

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of: Gregory G. Raleigh, et al. Attorney Docket No. 39843-0191IP1
U.S. Patent No.: 10,080,250
Issue Date: September 18, 2018
Appl. Serial No.: 15/287,603
Filing Date: October 6, 2016
Title: ENTERPRISE ACCESS CONTROL AND ACCOUNTING ALLO-
CATION FOR ACCESS NETWORKS

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. 10,080,250 PURSUANT TO 35 U.S.C. §§ 311–319,
37 C.F.R. § 42

TABLE OF CONTENTS

I.	REQUIREMENTS FOR IPR	1
A.	Grounds for Standing.....	1
B.	Challenge and Relief Requested.....	1
C.	Claim Construction	2
II.	THE '250 PATENT	3
A.	Brief Description.....	3
B.	Prosecution History.....	4
III.	THE CHALLENGED CLAIMS ARE UNPATENTABLE.....	4
A.	Ground 1A: Owen-Bartlett renders claims 1-2 obvious.....	4
1.	Owen-Bartlett	4
(a)	Owen.....	4
(b)	Bartlett	9
(c)	Combination	12
2.	Analysis	18
B.	Ground 1B: Owen-Bartlett-Smith renders claims 1-2 obvious	40
1.	Owen-Bartlett-Smith	40
(a)	Smith.....	40
(b)	Combination	44
2.	Analysis	51
C.	Ground 2: Lambert-Rao-Smith renders claims 1-2 obvious	55
1.	Lambert-Rao-Smith.....	55
(a)	Lambert.....	55
(b)	Rao.....	59
(c)	Combination	63
2.	Analysis	76
IV.	INSTITUTION SHOULD NOT BE DENIED ON DISCRETION.....	96
V.	CONCLUSION AND FEES	97
VI.	MANDATORY NOTICES UNDER 37 C.F.R. § 42.8(a)(1).....	97
A.	Real Party-In-Interest Under 37 C.F.R. § 42.8(b)(1).....	97
B.	Related Matters Under 37 C.F.R. § 42.8(b)(2)	97
C.	Lead And Back-Up Counsel Under 37 C.F.R. § 42.8(b)(3).....	97
D.	Service Information	97

EXHIBITS

SAMSUNG-1001	U.S. Patent No. 10,080,250 to Gregory G. Raleigh, et al. (“the ’250 Patent”)
SAMSUNG-1002	Excerpts from the Prosecution History of the ’250 Patent (“the Prosecution History”)
SAMSUNG-1003	Declaration and Curriculum Vitae of Dr. Patrick McDaniel.
SAMSUNG-1004	Complaint, Headwater Research LLC v. Samsung Electronics Co., Ltd. et al, 2-24-cv-00627, E.D. Tex., filed August 2, 2024
SAMSUNG-1005	U.S. Pat. App. Pub. No. 2005/0213763 (“Owen”)
SAMSUNG-1006	U.S. Pat. App. Pub. No. 2003/0177396 (“Bartlett”)
SAMSUNG-1007	U.S. Pat. App. Pub. No. 2002/0099952 (“Lambert”)
SAMSUNG-1008	(Excerpts) Smith, et al., 2005. “Virtual Machines: Versatile Platforms for Systems and Processes,” Elsevier, Inc, 2005, ISBN 1-55860-910-5 (“Smith”)
SAMSUNG-1009	U.S. Pat. App. Pub. No. 2006/0039354 (“Rao”)
SAMSUNG-1010	Interim Process for PTAB Workload Management
SAMSUNG-1011	U.S. Pat. App. Pub. No. 2009/0132998 (“Meijer”)
SAMSUNG-1012	Guérin, et al., "Quality-of-service in packet networks: basic mechanisms and directions." Computer networks 31, no. 3 (1999): 169-189 (“Guérin”)
SAMSUNG-1013	Goldberg, R.P., 1974. Survey of virtual machine research. Computer, 7(6), pp.34-45. (“Goldberg”)
SAMSUNG-1014	U.S. Pat. App. Pub. No. 2003/0182431 (“Sturniolo”)

SAMSUNG-1015	Kent, et al., IETF RFC 4301, “Security Architecture for the Internet Protocol,” downloaded from the Internet on April 17, 2025 (“Kent”)
SAMSUNG-1016	Declaration of June Munford
SAMSUNG-1017	U.S. Pat. App. Pub. No. 2005/0188208 (“Day”)
SAMSUNG-1018	Patent Trial and Appeal Board (PTAB) Boardside Chat: Interim processes relating to institution in AIA proceedings, downloaded from https://www.uspto.gov/sites/default/files/documents/boardside_chat_interim_process_for_aia_institution_decisions_.pdf on April 25, 2025
SAMSUNG-1019	Samsung Stipulation Letter
SAMSUNG-1020	Stone, G.N., Lundy, B. and Xie, G.G., 2001. Network policy languages: a survey and a new approach. IEEE network, 15(1), pp.10-21. (“Stone”)
SAMSUNG-1021	Verma, et. al., (2002). Policy-based management of content distribution networks. IEEE network, 16(2), 34-39. (“Verma”)
SAMSUNG-1022	Lobo, et. al., (1999). A policy description language. AAAI/IAAI, 1999, 291-298. (“Lobo”)
SAMSUNG-1023	Westerinen, et al., IETF RFC 3198, Terminology for Policy-Based Management, November 2001, downloaded from the Internet on May 27, 2024.
SAMSUNG-1024	Gonçalves, et al., 2009, October. A graphical user interface for policy composition in CIM-SPL. In 2009 International Conference on Ultra Modern Telecommunications & Workshops (pp. 1-7). IEEE. (“Gonçalves”)
SAMSUNG-1025	Agrawal, et. al., 2007, May. Issues in designing a policy language for distributed management of IT infrastructures. In 2007 10th IFIP/IEEE International Symposium on Integrated Network Management (pp. 30-39). IEEE. (“Agrawal”)

- SAMSUNG-1026 eXtensible Access Control Markup Language (XACML) Version 2.0, Committee draft 02, 30 Sep 2004, downloaded from the Internet at <https://xml.coverpages.org/XACMLv20CD-CoreSpec.pdf> on April 16, 2025 (“XACML”)
- SAMSUNG-1027 (Excerpts) The IBM System 370, downloaded from the Internet at <https://www.ibm.com/history/system-370> on April 16, 2025 (“System 370”)
- SAMSUNG-1028 Black, et al., IETF RFC 2983, Differentiated Services and Tunnels, October 2000, downloaded from the Internet on April 21, 2025.
- SAMSUNG-1029 Gleeson, et al., IETF RFC 2764, A Framework for IP Based Virtual Private Networks, February 2000, downloaded from the Internet on April 21, 2025 (“Gleeson”)
- SAMSUNG-1030 Blake, et al., IETF RFC 2475, An Architecture for Differentiated Services, December 1998, downloaded from the Internet on April 24, 2025.

CLAIMS

Claim	Identifier	Claim Language
1	[1.1]	A method of operating a wireless end-user device, the method comprising:
	[1.2.i]	storing secure policy information
	[1.2.ii]	indicative of an application-specific execution environment policy and
	[1.2.iii]	a configurable communication security policy;
	[1.3.i]	based on the secure stored information indicative of an application-specific execution environment policy, allowing a first set of multiple device application instances to run only in a first secure application environment on the wireless end-user device, and
	[1.3.ii]	allowing a second set of device application instances to run only outside of the first secure application environment; and
	[1.4.i]	based on the secure stored information indicative of a configurable communication security policy, initiate a first secure tunnel to a network tunnel endpoint, and
	[1.4.ii]	direct all network data communications to and from device application instances executing in the first secure application environment through the first secure tunnel.
2	[2.1]	The method of claim 1, wherein the configurable communication security policy defines a first configuration, which indicates that application instances in the second set of device application instances are not allowed to use the first secure tunnel, the method further comprising,
	[2.2]	based on the secure stored information indicative of a configurable communication security policy indicating the first configuration, directing all network data communications to and

		from device application instances executing outside of the first secure application environment through a network path outside of the first secure tunnel.
--	--	--

Samsung Electronics Co., Ltd (“Petitioner” or “Samsung”) petitions for *Inter Partes* Review (“IPR”) of claims 1-2 (“the Challenged Claims”) of U.S. Patent No. 10,080,250 (“the ’250 Patent”). Compelling evidence presented in this Petition demonstrates at least a reasonable likelihood that Samsung will prevail with respect to at least one of the Challenged Claims.

I. REQUIREMENTS FOR IPR

A. Grounds for Standing

Petitioner certifies that the ’250 Patent is available for IPR. This petition is being filed within one year of service of a complaint against Samsung. Samsung is not barred or estopped from requesting review of the Challenged Claims on the below-identified grounds.

B. Challenge and Relief Requested

Samsung requests IPR of the Challenged Claims on the grounds listed below. Dr. McDaniel provides supporting testimony in his Declaration. SAM-SUNG-1003, ¶¶1-255.

Ground	Claim(s)	35 U.S.C. §103
1A	1-2	Owen-Bartlett
1B	1-2	Owen-Bartlett-Smith
2	1-2	Lambert-Rao-Smith

Reference	Filing Date	Publication Date
Owen	8/19/2003	8/19/2003
Bartlett	1/28/2003	9/18/2003
Smith		2005 (<i>see</i> SAM-SUNG-1016)
Lambert	6/8/2001	7/24/2002
Rao	7/22/2005	2/23/2006

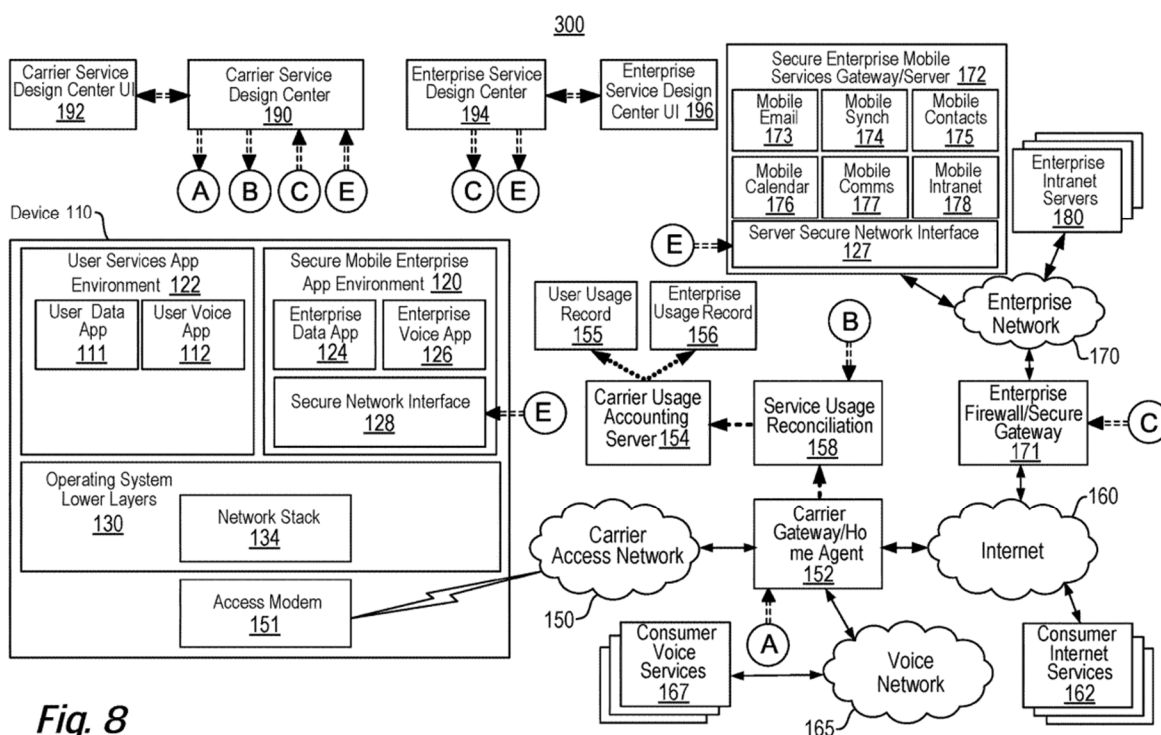
C. Claim Construction

Petitioner submits that no formal 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). Petitioner reserves the right to respond to any constructions offered by Patent Owner or adopted by the Board. Petitioner is not conceding that each challenged claim satisfies all statutory requirements, nor waiving arguments/grounds that cannot be raised in this proceeding. Petitioner applies prior art consistent with Patent Owner’s allegations of infringement before the district court. SAMSUNG-1004, *generally*.

II. THE '250 PATENT

A. Brief Description

The '250 Patent is directed to “[e]nterprise and consumer billing allocation for wireless communication device service usage activities,” which is determined by “using a classifier implemented on the wireless communication device that classifies the monitored service usage activity.” SAMSUNG-1001, Abstract, 10:21-31. A “secure mobile enterprise application environment 120 and secure network interface 128 within secure mobile enterprise application environment 120 are included in device 110.” SAMSUNG-1001, 25:60-63.



SAMSUNG-1001, FIG. 8

B. Prosecution History

The application that issued as the '250 Patent was allowed without any prior art rejection. SAMSUNG-1002, *generally*.

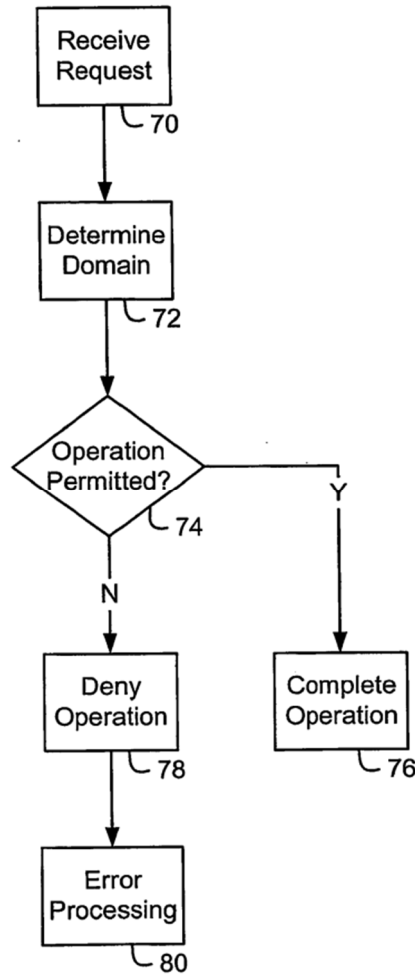
III. THE CHALLENGED CLAIMS ARE UNPATENTABLE

A. Ground 1A: Owen-Bartlett renders claims 1-2 obvious

1. Owen-Bartlett

(a) Owen

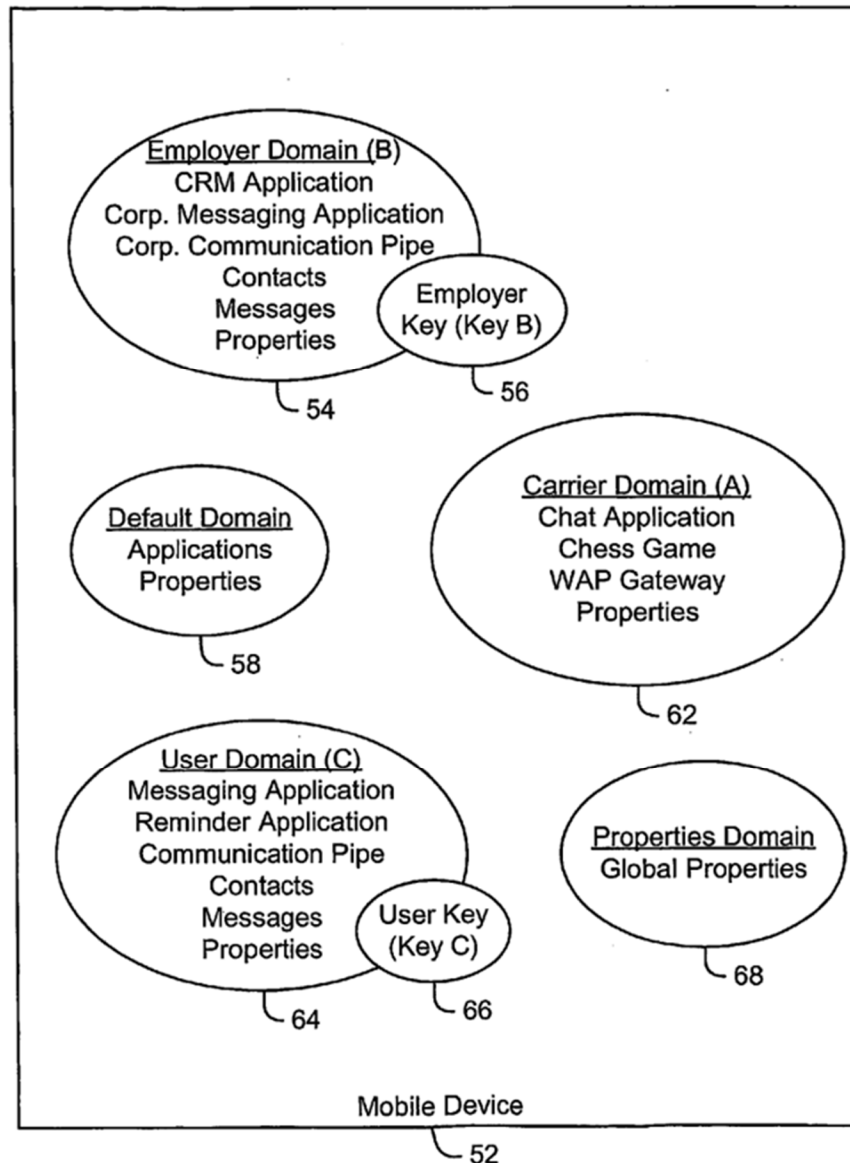
Owen discloses “secure control of a wireless mobile communication device,” in which assets, e.g., software applications, are “assigned to domains” and operations on the assets are “completed where it is permitted by the domain.” SAMSUNG-1005, Abstract, [0029]. Owen illustrates the process in FIG. 4:



SAMSUNG-1005, FIG. 4

In Owen, a “domain is a collection of objects that share a common level of trust,” and “[p]lacing an object in a domain means that the object is trusted, and that its use can be restricted to other domain members.” SAMSUNG-1005, [0030]. Owen’s “domain policies” control “which objects can become members of a domain.” SAMSUNG-1005, [0030]. For example, a “birthday reminder program in the user domain cannot access the company directory in the employer domain, even though they are both stored on the same mobile device.” SAMSUNG-1005,

[0035]. Such restrictions provide “important mobile device control and security benefits.” SAMSUNG-1005, [0030]; *see id.*, [0032]. FIG. 3, reproduced below, shows an employer domain, a user domain, a carrier domain.



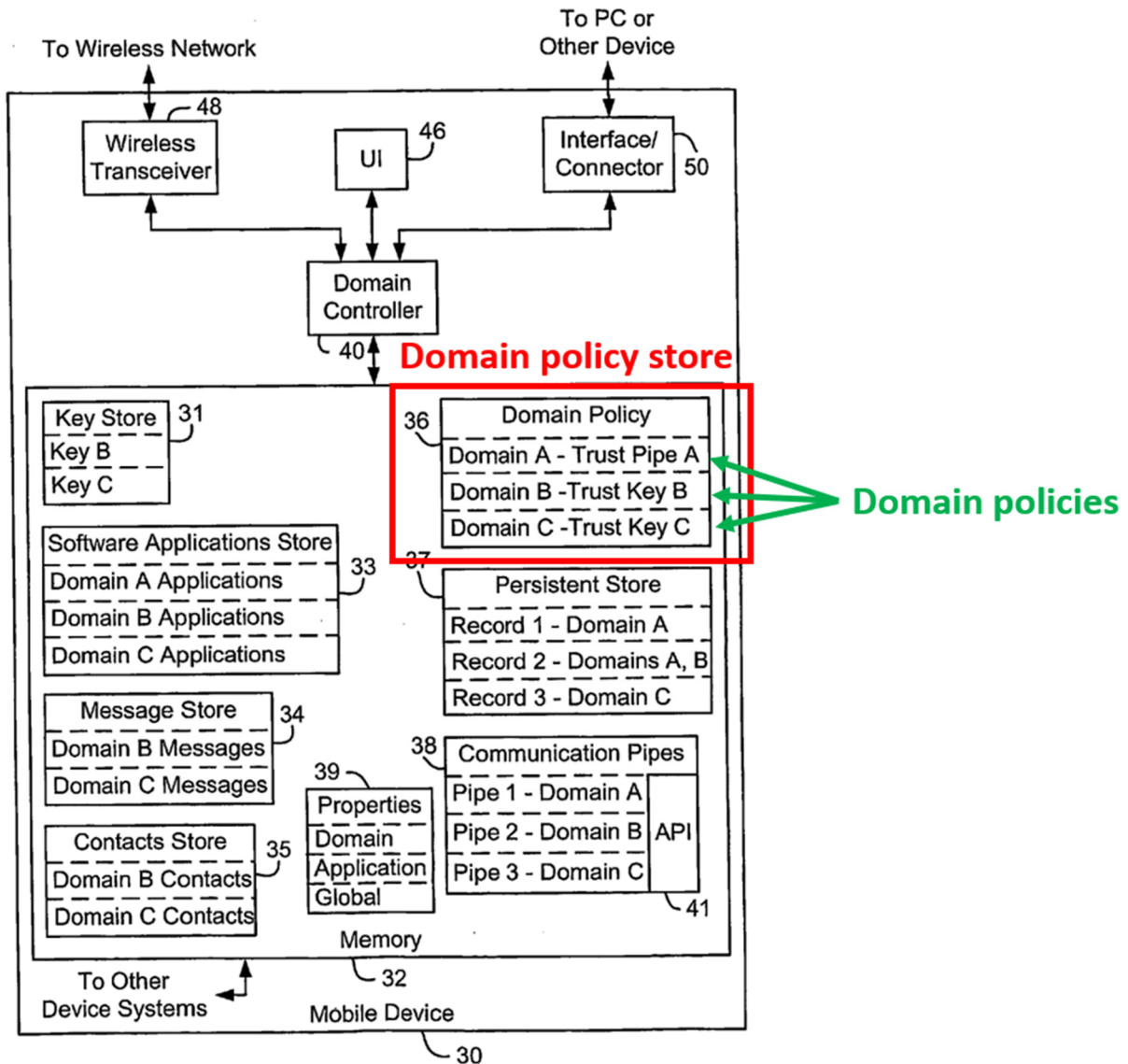
SAMSUNG-1005, FIG. 3

Owen’s domain policies further control access to communication pipes, which are “means of communication between the mobile device and some external

entity.” SAMSUNG-1005, [0042]; *see id.*, [0045].

Owen explains that “[p]lacing a software application in a domain gives it access to domain assets, most notably the domain communication pipe(s).” SAMSUNG-1005, [0042]. Owen further explains that “[p]lacing a communication pipe in a domain means that the domain owner trusts that communication pipe.” SAMSUNG-1005, [0046].

In Owen, assets such as software applications are assigned to domains using “domain policies” that “specify the criteria used to determine into which domain a software application, property, or other information should be placed.” SAMSUNG-1005, [0024]. In some alternatives, “when a domain is created, it may include an access list established by the domain owner that describes all allowable members,” which “works well with software applications.” SAMSUNG-1005, [0100].



