IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of:	Gregory G. Raleigh	
U.S. Patent No.:	9,615,192	Attorney Docket No.: 39843-0166IP1
Issue Date:	April 4, 2017	
Appl. Serial No.:	15/211,430	
Filing Date:	July 15, 2016	
Title:	MESSAGE LINK SERVER WITH PLURAL MESSAGE DELIVERY TRIGGERS	

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

PETITION FOR *INTER PARTES* REVIEW OF UNITED STATES PATENT NO. 9,615,192 PURSUANT TO 35 U.S.C. §§ 311–319, 37 C.F.R. § 42

Google Exhibit 1064
Google v. Headwater
IPR2026-00271

TABLE OF CONTENTS

I. IPR Requirements 1
- **A.** Standing 1
- **B.** Challenge, Relief Requested 1
- **C.** Claim Construction 2
- **D.** Level of Ordinary Skill in the Art 3

II. ’192 PATENT 3

III. THE CHALLENGED CLAIMS ARE UNPATENTABLE 5
- **A.** Ground 1 Claims are Rendered Obvious by TS-23.140 5
  - 1. TS-23.140 5
  - 2. Claim Analysis 6
- **B.** Ground 2 Claims are Rendered Obvious by TS123.140 And Shen 33
  - 1. Shen 33
  - 2. Combination of TS-23.140 and Shen 34
  - 3. Claim Analysis 38
- **C.** Ground 3: Claim 4 is Rendered Obvious by TS-23.140 and Ellison 39
  - 1. Ellison 39
  - 2. Analysis 40
- **D.** Ground 4: Claim 8 Is Rendered Obvious by TS-23.140 And Rakic 45
  - 1. Rakic 45
  - 2. Analysis 46
- **E.** Ground 5 Claims are Rendered Obvious by Houghton and Munson 49
  - 1. Houghton 49
  - 2. Munson 51
  - 3. Combination of Houghton and Munson 52
  - 4. Analysis 55
- **F.** Ground 6 Claims Are Rendered Obvious by Houghton, Munson, and Shen 85
  - 1. Combination of Houghton, Munson, and Shen 85
  - 2. Analysis 87
- **G.** Ground 7: Claim 4 is Rendered Obvious by Houghton, Munson, and Ellison 88
  - 1. Combination of Houghton, Munson, and Ellison 88
  - 2. Analysis 90
- **H.** Ground 8: Claim 8 Is Rendered Obvious by Houghton, Munson, and Rakic 92
  - 1. Analysis 92

IV. PTAB DISCRETION SHOULD NOT PRECLUDE INSTITUTION.....95
A. §325(d).....95
B. §314(a).....97

V. CONCLUSION AND FEES 98

VI. MANDATORY NOTICES UNDER 37 C.F.R § 42.8(a)(1).....98
A. Real Party-In-Interest Under 37 C.F.R. § 42.8(b)(1).....98
B. Related Matters Under 37 C.F.R. § 42.8(b)(2).....98
C. Lead And Back-Up Counsel Under 37 C.F.R. § 42.8(b)(3).....99
D. Service Information 99

APPENDIX OF CLAIMS

CLAIM 1	
1pre	A message link server comprising:
1a	a transport services stack to maintain a respective secure message link through an Internet network between the message link server and a respective device link agent on each of a plurality of wireless end-user devices,
1b	each of the wireless end-user devices comprising multiple software components authorized to receive and process data from secure message link messages received via a device link agent on that device;
1c1	an interface to a network to receive network element messages from a plurality of network elements,
1c2	the received network element messages comprising respective message content and requests for delivery of the respective message content to respective wireless end-user devices, the respective message content including data for, and an identification of, a respective one of the authorized software components; and
1d1	a message buffer system including a memory and logic,
1d2	the memory to buffer content from the received network element messages for which delivery is requested to a given one of the wireless end-user devices,
1d3	the logic to determine when one of a plurality of message delivery triggers for the given one of the wireless end-user devices has occurred, wherein for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a message delivery trigger, and for at least one of the message delivery triggers, the trigger is an occurrence of an asynchronous event with time-critical messaging needs, and
1d4	upon determining that one of the message delivery triggers has occurred, the logic further to supply one or more messages comprising the buffered content to the transport services stack for delivery on the secure message link maintained between the transport services stack and a device link agent on the given one of the wireless end-user devices.
CLAIM 2	
2	The message link server of claim 1, further comprising an encrypt function to encrypt the one or more messages supplied to the transport

	services stack for delivery on the secure message link maintained between the message link server and the device link agent on the given one of the wireless end-user devices.
CLAIM 3	
3	The message link server of claim 2, wherein the encrypted one or more messages are transported to the device link agent on the given one of the wireless end-user devices using one or more of encryption on the transport services stack, IP (Internet Protocol) layer encryption, and tunneling.
CLAIM 4	
4	The message link server of claim 1, wherein the device link agent executes in a secure execution environment on at least one of the devices, and at least one of the software components executes outside of the secure execution environment on that device.
CLAIM 5	
5pre	The message link server of claim 1,
5a	wherein the transport services stack is further to receive, over each of the respective secure message links, upload messages forwarded by the respective device link agents from at least a subset of the device software components,
5b	each of the upload messages identifying a corresponding one of the network elements to which the device respective software component has requested delivery,
5c	the network server system using the interface to a network to deliver content from the upload messages to the respective identified network elements.
CLAIM 6	
6	The message link server of claim 1, wherein at least one of the one or more messages for delivery by the transport services stack comprises multiple identifier/data pairs.
CLAIM 7	
7	The message link server of claim 1, the device messaging agent on at least one of the wireless end-user devices further to initiate the respective secure Internet data message link to the transport services stack.
CLAIM 8	
8	The message link server of claim 1, further comprising a secure server to provide secure authorization signatures to the given one of the wireless end-user devices, the secure authorization signatures indicating the

	authorized software components that are allowed to receive data from secure message link messages via the message link server.
CLAIM 9	
9	The message link server of claim 1, wherein one of the message delivery triggers is the expiration of a periodic timer.
CLAIM 11	
11	The message link server of claim 1, wherein one of the message delivery triggers is the receipt of a transmission on the respective secure message link from the device link agent of the given one of the wireless end-user devices, or a response generated to a transmission received from that device link agent.
CLAIM 12	
12	The message link server of claim 11, wherein the transmission is a heartbeat message generated by the given device link agent, or a request received from the given device link agent.
CLAIM 13	
13	The message link server of claim 1, wherein one of the message delivery triggers is the receipt of a particular network element message from one of the network elements.
CLAIM 15	
15pre	A method of operating a message link server, comprising:
15a	maintaining a respective secure message link through an Internet network between the message link server and a respective device link agent on each of a plurality of wireless end-user devices,
15b	each of the wireless end-user devices comprising multiple software components authorized to receive and process data from secure message link messages received via a device link agent on that device;
15c1	receiving network element messages from a plurality of network elements,
15c2	the received network element messages comprising respective message content and requests for delivery of the respective message content to respective wireless end-user devices, the respective message content including data for, and an identification of, a respective one of the authorized software components; and
15d1	buffering content from the received network element messages for which delivery is requested to a given one of the wireless end-user devices;

15d2	determining when one of a plurality of message delivery triggers for the given one of the wireless end-user devices has occurred, wherein for at least some of the received network element messages, the receipt of such a message is not a message delivery trigger, and for at least one of the message delivery triggers, the trigger is an occurrence of an asynchronous event with time-critical messaging needs; and
15d3	upon determining that one of the message delivery triggers has occurred, supplying one or more messages comprising the buffered content for delivery on the secure message link maintained between the message link server and a device link agent on the given one of the wireless end-user devices.

EXHIBITS

SAMSUNG-1001	U.S. Patent No. 9,615,192 to Raleigh ("the '192 Patent")
SAMSUNG-1002	Excerpts from the Prosecution History of the '192 Patent ("the Prosecution History")
SAMSUNG-1003	Declaration and Curriculum Vitae of Dr. Patrick Traynor
SAMSUNG-1004	3GPP TS 23.140 v6.9.0 (2005-03); 3rd Generation Partnership Project; Technical Specification Group Terminals; Multimedia Messaging Service (MMS); Functional Description; Stage 2 ("TS-23.140")
SAMSUNG-1005	U.S. Patent Pub. No. 2006/0190720 to Ozaki et al. ("Ozaki")
SAMSUNG-1006	WO 2008/048075 A1 to Lee et al. ("Lee")
SAMSUNG-1007	WO 2006/077283 Al to Houghton et al ("Houghton")
SAMSUNG-1008	**RESERVED**
SAMSUNG-1009	U.S. Patent No. 7,925,717 to Chou et al. ("Chou")
SAMSUNG-1010	Open Mobile Alliance; Multimedia Messaging Service Architecture Overview (MMSARCH) specification, *available at* *https://www.openmobilealliance.org/release/MMS/V1_1-20040715-A/OMA-WAP-MMS-ARCH-V1_1-20040715-A.pdf*
SAMSUNG-1011	Open Mobile Alliance; OMA-ERELD-MMS-v1_2-20030923-C, Enabler Release Definition for MMS Version 1.2," *available at* https://www.openmobilealliance.org/release/MMS/V1_2-20030923-C/OMA-ERELD-MMS-V1_2-20030923-C.pdf
SAMSUNG-1012	U.S. Patent No. 7,509,487 to Lu et al. ("Lu")

SAMSUNG-1013	Technical Specification Group Services and System Aspects Meeting #19, TSGS#19(03)0167, European Telecommunications Standards Institute February 2003 (available at https://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_19/Docs/PDF/SP-030167.pdf)
SAMSUNG-1014	U.S. Patent Pub. No. 2005/0207379 to Shen et al. ("Shen")
SAMSUNG-1015	U.S. Patent Pub. No. 2009/0282256 to Rakic et al. ("Rakic")
SAMSUNG-1016	Declaration of Friedhelm Rodermund
SAMSUNG-1017	U.S. Patent Pub. No. 2009/0240807 Al ("Munson")
SAMSUNG-1018	EP Patent Application EP1853044B1 to Shenfield ("Shenfield")
SAMSUNG-1019	U.S. Patent No. 7,082,615 B1 ("Ellison")
SAMSUNG-1020	Memorandum, Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings, June 21, 2022, available at *https://www.uspto.gov/sites/default/files/documents/interim_proc_discretionary_denials_aia_parallel_district_court_litigation_memo_20220621_.pdf*
SAMSUNG-1021	Docket Control Order, *Headwater Research LLC v. Samsung Electronics Co.*, 2:23-cv-00103-JRG-RSP (EDTX), filed October 27, 2023
SAMSUNG-1022	**RESERVED**

SAMSUNG-1023	Samsung Stipulation letter regarding IPR grounds in District Court Litigation
SAMSUNG-1024	**RESERVED**
SAMSUNG-1025	**RESERVED**
SAMSUNG-1026	**RESERVED**
SAMSUNG-1027	IETF RFC 793, Transmission Control Protocol (Sept. 1981) (available at https://www.ietf.org/rfc/rfc793.txt)
SAMSUNG-1028	The TLS Protocol Version v 1.0 (Jan. 1999) (available at https://datatracker.ietf.org/doc/html/rfc2246)
SAMSUNG-1029	Complaint for Patent Infringement, *Headwater Research LLC v. Samsung Electronics Co.*, Case No. 2:23-cv-00103 (Dkt. 1, Mar. 10, 2023)
SAMSUNG-1030	Needham et al., "Using Encryption for Authentication in Large Networks of Computers" (ACM, Vol. 21, No. 12, Dec. 1978) ("Needham")
SAMSUNG-1031	Schroeder et al., "A Hardware Architecture for Implementing Protection Rings" (ACM, Vol. 15, No. 3, Mar. 1972) ("Schroeder")
SAMSUNG-1032	Saltzer et al., "The Protection of Information in Computer Systems" (IEEE Proceedings, Vol. 63, No. 9, Sept. 1975) ("Saltzer")
SAMSUNG-1033	Li et al., "Symbian OS platform security model," available at https://www.usenix.org/system/files/login/articles/73507-li.pdf (Login Magazine, Aug. 2010)

SAMSUNG-1034	Philip Zimmermann, "Pretty Good Privacy: RSA Public Key Cryptography for the Masses" PGP User's Guide. Version 1.0, June 1991), available at https://www.tech-insider.org/free-software/research/acrobat/910605.pdf ("Zimmerman")
SAMSUNG-1035	B. Ramsdell, S/MIME Version 3 Message Specification, IETF RFC 2633, June 1999, available at https://data-tracker.ietf.org/doc/html/rfc2633 ("Ramsdell")
SAMSUNG-1036	Mostafa, "Transporting data between wireless applications using a messaging system—MMS" (Wireless Comms. and Mobile Computing, July 7, 2006) ("Mostafa")
SAMSUNG-1037	Proof of Service, *Headwater Research LLC v. Samsung Electronics Co.*, Case No. 2:23-cv-00103 (Dkt. No. 6, Mar. 15, 2023)

I. IPR Requirements

A. Standing

'192 Patent is IPR eligible. This petition is being filed within one-year of service of complaint against Samsung. SAMSUNG-1037. Samsung is not barred/estopped from requesting this review.

B. Challenge, Relief Requested

Samsung requests an IPR of the Challenged Claims on the below grounds. Dr. Traynor provides supporting testimony in his Declaration. SAMSUNG-1003, ¶¶1-457.

Ground	Claim(s)	§103
1	1, 5-7, 9, 11-13, 15	TS-23.140
2	2-3	TS-23.140-Shen
3	4	TS-23.140-Ellison
4	8	TS-23.140-Rakic
5	1, 5-7, 9, 11-13, 15	Houghton-Munson
6	2-3	Houghton-Munson-Shen
7	4	Houghton-Munson-Ellison
8	8	Houghton-Munson-Rakic

The '192 Patent claims priority to multiple provisional applications as early as January 28, 2009 ("Critical Date"). SAMSUNG-1001, Cover. Petitioner does

not concede that the claimed priority date is correct, but applies prior art predating it.

Reference	Filing Date	Publication Date	Basis
TS-23.140	-	March 2005 (SAMSUNG-1016[1])	102(b)
Shen	03/18/2005	09/22/2005	
Houghton	01/19/2006	07/27/2006	
Ellison	09/22/2000	07/25/2006	
Munson	03/21/2008	09/24/2009	102(e)
Rakic	05/12/2008	11/12/2009	

C. Claim Construction

Petitioner submits that no claim constructions are necessary because "claim terms need only be construed to the extent necessary to resolve the controversy." *Wellman, Inc. v. Eastman Chem. Co.*, 642 F.3d 1355, 1361 (Fed. Cir. 2011); SAMSUNG-1003, ¶¶63-64. Petitioner reserves the right to respond to construction(s)

[1] Confirming public availability/accessibility of TS-23.140 on/around March 2005, and similar for OMA references (SAMSUNG-1010-to-1011).

offered/adopted by Patent Owner or the Board. Petitioner is not conceding that the Challenged Claims satisfy all statutory requirements. Petitioner is not waiving any arguments regarding claim scope or grounds that cannot be raised here. Petitioner applies art herein consistent with Patent Owner's allegations of infringement in the co-pending litigation.

D. Level of Ordinary Skill in the Art

A person of ordinary skill in the art ("POSITA") relating to the '192 Patent's subject matter as of January 28, 2009 ("Critical Date") would have had (1) at least a bachelor's degree in computer science, electrical engineering, or a related field, and (2) 3-5 years of experience in services and application implementation in communication networks. SAMSUNG-1003, ¶¶21-22. Additional graduate education could substitute for professional experience, and vice versa. *Id.*

II. '192 PATENT

The '192 Patent relates to "a message link server that maintains secure message links with device link agents on" user devices. SAMSUNG-1001, Abstract; SAMSUNG-1003, ¶¶41-61.

FIG. 16 (below; annotated) shows communication between service controller 122 and device 100 over service control link 1653:

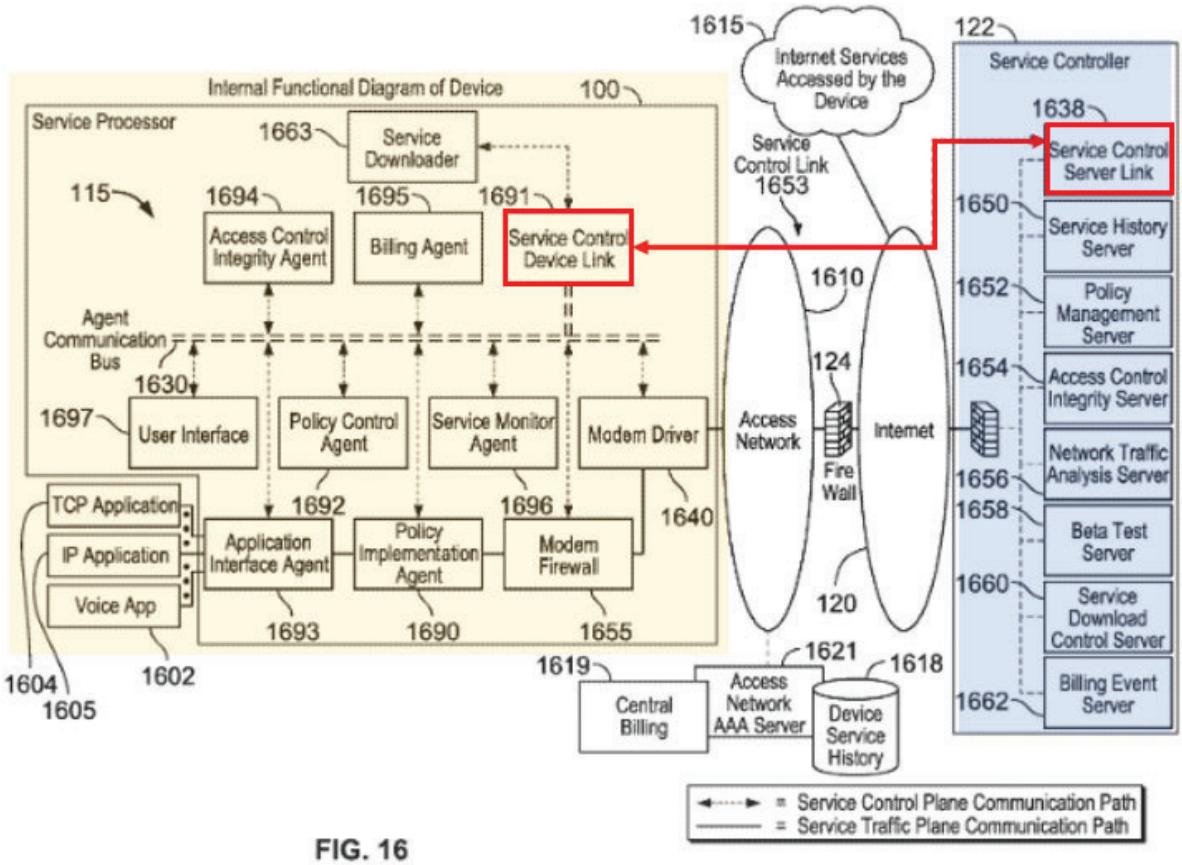


FIG. 16

Service Controller 122 includes servers (1650-1652-1654-1656-1662) coupled to service control server link 1638 that includes transport services stack 2420. SAMSUNG-1001, 68:19-20, Fig. 24, 87:49-52. Device 100 includes service control device link 1691 coupled to device agents (1697, 1695). SAMSUNG-1001, 37:34-46; 41:45-53; 45:26-30.

"Network elements send messages" to "software components on" the devices. SAMSUNG-1001, Abstract. The messages are buffered until a trigger occurs, causing message delivery to service control device link 1691, which delivers

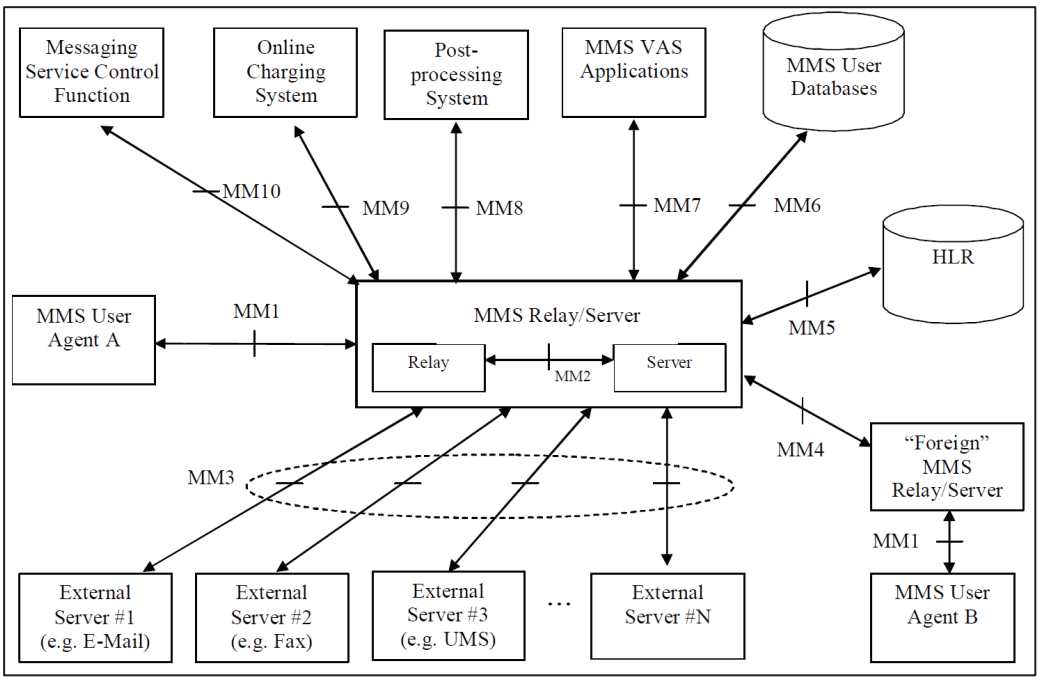
an "agent first function message" "to the appropriate agent" on the device based on an "agent first function ID." *Id.*, Abstract, 89:21-90:53, FIG. 25.

III. THE CHALLENGED CLAIMS ARE UNPATENTABLE

A. Ground 1 Claims are Rendered Obvious by TS-23.140

1. TS-23.140[2]

TS-23.140 describes "Multimedia Messaging Service, MMS," with an example MMS environment shown below:



SAMSUNG-1004, Figure 3, 10, 23[3]; SAMSUNG-1003, ¶¶65-72.

---

[2] References/combinations' descriptions are incorporated into each mapping including citations to them. Emphasis added unless otherwise indicated.

[3] Citations to TS-23.140 refer to publication page number.

