event notification in a distributed computer environment...a local event registry to identify local event consumers that should be notified when an event occurs. A “global event registry [] identifies other servers which need notification

when an event occurs,” “[t]hese other servers will then, in turn, notify their local event consumers of the event, and “filtering to allow event consumers to remove notification of events having little or no interest.” Ex.1005, Abstract.

The “filtering can be performed on any of the information available from a particular event ... include[ing] not only the information available in the event data itself, but also in the event packet, such as the [event type, event producer, server, etc. field in the] event packet.” Ex.1005, 12:3-11, 20:10-30; see also 5:26-65, 16:63-17:6, 17:67-18:3, 22:9-11, Figs. 5, 6.



See also Lawson’s Event Packet:

Event Producer	Server	Type of Event	Time of Event	Size of Event Data	Event Data
----------------	--------	---------------	---------------	--------------------	------------

Ex.1003, ¶¶37-41.

B. Tsutsumitake (Ex.1006)

Tsutsumitake, also not considered during prosecution, discloses “[a] real-time information transmission system can immediately transmit information updated on a server to a client” in a web-enabled environment. Ex.1006, Abstract, 1:6-32.

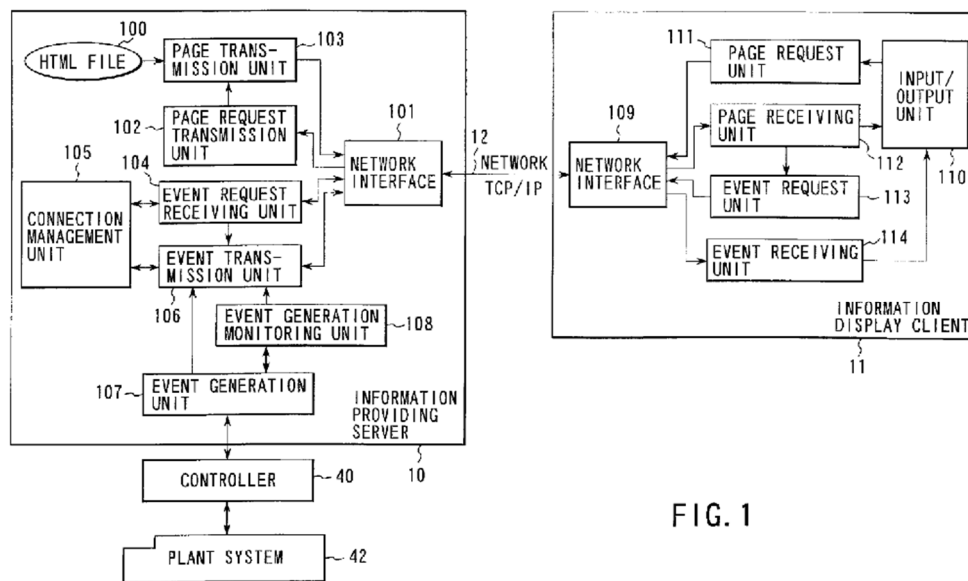
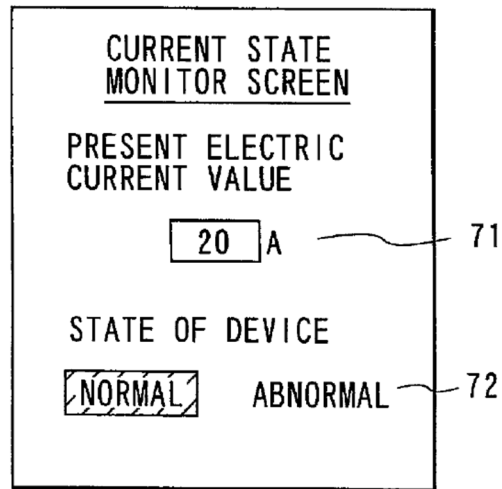


FIG. 1

A web “client analyzes [a web]page” and “[i]f an event request is included in the analyzed page ... [a] connection ID [is used by] the server [to] send[] an event to the client each time the event has occurred.” The “client processes the event and reflects the processed result on a screen of a display” to, e.g., update in real-time data being monitored. Ex.1006, Abstract, 13:31-65. Ex.1003, ¶¶42-44.

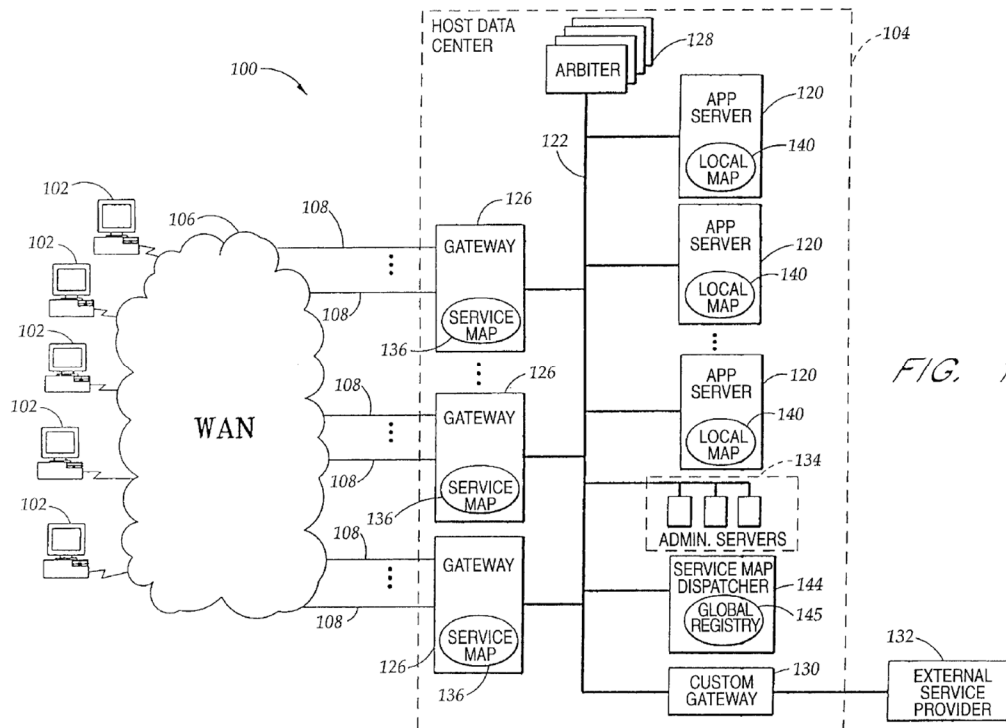
FIG. 7



C. Choquier (Ex.1007)

Choquier, also not considered during prosecution, discloses “a client-server architecture for an on-line services network” using “groups of application servers.”

Ex.1007, 1:45-48. See FIG. 1:



Application servers are “arranged into service groups, with each service group corresponding to a particular service,” like bulletin board system (BBS), stock quote, chat, mail, news, and interactive games services. Ex.1007, 2:1-4, 6:8-22, 23:26-33. Service groups can include one server or multiple servers, and a server can be assigned to multiple groups. Ex.1007, 2:14-18, 6:24-31, 9:40-49, 11:44-47. See. Fig. 3:

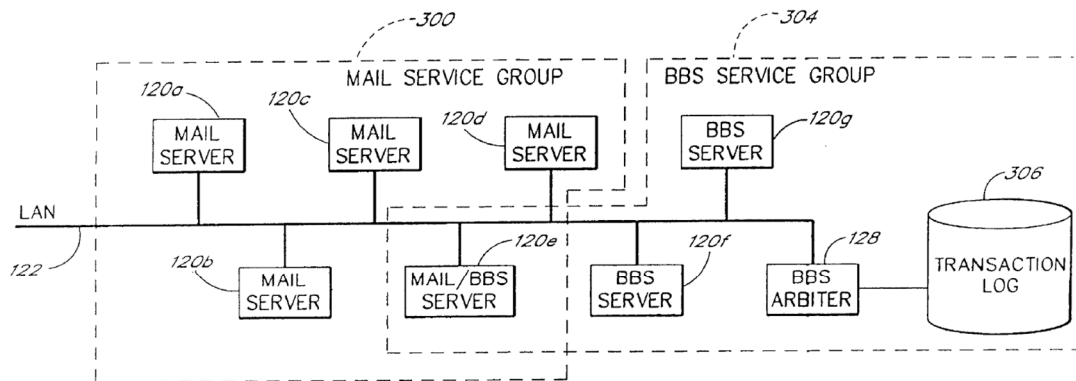


FIG. 3

Service groups can also be assigned based on topics with “each...service group handling a specific topic or set of topics.” Ex.1007, 10:17-20.

Gateways 126 facilitate communication between clients and application servers. One or more custom gateways 130 “links the host data center 104 to one or more external service providers 132.” Ex.1007, 6:37-47.

Arbiters 128 handle “update transactions” for services that update over time. Ex.1007, 9:50-65, 10:4-12. A service map dispatcher maintains a “global registry”

including, for each service group, “IDs of the servers 120 of the service group,” and the dispatcher creates service maps from local maps stored at each server and broadcasts service maps to gateways and Arbiters. Ex.1007, 10:62-11:11, 11:66-12:18. See FIG. 4.

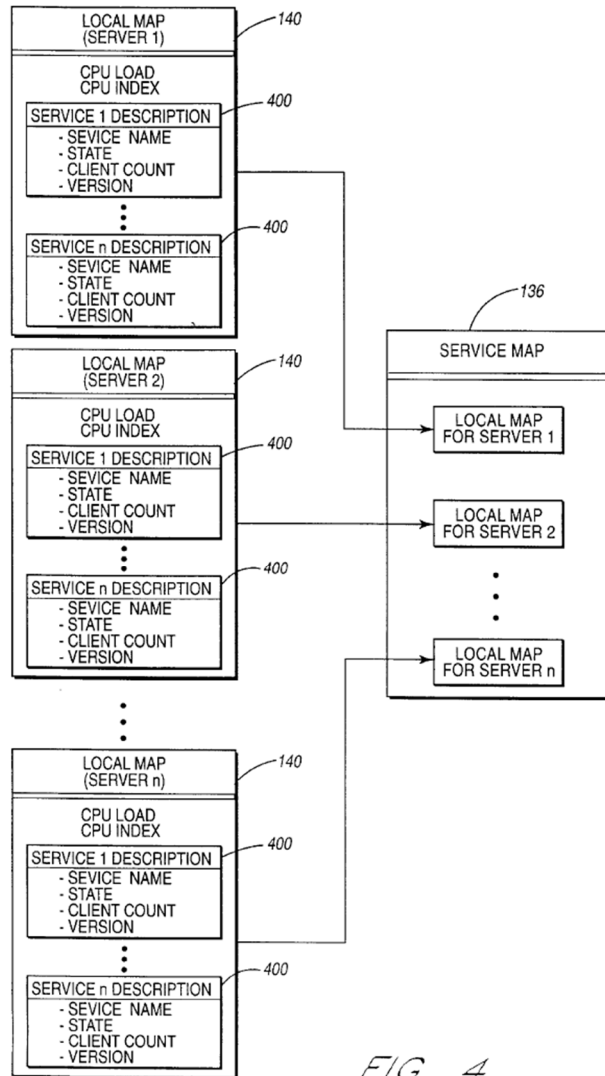


FIG. 4

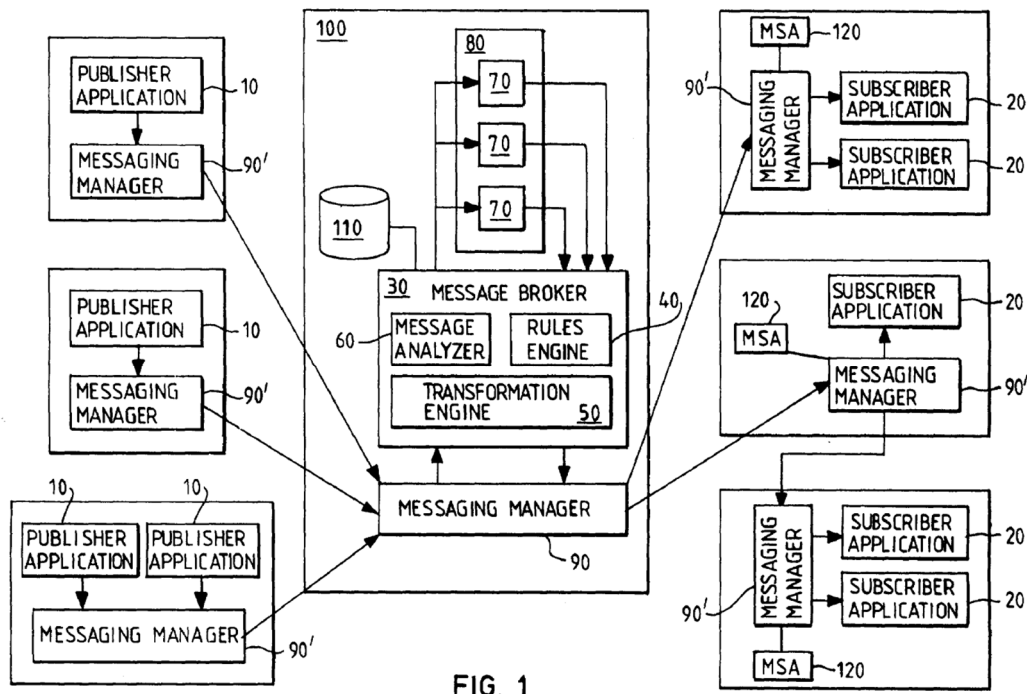
Arbiters use service maps to “determine the status of servers 120 that are currently assigned to respective service groups.” Ex.1007, 10:67-11:3. The service

maps include IDs for application servers and the service types each server supports, and server load information for load balancing. Ex.1007, 2:58-3:3, 10:32-40.

Update transactions can “originate from a variety of sources,” such as “an external data provider connected to the host data center 104 by the Internet.” Ex.1007, 9:57-65. Arbiters record aspects of received update transaction in a log, such as the originator of the update, a transaction ID, a data set affected by the update, etc., and the Arbiter dispatches the updates to the server group so that updated data can be provided to clients. Ex.1007, 9:50-65, 21:33-53, 22:34-45. “Two or more service groups” can be assigned to an Arbiter. Ex.1007, 23:9-19. Ex.1003, ¶¶45-53.

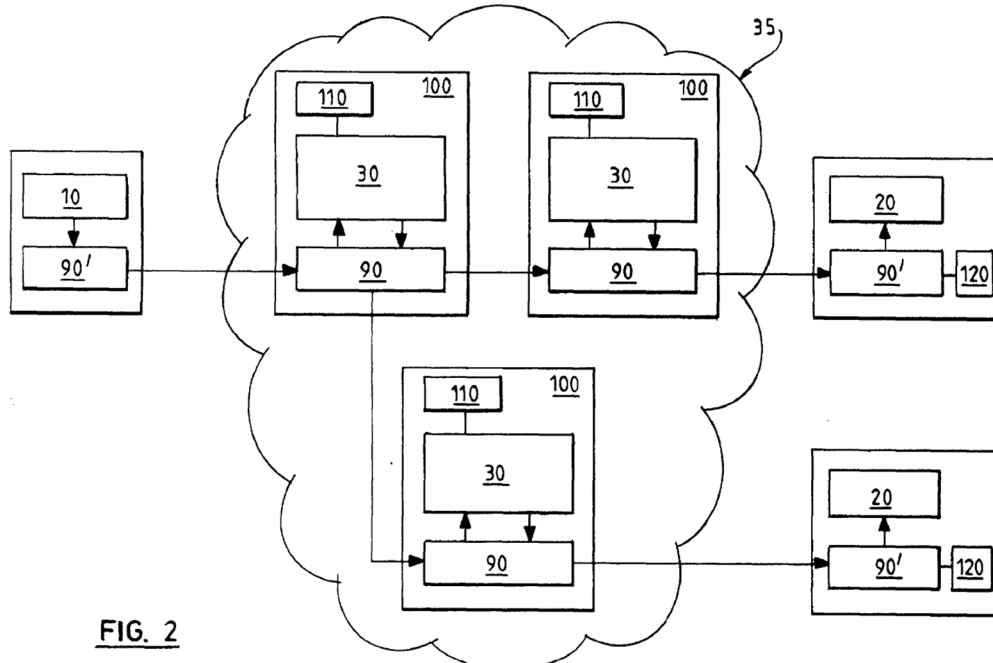
D. Bird (Ex.1008)

Bird, also not considered during prosecution, discloses “a publish/subscribe message distribution service.” Ex.1008, Abstract. See FIG. 1:



Subscriber clients subscribe to messages from specific publishers (“company name,” “specific publisher organisation”) and, when messages are published, a broker at a server 100 uses a message analyzer and rules engine to “investigate...the message header for...identification of the sender” to determine the publisher identity and route the messages to subscribers. Ex.1008, [0003], [0044]-[0045], [0060].

The system could also be distributed across multiple servers 100. Ex.1008, [0037]. See FIG. 2:



Ex.1003, ¶¶54-55.

E. Herz (Ex.1009)

Herz, cited in an IDS during prosecution but not otherwise used (Ex.1002, 609), discloses a proxy server between clients and servers, bidirectionally transfers communications between the client and servers, and stores user-specific information in a secure and private manner, such as a user's "target profile interest summary" that "describes the user's interest level in various types of target objects." Ex.1009, 5:5-13, 32:7-49.

FIG. 2 shows proxy server S2:

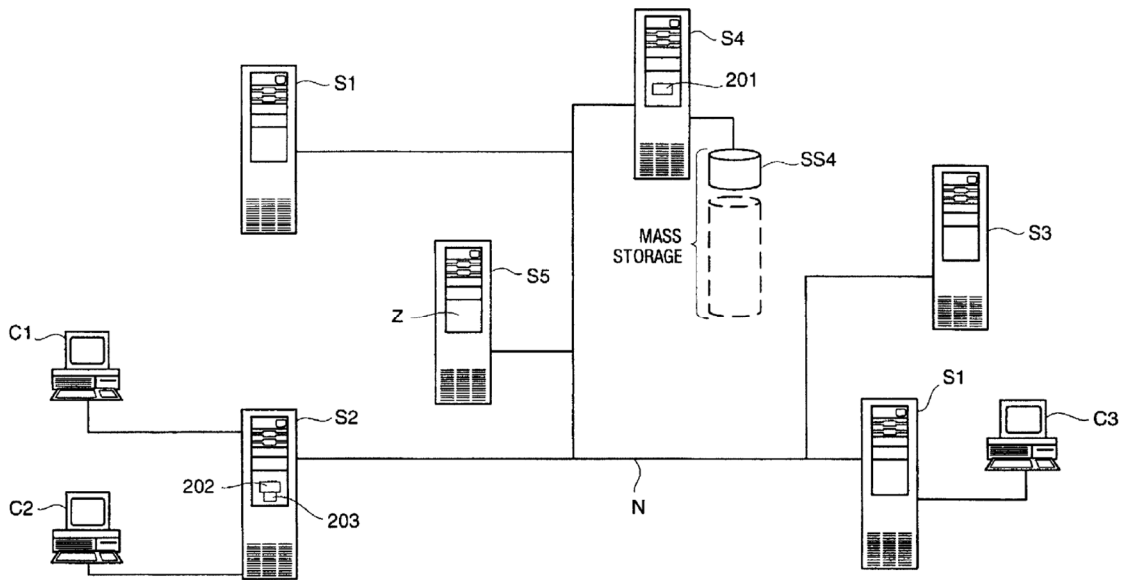


FIG. 2

Ex.1003, ¶¶56-58.