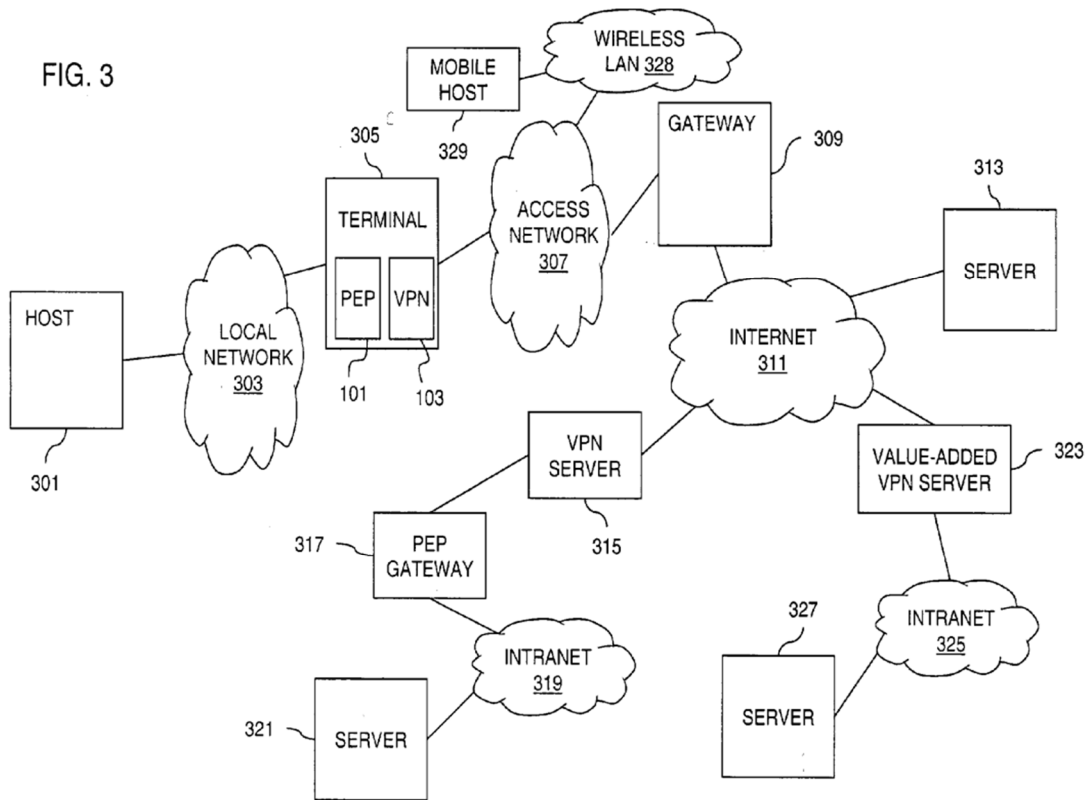
SAMSUNG-1005, FIG. 2 (annotated)

In Owen, domain policies are configurable. SAMSUNG-1005, [0032]-[0034]. For example, the “user domain may also have user-configurable settings, such as a domain backup frequency” and the “employer domain configuration is established by the employer.” SAMSUNG-1005, [0033]-[0034].

(b) Bartlett

Bartlett describes an “approach for adaptively providing network performance enhancing functions in a secure environment, such as a virtual private network.” SAMSUNG-1006, Abstract. In Bartlett, when network traffic “is received for transport over an access network,” then characteristics of “the access network are determined,” and a connection “is selectively established based on the determined characteristics for transport[ing] the received packets over the access network.” *Id.* “An encrypted tunnel is provided over the established connection to transmit the received packets.” *Id.*

Bartlett provides “an approach for integrating Virtual Private Network (VPN) and Performance Enhancing Proxying (PEP) functionalities, such that the PEP functions are adaptively applied.” SAMSUNG-1006, [0011]. “The PEP functions” can be “automatically tuned based on the characteristics (e.g., latency) of a particular network.” *Id.* The “VPN and PEP functionalities” can be integrated into a “host,” e.g., a “user’s PC (Personal Computer) or other client, such as a Personal Digital Assistant (PDA))” connected over a LAN or wireless LAN. SAMSUNG-1006, [0050]-[0051], [0055], [0152].

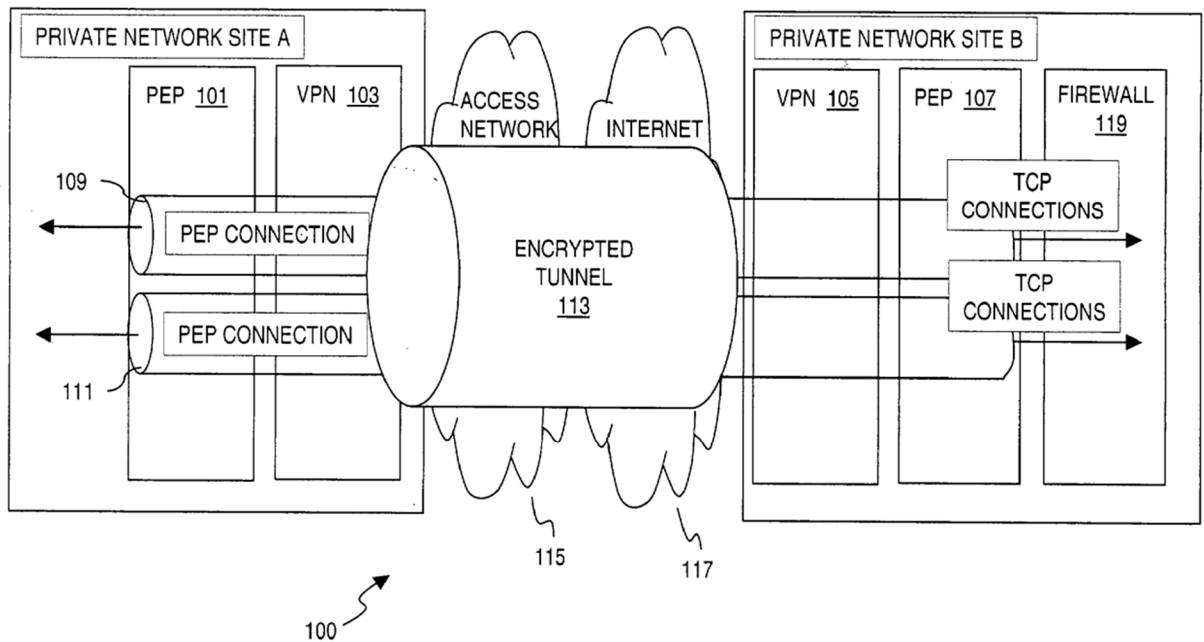


SAMSUNG-1006, FIG. 3

Bartlett explains that “integration of the PEP function and the VPN function provides several options for configuring the method used to establish the PEP and VPN connections.” SAMSUNG-1006, [0057]. One method “involves the user explicitly requesting that the VPN connection be established,” and in an alternative method, “the user generat[es] traffic to automatically invoke the PEP function as to trigger the creation of the VPN connection and the PEP connection.” SAMSUNG-1006, [0057]-[0058].

Bartlett explains that the PEP functionality “can be implemented at the private network site ‘outside’ of the VPN tunnel.” SAMSUNG-1006, [0041]. The

“VPN peers 103, 105 ‘tunnel’ the PEP connections 109, 111 by establishing an encrypted tunnel 113 (e.g., VPN tunnel) to encrypt the private network packets carried by the PEP connections 109, 111 over the access network 115 and a public network 117 (e.g., the Internet),” so that the “packets are completely protected in that they only appear in the clear within the customer's premises (at the private network sites A and B), and thus are not viewable by any network element between the VPN peers 103, 105.” SAMSUNG-1006, [0040].



SAMSUNG-1006, FIG. 1

For traffic routed from the PEP to the VPN, the “PEP peer to VPN tunnel routing” can be performed “by a routing table, where this routing table identifies the VPN tunnel and its VPN peer’s IP address and provides one or more IP address

masks such that one of which matches the IP address of the PEP peer.” SAM-SUNG-1006, [0042].

(c) Combination

As described above, in Owen, communication pipes are “means of communication between the mobile device and some external entity,” and “[p]lacing a software application in a domain gives it access to domain assets, most notably the domain communication pipe(s).” SAMSUNG-1005, [0042]; §III.A.1(a). Owen further explains that “[p]lacing a communication pipe in a domain means that the domain owner trusts that communication pipe,” which “is usually due to their control of the gateway or of the encryption keys that are used for communications over the communication pipe.” SAMSUNG-1005, [0046]. Owen explains that “[t]he employer domain 54 and the user domain 64 are respectively secured using the employer key 56 and a user key.” SAMSUNG-1005, [0041].

Owen explains that domains ensure that user applications cannot “compromis[e] the employer’s network security.” SAMSUNG-1005, [0036]. For example, “[o]nly software applications in the employer domain can access corporate data which normally resides behind a security firewall in the employer’s corporate network,” and “none of the software applications in the carrier domain or the user domain can access the communication pipe in the employer domain.” SAMSUNG-

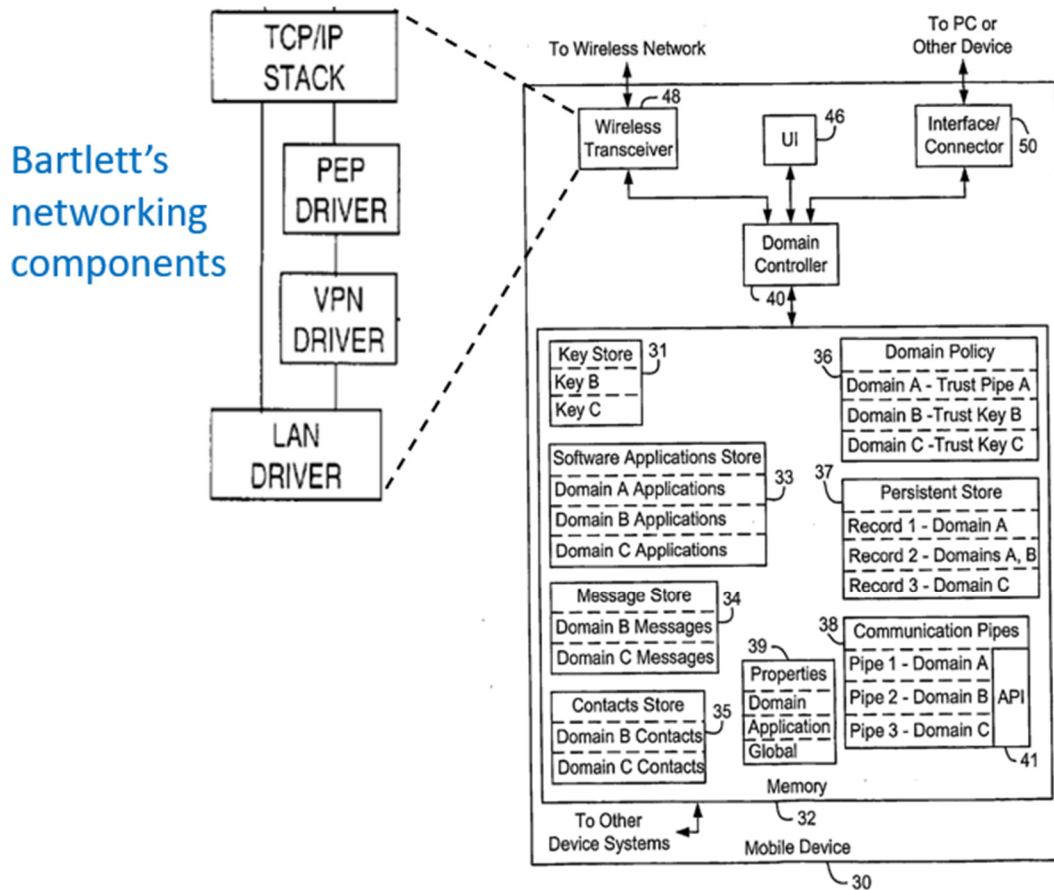
1005, [0036]. Further, “[i]n the employer domain 54, for example, the CRM software application and the corporate messaging software application have access to the corporate communication pipe, data, and properties.” SAMSUNG-1005, [0042].

While Owen explains that applications in the employer domain access the secure employer communication pipe, Owen does not explicitly describe *how* that secure pipe is established, so a POSITA¹ would have naturally looked to references such as Bartlett, which explains that the user generating traffic (e.g., CRM data directed to the corresponding pipe in the employer domain) automatically “trigger[s] the creation of the VPN connection and the PEP connection.” SAMSUNG-1006, [0058]; SAMSUNG-1003, ¶81.

The Owen-Bartlett device (OBD), illustrated below, comprises a mobile device, per Owen, and networking components (TCP/IP stack, PEP driver, VPN driver and LAN driver), per Bartlett. SAMSUNG-1003, ¶82.

¹ See SAMSUNG-1003, §IV.

Owen's mobile device with domains



SAMSUNG-1005, FIG. 2 (modified, annotated); SAMSUNG-1006, FIG. 14 (modified, annotated); SAMSUNG-1003, ¶82

In operation, per Owen, OBD's objects (e.g., applications, communication pipes) are assigned to domains, and their use is "restricted to other domain members." SAMSUNG-1005, Abstract, [0030], [0041]; SAMSUNG-1003, ¶83. For example, applications in the employer domain (e.g., CRM) communicate over a secure communication pipe using a gateway through a corporate security firewall.

SAMSUNG-1005, [0042], [0046]; §III.A.1(a); SAMSUNG-1003, ¶83. Per Bartlett, when data transmission from the application begins, if the secure communication pipe through the gateway has not yet been established, OBD automatically creates a VPN connection to allow the network traffic to pass through the firewall to an employer's server. SAMSUNG-1006, [0046], [0052], [0058]-[0059]; §III.A.1(b); SAMSUNG-1003, ¶83. The transmission from the application is then directed to the employer's servers using the VPN. SAMSUNG-1005, [0042], [0046]; SAMSUNG-1006, [0058]; SAMSUNG-1003, ¶83. A POSITA would have understood or found obvious that automatically establishing the VPN simplifies the user experience by not requiring "the user [to] explicitly request[] that the VPN connection be established," e.g., "by starting up the VPN client on the mobile host," which was "how VPN connections [were] traditionally established." SAMSUNG-1006, [0057]; SAMSUNG-1003, ¶83. Absent Bartlett's teachings, this improvement was "not viable because the user's connection will time out while waiting for the VPN connection to establish." SAMSUNG-1006, [0058]; SAMSUNG-1003, ¶83.

Additionally, the assigned domain, per Owen, is used as a characteristic "for selectively establishing a connection, based on the determined characteristics, to transport the received packets over the network." SAMSUNG-1005, [0024],

[0030]; SAMSUNG-1006, [0012]-[0014]; SAMSUNG-1003, ¶84. In OBD, a domain's (e.g., employer domain, per Owen) "connection is tuned based on the determined characteristics, and the peer is configured to establish an encrypted tunnel over the connection." SAMSUNG-1005, [0034]; SAMSUNG-1006, [0014]; SAMSUNG-1003, ¶84. For example, OBD can use the domain assignment (e.g., to the employer domain), per Owen, as a characteristic that causes transmissions from the domain to be assigned a quality-of-service level (QoS), per Bartlett (e.g., transmissions from the employer domain can be assigned a superior QoS compared to transmissions from the user domain). SAMSUNG-1005, [0034]; SAMSUNG-1006, [0053], [0075]; SAMSUNG-1003, ¶84. A POSITA would have understood that assigning traffic to/from applications in the employer domain to a higher QoS category provides such applications differentially better network service. SAMSUNG-1003, ¶84 (citing SAMSUNG-1030, *generally*).

A POSITA would have had a reasonable expectation of success in combining components of a mobile device, per Owen, with networking components (TCP/IP stack, PEP driver, VPN driver, LAN driver), per Bartlett, because both describe complementary functions of a mobile device. SAMSUNG-1005, Abstract, [0014]; SAMSUNG-1006, [0055], [0161]; §§III.A.1(a)-(b); SAMSUNG-1003, ¶85. For example, Owen's mobile device includes a "wireless transceiver 48 [that]

enables the mobile device 30 for communications via a wireless network.” SAMSUNG-1005, [0025]. Owen further explains that “[p]lacing a communication pipe in a domain means that the domain owner trusts that communication pipe,” which “is usually due to their control of the gateway or of the encryption keys that are used for communications over the communication pipe.” SAMSUNG-1005, [0046]. “[T]he communication pipe often includes a gateway through a corporate security firewall.” *Id.* While Owen provides limited details regarding the gateway or other details regarding securing network traffic, Bartlett explains that the VPN function, which encrypts data sent over a public network, can be integrated with other network functions in a mobile device so that “user’s traffic remains secure,” including over a “wireless local area network 328.” SAMSUNG-1005, *generally*; SAMSUNG-1006, [0055]-[0056]; SAMSUNG-1003, ¶85 (citing SAMSUNG-1015, 48). Thus, a POSITA would have naturally looked to Bartlett, which provides implementation details suggested by Owen. SAMSUNG-1003, ¶85.

The combination of Owen and Bartlett would have required only routine programming knowledge well within the skill of a POSITA before the Critical Date². SAMSUNG-1003, ¶86. Indeed, implementing the combination would have amounted to using known techniques to improve similar systems (mobile device networking) and combining prior art elements according to known methods to

² January 28, 2009. *See* SAMSUNG-1003, ¶26.

yield the predictable results described above. SAMSUNG-1003, ¶86; *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 417 (2007).

Finally, the elements of the resulting OBD would each perform functions they performed prior to combination—Owen limiting an application's traffic to communication pipes in the application's domain, and Bartlett establishing a VPN and using the VPN to ensure secure transmission of the data, e.g., over public/untrusted networks. SAMSUNG-1003, ¶87. A POSITA would have therefore expected success when combining Owen's and Bartlett's teachings. SAMSUNG-1003, ¶87.

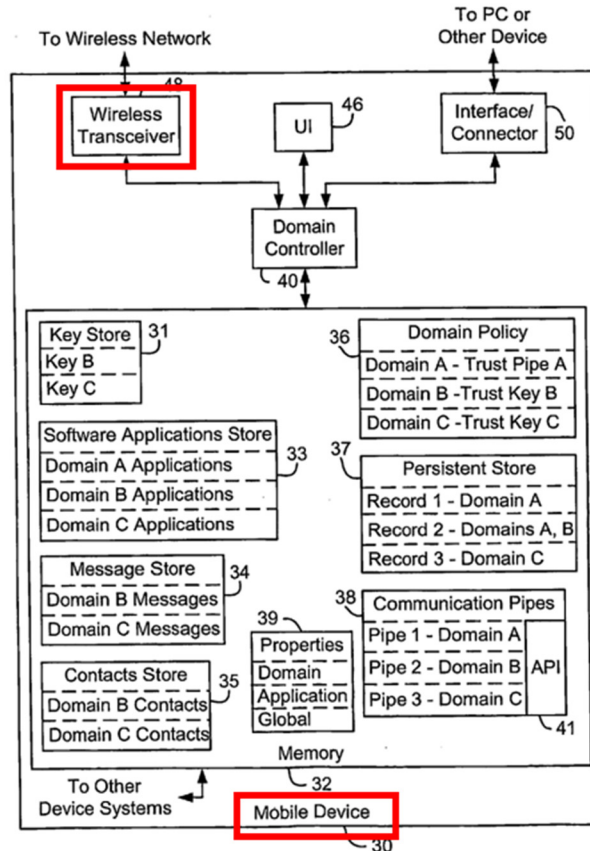
2. Analysis

[1.1]

To the extent the preamble is limiting, OBD renders it obvious. SAMSUNG-1003, ¶¶142-143.

For example, Owen discloses “secure control of a *wireless mobile communication device*.”³ SAMSUNG-1005, Abstract. Owen further states that “FIG. 2 is a block diagram of an exemplary *wireless mobile communication device* in which a system and *method* for secure control may be implemented.” SAMSUNG-1005, [0022].

³ Emphasis added throughout unless otherwise noted.



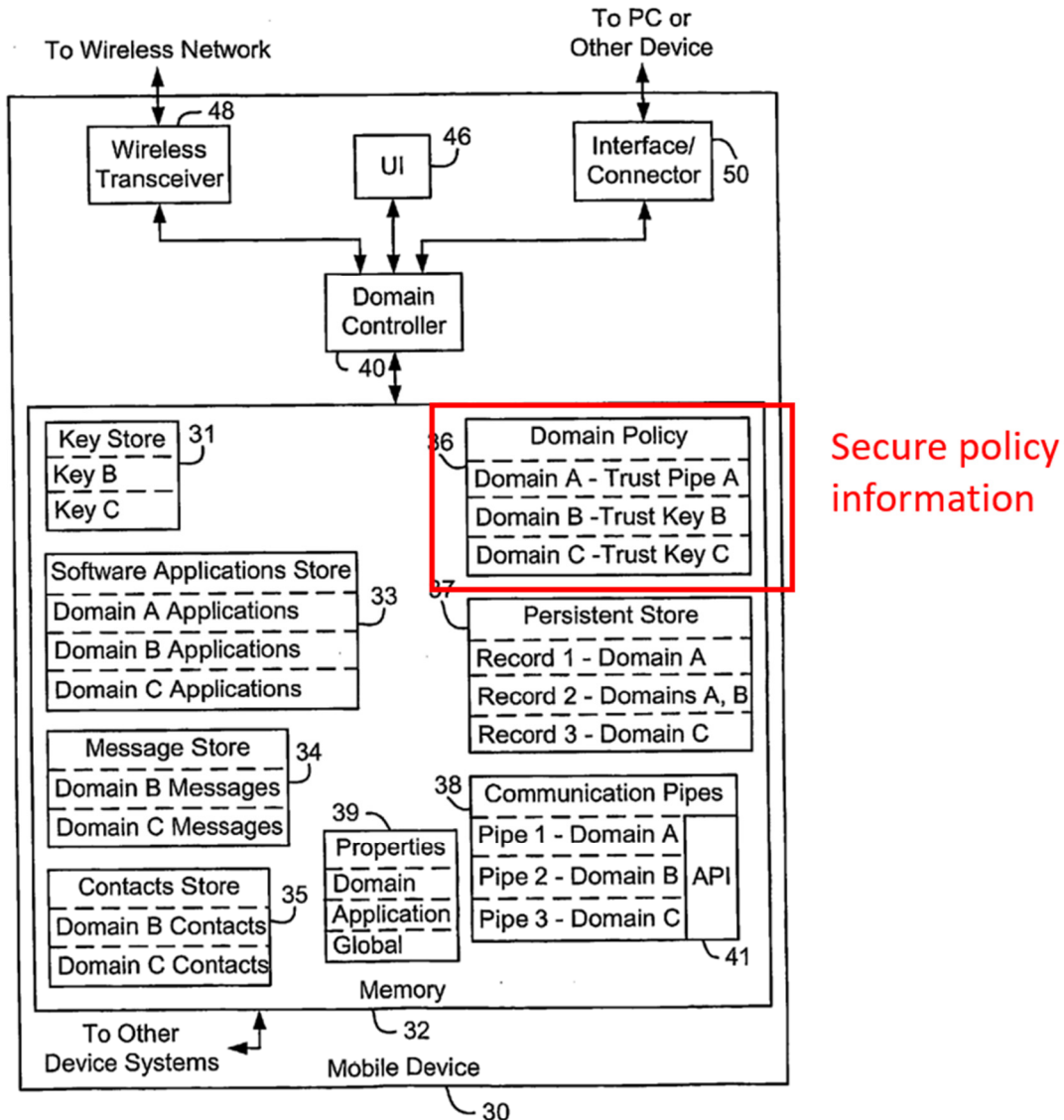
SAMSUNG-1005, FIG. 2(annotated)

[1.2.i]

OBD renders [1.2.i] obvious. SAMSUNG-1003, ¶¶144-146.