In this environment, MMS Relay/Server handles storage and notification, reports, and handling of multimedia messages (MM) between, e.g., MMS User Agent, Foreign MMS Relay/Server, and VAS (Value Added Services) Applications. *Id.*, 21, 23-24. MMS can be used to send messages including "data specific to applications between two MMS User Agents or an MMS User Agent and an MMS VAS Application (or vice versa)." *Id.*, 54-55.

2. Claim Analysis

(a) *Claims 1, 15*

[1pre]/[15pre]

If the preamble is limiting, TS-23.140 renders obvious a message link server (MMS Server/Relay) and a method for operating the same. SAMSUNG-1003, ¶¶95-102. As described below, MMS Relay/Server is a message link server because it facilitates message delivery between devices/elements (user devices, other network elements) over a network:

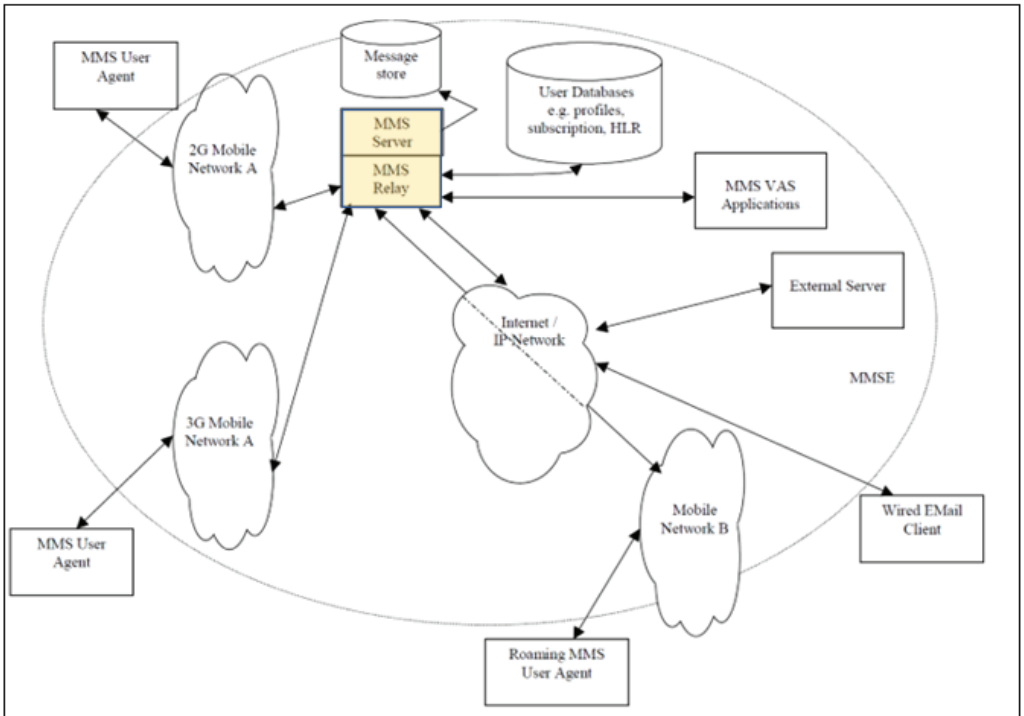


Figure 2: MMS Architectural Elements

SAMSUNG-1004, 17, Fig. 2 (annotated); SAMSUNG-1003, ¶¶95-96. MMS Relay/Server stores and handles "incoming"/"outgoing messages" and transfers messages "between different messaging systems," including between MMS User Agents/UEs, or MMS User Agents and MMS VAS Applications *Id.*, 14; SAMSUNG-1003, ¶¶97-99. MMS Relay/Server also facilitates data transport to applications between two MMS User Agents or an MMS User Agent and an MMS VAS Application. *Id.*, 54-55; SAMSUNG-1003, ¶¶100-101. Additional details regarding the MMS Relay/Server and the method for operating it are provided below with reference to [1a]/[15a]-[1d4]/[15d3]. *Id.*

[1a]/[15a]

TS-23.140 renders obvious a transport services stack (transport layer security (TLS)-based transport protocol) to maintain a respective secure message link through an Internet network between the message link server (MMS Relay/Server) and a respective device link agent (MMS User Agent) on each of multiple wireless end-user devices (mobile phones/terminals/UEs). SAMSUNG-1003, ¶¶103-116.

As shown below, MMS Relay/Server communicates via an Internet network with MMS User Agents executing on each UE/phone:

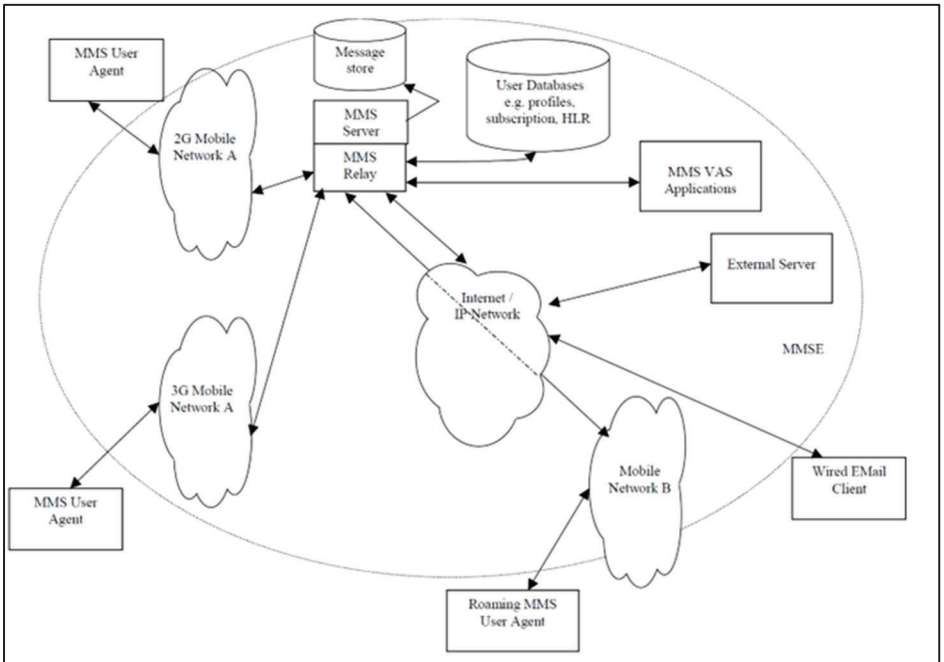
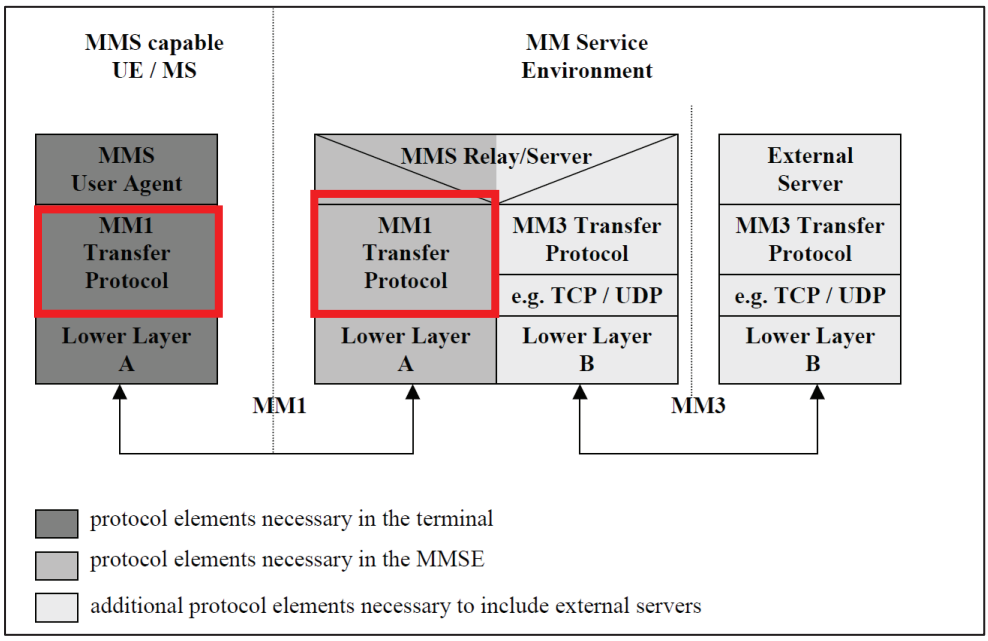


Figure 2: MMS Architectural Elements

SAMSUNG-1004, 17, FIG. 2 (environment with different network types provided using "Internet protocol" that "enables messaging in 2G and 3G wireless networks" and are "compatible with messaging systems" on "the Internet"), FIG. 2;

SAMSUNG-1003, ¶¶104-105; SAMSUNG-1010, FIG. 3. MMS User Agent is a device link agent facilitating transmission/reception of multimedia messages (MMs) with/from different devices (e.g., MMS User Agents/UEs, MMS Relay/Server, MMS VAS Applications) linked over a communication network (above). SAMSUNG-1004, 19-20; SAMSUNG-1003, ¶106.

TS-23.140 discloses or renders obvious a "transport services stack" consistent with its description in the '192 Patent. *See* SAMSUNG-1001, 89:24-41, 90:34-50); SAMSUNG-1003, ¶¶107-108 (citing SAMSUNG-1027, -1028). In TS-23.140, MM1 Transfer Protocol and associated functionality (utilizing TCP and TLS/transport layer security) provides a secure message link through an Internet Network between the message link server (MMS Relay/Server) and wireless end-user devices (UEs/terminals). SAMSUNG-1003, ¶¶109-110. As shown below, MM1 and MM3 Transfer Protocols are implemented as the "protocol framework" at the MMS Relay/Server to enable communications with the corresponding Transfer Protocols (MM1 or MM3) implemented at, e.g., another UE/External Server:



SAMSUNG-1004, 24-25, FIG. 4 (annotated); SAMSUNG-1003, ¶¶110-111. A POSITA would have understood or found obvious that the MM1 Transfer Protocol is a transport services stack that facilitates transmission/transport of network communications between the MMS Relay/Server and network elements (e.g., MMS User Agent, MMS VAS Applications). SAMSUNG-1003, ¶¶111-114; SAMSUNG-1013 (MM1 Transfer "protocol stack" is the "transport mechanism" that uses HTTP/TCP/IP).

TS-23.140 contemplates "WAP/OMA implementation" for the "MM1 Transfer Protocol" and incorporates by reference Open Mobile Alliance (OMA) specifications, e.g., SAMSUNG-1011, that explain that "a device implementing OMA MMS ***must have*** … ***WAP WSP stack or HTTP/TCP/IP stack***." *See* SAMSUNG-1004, 13, 162; SAMSUNG-1003, ¶¶112-113; SAMSUNG-1011, 11.

Moreover, OMA specifications describe the "TLS" "security protocol" as providing "secure data transmission between the MMS Client and the MMS Proxy-Relay in … HTTP based protocol stacks for MMSM implementation." SAMSUNG-1010, 22; SAMSUNG-1003, ¶¶112-113.

Given these disclosures (expressly incorporated into TS-23.140 (SAMSUNG-1004, 13, 162)) and TLS's well-known use for securing/encrypting network communications within transport stacks, a POSITA would have understood or found obvious that MMS Relay/Server's transport services stack would use TLS for securing the communication link between the MMS User Agent and the MMS Relay/Server. SAMSUNG-1003, ¶114; *see id.*, ¶¶26-31 (citing SAMSUNG-1027, -1028, -1030); SAMSUNG-1014, [0017] ("encrypt[ing] the communication channel" between the client and MMS server with "Wireless ***Transport Layer Security***").

A POSITA would have understood or found obvious that, in MMS environments, multiple MMS User Agents/terminals are in communication with the MMS Relay/Server, thereby maintaining a respective secure TLS-based link between each MMS User Agent and MMS Relay/Server. SAMSUNG-1003, ¶¶115-116; SAMSUNG-1036, 2-3, FIG. 1, 3 (showing communication links maintained between multiple devices/MMS User Agents and MMS Relay/Server); SAMSUNG-

1007, 23 (the connection between a push server and push client is "persistent managed, tested, and configured"); SAMSUNG-1017, FIG. 1, [0007]-[0008] (showing multiple data connections with multiple end-user devices).

[1b]/[15b]

TS-23.140 renders obvious that each wireless end-user device (wireless/mobile devices/UEs) includes multiple software components (applications) authorized to receive and process data (application data) from secure message link messages received via a device link agent. SAMSUNG-1003, ¶¶117-128.

TS-23.140 describes using MMS to "transport data specific to ***applications***" downloaded on a mobile phone/terminal, and such application-specific data transport occurs "between two MMS User Agents or an MMS User Agent and an MMS VAS Application." SAMSUNG-1004, 54-55. Thus, a POSITA would have understood or found obvious that each UE includes multiple software applications/components. SAMSUNG-1003, ¶118; SAMSUNG-1036, 2-4 (showing/listing multiple applications in UEs that receive application data via MMS)

Each application has to be authorized, i.e., "***need[s] to register with the appropriate MMS User Agent or MMS VAS Application***," to receive and process messages via MMS. SAMSUNG-1003, ¶119; SAMSUNG-1004, 54-55, 30 (describing application registration process). Once registered, a message including

application data for an intended application is delivered by the MMS Relay/Server to the registered/intended application (via MMS Relay/Server) upon determining that the MMS User Agent can support application data (i.e., has the capability to support application data transport) and the terminal includes the intended application. *Id.*, 30, 54-55. A POSITA would have therefore understood or found obvious that such applications are ***authorized*** to receive application data messages. *Id.*; SAMSUNG-1003, ¶¶119-121.

A POSITA would have further understood or found obvious that application data received by a particular destination application would be processed by that application for displaying data or performing operations on that data. SAMSUNG-1003, ¶122; SAMSUNG-1004, 56 ("***handling and processing***" received application data "by the destination application"); *see id.*, 54-55.

Because this communication occurs between MMS User Agents and/or between MMS User Agent and MMS VAS Application, and via MMS Relay/Server (*see* [1a]/[15a] *supra*), a POSITA would have understood that the messages including application data are received via MMS User Agent (device link agent) on the particular terminal/device. SAMSUNG-1003, ¶123; SAMSUNG-1004, 54-55 ("receiving MMS User Agent or MMS VAS Application shall … ***route the received MMS information on to the destination application***").

Because a secure message link enables communication of messages between

MMS Relay/Server and MMS User Agent (per [1a]/[15a] above), a POSITA would have understood that the messages including the application data are "secure message link messages." SAMSUNG-1003, ¶125.

A POSITA would have also found obvious that the above-described application data delivery to an application would be performed for multiple applications resident on a mobile terminal that seek to send/receive application data via MMS (i.e., from MMS Relay/Server and via MMS User Agent). SAMSUNG-1003, ¶¶125-128; SAMSUNG-1004, 54-55 (applications that "transport application specific data using MMS" reside "on an MMS User Agent…. Applications that want to transport data specific to applications other than MMS will initially need to register with the appropriate MMS User Agent"; describing different types of applications, e.g., chess and messaging applications); SAMSUNG-1036, 3-4 (explaining and showing multiple applications on an MMS-capable terminal that uses MMS to send and receive application data).

<u>[1c1]/[15c1]</u>

TS-23.140 renders obvious an interface (interfaces including MM1, MM3, MM7) to a network to receive network element messages (messages including application data and addressing data) from network elements (other MMS User Agents/UEs, MMS VAS Applications). SAMSUNG-1003, ¶¶129-135.

TS-23.140's MMS network environment (below) includes "***a collection of MMS-specific network elements***" and enables communications between these elements (MMS User Agents, MMS Server/Relay, MMS VAS Applications, and external server(s)) over various networks:

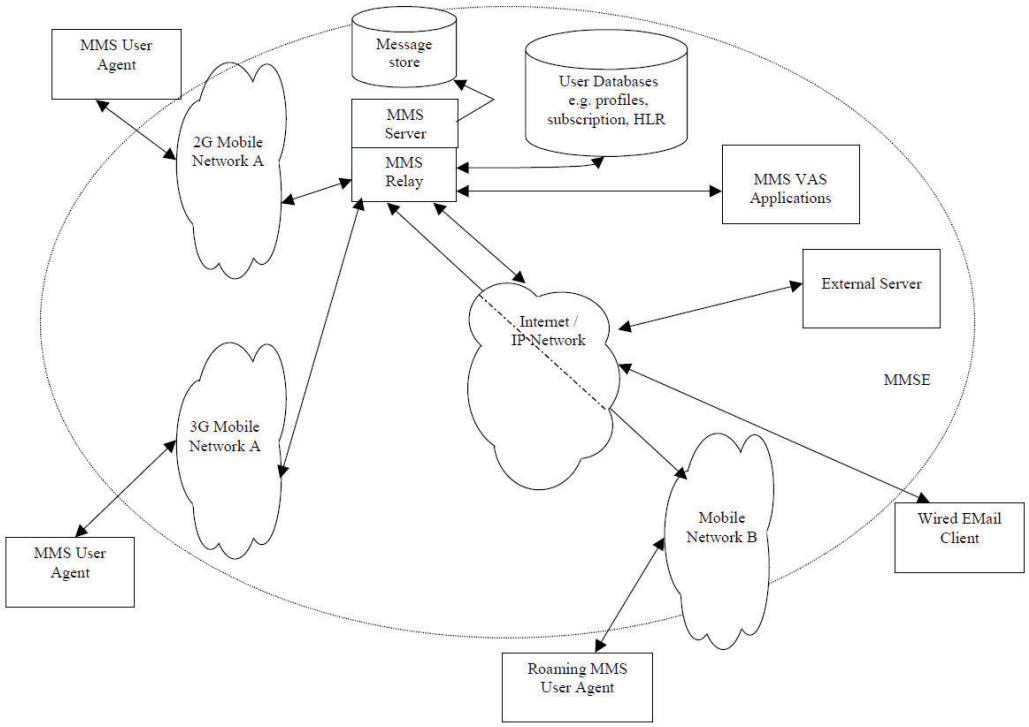
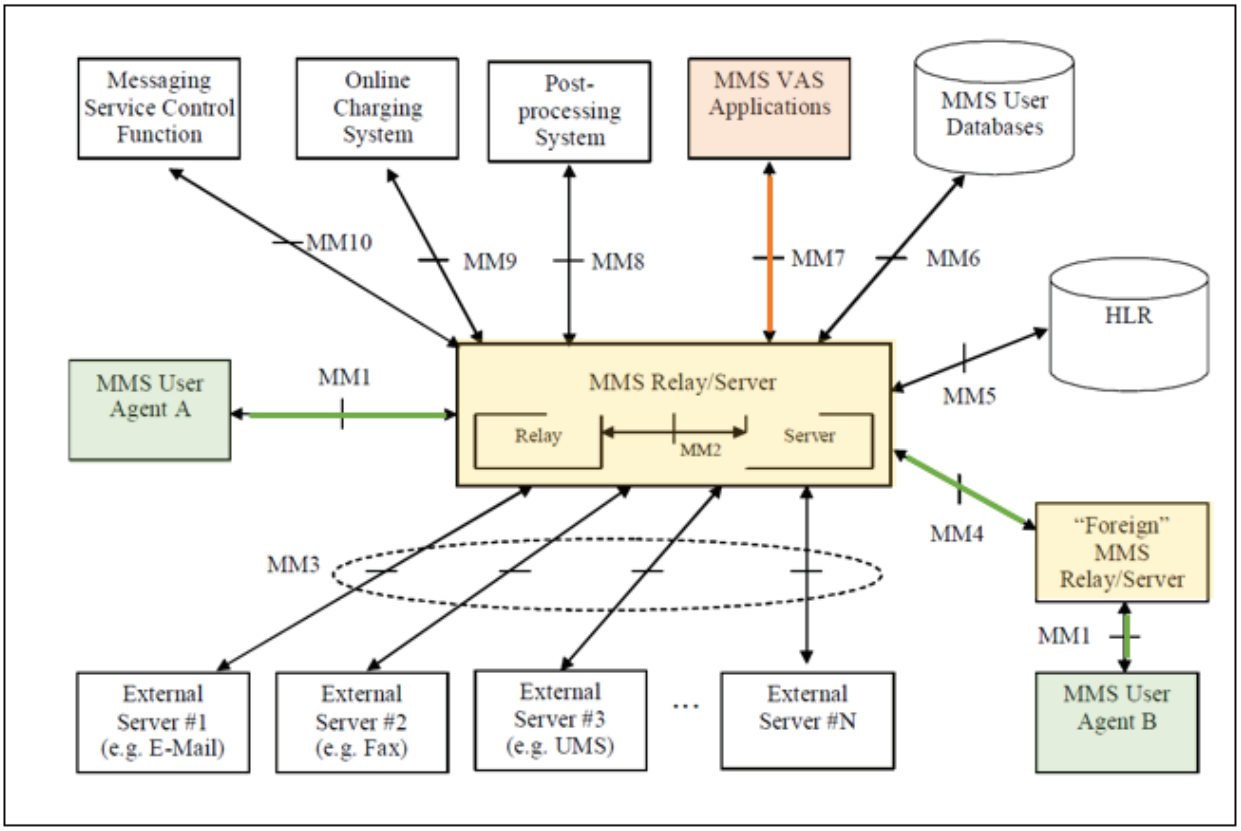


Figure 2: MMS Architectural Elements

SAMSUNG-1004, 17; SAMSUNG-1003, ¶¶129-130. As explained above ([1a]/[15a], [1b]/[15b] *supra*), MMS Relay/Server receives messages from network element(s) (i.e., network element messages), e.g., MMS User Agents and MMS VAS Applications. SAMSUNG-1003, ¶131.

Various "interfaces" facilitate network communication of messages (i.e., network element messages) between MMS User Agents, MMS VAS Applications,

and MMS Relay/Server, using MM1, MM2, MM7, and MM4 interfaces. SAMSUNG-1003, ¶132; SAMSUNG-1004, 23-24, FIG. 2 (below; annotated).



MM1 interface is the "reference point between the MMS User Agent and the MMS Relay/Server," MM4 interface is the "reference point between the MMS Relay/Server and another MMS Relay/Server," and MM7 interface is "the reference point between the MMS Relay/Server and MMS VAS Applications." SAMSUNG-1004, 23-24; SAMSUNG-1003, ¶¶132-133.

These are interfaces to various networks, including 2G/3G mobile networks, and IP/internet networks, shown in Fig. 2 above. SAMSUNG-1003, ¶134.