VII. GROUND 1: CLAIMS 1-37 ARE OBVIOUS OVER LAWSON, TSUTSUMITAKE, AND CHOQUIER

A. Motivations to Combine

The Federal Circuit adopted a flexible approach when evaluating whether a patent claiming a combination of elements of prior art is obvious. *KSR International Co. v. Teleflex Inc.*, 550 U.S. 398 (2007). Under *KSR*, a POSITA would have been motivated to combine Lawson with Tsutsumitake and Choquier for numerous reasons. As explained, *supra* §VI., Lawson, Tsutsumitake, and Choquier disclose distributed computing systems that provide data updates from servers to clients (event notifications in Lawson, real-time event updates in Tsutsumitake, and update transactions in Choquier).

Modifying Lawson with Tsutsumitake would ensure events are transmitted in real-time to increase efficiency and reliability. Ex.1006, 2:46-58, 3:26-33. Modifying Lawson with Choquier would expand Lawson’s individual identified servers to include multiple servers in server/service groups based on server type, including assigning server groups to categories of messages (*see* Ex.1007,10:17-20) categorized according to specific event producers (*see* Ex.1005, 12:3-11, 20:10-30). Doing so would provide Lawson’s system with “a high degree of scalability,” “dynamic[] balancing of processing load among...servers,” and redundancy regarding server failure or overloading. Ex.1007, 1:52-62, 9:66-10:8, 25:5-35. Ex.1003, ¶¶70-74.

There is much overlap and many similarities between Lawson, Tsutsumitake, and Choquier that would have motivated combination. Lawson explicitly suggests combination with other systems stating it “can be used with virtually any underlying event notification system” and “is designed to work in conjunction with current event notification systems.” Ex.1005, 4:27-32. Choquier explicitly specifies its “ability to be scaled in capacity without having to change the existing system and application software” and its Arbiter is “generic (i.e., service-independent)” and “new service applications can be written to make use of the services provided by the Arbiter.” Ex.1007, 25:5-21. Lawson and Choquier also both disclose use of a “global registry” to track server mappings. Ex.1005, 4:45-48; Ex.1007, 10:62-11:11, 11:66-12:18.

Choquier is also related to Tsutsumitake, as both deal with providing periodic data updates via servers. Ex.1003, ¶¶74-77.

Thus, modifying Lawson with Tsutsumitake and Choquier would have been straightforward, have had numerous benefits, and would not have required extensive experimentation or imposed undue burden. A POSITA would have a high expectation of success in the combination, because Lawson can be used with any event data, including transmitting events in real time over the Internet as disclosed by Tsutsumitake, and because server groups or clusters disclosed in Choquier were well-known and preferable for efficiency, scalability, and redundancy. For the same reasons, the combination would have yielded predictable results. Ex.1003, ¶78.

B. Lawson, Tsutsumitake, and Choquier meet all claim elements.

1. Independent Claims 1, 7, 14, 20, 26, and 32

Although claims 1 and 7 do not recite the first two elements of claims 14, 20, 26, and 32, the claims otherwise recite substantially overlapping elements and are discussed together. *See* Ex.1021.

[1.0]/[7.0]/[14.0]/[20.0]/[26.0]/[32.0]: “[1.0:A method][7.0:A routing network][14.0: A method][20.0: An apparatus][26.0: An article of manufacture including a computer-readable storage medium having instructions stored thereon, execution of which by a computing device causes the computing device to perform operations][32.0: A non-transitory computer readable storage

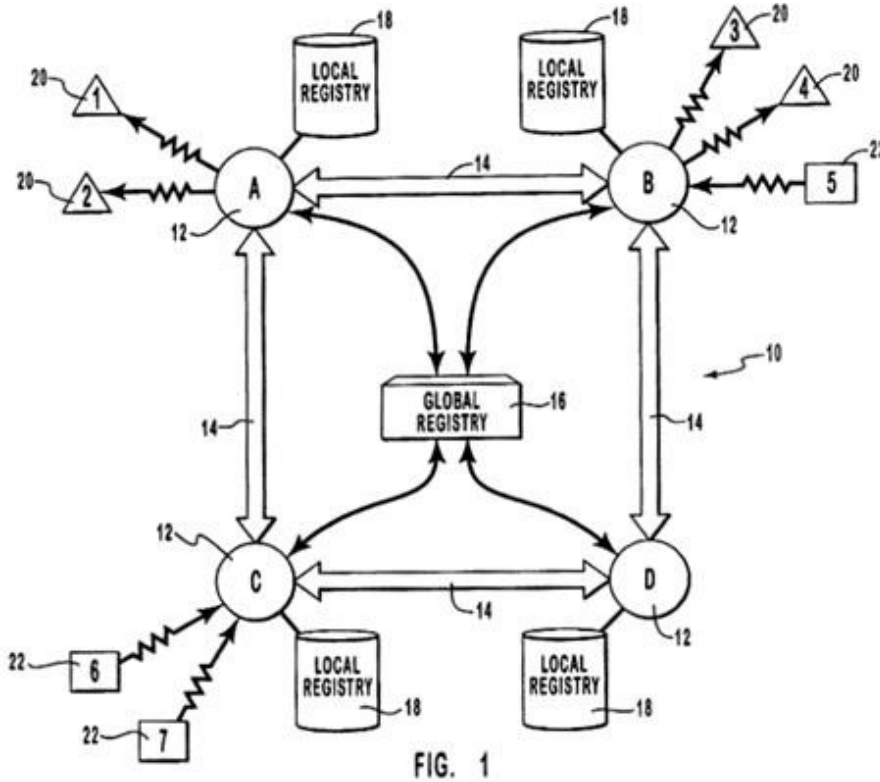
medium having instructions stored thereon, the instructions]
comprising:”

To the extent the preambles are limiting, Lawson discloses an apparatus, “a system,” and “method” “for globalizing event notifications in a distributed computing environment.” Ex.1005, Abstract, 4:25-27. As further described below, Lawson also discloses a routing network (“distributed computing environment”) and an article of manufacture and computer readable storage mediums (computers including CPUs and “program storage means”). *See* Ex.1005, 1:14-17, 5:11-12, 7:22-28. Ex.1003, ¶¶88-95.

[14.1]/[20.1]/[26.1]/[32.1]: “[14.1/26.1:providing, using a processing device of an input source,][20.1:an input source device configured to provide][32.1:instructions for providing] a data representation to a client device, **[14.1/20.1/26.1:different from the input source,]** coupled to a routing network, wherein the data representation includes at least one live object **[20.1/26.1:that is]** recognizable by the client device, and”

Lawson discloses input sources (event producers) and client devices (event consumers). FIG. 1 shows event consumers as “triangular regions 20,” numbered 1-4, and event producers as “rectangular regions 22,” numbered 5-7, where event producers “refer to a process, user, device, or other item that produces or generates

an event” and event consumers are “refer to any process, user, device, or other item that desires notification of an event. Ex.1005, 9:1-15; *see also* 1:51-60.



Event producers and consumers are different from each other and can be or include general or special purpose computers and software. Ex.1005, 7:17-39, Figs. 1-3; *see also* claims 38-39, 43-44. Ex.1003, ¶¶99-101.

Lawson discloses a routing network in which the event producers 22 provide events to servers connected to client devices that register for the events via servers (servers A, B, C, D). Ex.1005, 4:54-65, 7:52-8:4. Lawson discloses an “event packet” and “[f]or most events, packaging event data into an event packet has the

benefit of adding data that may be useful in processing the event.” Ex.1005,12:3-10.

See Lawson’s event packet:

Event Producer	Server	Type of Event	Time of Event	Size of Event Data	Event Data
----------------	--------	------------------	------------------	-----------------------	------------

Lawson explains “the event producer field is an identifier that identifies the process, device, user, etc., which generated the event,” “[t]he server field identifies the server location where the event was produced,” “[t]he type of event field is an identifier describing the type of event which occurred,” “[t]his [event type] identifier preferably identifies an event in the global event registry and/or local event registry so that appropriate servers or event consumers can be identified to receive the event,” and “[t]he event data is the data associated with the event.” Ex.1005, 12:12-26. The “event data” in Lawson can be any data or records, such as employee’s name, phone number, address, payroll data, program function call, computer program code, etc. Ex.1005, 6:4-15, 7:23-51, 21:15-26, 24:1-4. Ex.1003, ¶¶102-105.

Lawson discloses examples in which “event consumer 4 [in Fig. 1] is a company phone directory” and “event consumer 1 [in Fig. 1] is a payroll database.” 23:64-65, 24:1. These event consumers dynamically update data/records they store based upon receipt and processing of an event packet (e.g., update phone number, update payroll database based on data/records in the event packet). 21:16-25, 24:64-24:6, Fig. 7 #178 (phone directory), 24:1-4, 25:16-21, Fig. 7 #186 (payroll database).

Lawson illustrates these two examples via Fig. 7, which is “an example of how [Lawson’s system and method] works ... [where t]he network topology used for th[e] example is presented in Fig. 1” and the “event globalization process is described in Figs. 3-6.” 23:60-63, 24:17-18. Ex.1003, ¶104.

Tsutsumitake discloses providing data representations including at least one live object recognizable to client devices in the form of HTML webpages with embedded event data. Tsutsumitake discloses “a real-time information transmission system for transmitting information in real time from a server to a client via a network.” Ex.1006, 3:39-41. As shown in Fig. 1, “A [web]page request receiving unit 102 [on server 10] receives a [web]page request sent from the client 11 via the network interface 101, analyzes it, and informs a [web]page transmission unit 103 of a [web]page to be sent to the client 11.” The “page transmission unit 103 transmits an HTML file stored in the server 10 to the network 12 via the network interface 101 [to client 11]” and “[i]nformation to be transmitted is not limited to the HTML file (100) [webpage], and may be freely chosen one such as image or voice.” Ex.1006, 8:39-49; *see also* 3:39-56.

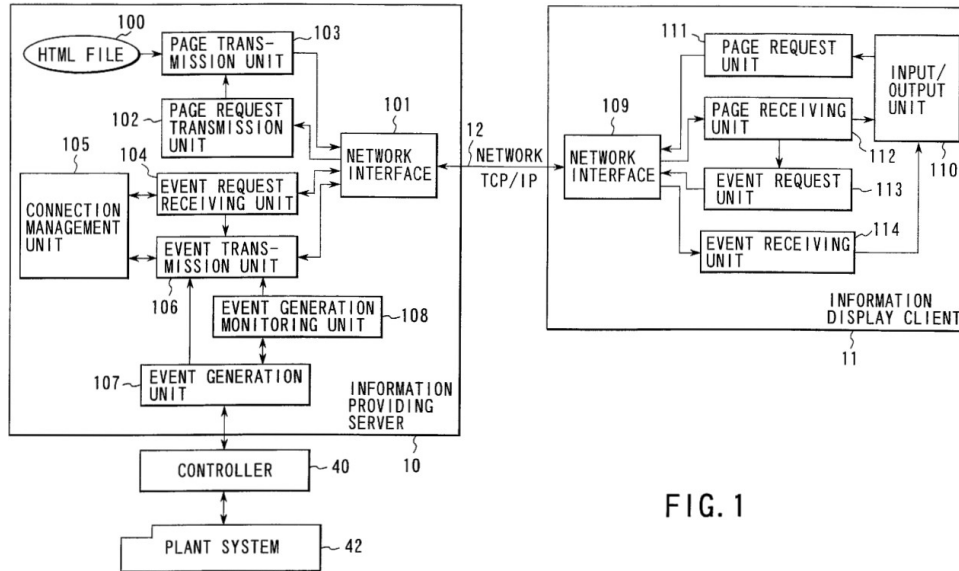


FIG. 1

Ex.1003, ¶107.

Fig. 2 of Tsutsumitake shows an example of the webpage data, including an embedded event, sent by server 10 to client 11.

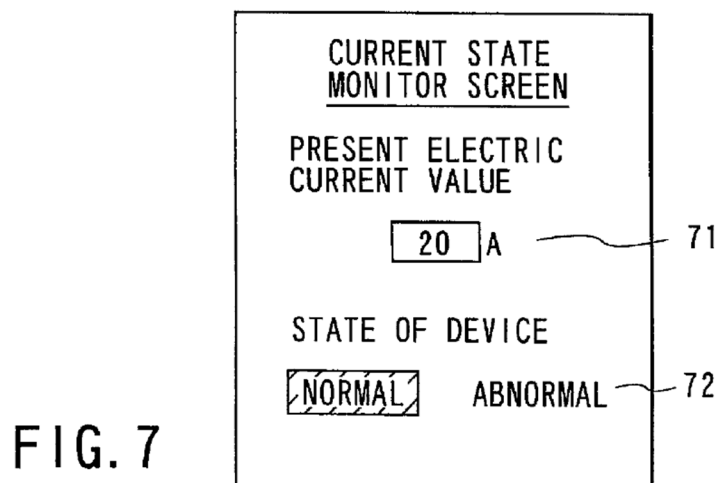
```

< HTML >
< HEAD >
< TITLE > SAMPLE PAGE < /TITLE >
< /HEAD >
< BODY >
< H1 > SAMPLE PAGE < /H1 >
THIS PAGE INCLUDES EVENT REQUEST
< EMBED >
  < EVENT URL=http://abc.def/hij.evt >
< /EMBED >
< /BODY >
< /HTML >
  
```

FIG. 2

An “event request to the server is included in the [web]page. The event request is expressed by an attribute ‘URL’ in a tag ‘EVENT’ in a tag ‘EMBED’.” Ex.1006, 10:20-23. The client 11 recognizes the webpage as containing a “live object” because upon receiving the webpage, “the page receiving unit 112 [in client 11] informs an event request unit 113 of the URL of the event request.” Ex.1006, 9:61-68. *See also* 14:3-29, Fig. 8.

An example live object is shown below in FIG. 7 of Tsutsumitake.



Ex.1003, ¶¶108-110.

Therefore, the combination meets these elements. Ex.1003, ¶¶99,111.

[14.2]/[20.2]/[26.2]/[32.2]: “[14.2:causing the client device to] [20.2/26.2:that causes the client device] [32.2:wherein the client device is configured] [14.2/26.2/32.2:to respond to the live object of the data representation by determining][20.2:to determine] an object identifier (ID) of the live object [14.2:and] to register for updates

of the live object with the routing network, such that registering the client device with the routing network provides client connection information to **[14.2:a node in]** the routing network;”

Lawson discloses client registration for updates including “[w]hen an event consumer receives notification that a certain event happened on an event producer, the event consumer can take appropriate action” and “event consumers which desire to receive notification of certain events ‘register’ with the event producing system.” Ex. 1005, 1:61-63, 2:6-9. Lawson discloses “[w]hen an event consumer needs to register for a new event...the event consumer can send a ‘register for event’ packet...to the event globalization process on the local server.” Created events are stored in the “global registry” and when “an event consumer registers for an event, the event globalization process will place an entry in the local event registry” at the servers “identifying the event and the event consumer wishing to receive notification of the event.” Ex.1005, 15:41-56. For example, “the employee phone book database could register for notification of ‘a modified object’ type event.” Ex.1005, 21:20-21. As another example, “a payroll database [could] also register[] for the add user event so that information can be extracted to update the employee payroll database.” Ex.1005, 24:1-4. Ex.1003, ¶¶117-119.

Lawson’s “event type” corresponds to the ’722 Patent’s “object identifier (ID) of the live object” because the ’722 Patent explains an “objectID [] is preferably a

string. The string can encode any information ... [and can be] a simple textual and/or numeric identifier. Ex.1001, 8:34-37, 8:63-9:2, 14:14-22. Likewise, in Lawson, the event field in the event packet “identifies an event in the global event registry and/or local event registry so that appropriate servers or event consumers can be identified to receive the event,” and “event field is used to extract from the global event registry...those servers that should receive notification of the event and is used to extract from the local event registry...those local event consumers that should receive notification of the event.” Ex.1005, 12:15-34. Lawson’s “registry compris[es] a list of local event consumers and associated list of event types [] identify which local event consumers are registered for events of the type included in [the] event packet.” Ex.1005, 27:61-64 (claim 19). Ex.1005, 12:34-37, 21:15-26, 24:1-4. Ex.1003, ¶¶112-116,119.

Lawson as modified by Tsutsumitake provides for causing a client device to respond to a live object to determine the ID and register for updates. Tsutsumitake discloses “[t]he page receiving unit 112 [on the client 11] analyzes the content of the [web]page sent from the server 10...to determine whether an event request is included in the page.” Ex.1006, 10:56-59. If “the event request is included, the page receiving unit 112 informs the event request unit 113 of the URL of the event and causes the event request unit 113 to send the event request to the server 10.” Ex.1006, 10:64-67, Figs. 1, 3.

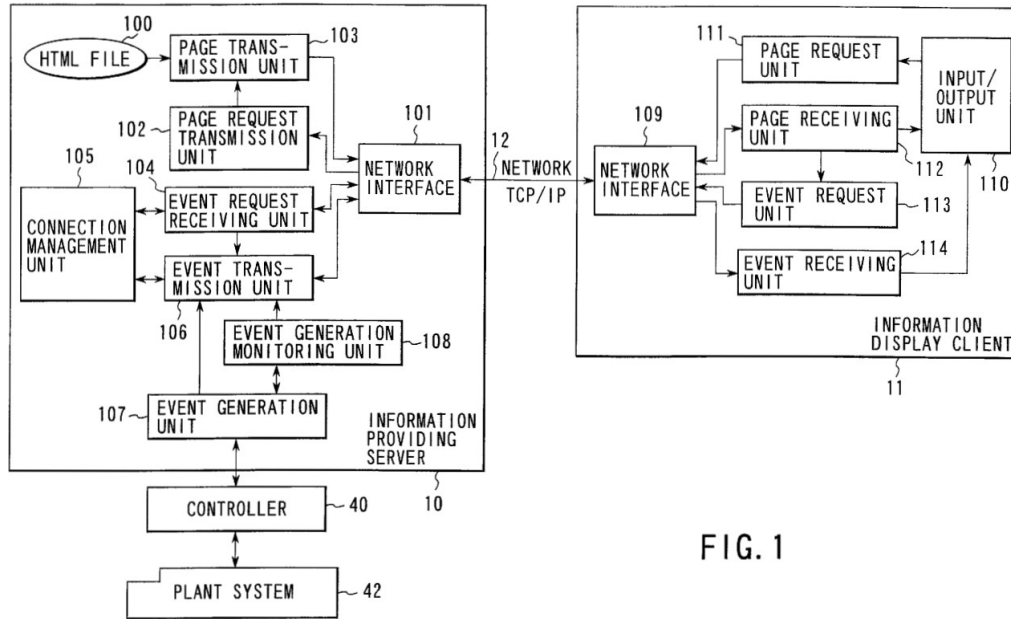
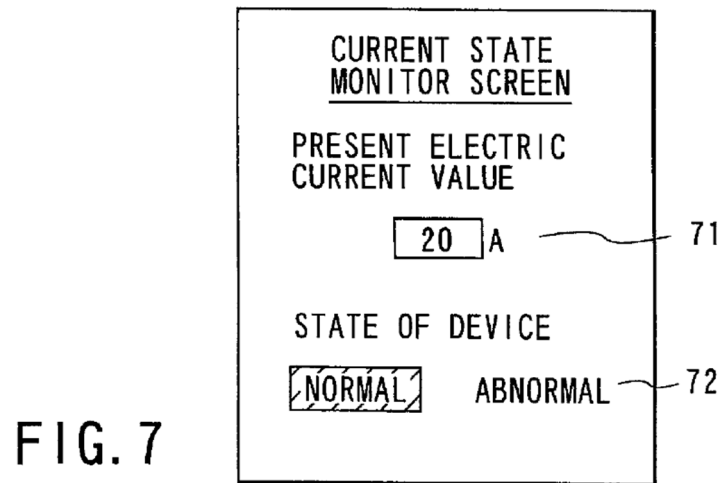


FIG. 1

This causes the client to register for updates because “[i]n this structure, the client [11], which received the requested [web]page information from the server, can automatically issue the event request to the server according to the event request information set in the [web]page information, and can activate the event notification,” and “[s]pecifically, by associating the [web]page information with the event, the [web]page information opened on the client side can be dynamically changed according to the event notification sent from the server.” Ex.1006, 4:22-29; *see also* 1:7-43, 7:63-8:23, 9:43-68, 13:32-14:3, Fig. 7. Ex.1003, ¶¶120-121.