Per Owen, OBD stores secure policy information. SAMSUNG-1003, ¶145.

For example, Owen's memory includes "a domain policy store." SAMSUNG-1005, [0023]. The domain policy store stores "domain membership control policies ('domain policies'), which specify the criteria used to determine into which domain a software application, property, or other information should be placed." SAMSUNG-1005, [0024].



SAMSUNG-1005, FIG. 2(annotated)

The secure policy information is indicative of an application-specific execution environment policy and a configurable communication security policy, as explained in reference to elements [1.2.ii] and [1.2.iii]. SAMSUNG-1003, ¶146; §§III.A.2[1.2.ii]-[1.2.iii].

[1.2.ii]

OBD renders [1.2.ii] obvious. SAMSUNG-1003, ¶¶147-154.

Per Owen, in OBD, a “domain is a collection of objects that share a common level of trust,” and “[p]lacing an object in a domain means that the object is trusted, and that its use can be restricted to other domain members.” SAMSUNG-1005, [0030]. Owen provides an example of a “carrier domain [that] may include such assets as a browser that can be used to access Internet information and services via the carrier's wireless communication network.” *Id.*, [0032]. “The user may then use the browser in the carrier domain to subscribe to a chat service, download the required software application from the Internet, and place the downloaded software application into the carrier domain.” *Id.* Owen provides additional examples of a “user domain” and “employer domain.” *Id.*, [0033]-[0036].

Further, Owen teaches “domain policies” that control “which objects can become members of a domain.” SAMSUNG-1005, [0030]. Owen notes that the “introduction of domains in this example has some important mobile device control and security benefits,” e.g., “the birthday reminder program in the user domain cannot access the company directory in the employer domain, even though they are both stored on the same mobile device.” SAMSUNG-1005, [0035].

Additionally, per Owen, OBD’s domain policies are application-specific. SAMSUNG-1003, ¶150. For example, Owen explains that “[e]ach domain has a

domain policy that controls which objects can become members of a domain” and “domain membership control policies (‘domain policies’), which specify the criteria used to determine into which domain *a software application* [] should be placed.” SAMSUNG-1005, [0024], [0030]. For example, the “employer domain” can include “the company directory and a Customer Relationship Management (CRM) software application.” SAMSUNG-1005, [0034]. Owen explains that, “when a domain is created, it may include an access list established by the domain owner that describes all allowable members,” which “works well with software applications.” SAMSUNG-1005, [0100]. Since the domain policies determine the placement of software applications (e.g., CRM and corporate directory), a POSITA would have understood that domain policies are application-specific. SAMSUNG-1003, ¶150.

Owen further teaches that domain policies are indicative of execution environment policies. SAMSUNG-1003, ¶151 (citing SAMSUNG-1001, SAMSUNG-1011, defining an execution environment). For example, as described above, domain policies determine a software application’s domain. SAMSUNG-1005, [0024]. Further, an application’s domain determines “access to domain assets, most notably the domain communication pipe(s) and domain data such as persistent data and properties.” SAMSUNG-1005, [0042]. Thus, the domain determines access rights for applications in the domain, thereby determining properties of an

application's execution environment – e.g., which assets applications in the domain can and cannot access. SAMSUNG-1003, ¶151. Since domain policies determine to which domain an application is assigned, and the domain determines properties of the execution environment (e.g., determining access to resources in the domain), a POSITA would have understood that domain policies are indicative of application-specific execution environment policies (e.g., access control policies that indicate access rights). SAMSUNG-1005, [0024], [0042]; SAMSUNG-1003, ¶152, §VII.A.

Additionally, Owen explains that the “domain backup frequency” “specifies how often domain data should be backed up.” SAMSUNG-1005, [0032]. The backup frequency can be configurable, e.g., by the carrier for the carrier domain, and by the user for the user domain. SAMSUNG-1005, [0032]-[0033]. Since the backup frequency is a configurable setting (policy) that determines operation of the execution environment (how often application data is backed up for a domain), and domain policies assign applications to domains, a POSITA would have understood or found obvious that domain policies are indicative of execution environment policies (e.g., backup frequency). SAMSUNG-1003, ¶153.

Thus, a POSITA would have understood or found obvious that, per Owen, OBB renders obvious policy information (domain policies) indicative of an appli-

cation-specific execution environment policy (e.g., access rights and backup frequency). SAMSUNG-1003, ¶154.

[1.2.iii]

OBD renders [1.2.iii] obvious. SAMSUNG-1003, ¶¶155-159.

As described above, per Owen, “[p]lacing a software application in a domain gives it access to domain assets, most notably the domain communication pipe(s),” which are “means of communication between the mobile device and some external entity.” SAMSUNG-1005, [0042], [0045]. Owen explains that “[p]lacing a communication pipe in a domain means that the domain owner trusts that communication pipe,” which “is usually due to their control of the gateway” or the communication pipe’s encryption keys. SAMSUNG-1005, [0046]. Owen further explains that “[o]nly software applications in the employer domain can access corporate data which normally resides behind a security firewall in the employer’s corporate network.” SAMSUNG-1005, [0036]. Therefore, for a “corporate communication pipe, the communication pipe often includes a gateway through a corporate security firewall, such that access to the communication pipe represents a possible avenue of attack against a corporate infrastructure,” and “[b]y placing the communication pipe inside a domain that they control, a corporate entity such as an employer restricts access to the communication pipe and reduce[s] the likelihood of attack.” SAMSUNG-1005, [0046]. Owen provides an example in which “[i]n the employer

domain,” e.g., “the CRM software application and the corporate messaging software application have access to the corporate communication pipe.” SAMSUNG-1005, [0042].

Therefore, since the domain policy determines an application’s (e.g., the CRM application’s) access to communications pipes (e.g., the corporate communication pipe), a domain policy is indicative of a communication policy. SAMSUNG-1003, ¶157. Further, since access to the communication pipe can be restricted, e.g., by controlling encryption keys, a POSITA would have understood or found obvious that the communication policy is used for secure communication and is thus a secure communication policy. SAMSUNG-1005, [0046]; SAMSUNG-1003, ¶157.

Per Owen, OBD’s communication policies are configurable. SAMSUNG-1005, [0100]; SAMSUNG-1003, ¶158. For example, Owen explains that, “[a]s a further alternative domain policy, when a domain is created, it may include an access list established by the domain owner that describes all allowable members.” SAMSUNG-1005, [0100]. Since the “domain owner” determines the “access list” (i.e., the domain policy), the domain policy is configurable (e.g., by the domain owner). SAMSUNG-1005, [0100]; SAMSUNG-1003, ¶158.

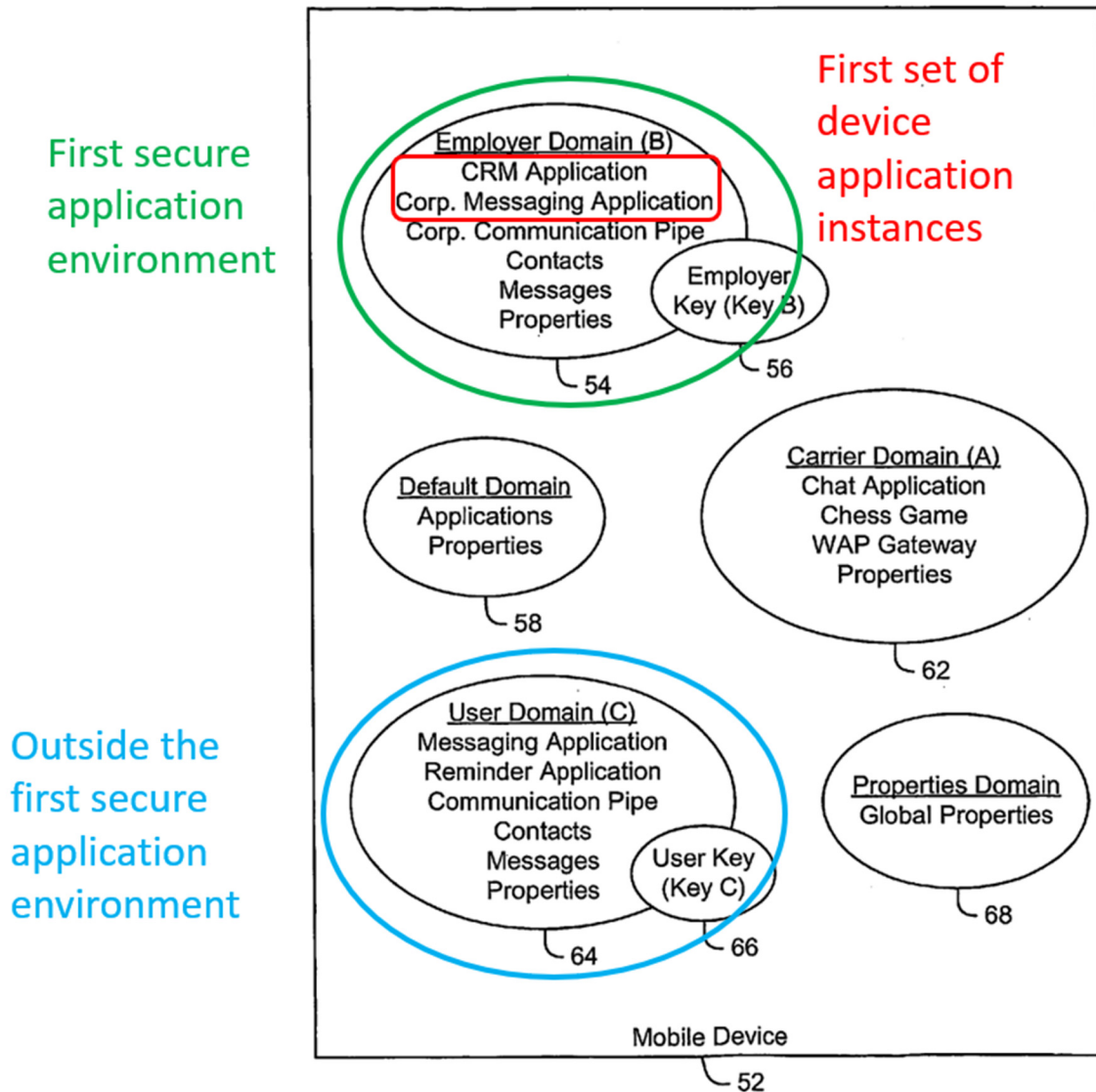
Thus, Owen renders obvious policy information (domain policy) indicative of a configurable (by the domain owner) communication security policy (access to

a communication pipe secured by encryption keys). SAMSUNG-1003, ¶159.

[1.3.i]

Owen renders [1.3.i] obvious. SAMSUNG-1003, ¶¶160-165.

As described in [1.2.ii], applications are assigned to application environments on the wireless end-user device. §III.A.2[1.2.ii]; SAMSUNG-1003, ¶161. For example, Owen teaches running some applications in the employer domain, e.g., “the company directory and a Customer Relationship Management (CRM) software application,” and FIG. 3 further illustrates a corporate messaging application. SAMSUNG-1005, [0034]-[0036], [0041]-[0043], FIG. 3; SAMSUNG-1003, ¶161. Therefore, Owen renders obvious allowing a first set of multiple device application instances (e.g., CRM, company directory, and corporate messaging applications) to run only in a first application environment on the wireless end-user device, e.g., the employer domain. SAMSUNG-1003, ¶161; SAMSUNG-1005, FIG. 3.



SAMSUNG-1005, FIG. 3(annotated)

The '250 patent does not define the term “application instance.” SAMSUNG-1001, *generally*. Absent such a definition, a POSITA would have understood an “application instance” simply as an instance of an application, e.g., the instance of the application that exists on the mobile device. SAMSUNG-1003, ¶163 (citing SAMSUNG-1001, 12:49-13:25).

Owen further teaches that application environments are secure, e.g. using encryption keys. SAMSUNG-1003, ¶165; SAMSUNG-1005, [0041], [0093]-[0098]. Owen provides examples in which “[t]he employer domain 54 and the user domain 64 are respectively *secured using the employer key 56 and a user key 66.*” SAMSUNG-1005, [0041]. Using cryptographic keys provides a “stronger method of domain security.” SAMSUNG-1005, [0093]. For example, “[i]n a public key system, each domain has an associated public and private key,” and “[a]nything that is added to a domain must be digitally signed using the domain private key.” *Id.*

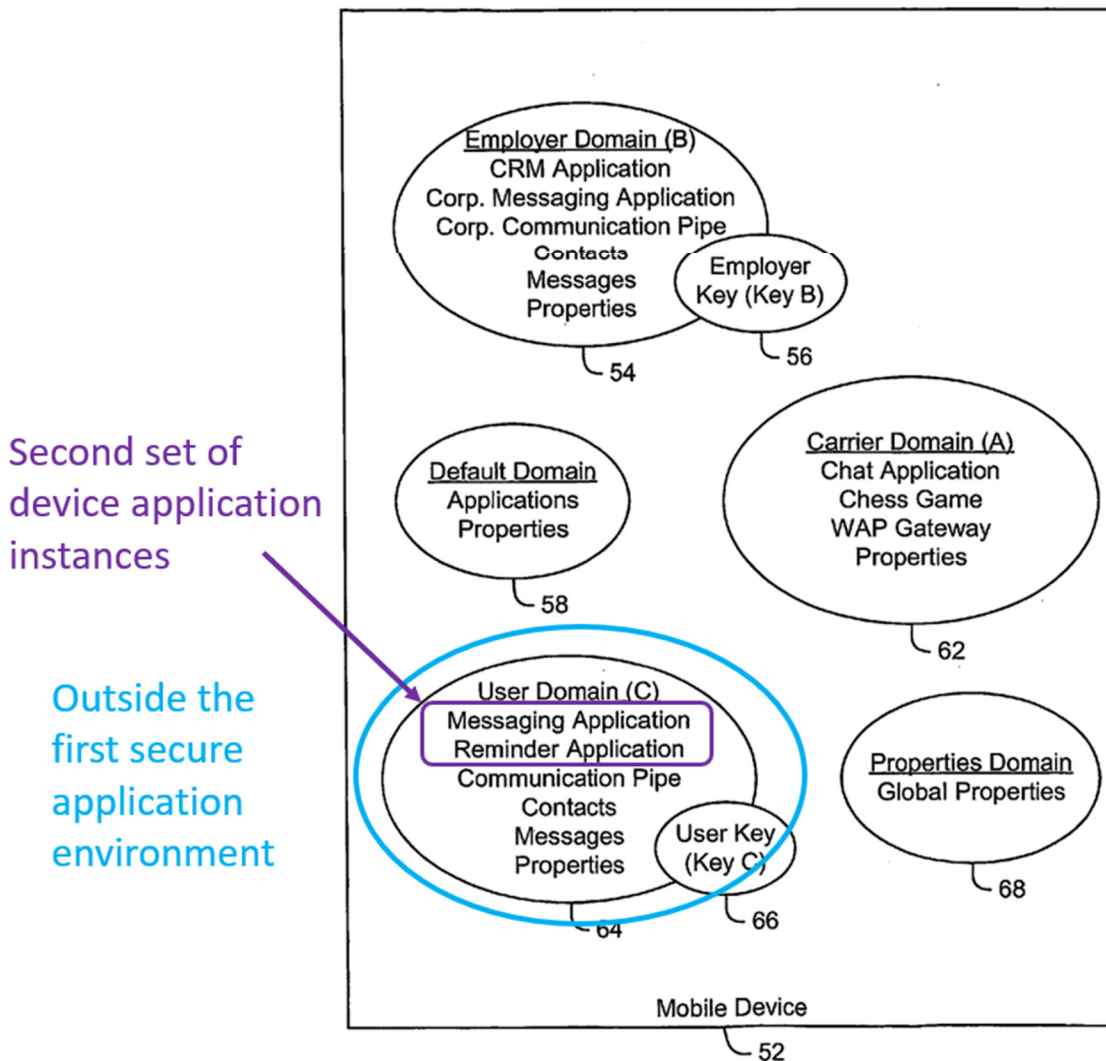
Therefore, Owen renders obvious allowing a first set of multiple device application instances (e.g., CRM, Corporate Directory) to run only in a first secure application environment (employer domain) on the wireless end-user device. SAMSUNG-1005, [0034]; SAMSUNG-1003, ¶165.

[1.3.ii]

Owen renders [1.3.ii] obvious. SAMSUNG-1003, ¶¶166-167.

As described in [1.2.ii], applications are assigned to domains. §III.A.2[1.2.ii]; SAMSUNG-1003, ¶167. For example, Owen teaches running some applications in the user domain, e.g., a “birthday reminder program” and “personal messaging,” and FIG. 3 further illustrates a “messaging application.” SAMSUNG-1005, [0033], [0035]-[0036], [0041], [0043]. Since the user domain is outside the employer domain (first secure application environment), Owen renders

obvious allowing a second set of device application instances to run only outside of the first secure application environment (e.g., in the user domain which is outside of the employer domain). SAMSUNG-1003, ¶167; SAMSUNG-1005, FIG. 3.



SAMSUNG-1005, FIG. 3 (annotated)

[1.4.i]

OBD renders [1.4.i] obvious. SAMSUNG-1003, ¶¶168-174.

As described in [1.2.iii], OBD includes domain policies, which are policy information indicative of configurable communication security policies. SAMSUNG-1003, ¶169.

Further, as described in §III.A.2[1.2.iii], applications access communication pipes based on the configurable communication security policies (e.g., domain policies). For example, Owen explains that “[p]lacing a software application in a domain gives it access to domain assets, most notably the domain communication pipe(s) and domain data such as persistent data and properties.” SAMSUNG-1005, [0042]. Owen further explains that applications in a domain transmit data over communication pipes assigned to that domain. SAMSUNG-1003, ¶170. For example, Owen writes, “the CRM software application and the corporate messaging software application have access to the corporate communication pipe.” SAMSUNG-1005, [0042]. Owen further explains that “any synchronization or backup of the employer domain data is performed through the employer domain communication pipe, so that the user’s personal contacts are not copied to the corporate server.” SAMSUNG-1005, [0035]; SAMSUNG-1003, ¶170.

Thus, per Owen, in OBD, data are transmitted through a communication pipe assigned to the domain by a domain policy, e.g., since the CRM application is in the employer domain, synchronization and backup of CRM data traverses the employer domain communication pipe. SAMSUNG-1003, ¶171.

As described in §III.A.2[1.2.iii], “[o]nly software applications in the employer domain can access corporate data which normally resides behind a security firewall in the employer's corporate network,” so “the communication pipe often includes a gateway through a corporate security firewall.” SAMSUNG-1005, [0036], [0045]. A POSITA would have understood that a “gateway through a corporate security firewall” is a tunnel, e.g., a VPN. SAMSUNG-1003, ¶172 (citing SAMSUNG-1014, [0034]).

In OBD, when data are first transmitted through the communication pipe (and the pipe is assigned to a domain based on a domain policy) that “includes a gateway through a corporate security firewall,” per Bartlett, ODB initiates the tunnel. SAMSUNG-1003, ¶173. For example, Bartlett describes a “method of establishing the PEP and VPN connections [which] involves the user generating traffic to automatically invoke the PEP function as to trigger the creation of the VPN connection and the PEP connection.” SAMSUNG-1006, [0058].

Since generating the traffic occurs as a result of the communication security policy (as described above), and generating the traffic causes the initiation of the tunnel, OBD discloses or renders obvious initiating a first secure tunnel to a network tunnel endpoint (“automatically invoke the PEP function as to trigger the creation of the VPN connection” which provides the “gateway through a corporate se-

curity firewall”) based on the secure stored information indicative of a configurable communication security policy (e.g., assigning the CRM application to the employer domain). SAMSUNG-1005, [0045]; SAMSUNG-1006, [0058]; SAMSUNG-1003, ¶174.

[1.4.ii]

Owen renders [1.4.ii] obvious. SAMSUNG-1003, ¶¶175-180.

As described in §III.A.2[1.2.iii], applications access communication pipes based on configurable communication security policies. For example, Owen explains that “[p]lacing a software application in a domain gives it access to domain assets, most notably the domain communication pipe(s).” SAMSUNG-1005, [0042]. Owen further explains that applications in a domain transmit data over communication pipes assigned to that domain. SAMSUNG-1003, ¶176. For example, Owen explains, “the CRM software application and the corporate messaging software application have access to the corporate communication pipe.” SAMSUNG-1005, [0042]. Owen further explains that “any synchronization or backup of the employer domain data is performed through the employer domain communication pipe.” SAMSUNG-1005, [0035]; SAMSUNG-1003, ¶176.

More generally, Owen explains that “[o]ne of the primary uses for domains is to provide data security,” and “[b]y placing a software application in a domain, access to that software application's data is restricted to other software applications

in that domain.” SAMSUNG-1005, [0113]; SAMSUNG-1003, ¶177. A POSITA would have understood or found obvious that directing all network data communications to and from device application instances executing in the first secure application environment through the first secure tunnel provides data security, as taught by Owen. SAMSUNG-1005, [0113]; SAMSUNG-1003, ¶177.

Thus, per Owen, in OBD, data are transmitted through a communication pipe assigned to the domain, e.g., since the CRM application is in the employer domain, synchronization and backup of CRM data traverses the employer domain communication pipe. SAMSUNG-1003, ¶178.