As explained above ([1b]/[15b] *supra*), applications registered with MMS User Agents/VAS applications transmit application data to other applications (corresponding to other MMS user agents and VAS applications) via the MMS Relay/Server. *See* SAMSUNG-1004, 54-55 (delivering data to an application via MMS User Agent, using application's identifier); SAMSUNG-1003, ¶135.

[1c2]/[15c2]

TS-23.140 renders obvious that the received network element messages (messages from applications registered with MMS User Agent/MMS VAS Applications) comprise respective message content, including data (application and control data) and identification of a respective one of the authorized software components (registered application(s)), and requests for delivery of the respective message content to respective wireless end-user devices. SAMSUNG-1003, ¶¶137-144.

As described above ([1a]/[15a], [1b]/[15b] *supra*), MMS is used to transport application data from one device (terminal, server) and its associated agent (MMS User Agent, MMS VAS Application(s)) to another device and its associated agent (MMS User Agent, MMS VAS Application(s)). SAMSUNG-1004, 54-55; SAMSUNG-1003, ¶138. This application data transmission occurs upon an application

"trigger[ing]" the MMS User Agent or MMS VAS Application to send a message—including application data and/or "control information" along with a destination "application identifier"—to a destination application and the MMS User Agent/VAS Application coordinates message transmission. SAMSUNG-1004, 54-56 ("MMS User Agent … route[s] the received MMS information on to the destination application … referred to from the destination application identifier (based on the negotiated details upon application registration process)"), 14; SAMSUNG-1003, ¶¶139-140. MMS Relay/Server receives this message and passes "***on the destination application identifier***" and "application data" to MMS User Agent. *See id*.; *see also* [1c1] *supra*.

The recipient/destination application would be an authorized/registered application (*see* [1b] *supra*) because applications intending to send/receive application data "need to register with the appropriate MMS User Agent" using their "application identification value" and then, upon message receipt, the recipient application would be identified by MMS User Agent using its identification value as being resident on the device before the message is transmitted to this application. *See* SAMSUNG-1004, 55-56; SAMSUNG-1003, ¶¶141-143.

Because messages are sent by an originating MMS User Agent to the MMS Relay/Server for delivery to an MMS User Agent with which the destination application is registered, a POSITA would have understood or found obvious that the

transmitted/received message comprises a request for delivery of the respective message content to a respective wireless end-user device (including the destination application(s) resident on that device). SAMSUNG-1003, ¶¶141-143 (explaining that MMS Relay/Server interfaces with multiple devices and their respective MMS User Agents, to facilitate message transmission to multiple MMS User Agents/devices for delivery to their respective applications); SAMSUNG-1004, 54-55 (application data capability assessment and confirmation of application being resident on device prior to sending to application), FIG. 6 (showing message exchange between MMS User Agents via MMS Server/Relay;); *id.*, 28-29; SAMSUNG-1036, 2-4.

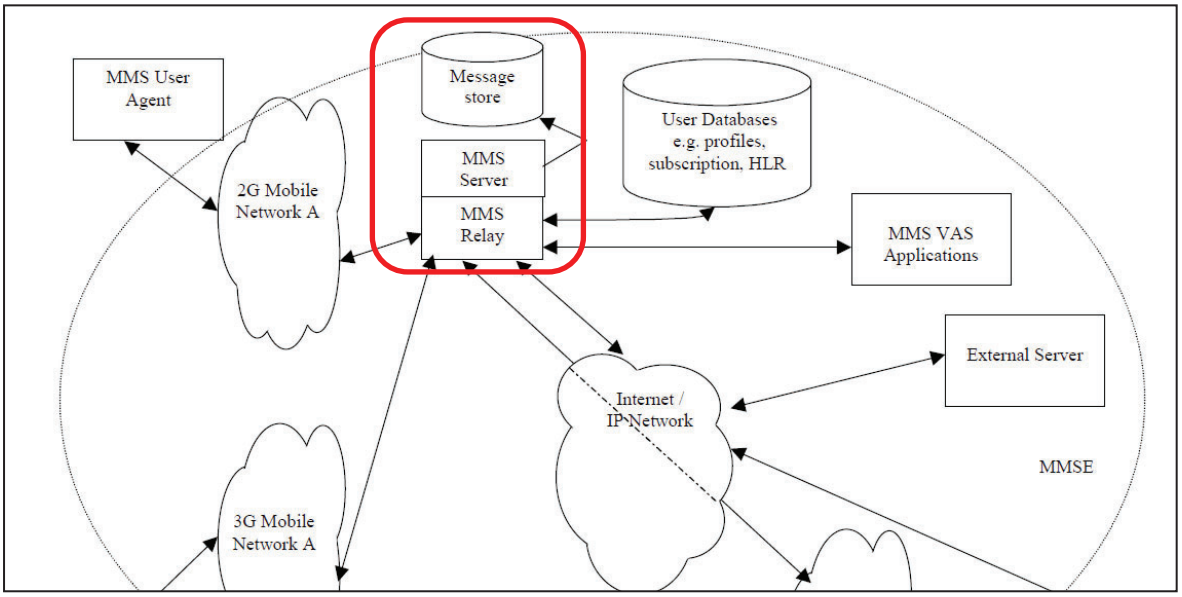
[1d1]-[1d2]/[15d1]

TS-23.140 renders obvious a message buffer system including a memory that buffers content from received network element messages (message store storing the received message), for which delivery is requested to a given one of the wireless end-user devices (application running on a terminal/UE associated with the recipient MMS User Agent). SAMSUNG-1003, ¶¶145-152.

As described above ([1c1]-[1c2] *supra*), network element messages are received from network element(s) by the MMS Relay/Server, and delivery of these

messages is requested to application(s) resident on a terminal/UE (and its associated MMS User Agent). SAMSUNG-1003, ¶146.

MMS Relay/Server "stores and handl[es]" "***incoming and outgoing messages***" and handles message transfer "between different messaging systems":



SAMSUNG-1004, 17, Fig. 2 (annotated; "message store" coupled to MMS Server/Relay), 21 (describing "temporary storage of messages"; facilitating application data transport); SAMSUNG-1003, ¶¶147-149.

Upon receiving a message, the originator MMS Relay Server "***retain[s] the MM until the earliest desired time of delivery***" (SAMSUNG-1004, 26-28) and the recipient MMS Relay/Server (which can be the same as the originator server)" "***store[s] the MM at least until***" "the associated time of expiry is reached, the MM

is delivered, or the recipient MMS User Agent requests the MM to be routed forward or the MM is rejected." *Id.*

Messages can be "persistent[ly] stored" in a "Persistent Network-Based Storage" (MMBox) associated with "the MMS Relay/Server." SAMSUNG-1004, 21-22, 26-28; SAMSUNG-1003, ¶¶150-151.

Additionally, as described below ([1d3]-[1d4], *infra*), MMS Relay/Server includes a memory buffer system including logic for delivering messages upon one or more message delivery triggers occurring. SAMSUNG-1003, ¶¶151-152.

<u>[1d3]/[15d2]</u>

TS-23.140 renders obvious multiple message delivery triggers for transmitting the stored message data to a MMS User Agent for a UE that includes the destination application. SAMSUNG-1003, ¶¶153-160. For example, the message is not delivered by the MMS Relay/Server to a MMS User Agent/UE-terminal (resident on a UE/terminal) until one or more of the following triggers occur (SAMSUNG-1003, ¶¶155-156):

- The MMS Relay/Server has sent a notification to the recipient User Agent (SAMSUNG-1004, §7.1.2 (upon MM receipt, ***MMS Relay/Server*** "***generate[s] a notification to the recipient MMS User Agent***")).

- The recipient MMS User Agent requests message retrieval upon MMS Relay/Server receiving the notification—e.g., within a message expiry period. *See* SAMSUNG-1004, §7.1.2.1 ("In a response to the notification the MMS User Agent … retrieve[s] the MM," either automatically or manually, where the "retrieval mode" is based on user settings or the recommendation in the MM notification).
- The specified deferred delivery period (e.g., message expiry period) is met (e.g., where the ***recipient MMS User Agent requests deferred message delivery in the message retrieve request***) or until message is retrieved/rejected. *See* SAMSUNG-1004, §7.1.2.1.
- When recipient MMS User Agent becomes available/reachable (e.g., moves into coverage, switches MMS User Agent on) or until message expires. *See* SAMSUNG-1004, §7.1.3.
- The message conforms to the message retrieval request's "size restriction." *See id.*

Because MMS Relay/Server does not deliver the message to the recipient MMS User Agent (for a particular user terminal) until the above condition(s)/trigger(s) are met, a POSITA would have understood or found obvious that the MMS Relay/Server includes logic that is configured to determine when one (or more) of

these message delivery triggers for the particular terminal/end-user device has occurred and if so, delivering the messages. SAMSUNG-1003, ¶156.

Given the above-described message delivery triggers, a POSITA would have found obvious that message receipt alone would not trigger message delivery, particularly considering the other condition(s)/trigger(s) that would be implemented (per above) and would be satisfied before message(s) is/are delivered. SAMSUNG-1003, ¶157.

Moreover, a POSITA would have understood that MMS User Agent can request delivery of the message received/stored by the MMS Relay/Server, and a POSITA would have found obvious (per the above disclosures) for such requested delivery to be based on user request—which, per the '192 specification (*see* SAMSUNG-1001, 38:50-63), constitutes a message delivery trigger that is "an asynchronous event with time-critical messaging needs." *See* SAMSUNG-1004, 28-29 (MMS User Agent retrieves message[s] "***either manually or automatically***"); 69 (recipient MMS User Agent requests message retrieval); 20 (in "manual mode," "***user*** is made aware of the MM notification and ***… make[s] a decision whether to download the MM***"); SAMSUNG-1003, ¶¶158-160.

[1d4]/[15d3]

TS-23.140 renders obvious that, upon determining that message delivery

trigger(s) has/have occurred, the message buffer system includes logic for supplying one or more messages comprising the buffered content (stored message in the message store or MMSBox) to the transport services stack for delivery on the secure message link maintained between the transport services stack and a device link agent (MMS User Agent) on the given one of the wireless end-user devices. SAMSUNG-1003, ¶¶161-165.

As described above ([1d1]-[1d3]/[15d1]-[15d2], *supra*), MMS Relay/Server delivers stored messages (in MMBox or another temporary storage) upon the occurrence of one or more of the delivery triggers. SAMSUNG-1003, ¶163-164; SAMSUNG-1004, 28-31. A POSITA would have found obvious that MMS Relay/Server includes logic for performing such message delivery upon detecting occurrence of one or more of the delivery triggers ([1d3]/[15d2] *supra*). *Id.*

As described above ([1a] *supra*), TS-23.140 discloses/renders obvious message delivery via MMS Relay/Server's transport services stack (MM1 Transfer Protocol) that maintains a secure message link (TLS-based link) to a device link agent (recipient MMS user agent) of a particular wireless end-user device (UE/terminal of recipient MMS User Agent). *See* SAMSUNG-1004, 24, Fig. 4; SAMSUNG-1010, 22 (describing TLS-based HTTP-protocol link for such data transmission); SAMSUNG-1003, ¶¶165-166.

(b) *Claim 5*

[5a]

TS-23.140 renders obvious that the transport services stack receives, over each respective secure message link, upload messages forwarded by the respective device link agents (MMS User Agents) from a subset of the device software components (application(s)). SAMSUNG-1003, ¶¶168-174.

TS-23.140 discloses that an application executing on a user device/UE (device software component) forwards a message including application data (upload message) to the UE's MMS User Agent (device link agent), which submits the message to the MMS Relay/Server. SAMSUNG-1004, 54-55 (sending MMS including application data to a recipient application on another device/MMS User Agent), 14; SAMSUNG-1003, ¶¶169-171 (explaining that an application receiving application data using MMS can send application data to an application on another device).

The MMS Relay/Server "transports" "application data" between an MMS User Agent and another MMS User Agent/VAS Application. SAMSUNG-1004, 19-21, 35, 55; SAMSUNG-1003, ¶172. As described above ([1a]/[15a] *supra*), MMS Relay/Server's MM1 Transfer Protocol receives/transports application data to/from the MMS-capable UE/MS to the MMS Relay/Server. SAMSUNG-1004, 24, Fig. 4. A POSITA would have therefore understood or found obvious that

MMS Relay/Server's transport services stack receives messages including application data from an application via a particular MMS User Agent. SAMSUNG-1003, ¶172.

Moreover, in the MMS Environment (FIGS. 1-2 above), a POSITA would have understood that multiple devices/terminals use an MMS Relay/Server for transmitting data to multiple devices/servers/network elements. SAMSUNG-1003, ¶173; SAMSUNG-1036, 2-3, FIG. 1, 3; SAMSUNG-1007, 23; SAMSUNG-1017, FIG. 1, [0007]-[0008]. A POSITA would have therefore found obvious for MMS Relay/Server to receive upload messages (including application data) from multiple MMS User Agents and their respective applications requesting message delivery, and such communications would happen over dedicated TLS-based secure links over the MM1 Transfer Protocol between each MMS User Agent and the MMS Relay/Server (*see* [1a]-[1b]/[15a]-[15b] *supra*). SAMSUNG-1003, ¶174.

<u>[5b]</u>

TS-23.140 renders obvious that the upload messages identify a corresponding network element (recipient device/application) to which the device respective software component (originating application) has requested delivery. SAMSUNG-1003, ¶¶175-180.

As described above ([5a] *supra*), upload messages from the originating

MMS Agent (and associated device) includes application data and the "application identifier of the destination application." SAMSUNG-1004, 54-55. Because the message includes an identifier of a destination application resident on a particular device/terminal, a POSITA would have understood or found obvious that each upload message identifies the corresponding device/terminal (network element). SAMSUNG-1003, ¶¶175-176.

Additionally/alternatively, as with sending MMs/messages, including those including application data, the sender/originating MMS User Agent "indicate[s]" message recipient's address. *See* SAMSUNG-1004, 26, 90 190 (including "Recipient address" as included in messages sent over MM1); SAMSUNG-1003, ¶¶177-178. Because messages are transmitted from a MMS User Agent to a recipient MMS User Agent (or MMS VAS Application) (on another device/server), a POSITA would have found obvious for the message to include an identifier/address for the device/server to which the message is being transmitted. *Id.*

A POSITA would recognize that routing messages to particular devices would be facilitated by including some device identification to facilitate that routing over a network. SAMSUNG-1003, ¶¶178-179. Indeed, TS-23.140 corroborates that messages sent to devices include "a user's address, a user's terminal address, or a short code." SAMSUNG-1004, 57 (describing formats of addresses).

Thus, given TS-23.140's disclosures and well-known device addressing aspects for network communications, a POSITA would have found obvious to include such addressing in the upload message for transmitting data between network devices. SAMSUNG-1003, ¶180.

[5c]

As described above ([1a]-[1c] *supra*), MMS Relay/Server communicates with network elements, e.g., recipient MMS User Agents resident on UEs/terminals, using MM1 and MM4 interfaces.[4] SAMSUNG-1003, ¶¶181-182.

(c) *Claim 6*

As described above ([1c1]-[1c2], [1d3]-[1d4]), an originating MMS User Agent/VAS Application sends, via the MMS Relay/Server, a message with an identifier-data pair, including a destination application's identifier and application data. SAMSUNG-1003, ¶¶183-189.

A POSITA would have found obvious that this message would be received,

[4] Claim 5's "the network server system" lacks antecedent basis. The analysis here considers claim 1's message link server and its network interface.

by MMS Relay/Server's transport services stack and from different network elements (MMS User Agents, VAS Applications). SAMSUNG-1003, ¶¶184; *see* SAMSUNG-1004, 54-56.

As described above ([1c1]-[1c2] *supra*), a POSITA would have understood/found obvious that multiple applications on a terminal/MMS User Agent would receive application data via MMS (and the MMS Relay/Server). SAMSUNG-1004, 54-56; SAMSUNG-1036, 3-4; SAMSUNG-1003, ¶184. Moreover, a POSITA would have recognized that consolidating application data and corresponding application identifiers in a message would have been straightforward and readily implementable, and would achieve network efficiencies from using a single communication/message to transmit data/identifiers for multiple applications. SAMSUNG-1003, ¶185.

Indeed, TS-23.140 discloses triggering message delivery when the MMS User Agent becomes available/reachable (SAMSUNG-1004, §§7.1.2.1-7.1.3) and a POSITA would have found obvious, from a network efficiency standpoint, to implement delivery of queued/buffered message content intended for multiple applications to be transmitted to the device/MMS User Agent in a single message/communication including data and application identifier for each application (multiple identifier/data pairs). SAMSUNG-1003, ¶¶186-189.

(d) *Claim 7*

As described above ([1a] *supra*), TLS is used for secure message communication between MMS User Agent and MMS Relay/Server. SAMSUNG-1003, ¶¶190-194. A POSITA would have found obvious that the TLS-based communication link would be initiated by terminals/UE. SAMSUNG-1003, ¶190. Before the Critical Date, it was well known for TLS-based client-server communications to be initiated by the client device. SAMSUNG-1003, ¶191; SAMSUNG-1012, 29:31-30:24.

Because MMS User Agent sends messages to the MMS Relay/Server via a TLS-based connection and TLS communications are initiated by a terminal/UE/client device, a POSITA would have found obvious that the MMS User Agent (device messaging agent on end-user devices) initiates the respective secure Internet Data message link to MMS Relay/Server's transport services stack. SAMSUNG-1003, ¶¶192-194.

(e) *Claim 9*

TS-23.140 discloses that the originating MMS User Agent can set the MM's desired delivery time, such that the MMS Relay/Server stores the message until the ***earliest desired time of delivery***, before delivering the message. SAMSUNG-1004, 27; SAMSUNG-1003, ¶¶195-196. A POSITA would have understood or

found obvious that specifying the desired delivery time triggers a periodic timer upon message receipt that expires when the desired delivery time is reached, triggering message delivery. SAMSUNG-1003, ¶197. This timer is periodic because whenever a message is received, the timer is triggered and continues for a particular period—the expiration of which triggers message delivery. *Id.*

Additionally/alternatively, MMS User Agent uses "periodic polling" to retrieve messages on/from an external server—providing another example of a message delivery trigger upon expiration of a periodic timer. *See* SAMSUNG-1004, 90-91; 14; SAMSUNG-1003, ¶198. A POSITA would have understood that the above-described periodic polling is akin to a periodic timer expiring because polling happens at periodic intervals, and this periodic polling is a message delivery trigger because, in response to such polling, messages from the respective servers would be provided/delivered to the MMS User Agent via the MMS Relay/Server. SAMSUNG-1003, ¶199.

(f) *Claims 11-12*

As described above ([1d2] *supra*), one of the message delivery triggers is "the receipt of a transmission on the respective secure message link from the device link agent of the given one of the wireless end-user devices," where the transmission is "a request received from the given device link agent." SAMSUNG-

1003, ¶201. MMS User Agent sends a retrieval request to the MMS Relay/Server, which triggers MMS Relay/Server's message delivery to the MMS User Agent. *See* SAMSUNG-1004, §7.1.2.1 (MMS Relay/Server sends "a notification" to the recipient MMS User Agent and MMS User Agent then "retriev[es]" the message at the user's request); SAMSUNG-1003, ¶202.

(g) *Claim 13*

TS-23.140 discloses that, if a message is received without an "earliest desired time of delivery" indicator or if the originator MMS Relay/Server does not support this feature, the message is "immediately routed forward" to the intended MMS User Agent. *See* SAMSUNG-1004, 27, 32; SAMSUNG-1003, ¶¶204-205.

Additionally/alternatively, for messages received by the MMS Relay/Server and intended for a recipient MMS User Agent, "[i]f the MMS Relay/Server finds from the recipient MMS User Agent's capability indication … that the recipient MMS User Agent supports transport of application data, the MMS Relay/Server shall not perform any type of content adaptation to a multimedia message (MM) that may be contained in the" message payload—which includes "the destination application identifier"—and passes this message to the MMS User Agent unaltered. *See id.*, 55-56; SAMSUNG-1003, ¶¶206-207.

Therefore, the above-described message delivery triggers include receipt of

certain types of messages (particular network element messages) from network elements (originating MMS User Agents operating of UEs/client devices). SAMSUNG-1003, ¶¶204-208.

B. Ground 2 Claims are Rendered Obvious by TS123.140 And Shen

1. Shen

Like TS-23.140, Shen provides a "MMS system," as shown below, that provides "an end-to-end security solution for MMS applications." SAMSUNG-1014, [0001], [0021]; SAMSUNG-1003, ¶¶73-78.

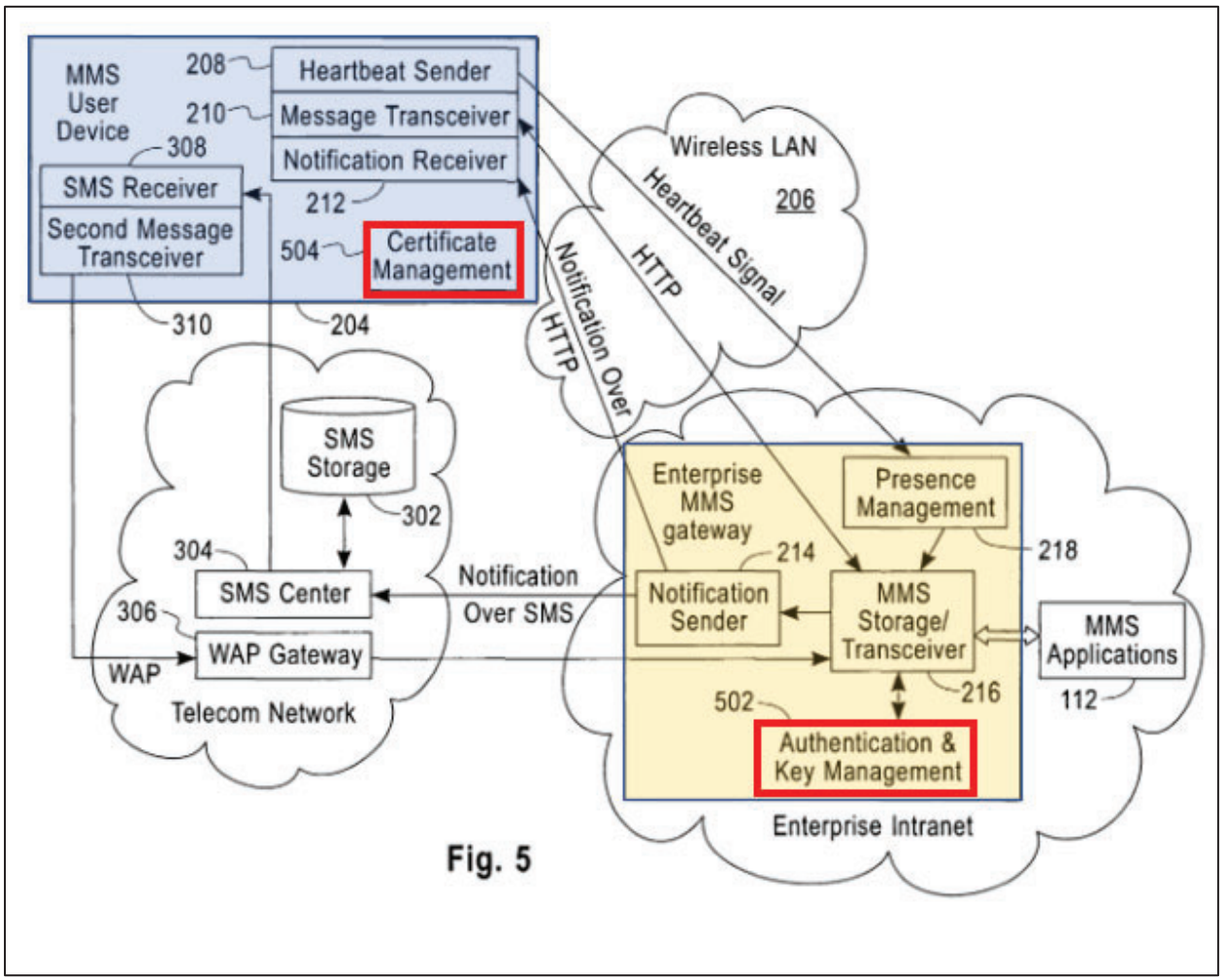


Fig. 5

SAMSUNG-1014, FIGS. 1, 5, [0001], [0004], [0021], [0030]. Shen's "authentication and key management module 502 is added in the MMS gateway to distribute symmetric keys to users and a certificate management module 504 is added in the user device." SAMSUNG-1014, [0054]. "MMS gateway and the user device deploy the same symmetric cipher to encode and decode MMS messages." *Id.*, [0054]-[0060].