At the server 10, “[i]f they are not registered, a pair of the...event ID...and...client ID...are registered in the connection management unit 105 (step B11).” Ex.1006, 11:45-55, Figs. 1, 4.

An example updating live object is shown in FIG. 7 of Tsutsumitake as an updating current value and a device state. Ex.1006, 13:46-57.



Ex.1003, ¶¶122-124.

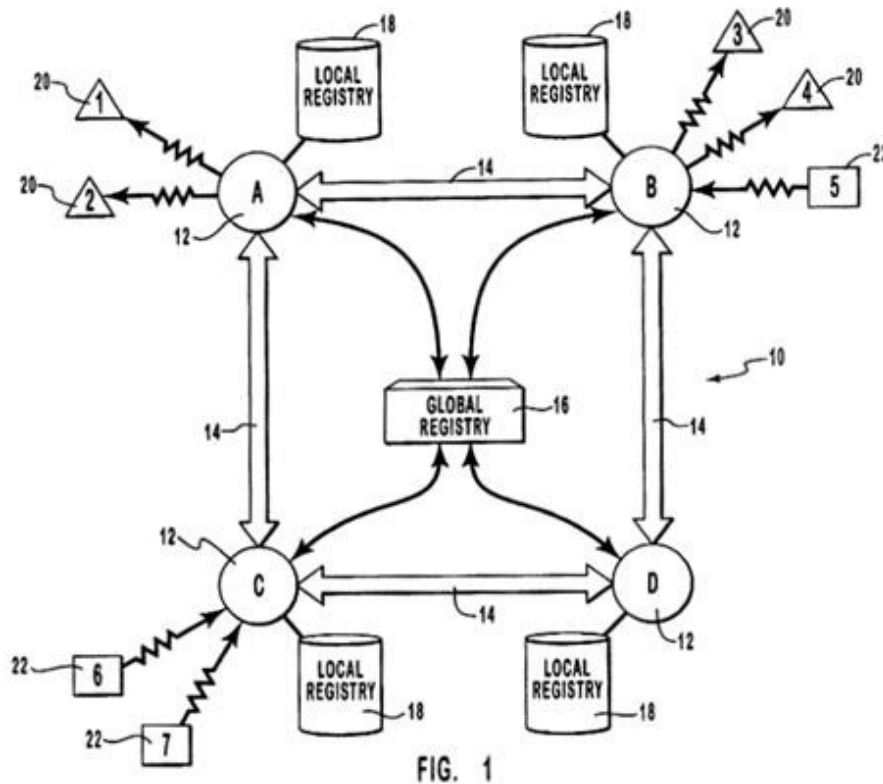
Therefore, the combination meets these elements. Ex.1003, ¶¶112,125.

[1.1]/[7.1]: “[1.1:receiving, using a processing device,][7.1:a gateway device configured to: receive] an update message from an input source, the update message identifying a live object and containing data for updating a property of the live object;”

[14.3]/[20.3]/[26.3]/[32.3]: “[14.3/26.3:sending, using the processing device of the input source,][20.3:wherein the input source device is configured to route][32.3:instructions for providing, using a processing device of an input source, different from the client device,] an update message to the routing network, wherein the update

message identifies the live object and contains update data [14.3:that updates][20.3/26.3/32.3:for updating] a property of the live object;”

Lawson’s routing network and event packet (update message) are shown below:

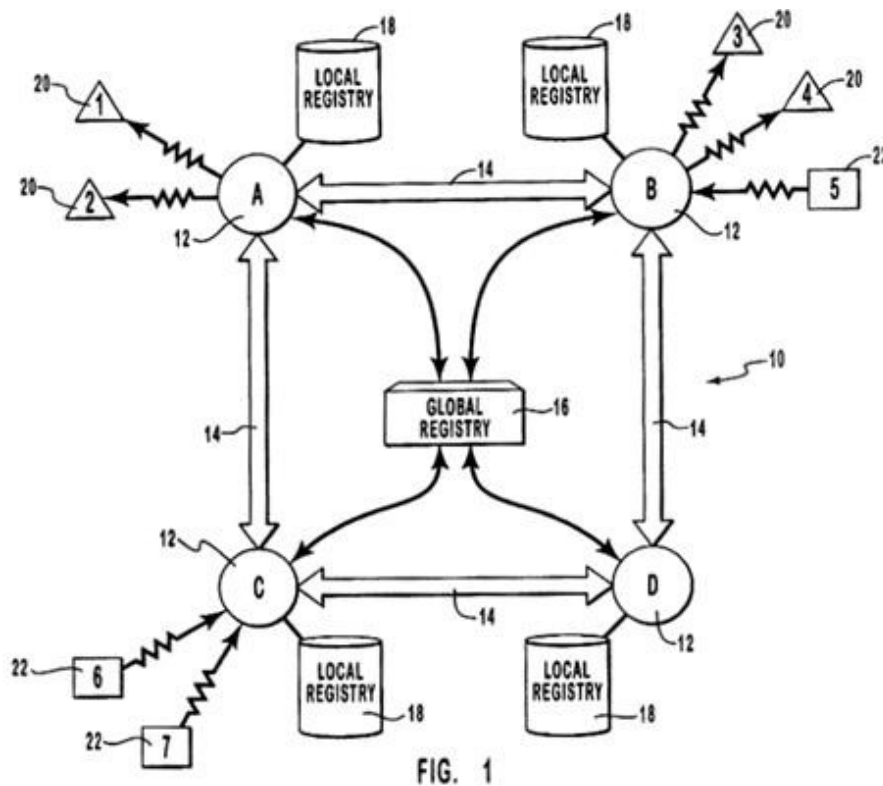


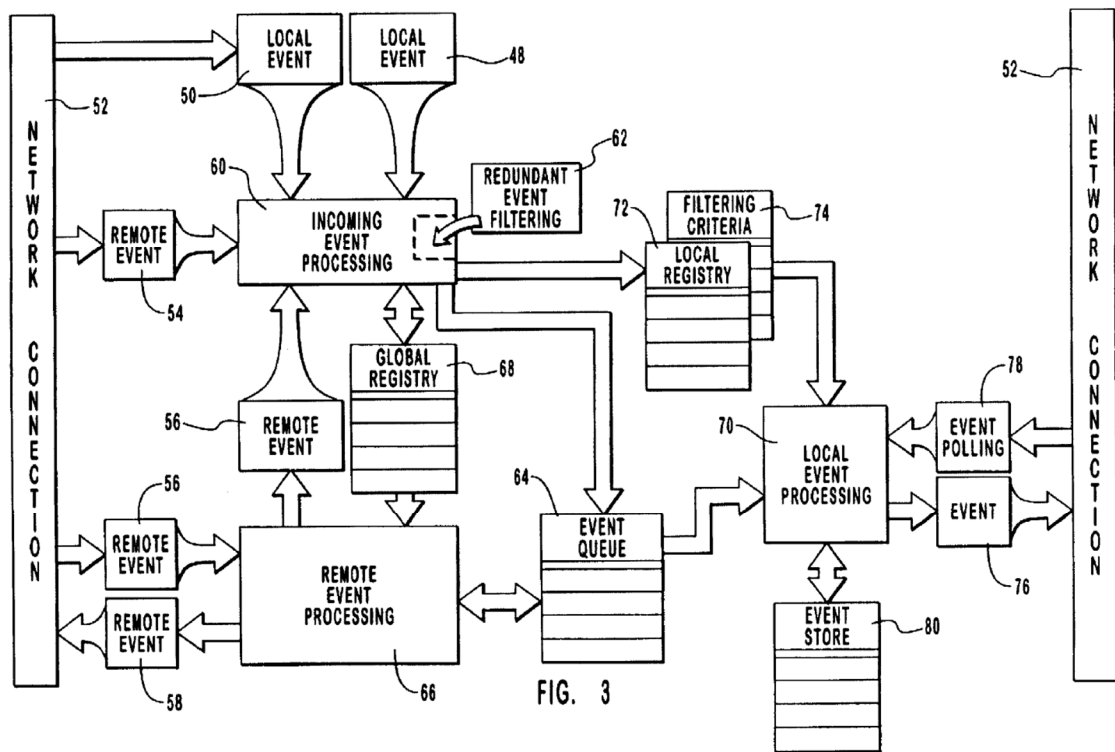
Event Producer	Server	Type of Event	Time of Event	Size of Event Data	Event Data
----------------	--------	---------------	---------------	--------------------	------------

Lawson provides an example in which “event producer 5 [at server B] added a new user to a database stored on the network” and “that event consumer 4 [at server B] is a company phone directory that has registered to receive an ‘add user’ [or ‘modified object’] event so that it can extract the phone number of the added user

and place [or update] it in the company phone book,” and event consumer 1 [at server A] is a payroll database that has also registered for the add user event so that information can be extracted to update the employee payroll database.” Ex.1005, 21:16-26, 23:60-24:4. “After the user has been added to the database, step 156 indicates that an event packet is created [by event producer 5 at server B].” Ex.1005, 24:7-9. Ex.1003, ¶¶126-135.

Lawson’s servers include a “remote event processing block 66” that acts as a gateway. See FIGS. 1 and 3:





The server that receives the event/update from a consumer (e.g., if server C receives an event/update from event producer 6 or 7 shown in FIG. 1), places the incoming event/update into an event queue 64 for dispatch by the remote event processing block 66 to other servers that need the event, thus acting as a gateway between its event consumer (input source) (e.g., event consumer 6 or 7) and the other servers (nodes) (e.g., servers A, B, D) in Lawson's routing network. Ex.1005, 12:54-58.

Lawson's event packets include data identifying and for updating live objects. Among other things, an "event type" identifying the "appropriate servers or event consumers [] to receive the event" and "event data" comprising any information,

data, or records “associated with the event.” Ex.1005, 12:7-10, 12:17-20, 12:25-26. Ex.1003, ¶¶136-138.

Tsutsumitake similarly discloses “[i]f an event has occurred, the server 10 sends it to the client 11 with use of [a] connection ID. Subsequently, as shown in Fig. 6, each time an event has occurred, it is sent [to client 11] repeatedly with use of the [connection] ID.” Ex.1006, 13:1- 5; *see also* 12:10-15, Figs. 1, 5. The example of Fig. 7 discloses “the event indicat[es] the present value of current and...the state of the device being transmitted in real time.” Ex.1006, 13:59-66. Ex.1003, ¶139.

To the extent it is asserted the claimed “gateway” must be a device separate from the claimed “node,” Choquier would have rendered this obvious. Choquier discloses using an Arbiter microcomputer 128 and gateways 126, 130 that are separate devices from its servers 120, although Choquier specifies Arbiter functionality can be implemented in other devices, like the servers. Ex.1007, 23:9-19. Ex.1003, ¶129.

Therefore, the combination meets these elements. Ex.1003, ¶¶126,142.

[1.2]/[7.2]: “identifying a category of the update message based on the input source; **[1.2:determining][7.2:determine]** a node having a node type to which the update message is to be routed based on a mapping of categories of update messages to node types, the mapping controlling

an amount of update message traffic through nodes of a routing network;”

[14.4]/[20.4]/[26.4]/[32.4]: “wherein a gateway device at the **[14.4/26.4/32.4:routing network][20.4:network]** is configured to identify a category of the update message based on the input source, to determine a node type to which the identified category maps”

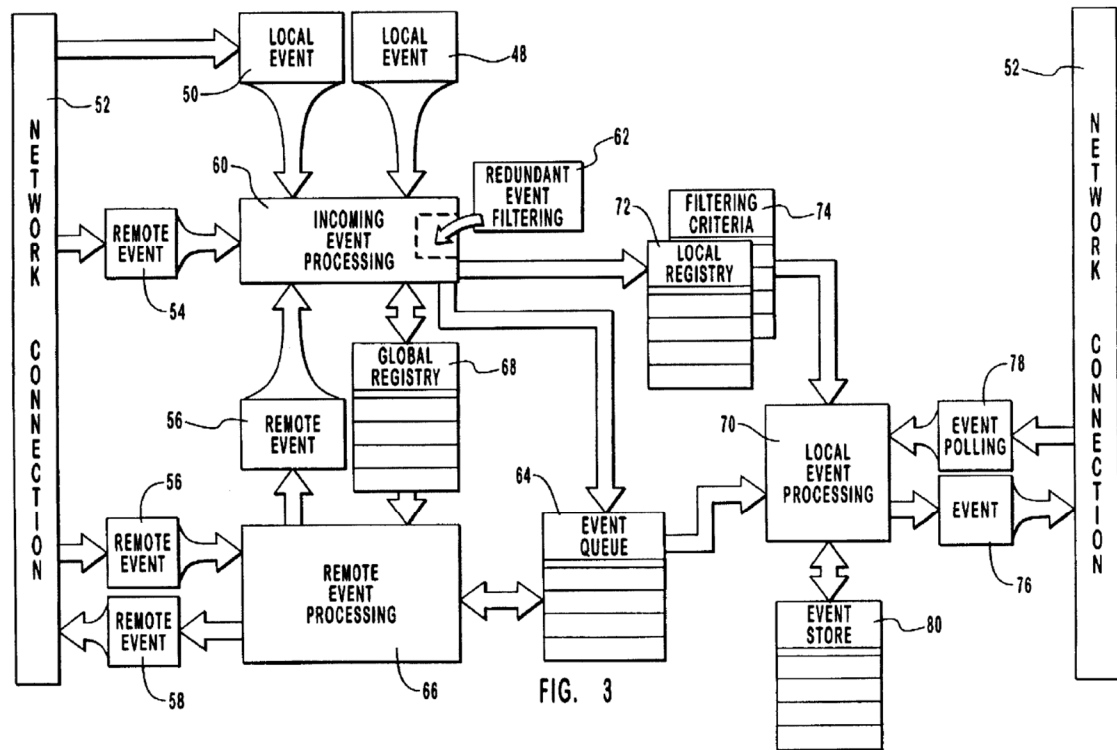
Each independent claim recites identifying (by a gateway device) a category of the update message based on the input source and a mapping of a node type to the message category.

Lawson discloses categorizing messages based on input source. Lawson discloses event/object filtering so event consumers registering for events can “prevent notifications of irrelevant events.” Ex.1005, 5:26-30. Lawson’s filtering can be used so “notification of events affecting a certain class of objects or category of objects is sent,” such as filtering based data “in the event packet,” which includes filtering based on *event producer* (input source):

Event Producer	Server	Type of Event	Time of Event	Size of Event Data	Event Data
----------------	--------	---------------	---------------	--------------------	------------

Ex.1005, 5:45-48, 20:10-13; *see also* 5:26-65, 16:63-17:6, 17:67-18:3, 22:9-11, Fig. 5. Ex, 1003, ¶¶143-145.

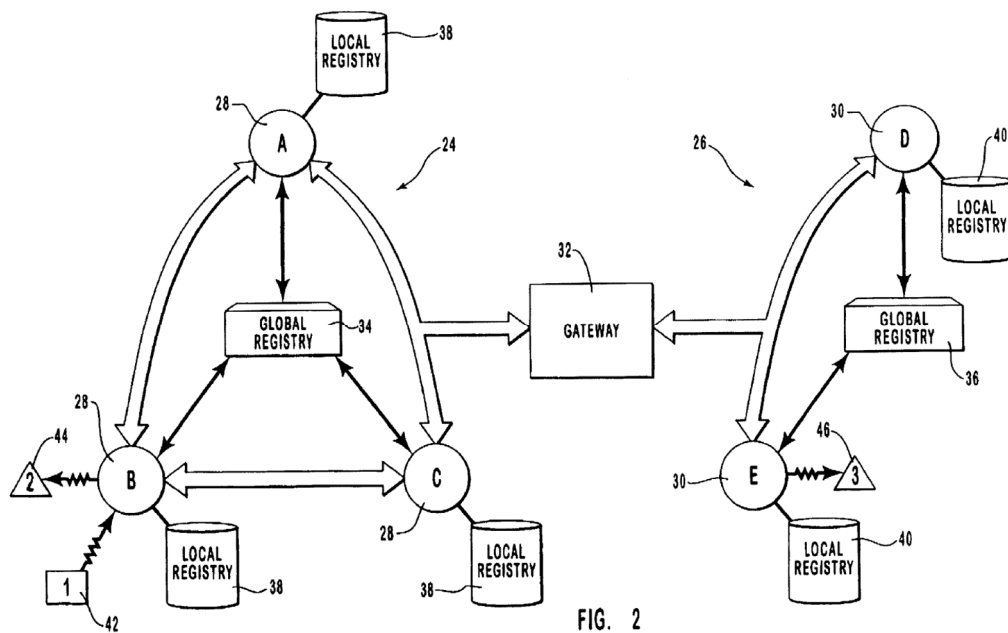
As described above, Lawson’s servers include a “remote event processing block 66” on servers acting as a gateway. See FIG. 3:



The remote event processing block 66 “monitors event queue 64 and when events are placed therein...that need to be transferred to other servers in the network, remote event processing block 66 will establish a connection to those servers and transfer the events to the other server.” Ex.1005, 13:10-17.

In Lawson, to determine which servers (nodes) should receive events (updates), entries are placed in the “global event registry” so “the global event registry is updated to contain an entry identifying the server where the event consumer is located and the desired event.” Ex.1005, 4:54-65, 7:52-8:4. An “event

is only transferred to said another of said plurality of servers if said event meets designated filtering criteria,” (e.g., filtered for specific event producers). Ex.1005, 4:45-48, 30:25-30. Remote event processing block 66 “identifies the servers that should receive events queued in event queue 64 by checking global registry 68.” 13:12-24; *see also* 12:12-20, 12:29-34.



Ex, 1003, ¶¶146-148.