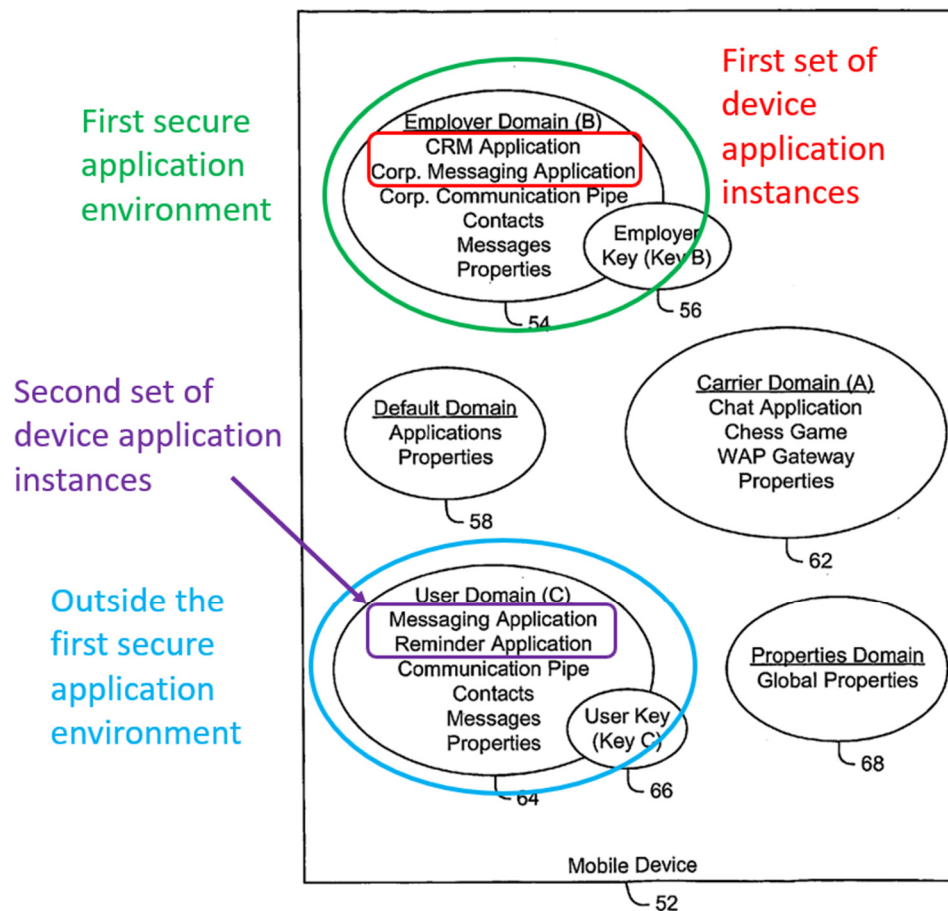
As described in §III.A.2[1.2.iii], “the communication pipe often includes a gateway through a corporate security firewall.” SAMSUNG-1005, [0045]. A POSITA would have understood that a “gateway through a corporate security firewall” is a tunnel, e.g., a VPN. SAMSUNG-1003, ¶179 (citing SAMSUNG-1014).

Therefore, OBD discloses or renders obvious directing all network data communications to and from device application instances (e.g., the CRM application instance on OBD) executing in the first secure application environment (the employer domain) through the first secure tunnel (the employer communication pipe tunneling through the corporate firewall) based on the secure stored information indicative of a configurable communication security policy (the domain policy for the employer domain). SAMSUNG-1003, ¶180.

[2.1]

OBD renders [2.1] obvious. SAMSUNG-1003, ¶¶181-186.

Owen's domain policies determine "which objects can become members of a domain." SAMSUNG-1005, [0030]. Since the domain policies determine which objects (e.g., applications and communication pipe) are assigned to a domain (by specifying "the criteria used to determine into which domain a software application" should be placed), a POSITA would have understood that OBD's configurable communication security policies (domain policies, per Owen) define a first configuration. SAMSUNG-1005, Abstract, [0024]; §III.A.2[1.2.iii]; SAMSUNG-1003, ¶182. Owen provides an example of such a configuration in FIG. 3, reproduced with annotations below. SAMSUNG-1005, FIG. 3 (annotated); SAMSUNG-1003, ¶182.



SAMSUNG-1005, FIG. 3 (annotated); SAMSUNG-1003, ¶182

Further, as described above, per Owen, an OBD domain policy specifies “the criteria used to determine into which domain a software application, property, or other information should be placed.” SAMSUNG-1005, [0024]. For example, Owen describes an “alternative domain policy, when a domain is created, it may include an access list established by the domain owner that describes all allowable members.” SAMSUNG-1005, [0100]. Per Owen, in OBD, “[p]lacing an object in a domain means that the object is trusted, and that its use can be restricted to other domain members.” SAMSUNG-1005, [0030]. Thus, since a domain policy “may

include an access list” that “describes *all allowable members*,” a POSITA would have understood of found obvious that domain policies indicate which application instances can and cannot (“allowable”) be members of a domain. SAMSUNG-1005, [0100]; SAMSUNG-1003, ¶183.

Owen further explains that communication pipes are assigned to domains, noting that “by placing the communication pipe inside a domain that they control, a corporate entity such as an employer restricts access to the communication pipe and reduce[s] the likelihood of attack.” SAMSUNG-1005, [0046]; SAMSUNG-1003, ¶184. Owens states that such restrictions can be enforced by a domain controller, e.g., “[w]hen data is to be sent using a particular domain’s communication pipe, for example, the domain controller first determines whether an object or domain attempting to send the data has been granted access to the communication pipe by the domain owner, such as by determining whether the sending object or domain is in the same domain as the communication pipe.” SAMSUNG-1005, [0107]. Then, the domain controller determines “to which domain a communication pipe belongs, and then send data to the communication pipe via the API 41 where the sending object in that same domain or has been granted access to the communication pipe in that domain by the domain owner.” SAMSUNG-1005, [0107].

Since access to objects in a domain are “restricted to other domain members,” a POSITA would have understood or found obvious that, in OBD, in a configuration in which a set of application instances are assigned to a second domain (e.g., user application instances assigned to a user domain) and a communication pipe (secure tunnel) is assigned to a first domain (e.g., employer domain), application instances in the second domain cannot access the first communication pipe. SAMSUNG-1005, [0030]; SAMSUNG-1003, ¶185. For example, Owen explains that “none of the software applications in the carrier domain or the user domain can access the communication pipe in the employer domain,” and “[o]nly software applications in the employer domain can access corporate data which normally resides behind a security firewall in the employer's corporate network.” SAMSUNG-1005, [0036]. Therefore, per Owen, OBD discloses or renders obvious that, as indicated by the configuration, application instances in the second set of device application instances (e.g., applications assigned to the user domain by the domain policy) are not allowed to use the first secure tunnel (the corporate communication pipe, which is in the employer domain). SAMSUNG-1003, ¶185.

Thus, Owen teaches or renders obvious that a communication security policy (domain policy) defines a first configuration (assignment of applications, communication pipes, etc. to domains), which indicates that application instances in the second set of device application instances are not allowed to use the first secure

tunnel (since in the configuration, application instances in the second set of device application instances are in a different domain from the first secure tunnel). SAMSUNG-1003, ¶186.

[2.2]

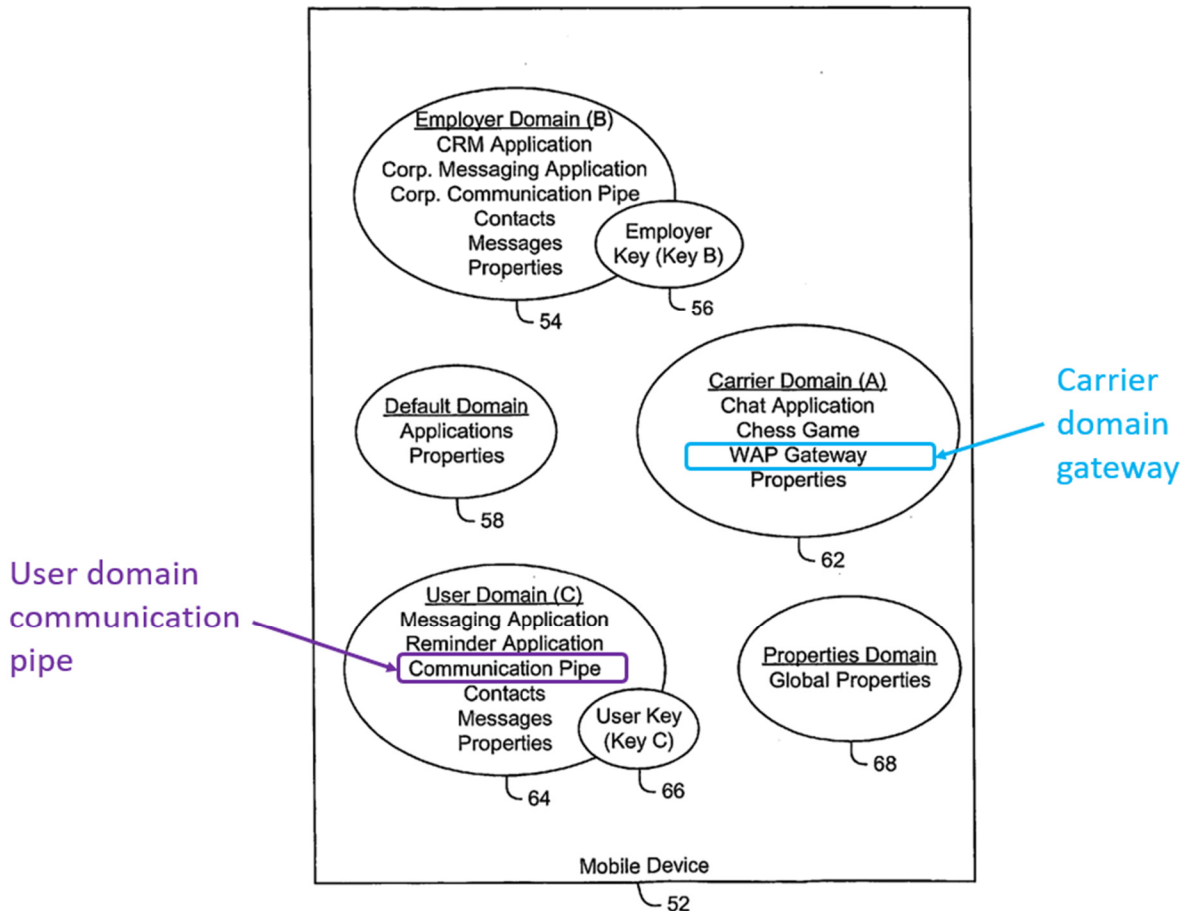
OBD renders [2.2] obvious. SAMSUNG-1003, ¶¶187-192.

As described in [2.1], OBD renders obvious that the secure stored information is indicative of a configurable communication security policy indicating the first configuration. §III.A.2[2.1].

As further described in [2.1], application instances in the second set of device application instances (e.g., applications in the user domain) are not permitted to use the first secure tunnel (e.g., the tunnel for the corporate communication pipe, which is in the employer domain). §III.A.2[2.2]; SAMSUNG-1003, ¶189.

Instead, per Owen, OBD directs traffic through other network paths, e.g., “carrier WAP gateway in the carrier domain,” and Owen explains that the “WAP gateway and the corporate communication pipe represent *separate communication pipes*.” SAMSUNG-1005, [0045]; SAMSUNG-1003, ¶190. Owen provides another example in which “synchronization of the user domain data, such as personal contacts, between the mobile device and a user’s PC is accomplished through a user domain communication pipe.” SAMSUNG-1005, [0035]. Such restrictions are enforced by OBD’s domain controller, as explained in [2.1]. SAMSUNG-

1005, [0107]; §III.A.2[2.1]; SAMSUNG-1003, ¶190.



SAMSUNG-1005, FIG. 3(annotated)

More generally, a POSITA would have understood that, since use of a communication pipe is “restricted to other domain members,” all network data communications to and from device application instances executing outside of the first secure application environment (e.g., assigned to the user domain) are directed through a communication pipe assigned to the application instance’s domain (e.g., a communication pipe assigned to the user domain), which is a network path outside of the first secure tunnel (since the communication pipe assigned to the user

domain is, at least in some cases, not also assigned to the employer domain).

SAMSUNG-1005, [0030]; SAMSUNG-1003, ¶191.

Thus, OBD renders obvious that, based on the secure stored information indicative of a configurable communication security policy (e.g., a domain policy) indicating the first configuration (e.g., user and employer domain policies), directing all network data communications to and from device application instances executing outside of the first secure application environment (e.g., applications assigned to the user domain) through a network path outside of the first secure tunnel (e.g., through the communication pipe assigned to the user domain, which is outside the communication pipe (tunnel) assigned to the employer domain). SAMSUNG-1003, ¶192.

B. Ground 1B: Owen-Bartlett-Smith renders claims 1-2 obvious

1. Owen-Bartlett-Smith

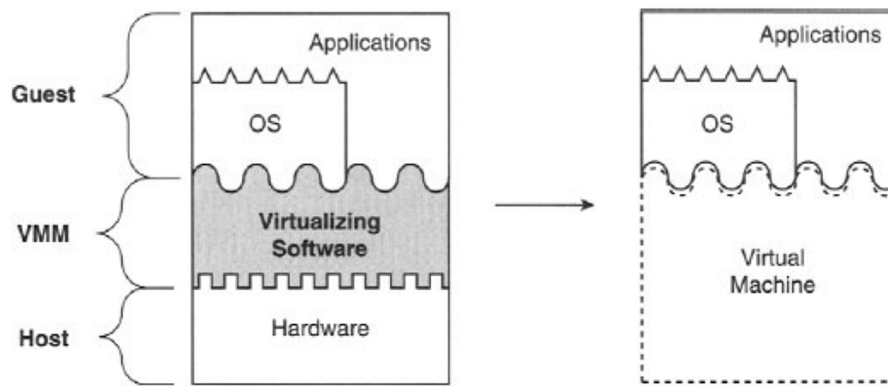
(a) Smith

Smith is a 2005 book that explains “[v]irtual machine (VM) technologies” “enable new capabilities and to solve a variety of problems in interfacing major computer system components.” SAMSUNG-1008, xi; SAMSUNG-1016, 37.

Smith defines a system as “a full execution environment that can simultaneously support a number of processes potentially belonging to different users,” and explains that, with virtualization, “the real system is transformed so that it appears

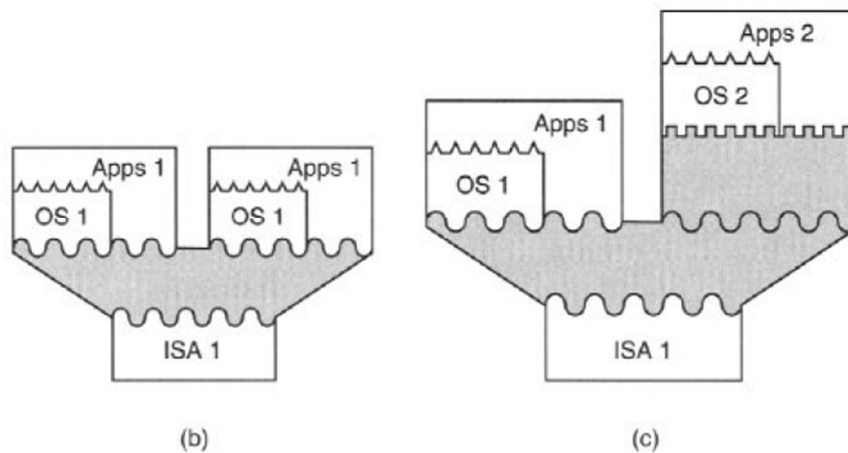
to be a different, virtual system or even a set of multiple virtual systems.” SAMSUNG-1008, 3, 9; SAMSUNG-1016, 54, 62. Smith further explains that, “[i]n a system VM environment, real resources of the host platform are shared among the guest system VMs, with a layer of software, the virtual machine monitor (VMM), managing the allocation of, and access to, the hardware resources of the host platform.” SAMSUNG-1008, 369; SAMSUNG-1016, 505. “The VMM owns the real system resources and makes them available to one or more guest operating systems that alternately execute on the same hardware,” and “a guest operating system in a virtual machine environment is given the illusion of owning the resources it subsequently allocates to its various user programs.” *Id.*

Smith states that “[a] virtual machine (VM) is implemented by adding a layer of software to a real machine to support the desired virtual machine’s architecture.” SAMSUNG-1008, 4; SAMSUNG-1016, 55. Such “virtualizing software is often referred to as the virtual machine monitor (VMM).” SAMSUNG-1008, 11; SAMSUNG-1016, 64.



SAMSUNG-1008, FIG. 1.7; SAMSUNG-1016, FIG. 1.7

Smith further explains that “[m]ultiple, replicated virtual machines can be implemented on a single hardware platform to provide individuals or user groups with their own operating system environments.” SAMSUNG-1008, 5; SAMSUNG-1016, 56. In Figure 1.8(b), Smith illustrates an example in which a set of applications (Apps 1) are running on the same operating system (e.g., where both instances of the operating system “OS1” are Linux or both are Windows) in two different VMs, and in Figure 1.8(b), Smith illustrates two different sets of applications (Apps 1 and Apps 2) are running on different operating systems (e.g., “OS1” is Windows and “OS2” is Linux). SAMSUNG-1008, Figures 1.8 (b)-(c); SAMSUNG-1016, Figures 1.8 (b)-(c); SAMSUNG-1003, ¶91.



SAMSUNG-1008, Figures 1.8 (b)-(c); SAMSUNG-1016, Figures 1.8 (b)-(c)

Smith explains that a “system VM provides a sandbox that isolates one system environment from other environments, thus ensuring a level of security that may not be provided by a single operating system.” SAMSUNG-1008, 370-371; SAMSUNG-1016, 507. Further, with VMs, “different system environments (possibly with different operating systems) also provide isolation and enhanced security.” SAMSUNG-1008, 5; SAMSUNG-1016, 56. Smith notes an important feature of system VMs “is that they provide a secure way of partitioning major software systems that run concurrently on the same hardware platform” such that “[s]oftware running on one guest system is isolated from software running on other guest systems.” SAMSUNG-1008, 18; SAMSUNG-1016, 72.

Smith further explains that “[r]unning each application in its own virtual machine increases the robustness of the system,” since erratic behavior “on a virtual machine” is “less likely to affect the operation of another application running on a

different virtual machine,” thus isolating each VM from security breaches on other VMs. SAMSUNG-1008, 370; SAMSUNG-1016, 507; *see* SAMSUNG-1008, 18; SAMSUNG-1016, 72.

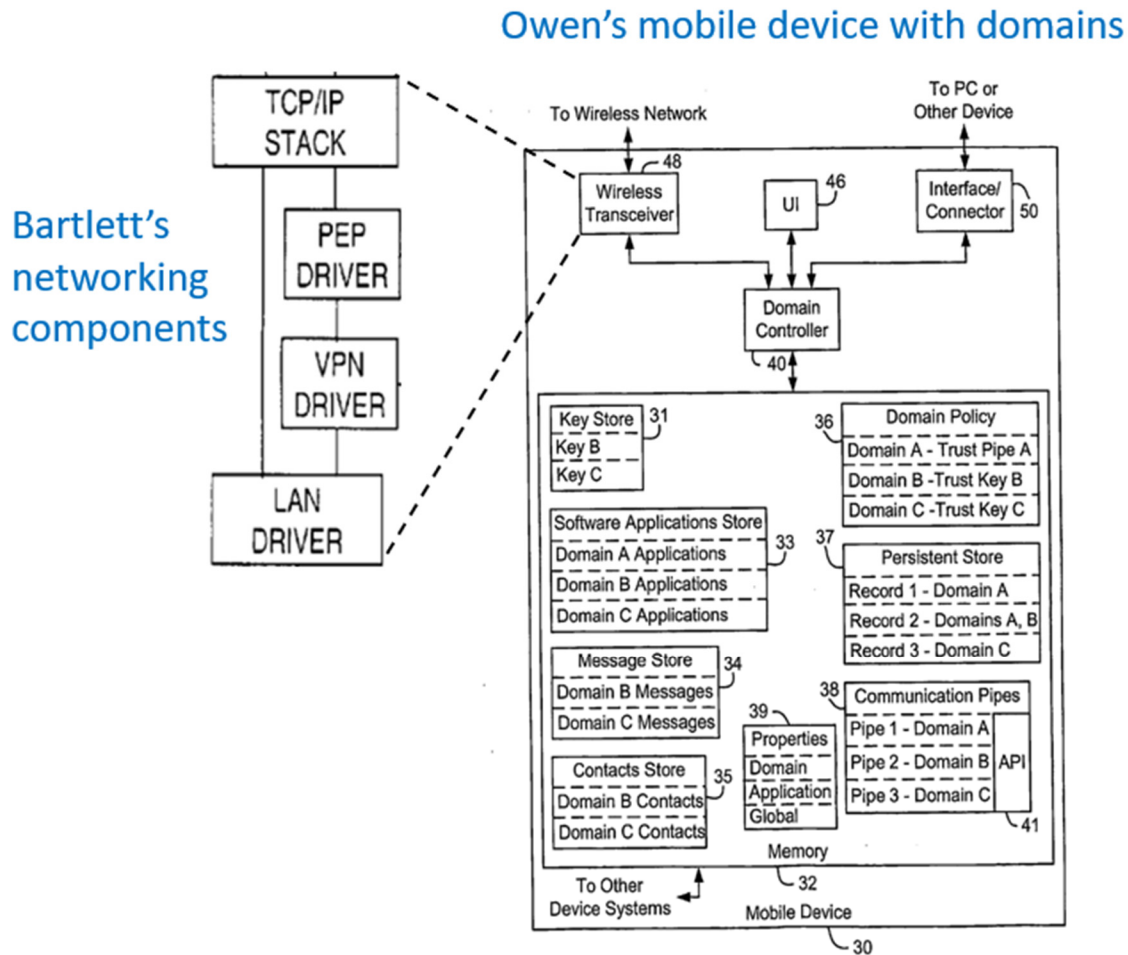
(b) Combination

Owen explains that a “domain is a collection of objects that share a common level of trust, and can be owned and controlled by a mobile device stakeholder,” and that use of objects in a domain “can be restricted to other domain members.” SAMSUNG-1005, [0030].

However, to the extent Owen does not *explicitly* describe how domains are implemented, including restricting access to other domain members, a POSITA would have looked to references such as Smith, which explains that “[p]erhaps the most important feature of today’s system VMs is that they provide a secure way of *partitioning major software systems that run concurrently on the same hardware platform*” such that “[s]oftware running on one guest system is isolated from *software running on other guest systems*,” and a “system VM provides a sandbox that isolates one system environment from other environments, thus ensuring a level of security that may not be provided by a single operating system.” SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, ¶95. Smith further explains that, by using VMs, different system environments “provide isolation and enhanced security.” SAMSUNG-1008, 5; SAMSUNG-1016, 56;

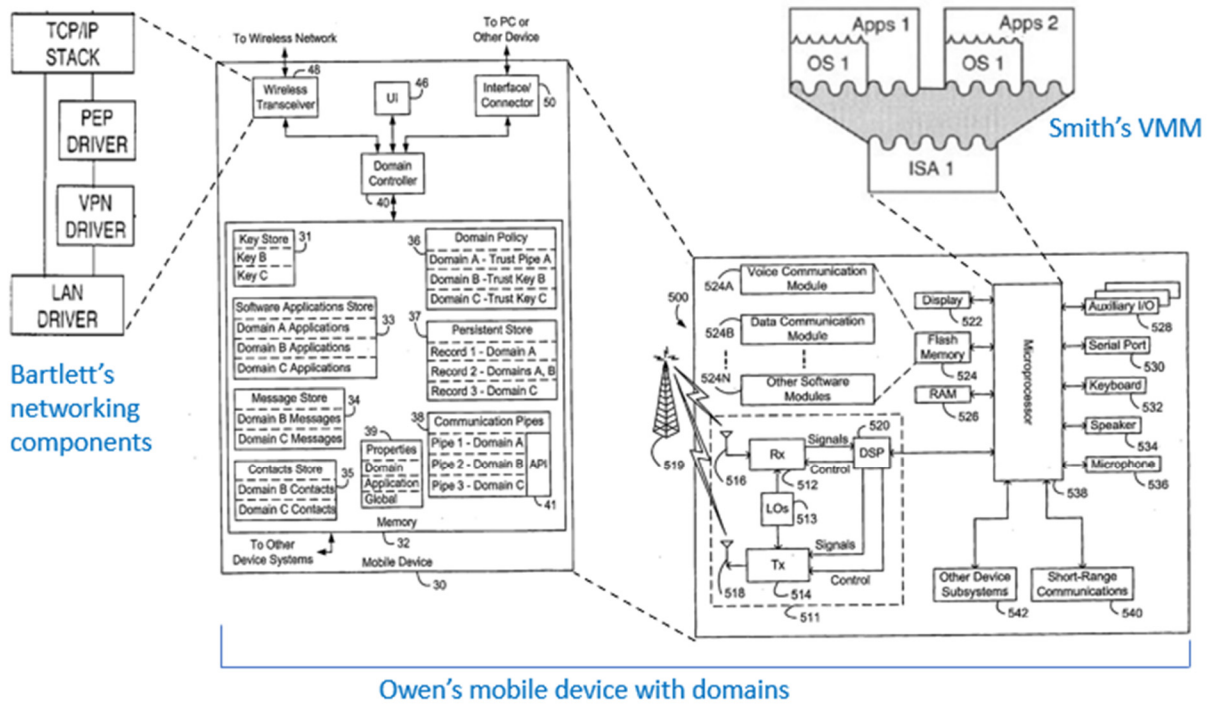
SAMSUNG-1003, ¶95. In fact, Owen discloses an alternative that uses VM technology, providing an alternative “where the mobile device 30 is a Java™-enabled device including a Java Virtual Machine (JVM) as its operating system.” SAMSUNG-1005, [0025]; SAMSUNG-1003, ¶95. Thus, a POSITA would have been motivated to combine OBD with Smith at least to isolate applications in different domains by assigning them to different VMs, as described further below. SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, ¶95.