2. Combination of TS-23.140 and Shen

TS-23.140 and Shen disclose a MMS system where MMS Relay/Server receives/stores messages before transmitting them to MMS User Agents. SAMSUNG-1004, 16-18, 54-56; SAMSUNG-1014, [0029]-[0034], FIGS. 1-2, 5; SAMSUNG-1003, ¶209.

Per Shen, MMS Relay/Server's store and forward functionality exposes a "security problem" given that it is "not an end-to-end solution." SAMSUNG-1014, [0004]; SAMSUNG-1003, ¶210. To overcome this "problem," Shen provides "an authentication and key management module 502" "in the MMS gateway to distribute symmetric keys to users and a certificate management module 504 is added in the user device," as shown below:

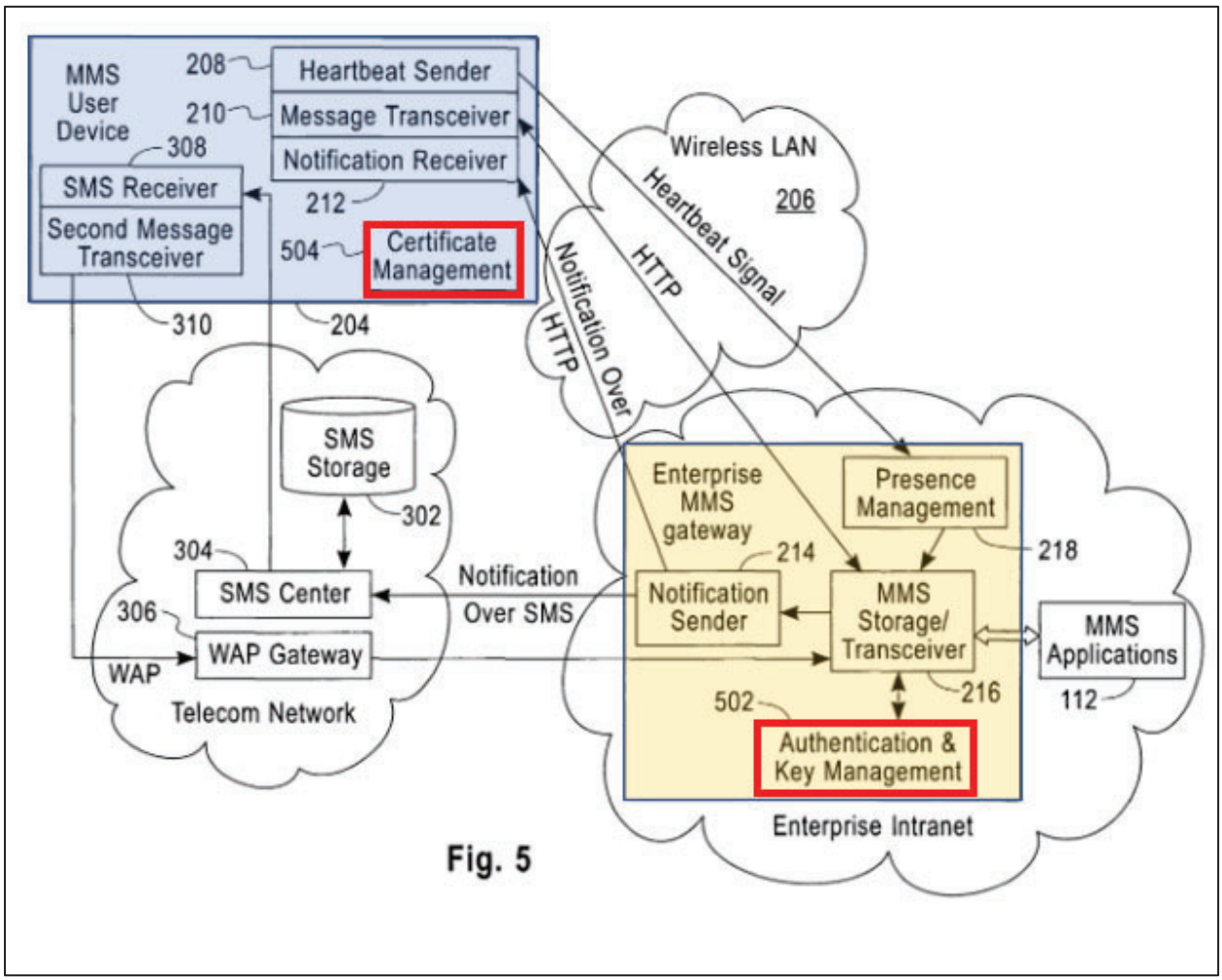


Fig. 5

Using these modules, a symmetric key is generated that MMS Relay/Server uses to encrypt data communications transmitted to the MMS User Device/Agent. SAMSUNG-1014, [0056]-[0060]; SAMSUNG-1003, ¶211; *see* §II.B.1 *supra.*

TS-23.140 and Shen provide similar MMS architectures, and given the "security problems" in MMS environments, a POSITA would have been motivated to implement Shen's known techniques to solve these problems—implementing "authentication and key management module 502" at the MMS Relay/Server that generates and "distribute[s] symmetric keys to users" and uses the symmetric keys to

encrypt data transmitted to "user device[s]." SAMSUNG-1014, [0055]-[0060]; SAMSUNG-1003, ¶¶213-214. A POSITA would have been motivated to do so to achieve an MMS system "having improved security," including "providing an end-to-end security solution for MMS applications," which is particularly beneficial for "enterprise applications." SAMSUNG-1014, [0017], [0021] (additional advantages); SAMSUNG-1003, ¶214; *see* SAMSUNG-1009, ¶¶[0054]-[0060].

Configuring TS-23.140's system to implement Shen's above-described modules and associated encryption key generation functionality for encrypting messages would have amounted to using a known technique to improve similar devices in the same way, and combining prior art elements according to known methods to yield predictable results with a reasonable expectation of success. SAMSUNG-1003, ¶215. Because TS-23.140 describes message delivery in a similar MMS environment (per Shen) involving a similar communication between MMS User Agents/Devices and the MMS Relay/Server, a POSITA would have found straightforward to modify and/or program the MMS Relay/Server (and the MMS User Agent) to implement above-described modules and associated functionality (per Shen) for generating an encryption key and encrypting communications between the MMS User Agent and the MMS Relay/Server using that key. SAMSUNG-1003, ¶216.

Additionally, the resulting system (TS-23.140-Shen) would include components performing functions prior to combination—TS-23.140's MMS system would facilitate message exchange between MMS Relay/Server and MMS User Agents, and Shen's teachings (in combination) would facilitate encrypting/decrypting such messages. SAMSUNG-1003, ¶¶217-218.

3. Claim Analysis

(a) <u>*Claim 2*</u>

As described above ([1a], [1d4] *supra*), messages would be supplied to the MMS Relay/Server's transport services stack for delivery on the secure message link between MMS Relay/Server and a device/terminal's MMS User Agent. SAMSUNG-1003, ¶¶219-220.

Additionally, as described above (§III.B.1-2 *supra*), TS-23.140-Shen would have implemented Shen's teachings of authentication and key management module at the MMS Relay/Server, thereby facilitated generating a shared key for "encrypting MMS messages." SAMSUNG-1014, [0059]; SAMSUNG-1003, ¶¶221-223. Therefore, TS-23.140-Shen would have implemented an encryption function that encrypts the messages sent by the MMS Relay/Server to the MMS User/Agent using the transport services stack and over the secure link, as described above. *Id.*

(b) <u>*Claim 3*</u>

As described above (claim 2 *supra*), TS-23.140-Shen renders obvious encrypting messages and transporting the encrypted messages to a particular MMS User Agent on a UE/terminal. SAMSUNG-1003, ¶¶224-225. Additionally, TS-23.140-Shen renders obvious encryption on the transport services stack based on the use of TLS on the transport/communication link between the MMS Relay/Server and MMS User Agent (*see* [1a] *supra*). SAMSUNG-1003, ¶226.

C. Ground 3: Claim 4 is Rendered Obvious by TS-23.140 and Ellison

1. Ellison

Ellison discloses techniques for "protect[ing] a subset of a software environment." SAMSUNG-1019, Abstract; SAMSUNG-1003, ¶¶228-229, 94. This includes multiple "operating system nub key[s] (OSNK)" "unique to an operating system (OS) nub." *Id*. A "usage protector" uses the OSNK to "protect usage of" a software environment's subset. *Id*.

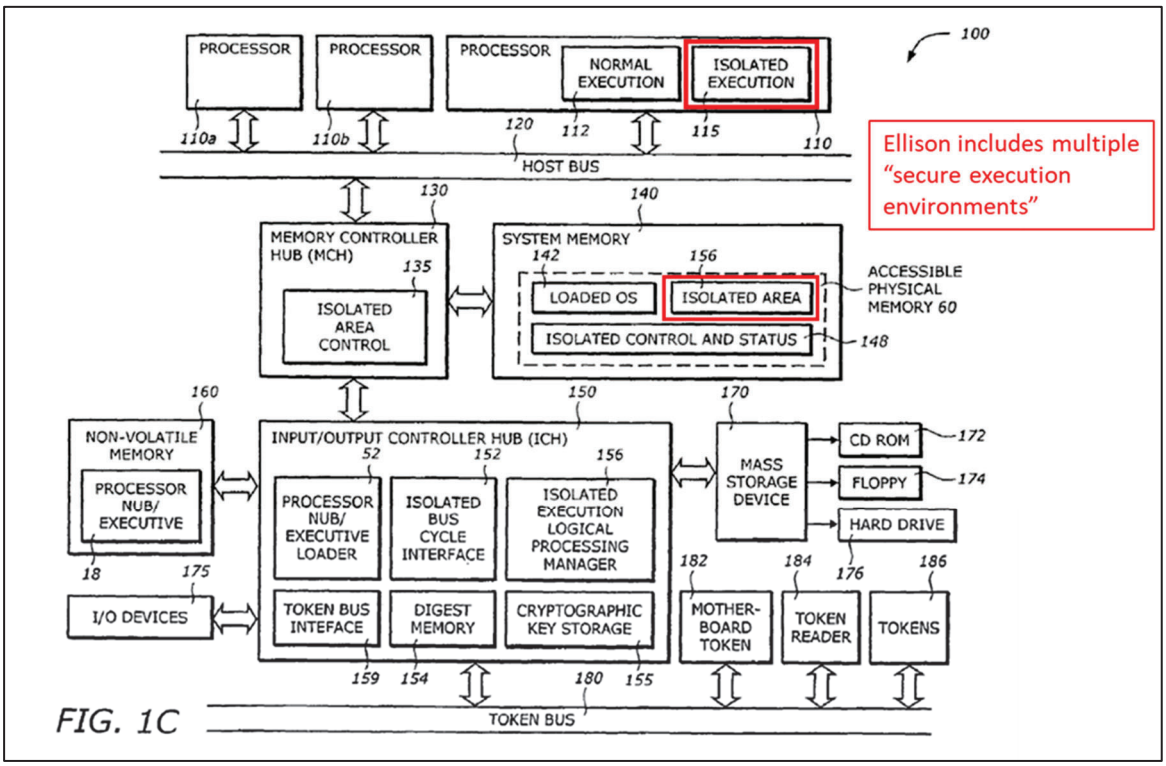
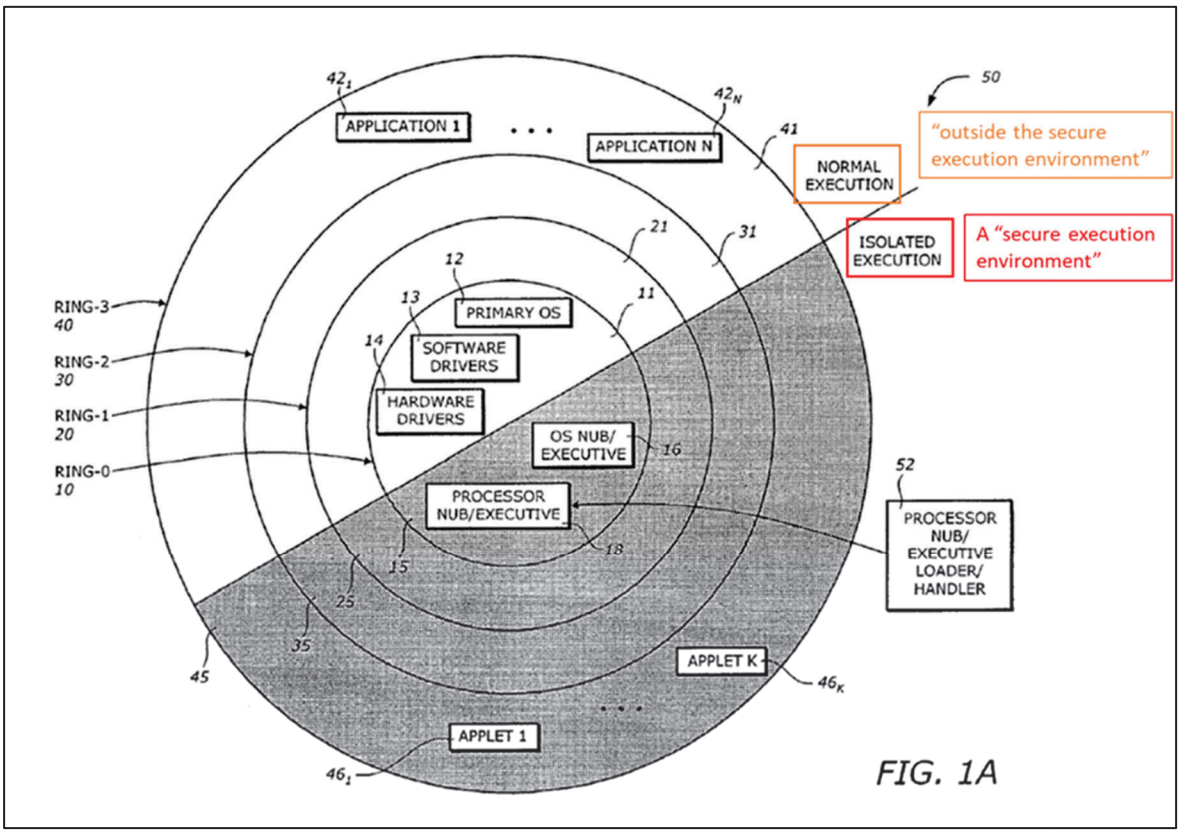


FIG. 1C

SAMSUNG-1019, FIG. 1C.

2. *Analysis*

Ellison discloses an "isolated execution mode" where access "is restricted" and a "normal execution mode" that "operates in a non-secure/normal environment" without isolated execution mode's security features. SAMSUNG-1019, 4-65-5:1, 6:1-26, 8:25-32, FIGS. 1A-1C; SAMSUNG-1003, ¶¶239-241. Ellison enables "normal" execution for applications "***outside***" the device's "***secure execution environment***." *Id.*

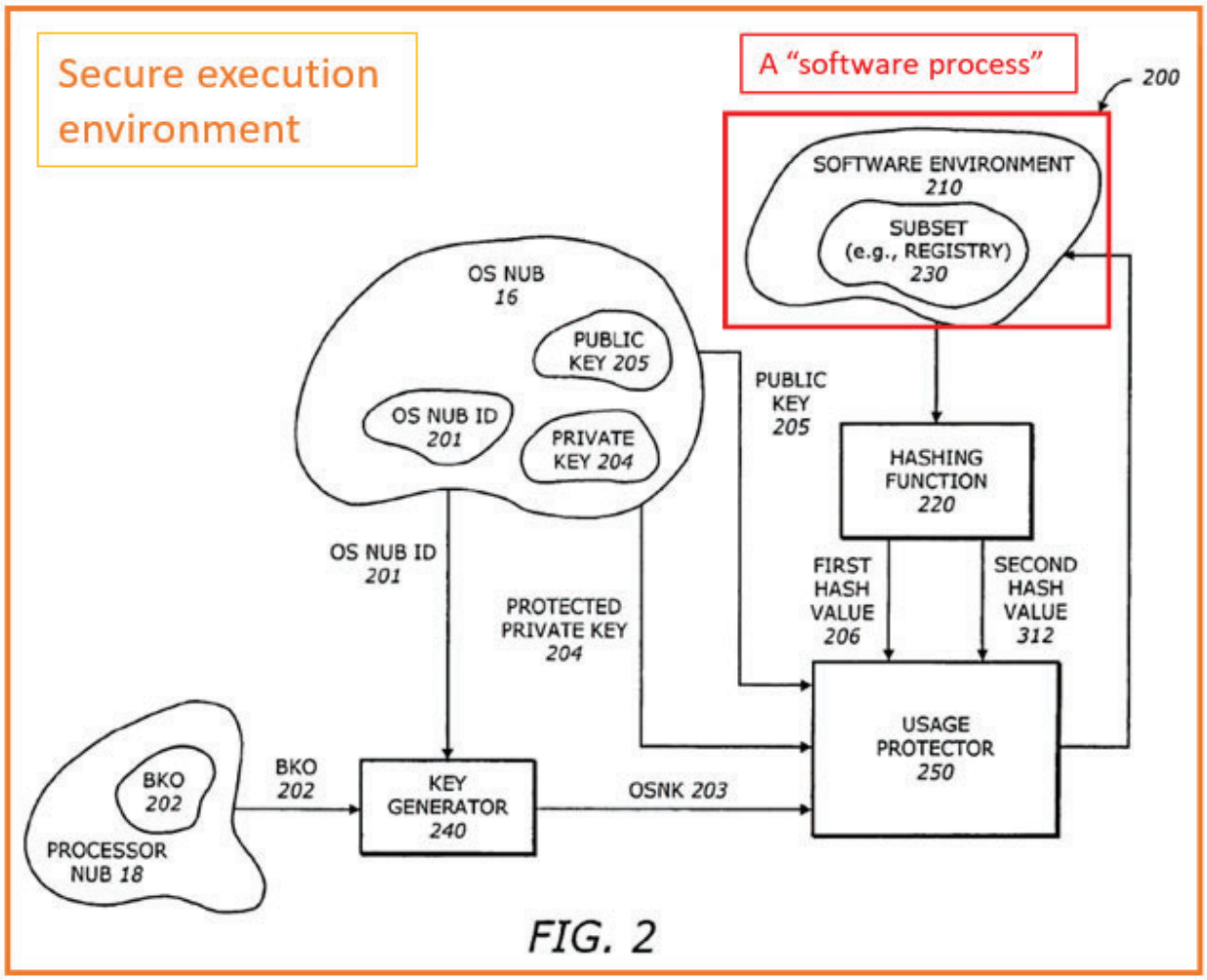


SAMSUNG-1019, FIG. 1A.

In Ellison, "isolated area 70" is the memory area that is defined by processor 110 "when operating in isolated execution mode." SAMSUNG-1019, 6:1-26, 8:25-32, FIGS. 1A-1C, 2. This technique, shown in Figure 2, is referred to as a "secure platform." SAMSUNG-1019, 6:1-26, 8:25-9:62, FIGS. 1A-1C, 2; SAMSUNG-1003, ¶241.

The secure platform includes a "key generator 240" that "generates a key operating system nub key (OSNK) 203" supplied to "trusted agents," e.g., "usage protector 250" that uses OSNK 203 to protect subset 230's usage. *Id.*, 8:66-9:40,

FIG. 2; SAMSUNG-1003, ¶242. Usage protector 250 uses a hashing function with subset 230's ONSK 203 to determine if it has been altered (e.g., reads/writes), thereby protecting against "unauthorized reads" and detecting "intrusion, tampering or unauthorized modification." *Id*., 9:47-62.



SAMSUNG-1019, FIG. 2.

In implementing the TS-23.140-Ellison combination, Ellison's above-described teachings of a usage protector and secure platform, which operate within the secure, isolated area (secure execution environment) would be incorporated into TS-23.140's MMS User Agent executing on UEs/terminals, thereby providing secure access to applications outside the secure execution environment. SAMSUNG-1003, ¶¶243. A POSITA would have been motivated to do so because implementing Ellison's secure platform would have improved device security in the MMS system (SAMSUNG-1019, 8:25-9:62, FIG. 2; SAMSUNG-1003, ¶¶244-245), which would be an efficient, beneficial option for TS-23.140's system that contemplates secure/encrypted messaging. *Id.*; SAMSUNG-1003, ¶¶230-233, ¶¶33-40 (citing SAMSUNG-1030, -1031, -1032, -1033). Therefore, the combination would enable more secure message transmission, and prevent bad or malicious actions (malware, unauthorized access). *Id.*; SAMSUNG-1019, 8:25-9:62, FIG. 2; SAMSUNG-1003, ¶¶245, 235-238.