Lawson provides an example, referring to Fig. 2, in which “[w]hen server B recognized that event producer 42 had initiated an event, server B would check global event registry 34,” which “would identify that server E needed notification of the event,” and “[s]erver B would then send the event through gateway 32 to server E.” 10:23-28. In this example, server B acts as the gateway, as server B receives the

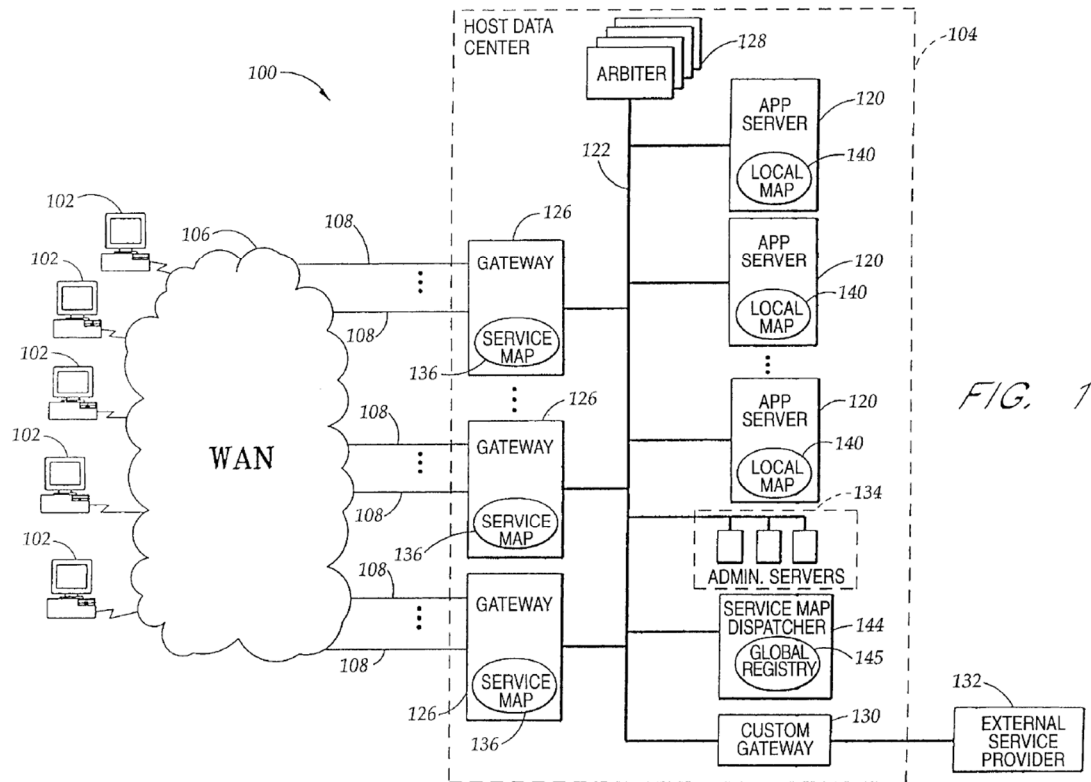
initial event from event producer 42, and server B checks the mapping provided by the global event registry to determine which nodes (servers) need the event (update) based on the message category (filtering).

Lawson further discloses this mapping provided by the global registry is for “controlling...message traffic” as recited in claims 1 and 7 of the ’722 Patent. Lawson’s example above regarding Fig. 2 specifies the “event notification was transferred from server B to server E without broadcasting the event to servers A, C and D,” and “[t]hus, such a method can globalize an event to required locations while, simultaneously, *reducing message traffic* due to broadcast type events.” Ex.1005, 10:43-47 (emphasis added); *see also* 22:62-63.

As described above, messages in Lawson’s routing network can be categorized based on input source using filtering by event producer. Further, each of the nodes in Lawson’s network can be considered different node types (types A, B, C, D, etc.) and the global registry is used to map message categories to node types. Ex, 1003, ¶¶149-154.

To the extent it is asserted Lawson does not explicitly disclose “node types” it would be obvious to include “node types” in Lawson based on Choquier. As described above, Lawson discloses servers in its network can service application-specific events, such as “a payroll database” and “a company phone directory.” Ex.1005, 23:64-24:1.

As described above, *supra* §VI.C., Choquier discloses assigning servers to server groups (node types) based on message categories (“a specific topic or set of topics”) using service maps. Ex.1007, 10:17-20, *see also* 2:1-4, 2:14-18, 6:8-31, 9:40-49, 11:44-47.



When an “update transaction” is received, such as from “an external data provider connected to the host data center 104 by the Internet,” an Arbiter corresponding to a service type associated with the “update transaction” stores in a log information such as a transaction ID and an originator of the update, and then disseminates the update to servers in the server group (servers of a particular server type) according to the service map. Ex.1007, 9:50-65, 21:33-53, 22:34-45. It would

have been obvious based on the combination of Lawson and Choquier for Lawson's servers to be included in server groups to provide for server groups/clusters (e.g., providing servers groups at each of servers A, B, C, D in Lawson) assigned according to "topic" or set of "topics" (message category of "event producer" based on Lawson's filtering) as disclosed in Choquier, and for Lawson's servers to determine which server groups (node types) should receive an event update based on this server groups to event producer mapping. As described above, *supra* §VII.A., a POSITA would have been motivated to include server groups/clustering in Lawson's system, as it was well-known to do so to increase efficiency, redundancy, and scalability of Lawson's routing network. Ex.1007, 25:5-21, 1:52-67. Ex.1003, ¶¶155-159.

Therefore, the combination meets these elements. Ex.1003, ¶¶143,160.

[1.3]/[7.3]: "[1.3:routing, using the processing device,][7.3:route] the update message [1.3:to the node having the determined][7.3:and the node configured to: receive the update message from the

gateway device, wherein the node is configured to be mapped to the] node type”

[14.5]/[20.5]/[26.5]/[32.5]: “to route the update message **[14.5:to the node having the][20.5/26.5/32.5:to a node of the]** node type, at the routing network,”

As described above, Lawson’s “remote event processing block 66 monitors event queue 64,” based on the “global event registry to identify which servers should receive which events” determines “events...that need to be transferred to other servers in the network, and “establish[es] a connection to those servers and transfer[es] the events to the other server. Ex.1005, 13:12-24; *see also* 12:12-20, 12:29-34. As also described above, Choquier discloses routing update messages to servers of particular server groups (server/node type) based on service maps that specify server groups. Ex.1007, 9:50-65, 21:33-53, 22:34-45. Ex.1003, ¶¶161-167.

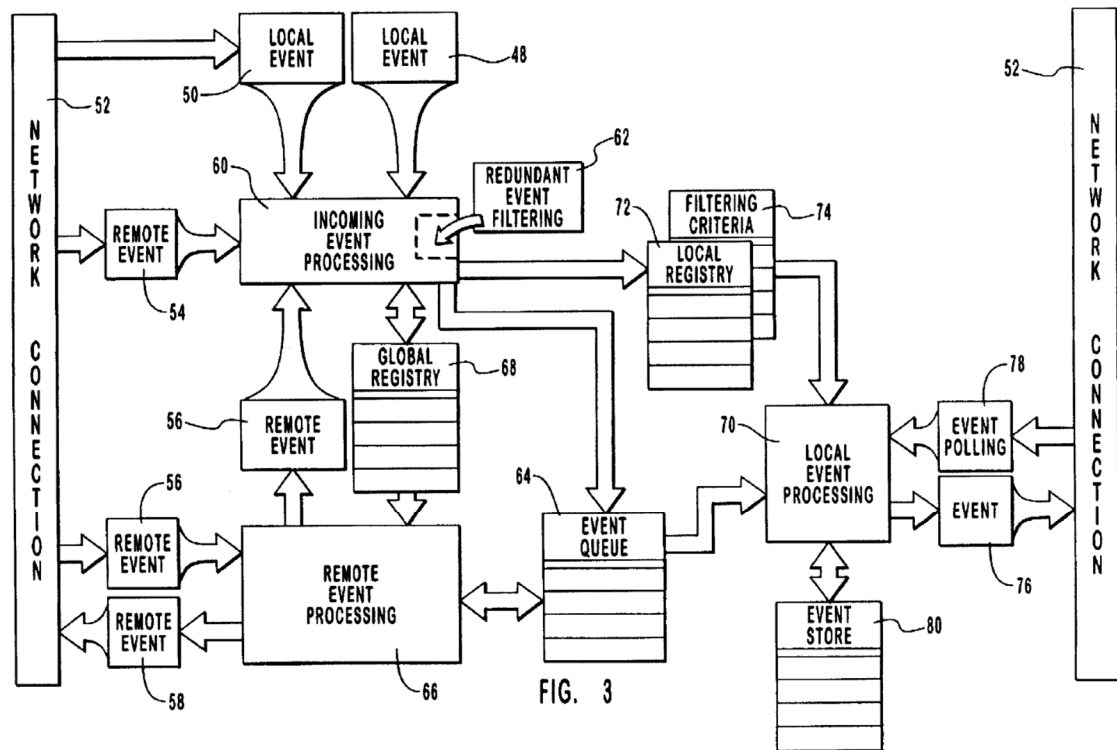
Therefore, the combination meets these elements. Ex.1003, ¶¶161,168.

[1.4]/[7.4]: “[1.4:causing the node, through the update message, to] determine a client, different from the input source, that has registered

for updates of the live object; **[1.4:causing the node to]** route the update message from the node to the client;”

[14.6]/[20.6]/[26.6]/[32.6]: “wherein the node is configured to identify the client device as a registered device and route the update message **[20.6/32.6:containing the update data]** to the client device,”

Lawson’s servers (nodes) include a “local event processing block 70” that monitors event queue 64 for events that should be transferred to a local event consumer” using a “local event registry.” Ex.1005, 13:43-48. See FIG. 3:



The local event registry “identif[ies] which local event consumers should receive which events” and “local event consumers that should receive events queued

in event queue 64 are identified by checking local event registry 72 and any associated filtering criteria.” Ex. 1005, 13: 13:45-52.

Once the server in Lawson identifies which event consumers need to receive the event, the event is routed to those identified event consumers. Ex.1005, 13:43-52, 13:65-67; *see also* Fig. 7 (showing at steps 174 and 182 events routed from a server to an event consumer), 24:60-64, 25:16-18. In Lawson’s phone directory example, “the event notification process on server B dispatches the event packets to registered local event consumers” and “the event packet is received by the company phone directory.” Ex.1005, 24:60-66. In Lawson’s payroll example, server B in Fig. 7 “dispatches the events to the event globalization process running on server A.” Ex.1005, 24:44-47. Then, “[S]erver A proceeds to identify the registered local event consumers for this event...by checking the local event registry on server A,” and “server A transfers the event packet to the local event consumer.” Ex.1005, 25:8-18. Ex.1003, ¶¶169-176.

Lawson as combined with Tsutsumitake also provides this element. Tsutsumitake discloses “each time an event has occurred in the server the event notification has to be sent to the [registered] client[s].” Ex.1006, 4:36-38. Ex.1003, ¶177.

Therefore, the combination meets these elements. Ex.1003, ¶169,178.

[1.5]/[7.5]: “[1.5:causing the client][7.5:wherein the client is adapted] to process the update message and to update the property of the live object.”

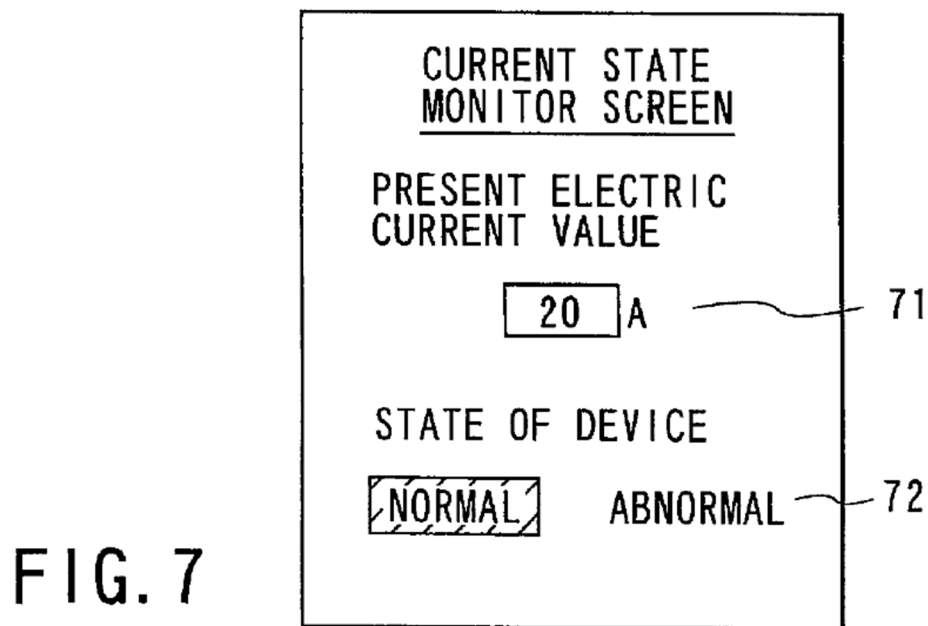
[14.7]/[20.7]/[26.7]/[32.7]: “wherein the client device [14.7:processes][20.7/26.7/32.7:is configured to process] the update message upon receipt to update the property of the live object at the client device.”

In Lawson’s phone directory example in which “event consumer 4 is a company phone directory,” “[a]fter the event has been transferred, the company phone directory process creates a new record[/updates a record] in the company phone book as indicated by step 178.” Ex.1005, 23:64-65, 24:64-66, 25:4-6. In Lawson’s payroll example, “[t]he payroll database process receives the information...and proceeds to create a new entry[/update an entry] in the employee payroll database as indicated by step 186.” Ex.1005, 25:16-21.

Lawson’s updating of data/records corresponds to the ’722 Patent’s “[l]ive objects” that “have associated ‘properties’ which include any modifiable data related to the object or referenced with respect to the object.” Ex.1001, 7:22-24. Ex.1003, ¶¶179-183.

Lawson modified by Tsutsumitake also meets this element. Tsutsumitake discloses “generated events are transmitted in real time to the client 11” and “[t]he browser (client 11) processes a plurality of events (both the event indicating the

present value of current and [] the state of the device being transmitted in real time)
sent from the server 10 in response to one event request issued in accordance with
the displayed page, and reflects on the screen the event processing results on the
current value display section 71 or device state display section 72,” as shown in Fig.
7. Ex. 1006, 13:59-66.



Ex.1003, ¶¶184-185.

Therefore, the combination meets these elements. Ex.1003, ¶179,186.

2. Claims 2 and 8

[2.0]/[8.0]: “[2.0:causing the node][8.0:wherein the node is configured] to extract an object identifier (ID) from the update message[2.0:, to establish a connection with][8.0: and to determine a connection to] the client[2.0:, and to determine if the client has

registered for updates of the live object based on the object ID][8.0:to route the data to the client]”

As explained, *supra* §VII.B.1., the combination provides for routing messages over connections to clients based on the update message, and that update messages have IDs. In addition, Lawson alone or combined with Tsutsumitake disclose “determine if the client has registered for updates”/“a connection to the client” based on the “object ID.”

Lawson discloses “entries...into the local and global event registries” identify “the event and the event consumer wishing to receive notification of the event” via the network. Ex.1005, 15:48-56, 16:35-44. Events have identifiers such as “event producer field,” “type of event field,” etc. and the identifier “preferably identifies an event in the global event registry and/or local event registry so that appropriate servers or event consumers can be identified to receive the event.” Ex.1005, 12:3-26.

Tsutsumitake also discloses “[t]he event request receiving unit 104 asks the connection management unit 105 whether or not an...event ID...and...client ID...have already been registered” and “[i]f they are not registered,...event ID...and...client ID...are registered in the connection management unit 105.” Ex.1006, 11:45-55, Fig. 4. This registration identifies the event and client to receive the event. Ex.1006, 11:56-60.

Therefore, the combination meets these elements. Ex.1003, ¶¶187-194.

3. Claims 3 and 9

[3.0]/[9.0]: “[3.0:causing the node to determine][9.0: wherein the node is configured to determine the connection by determining] at least one client proxy with which the client communicates[3.0;; causing the node] to route the update message to the client proxy[3.0;; and causing the client proxy to route the update message to the client to determine the connection].”

Lawson discloses “[w]hen a local event globalization process” at the server (node) “receives notification of an event that has occurred locally, the local process looks in the local registry in order to identify local event consumers that need notification of the event,” “checks the global event registry to identify any additional servers which also need notification of the event,” and “sends the event to the corresponding event globalization process running on the identified servers.” The “corresponding event globalization processes, in turn, check their local event registries and provide the event to any event consumers identified therein,” such that “local events can be routed to any event consumer in the network without the need to register at each server.” Ex.1005, 4:66-5:12; *see also* Figs. 1-2, 5.

This corresponds to the '722 Patent's “client proxy,” which “merely passes messages among the clients and nodes” and/or “stores client registration information ... [to] determine which client proxies have registered for the messages[] and [to]

send the messages to the appropriate client proxies” to thereafter send to registered clients. Ex.1005, 3:60-4:2. Ex.1003, ¶¶195-198.

Tsutsumitake explicitly discloses a proxy server. Tsutsumitake states “though not shown, a relay server called ‘proxy server’ may be provided between the information providing server 10 and the information display client 11” and the “proxy server receives a request from the client, sends the request to the server as it is, and returns an result from the server directly to the client.” A “plurality of proxy servers” can also be included. Ex.1006, 8:7-14. It would have been obvious to include a proxy server in Lawson’s routing network based on Tsutsumitake. Ex.1003, ¶¶199-203.

Therefore, the combination meets these elements. Ex.1003, ¶¶195,204.

4. Claims 4 and 10

[4.0]/[10.0]: “[4.0:causing the node to maintain][10.0:wherein] client registration information concerning the client connection [10.0:is configured to be maintained at the node].”

As explained, *supra* §§VI.A.,VII.B.1., Lawson’s local registry serves to maintain registration information regarding the client connection, e.g., the client connection to the node. Ex.1005, 4:66-5:12, Figs. 1-2, 5. To the extent Lawson’s disclosure is not explicit, it would be obvious that the node in Lawson maintains

connection information for clients since the node must be able to send events to event consumers.

Therefore, the combination meets these elements. Ex.1003, ¶¶207-212.

5. Claims 5 and 11

[5.0]/[11.0]: “[5.0:causing the node to maintain][11.0:wherein] client registration information concerning the client connection [11.0:is configured to be maintained] at the client proxy.”

As explained, *supra* §§VII.B.1.,VII.B.3., with respect to the independent claims and claims 3 and 9, Lawson’s global registry serves to maintain registration information regarding the client connection at a client proxy. Ex.1005, 4:66-5:12, Figs. 1-2, 5. To the extent Lawson’s disclosure is not explicit, it would be obvious based on Lawson alone or in combination with Tsutsumitake to maintain connection information for clients since the nodes and client proxy must be able to send events to event consumers. As also described with respect to claims 3 and 9, Tsutsumitake discloses a proxy server and it would have been obvious to include a proxy server in Lawson’s routing network based on Tsutsumitake.

Therefore, the combination meets these elements. Ex.1003, ¶¶213-219.

6. Claim 6

[6.0]: “wherein the property of the live object has a direct effect on a visual representation of the live object in a data representation, has an

effect on an internal aspect of the live object and has no effect on the visual representation of the live object in the data representation, or has a direct effect on one aspect of the visual representation of the live object in the data representation and has no effect on other aspect of the visual representation of the live object in the data representation.”

Lawson’s phone directory and payroll examples and Tsutsumitake’s electric plant monitoring example disclose updates having a direct effect on a visual representation (e.g., user sees updated phone number, payroll data, current value 71, device state 72). *See supra*, §VII.B.1.

Tsutsumitake also discloses events updating data that would have no direct effect on a visual representation (e.g., refresh “period,” “target,” “name” in Fig. 8’s HTML code). 14:4-15.

```
< HTML >
< HEAD >
< TITLE > CURRENT MONITOR < /TITLE >
< /HEAD >
< BODY >
< H1 > CURRENT MONITOR SCREEN < /H1 >
< H2 > PRESENT CURRENT VALUE < /H2 >
< HANDLER type=text name=currency > A
< EMBED >
  < EVENT url=http://abc.def/current period=! target=currency >
< /EMBED >
< H2 > STATE OF DEVICE < /H2 >
< HANDLER type=text value=normal name=valid >< HANDLER type=text value=abnormal name=invalid >
< EMBED >
  < EVENT url=http://abc.cef/state target=valid,invalid >
< /EMBED >
< /BODY >
< /HTML >
```

FIG. 8

Therefore, the combination meets these elements. Ex.1003, ¶¶222-229.

7. Claim 12

[12.0]: “wherein:

the gateway device is configured to route the update message to the node adapted to receive messages of more than one message category, and

the node is configured to determine a connection with at least one client proxy with which the client communicates and to route the data to the client proxy.”