As described in §III.A.1(c), OBD comprises a mobile device, per Owen, and further includes networking components (TCP/IP stack, PEP driver, VPN driver and LAN driver), per Bartlett. §III.A.1(c).



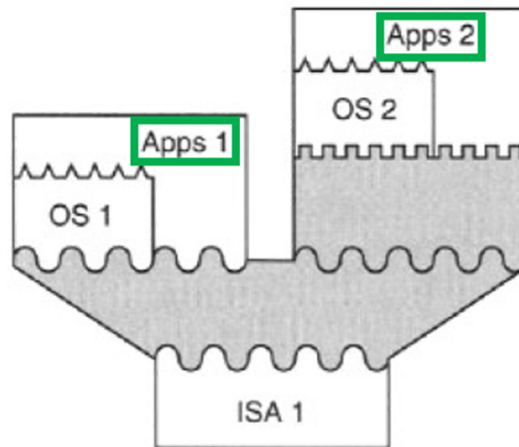
SAMSUNG-1005, FIG. 2 (modified, annotated); SAMSUNG-1006, FIG. 14 (modified, annotated); SAMSUNG-1003, ¶¶96

The Owen-Bartlett-Smith Device (OBSD) comprises ODB and further includes a VMM that provides virtual machines, per Smith. §§III.A.1(c), III.B.1(a); SAMSUNG-1003, ¶¶97-104.



SAMSUNG-1005, FIG. 2 (modified, annotated); SAMSUNG-1006, FIG. 14 (modified, annotated); SAMSUNG-1008, Figure 1.8(b) (modified, annotated); SAMSUNG-1016, Figure 1.8(b) (modified); SAMSUNG-1003, ¶¶97-104

Note that while Smith illustrates the same applications running on both VMs, a POSITA would have understood that each VM can run a different set of applications, e.g., as Smith illustrates in FIG. 1.8(c), which shows “Apps 1” and “Apps 2.” SAMSUNG-1003, ¶¶98-104.

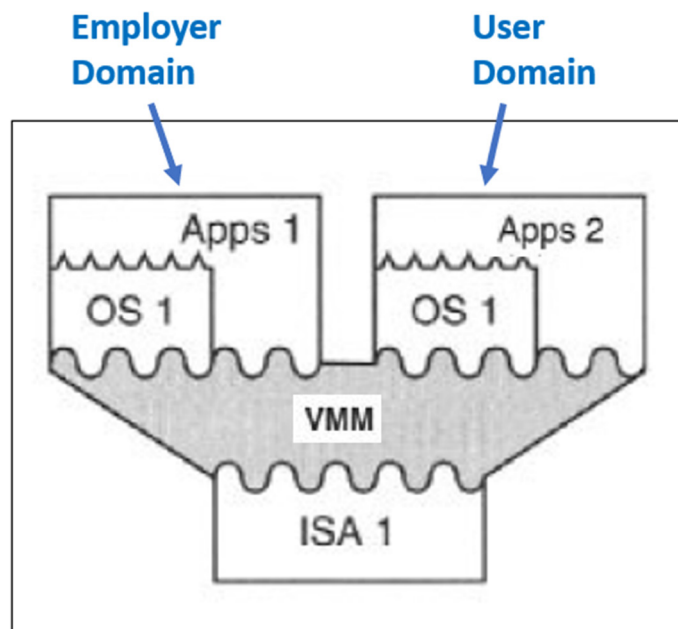


SAMSUNG-1008, Figure 1.8(c) (modified); SAMSUNG-1016, Figure 1.8(c)
(modified)

In operation, per Owen, OBSD's objects (e.g., applications, communication pipes) are assigned to domains, and their use is "restricted to other domain members." SAMSUNG-1005, Abstract, [0030], [0041]; SAMSUNG-1003, ¶105. For example, applications in the employer domain (e.g., CRM) communicate over a secure communication pipe using a gateway through a corporate security firewall. SAMSUNG-1005, [0042], [0046]; §III.A.1(a); SAMSUNG-1003, ¶105. Per Bartlett, when data transmission from the application begins, if the secure communication pipe through the gateway has not yet been established, OBSD automatically creates a VPN connection. SAMSUNG-1006, [0058]; §III.A.1(b); SAMSUNG-1003, ¶105. The transmission from the application is then directed to the employer's servers using the VPN. SAMSUNG-1005, [0042], [0046]; SAMSUNG-

1006, [0058]; SAMSUNG-1003, ¶105.

Further, to isolate domains, OBSD's VMM creates a VM for each domain. SAMSUNG-1003, ¶106. Applications assigned a domain run in the corresponding domain, as illustrated below. *Id.*



SAMSUNG-1008, FIG. 1.8(b) (modified); SAMSUNG-1016, FIG. 1.8(b) (modified); SAMSUNG-1003, ¶106

A POSITA would have had a reasonable expectation of success in combining components of OBD and Smith because they describe complementary functions of a computing device. §§III.A.1(a)-(c); §III.B.1(a); SAMSUNG-1003, ¶107. For example, per Owen, in ODB “[p]lacing an object in a domain means that the object is trusted, and that its use can be restricted to other domain members,” and

using a VM (per Smith) “provide[s] isolation and enhanced security.” SAM-SUNG-1005, [0030]; SAMSUNG-1008, 5; SAMSUNG-1016, 56; §III.A.1(c); SAMSUNG-1003, ¶107. Thus, a POSITA would have understood that Smith’s VM isolation enforces Owen’s domain restrictions. SAMSUNG-1003, ¶107.

The combination of OBD and Smith would have required only routine programming knowledge well within the skill of a POSITA before the Critical Date. SAMSUNG-1003, ¶108. Indeed, implementing the combination would have amounted to using known techniques (adding a VMM, per Smith, to the OBD mobile device) to improve similar systems (mobile devices) and combining prior art elements according to known methods to yield the predictable results described above. SAMSUNG-1003, ¶108; *KSR*, 550 U.S. at 417.

Finally, the elements of the resulting OBSD would each perform functions they performed prior to combination—Owen limiting an application’s traffic to communication pipes in the application’s domain, Bartlett establishing a VPN and using the VPN to ensure secure transmission of the data, and Smith providing a VMM to isolate domains. SAMSUNG-1003, ¶109. A POSITA would have therefore expected success when combining OBS and Smith’s teachings. SAMSUNG-1003, ¶109.

2. Analysis

[1.1]

To the extent the preamble is limiting, OBSD renders it obvious. *Supra* §III.A.2[1.1].

[1.2.i]

OBSD renders [1.2.i] obvious. *Supra* §III.A.2[1.2.i].

[1.2.ii]

OBSD renders [1.2.ii] obvious. *Supra* §III.A.2[1.2.ii].

[1.2.iii]

OBSD renders [1.2.iii] obvious. *Supra* §III.A.2[1.2.iii].

[1.3.i]

OBSD renders [1.3.i] obvious. SAMSUNG-1003, ¶¶197-202.

As described in §III.A.2[1.2.ii], applications are assigned to application environments on the wireless end-user device based on secure stored information indicative of an application-specific execution environment policy. §III.A.2[1.2.ii]; SAMSUNG-1003, ¶198. For example, Owen explains that “domain membership control policies (‘domain policies’) [] specify the criteria used to determine into which domain a software application, property, or other information should be

placed." SAMSUNG-1005, [0024]. For instance, Owen teaches running some applications in the employer domain, e.g., "the company directory and a Customer Relationship Management (CRM) software application," and FIG. 3 further illustrates a corporate messaging application. SAMSUNG-1005, [0034]-[0036], [0041]-[0043], FIG. 3; SAMSUNG-1003, ¶198.

In OBSD, each domain is assigned to a VM. §III.B.1(b); SAMSUNG-1003, ¶200. Per Smith, VMs provide "a set of multiple virtual systems," which include "a full execution environment that can simultaneously support a number of processes potentially belonging to different users." SAMSUNG-1008, 3, 9; SAMSUNG-1016, 54, 62. Therefore, in OBSD, a first set of multiple device application instances (e.g., the CRM and corporate message applications) run only in a first secure application environment (e.g., the VM hosting the employer domain). SAMSUNG-1005, [0034], [0042], FIG. 3; SAMSUNG-1003, ¶200.

Per Smith, VMs are secure and provide isolation. SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, ¶201. For example, Smith explains that "[p]erhaps the most important feature of today's system VMs is that they provide a secure way of partitioning major software systems that run concurrently on the same hardware platform" such that "[s]oftware running on one guest system is isolated from software running on other guest systems," and a "sys-

tem VM provides a sandbox that isolates one system environment from other environments, thus ensuring a level of security that may not be provided by a single operating system.” SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, ¶201. Therefore, since Smith teaches that VMs are secure, a POSITA would have understood that the OBSD execution environment is secure. SAMSUNG-1003, ¶201.

Therefore, OBSD renders obvious allowing a first set of multiple device application instances (e.g., CRM, corporate messaging) to run only in a first secure application environment (the VM hosting the employer domain) on the wireless end-user device based on the secure stored information indicative of an application-specific execution environment policy (Owen’s domain policies). SAMSUNG-1003, ¶202.

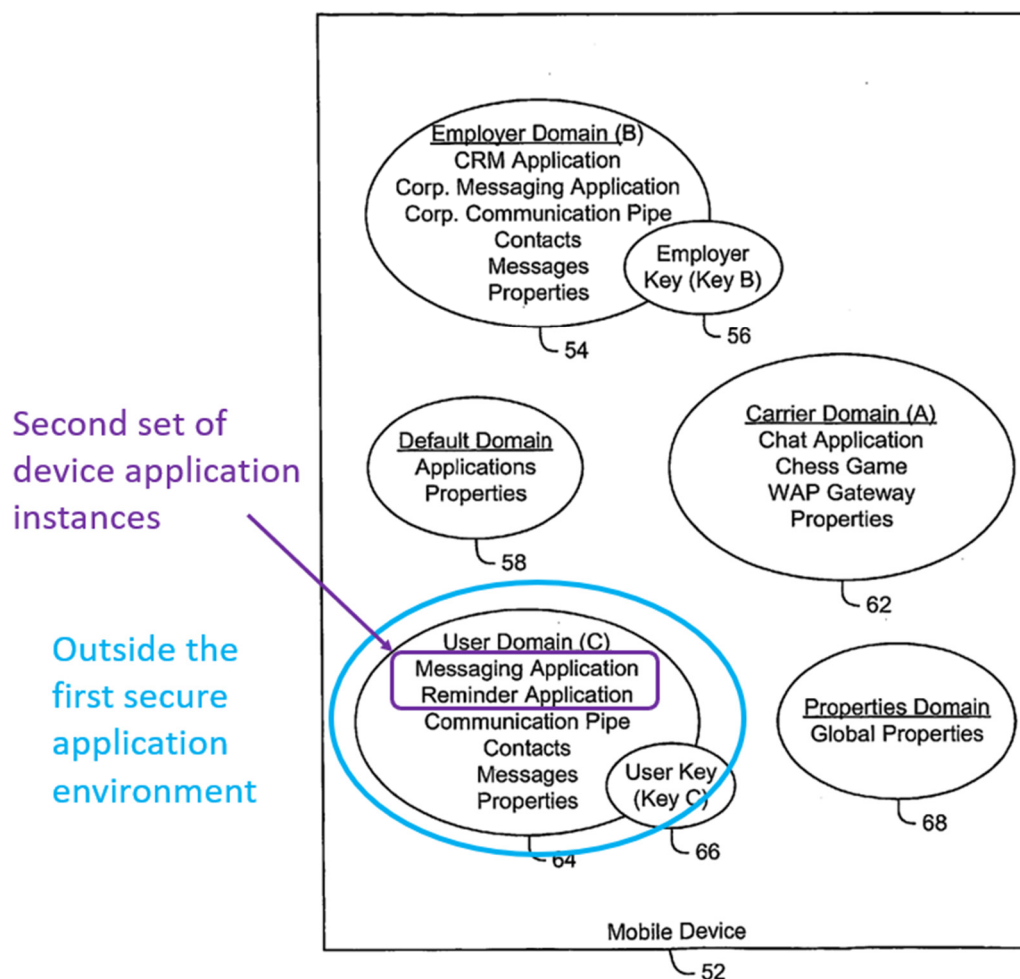
[1.3.ii]

OBSD renders [1.3.ii] obvious. SAMSUNG-1003, ¶¶203-204.

As described in §III.A.2[1.2.ii], applications are assigned to domains. §III.A.2[1.2.ii]; SAMSUNG-1003, ¶204. For example, Owen teaches running some applications in the user domain, e.g., a “birthday reminder program” and “personal messaging,” and FIG. 3 further illustrates a “messaging application.” SAMSUNG-1005, [0033], [0035]-[0036], [0041], [0043]. Since the user domain is outside the employer domain (which is hosted in a VM providing the first secure

application environment), Owen renders obvious allowing a second set of device application instances to run only outside of the first secure application environment (e.g., in the VM hosting the user domain which is outside of the VM hosting the employer domain). §III.B.1(b); SAMSUNG-1003, ¶204; SAMSUNG-1005, FIG.

3.



SAMSUNG-1005, FIG. 3 (annotated)

[1.4.i]

OBSD renders [1.4.i] obvious. *Supra* §III.A.2[1.4.i].

[1.4.ii]

OBSD renders [1.4.ii] obvious. *Supra* §III.A.2[1.4.ii].

[2.1]

OBSD renders [2.1] obvious. *Supra* §III.A.2[2.1].

[2.2]

OBSD renders [2.2] obvious. *Supra* §III.A.2[2.2].

C. Ground 2: Lambert-Rao-Smith renders claims 1-2 obvious

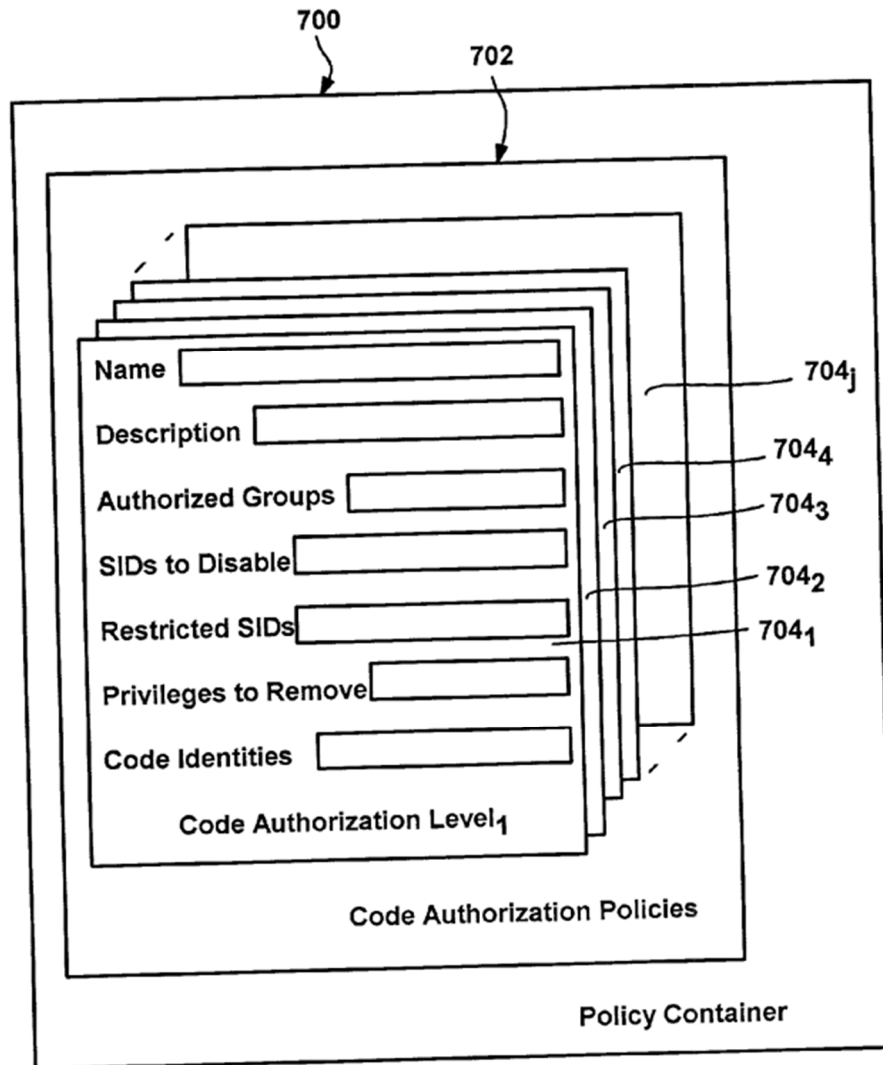
1. Lambert-Rao-Smith

(a) Lambert

Lambert is “directed to restricting what executable code can do, including limiting its access to resources.” SAMSUNG-1007, [0037]. Lambert describes controlling “software execution by identifying and classifying software, and locating a rule and associated security level for executing executable software.” SAMSUNG-1007, Abstract. For example, “[i]n an enterprise environment, the policy may be set for sites, domains, organizational units, groups and/or users via group policy technology.” SAMSUNG-1007, [0012]. In another example, Lambert “provides a transparent way to run untrusted (and hence potentially harmful) code in a constrained environment in which such code is not allowed to perform certain operations, such as operations that deal with system data or user data.” SAM-

SUNG-1007, [0009]. Lambert explains it can “be used with software-fault isolation on a per-thread basis, or with a virtual machine where restrictions are determined from the stack of classes currently executing.” SAMSUNG-1007, [0037].

In Lambert, a “software restriction policy comprising a set of rules is provided to correlate the classification information for any software to a security level that may restrict or prevent the software from running.” SAMSUNG-1007, [0011], FIGS. 6, 7; SAMSUNG-1003, ¶111, §VII.A. Such rules can be included in a “policy container,” as illustrated in Lambert’s FIG. 7. SAMSUNG-1007, [0079], FIG. 7, claim 54; SAMSUNG-1003, ¶111.



SAMSUNG-1007, FIG. 7

Lambert explains that “executable code is identified and classified, and then a rule corresponding to the classification (or if none correspond, a default rule) automatically and transparently determines the software’s ability to execute.” SAMSUNG-1007, [0008]. Lambert’s “rule specifies for the classification that is determined for any given software module (e.g., software file), [that] the software may execute in a normal (unrestricted) execution environment (context), in a restricted

execution environment, or the software may be prevented from running at all.”

SAMSUNG-1007, [0008]. Additionally, “an administrator can select any software file, obtain a hash value from its contents, and enter and associate a rule (that specifies a security level) with that hash value.” SAMSUNG-1007, [0055]; *see id.*, [0056].

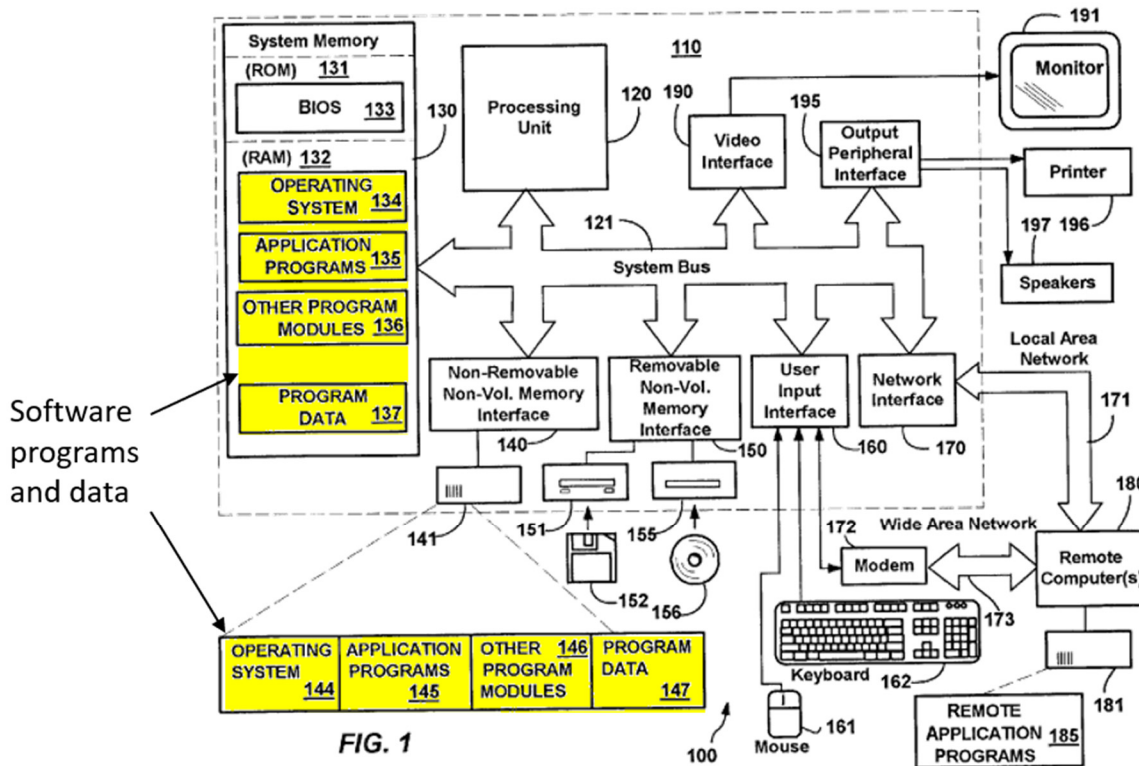
Lambert further describes “network access policy data” that can be “used to generate network quality of service (QoS) restrictions” and prevent “a particular program or class thereof” from “obtaining network access or otherwise [be] restricted to some reduced level of network access.” SAMSUNG-1007, [0077].