Combining TS-23.140 and Ellison's teachings would have involved combining prior art elements according to known methods to yield predictable results. SAMSUNG-1003, ¶¶247-248, 235-238. Incorporating Ellison's security techniques into TS-23.140's mobile terminals would have been predictable and foreseeable with a reasonable expectation of success because Ellison describes that its techniques can be implemented in "computer system[s]" (e.g., TS-23.140's UEs)

that would include a "processor." SAMSUNG-1019, 5:11-16; SAMSUNG-1003, ¶¶245, 235-238.

Moreover, it was well known for operating systems to include the messaging services and clients/agents responsible for providing those services, within the trusted/secure computing environment, and for applications (to which messages are directed) to be tiered/organized outside this environment. SAMSUNG-1003, ¶¶235-238, 245-247 (citing SAMSUNG-1035). Given that TS-23.140 contemplates mobile terminals with an MMS User Agent that communicates with applications on the device, the well-known need to protect computing environments (including message providing services) from other applications/components of the environment, and Ellison's disclosures confirming the same, a POSITA would have been motivated to implement Ellison's teaching of secure execution environment that enable such improved security within the MMS environment, such that the MMS User Agent is organized as part of the trusted computing environment (secure execution environment) and enables communications with applications residing outside this trusted/secure environment. SAMSUNG-1003, ¶¶245-248, 235-238.

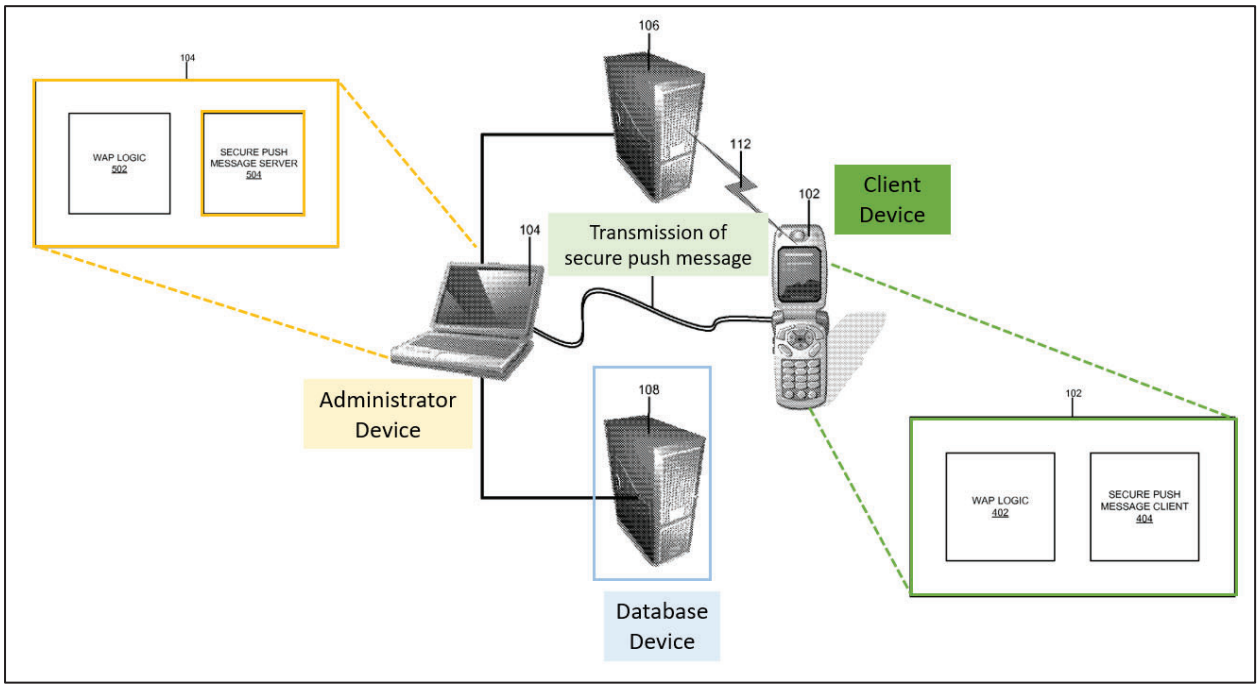
Therefore, TS-23.140-Ellison renders obvious that the MMS User Agent (device link agent) executes in a secure execution environment and one or more of

the applications (software component(s)) to which push messages are routed, execute outside this environment. SAMSUNG-1003, ¶¶230-249, 35-40.

D. Ground 4: Claim 8 Is Rendered Obvious by TS-23.140 And Rakic

1. Rakic

Rakic's administrator device 104 includes secure push message server 504 that communicates secure push messages to client device 102 (as shown below):



SAMSUNG-1015, FIG. 1, 4-5 (annotated); SAMSUNG-1003, ¶¶79-85.

The environment includes database device 108 that stores information related to client devices. SAMSUNG-1015, [0047]-[0049]. Secure push message server 504 generates a key, an electronic signature, and a secure push message for a message from a particular user/administrator. *Id.*, [0047]-[0049], [0066]-[0067].

Server 504 generates and sends the secure push message to client device 102 by appending the electronic signature to the message and/or other information. *Id.*, [0067], FIG. 8.

At client device 102, secure push message client 404 validates the message and sender using the electronic signature. *Id.*, [0109]-[0112], [0041].

2. Analysis

In TS-23.140, MMS User Agent routes application-specific messages to a destination application referred to "from the destination application identifier (***based on the negotiated details upon application registration process***)." SAMSUNG-1004, 54-56 Therefore, the message includes data indicating the authorized/registered applications (software components) allowed to receive data from secure message link messages via the MMS Relay/Server (message link server). SAMSUNG-1003, ¶¶250-255.

TS-23.140 does not expressly describe message validation/authentication that verifies the sender's identity or that the destination application is the intended recipient. SAMSUNG-1004, 54-56; SAMSUNG-1003, ¶¶256-257.

Rakic provides a known solution to address these security concerns in push communications (including MMS communications). SAMSUNG-1003, ¶¶257-258. Rakic discloses database device 108 that stores information related to client devices (and/or components included therein), and secure push message server 504

uses this stored information to generate a key, an electronic signature, and a secure push message for a push message intended for a particular client device. SAMSUNG-1015, [0047]-[0049], [0065]. The secure push message server generates the key based on the above-described code(s) that are obtained from client device 102 or a component thereof, and subsequently, generates an electronic signature based on the key and the message. *Id.*, [0066]. Server 504 generates and sends the secure push message to client device 102, e.g., by appending the electronic signature to the message and/or other information. *Id.*, [0067]. At the client device, a secure push message client validates the message and the sender by generating a signature using a generated key and the message's data block. *Id.*, [0109]-[0112]. Message validity is confirmed if the received and generated signature portions match. *Id.*; SAMSUNG-1003, ¶¶259-260.

A POSITA would have been motivated to implement Rakic's teachings within the MMS Relay/Server to address the above-described security problems to achieve data and computer security improvements for the client device and the underlying applications, e.g., to enable (1) the client device to "authenticate the sender" and (2) "verify that an intended recipient of the secure push message is the client device, and that the client device includes a correct component." SAMSUNG-1015, [0041]; SAMSUNG-1003, ¶¶258-263 (citing SAMSUNG-1034, -1035).

A POSITA would have therefore implemented techniques performed by Rakic's secure push message server 504 and database device 108, including its signature generation functionality/components and database storing the device information, within MMS Relay/Server. SAMSUNG-1003, ¶¶261-264. This would amount to using a known technique to improve similar devices in the same way, and combining prior art elements per known methods to yield predictable results with a reasonable expectation of success. *Id.* Because TS-23.140 and Rakic contemplate similar messaging architectures/systems, a POSITA would have found it straightforward to implement Rakic's teachings within MMS Relay/Server. *Id.* Additionally, the resulting system's elements would perform functions performed prior to combination—MMS Relay/Server would communicate messages to MMS User Agents and other network elements, and Rakic's teachings (in combination) would enable providing secure signatures (with messages) to the receiving device(s). SAMSUNG-1003, ¶265.

In combination, Rakic's secure push message server and associated signature generation functionality would be readily implemented in the MMS Relay/Server and would be used to generate signatures for messages to different UE/terminals and their respective MMS User Agents. SAMSUNG-1003, ¶266. These signatures are authorization signatures because they authenticate the sender and/or verify that the message's intended recipient is the client device and that the

device includes a correct component. SAMSUNG-1015, [0041]; SAMSUNG-1003, ¶267.

Additionally, a POSITA would have found obvious that a "component" would be a device application, given that Rakic's teachings are not limited to particular hardware components (and would be readily understood to include software components—considering that these are the components to which messages are being routed). SAMSUNG-1003, ¶268. Therefore, in combination, a POSITA would have found obvious that the above-described secure authorization signatures would indicate the authorized/registered applications (software components) are allowed/authorized to receive messages including application data via the MMS Relay/Server (via the secure message link, per claim 1 above). SAMSUNG-1003, ¶¶250-269.

E. Ground 5 Claims are Rendered Obvious by Houghton and Munson

1. Houghton

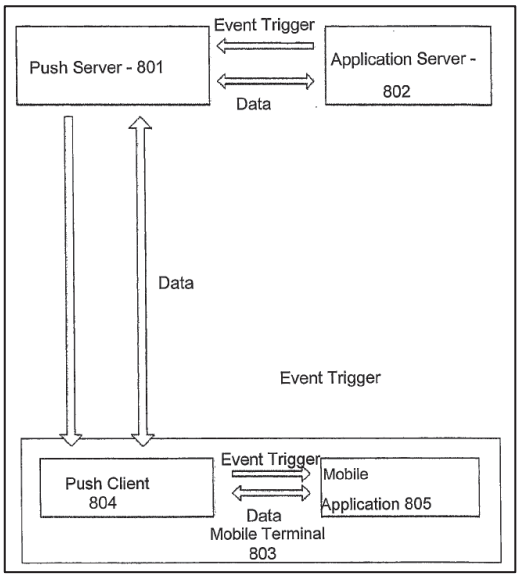
Houghton discloses a server that "push[es] messages to a" push client executing on a "mobile terminal" in a wireless network. SAMSUNG-1007, Abstract, 16:21-25[5]; SAMSUNG-1003, ¶¶86-92. These messages "cause programs to start

[5] Citations in Houghton refer to the publication page number.

to a specified operating state" or "change [the] operating state" to a particular, specified state. *Id*.

The push messaging system is implemented using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." SAMSUNG-1007, 19. A "persistent managed, tested and configured data connection" is established "between push server 401/801 and push client 405/804." *Id.*, 23:3-21. Additionally, "IP or other API (application programming interface) connection[s]" are established between the application/push server and client/application. *Id*.

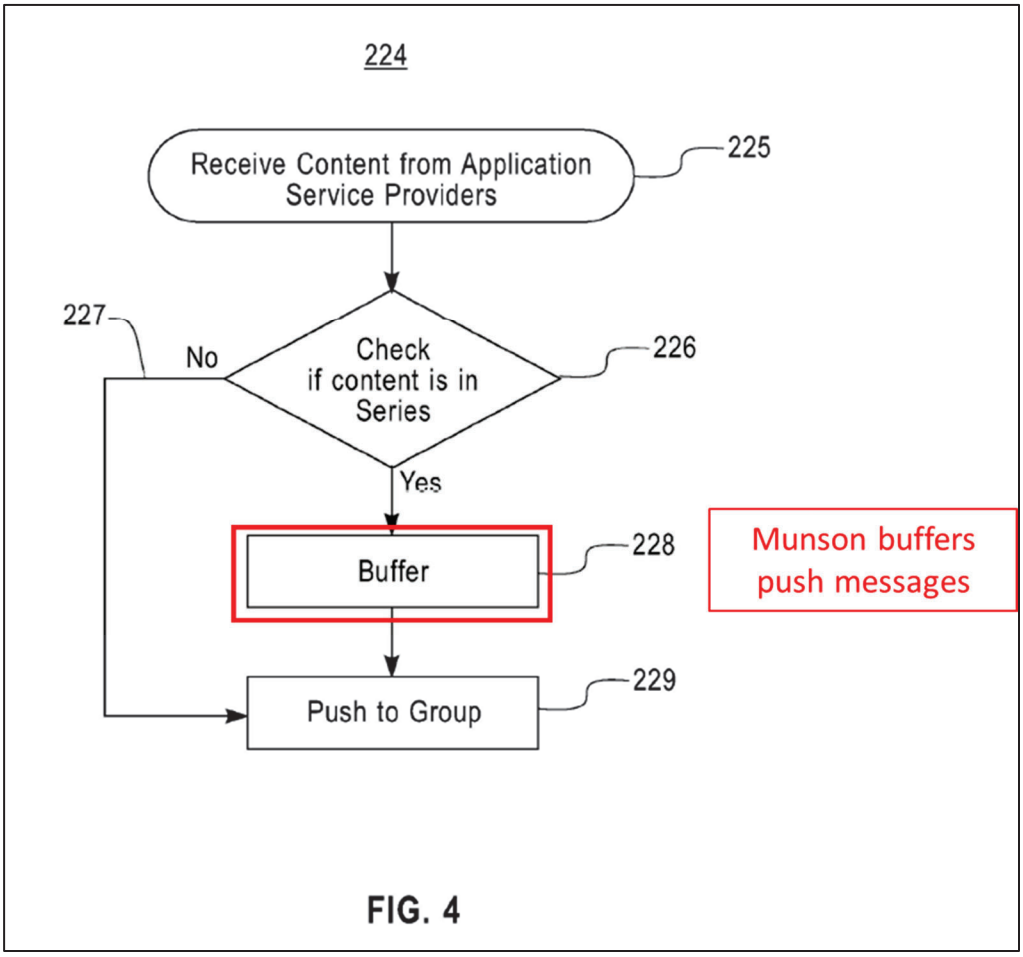
Figure 8 (below) depicts the above-described communication between the mobile terminal and push server 801, and "a data connection between application server 802 and mobile application 805" (arrow C in FIG. 4) between push server 401/801 and push client 405/804. *Id*., 23.



SAMSUNG-1007, Figure 8; Figure 4.

2. Munson

Munson discloses a method of "pushing contents to client devices." SAMSUNG-1017, Abstract; SAMSUNG-1003, ¶93. Munson discloses "group pushes" where content is buffered and sent to multiple devices simultaneously, "serializ[ing]" content such that a series of messages are delivered to a particular device simultaneously. SAMSUNG-1017, 3:7-67, 4:1-56, FIGS. 1-5. Illustrated below is an example process, showing buffering of received push messages:



SAMSUNG-1007, FIG. 4.

3. Combination of Houghton and Munson

A POSITA would have found obvious to combine Houghton's and Munson's teachings such that Munson's method of buffering and pushing messages would have been incorporated into Houghton's push messaging system. SAMSUNG-1003, ¶¶270-271; *see supra* §§III.E.1-2.

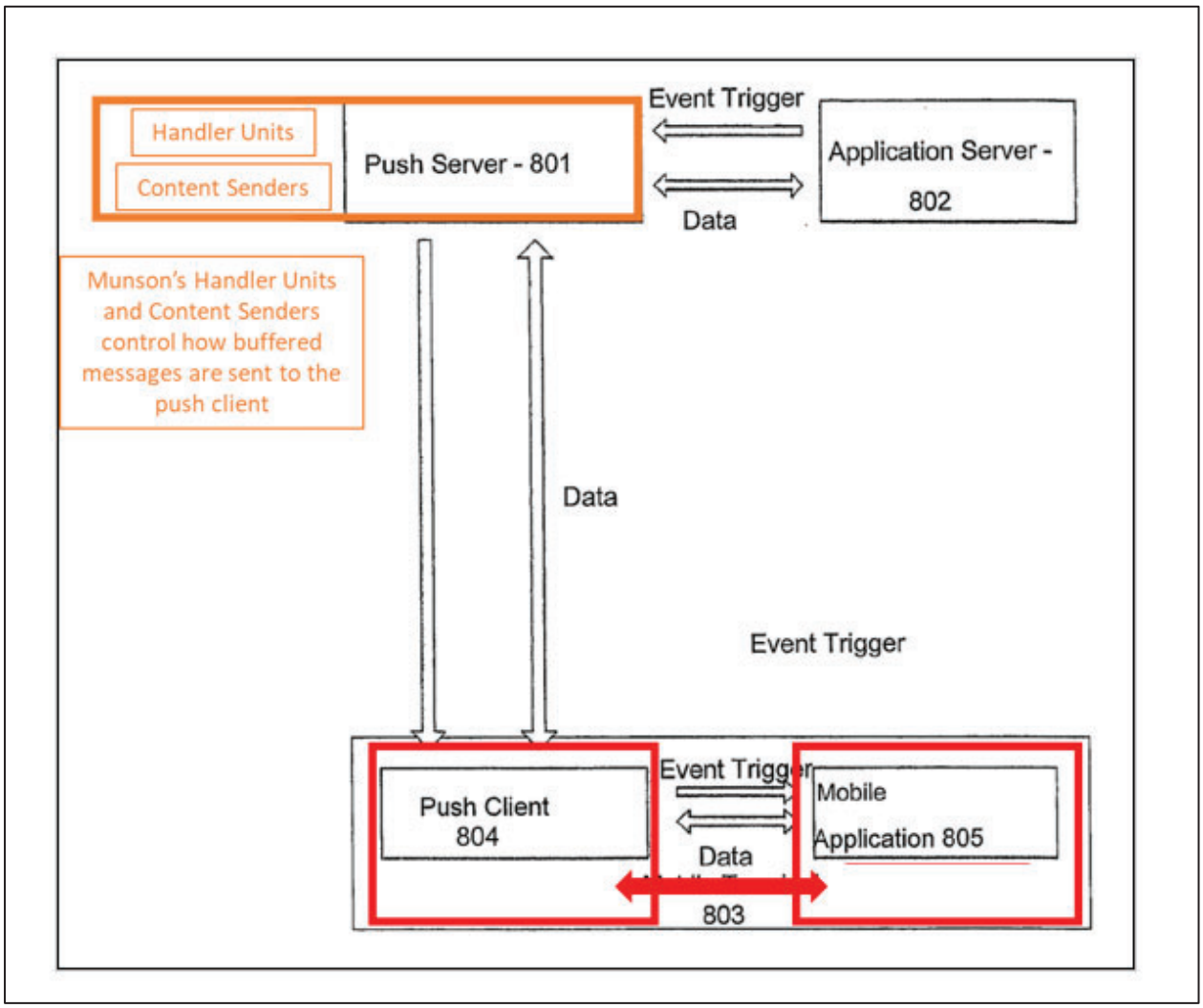
A POSITA would have recognized or found obvious that a recipient mobile terminal may not have network connectivity, such that a push server would store/buffer the message until the terminal has network connectivity and the message can be delivered. SAMSUNG-1003, ¶272. Houghton contemplates such "store and forward messaging systems" and discusses that messages may not deliver while the user is "unavailable" or out of network. SAMUNG-1007, 3, 7. Moreover, it was well-known for store-and-forward push systems to store messages in buffers/memory "when the mobile terminal is not connected to the mobile internet service network" and "re-transmitting the stored push information when" the terminal is reconnected to the network. SAMSUNG-1003, ¶272; SAMSUNG-1006, claim 14, FIG. 1, ¶34.

Similarly, Munson provides a push messaging solution for storing messages for later delivery to mobile terminals that would have been efficient and straightforward to implement within Houghton's push messaging system. SAMUNG-

1007, 3, FIGS. 3-4; SAMSUNG-1017, 3:66-4:11, 5:22-33, FIG. 3; SAMSUNG-1003, ¶273.

A POSITA would have been motivated to implement Munson's message buffering (and associated functionality/components) within Houghton's system to avoid losing/discarding received messages. SAMSUNG-1003, ¶273. Additionally, like Munson, Houghton already contemplates "store and forward messaging systems" and the above-described scenario where a mobile terminal may be outside network range/coverage. SAMAUNG-1007, 3, 7; SAMSUNG-1003, ¶275. Thus, combining Houghton and Munson's teachings would have been straightforward and would have amounted to using known prior art techniques (message buffering) to improve similar devices/systems (push messaging systems) in the same way that yields predictable results (push messaging systems that store and forward messages). SAMSUNG-1003, ¶¶276-277.

In the resulting Houghton-Munson combination, and as illustrated below, Munson's handler units and content senders would have been incorporated into Houghton's push server to enable buffering push messages and subsequently delivering those messages upon occurrence of certain conditions/triggers. SAMUNG-1007, 3, 7, FIGS. 3-4; SAMSUNG-1017, 3:66-4:11, 5:22-33, FIG. 3; SAMSUNG-1003, ¶¶277-278.



SAMSUNG-1007, FIG. 8 (modified to incorporate Munson).

Incorporating Munson's buffering and later message transmission (e.g., upon occurrence of a trigger) into Houghton's push messaging system would have been predictable and foreseeable with a reasonable expectation of success because (1) Houghton and Munson disclose push messaging systems where push messages are transmitted from a push server to a mobile terminal upon occurrence of a trigger

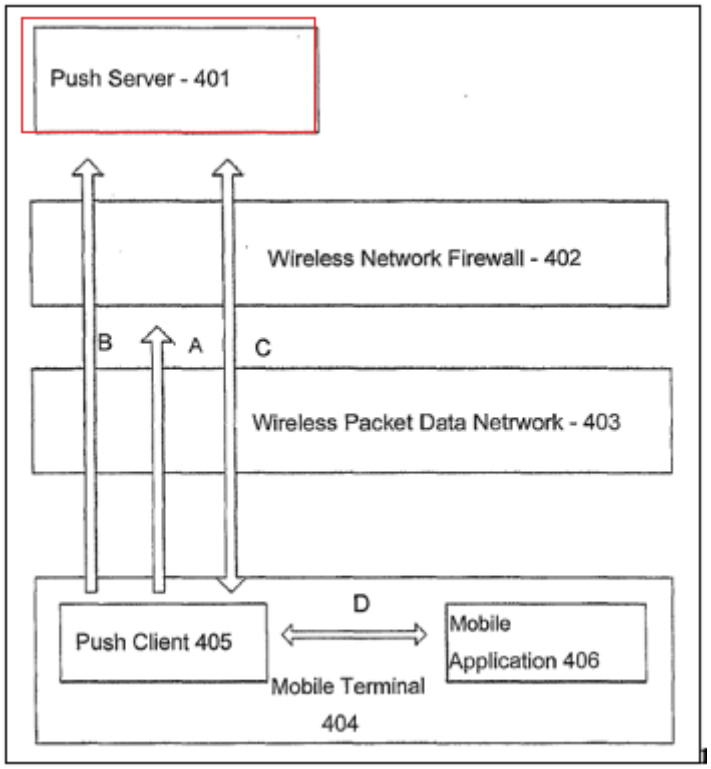
event and (2) Houghton contemplates storage and forward systems, which are described in Munson. SAMUNG-1007, 3, 7, FIGS. 3-4; SAMSUNG-1017, 1:11-26; SAMSUNG-1003, ¶¶276-277. Thus, in combination, Houghton and Munson's teachings would operate like they did prior to combination—Houghton's system would push messages between push server and push client, and Munson's teachings (in combination) would facilitate buffering push messages in the server's buffer/memory and delivering them upon occurrence of certain triggers (e.g., device returns to network range). *Id.*

4. Analysis

(a) *Claims 1, 15*

[1pre]/[15pre]

If the preamble is limiting, Houghton-Munson renders obvious a message link server (push server 401/801) and method for operating the same. SAMSUNG-1003, ¶¶279-282. As shown below, push server 401/801 sends push messages via communication links over a network to push client 405/805 on mobile terminals 404/804 for delivery to mobile applications 406/805:



SAMSUNG-1007, FIG. 4 , FIG. 8, 16-17, Abstract; SAMSUNG-1003, ¶¶279-280.