Claim 12 is obvious for at least the reasons described regarding the independent claims and claims 3 and 9, *supra* §§VII.B.1.,VII.B.3. Further, it would be obvious that Lawson’s servers could be adapted to receive messages of more than one message category. Choquier demonstrates that it was well-known for servers to be configured to process multiple application and message types because Choquier’s individual servers can be assigned to multiple service and message groups/types. Ex.1007, 2:14-18, 6:24-31, 9:40-49, 10:17-20, 11:44-47.

Therefore, the combination meets these elements. Ex.1003, ¶¶230-237.

8. Claim 13

[13.0]: “wherein the node type is configured to identify the node that receives the update message from the gateway device.”

Claim 13 is obvious for at least the reasons described with respect to the independent claims. As explained, *supra* §VII.B.1., the combination provides a gateway that routes messages to nodes based on node types.

Therefore, the combination meets these elements. Ex.1003, ¶¶239-240.

9. Claims 15, 21, 27, and 33

[15.0]/[21.0]/[27.0]/[33.0]: “wherein [15.0:providing the data representation to the client device includes providing] the live object [21.0/27.0/33.0:of the data representation] [15.0:that][21.0:to cause] causes the client device to register with a client proxy of the routing network.”

As described with respect to the independent claims, Lawson as modified by Tsutsumitake renders obvious that providing the data representation causes a client device to register for updates. *See supra*, §VII.B.1. Additionally, the elements of claims 15, 21, 27, and 33 concerning a client proxy are obvious for similar reasons as claims 3 and 9. Lawson’s nodes use the local and global registries to transfer events between servers in order to reach event consumers (clients) through such transfers. *See supra*, §VII.B.3. As also described with respect to claims 3 and 9,

Tsutsumitake discloses a proxy server and it would have been obvious to include a proxy server in Lawson's routing network based on Tsutsumitake.

Therefore, the combination meets these elements. Ex.1003, ¶¶241-246.

10. Claims 16, 22, 28, and 34

[16.0]/[22.0]/[28.0]/[34.0]: “wherein [16.0:providing the data representation to the client device includes providing] the live object [22.0/28.0/34.0:of the data representation] [16.0:that causes][22.0:is configured to cause][28.0/34.0:causes] the client device to register with the node of the routing network.”

As explained, *supra* §VII.B.1., the combination causes the client device to register for updates with nodes.

Lawson discloses “the event consumer [] send[s] a ‘register for event’ packet...to the event globalization process on the local server” to register the event consumer for the event. Ex.1005, 15:41-44.

Tsutsumitake discloses an HTML file that includes an embedded event (live object), such as shown below.

```
< HTML >  
< HEAD >  
< TITLE > SAMPLE PAGE < /TITLE >  
< /HEAD >  
< BODY >  
< H1 > SAMPLE PAGE < /H1 >  
THIS PAGE INCLUDES EVENT REQUEST  
< EMBED >  
  < EVENT URL=http://abc.def/hij.evt >  
< /EMBED >  
< /BODY >  
< /HTML >
```

FIG. 2

Ex.1006, 4:22-29, 8:39-49, 11:45-55.

Therefore, the combination meets these elements. Ex.1003, ¶¶248-253.

11. Claims 17, 23, 29, and 35

[17.0]/[23.0]/[29.0]/[35.0]: “wherein [17.0:providing the data representation to the client device includes providing][23.0/29.0/35.0: the [23.0/29.0:received] data representation includes] an activation module [15.0/35.0:that is][23.0:that is configured to be] executed by the client device and [15.0:that registers][35.0:that is][23.0/29.0/35.0:adapted to register] the live object with the routing network.”

Both Lawson and Tsutsumitake disclose a client executing an activation module to register a live object with a routing network. Lawson discloses “[w]hen an event consumer receives notification that a certain event happened on an event

producer, the event consumer can take appropriate action,” such as the event consumer causing “an associated process [] to be initiated which receives the custom event and takes appropriate action.” Ex.1005, 1:61-63, 6:2-4. One such function in Lawson may execute is “the event consumer [] send[ing] a ‘register for event’ packet ... to the event globalization process on the local server” to register the event consumer for the event. Ex.1005, 15:41-44; *see also* 7:48-51.

Tsutsumitake discloses that in the webpage the server sends to client containing the event request “[i]t is also possible to adopt a method wherein the event request [to register the client 11 with the server 10] is generated dynamically from a certain program incorporated in the page (e.g. ‘Applet’ of Java developed by Sun Microsystems, or “JavaScript” developed by Netscape).” Ex.1006, 10:20-33. This Java/JavaScript code, for example, parses the webpage to identify “an attribute ‘URL’ in a tag ‘EVENT’ in a tag ‘EMBED’” of the webpage to send to the server to register the client for the event. *See* Ex.1006, 10:20-23; *see also* 10:64-67, 11:45-55, Fig. 4.

These disclosures correspond to the “activation module 124” of the ’722 Patent, which “preferably contains JAVA[®] instructions ... [to] register[] object IDs from the web page 118 downloaded by the client 114 with the routing network 110” by, for example, “pars[ing] 610 the web page 118 received from the server 112 and

identif[ing] the object IDs of the live objects ... to register [] by communicating to the routing network 116.” Ex.1001, 11:1-14, 14:33-35, 14:55-59, Fig. 6.

Therefore, the combination meets these elements. Ex.1003, ¶¶254-257.

12. Claims 18, 24, 30, and 36

[18.0]/[24.0]/[30.0]/[36.0]: “wherein [18.0:providing the activation module includes providing] the activation module [18.0:that] [18.0/24.0:is configured to determine][30.0/36.0:determines] a node type [18.0/30.0:that handles][24.0/36.0:for handling] registration and [18.0:that] [18.0/30.0/36.0:causes][24.0:to cause] the client device to register with the node [18.0:corresponding to][24.0/30.0/36.0:of] the node type.”

Lawson’s and Tsutsumitake’s disclosures correspond to the recited “activation module,” which the ’722 Patent explains is “functionality [on the client] for determining which node type handles which registration requests.” Ex.1001, 19:61-63. One way the ’722 Patent’s “activation module” does so is by “determine[ing] which categories of messages it needs and to which nodes types it should connect [to get them]. The activation module 124 then makes a connection to one node of each relevant type and registers the appropriate object IDs with the appropriate node.” Ex.1001, 21:47-52. Ex.1003, ¶¶258-259.

As described regarding the independent claims, *supra* §VII.B.1., the combination discloses node types and client devices registering with nodes for

updates. As described with respect to claim 17, the combination also includes an activation module.

Specifically, Lawson discloses “the event consumer [] send[ing] a ‘register for event’ packet...to the event globalization process on the local server” to register the event consumer for the event. Ex.1005, 15:41-44. “Registering” for the event is a prerequisite for receiving “notification of events that occur.” Ex.1005, 4:33-37. Lawson discloses the event consumer is configured to determine a server that handles registration and to cause the event consumer to register with that server, which results in the event consumer receiving the events it is registered for. Ex.1003, ¶¶117-119, 259.

Tsutsumitake discloses “the client, which received the requested [web]page information from the server, can automatically issue the event request to the server according to the event request information set in the [web]page information,” including via Java/JavaScript code. Ex.1006, 4:22-25, 10:64-67, 11:45-55, Fig. 4. Tsutsumitake determines the server to receive the event registration via “[t]he page receiving unit 112 [on the client 11, which] analyzes the content of the [web]page sent from the server 10...to determine whether an event request is included in the page.” Ex.1006, 10:56-59. “[I]f the event request is included, the page receiving unit 112 informs the event request unit 113 of the URL of the event and causes the event request unit 113 to send the event request to the server 10 [for registration to

receive events].” Ex.1006, 10:64- 67, Figs. 1, 3. As such, Tsutsumitake discloses Java/JavaScript code (use of which was commonplace) at the client is configured to determine a server that handles registration and cause the client to register with that server to receive events. Ex.1003, ¶¶120-123, 259-260.

Therefore, the combination meets these elements. Ex.1003, ¶¶258,261.

13. Claims 19, 25, 31, and 37

[19.0]/[25.0]/[31.0]/[37.0]: “wherein **[19.0:providing the activation module includes providing]** the activation module **[19.0:that is configured to determine][25.0:determines]** a message category of the data representation and **[19.0:that]** causes the client device to register with **[19.0:a][25.0/31.0/37.0:the]** node having **[19.0:/25.0/31.0:a][37.0:the]** node type corresponding to the message category.”

Lawson and Tsutsumitake disclose program or Java/JavaScript code at the event consumer/client to determine the producer or server an event came from and to register for those events. *See supra*, §VII.B.12. The ’722 Patent explains that “[o]ne way [to assign messages into categories] is to assign all messages from a given input source 710 [(e.g., a publisher, server)] into a certain category.” Ex.1001, 18:52-54; Ex.1002, 1014.

As also described, *supra* §VII.B.1., Lawson discloses message categories based on input source by filtering messages based on event producer (input source),

such that messages are routed to nodes based on the message category. Ex.1005, 5:39-48,12:11-14. Further, Choquier discloses client devices are assigned to communicate with servers of certain server types based on the type of messages/applications being used by the servers. For example, Choquier uses gateways that “serve as intermediaries between the client microcomputers and the application servers,” and, even though “the user typically opens and closes multiple services,” the gateways “rout[e] service requests (from the client microcomputer) to the appropriate application servers” using the “service map.” Ex.1007, 2:32-61.

Therefore, the combination meets these elements. Ex.1003, ¶¶262-267.

VIII. GROUND 2: CLAIMS 3, 5, 9, 11, 12, 15, 21, 27, AND 33 ARE OBVIOUS OVER LAWSON, TSUTSUMITAKE, CHOQUIER, AND HERZ

A. Motivations to Combine

A POSITA would have been motivated to combine the Lawson-Tsutsumitake-Choquier combination with Herz.

As described, *supra* §VI.E., Herz discloses a proxy server that bidirectionally transfers communications between the client and servers, and stores user-specific information in a secure and private manner, such as a user’s “target profile interest summary.” Ex.1009, 5:5-13.

A POSITA would have known that use of proxies between clients and servers was well-known, and would have been motivated to use a proxy such as Herz’s proxy server to provide a common point of connection to clients in Lawson’s system

as well as provide data privacy in storing client registration information, as disclosed in Herz.

A POSITA would also have a high expectation of success in the combination, because Lawson, Tsutsumitake, and Choquier all deal with client server communications over server networks. For the same reasons, the combination would have yielded predictable results well within the skill of POSITA. Ex.1003, ¶¶79-86.

B. Lawson, Tsutsumitake, Choquier, and Herz meet all claim elements.

[3.0]/[9.0]: “determine...client proxy...to route the update message to the client proxy”

[5.0]/[11.0]: maintain at the client proxy “client registration information”

[12.0]: “...node adapted to receive messages of more than one message category, and...determine a connection with...client proxy and to route...to the client proxy.”

[15.0]/[21.0]/[27.0]/[33.0]: “the live object [to cause/causes]...register with a client proxy...”

As described above, *supra* §§VII.B.3.,VII.B.5.,VII.B.7.,VII.B.9, the Lawson-Tsutsumitake-Choquier combination discloses these elements. To the extent the Lawson-Tsutsumitake-Choquier combination does not explicitly disclose a client proxy that routes messages and stores client information, it would have been

obvious, and a POSITA would have been motivated as described above, to include one or more proxy servers as a client proxy, as disclosed in Herz, in the system of the Lawson-Tsutsumitake-Choquier combination to transfer communications between servers and clients and handle client registration and connection information, as disclosed in Herz. *See supra*, §VIII.A.

Herz also discloses a proxy server, and one specifically to route messages: "Any server in the network N may be configured to act as a proxy server in addition to its other functions. Each proxy server provides service to a set of users, which set is termed the "user base" of that proxy server" and "[i]f the user is a subscriber to the news clipping service described below. request R may instruct proxy server S2 to forward to the user all target objects that the news clipping service has sent to proxy server S2 for the user's attention" Ex.1009, 32:20-24, 38:44-47.

Therefore, the combination meets these elements. Ex.1003, ¶¶205-206, 220-221, 238, 247.

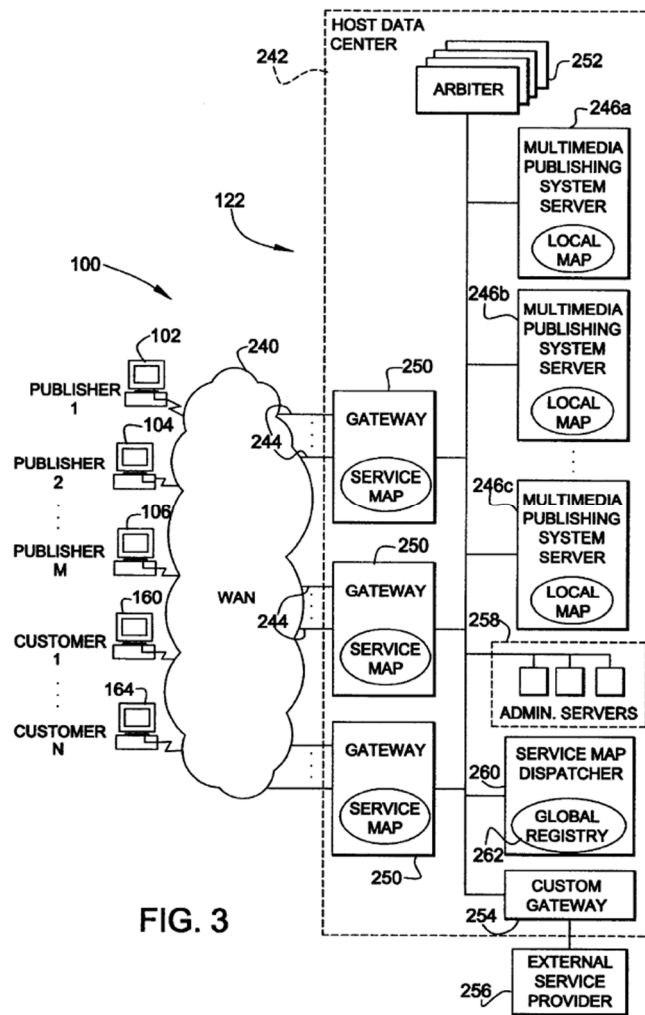
IX. GROUND 3: CLAIMS 1-37 ARE OBVIOUS OVER CHOQUIER, BIRD, AND TSUTSUMITAKE

A. Motivations to Combine

A POSITA would have been motivated to combine Choquier with Bird and Tsutsumitake for numerous reasons. As explained, *supra* §VI., Choquier, Bird, and Tsutsumitake disclose distributed computing systems that provide data updates from

servers to clients (update transactions in Choquier, subscription messages based on publisher type in Bird, real-time event updates in Tsutsumitake).

A POSITA would have been motivated to combine Choquier with Bird to use Choquier's server system for providing publish/subscribe services as taught by Bird. A POSITA would have been further motivated to use message types according to specific publishers as taught in Bird (*see* Ex.1008, [0060]) to map the message types (based on publisher) to certain server groups according to service maps for improved scalability, resource allocation, and load balancing, as taught by Choquier. Ex.1007, 1:52-62, 9:66-10:8, 25:5-35. Choquier specifically specifies assigning different server groups to handle specific topics or sets of topics. Ex.1007, 10:17-20. Choquier's system was designed to be adaptable to provide various "communications-based and information-based on-line services to end users" providing high scalability "without degradation of on-line services" as "the number of end users increases." Ex.1007,1:52-56, 6:8-10. In fact, other patents filed after Choquier used its core architecture for expanded uses, such as a publisher-to-client multimedia publishing system as shown in FIG. 3 of U.S. Patent No. 6,230,173:



Ex. 1018, *see also* Ex.1019 (USP 5,907,837), FIG.3, and Ex.1020 (USP 6,199,082), FIG.3. Ex.1003, ¶¶268-270.

A POSITA would have been further motivated to combine the Choquier-Bird combination with Tsutsumitake to provide for updating webpage content using pushed updates and embedded webpage events. Ex.1006, 3:20-26, 3:3:56-4:6. As described in Tsutsumitake, browser-based, HTML webpage, applications had become widely popular by the late 1990s when Choquier, Bird, and Tsutsumitake

were filed, and, although updating webpage content like news and weather were available, they typically required the client to periodically issue a request for a refreshed webpage, due to issues such as firewalls between clients and servers. Ex.1006, 1:14-3:3. A POSITA would have been motivated to use real-time updates as provided by Tsutsumitake using established server systems such as that disclosed in Choquier and to provide for continuously updating webpage content in publish/subscribe applications like that disclosed by Bird. Ex.1003, ¶¶271-272.

A POSITA would have a high expectation of success, for example because publish/subscribe systems were well-known at the time (*see, e.g.*, Ex.1008, [0002]-[0003]; Ex.1006, 1:60-2:30) and use of server groups as disclosed in Choquier was well-known and preferable for efficiency, scalability, and redundancy. Ex.1007, 1:52-65. For the same reasons, the combination would have yielded predictable results. Ex.1003, ¶273.

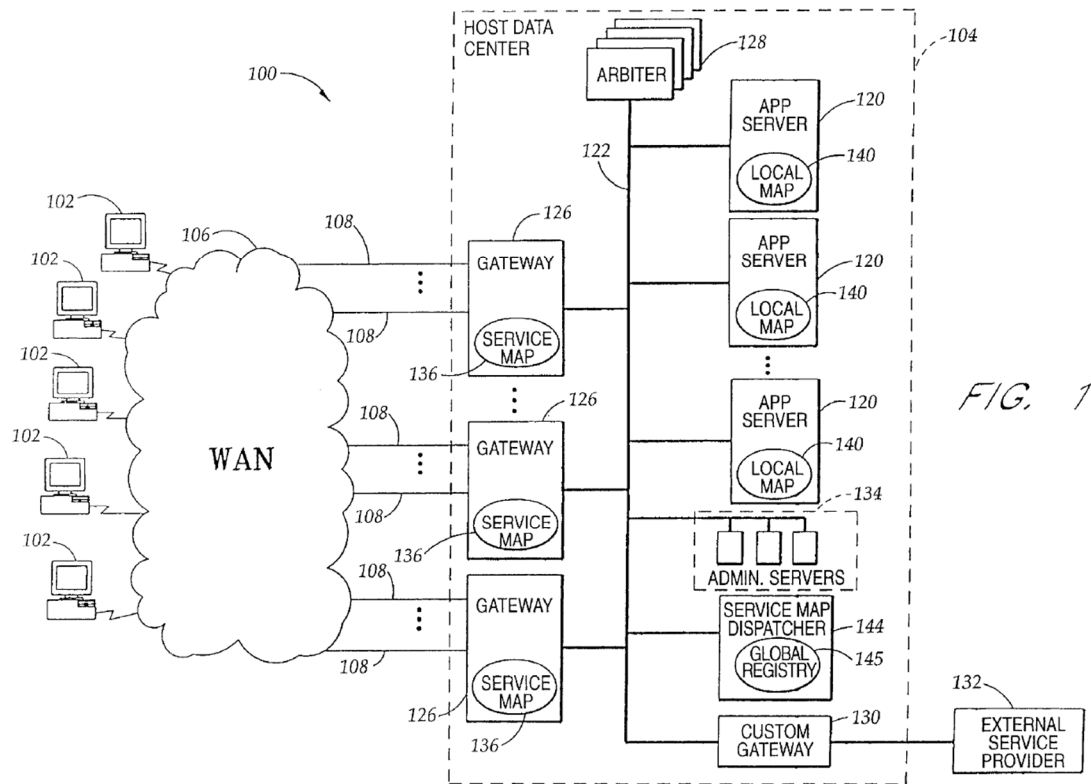
B. Choquier, Bird, and Tsutsumitake meet all claim elements.