Lambert notes it “can combine these and other ways (e.g., restricted token, job object and/or QoS restrictions) of restricting the execution environment of software programs.” *Id.*

Lambert’s “rules that control execution for a given machine or user may be maintained in a restriction policy, e.g., locally maintained and/or in a group policy object.” SAMSUNG-1007, Abstract, [0080]; SAMSUNG-1003, ¶114.

Lambert describes “an exemplary system for implementing the invention [that] includes a general purpose computing device,” e.g., a computer, and “configurations that may be suitable for use with the invention include, but are not limited to, personal computers, server computers, hand-held or laptop devices,” etc., including hardware and software components as shown in FIG. 1. SAMSUNG-1007,

[0027]-[0035], FIG. 1 ; SAMSUNG-1003, ¶115.



SAMSUNG-1007, FIG. 1 (annotated)

(b) Rao

Rao discloses a “remote access architecture” for “peer-to-peer communications and remote access connectivity” on a computing device, which can be “any workstation, desktop computer, laptop or notebook computer, server, handheld computer, mobile telephone, any other computer, or other form of computing or telecommunications device.” SAMSUNG-1009, Abstract, [0098], [0130]. The system (e.g., Figure 1A, below) includes clients (e.g., computing devices 102) that communicate over a network to other devices, such as servers and server farms.

Id., [0086]-[0096].

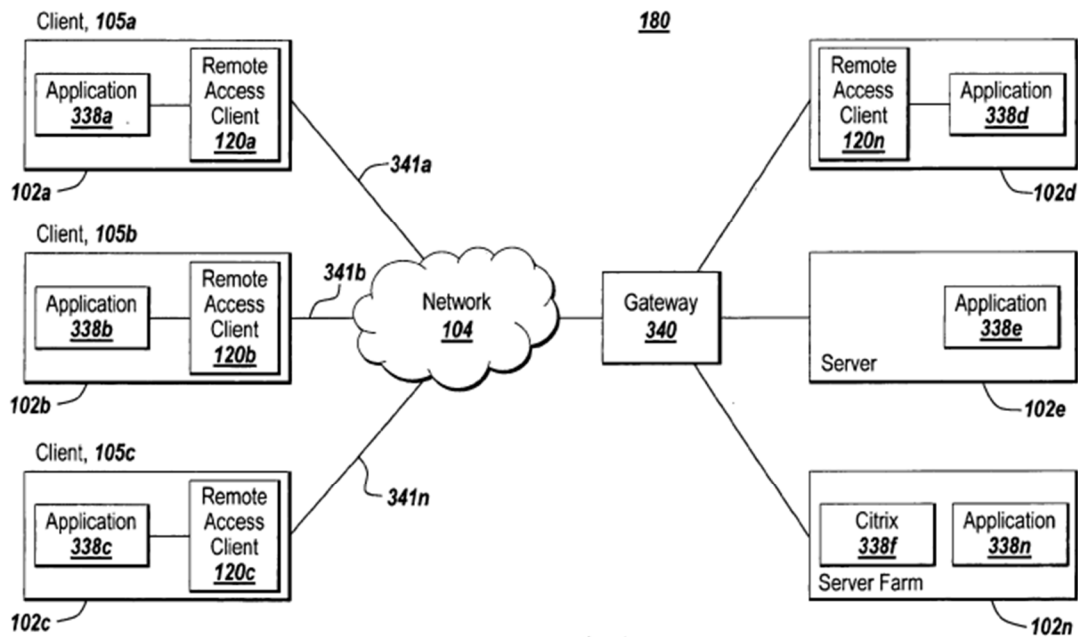
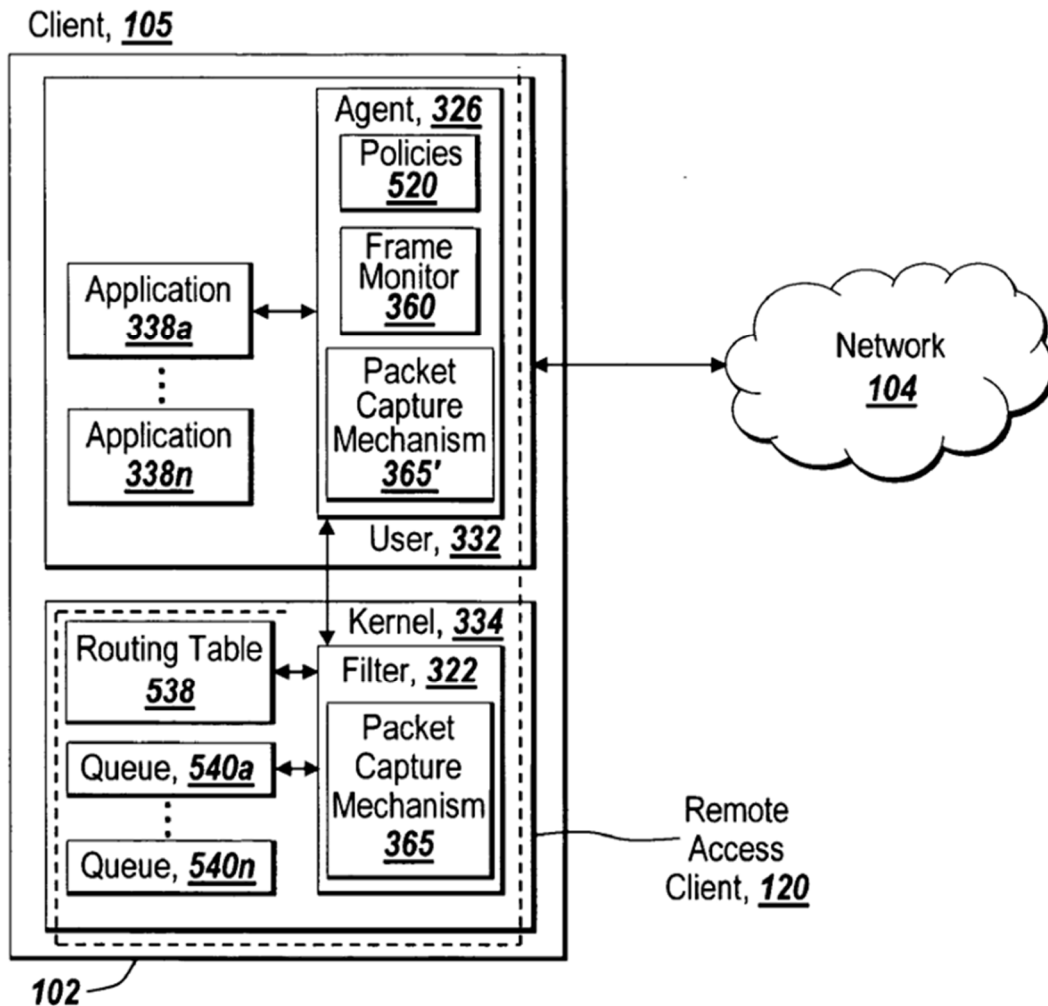


Fig. 1A

SAMSUNG-1009, FIG. 1A

Further, each client 105a-105n “includes a remote access client” (RAC) and “communicate[s] over the network 104 to the gateway 340 via a tunneling or gateway connections 341a-341n using any type and/or form of suitable tunneling or gateway protocols.” SAMSUNG-1009, [0087]. For example, Rao’s “system for transmitting packets” “includes a first device and a second device” and the “first device has an application that generates a first and second packet.” *Id.*, [0030]. A filtering process “intercepts the first and second packets from the application and

forwards the intercepted packets to the tunnel process.” *Id.* Then, the “tunnel process requests the opening of a TCP connection between the first device and a second device” and “forwards the first and second packets as payloads in a first and second TCP packet to the second device.” *Id.*



SAMSUNG-1009, FIG. 5A

Rao’s RAC “may have one or more policies 520 for specifying client-side prioritization of network communications related to applications 338a-338n,” and

“may be specified by the name of the application 338a-338n and/or the type of application 338a-338n.” SAMSUNG-1009, [0182]. Rao explains that “the client 105 may intercept network packets of one or more applications” and “determine[s] priorities and appl[ies] any priority based policies.” SAMSUNG-1009, [0185], [0187]. RAC’s “filter 322 may place, arrange, or coordinate network packets into queues 540a-540n in support of the priorities,” e.g., “[o]ne ordinarily skilled in the art will recognize and appreciate that the network packets may be arranged in various priority queues 540a-540n, such as high, medium, low, or by any other granularity.” SAMSUNG-1009, [0191].

Rao explains that “all network packets for an application 338a-338n are placed in a queue 540a-540n associated with the application 540a-540n.” SAMSUNG-1009, [0192].

Rao states that “the network packets are communicated from the queues 540a-540n according to the determined priorities for the network packets.” SAMSUNG-1009, [0194]. For example, “the network packets are organized into priority queues 540a-540b so that the network packets of the highest priority queue 540a-540n are communicated first, and then the next highest priority queue 540a-540n second, and so on.” *Id.* Transmitting packets from the client based on priority “may compliment Quality of Service (QOS) networks where network traffic

prioritization occurs in intermediary network devices, such as switches and routers.” *Id.*, [0184].

(c) Combination

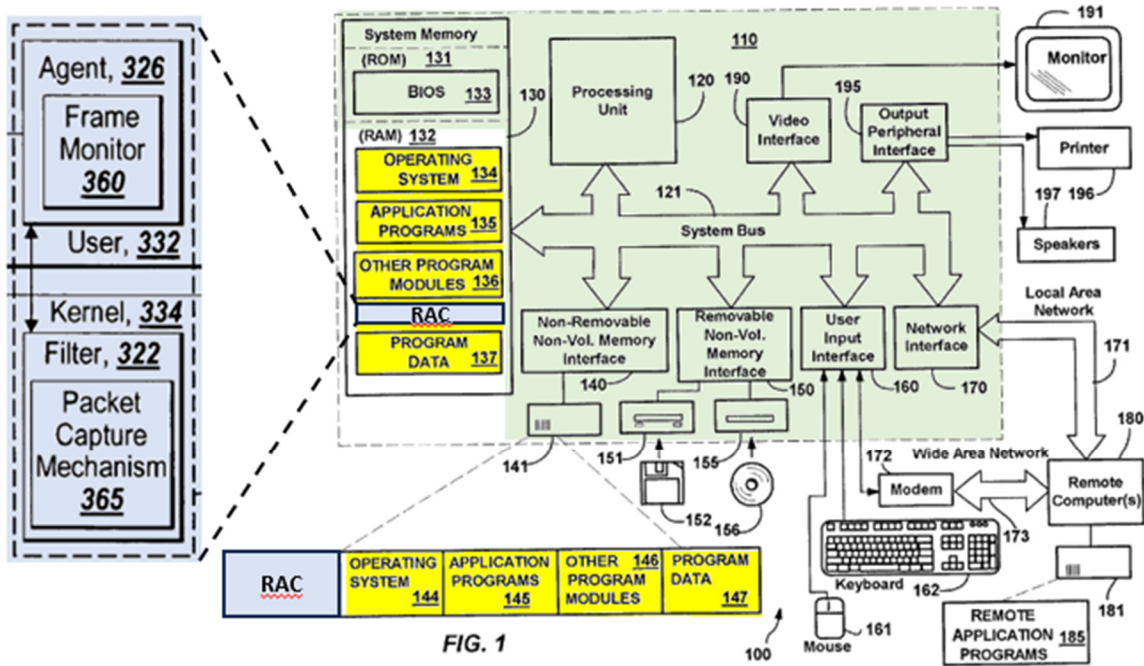
As described above, Lambert’s policies determine whether “software may execute in a normal (unrestricted) execution environment (context), in a restricted execution environment, or the software may be prevented from running at all.” SAMSUNG-1007, [0008]; §III.C.1(a). Lambert further explains that “network access policy data” can be “used to generate network quality of service (QoS) restrictions” and prevent “a particular program or class” from “obtaining network access or otherwise [be] restricted to some reduced level of network access.” SAMSUNG-1007, [0077].

To the extent Lambert does not provide details regarding how network quality of service (QoS) restrictions are generated for particular programs (or classes), a POSITA would have looked to references such as Rao, which describes a RAC that assigns packets to priority queues based on the application, or program, that generated the packet. §§III.C.1(a)-(b); SAMSUNG-1003, ¶122 (discussing Lambert’s similar treatment of programs and applications). For example, Rao’s RAC “provides for the intelligent and client centric prioritization of application network communications on a client based on the type and/or priority of an application.” SAMSUNG-1009, [0179]. In Rao, “all network packets for an application 338a-

338n are placed in a queue 540a-540n associated with the application 540a-540n” and then “communicated from the queues 540a-540n according to the determined priorities for the network packets.” SAMSUNG-1009, [0192], [0194]. Thus, a POSITA would have understood or found obvious that Rao implements QoS restrictions for applications (programs) by assigning packets from an application to a queue based on the application’s priority and communicating packets from the queue based on the priority. *Id.*; SAMSUNG-1003, ¶122 (citing SAMSUNG-1012, 4).

The resulting Lambert-Rao Device (LRD) comprises software components (“LRD software instance(s)”), e.g., a RAC (per Rao, highlighted in blue) and operating system, application programs, etc. (per Lambert, highlighted in yellow).

§§IX.F-G. LRD executes on the hardware components of a computing system (the device hardware, described further below), including a “network interface or adapter 170” and/or “a modem 172” (per Lambert, highlighted in green). SAMSUNG-1003, ¶123.



SAMSUNG-1007, FIG. 1 (modified); SAMSUNG-1009, FIG. 1C (modified);

SAMSUNG-1003, ¶123

Additionally, Lambert explains that it “may also be used with software-fault isolation on a per-thread basis, or with a *virtual machine* where restrictions are determined from the stack of classes currently executing.” SAMSUNG-1007, [0037]. “[A]lthough restricted tokens are generally described herein, as will be understood, the present invention is not limited to restricted tokens, but can be implemented in virtually any security model in which it is feasible to modify executable code’s ability to operate based on information known about the executable code, e.g., its source, path information or hash.” *Id.*

Since Lambert “is not limited to restricted tokens, but can be implemented in virtually any security model in which it is feasible to modify executable code’s ability to operate based on information known about the executable code” and explicitly enumerates virtual machines as one example, a POSITA would have looked to similar references that provide further details regarding the operation and use of virtual machines. SAMSUNG-1003, ¶125. One such example is Smith, which teaches that “[p]erhaps the most important feature of today’s system VMs is that they provide a secure way of *partitioning major software systems that run concurrently on the same hardware platform*” such that “[s]oftware running on one guest system is isolated from software running on other guest systems,” and a “system VM provides a sandbox that isolates one system environment from other environments, thus ensuring a level of security that may not be provided by a single operating system.” SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, 125. Further, Lambert teaches that “the present invention may also be used with software-fault isolation,” and Smith explains both that “[r]unning each application in its own virtual machine *increases the robustness of the system*,” and that “if security on one guest system is compromised or if the *guest OS suffers a failure*, the software running on other guest systems is not affected.” SAMSUNG-1007, [0037]; SAMSUNG-1008, SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507. Thus, a POSITA would have recognized that the

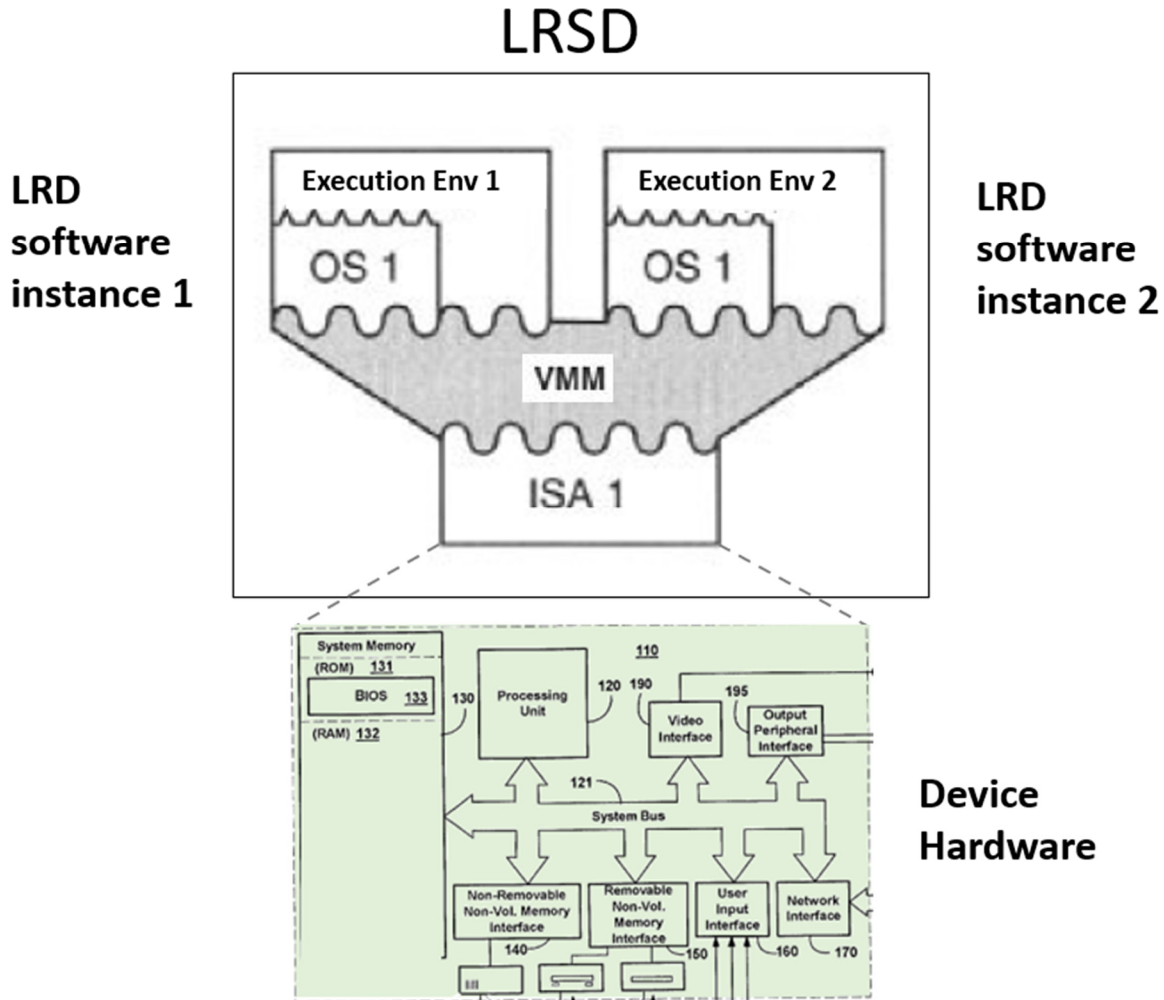
use of VMs (per Smith) advantageously provides software-fault isolation (as described in both Lambert and Smith). *Id.*; SAMSUNG-1003, ¶125.

The resulting Lambert-Rao-Smith Device (LRSD) comprises LRD software instances running on device hardware (e.g., per Lambert) and, per Smith, LRSD further includes a virtual machine monitor (VMM) that provides the virtualizing software, thereby adding “a layer of software to a real machine to support the desired virtual machine’s architecture.” §§III.C.1(a),(c); SAMSUNG-1008, 11; SAMSUNG-1016, 64; SAMSUNG-1003, ¶126.

Execution environments are components of LRD software (per Lambert), which are implemented as individual system VMs (per Smith and Lambert) so that “trusted content [is] run in a different execution environment/context from the other software.” SAMSUNG-1007, [0037], [0054]; SAMSUNG-1008, 18, 369-371; SAMSUNG-1016, 72, 505-507; SAMSUNG-1003, ¶127. As a result, per Smith, LRSD is “transformed so that it appears to be [] a set of multiple virtual systems,” each containing a virtual instance of LRD software, as illustrated below. SAMSUNG-1008, 3-4, 9; SAMSUNG-1016, 54-55, 62; SAMSUNG-1003, ¶127.

Thus, using virtualization, as described in Smith, LRSD is a physical wireless-end user device (e.g., device hardware, per Lambert) that hosts multiple LRD software instances, one per execution environment, as illustrated below. SAMSUNG-1008, 18, 369-371; SAMSUNG-1016, 72, 505-507; SAMSUNG-1003,

¶¶128-129 (citing SAMSUNG-1008; SAMSUNG-1013); §VII.B.



SAMSUNG-1008, Figure 1.8(b) (modified); SAMSUNG-1016, Figure 1.8(b) (modified); SAMSUNG-1007, FIG. 1 (modified); SAMSUNG-1003, ¶128

Since each of LRSD's VMs hosts a complete LRD software instance, each VM includes the software components of a LRD, e.g., an execution environment (per Lambert and Smith) and a RAC (a component of LRD, per Rao), which pro-

vides prioritized network queues to the applications assigned to the execution environment. §§III.C.1(a)-(b); SAMSUNG-1003, ¶130.

In operation, per Lambert, LRSD's "executable code is identified and classified, and then a rule corresponding to the classification" "automatically and transparently determines the software's ability to execute." SAMSUNG-1007, [0008]; SAMSUNG-1003, ¶131. Further, per Lambert, LRSD's "rule specifies for the classification that is determined for any given software module (e.g., software file), [that] the software may execute in a normal (unrestricted) execution environment (context), in a restricted execution environment, or the software may be prevented from running at all." SAMSUNG-1007, [0008]; SAMSUNG-1003, ¶131. As described above, separate VMs host the execution environments to which applications are assigned (e.g., normal, restricted), thereby isolating trusted applications from untrusted applications, and vice-versa. SAMSUNG-1008, 369-371; SAMSUNG-1016, 505-507; SAMSUNG-1003, ¶131.

Additionally, per Rao, when an application intends to transmit data, the packets sent by the application are assigned to the priority queue associated with the application. SAMSUNG-1007, [0182], [0184], [0191]-[0193]; SAMSUNG-1003, ¶132. Since packets are transmitted according to queue priority (which is based on application priority), per Rao, each LRD software instance provides per-application quality of service (QoS) restrictions, as described by Rao and Lambert.

SAMSUNG-1007, [0077]; SAMSUNG-1009, [0194]; SAMSUNG-1003, ¶132.

Further, per Rao, each LRD software instance includes a filtering process that “intercepts the first and second packets from [an] application and forwards the intercepted packets to the tunnel process.” SAMSUNG-1009, [0030]; SAMSUNG-1003, ¶133. Then, the “tunnel process requests the opening of a TCP connection between the first device and a second device,” and “forwards the first and second packets as payloads in a first and second TCP packet to the second device.” SAMSUNG-1009, [0030]; SAMSUNG-1003, ¶133. Subsequently, per Rao, the LRD software instance “communicate[s] over the network 104 to the gateway 340 via a tunneling [] connection[] 341a-341.” SAMSUNG-1009, [0087].