Additional details regarding Houghton-Munson's push server and the method for operating this server are described below ([1a]-[1d4]/[15pre]-[15d3] *infra*). SAMSUNG-1003, ¶¶281-282.

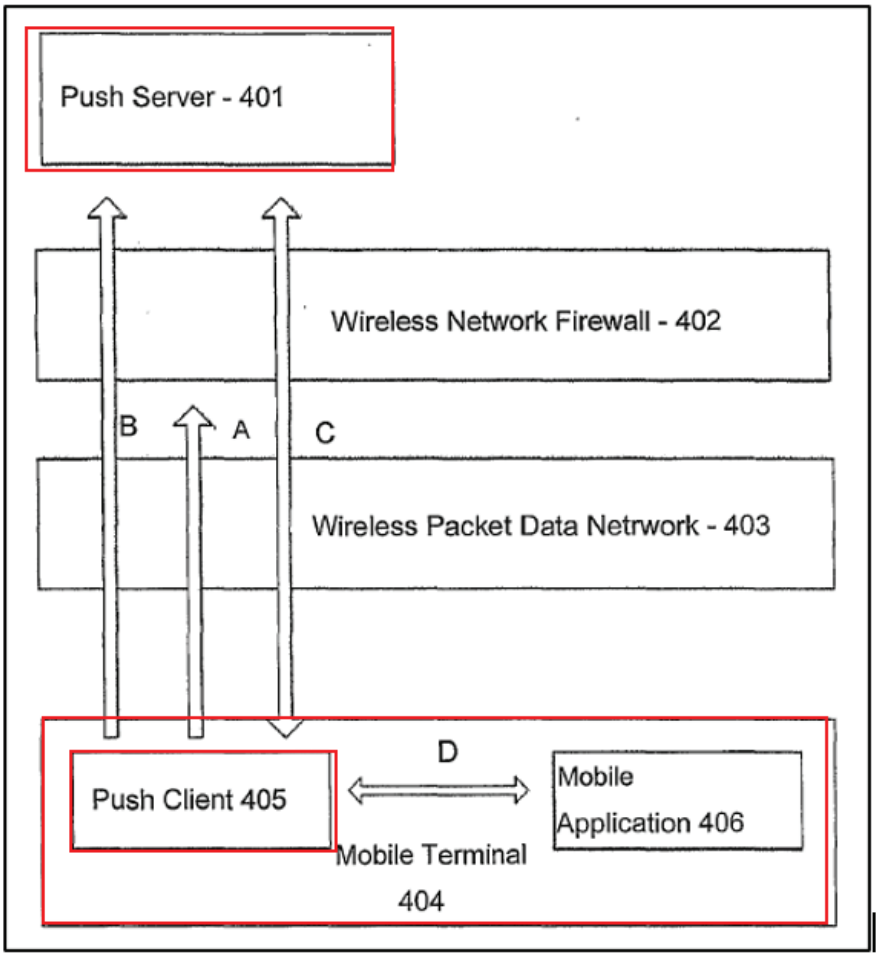
<u>[1a]/[15a]</u>

Houghton-Munson renders this limitation obvious consistent with its description in the '192 specification. SAMSUNG-1003, ¶¶283-298; SAMSUNG-1001, 90:50-61, 99:8-32, 17:13-22 ; *see* [1a]/[15a] in §III.A.2 *supra*.

As described below, Houghton-Munson renders obvious a transport services

stack that maintains a secure message link through an Internet network between the message link server (push server 401/801) and a device link agent (push client 405/805) on respective client devices (mobile terminals 404) and enabling Internet communications using secure transport protocols facilitating such links. SAMSUNG-1003, ¶¶284-286.

Specifically, push client software 405 on mobile terminal 404 connects to "***push server 401 over a data network 403***" (e.g., "***a wireless network***" for "***transmitting Internet Protocol (IP) packets***"). SAMSUNG-1007, 17, FIG. 4 (below; annotated).



Push client 405/805 constitutes the device link agent because it is "software" running on mobile terminal 404 that "initiates and maintains" a link using "Internet technologies" between the terminal and push server 401, and using this link, push server 401 pushes messages to the mobile terminal. *See* SAMSUNG-1007, 11, Abstract, 20; SAMSUNG-1001, 43:18-50 (interchangeably using agent and software (such as an application) running on a device); SAMSUNG-1003, ¶¶287-288.

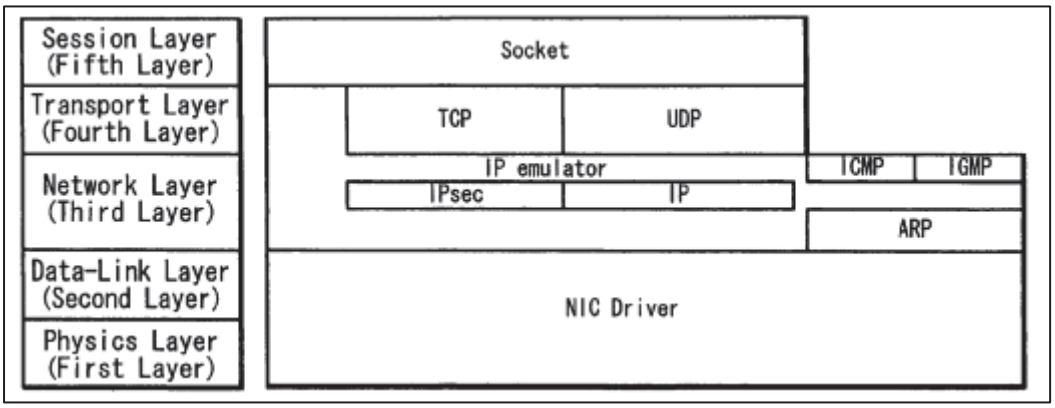
For this connection, Houghton uses known transport protocols, including

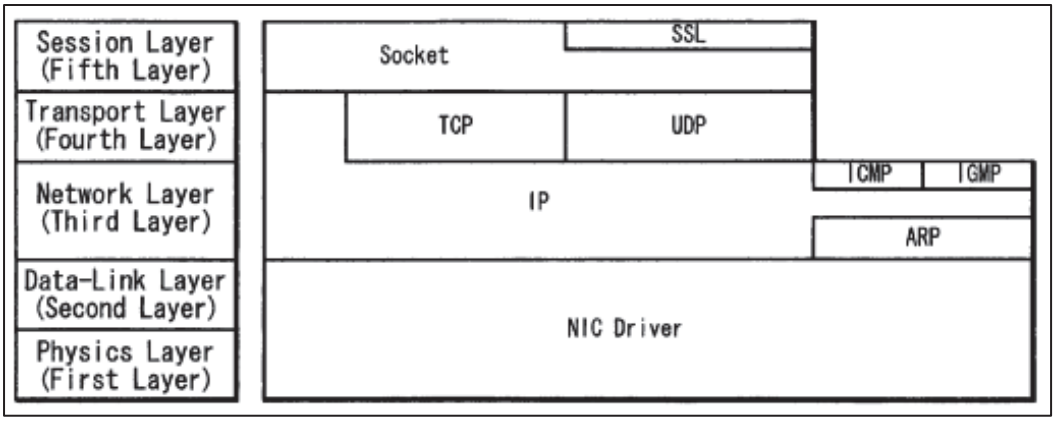
"***connection-oriented protocol[s]" such as TCP/IP***," "***connectionless protocol[s]***" such as UDP/IP, "***alternate connection-oriented protocol[s]***" such as HTTP, or "***secure protocol[s]***" such as "***HTTPS, IP-Sec, secure IP6 or a proprietary security protocol***" that prevents message interception and notification from third parties and identifies communicating parties. SAMSUNG-1007, 18-20, claim 10. Upon connection establishment, push client and push server "push a message" to each other (*see* arrow C in Figure 4). *Id.*, 19; SAMSUNG-1003, ¶¶289-290.

Thus, Houghton describes the server-client communications using secure transport protocols similar to those in the '192 specification. SAMSUNG-1003, ¶291; SAMSUNG-1001, 90:33-61, 100:8-32, 17:13-22 (identifying secure encryption protocols, e.g., TLS/SSL, IP-Sec, HTTPS).

A POSITA would have understood or found obvious that Houghton's push server would include software and hardware components/drivers that implement such secure transport protocols/communications over an Internet network with mobile terminal 404. SAMSUNG-1007, 18-20; SAMSUNG-1003, ¶291. Before the Critical Date, it was well known for devices implementing such transport protocols to implement a transport services/protocol stack using software and hardware components, that interoperate to facilitate secure network communications with other network elements. SAMSUNG-1003, ¶¶292-293; *see id.*, ¶¶26-31 (citing SAMSUNG-1027, -1028, -1030); SAMSUNG-1005, ¶¶12-22 (implementing TCP/IP

protocol stacks in a device's "software or firmware," including hardware and associated drivers; using SSL and IPSec for encrypting packets/data), FIGS. 25-26 (below).





Because Houghton contemplates TCP/IP-based transport protocols for transporting data/messages between network entities, a POSITA would have been motivated to implement well-known TCP/IP transport stacks in the push server's software and hardware (as Ozaki discloses/corroborates) to implement its data transport over a network. SAMSUNG-1003, ¶294. Implementing such well-known TCP/IP transport stacks in the push server, which supports and facilitates

TCP/IP protocols and associated network communications, would therefore have amounted to implementing known techniques within known devices to achieve predictable results. *Id.*

Moreover, the communication link established between the push client/mobile terminal and the push server 401 is a secure message link because it facilitates sending push messages between push client 405 and push server 401 using secure/encryption protocols (e.g., IPsec, HTTPS, SSL) over a network link. SAMSUNG-1007, 11, 18-20, claim 10; SAMSUNG-1003, ¶¶295-297; SAMSUNG-1001, 39:20-32, 70:31-44, 90:33-61, 96:15-28, 17:13-22.

Given Houghton uses secure protocols (e.g., SSL, HTTPS, IPSec), and the well-known use of such protocols to secure TCP/IP-based network communications, a POSITA would have understood or found obvious that the transport services stack that would be implemented in Houghton's push server would use such protocols for securing the TCP/IP based communication link between push client 405/804 and push server 401/801. SAMSUNG-1003, ¶296.

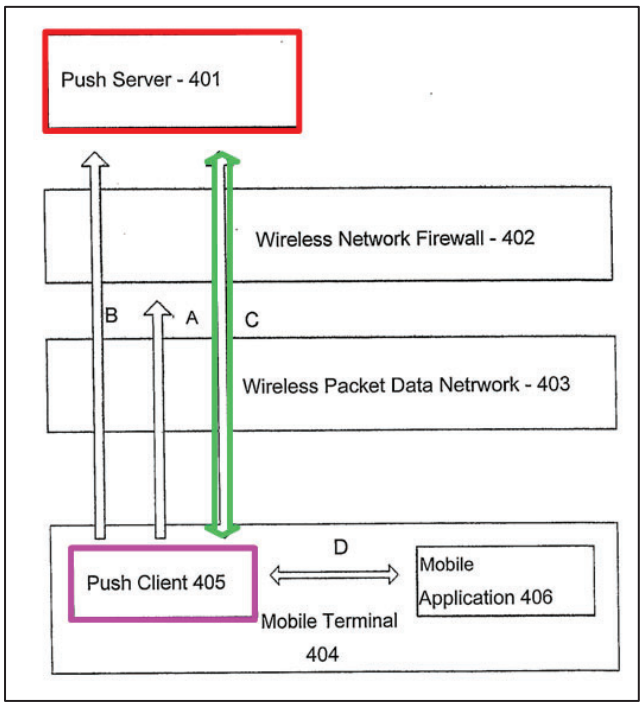
A POSITA would have understood that, in push messaging (per Houghton), multiple mobile terminals/associated push clients would be in communication with push server 401, thereby forming separate secure SSL or IPSec-based TCP/IP communication links between the push server and each respective terminal/push client. SAMSUNG-1003, ¶¶297-298; SAMSUNG-1037, 2-3; SAMSUNG-1007,

18-19 (disclosing multiple "terminals").

[1b]/[15b]

Houghton-Munson renders obvious that each wireless end-user device (mobile terminals 404) includes multiple software components (mobile applications 406) authorized to receive and process data (application-specific data) from secure message link messages received via a device link agent on that device. SAMSUNG-1003, ¶¶299-318.

In Houghton, wireless end-user device (mobile terminal 404) includes multiple software components (mobile application(s) 406), and messages (command push or application command messages) are provided to the respective applications via device link agent (push client 405/805) on the device/mobile terminal 404. SAMSUNG-1003, ¶300; SAMSUNG-1007, FIGS. 4, 8, 21-22, 24-25.



Push client is implemented in "software run" in mobile terminal 404's processor and operates within "a message passing system" coordinating with Push Server 401. SAMSUNG-1007, 16-17; SAMSUNG-1001, 43:18-31, 11:51-57 (describing device agent as software executing in a device processor); SAMSUNG-1003, ¶301.

Upon connection establishment between push client 405 and push server 401, "the ***server may send a push message***" to push client 405 through the "previously established, connection-oriented protocol such as TCP/IP, SSL, HTTP or HTTPS" (arrow C above). SAMSUNG-1007, 20, 34, claim 33 (push client "unwrap[s]" security on "behalf of the mobile application"). Because this link is secured using SSL/IPSec/HTTPS, the received push message from push server over

this secure message link constitutes a secure message link message. SAMSUNG-1003, ¶302.

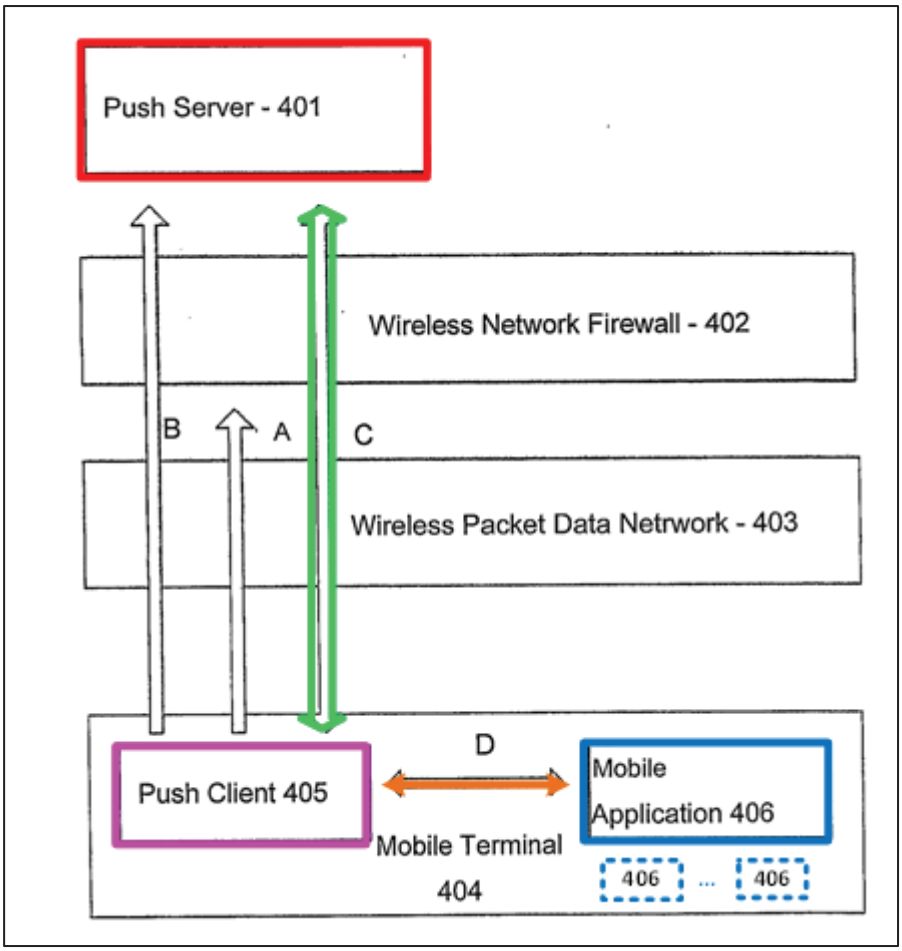
A POSITA would have understood or found obvious that push server 401 communicates push messages to multiple "mobile terminals," each including its respective push client 405 and mobile applications 406. SAMSUNG-1003, ¶303; SAMSUNG-1007, 21 (routing messages to "servers, ***mobile terminals, push clients and computing devices***").

The received push messages include "application-specific data," e.g., "application commands," "updates," or "data messages" "***directed to a push application***" from among ***multiple applications*** via the client-initiated and maintained permanent connection. SAMSUNG-1007, claim 1.

The received data is processed by the receiving application to, e.g., present certain data or feature or operate the application in a particular state. SAMSUNG-1003, ¶¶304-305. For example, push messages include "a message stimulating, triggering or commanding" an ***"application"*** to a "***designed application view or location, present new information of the availability of new information, a new feature or operating state***." SAMSUNG-1007, 11-12.

The push message includes "a data packet" with "information ***specifying which mobile application 406 from a plurality of such applications***" and additional "information to be passed to the … specified mobile application." SAMSUNG-1007, 21-22; SAMSUNG-1003, ¶307.

Push client 405 directs the push message to the specified mobile application 406 via communication path D:



SAMSUNG-1007, FIG. 4 (annotated), 21, 24.

Because push messages are routed to a particular application from among

multiple applications, and the destination application operates on the received message data (e.g., launch application, present content, etc.), a POSITA would have understood or found obvious that these applications are authorized to receive and process data included in the command push/application command messages. SAMSUNG-1003, ¶¶308-309.

Additionally/alternatively, if it is determined that Houghton does not explicitly disclose that multiple software components/applications are ***authorized*** to receive and process data from secure message link messages, a POSITA would have found that to be obvious. SAMSUNG-1003, ¶¶310-311. Before the Critical Date, applications were known to register with a push server and/or push client before receiving messages via push frameworks. SAMSUNG-1003, ¶¶311-312; SAMSUNG-1006, ¶23; SAMSUNG-1018, ¶¶17, 63, 106-109; SAMSUNG-1004, 54-56 (similar).

A POSITA would have found obvious to implement the above-described and well-known authorization/registration processes within Houghton to enable registration of a particular application with a push client and/or push server. SAMSUNG-1003, ¶313. A POSITA would have been motivated to do so for multiple reasons, including to enable dynamic content delivery "to have information or data pushed" to devices without users of such devices having to "seek out that data" and

ensure resource-efficient message delivery. SAMSUNG-1003, ¶¶314-315; SAMSUNG-1018, ¶¶3-6.

Moreover, because Houghton contemplates that push clients/servers coordinate on message delivery to and from mobile terminal applications, implementing well-known application registration/authorization teachings in push environments, within Houghton's system, would have been straightforward and a POSITA would have had a reasonable expectation of success in combining the teachings in the above-described manner. SAMSUNG-1003, ¶316.

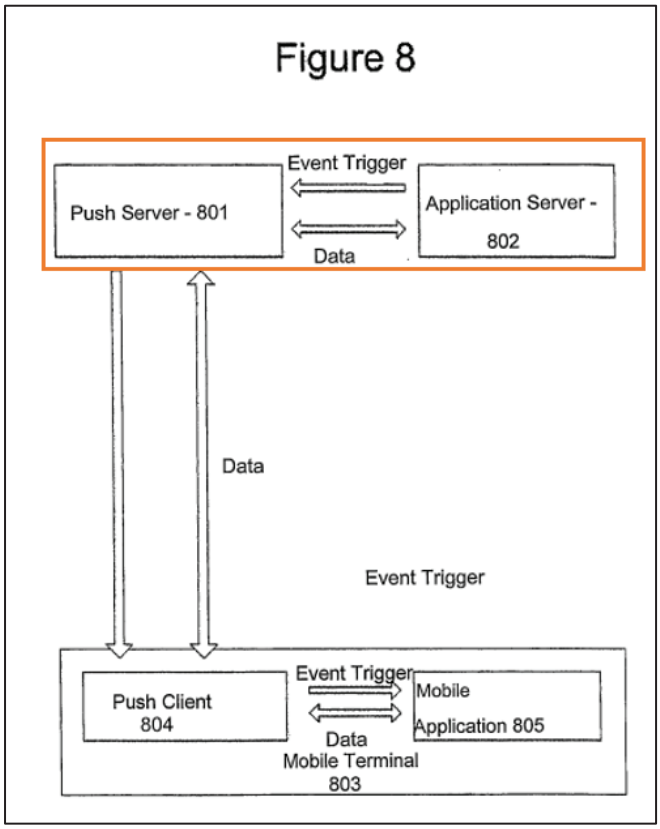
Therefore, a POSITA would have found obvious that Houghton's push messages would be routed by push client 405 to applications that are authorized/registered to receive them. SAMSUNG-1003, ¶¶317-318.

[1c1]/[15c1]

Houghton-Munson renders obvious an interface to a network (application programming interface (API)) to receive network element messages (application-specific messages/data) from multiple network elements (application server(s) 702/802). SAMSUNG-1003, ¶¶319-329.

Houghton discloses a "COMMAND PUSH" procedure where the push server 701 is triggered by a trigger event from "***application server 702***" to push an application command message ***to*** push client 704. SAMSUNG-1007, 21-22. As

shown below, "a data connection between application server 802 and mobile application 805 is established" using , in part, "[a]***n IP or other API … connection between application server 802 and push server 801***." *Id.*, 22-23, 14; FIG. 8 (below; annotated), 14 (using an API), 21 (transmitting data from applications to push server for "***redirection through push server API*** to ***other servers***"); SAMSUNG-1003, ¶¶320-323.