1. Independent Claims 1, 7, 14, 20, 26, and 32

See §VII.B.1. and Ex.1021 for full claim language.

[1.0]/[7.0]/[14.0]/[20.0]/[26.0]/[32.0]: [Preambles] “...comprising:”

To the extent the preambles are limiting, Choquier provides systems and routing networks: “large-scale computer networks for providing information-related services and communications services to end users.” Ex.1007, 1:14-17.



Choquier discloses methods. Choquier describes processes for routing “update transactions” from via servers to client devices. Ex.1007, 9:50-65, 10:4-12, 10:67-11:3, 21:33-53, 22:34-45.

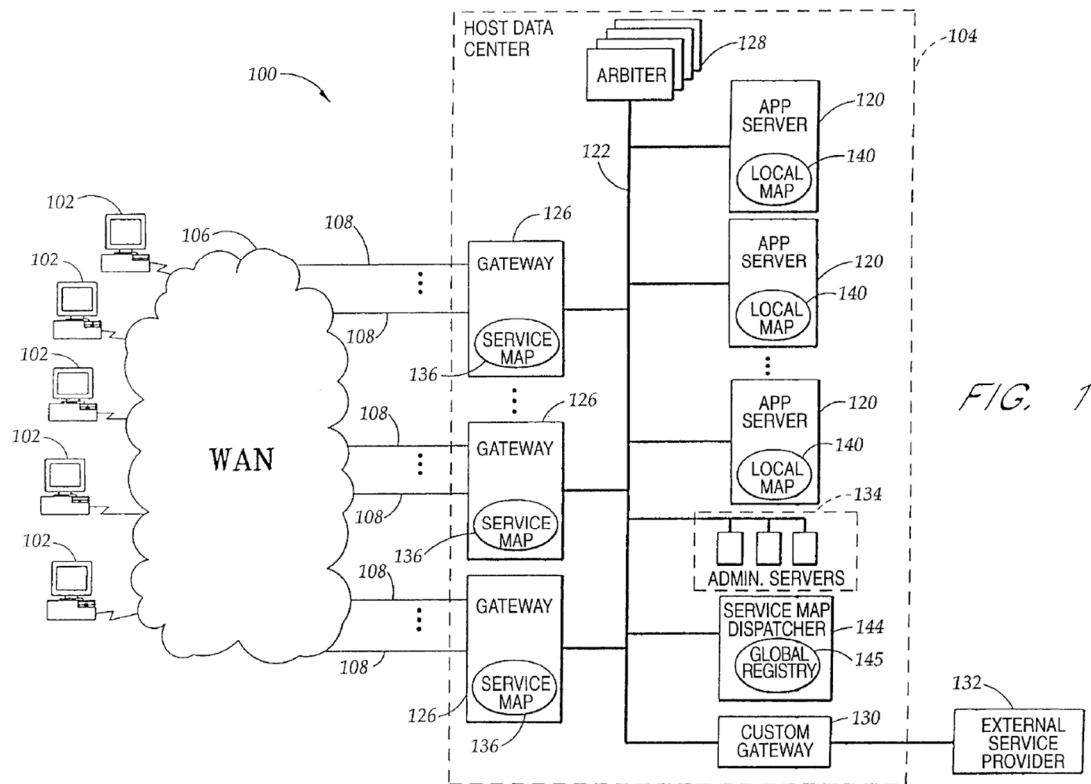
Choquier’s discloses “microcomputers” and “computer-readable mediums.” Ex.1007, 2:26-47, 6:32-47, 26:23-39, 27:19-28:13.

Therefore, the combination meets these elements. Ex.1003, ¶¶279-284.

[14.1]/[20.1]/[26.1]/[32.1]: providing/provide “...a data representation ...includes...live object”

As described above, *supra* §VI.C., Choquier discloses a routing network in which data received by a host data center 104 is routed to client devices 102 using

application servers arranged into “service groups.” Ex.1007, 2:1-4. See FIG. 1 below.



Data routed to the clients, coupled to the routing network, via the application servers can originate from “an external data provider connected to the host data center 104 by the Internet” (an input source). Ex.1007, 9:57-65. Ex.1003, ¶285.

As described above, *supra* §IX.A., it would have been obvious to modify Choquier’s architecture to provide publish/subscribe system based on Bird and using an updating browser/webpage based on Tsutsumitake. Tsutsumitake, as also described, *supra* §VII.B.1., discloses providing a data representation that includes a

live object to a client device, where the client device is a different device from the input source (a server).

In Tsutsumitake, a “page transmission unit 103” at a server transmits an HTML file to client 11. Ex.1006, 8:39-49.

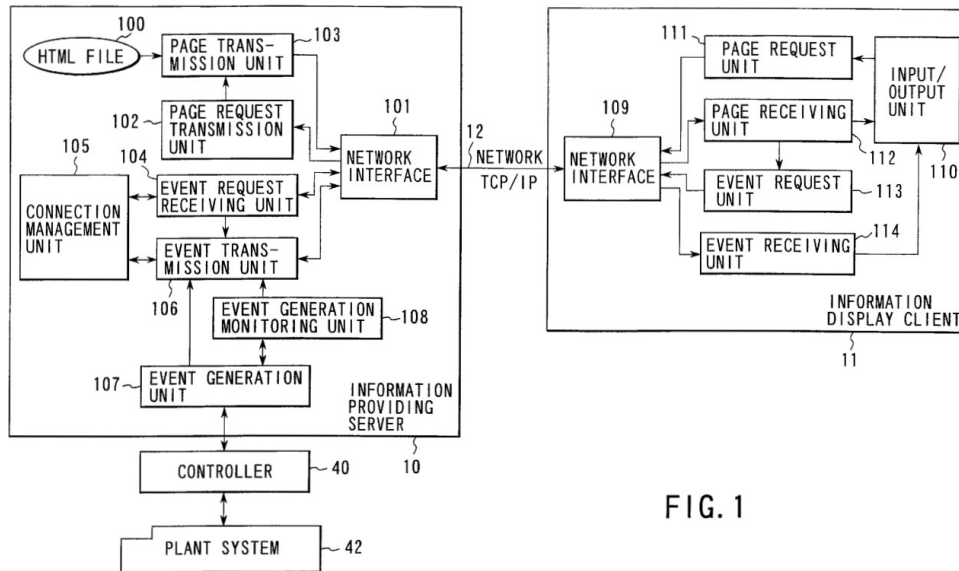


FIG. 1

Fig. 2 shows an example of the webpage, including an embedded event, sent by server 10 to client 11.

```
< HTML >
< HEAD >
< TITLE > SAMPLE PAGE < /TITLE >
< /HEAD >
< BODY >
< H1 > SAMPLE PAGE < /H1 >
THIS PAGE INCLUDES EVENT REQUEST
< EMBED >
  < EVENT URL=http://abc.def/hij.evt >
< /EMBED >
< /BODY >
< /HTML >
```

FIG. 2

The client recognizes a “live object” (embedded URL) and “informs an event request unit 113” at the server “of the URL of the event request.” Ex.1006, 9:61-68, 10:20-23. Ex.1003, ¶¶286-290.

Therefore, the combination meets these elements. Ex.1003, ¶291.

[14.2]/[20.2]/[26.2]/[32.2]: determining/determine “an object identifier (ID) of the live object...to register for updates...”

Choquier specifies that communication links are established via between clients and servers of Choquier’s network to facilitate exchange of information such as “update transactions.” Ex.1007, 8:8-12. Bird specifies that client devices initiate requests to register for content updates. Ex.1008, [0039]. As explained, *supra* §VII.B.1., Tsutsumitake discloses determining an object ID and registering for updates in response to receiving a data representation including a live object. Tsutsumitake

discloses the client “analyzes the content of the [web]page sent from the server,” and, if “the event request is included” the client uses the URL (object ID) of the event and sends “the event request to the server.” Ex.1006, 10:56-67, Figs. 1, 3.

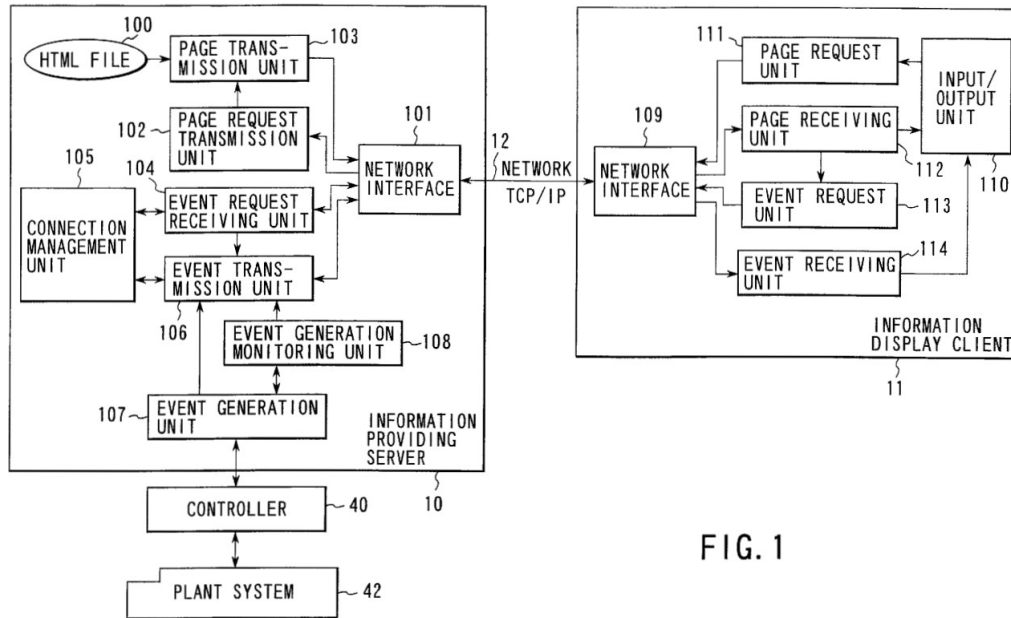
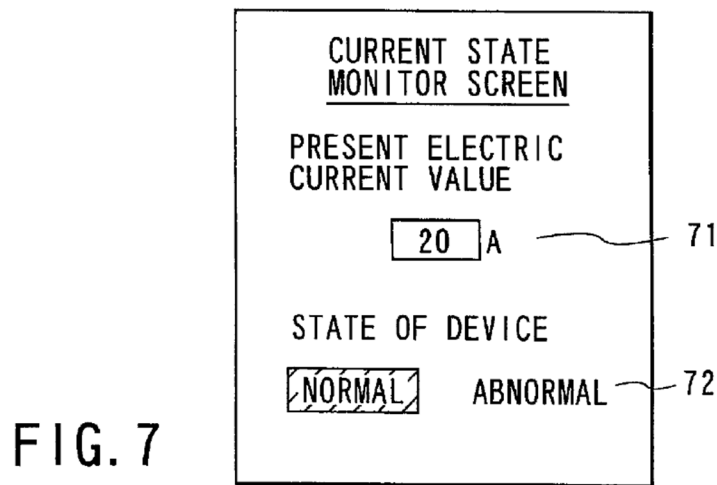


FIG. 1

This causes the client to register for updates because it “activate[s] the event notification,” and can then “be dynamically changed according to the event notification sent from the server.” Ex.1006, 4:22-29; *see also* 1:7-43, 7:63-8:23, 9:43-68, 13:32-14:3, Fig. 7.



Connection information is also provided as part of the registration request. At the server 10, “[t]he event request receiving unit 104 asks the connection management unit 105 whether or not an identifier (event ID) of the event requested by the event request and an identifier (client ID) of the client 11 at the request originating point have already been registered” and “[i]f they are not registered, a pair of the...event ID...and...client ID...are registered in the connection management unit 105.” Ex.1006, 11:45-55.

As described above, *supra* §IX.A., the combination would not be limited to Tsutsumitake’s example live object. Bird discloses updating stock prices. Ex.1008, [0039], [0044], [0065]. Choquier discloses various information-based updating content, like bulletin board messages, news, or stock quotes. Ex.1007, 6:8-22, 23:26-33. As disclosed in Bird, publish/subscribe was “a known information distribution mechanism

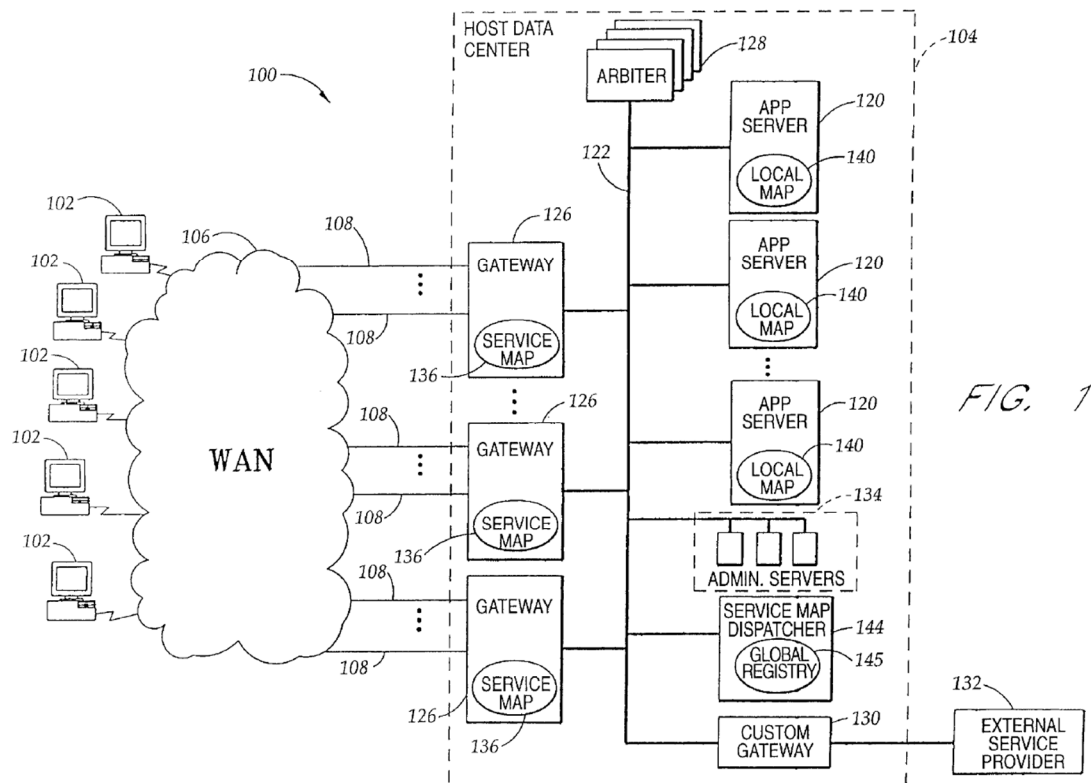
in which information providers (Publishers) send information electronically across a network to a community of information users (Subscribers).” Ex.1008, [0002].

Therefore, the combination meets these elements. Ex.1003, ¶¶292-296.

[1.1]/[7.1]: receiving/receive “...an update message...identifying a live object and containing data for updating...;”

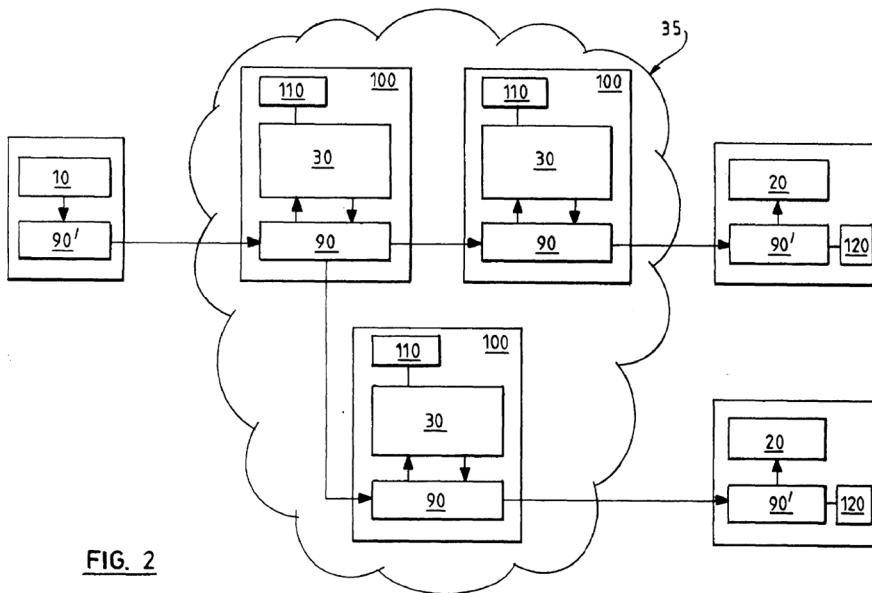
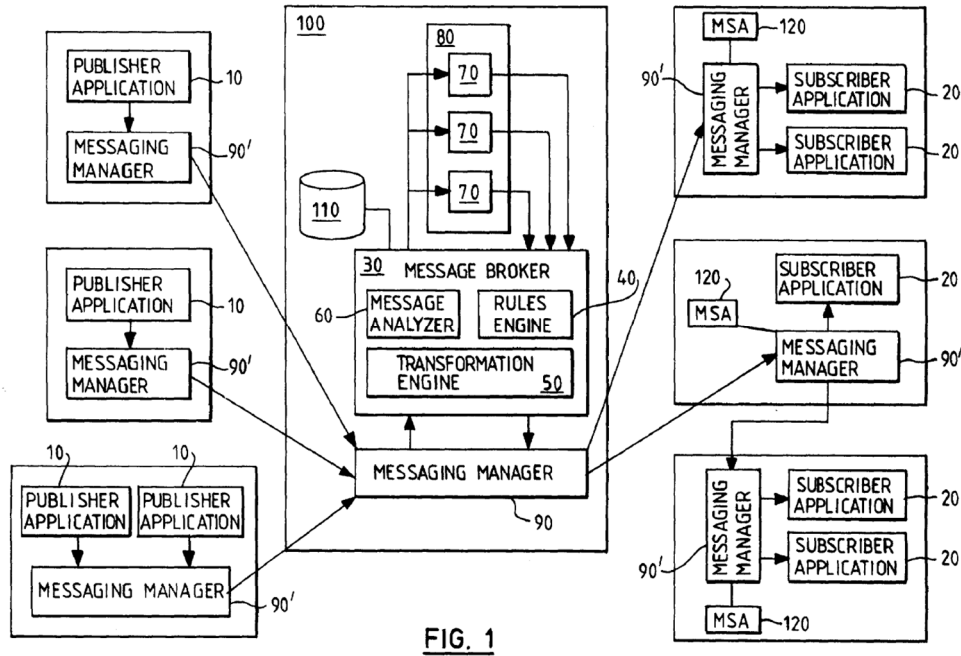
[14.3]/[20.3]/[26.3]/[32.3]: sending/route/providing “...an update message...identifies the live object and contains update data”

Choquier discloses an input source, different from a client device, can send/route an update message to Choquier’s network.



In Choquier, update messages (“update transactions”) for updating services can be sent/routed from input sources such as “an external data provider connected to the host data center 104 by the Internet.” Ex.1007, 9:50-65, 10:4-12. Choquier’s update messages can be received by a gateway device because Choquier discloses using a gateway 130 that “links the host data center 104 to one or more external service providers 132” and Arbiter microcomputers 128 for disseminating data to appropriate server groups using service maps 120. Ex.1007, 6:37-47, 23:9-19. Ex.1003, ¶¶298-301.