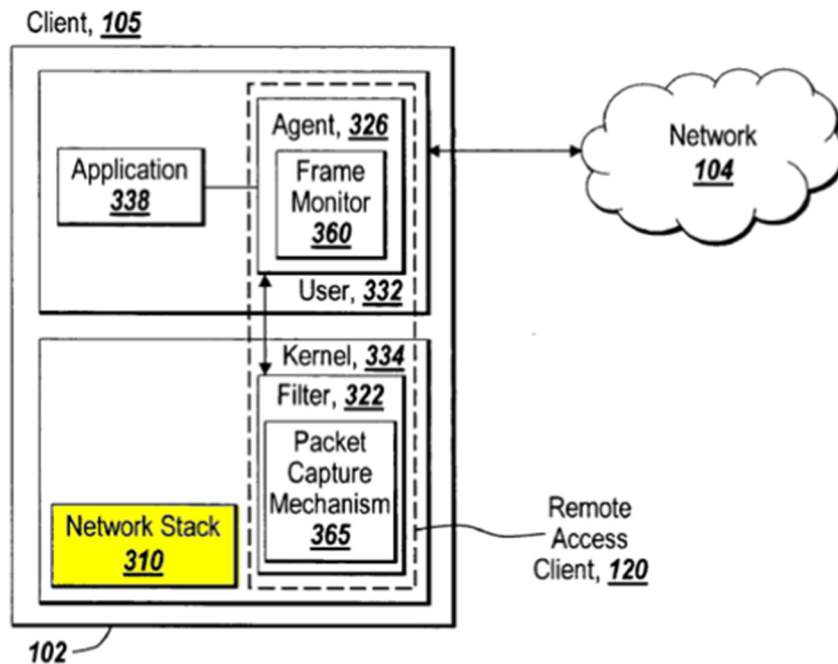
Since, as described above, each LRD software instance (including the virtual LRD’s execution environment) runs in its own VM, each execution environment has its own instance of a RAC. SAMSUNG-1003, ¶134. For example, per Smith, since each “system VM provides a sandbox that *isolates one system environment from other environments*,” inbound and outbound network traffic to/from applications in each execution environment traverses the tunnel created by the RAC (per Rao) in that execution environment (per Smith). SAMSUNG-1007, 482; SAMSUNG-1009, [0087]; SAMSUNG-1003, ¶134. As Dr. McDaniel explains, as a result of the isolation described by Smith, a VM is isolated from all tunnels except

the tunnel that VM created, and therefore network traffic can only traverse the particular tunnel created by the VM. SAMSUNG-1007, 482; SAMSUNG-1003, ¶134 (citing SAMSUNG-1008, 18, 23).

A POSITA would have had a reasonable expectation of success in combining components of a mobile device, as described in Lambert, with a RAC (per Rao) because both describe complementary functions of a mobile device, such as a hand-held or laptop device. SAMSUNG-1007, [0026]-[0035]; §III.C.1(a)-(b); SAMSUNG-1003, ¶135. For example, per Lambert, LRD assigns software to execution environment and operates “in a networked environment using logical connections to one or more remote computers, such as a remote computer 180” over logical connections such as “local area network (LAN) 171 and a wide area network (WAN) 173, but may also include other networks.” SAMSUNG-1007, [0034]; SAMSUNG-1003, ¶135. Data transmitted from LRD in the networked environment are processed by RAC, which performs application-specific prioritization (per Rao) before transmitting the data. SAMSUNG-1009, [0179]-[0195]; SAMSUNG-1003, ¶135.

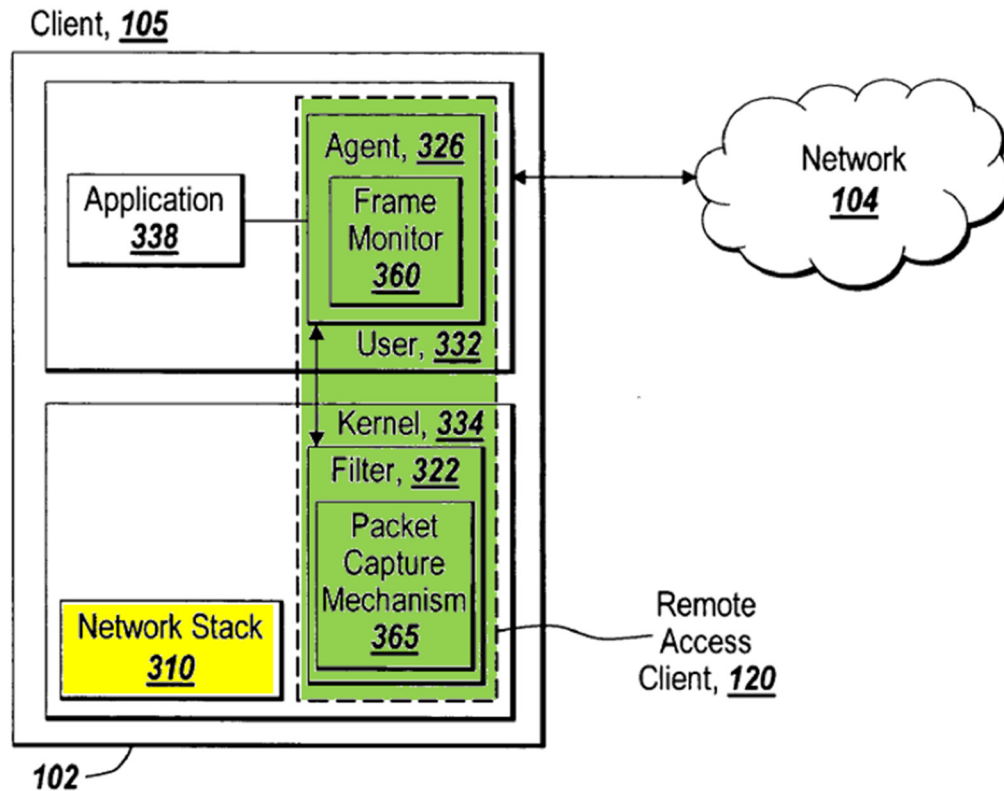
Further, to the extent Lambert does not describe the networking stack used to access such networks, which are “commonplace in offices, enterprise-wide computer networks, intranets and the Internet,” a POSITA would have understood from

references such as Rao that such stacks are commonly present in networked devices. SAMSUNG-1007, [0034]; SAMSUNG-1009, [0100]-[0101], [0152]-[0154], [0196]-[0198], FIGS. 1C, 3A, 6A; SAMSUNG-1003, ¶136.



SAMSUNG-1009, FIG. 1C (annotated)

Additionally, at least from Rao, a POSITA would have understood that a RAC (per Rao) can be used to augment the function of a network stack, e.g., to “manage[] and control[] network communication prioritizations on the client.” SAMSUNG-1009, [0080] [0087], [0100], [0179]-[0195], FIG. 1C; SAMSUNG-1003, ¶137.

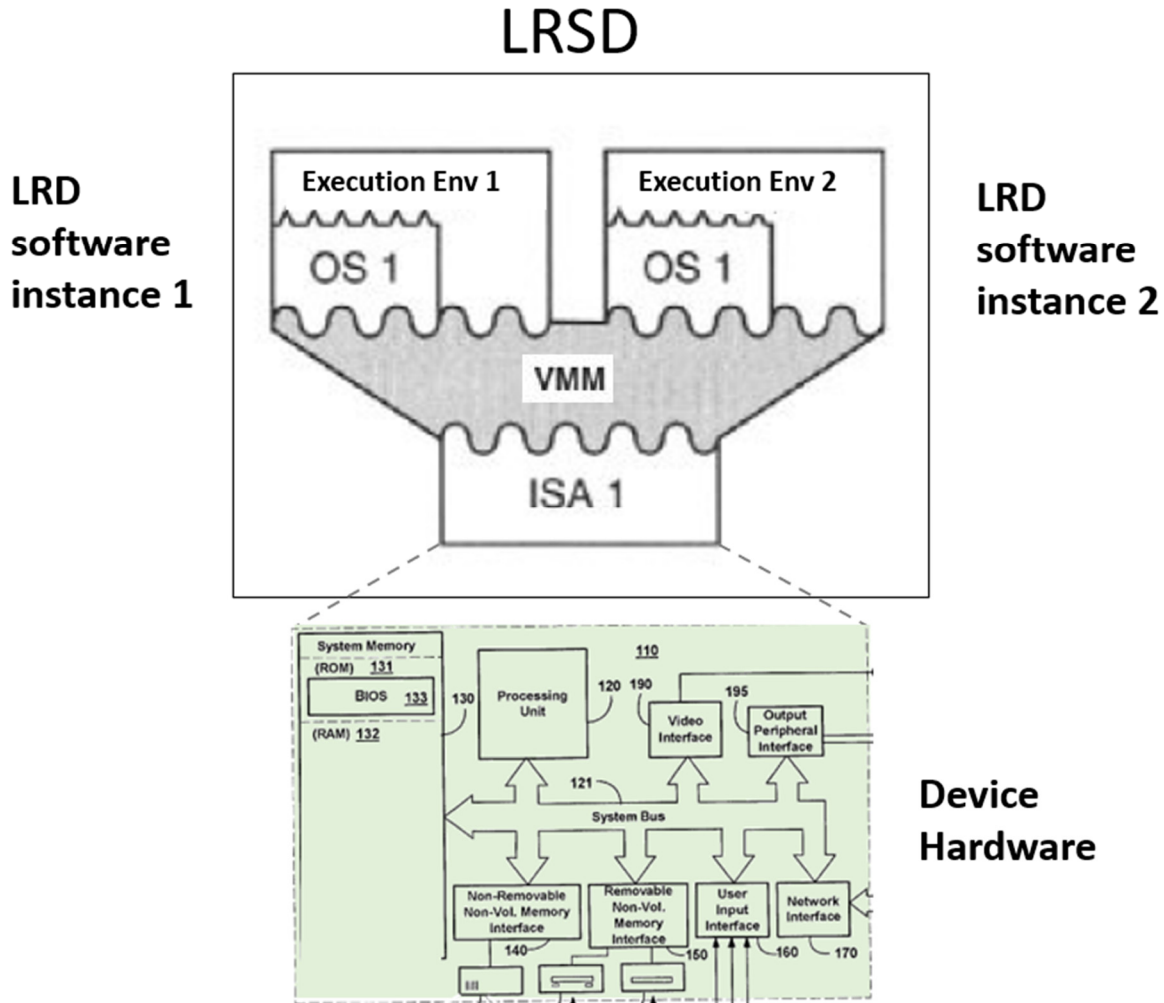


SAMSUNG-1009, FIG. 1C (annotated)

Thus, the elements of the resulting LRD software instances would each perform functions they performed prior to combination—Lambert assigning software to execution environments, and Rao transmitting the packets according to an application-specific priority. SAMSUNG-1003, ¶138. A POSITA would have therefore expected success when combining Lambert’s and Rao’s teachings. SAMSUNG-1003, ¶138.

Additionally, the elements of the LRSD (e.g., LRD software instances, a VMM and device hardware) would each perform functions they performed prior to

combination—Lambert assigning software to execution environments, Rao transmitting the packets according to an application-specific priority, and Smith virtualizing the hardware by providing a VMM. SAMSUNG-1003, ¶139. A POSITA would have therefore expected success when combining the teachings of Lambert, Rao, and Smith. SAMSUNG-1003, ¶139. A POSITA would have had a reasonable expectation of success in combining LRD with a VMM (per Smith) to form LRSD (illustrated below) because they provide complementary functions -- the VMM merely providing a virtual platform on which LRD's software instances executes. SAMSUNG-1003, ¶139.



SAMSUNG-1008, Figure 1.8(b) (modified); SAMSUNG-1016, Figure 1.8(b) (modified); SAMSUNG-1007, FIG. 1 (modified); SAMSUNG-1003, ¶139

Both the combination of Lambert and Rao (to create LRD software instances) and the combination device hardware and a VMM (per Smith) would have required only routine programming knowledge well within the skill of a POSITA before the Critical Date. SAMSUNG-1003, ¶140. Indeed, implementing the combination of Lambert and Rao would have amounted to using known techniques to

improve similar systems (mobile device networking) and combining prior art elements according to known methods to yield the predictable results described above. SAMSUNG-1003, ¶140; *KSR*, 550 U.S. at 417. Further, implementing the combination of device hardware and Smith would have amounted to using known techniques (e.g., using a VMM to run software in a virtual machine that abstracts a physical device) to improve similar systems (mobile device networking) and combining prior art elements according to known methods to yield the predictable results described above. SAMSUNG-1003, ¶140; *KSR*, 550 U.S. at 417.

2. Analysis

[1.1]

To the extent the preamble is limiting, LRSD renders it obvious. SAMSUNG-1003, ¶209. For example, Lambert describes a “*method* that automatically, transparently and *securely controls software execution* by identifying and classifying software, and locating a rule and associated security level for executing executable software.” SAMSUNG-1007, Abstract. Lambert further explains that the “invention is operational with numerous other general purpose or special purpose computing system environments or configurations,” e.g., “personal computers, server computers, *hand-held or laptop devices*,” “and the like.” SAMSUNG-1007, [0027]. Lambert further explains that its devices include communication media “such as a wired network or direct-wired connection, and *wireless media*

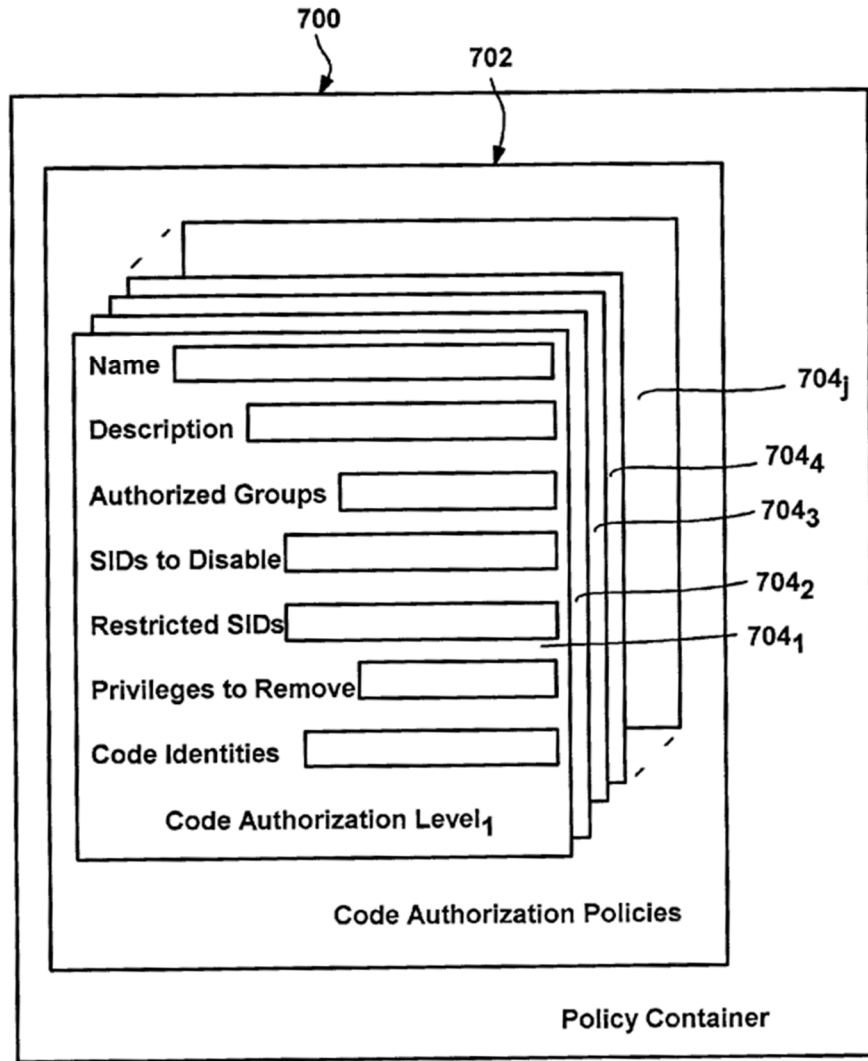
such as acoustic, RF, infrared and other wireless media.” SAMSUNG-1007,

[0030]. A POSITA would have recognized or found obvious that hand-held and laptop devices with wireless media are wireless end-user devices. SAMSUNG-1003, ¶209.

[1.2.i]

LRSD renders [1.2.i] obvious. SAMSUNG-1003, ¶¶210-213.

Per Lambert, LRSD stores secure policy information. SAMSUNG-1003, ¶¶211. For example, in Lambert, a “software restriction policy comprising a set of rules is provided to correlate the classification information for any software to a security level that may restrict or prevent the software from running.” SAMSUNG-1007, [0011]. Thus, per Lambert, LRSD includes secure policy information, e.g., a set of rules that correlate the classification information for any software to a security level that may restrict or prevent the software from running. SAMSUNG-1007, [0011]; SAMSUNG-1003, ¶211. Such rules can be included in a “policy container,” as illustrated in Lambert’s FIG. 7. SAMSUNG-1007, [0079], FIG. 7, claim 54; SAMSUNG-1003, ¶211.

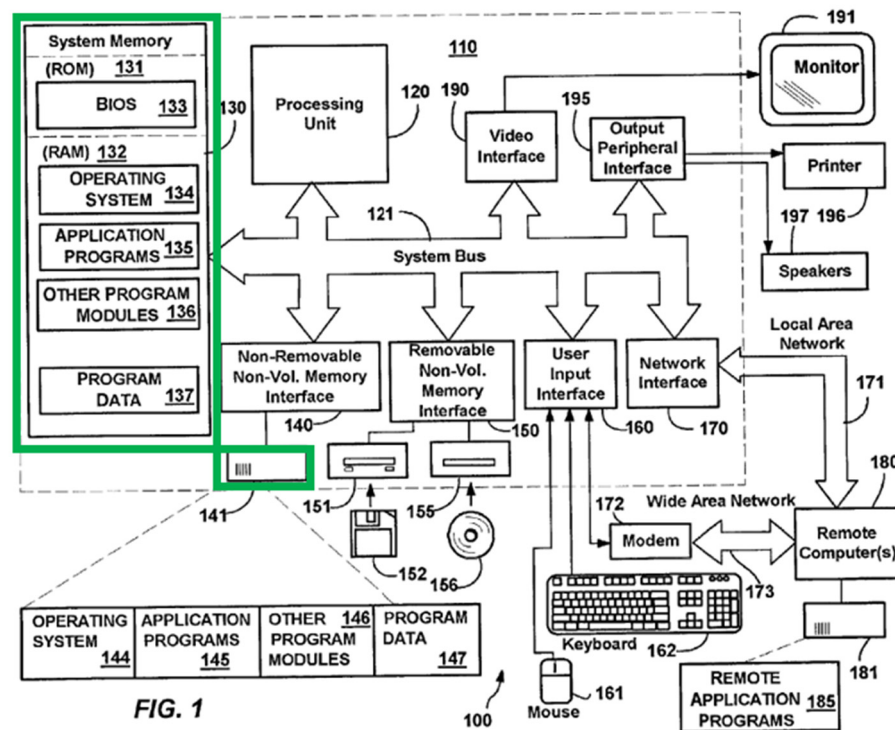


SAMSUNG-1007, FIG. 7

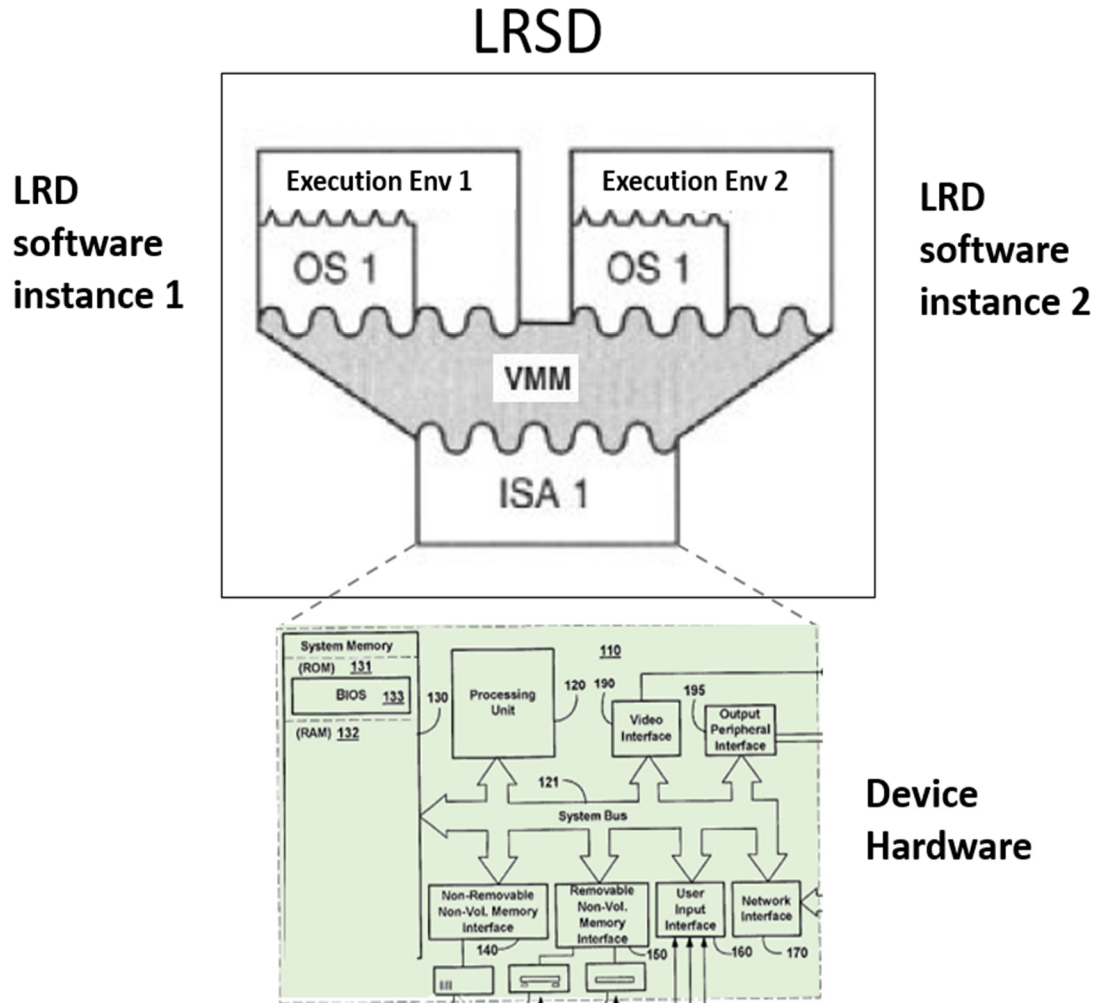
Additionally, per Lambert, in LRSD, “rules that control execution for a given machine or user may be maintained in a restriction policy, e.g., locally maintained and/or in a group policy object.” SAMSUNG-1007, Abstract. Further, Lambert provides an example in which “[r]ules can also be set for a user or machine locally and *maintained in the local system registry*, e.g., under HKEY_LOCAL_MACHINE and HKEY_CURRENT_USER.” SAMSUNG-1007, [0064];

see also SAMSUNG-1007, [0060], [0078], claim 37; SAMSUNG-1003, ¶212.