Moreover, a POSITA would have found obvious that the application server communicates over a network using the API or IP connection with push server

401. *See* SAMSUNG-1007, 23; SAMSUNG-1003, ¶¶324-325.

Additionally, a POSITA would have understood or found obvious that the Houghton's push environment would include multiple application servers, each communicating with the push server to exchange data/messages with mobile terminal(s). SAMSUNG-1003, ¶¶326-329; SAMSUNG-1007, 21.

[1c2] / [15c2]

Houghton's push server sends push messages from an application server to a mobile application (from among multiple such applications) on a mobile terminal. SAMSUNG-1007, 21-22. The message is sent using "***a data connection*** between application server 802 and mobile application 805," using the data connection (arrow C; FIG. 4) between push server 401/801 and push client 405/804. *Id.*, 23; SAMSUNG-1003, ¶¶330-332

The message pushed from push server 401/801 to push client 404/804 includes "a data packet" including (1) "information ***specifying which mobile application 406 from***" multiple "***applications***" is the message recipient and (2) "***information***" for the "***specified mobile application***" along with ***additional "information specifying how***" to present/use/process such information. *Id.*, 21; claim 1 ("***pushing commands and data to a wireless terminal*** from a push server"), 8.

A POSITA would have therefore understood that Houghton discloses/suggests that the received network element messages (push messages from application server(s) 702/802) include message content with data for the destination mobile application and application identifier. SAMSUNG-1003, ¶¶333-334.

If the present limitation requires an application identifier for the application, Houghton does not expressly disclose that. Houghton, however, states that the message "specif[ies]" the destination "mobile application from a plurality of such applications." SAMSUNG-1007, 21; SAMSUNG-1003, ¶¶334-335. Before the Critical Date, it was well-known for such information specifying/identifying a particular application to be an application identifier. SAMSUNG-1003, ¶336; SAMSUNG-1015, ¶22; SAMSUNG-1006, ¶¶13, 22 (push message includes application's "app-ID"); SAMSUNG-1004, 54-56 (similar).

Because Houghton routes messages to an application based on the information specifying that application and the well-known use of application identifiers for application identification, a POSITA would have found obvious to implement Houghton's teaching of the "information specifying" the mobile application as the application identifier. SAMSUNG-1003, ¶337. So implementing Houghton's system amounts to implementing a known technique (application identifier)

to a known system (Houghton's push messaging system) to achieve predictable results (routing messages to applications based on their identifiers, as suggested by Houghton and well-known in the art). SAMSUNG-1003, ¶338.

Moreover, a POSITA would have found obvious that the received push message requests message content delivery to one or more mobile terminals. SAMSUNG-1003, ¶¶339-341. As explained above, "push server 701 is triggered by a trigger event … from an application server 702 to push an application command message" to push client 704 and "***thereby initiate a mobile terminal client trigger event in a mobile application 705***" from multiple applications "on the terminal 705." SAMSUNG-1007, 21-22. The trigger mechanism is "***an IP-triggered application launch and data transfer***" e.g., "***mobile application launched by contact to a specified IP port***." *Id.* "[P]ush server 901 is triggered by a trigger event of which the application server 902 is aware to ***send push message to the mobile terminal 905***," causing "pushing application launch commands, lazy application updates, and lazy application data updates ***through push clients 904 and 905 to mobile application 907***." *Id.*, 24, FIG. 5; SAMSUNG-1003, ¶¶339-341.

<u>[1d1]-[1d2] / [15d1]</u>

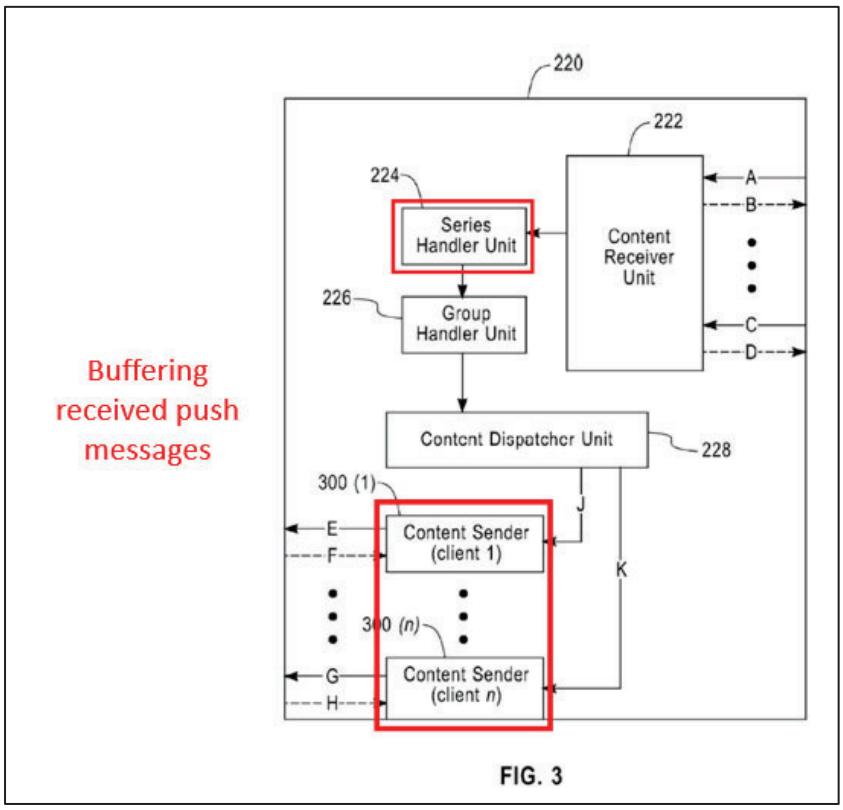
Houghton-Munson renders obvious a message buffer system including a memory that buffers content from the received network element messages for

which delivery is requested to a given one of the wireless end-user devices. SAMSUNG-1003, ¶¶343-352.

As explained above ([1c1]-[1c2] *supra*), Houghton-Munson renders obvious that the push server receives network element messages (push messages) from application servers, where message delivery is requested to one or more mobile terminals/wireless end-user devices. SAMSUNG-1003, ¶344.

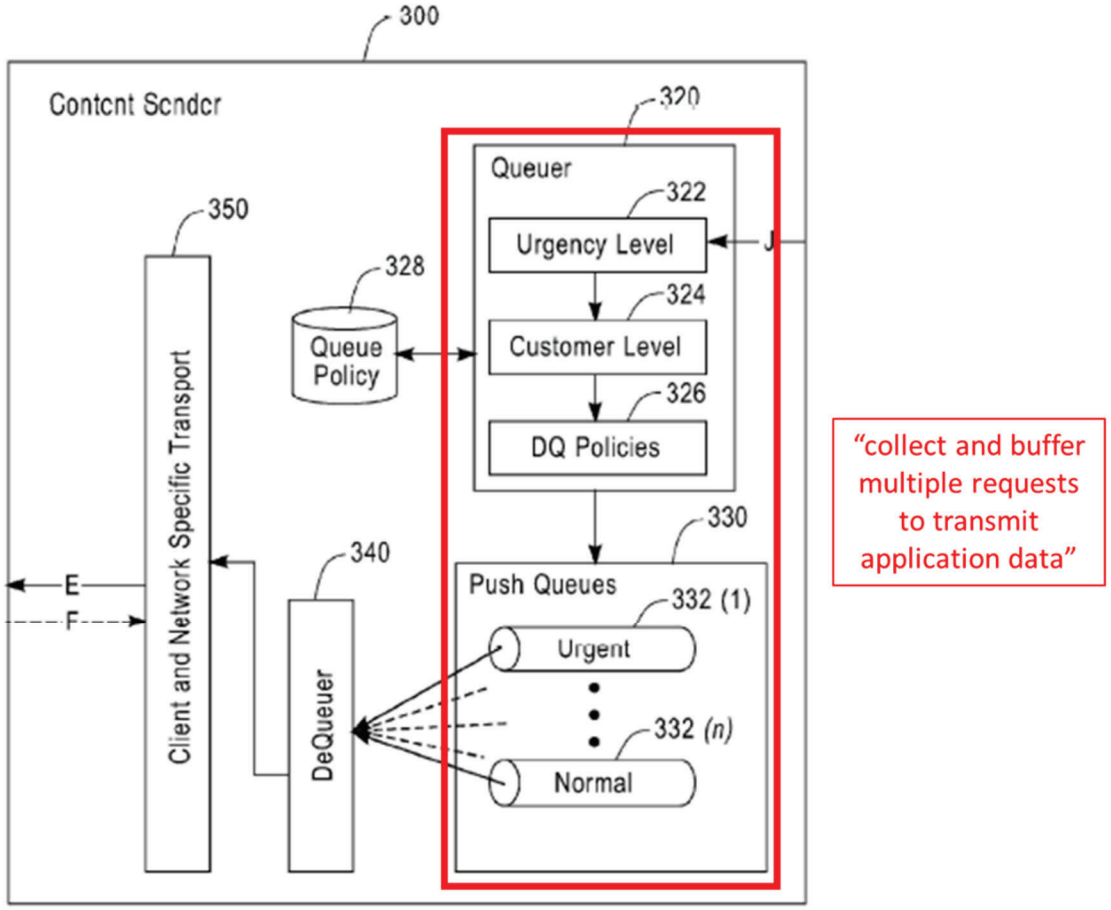
The push server includes a message buffer system including logic (*see* [1d3]- [1d4] *infra*) and based on Munson, would have included memory that buffers content from the received push messages. Specifically, as described in §III.E.3 *supra*, a POSITA would have found obvious, per Munson and Houghton, to implement a buffer to store received push messages on Houghton's push server. SAMSUNG-1003, ¶¶345-347; SAMSUNG-1007, 3, 7, FIGS. 3-4.

Indeed, the combination leverages Munson's teachings of storing push messages in push server's memory/storage, including a "Series Handler Unit 224" that "receives contents from application service providers" and uses a "series buffer … to keep the contents." SAMSUNG-1017, 3:66-67, 4:1-11; SAMSUNG-1003, ¶349. Munson's FIG. 3 (below; annotated) shows a structure queuing messages in content push service 220. SAMSUNG-1003, ¶346.



SAMSUNG-1017, FIG. 3.

Munson discloses multiple "Content Sender[s] 300" that each include a "Queuer 320" and "Push Queues 330." SAMSUNG-1017, 4:12-42, FIGS. 3, 5; SAMSUNG-1003, ¶350. When pushing messages, content senders 300 check "urgency level of contents," "customer level," and "dequeue policy," and each "client device" (Houghton-Munson's mobile terminal) is assigned a content sender 300. SAMSUNG-1007, 3:40-65, 4:12-42, FIGS. 3, 5. Munson's Figure 5 illustrates a structure for queuing messages in a content sender 300. SAMSUNG-1007, FIG. 5; SAMSUNG-1003, ¶¶350.



SAMSUNG-1017, FIG. 5.

Per the above-described teachings in Munson and Houghton, and as explained in §III.E.3 *supra*, a POSITA would have found obvious to implement Munson's message buffering functionality and associated components within Houghton's push server to enable storage and later forwarding of messages to one or more mobile terminals/push clients, e.g., upon occurrence of certain trigger events. SAMSUNG-1003, ¶¶270-278, 344-351. Thus, in the Houghton-Munson

push server, the received network elements messages would be stored in the buffer and associated components, per Munson. *Id.*

Additionally, as described below ([1d3]-[1d4] *infra*), Houghton's push server includes a message buffer system with logic facilitating delivery of messages upon occurrence of one or more message delivery triggers.

[1d3] / [15d2]

Houghton discloses occurrence of trigger events that trigger sending/delivering push messages to one of the wireless end-user devices (mobile terminals). SAMSUNG-1003, ¶¶353-365. In Houghton, "***push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server***" and "***include an alarm, notification, or measurement result received to the push server 401 from another device or system***." SAMSUNG-1007, 21; *see id.*, 21-22, 25-28 (describing various trigger events); SAMSUNG-1003, ¶¶353-355.

These triggers—e.g., alarm, notification, or measurement result—are message delivery triggers that are not based on message receipt by the push server. SAMSUNG-1003, ¶356.

Additionally, Houghton and Munson render obvious that the message delivery triggers include an occurrence of "an asynchronous event with time-critical

messaging needs," which includes a user request for message delivery or occurrence of a transaction, per the '192 patent. SAMSUNG-1001, 38:50-64; *see* III.A.2 ([1d3] analysis); SAMSUNG-1003, ¶¶357-358.

This teaching and its application in push systems was well known in the art. For example, Munson discloses that its push system provides "***asynchronous content push*** (i.e., pushing a content) to clients on diverse wireless networks" "according to a schedule of time ***or event***" such as coordinating pushes "for a system maintenance purpose during off-peak hours." SAMSUNG-1017, ¶¶40, 44; SAMSUNG-1003, ¶¶359-360.

Additionally, Houghton's trigger event is an alarm or notification received from another device (as described above). SAMSUNG-1003, ¶¶359-360. The message delivery can be adjusted based on whether the message to be delivered is time-critical or "non-time-critical." *See* SAMSUNG-1007, 23-24 (describing use of a "more desirable network connection" that "***benefits customers using the wireless terminal in that data is managed and updated automatically for their mobile applications in the most cost efficient manner for non-time-critical capabilities such as lazy application code update and lazy application data update***."), 21-22 (contemplating "time critical and the fastest available" techniques). A POSITA would have therefore found obvious, per Houghton and Munson's teachings, to

modify message delivery based on the message's time criticality and therefore, adjust message delivery and the associated trigger on an asynchronous basis. SAMSUNG-1003, ¶¶359-361.

Moreover, Houghton's push client sends a message based on a user request to the push server, requesting message delivery. SAMSUNG-1003, ¶361. "IP push command messages" are triggered when "push client makes" push server "aware of" "client-side events," which trigger push server to deliver messages intended for the terminal's mobile application(s) 406. SAMSUNG-1007, 14, 21; SAMSUNG-1003, ¶361. "[C]lient-side events" include creating "***photographs, video or other media***," "***video game actions or events***," "***messaging actions***" and "***remote application user*** or application server ***event***, ***remote user action***." *Id.*, claims 22, 14, 21.

Therefore, per Houghton and Munson, a POSITA would have found obvious that the combination discloses asynchronous client-side events, including user-requested events or transactions/events occurring on a mobile terminal (e.g., creating media, gaming events, messaging). SAMSUNG-1003, ¶¶361-362. A POSITA would have thus understood or found obvious that such events/user requests/transactions constitute message delivery triggers that include occurrence of an asynchronous event with time-critical messaging needs. SAMSUNG-1003, ¶¶363-364.

Because push server awaits these triggers to occur before delivering messages to push clients, a POSITA would have understood or found obvious that push server includes logic that determines when one (or more) of these message delivery triggers for the particular terminal/end-user device has occurred and if so, delivering the messages. SAMSUNG-1003, ¶¶365-366.

[1d4] / [15d3]

As described in [1a]-[1b], Houghton's push server would have included a transport services stack that facilitates a secure message link between the push server and the push client. SAMSUNG-1003, ¶¶366-371. Houghton, in view of Munson and as per [1d1]-[1d2] (*supra*), would store the received push messages in push server's memory. SAMSUNG-1003, ¶366. In combination, Houghton's push server would include logic for delivering the stored push messages (and the associated data/content) to the push client upon the occurrence of one or more message delivery triggers (*see* [1d3] *supra*). SAMSUNG-1003, ¶¶367-368. Such message delivery would happen using the secure message link established/maintained between the push server and the terminal's push client. SAMSUNG-1007, claim 1 ("push server sends, in response to predetermined trigger events," application data to "a push application" using "client-initiated" permanent connection); SAMSUNG-1003, ¶¶370-371.

(b) *Claim 5*

[5a]-[5b]

Houghton's push client 405 "accept[s] ***data D***" from multiple mobile applications 406 and directs such data ***to push server 401***, for transmission to "***other servers, mobile terminals, push clients and computing devices***." SAMSUNG-1007, 27-28; SAMSUNG-1003, ¶¶372-373.

Because Houghton's push server includes a transport services stack (*see* [1a] *supra*), including API or IP-based connections between the push server and the application server, and includes connections between mobile terminals and the push server, a POSITA would have understood or found obvious that push server's transport services stack would receive, over the secure message link (as described in claim 1), messages including data (upload messages) forwarded by the push client(s) (i.e., respective device link agents) from some of the applications 805 (i.e., device software components). SAMSUNG-1003, ¶¶374-375.

Moreover, as explained above, a POSITA would have understood or found obvious that each such message would identify the particular application server (network element) to which the initiating software application (i.e., device respective software component) requested delivery. SAMSUNG-1003, ¶376.

[5c][6]

As described above ([5a]-[5b], claim 1 *supra*), Houghton's push server (network server system) includes interfaces facilitating communications with network elements, including mobile terminals and application servers, to deliver messages and associated data (content from the upload messages) to one or more application servers (respective identified network element, of which there can be more than one). SAMSUNG-1003, ¶377. Moreover, as described above ([5a]-[5b] *supra*), push server uses an API/IP connection for communications between the push server and the application server, and a POSITA would have understood that this interface/network connection would have been used for communicating upload messages to the application server(s) from the push server. SAMSUNG-1003, ¶¶378-379.

(c) *Claim 6*

As described above ([1b], [1c2] *supra*), Houghton's push messages would be received by the push server's transport services stack (per claim 1) and from different network elements (e.g., application server or other push client resident on a respective mobile terminal), and each such message intended for an application

[6] *See* fn 4.

executing on one of the mobile terminals would include data and the intended application's corresponding identification. SAMSUNG-1003, ¶380.

A POSITA would have recognized or found obvious that messages received from the push server would have been directed to multiple applications from among multiple applications ("***multiple identifier/data pairs***") because Houghton discloses various "actions" that occur across applications when receiving a push message. SAMSUNG-1007, 21; SAMSUNG-1003, ¶381. Houghton, discloses that, in some cases, the "packaging of mobile applications involves combining multiple applications delivered in a single bundle" (e.g., a "bundle"/"suite" of applications would receive messages packaged together – a message containing "***multiple identifier/data pairs***"). SAMSUNG-1007, 12; SAMSUNG-1003, ¶¶381-383. Moreover, duplicating parts has "no patentable significance unless a new and unexpected result is produced." *In re Harza*, 274 F.2d 669.

(d) *Claim 7*

Houghton's terminal includes "software" that "***initiates and maintains contact with the server using Internet technologies***" and this "***mobile-initiated permanent IP connection*** allows the server to" "push messages" to "the mobile terminal." SAMSUNG-1007, 11, claim 1, 25; SAMSUNG-1003, ¶384.

Additionally, as described above ([1a] *supra*), SSL is used for secure messaging between push client and push server, and it was well-known for SSL communications to be client device-initiated. SAMSUNG-1003, ¶385; SAMSUNG-1012, 29:31-30:24.

Because push client logs into the push server and facilitates communication between these entities using a client-established IP connection secured using SSL-protocols, a POSITA would have found obvious that push client (device messaging agent on at least one of the wireless end-user devices) initiates the respective secure Internet Data message link (message link on which data is sent over the Internet) to push server's transport services stack (per [1a] *supra*). SAMSUNG-1003, ¶386.

(e) <u>*Claim 9*</u>

Houghton-Munson leverages Munson's teachings that "content can be pushed according to a schedule of time or event" ("***message delivery trigger***"), with Munson providing an example where a "group push" is performed "during off-peak hours." SAMSUNG-1017, [0044]; SAMSUNG-1003, ¶¶388-389.

Houghton contemplates multiple message delivery triggers (SAMSUNG-1007, 21) and discloses a periodic message (sent upon "expiration of a timer") so that devices in the communication path (push client at mobile terminal and push

server) "do not time expire the connection." *Id.*, 26, 19.

Given Houghton's delivery triggers and Munson's use of periodic timers for message delivery, a POSITA would have been motivated and would have had a reasonable expectation of success in implementing Munson's use of periodic timers for message delivery. SAMSUNG-1003, ¶¶390-392. A POSITA would have been motivated to use periodic timers as a message delivery trigger, per Munson, to enable more efficient use of network resources and avoid repeated message requests that can impact performance of network communications. *Id.*

(f) *Claims 11-12*

Houghton's message delivery triggers include "the receipt of a transmission on the respective secure message link from the device link agent of the given one of the wireless end-user devices," where the transmission is a request received from the given device link agent." SAMSUNG-1003, ¶¶393.

For example, Houghton describes that "client side events" result in "pushing application commands to a mobile terminal." SAMSUNG-1007, 14, 21, 16, Fig. 4; SAMSUNG-1003, ¶394 Houghton's "client" notifies the "server" of a "terminal event" that triggers a "return message." SAMSUNG-1007, 20-21.

Also, per Houghton, a connectionless protocol is used "when the ***client 405 notifies the server 401*** of a[] terminal event, user interface event or application

event" and the ***push server then "push[es] to the client 405 a service triggered by the event***…. The sending of such a return message is frequently time critical and the fastest available combination of techniques will be used." SAMSUNG-1007, 21-22; SAMSUNG-1003, ¶¶395-396

For established IP connections, push client 405 "monitors the local resources for ***a state change (a trigger event)***" and if so, "***send[s] a message to the server 401***." SAMSUNG-1007, 26; SAMSUNG-1003, ¶397.

Moreover, as described above (claim 9 *supra*), an example message trigger includes periodic polling so that devices in the communication path (push client at mobile terminal and push server) "do not time expire the connection," which a POSITA would have recognized or found obvious as a heartbeat message generated by the given device link agent. SAMSUNG-1007, 19, 21, 26; SAMSUNG-1003, ¶398; SAMSUNG-1001, 70:6-71:16.

(g) <u>*Claim 13*</u>

Houghton-Munson renders obvious that a message delivery trigger is the receipt of a particular network element message from one of the network elements (particular application server). SAMSUNG-1007, 21-22, FIG. 7 ("***push server 701 is triggered by a trigger event … from an application server 702 to push an appli-***

cation command message to the push client 704 and thereby initiate a mobile terminal client trigger event"); SAMSUNG-1003, ¶¶400-401.

F. Ground 6 Claims Are Rendered Obvious by Houghton, Munson, and Shen

1. Combination of Houghton, Munson, and Shen

Like Houghton and Munson, Shen discloses a push messaging system wherein a push server (MMS Relay/Server) receives and stores messages before transmitting them to MMS User Agents. SAMSUNG-1004,16-18, 54-56; SAMSUNG-1014, [0029]-[00034], FIGS. 1-5; SAMSUNG-1003, ¶¶402-403. However, such store and forward functionality exposes a "potential security problem" because such systems are "not an end-to-end solution." SAMSUNG-1014, [0004]; SAMSUNG-1003, ¶403.