Bird also discloses various input sources (multiple publisher applications) sending published messages (with “message headers” and “content fields”) to a routing network (computer system(s) 100) for eventual routing of messages to subscriber applications on client devices. Ex.1008, [0003], [0036]-[0039]. See. FIGS. 1-2 of Bird:



Tsutsumitake discloses “[i]f an event has occurred, the server 10 sends it to the client 11 with use of [a] connection ID. Subsequently, as shown in Fig. 6, each time an event has occurred, it is sent [to client 11] repeatedly with use of the

[connection] ID.” Ex.1006, 13:1- 5; *see also* 12:10-15, Figs. 1, 5. “[E]ach time an event has occurred in the server the event notification has to be sent to the [registered] client[s].” Ex.1006, 4:36-38; *see also* 12:23-28.

Therefore, the combination meets these elements. Ex.1003, ¶¶302-305.

[1.2]/[7.2]: “identifying a category...based on the input source;
[1.2:determining][7.2:determine] a node having a node type...based
on a mapping...controlling...traffic...”

[14.4]/[20.4]/[26.4]/[32.4]: “wherein a gateway device...identify a
category...based on the input source, to determine a node type...”

As described, *supra* §VI.C., Choquier discloses assigning servers to server groups (node types) based on the type of services they are configured to provide using a service map. Ex.1007, 2:1-4, 2:14-18, 6:8-31, 9:40-49, 11:44-47. Service groups can include one server or multiple servers, and it is possible for a server to be assigned to multiple service groups. Ex.1007, 2:14-18, 6:24-31, 9:40-49, 11:44-47.

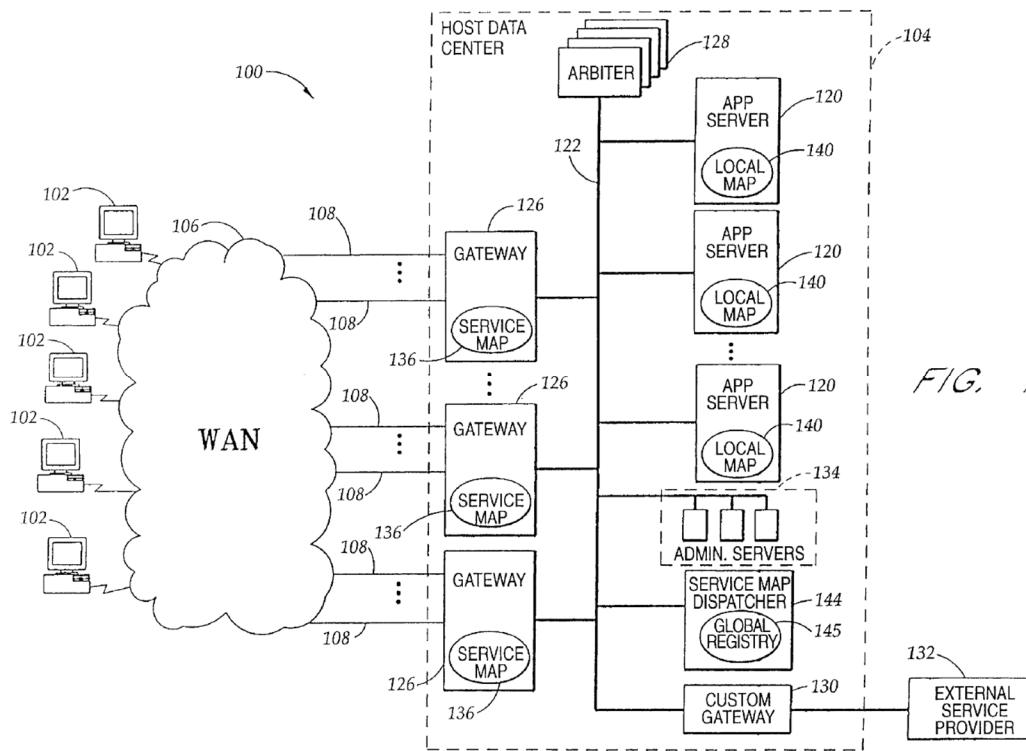


FIG. 1

When an “update transaction” is received, an Arbiter stores in a log information such as a transaction ID and an originator of the update, and then “dispatches the update transactions” to servers in the server group (servers of a particular server type) according to the service map to update at servers content to be provided to client devices, such as new messages for an online bulletin board application or an update to an interactive game. Ex.1007, 9:50-65, 21:33-53, 22:34-45.

Choquier further discloses service groups assist with controlling traffic on the network. First, a POSITA would have understood that sending communications to only certain server groups would reduce network traffic in Choquier’s system.

Second, Choquier discloses load balancing using server loads recorded in the service maps and performance of dynamic server allocation to server groups to control overall network load and efficient use of network resources. Ex.1007, 23:26-35, 23:49-24:3. Ex.1003, ¶¶306-308.

Bird discloses it was well-known for publish/subscribe systems to categorize messages according to input source, that is, messages from specific publishers (“company name” “specific publisher organisation”) and that headers of received messages can be analyzed to determine the identification of the sender for the message for use in subsequent routing of the message. Ex.1008, [0003], [0044]-[0045], [0060]. As described above, *supra* §IX.A., a POSITA would have been motivated to map Choquier’s server groups, which Choquier discloses can be organized by type to handle a specific topic or a set of topics, to handle messages from specific publishers (message categories based on input source), as disclosed by Bird. Ex.1007, 1:52-67, 10:17-20, 25:5-21; Ex.1008, [0003], [0044]-[0045], [0060]. Ex.1003, ¶¶309-311.

Therefore, the combination meets these elements. Ex.1003, ¶311.

[1.3]/[7.3]: routing/route “...the update message...to...node type”

[14.5]/[20.5]/[26.5]/[32.5]: “...route the update message...to...node type”

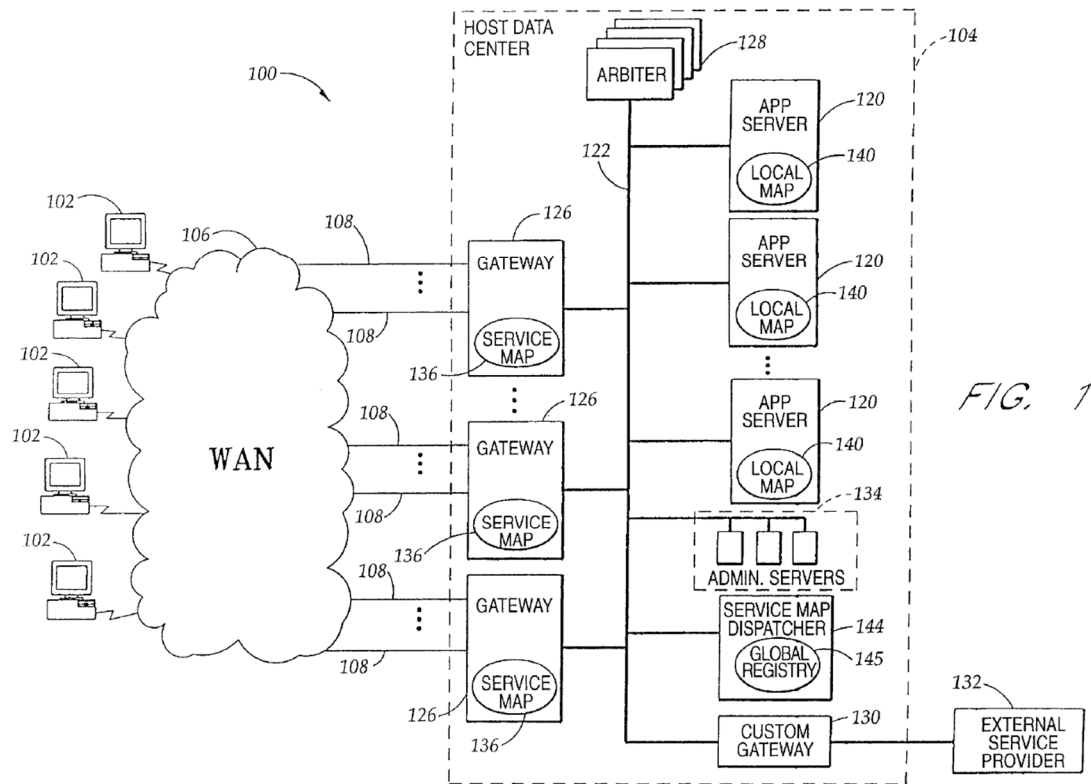
As described above, Choquier discloses routing update messages to servers of particular server groups (server/node type) based on service maps that map servers to server groups associated with message categories. Ex.1007, 9:50-65, 10:17-20, 21:33-53, 22:34-45.

Therefore, the combination meets these elements. Ex.1003, ¶¶312-314.

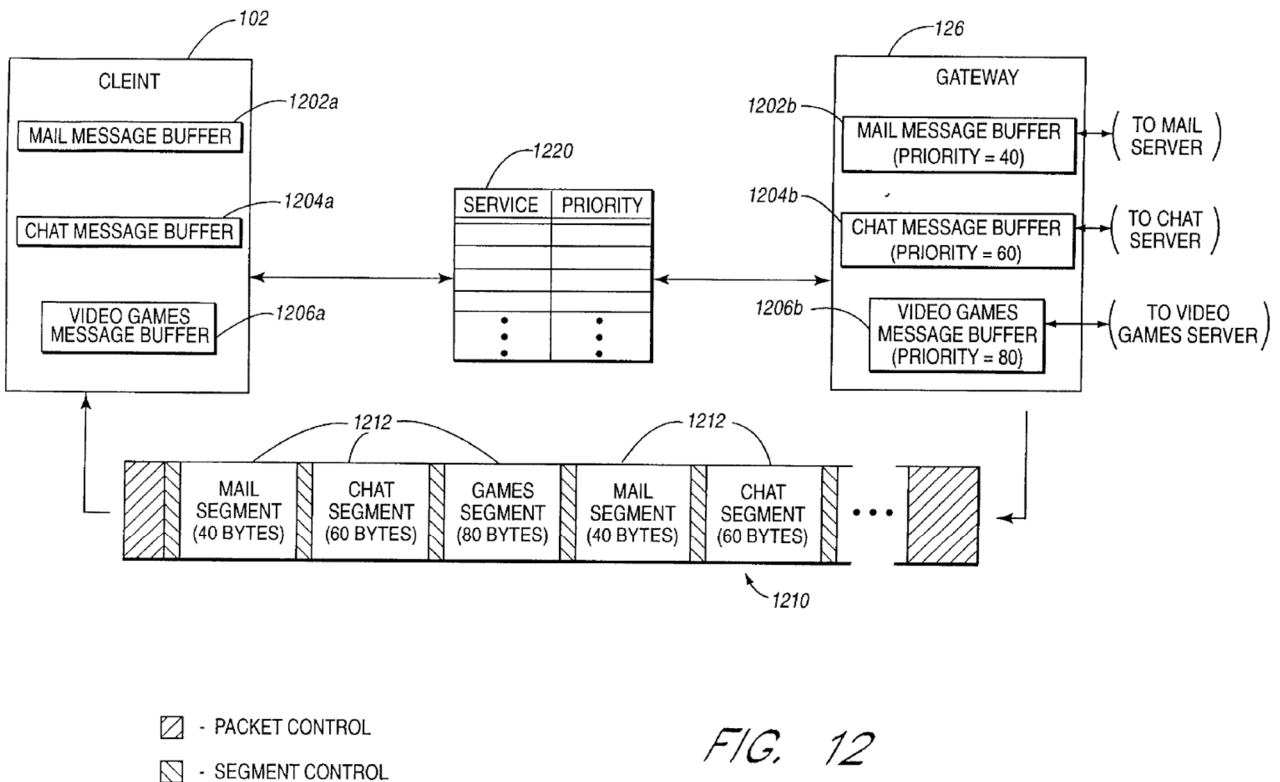
[1.4]/[7.4]: “...determine a client...that has registered for updates...; ...route the update message...to the client;”

[14.6]/[20.6]/[26.6]/[32.6]: “...identify the client device...and route the update message...to the client device,”

Choquier discloses servers communicating with client devices, through associated gateways 126, to route updates messages from the servers to clients.



Servers and clients communicate over service instance channels which can use TCP/IP. Ex.1007, 8:17-24. Messages are routed to the clients in packets. Ex.1007, 18:24-19:60, FIGS. 11A-11B. FIG. 12 illustrates messages being sent to the client (which can be multiplexed if the client requested multiple categories). Ex.1007, 20:16-31.



Bird's servers use a message analyzer and rules engine to analyze message headers for the publisher identity (input source), determine which clients are subscribed to publisher messages, and to route messages to the clients. Ex.1008, [0043]-[0044], [0052]. It would have been obvious to include such functionality when implementing a publish/subscribe service in Choquier's system so that servers can determine which subscribed clients need to receive messages. Likewise, Tsutsumitake discloses "each time an event has occurred in the server the event notification has to be sent to the [registered] client[s]" and "generated events are transmitted in real time to the client 11." Ex.1006, 4:36-38, 13:56-57. Ex.1003, ¶¶315-325.

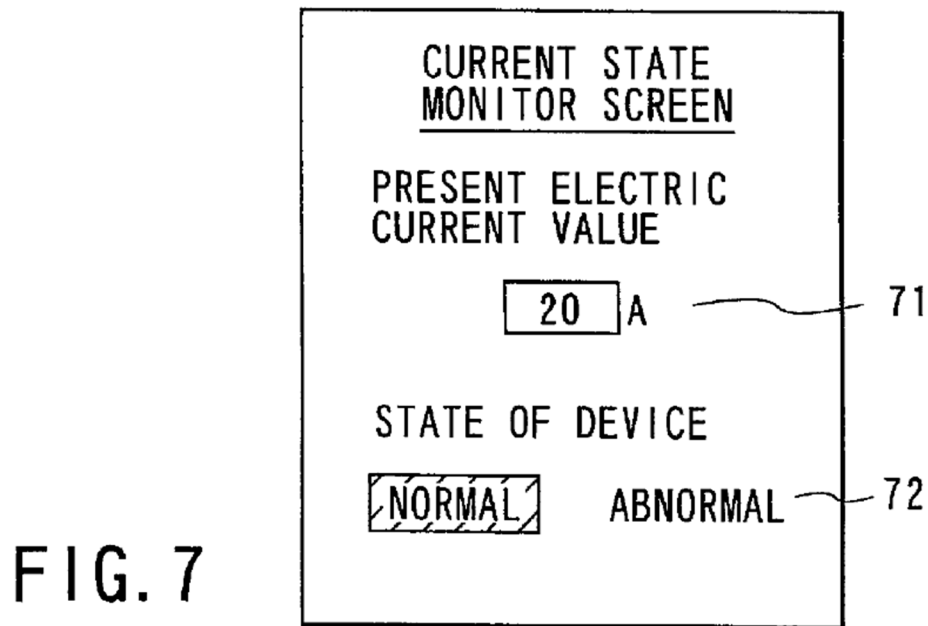
Therefore, the combination meets these elements. Ex.1003, ¶326.

[1.5]/[7.5]: “...client...to update...the live object.”

[14.7]/[20.7]/[26.7]/[32.7]: “...client device...update...the live object...”

As described above, Choquier discloses packets providing content updates are disseminated through its network to clients. Ex.1007, 9:50-58, 18:24-19:60, 20:16-31.

Additionally, *see supra* §VII.B.1., Tsutsumitake discloses updating live webpage objects: “[t]he browser (client 11) processes a plurality of events (both the event indicating the present value of current and [] the state of the device being transmitted in real time) sent from the server 10 in response to one event request issued in accordance with the displayed page, and reflects on the screen the event processing results,” as shown in Fig. 7. Ex. 1006, 13:59-66.



Therefore, the combination meets these elements. Ex.1003, ¶¶327-329.

2. Claims 2 and 8

[2.0]/[8.0]: “node...extract...ID from the update message...[establish/determine]...connection [with/to] the client...and determine...client...registered”

As explained, *supra* §IX.B.1., the combination provides for routing messages to clients over connections to clients based on the update message, and that update messages have IDs.

Bird discloses its servers analyze message headers for the topic and/or sender (input source) identification, determines which clients are subscribed to received the message, and routes the message to the determined clients. Ex.1008, [0043]-[0044],

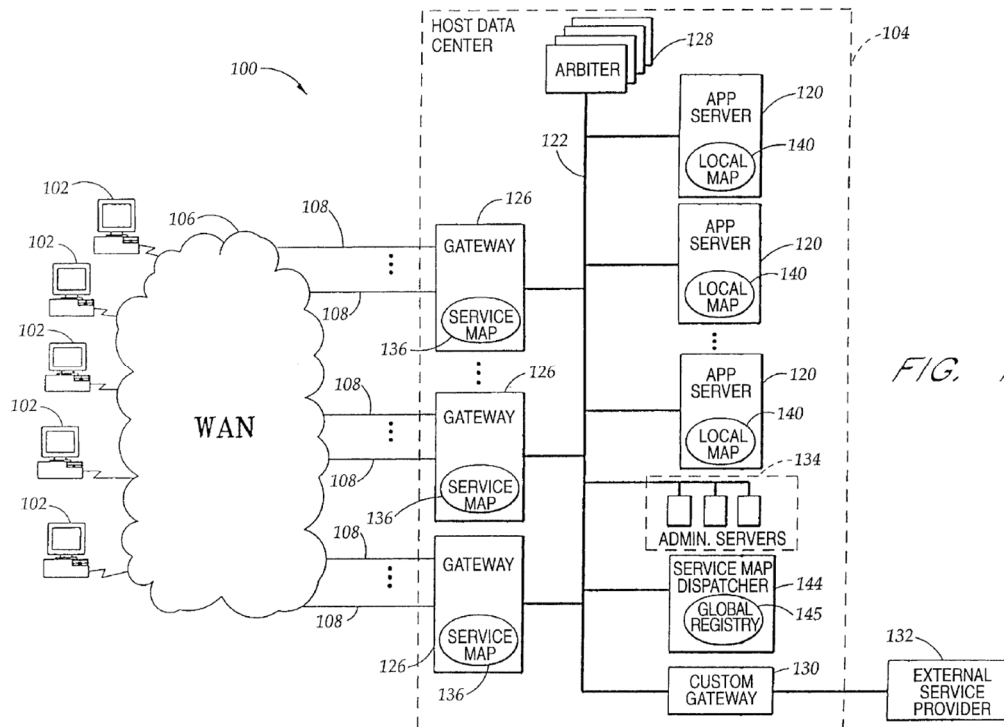
[0052]. Tsutsumitake also discloses associated event IDs with client IDs to determine which clients need event updates and “are registered in the connection management unit 105.” Ex.1006, 11:45-55, Fig. 4.

Therefore, the combination meets these elements. Ex.1003, ¶¶330-334.

3. Claims 3 and 9

[3.0]/[9.0]: “determine...client proxy...to route the update message to the client proxy”

Choquier discloses gateways 126 that establish communication links with one or multiple clients to allow the client(s) to “communicate with multiple servers 120.” Ex.1007, 8:8-16. Messages are passed “between the gateway 126 and the server 120” and sent to the clients via the gateways 126. Ex.1007, 8:17-35.



This corresponds to the '722 Patent's "client proxy," which "merely passes messages among the clients and nodes" and/or "stores client registration information ... [to] determine which client proxies have registered for the messages[] and [to] send the messages to the appropriate client proxies" to thereafter send to registered clients. Ex.1005, 3:60-4:2.