Since Lambert teaches that the policy object can be locally maintained (e.g., in the system registry), a POSITA would have understood or found obvious that the policy objects are stored in Lambert's system memory 130 and/or hard disk drive 141 "which can be used to store the desired information and which can accessed by the computer." SAMSUNG-1007, [0030], [0032], [0064]; FIG. 1; SAMSUNG-1003, ¶212.



SAMSUNG-1005, FIG. 1 (annotated)



SAMSUNG-1008, Figure 1.8(b) (modified); SAMSUNG-1016, Figure 1.8(b) (modified); SAMSUNG-1007, FIG. 1

Since Lambert’s method “automatically, transparently and *securely* controls software execution,” and can be stored in a registry, a POSITA would have understood or found obvious that LRSD’s policy information (e.g., Lambert’s software restriction policy) is securely stored. SAMSUNG-1007, Abstract, [0011]; SAMSUNG-1003, ¶213 (citing SAMSUNG-1017, [0024], *generally*).

[1.2.ii]

LRSD renders [1.2.ii] obvious. SAMSUNG-1003, ¶¶214-216.

In LRSD, per Lambert, policy information (e.g., a software restriction policy comprising a set of rules that “may restrict or prevent the software from running”) is indicative of execution environment policies. SAMSUNG-1007, [0011]; SAMSUNG-1003, ¶215. For example, Lambert explains that “executable code is identified and classified, and then a rule corresponding to the classification (or if none correspond, a default rule) automatically and transparently determines the software’s ability to execute.” SAMSUNG-1007, [0008]. Lambert’s “rule specifies for the classification that is determined for any given software module (e.g., software file) [that] the software may execute in a *normal (unrestricted) execution environment* (context), in a *restricted execution environment*, or the software may be prevented from running at all.” *Id.* Since such rules “may be maintained in a restriction policy” and determine whether the software executes in an unrestricted or a restricted execution environment, a POSITA would have understood that the software restriction policy (comprising rules) is an execution environment policy. *Id.*; SAMSUNG-1003, ¶215.

Further, Lambert teaches that execution environment policies are application-specific. SAMSUNG-1003, ¶216. Lambert explains that “an administrator can set a rule for any file based on its unique hash value with the rule specifying

whether (and if so, how) a software file having that hash value will execute,” because “an administrator may not want users to run a particular version of a program, such as if that version is known to have security or privacy bugs, or compromises system stability.” SAMSUNG-1007, [0056]. Further, in Lambert, “[v]ia a user interface, the administrator selects that *application version*, obtains its hash value, sets up a hash rule for it that specifies a security level.” SAMSUNG-1007, [0056]. In one example, “an administrator can select any software file, obtain a hash value from its contents, and enter and associate a rule (that specifies a security level) with that hash value,” e.g., “the administrator may set the rule to specify that any file with that hash will not be allowed to execute at all.” SAMSUNG-1007, [0055]-[0056]. Since, per Lambert, LRSD’s rules apply to specific applications (and more specifically, to an application version), a POSITA would have understood of found obvious that LRSD’s execution environment policies are application-specific. SAMSUNG-1007, [0056]; SAMSUNG-1003, ¶216.

[1.2.iii]

LRSD renders [1.2.iii] obvious. SAMSUNG-1003, ¶¶217-222.

Per Lambert, in LRSD, policy information (e.g., the software restriction policy including classification rules) is indicative of configurable communication security policies. SAMSUNG-1007, [0008], [0011]; SAMSUNG-1003, ¶218.

For example, as described above, in LRSD, per Lambert, policy information

(e.g., the software restriction policy), determines an application's execution environment. §III.C.2[1.2.ii]; SAMSUNG-1007, [0008], claim 54; SAMSUNG-1003, ¶219. Further, per Lambert, "a restricted execution environment comprises restricting network access for the software." SAMSUNG-1007, claim 26. Lambert provides examples of such restrictions, e.g., "a particular program or class thereof can be prevented from obtaining network access or otherwise restricted to some reduced level of network access." SAMSUNG-1007, [0077]; SAMSUNG-1003, ¶219.

Thus, per Lambert, since the policy information (e.g., the software restriction policy including classification rules) determines an execution environment, and a restricted execution environment restricts network access, a POSITA would have understood that Lambert's software restriction policy (resulting in network access restrictions) is a communication security policy. SAMSUNG-1007, [0008], [0011], [0077], claim 26; §III.C.2[1.2.ii]; SAMSUNG-1003, ¶220.

Lambert further explains that software restriction policies are configurable. SAMSUNG-1003, ¶221. For example, Lambert states that a "policy generally comprises a general (default) rule and exceptions to the default rule set by an administrator," and "an administrator can efficiently constrain (or prevent from running) everything on a machine, except for certain identified software files or software classifications that are allowed to run at some other security level, also set by

the administrator.” SAMSUNG-1007, [0011]. Lambert further explains that “[a]lternatively, via the default rule an administrator can allow everything to run at one security level, such as with the user's normal rights, except for certain specified software or software classifications.” SAMSUNG-1007, [0011]. Additionally, Lambert explains that an “administrator identifies/classifies the software and sets the rules, including a default rule that applies to any software file not classified under a specific rule.” SAMSUNG-1007, [0052]. Since policies are set by an administrator, a POSITA would have understood that the policies are configurable. SAMSUNG-1003, ¶221.

Since Lambert further teaches that such policies are securely stored (as described in §III.C.2[1.2.i], LRSD discloses or renders obvious storing secure policy information indicative of a configurable communication security policy.

§III.C.2[1.2.i]; SAMSUNG-1003, ¶222.

[1.3.i]

LRSD renders [1.3.i] obvious. SAMSUNG-1003 ¶¶223-229.

Per Lambert, in LRSD, secure stored information is indicative of an application-specific execution environment policy. §§III.C.2[1.2.i]-[1.2.ii].

Further, per Lambert, based on the secure stored information indicative of an application-specific execution environment policy, LRSD allows a first set of mul-

multiple device application instances to run only in a first secure application environment. SAMSUNG-1003, ¶225. For example, Lambert teaches that policies are used to classify software into security levels, explaining that a “software restriction policy comprising a set of rules is provided to correlate the classification information for any software to a security level that may restrict or prevent the software from running.” SAMSUNG-1007, [0011]; SAMSUNG-1003, ¶225. Lambert further explains that “[d]epending on what the rule specifies for the classification that is determined for any given software module (e.g., software file), the software may execute in a normal (unrestricted) execution environment (context), in a restricted execution environment, or the software may be prevented from running at all.” SAMSUNG-1007, [0008]. Lambert similarly describes “controlling execution of any executable content of the software based on the security level associated with the software,” where controlling execution can include “providing a restricted execution environment for the software.” SAMSUNG-1007, claims 1, 21. Lambert also states that the “invention provides a transparent way *to run untrusted (and hence potentially harmful) code in a constrained environment* in which such code is not allowed to perform certain operations, such as operations that deal with system data or user data.” SAMSUNG-1007, [0009]. Thus, per Lambert, in LRSD, the software restriction policy comprising a set of rules determines that application instances (first set of device application instances, e.g., untrusted code)

are assigned to the restricted execution environment. SAMSUNG-1003, ¶225.

Additionally, as described in §III.C.1(c), each execution environment (which is part of an LRD software instance) runs in its own VM. §III.C.1(c); SAMSUNG-1003, ¶227. Thus, since the first set of device applications is assigned to the restricted execution environment, and the restricted execution environment is run in a VM (which, per Smith, is isolated from other VMs), LRSD discloses or renders obvious running a first set of multiple device application instances to run *only* in a first secure application environment, e.g., the VM hosting the restricted execution environment. SAMSUNG-1008, 369, 370-371 (“system VM provides a sandbox *that isolates one system environment from other environments*”); SAMSUNG-1016, 505, 507; SAMSUNG-1003, ¶227.

Further, per Lambert, LRSD renders obvious allowing a first set of *multiple* device application instances to run only in a first secure application environment on the wireless end-user device. SAMSUNG-1003, ¶228. For example, Lambert explains that, “[d]epending on what the rule specifies for the classification that is determined for *any given software module* (e.g., software file), the software may execute in a normal (unrestricted) execution environment (context), in a restricted execution environment, or the software may be prevented from running at all.” SAMSUNG-1007, [0008]. Similarly, Lambert states that “various types of soft-

ware and specific *software files* (or other suitable data structures) may be identified and classified, with rules set and maintained for the software classifications that automatically and transparently determine the ability of software to execute.” SAMSUNG-1007, [0052]. Since Lambert describes “any given software module” and “software files”—plural, a POSITA would have understood or found obvious that Lambert’s teachings apply to multiple device applications. SAMSUNG-1003, ¶228.

Thus, per Lambert, software restriction policies classify software into security levels and security levels determine the execution environment and the associated VM, LRSD renders obvious, allowing a first set of multiple device application instances (e.g., untrusted code) to run only in a first secure application environment (e.g., the restricted execution environment) based on the secure stored information indicative of an application-specific execution environment policy (e.g., a software restriction policy). SAMSUNG-1007, [0008]-[0009], [0011], [0013], [0056]; §III.C.1(c); SAMSUNG-1003, ¶229.

[1.3.ii]

LRSD renders [1.3.ii] obvious. SAMSUNG-1003, ¶¶230-232.

Lambert explains that a “software restriction policy comprising a set of rules is provided to correlate the classification information for any software to a security

level” and “[d]epending on what the rule specifies for the classification that is determined for any given software module (e.g., software file), the software may execute in a normal (unrestricted) execution environment (context), in a restricted execution environment, or the software may be prevented from running at all,” SAMSUNG-1007, [0008], [0011]. Thus, per Lambert, a POSITA would have understood or found obvious that at least some of LRSD’s device application instances are assigned to run in the “normal (unrestricted) execution environment.” SAMSUNG-1007, [0011]; SAMSUNG-1003, ¶231. For example, Lambert explains that software content known to be trusted or as having come from a trusted source can be differentiated from other software sharing the same general path, whereby in keeping with the present invention *such trusted content can be run in a different execution environment/context from the other software.*” SAMSUNG-1007, [0054]; SAMSUNG-1003, ¶231.

As describe above, in LRSD, each execution environment is assigned to a separate VM. §III.C.1(c); SAMSUNG-1003, ¶232. Therefore, a POSITA would have understood or found obvious that the applications assigned to the normal execution environment run in a separate VM from the applications assigned to the restricted execution environment. §III.C.1(c); SAMSUNG-1003, ¶232. Further, since Smith teaches that “[s]oftware running on one guest system is isolated from software running on other guest systems,” a POSITA would have understood or

found obvious that application assigned to the VM hosting the normal execution environment are isolated from, and therefore run outside of, the restricted (first) application environment. SAMSUNG-1008, 18; SAMSUNG-1016, 72; SAMSUNG-1003, ¶232.

[1.4.i]

LRSD renders [1.4.i] obvious. SAMSUNG-1003, ¶¶233-237.

As described in §III.C.2[1.2.iii], LRSD includes policy information indicative of configurable communication security policies. SAMSUNG-1003, ¶234.

Per Rao, LRSD establishes a tunnel when packets are transmitted. SAMSUNG-1003, ¶235. For example, Rao states that the “first device has an application that generates a first and second packet,” and a filtering process “intercepts the first and second packets from the application and forwards the intercepted packets to the tunnel process.” SAMSUNG-1009, [0030], [0136]. Then, the “tunnel process requests the opening of a TCP connection between the first device and a second device” and “forwards the first and second packets as payloads in a first and second TCP packet to the second device. SAMSUNG-1009, [0030]. Thus, per Rao, LRSD, initiates a secure tunnel to a tunnel endpoint, e.g., the second device. SAMSUNG-1003, ¶235.

As described in §III.C.2[1.2.iii], per Lambert, LRSD’s policy information (e.g., the software restriction policy including classification rules) determines an

execution environment, and a restricted execution environment restricts network access (“a particular program or class thereof can be prevented from obtaining network access or otherwise restricted to some reduced level of network access”). SAMSUNG-1007, [0008], [0077], claim 26; §III.C.2[1.2.iii]; SAMSUNG-1003, ¶236. Therefore, since the classification rules in the policy determine the execution environment, a POSITA would have understood or found obvious that the secure stored information indicative of a configurable communication security policy determines (by assigning an execution environment, which can be restricted)—and is therefore based on—whether an application (in the assigned execution environment) is allowed to or prevented from accessing the network (according to the network restrictions assigned to the execution environment). SAMSUNG-1003, ¶236.

Since, LRSD initiate a first secure tunnel to a network tunnel endpoint when an application transmits packets (per Rao), and the software restriction policy (per Lambert) determines whether an application can transmit packets, a POSITA would have understood that LRSD renders obvious initiating a first secure tunnel to a network tunnel endpoint (when an application first transmits packets, per Rao) based on the secure stored information indicative of a configurable communication security policy (determining whether the application is permitted to transmit packets, per Lambert). SAMSUNG-1007, [0008], [0077], claim 26; §III.C.2[1.2.iii]; SAMSUNG-1009, [0030], [0136], [0207]; SAMSUNG-1003, ¶237.

[1.4.ii]

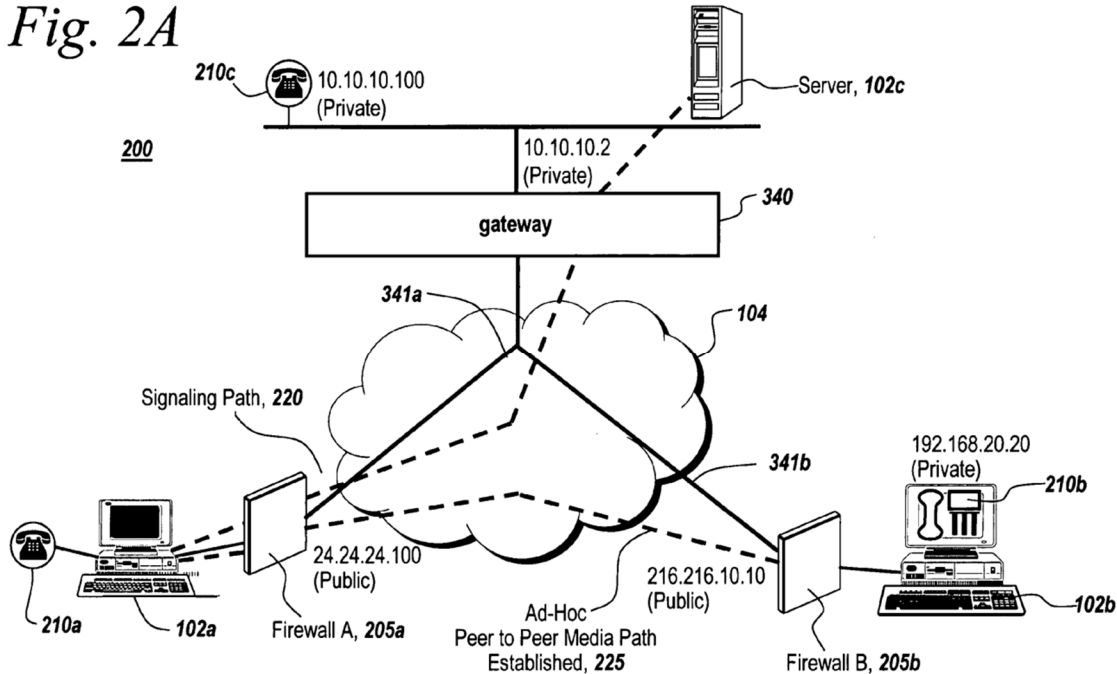
LRSD renders [1.4.ii] obvious. SAMSUNG-1003, ¶¶238-242.

As described in §III.C.2[1.2.ii], a first set of device applications is allowed to execute only in the secure execution environment (in an LRD software instance) assigned according to the software restriction policy. §III.C.2[1.2.ii]; SAMSUNG-1007, [0008], claim 54; SAMSUNG-1003, ¶239.

Per Rao, LRSD transmits packets from the first application through a tunnel. SAMSUNG-1003, ¶240. For example, Rao states that the “first device has an application that generates a first and second packet,” and a filtering process “intercepts the first and second packets from the application and forwards the intercepted packets to the tunnel process.” SAMSUNG-1009, [0030]. Then, the tunnel process forwards the first and second packets as payloads in a first and second TCP packet to the second device. *Id.*

Since Rao states that “[e]ach of the clients 105a-105n communicate over the network 104 to the gateway 340 via a tunneling or gateway connections 341a-341n using any type and/or form of suitable tunneling or gateway protocols,” in cases where “any type and/or form of suitable tunneling” protocol is used (e.g., tunneling is used instead of a gateway protocol), Rao provides no alternative to using the tunnel, as illustrated in FIG. 2A. SAMSUNG-1009, [0087], FIG. 2A; SAMSUNG-1003, ¶241.

Fig. 2A



SAMSUNG-1009, FIG. 2A

Therefore, a POSITA would have understood or found obvious that all network data communications to and from device application instances in the execution environment are directed through the tunnel (since no alternate pathway exists for applications in the LRD software instance). SAMSUNG-1003, ¶242.

[2.1]

LRSD renders [2.1] obvious. SAMSUNG-1003, ¶¶243-250.

As described in §§III.C.2[1.3.i]-[1.3.ii], in LRSD, per Lambert, application instances in the second set of device application instances (e.g., trusted applications), are assigned to a separate execution environment (e.g., the normal execution

environment in the corresponding LRD software instance) from untrusted application (e.g., which are assigned to the restricted execution environment).

§§III.C.2[1.3.i]-[1.3.ii]; SAMSUNG-1003, ¶244. As further explained above, in LRSD, each execution environment is assigned to a separate VM, which hosts an LRD software instance. §III.C.1(c); SAMSUNG-1003, ¶244.

Per Smith, VMs provide isolation from other VMs. SAMSUNG-1003, ¶245. For example, Smith explains that “a virtual machine can be used to provide an environment that, for all practical purposes, is isolated from the others and through which it is not possible for one user to observe or change another’s data and activity”; “VM provides a sandbox that isolates one system environment from other environments”; and “[s]oftware running on one guest system is isolated from software running on other guest systems.” SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507. Thus, a POSITA would have understood or found obvious that application instances in one VM are isolated from resources present in other VMs. *Id.*; SAMSUNG-1003, ¶245.

Therefore, in the combination, application instances in the second set of device application instances (running in the VM hosting the LRD software instance with the normal execution environment) are not allowed to use the first secure tunnel (running in the VM hosting the LRD software instance with the restricted exe-

cution environment), at least because the second set of device applications are isolated from resources in the VM hosting the restricted execution environment, including the first secure tunnel. SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, ¶246.

Further, as described in §III.C.2[1.2.iii], LRSD’s software restriction policy, including classification rules, is indicative of a configurable communication security policy. §III.C.2[1.2.iii]; SAMSUNG-1003, ¶247. For example, Lambert explains that “various types of software and specific software files (or other suitable data structures) may be identified and classified, with rules set and maintained for the software classifications that automatically and transparently determine the ability of software to execute,” and “a selected rule[] is used to determine whether the software can be loaded, and if so, to what extent any processes of the software will be restricted.” SAMSUNG-1007, [0052].

Additionally, as described in §III.C.2[1.3.i], per Lambert, the software restriction policy determines that a first set of multiple device application instances is allowed to run only the restricted execution environment, which is implemented as a VM (per Smith). §III.C.2[1.3.i]; §III.B.1(a); SAMSUNG-1003, ¶248. A POSITA would have understood or found obvious that assigning some application instances to a particular VM (hosting the restricted application environment) and

assigning other application instances to a different VM (hosting the normal application environment) defines a first configuration. SAMSUNG-1003, ¶248. Therefore, a POSITA would have understood or found obvious that, per Lambert, the configurable communication security policy defines a first configuration. SAMSUNG-1003, ¶248. As described above, in this first configuration, application instances in the second set of device application instances (assigned to the VM hosting the normal execution environment) are not allowed to use the first secure tunnel (which is in the VM hosting the restricted execution environment). SAMSUNG-1003, ¶248.

Therefore, LRSD teaches or renders obvious that the configurable communication security policy (software restriction policy comprising a set of rules) defines a first configuration (assignment of application instances to application environments), which indicates that application instances in the second set of device application instances (in a second VM) are not allowed to use the first secure tunnel (in a first VM). SAMSUNG-1007, [0011]; SAMSUNG-1003, ¶¶249-250.

[2.2]

LRSD renders [2.2] obvious. SAMSUNG-1003, ¶¶251-253.

As described in §III.C.2[2.1], in the first configuration (based on the stored information indicative of a configurable communication security policy, e.g., LRSD's software restriction policy), device application instances executing outside

of the first secure application environment are executed by a VM other than the VM hosting the restricted execution environment. §§III.C.2[2.1], III.C.1(c); SAMSUNG-1003, ¶252.

Since such other VMs are isolated from the VM hosting the restricted execution environment, all network data communications to and from device application instances executing outside of the first secure application environment is directed through a network path outside of the first secure tunnel. SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, ¶253. As Dr. McDaniel explains, per Smith, applications in a VM other than the VM hosting the restricted execution environment are unable to access any resource in the VM hosting the restricted execution environment, so all network data communications with device application instances executing outside of the first secure application environment is necessarily directed through a network path outside of the first secure tunnel. SAMSUNG-1008, 18, 370-371; SAMSUNG-1016, 72, 507; SAMSUNG-1003, ¶253.

IV. INSTITUTION SHOULD NOT BE DENIED ON DISCRETION

Petitioner believes that discretionary denial is unwarranted, and yet, Petitioner intends to utilize the bifurcated briefing process contemplated by the March 26, 2025, Stewart Memorandum to rebut contentions if offered by Patent Owner to the contrary. *See* SAMSUNG-1010; SAMSUNG-1018; Video at

<https://uspto.cosocloud.com/p9xpz6s4jn75/>; SAMSUNG-1019.

V. CONCLUSION AND FEES

The Challenged Claims are unpatentable. Petitioner authorizes charge of 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. are the real parties-in-interest.

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

The '250 Patent is the subject of civil action *Headwater Research LLC v. Samsung Electronics Co., Ltd. et al*, 2-24-cv-00627, E.D. Tex., filed August 2, 2024 (SAMSUNG-1004). Petitioner is not aware of any other disclaimers, reexamination certificates, or IPR petitions addressing the '250 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: IPR39843-0191IP1@fr.com	Jeremy J. Monaldo, Reg. No. 58,680 Jennifer J. Huang, Reg. No. 64,297 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 IPR39843-0191IP1@fr.com

(referencing No. 39843-0191IP1 and cc'ing PTABInbound@fr.com).

Respectfully submitted,

Dated April 30, 2025

/Jennifer J. Huang/

W. Karl Renner, Reg. No. 41,265
Jeremy J. Monaldo, Reg. No. 58,680
Jennifer J. Huang, Reg. No. 64,297
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
T: 202-783-5070
F: 877-769-7945

(Control No IPR2025-00932)

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 13,932 words, which is less than the 14,000 allowed under 37 CFR § 42.24.

Dated April 30, 2025

/Jennifer J. Huang/
W. Karl Renner, Reg. No. 41,265
Jeremy J. Monaldo, Reg. No. 58,680
Jennifer J. Huang, Reg. No. 64,297
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 April 30, 2025, 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
bradley@fr.com