To overcome this "problem," Shen discloses generating a symmetric key used by the MMS Relay/Server to encrypt data communications transmitted to the MMS User Device/Agent, with the same key being used to decrypt communications at the device. SAMSUNG-1014, [0056]-[0060], FIG. 5; SAMSUNG-1003, ¶¶404-405; *see* §III.B.1-2 *supra*; *see* SAMSUNG-1009, ¶¶[0054]-[0060].

Combining Shen's teachings with Houghton-Munson would have been obvious and would have involved (1) combining prior art elements according to known

methods to yield predictable results and (2) using known techniques to improve similar devices/methods/products in the same way. SAMSUNG-1003, ¶406. Houghton, Munson, and Shen provide similar push messaging architectures involving communications between push clients/MMS User Agents and push servers/MMS Relay/Server. SAMSUNG-1003, ¶407. Given data "security problems" in such environments, a POSITA would have been motivated to implement known techniques (per Shen) to solve these problems—implementing an "authentication and key management module 502" at Houghton's push server that would generate and "distribute symmetric keys" and would use this symmetric key to encrypt data transmitted to "the user device." SAMSUNG-1014, [0054]-[0060]; SAMSUNG-1003, ¶408.

A POSITA would have been motivated to so combine the references' teachings to achieve a push messaging system "having improved security," resulting from providing "an end-to-end security solution for" the messaging applications, which is particularly beneficial for "enterprise applications" that are contemplated by Houghton's system. SAMSUNG-1014, [0017], [0021] (describing additional advantages); SAMSUNG-1007, 22 ; SAMSUNG-1003, ¶¶409-410.

Additionally, because Houghton, Shen, and Munson describe message delivery in similar push messaging environments, a POSITA would have found it

straightforward to modify/program Houghton's push server and push client to implement Shen's above-described modules and functionality for generating an encryption key and encrypting/decrypting push messages using that key. SAMSUNG-1003, ¶¶410-411. The resulting system's elements (referred as Houghton-Munson-Shen) would perform functions they performed prior to the combination—i.e., Houghton-Munson's system would enable communicating push messages between push servers and push clients, and Shen's teachings (in combination) would enable encrypting messages sent to a push client. SAMSUNG-1003, ¶¶411-413.

2. Analysis

(a) *Claim 2*

As described above ([1a], [1d4] *supra*), Houghton's push server would supply messages to its transport services stack for delivery on the secure message link between the push server and push client (resident on a device/terminal). SAMSUNG-1003, ¶¶414-415.

Additionally, as described above (§III.F.1 *supra*), Houghton-Munson-Shen would have implemented Shen's teachings of authentication and key management module at Houghton's push server, which would have facilitated generating the shared encryption key between the mobile terminal/push client and the push server that is used to encrypt/decrypt messages. SAMSUNG-1014, [0059]; SAMSUNG-

1003, ¶¶416-417. Houghton-Munson-Shen therefore would have implemented an encryption function that encrypts messages sent by the push server to the push client using the shared keys (sent via the secure message link maintained therebetween, per claim 1 above). *Id.*

(b) *Claim 3*

As described above (claim 2 *supra*), Houghton-Munson-Shen renders obvious encrypting messages and transporting them to a terminal's push client. SAMSUNG-1003, ¶¶419-421. As explained above ([1a] *supra*), Houghton-Munson-Shen renders obvious using encryption on the transport services stack (e.g., SSL and HTTPS) and IP layer encryption (IPSec), as described above, for transmitting messages to push client. *Id.*

G. Ground 7: Claim 4 is Rendered Obvious by Houghton, Munson, and Ellison

1. Combination of Houghton, Munson, and Ellison

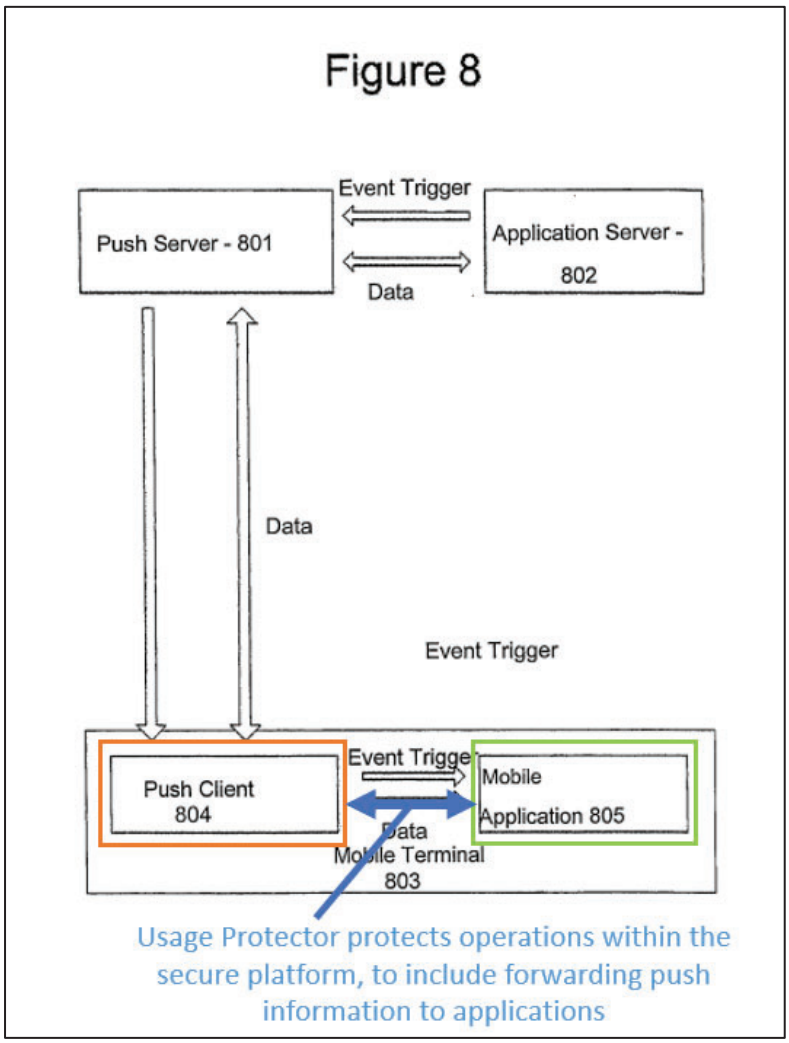
A POSITA would have found obvious to implement Ellison's secure platform within mobile terminals in Houghton-Munson. SAMSUNG-1007, FIGS. 3-4, 8; SAMSUNG-1019, 8:25-32, 8:66-9:6, 9:28-62, FIG. 2; SAMSUNG-1003, ¶422. A POSITA would have been motivated to combine these teachings to improve terminal security and for reasons described above (§III.C.2 *supra*). SAMSUNG-1003, ¶¶423-425 (improved data security at device); SAMSUNG-1019, 8: 8:25-32,

8:66-9:6, 9:28-62, FIG. 2.

Moreover, in mobile devices and their associated operating systems, it was well-known to implement more secure systems, with rings/tiers of protection around different device components. SAMSUNG-1003, ¶¶426, 35-40; SAMSUNG-1035, 2-4. In such systems, messaging services and clients/agents (e.g., push clients) responsible for providing those services, would be located within the trusted computing environment, and applications would reside outside this environment. *Id.*

Therefore, incorporating Ellison's security techniques into Houghton's mobile terminals would also have been predictable and foreseeable with a reasonable expectation of success because (1) Houghton contemplates security mechanisms to secure its messaging systems, (2) Ellison's techniques can be implemented in "computer system[s]" including a "processor" (as present in Houghton's client/server devices), and (3) it was well-known to use protection rings and tiers in computing environments (per Ellison). SAMSUNG-1019, 5:11-16; SAMSUNG-1003, ¶¶427-429, ¶¶33-40 (citing SAMSUNG-1030, -1031, -1032, -1033).

In an example of the combined system (below), Ellison's secure platform would have been incorporated into Houghton's mobile terminals. SAMSUNG-1019, 8:25-9:62, FIG. 2; SAMSUNG-1003, ¶¶430-434.



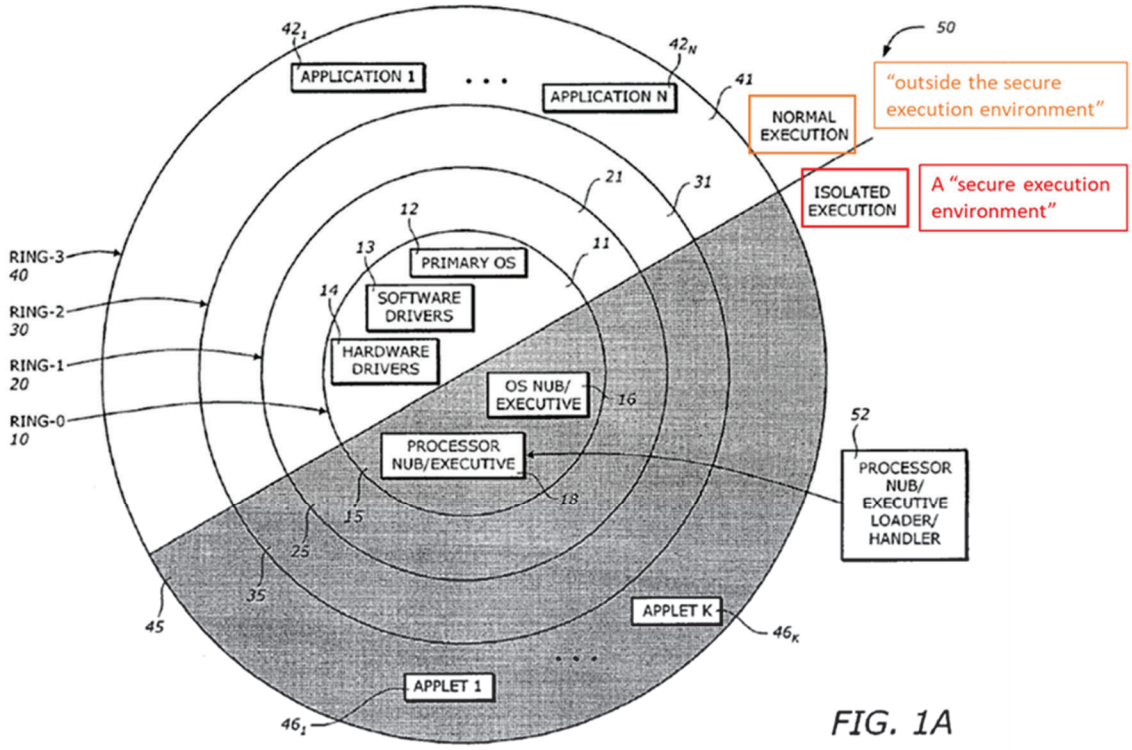
SAMSUNG-1007, FIG. 8 (modified to incorporate Ellison's teachings);

SAMSUNG-1003, ¶430-434.

2. Analysis

As in §III.C.I and as explained above in §III.G.1, in combination, Ellison's usage protector and secure platform, which operate within the secure, isolated area (secure execution environment) and coordinate with applications "outside" this environment—SAMSUNG-1019, 4-65-5:1, 6:1-26, 8:25-9:67, FIGS. 1A-1C, 2—

would be incorporated into Houghton's push client, which would provide secure access to applications outside the secure execution environment. SAMSUNG-1003, ¶¶435-439.



SAMSUNG-1019, FIG. 1A.

In doing so, the combined system would enable securely transmitting push messages to the terminal's applications, and preventing malicious activity. *Id.*; SAMSUNG-1019, 8:25-9:62, FIG. 2; *see supra* §III.G.1; SAMSUNG-1003, ¶¶440-441.

H. Ground 8: Claim 8 Is Rendered Obvious by Houghton, Munson, and Rakic

1. Analysis

As described above ([1b] *supra*), a POSITA would have found obvious that the mobile applications would register with the push client and/or the push server. SAMSUNG-1003, ¶443. Thus, when a registered application is a message's recipient, it is routed to the application based on information in the push message "specifying" the particular mobile application 406 as the recipient. SAMSUNG-1007, 21. Thus, a POSITA would have found obvious that the push client includes data indicating the authorized/registered applications (authorized software components) to receive data from secure message link message via the push server (message link server). SAMSUNG-1003, ¶¶444-445.

While a POSITA would have found obvious for the destination application identifier to be present in the received push message (as described above – [1b] *supra*), Houghton-Munson does not expressly describe validation/authentication of the message. SAMSUNG-1003, ¶446; SAMSUNG-1004, 14. Absent such validation/authentication, the device and/or the destination application can be subject to data and security vulnerabilities. SAMSUNG-1003, ¶447; *see* §III.D.2 *supra*.

In the context of push environments, Rakic provides one well-known solution

to address these security concerns by using electronic signatures to validate/authenticate the sender and intended message recipient. SAMSUNG-1003, ¶¶448-449; SAMSUNG-1015, [0047]-[0049], [0065]-[0067], [0109]-[0112]; *see supra* §III.D.1-2.

A POSITA would have been motivated to implement Rakic's techniques within Houghton's push server to address the above-described security problems. SAMSUNG-1003, ¶450. Indeed, a POSITA would have been motivated to do so because Rakic's techniques enable (1) the client device to "authenticate the sender" and (2) "verify that an intended recipient of the secure push message is the client device, and that the client device includes a correct component." SAMSUNG-1015, [0041]; SAMSUNG-1003, ¶451. A POSITA would have therefore recognized data and computer improvements for the client device and the underlying applications receiving the messages. *Id.*; *see* §III.D.1-2 *supra*.

A POSITA would have therefore implemented Rakic's secure push message server 504 and database device 108, and their above-described functionality, within Houghton's push server. SAMSUNG-1003, ¶¶451-453. Configuring Houghton-Munson to implement Rakic's teachings would have amounted to using a known technique to improve similar devices in the same way, and combining prior art elements according to known methods to yield predictable results. *Id.*

Because Houghton, Munson, and Rakic contemplate similar push messaging

architectures, a POSITA would have found it straightforward to implement a secure server, including Rakic's secure push messaging server and its signature generation, and its database storing device information, within Houghton's push server. *Id.* Additionally, the resulting system's elements would perform functions they performed prior to combination—i.e., Houghton-Munson's system enables message communication between push server and push clients, and Rakic's teachings (in combination) would provide secure signatures (with the messages) to the message receiving device(s). *Id.*

In the resulting combination, Rakic's secure push message server and its functionality would be implemented in push server and used to generate signatures for messages intended for mobile terminals and their respective push clients. SAMSUNG-1003, ¶454. As described in §III.D.2, a POSITA would have understood that Rakic's signatures are authorization signatures. SAMSUNG-1015, [0041]; SAMSUNG-1003, ¶¶455-456; *see* §III.D.2 *supra*.

Additionally, Rakic's signature generating functionality "verif[ies] that an intended recipient of the secure push message is the client device, and that the client device includes a correct component." *See id.*

A POSITA would have found obvious that such component would be a device application. SAMSUNG-1003, ¶457; *see* §III.D.2 *supra*. Thus, Houghton-Munson-

Rakic renders obvious the above-described secure authorization signatures that indicates the authorized application(s) (software component(s)) allowed/authorized to receive messages. SAMSUNG-1003, ¶457; ¶¶442-456.

IV. PTAB DISCRETION SHOULD NOT PRECLUDE INSTITUTION

A. §325(d)

None of the same/substantially the same references advanced herein and/or arguments related thereto were previously before the Office. *See generally* SAMSUNG-1002, SAMSUNG-1003, ¶¶49-62. During prosecution, the pending independent claims were rejected over Heinonen and Zhou. SAMSUNG-1002, 343-355; SAMSUNG-1003, ¶49-50. In response, Applicant argued that Heinonen failed to disclose claim features ([1pre], [1a], [1b], [1c1], and [1d1]-[1d3]), including a message link server, a secure message link between this server and device link agent(s), an interface for network elements, multiple software components authorized to receive messages via the device link agent, and a message buffering system. SAMSUNG-1002, 81-83; SAMSUNG-1003, ¶¶51-56. Zhou, per Applicant, did not overcome these purported deficiencies because its system operates in the same operating system and does not disclose a message link server nor address Internet communications between server and network devices/elements. SAMSUNG-1002, 83-86;

SAMSUNG-1003, ¶57. Also, neither reference, per Applicant, discloses "as a trigger, the occurrence of an asynchronous event with time-critical messaging needs." SAMSUNG-1002, 84; *see id.*, 84-86 (providing arguments for dependent claims); SAMSUNG-1003, ¶¶58-60. The Examiner allowed the claims based on Applicant's arguments for "independent claims." SAMSUNG-1002, 13-19; SAMSUNG-1003, ¶¶60-61.

In contrast, the combinations/grounds presented herein provide push messaging/MMS messaging systems disclosing/rendering obvious the above-referenced claim features and the delivery triggers that were alleged to be absent from the record art. *See* §§III.A-H *supra*. Moreover, material error occurred during prosecution because Examiner failed to consider messaging systems of the presented grounds, and how they rendered obvious every claim feature (including those alleged to be absent from the record). As highlighted in Grounds 1 and 5, the Examiner did not appreciate how the grounds/combinations presented herein, which were not before the Office, render obvious the independent claims and the other Challenged Claims. Indeed, Petitioner has shown a reasonable likelihood that at least one of the Challenged Claims is unpatentable over the applied art on the current record. *Supra* §IIIA-H; *see Tokyo Ohka Kogyo Co., Ltd. v. Fujifilm Elec. Materials U.S.A., Inc.*, PGR2022-00010, Paper 9, 8-9 (PTAB June 6, 2022). Therefore, §325(d) discretionary denial is not warranted.

B. §314(a)

This Petition's merits are compelling, and the evidence presented herein is substantial, counseling against discretionary denial under *Fintiv*. SAMSUNG-1020, 4-5. Moreover, the *Fintiv* factors counsel against denial.

Factor 1 is neutral because neither party has requested a stay in the co-pending litigation.

Factor 2 is neutral because the Court's trial date is speculative and subject to change. The Board will likely issue its Final Written Decision around May/June 2025, approximately 4-5 months after the currently-scheduled trial date (January 6, 2025). SAMSUNG-1021, 2. However, as the Board/Director have recognized, "scheduled trial dates are unreliable and often change." SAMSUNG-1020, 8.

Factor 3 favors institution because Petitioner has diligently filed this Petition months ahead of the one-year time bar, while the EDTX Litigation is in its early stages. Beyond exchanging preliminary infringement, the parties and the District Court have yet to expend significant resources on invalidity. SAMSUNG-1021. By the anticipated institution deadline in May/June 2024, the co-pending litigation will still be in early stages—fact and expert discovery will be ongoing, and the *Markman* hearing will not have occurred. *Id.*

Factor 4 favors institution because Petitioner stipulates to not pursuing the IPR grounds in the co-pending litigation. SAMSUNG-1023. Thus, institution

serves "efficiency and integrity goals" by "not duplicating efforts" and "resolving materially different patentability issues." *Apple, Inc. v. SEVEN Networks, LLC*, IPR2020-00156, Paper 10, 19 (June 15, 2020); *Sand Revolution II, LLC v. Continental Intermodal Group-Trucking LLC*, IPR2019,-01393, Paper 24, 12 (June 16, 2020); *Google LLC v. Flypsi, Inc.*, IPR2023-00360, Paper 9, 36-39 (August 2, 2023).

Factor 5: Same parties are in the co-pending litigation.

Factor 6 favors institution because this Petition's merits are compelling, as described herein.

V. CONCLUSION AND FEES

The Challenged Claims are unpatentable. Please charge fees to Deposit Account 06-1050.

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

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

Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc. (collectively, "Samsung") are the real parties-in-interest.

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

The '192 Patent is the subject of a civil action, *Headwater Research LLC v. Samsung Electronics Co., Ltd. et al.*, 2:23-cv-00103, E.D. Tex., filed March 10,

2023 (SAMSUNG-1029) and served on March 14, 2023 (SAMSUNG-1037). Petitioner and Headwater are also involved in case nos. 2:22-cv-00422 and 2:22-cv-00467, also in E.D. Tex. Petitioner is not aware of any other disclaimers, reexamination certificates, or IPR petitions addressing the '192 Patent.

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

Petitioner provides the following designation of counsel.

Lead Counsel	Backup counsel
W. Karl Renner, Reg. No. 41,265 Fish & Richardson P.C. 60 South Sixth Street, Suite 3200 Minneapolis, MN 55402 Tel: 202-783-5070 Fax: 877-769-7945 Email:IPR39843-0166IP1@fr.com	Jeremy J. Monaldo, Reg. No. 58,680 Karan Jhurani, Reg. No. 71,777 60 South Sixth Street, Suite 3200 Minneapolis, MN 55402 Tel: 202-783-5070 Fax: 877-769-7945 PTABInbound@fr.com

D. Service Information

Please address all correspondence and service to the address listed above. Petitioner consents to electronic service by email at IPR39843-0166IP1@fr.com (referencing No. 39843-0166IP1 and cc'ing PTABInbound@fr.com, axf-ptab@fr.com, jjm@fr.com, and jhurani@fr.com).

Respectfully submitted,

Dated November 17, 2023

/Karan Jhurani/

W. Karl Renner, Reg. No. 41,265
Jeremy J. Monaldo, Reg. No. 58,680
Karan Jhurani, Reg. No. 71,777
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
T: 202-783-5070
F: 877-769-7945

(Control No. IPR2024-00010)

Attorneys for Petitioner

CERTIFICATION UNDER 37 CFR § 42.24

Under the provisions of 37 CFR § 42.24(d), the undersigned hereby certifies that the word count for the foregoing Petition for *Inter Partes* Review totals 13,993 words, which is less than the 14,000 allowed under 37 CFR § 42.24.

Dated November 17, 2023

/Karan Jhurani/

W. Karl Renner, Reg. No. 41,265
Jeremy J. Monaldo, Reg. No. 58,680
Karan Jhurani, Reg. No. 71,777
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
T: 202-783-5070
F: 877-769-7945

Attorneys for Petitioner

CERTIFICATE OF SERVICE

Pursuant to 37 CFR §§ 42.6(e)(4)(i) *et seq.* and 42.105(b), the undersigned certifies that on November 17, 2023, a complete and entire copy of this Petition for *Inter Partes* Review and all supporting exhibits were provided by Federal Express, to the Patent Owner, by serving the correspondence address of record as follows:

Headwater Research LLC
C/O Farjami & Farjami LLP
26522 La Alameda Ave., Suite 360
Mission Viejo, CA 92691

/Diana Bradley/

Diana Bradley
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
(858) 678-5667