Additionally, as explained *supra* §VII.B.3., Tsutsumitake explicitly discloses a proxy server and it would be obvious to include a proxy server as a relay server in Choquier's system, as such was well-known. Ex.1006, 8:7-14.

Therefore, the combination meets these elements. Ex.1003, ¶¶335-341.

4. Claims 4 and 10

[4.0]/[10.0]: maintain at the node "client registration information"

As explained, *supra* §§VI.C.,IX.B.1., Choquier establishes communication links with nodes and Choquier as modified would maintain information on client subscriptions/registrations. Ex.1007, 8:8-35.

Therefore, the combination meets these elements. Ex.1003, ¶¶344-353.

5. Claims 5 and 11

[5.0]/[11.0]: maintain at the client proxy "client registration information"

As explained, *supra* §§IX.B.1.,IX.B.3., with respect to the independent claims and claims 3 and 9, Choquier's gateways 126 act as client proxies (or, alternatively,

Tsutsumitake discloses using proxy servers as relays) to maintain information on communication links with clients. Ex.1007, 8:17-35; Ex.1006, 8:7-14.

Therefore, the combination meets these elements. Ex.1003, ¶¶354-357.

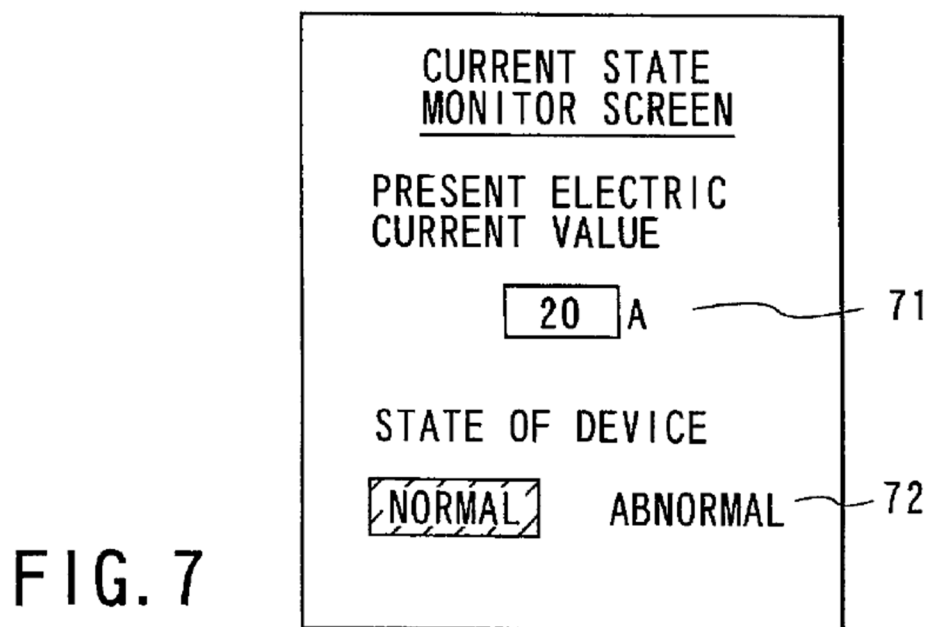
6. Claim 6

[6.0]: "...property of the live object has...effect on...live object"

Choquier's services cause visual updates, such as updates to news, chat messages, bulletin boards, stock quotes, etc. Ex.1007, 6:8-22.

Bird also discloses visual updates, such as updating stock prices. Ex.1008, [0039], [0044], [0065].

Tsutsumitake discloses updating its live object causes a visual change in electric current value or device state. Ex.1006, FIG. 7.



Tsutsumitake also discloses updating data that would have no direct effect on a visual representation (e.g., refresh “period,” “target,” “name” in Fig. 8’s HTML code). Ex.1006, 14:4-15.

```
< HTML >
< HEAD >
< TITLE > CURRENT MONITOR < /TITLE >
< /HEAD >
< BODY >
< H1 > CURRENT MONITOR SCREEN < /H1 >
< H2 > PRESENT CURRENT VALUE < /H2 >
< HANDLER type=text name=currency > A
< EMBED >
< EVENT url=http://abc.def/current period=1 target=currency >
< /EMBED >
< H2 > STATE OF DEVICE < /H2 >
< HANDLER type=text value=normal name=valid > < HANDLER type=text value=abnormal name=invalid >
< EMBED >
< EVENT url=http://abc.cef/state target=valid, invalid >
< /EMBED >
< /BODY >
< /HTML >
```

FIG. 8

Therefore, the combination meets these elements. Ex.1003, ¶¶360-364.

7. Claim 12

[12.0]: “...node adapted to receive messages of more than one message category, and...determine a connection with...client proxy and to route...to the client proxy.”

Regarding proxies, claim 12 is obvious for similar reasons as claims 3 and 9, *see supra* §IX.B.3. Choquier also demonstrates it was well-known for servers to be configured to process multiple application and message types because Choquier’s individual servers can be assigned to multiple service and message groups/types:

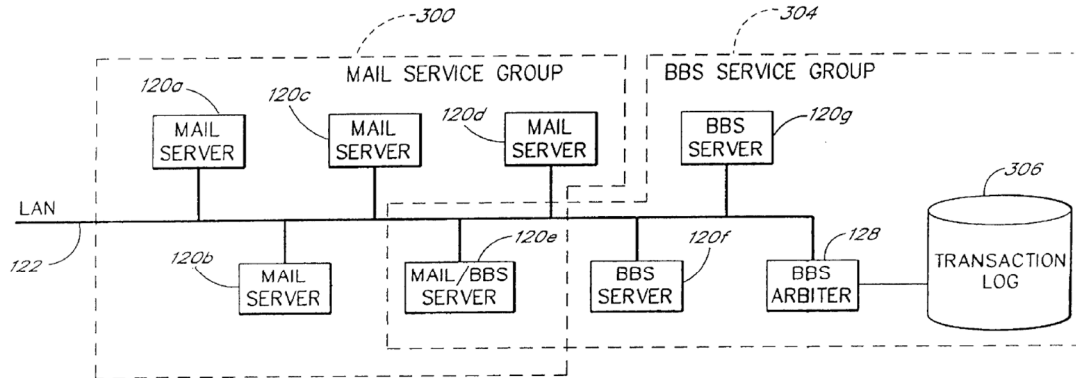


FIG. 3

Ex.1007, 2:14-18, 6:24-31, 9:40-49, 10:17-20, 11:44-47.

Therefore, the combination meets these elements. Ex.1003, ¶¶365-367.

8. Claim 13

[13.0]: "...the node type...identify the node that receives the update message..."

Claim 13 is obvious for at least the reasons described with respect to the independent claims. As explained, *see supra*, §IX.B.1., the combination provides a gateway that routes messages to nodes based on node types.

Therefore, the combination meets these elements. Ex.1003, ¶¶369-371.

9. Claims 15, 21, 27, and 33

[15.0]/[21.0]/[27.0]/[33.0]: “the live object [to cause/causes]...register with a client proxy...”

Choquier as modified by Tsutsumitake renders obvious that providing the data representation causes a client device to register for updates. *See supra*, §IX.B.1. Additionally, the elements of claims 15, 21, 27, and 33 concerning a client proxy are obvious for similar reasons as claims 3 and 9. *See supra* §IX.B.3.

Therefore, the combination meets these elements. Ex.1003, ¶¶372-375.

10. Claims 16, 22, 28, and 34

[16.0]/[22.0]/[28.0]/[34.0]: “the live object [to cause/causes]...register with the node...”

As explained, *supra* §IX.B.1., the combination causes the client device to register for updates with nodes. Choquier specifies that communication links are established via clients with servers of Choquier’s network to facilitate exchange of information like “update transactions.. Ex.1007, 8:8-12. Bird specifies that client devices initiate requests to register for content updates. Ex.1008, [0039]. Tsutsumitake discloses a live object causes registration: an HTML file including an embedded event (live object) is provided and the client “automatically issue[s] the event request to the server according to the event request information set in the

[web]page information, and...activate[s] the event notification” to register for updates. Ex.1006, 4:22-29, 8:39-49, 11:45-55.

```
< HTML >
< HEAD >
< TITLE > SAMPLE PAGE < /TITLE >
< /HEAD >
< BODY >
< H1 > SAMPLE PAGE < /H1 >
THIS PAGE INCLUDES EVENT REQUEST
< EMBED >
  < EVENT URL=http://abc.def/hij.evt >
< /EMBED >
< /BODY >
< /HTML >
```

FIG. 2

Therefore, the combination meets these elements. Ex.1003, ¶¶377-381.

11. Claims 17, 23, 29, and 35

[17.0]/[23.0]/[29.0]/[35.0]: “an activation module [to register/registers]...the live object...”

Choquier specifies that communication links are established via clients with servers of Choquier’s network to facilitate exchange of information like updates to “update transactions” type services. Ex.1007, 8:8-12. Bird discloses a messaging manager connected between a subscriber application running on client devices and the

server, and client devices register for messages with the server. Ex.1008, [0036]-[0039].
Tsutsumitake also discloses the claimed activation module. *See supra*, §VII.B.10.

These disclosures correspond to the “activation module 124” of the ’722 Patent, which “preferably contains JAVA[®] instructions ... [to] register[] object IDs from the web page 118 downloaded by the client 114 with the routing network 110” by, for example, “pars[ing] 610 the web page 118 received from the server 112 and identif[ing] the object IDs of the live objects ... to register [] by communicating to the routing network 116.” Ex.1001, 11:1-14, 14:33-35, 14:55-59, Fig. 6.

Therefore, the combination meets these elements. Ex.1003, ¶¶382-387.

12. Claims 18, 24, 30, and 36

[18.0]/[24.0]/[30.0]/[36.0]: “the activation module [determine/determines] a node type...[for handling/handles] registration and...[to cause/causes] the client device to register with...the node type.”

As described regarding the independent claims, *supra* §IX.B.1., the combination discloses node types and client devices registering with nodes for updates.

In Choquier, gateways 126 establish communications between clients and servers and use the service maps to determine the server groups that are “currently handling the requested services.” Ex.1007, 10:32-41. Thus, the combination would

allow for client devices to register for updates with nodes types (server groups) that handle registration.

Tsutsumitake also discloses the claimed activation module. *See supra*, §VII.B.10-11.

Choquier's and Tsutsumitake's disclosures correspond to the '722 Patent's recited "activation module that is configured to determine a node type that handles registration and that causes the client device to register with the node corresponding to the node type," which the '722 Patent explains is "functionality [on the client] for determining which node type handles which registration requests," such as making "a connection to one node of each relevant type and registers the appropriate object IDs with the appropriate node." Ex.1001, 19:61-63, 21:47-52.

Therefore, the combination meets these elements. Ex.1003, ¶¶388-393.

13. Claims 19, 25, 31, and 37

[19.0]/[25.0]/[31.0]/[37.0]: "the activation module [to determine/determines] a message category of the data representation...causes the client device to register with...node having...node type..."

The combination discloses program code at the event consumer/client to determine the producer or server an event came from and to register for those events. *See supra*, §IX.B.12., regarding claims 18, 24, 30, and 36. The '722 Patent explains

that “[o]ne way [to assign messages into categories] is to assign all messages from a given input source 710 [(e.g., a publisher, server)] into a certain category.” Ex.1001, 18:52-54; Ex.1002, 1014.

Bird discloses it was well-known for publish/subscribe systems to categorize messages according to input source topics, that is, messages from specific publishers (“company name” “specific publisher organisation”), a messaging manager connected between a subscriber application running on client devices and the server, and that clients subscribe/register via a messaging manager (activation module) to receive messages based on input source categories. Ex.1008, [0003], [0044]-[0045], [0060].

Choquier uses gateways that “serve as intermediaries between the client microcomputers and the application servers,” and, even though “the user typically opens and closes multiple services,” the gateways “rout[e] service requests (from the client microcomputer) to the appropriate application servers” using the “service map.” Ex.1007, 2:32-61. Moreover, Choquier specifies that server groups can be assigned to handle a specific topic or set of topics. Ex.1007, 10:17-20.

Thus, the combination provides an activation module and a client registering with a node having a node type (server group) corresponding to the message category (specific publisher).

Therefore, the combination meets these elements. Ex.1003, ¶¶394-400.

X. GROUND 4: CLAIMS 3, 5, 9, 11, 12, 15, 21, 27, AND 33 ARE OBVIOUS OVER CHOQUIER, BIRD, TSUTSUMITAKE, AND HERZ

A. Motivations to Combine

Under *KSR*, a POSITA would have been motivated to combine the Choquier-Bird-Tsutsumitake combination with Herz.

As described, *supra* §VI.E., Herz discloses a proxy server that bidirectionally transfers communications between the client and servers, and stores user-specific information in a secure and private manner, such as a user’s “target profile interest summary.” Ex.1009, 5:5-13.

A POSITA would have known that use of proxies between clients and servers was well-known, and would have been motivated to use a proxy such as Herz’s proxy server to provide a common point of connection to clients in Choquier’s system as well as provide data privacy in storing client registration information, as disclosed in Herz.

A POSITA would also have a high expectation of success in the combination, because Choquier, Bird, and Tsutsumitake all deal with client server communications over server networks. For the same reasons, the combination would have yielded predictable results well within the skill of POSITA. Ex.1003, ¶¶274-277.

B. Choquier, Bird, Tsutsumitake, and Herz meet all claim elements.

[3.0]/[9.0]: “determine...client proxy...to route the update message to the client proxy”

[5.0]/[11.0]: maintain at the client proxy “client registration information”

[12.0]: “...node adapted to receive messages of more than one message category, and...determine a connection with...client proxy and to route...to the client proxy.”

[15.0]/[21.0]/[27.0]/[33.0]: “the live object [to cause/causes]...register with a client proxy...”

As described above, *supra* §§IX.B.3., IX.B.5., IX.B.7., IX.B.9., the Choquier-Bird-Tsutsumitake combination discloses or suggests these elements. To the extent the Choquier-Bird-Tsutsumitake combination does not explicitly disclose a client proxy that routes messages and stores client information, it would have been obvious, and a POSITA would have been motivated as described above, to include one or more proxy servers as a client proxy, as disclosed in Herz, in the system of the Choquier-Bird-Tsutsumitake combination to transfer communications between servers and clients and handle client registration and connection information, as disclosed in Herz. *See supra*, §X.A.

Herz also discloses a proxy server, and one specifically to route messages: "Any server in the network N may be configured to act as a proxy server in addition

to its other functions. Each proxy server provides service to a set of users, which set is termed the "user base" of that proxy server” and “[i]f the user is a subscriber to the news clipping service described below. request R may instruct proxy server S2 to forward to the user all target objects that the news clipping service has sent to proxy server S2 for the user's attention" Ex.1009, 32:20-24, 38:44-47.

Therefore, the combination meets these elements. Ex.1003, ¶¶342-343, 358-359, 368, 376.

XI. THERE ARE NO DISCRETIONARY FACTORS FAVORING DENIAL OF INSTITUTION

A. The *Fintiv* factors either favor institution or are neutral

Institution should not be denied because all factors favor institution or, at the very least, are neutral. Factor 1 is neutral because neither party has requested a stay. *See, e.g., Sand Revolution II, LLC v. Continental Intermodal Grp.-Trucking LLC*, IPR2019-01393, Paper 24 at 7 (PTAB. June 16, 2020) (informative).

Factor 2 is neutral. Although no trial date is yet set in IV’s case against American, the schedule jointly submitted to the district court would provide for a trial more than eighteen months after the filing of this petition. And although trial is set for July 27, 2026 in the case against Southwest, that date often changes. For example, the most recent time-to-trial data show that the median time to trial for jury trials in the Western District of Texas is 34.9 months. Table T-3—U.S. District Courts—Trials Statistical Tables For The Federal Judiciary (December 31, 2024),

U.S. Courts, <https://www.uscourts.gov/data-news/data-tables/2024/12/31/statistical-tables-federal-judiciary/t-3> (last visited March 28, 2025). The Board has instituted IPR based on other petitions with similar facts. *Ericsson Inc. v. XR Commc 'ns LLC*, IPR2024-00613, Paper 9, 33-34 (PTAB Oct. 9, 2024).

Factor 3 favors institution because the parties have expended limited resources in the litigations and Petitioners acted expeditiously—*i.e.*, by filing the Petition around six months of the original complaints, before claim construction briefing, in the early stages of fact discovery, and before expert discovery. Exs.1022, 1023. *Sotera Wireless, Inc. v. Masimo Corp.*, IPR2020-01019, Paper 12 at 16-17 (PTAB Dec. 1, 2020) (precedential).

Factor 4 favors institution because Petitioners challenge all claims of the '722 Patent, although IV asserts only a subset in litigation. Exs.1022, 1023, 1024. Petitioners also stipulate that, if the Petition is instituted, Petitioners will not pursue in the District Court litigation any ground raised or that could have been reasonably raised in an IPR. *Sotera*, Paper 12 at 18.

Factor 5 favors institution because institution will promote judicial efficiency and integrity by reducing the number of issues to be addressed in the parallel litigations. *Sand Revolution*, Paper 24 at 12.

Factor 6 favors institution because Petitioners have set forth a strong case of unpatentability. *Sand Revolution*, Paper 24 at 13.

B. An analysis under 35 U.S.C. § 325(d) also favors institution

The prior art references relied on herein were not cited or substantively addressed during prosecution, and nothing suggests that their teachings relied on herein—much less the combination of those teachings as informed by a POSA’s knowledge and expert testimony—were considered by the examiner. *See, e.g., Mylan Pharm. Inc., v. Regeneron Pharm., Inc.*, IPR2022-01226, Paper 22 at 27-29 (PTAB Jan. 11, 2023). Further, this petition relies on the references Lawson and Choquier, which were not cited in the *ex parte* reexamination request filed by Unified Patents, LLC (Control No. 90/019,662), and seeks review of additional claims that are not part of the reexamination. Although Tsutsumitake and Bird were cited in the reexamination, they were not addressed in detail by the Examiner, as the Examiner rejected the grounds of the reexamination due to a different primary reference not used in this petition. Ex.1017, 2-4. Accordingly, this petition should not be denied under §325(d).

XII. CONCLUSION

Claims 1-37 are unpatentable under Pre-AIA §103. Petitioners request cancellation of claims 1-37.

Dated: May 9, 2025

Respectfully submitted,

By: /Keith D. Harden/
Keith D. Harden
Registration No. 74,472

CERTIFICATE OF WORD COUNT UNDER 37 C.F.R. § 42.24(d)

The undersigned certifies that the word count of the foregoing PETITION FOR *INTER PARTES* REVIEW, excluding the table of contents, table of authorities, exhibit list, mandatory notices, certificate of service, certificate of word count, and any appendix of exhibits or claim listing, is 13,917.

Dated: May 9, 2025

/Keith D. Harden/
Keith D. Harden
Registration No. 74,472

CERTIFICATE OF SERVICE

The undersigned certifies that a copy of the foregoing PETITION FOR *INTER PARTES* REVIEW and all exhibits identified herein were served on May 9, 2025 at the following address of record for the subject patent:

Brandon R. Theiss
Volpe Koenig
30 South 17th Street, 18th Floor
Philadelphia, PA 19103

via FedEx Overnight

Courtesy Copy via electronic mail to Patent Owner's Litigation Counsel:

Jonathan K. Waldrop
Kasowitz Benson Torres LLP
333 Twin Dolphin Drive, Suite 200
Redwood Shores, CA 94065
jwaldrop@kasowitz.com
IntellectualVentures@kasowitz.com

Dated: May 9, 2025

/Keith D. Harden/
Keith D. Harden
Registration No. 74,472