EXHIBIT B-7

Blackberry Push

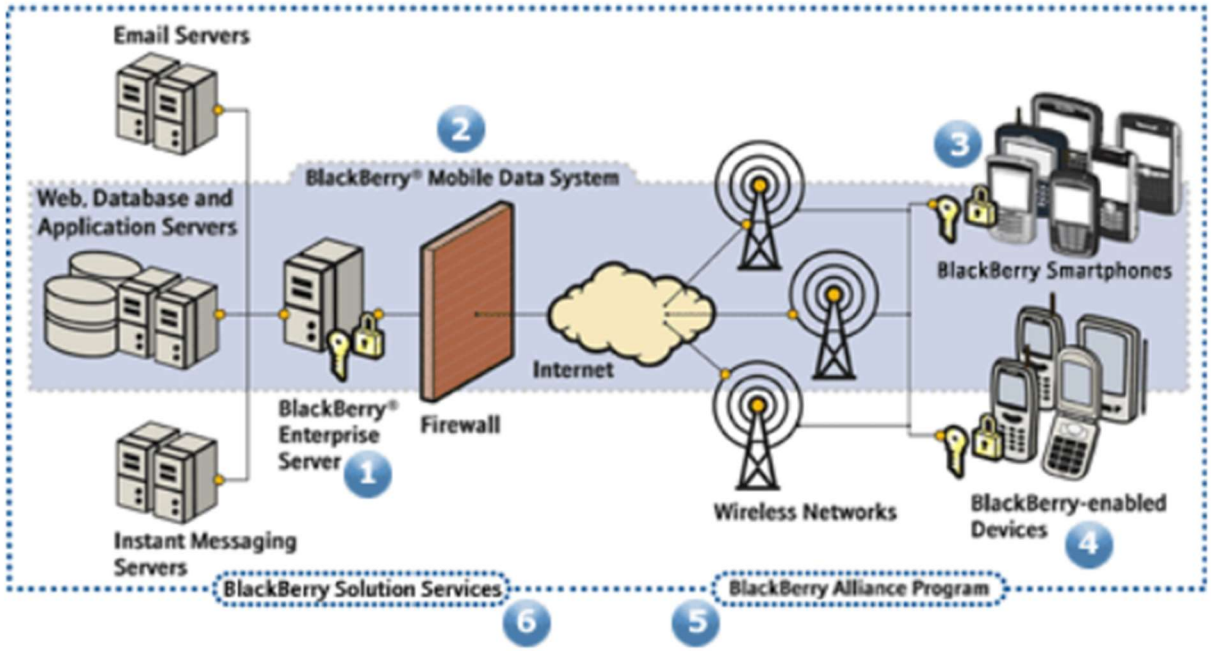
ANTICIPATES / RENDERS OBVIOUS U.S. PATENT NO. 10,321,320

As described in the following claim chart, claims 1-18 of U.S. Patent No. 10,321,320 (the "'320 patent") are invalid in view of Blackberry's Blackberry Enterprise System ("BES") v4.1.6 with its Mobile Data System ("MDS") v4.1, which constituted Blackberry's push notification system ("Blackberry Push"), alone or in combination with U.S. Patent No. 6,219,694 ("Lazaridis"); U.S. Patent Pub. No. 2005/0207379 ("Shen"); U.S. Patent No. 7,082,615 ("Ellison"); U.S. Patent Pub. No. 2009/0282256 ("Rakic"); U.S. Patent Pub. No. 2008/0162637 ("Adamczyk"); U.S. Patent Pub. No. U.S. 2009/0158397 ("Herzog"); WO 2006/077283 ("Houghton"); U.S. Patent Application No. 2009/0240807 ("Munson"); 3GPP TS 23.140 v6.9.0 (2005-03), 3rd Generation Partnership Project, Technical Specification Group Terminals, Multimedia Messaging Service ("MMS"), Functional Description, Stage 2 ("TS-23.140"); U.S. Patent Pub. No. 2008/0215883 ("Fok"); and/or U.S. Patent Pub. No. 2008/0243999 ("Pazhyannur"). On information and belief, Blackberry Push's notification system was available to the public on July 18, 2008. Blackberry Push is therefore prior art under at least 35 U.S.C. § 102(a) and § 102(b). It anticipates or renders obvious the asserted claims alone, in combination with the knowledge of a person of ordinary skill in the art, or in combination with Lazaridis, Shen, Ellison, Rakic, Adamczyk, Herzog, Houghton, Munson, TS.23-140, Fok, and/or Pazhyannur. Blackberry Push is described in at least the following:

- BlackBerry Mobile Data Service Technical Series ["MDS Technical Series"].
- BlackBerry MDS Studio v4.1 Feature and Technical Overview ["MDS Studio 4.1"].
- BlackBerry Mobile Data System v4.1 Technical Overview ["MDS 4.1 Overview"].
- Blackberry Enterprise Solution v4.1, Security Technical Overview ["BES 4.1 Technical Overview"]
- BlackBerry Enterprise Solution Overview (Wayback) ["BES Overview"].
- BlackBerry Enterprise Solution Architecture Overview (Wayback) ["BES Architecture Overview"].
- BlackBerry Mobile Data System (Wayback) ["MDS Overview"].
- Blackberry Java Development Environment Version 4.0 (Wayback) ["Java Development 4.0"]
- Blackberry MDS Applications (Wayback) ["MDS Applications"]
- Jan-Piet Mens, Real-World BlackBerry MDS Push Applications Framework (2006) ["Mens (2006)"].
- CIO: RIM BlackBerry Enterprise Server (BES) 4.1.6 Upgrade. July 18, 2008 ["BES 4.1.6"]

To the extent that these Invalidity Contentions rely on or otherwise embody particular constructions of terms or phrases in the Asserted Claims, Defendants are not proposing any such constructions as proper constructions of those terms or phrases. Various positions put forth in this document are predicated on Plaintiff's incorrect and overly broad interpretation of its claims as evidenced by its infringement contentions, dated October 17, 2025. Those positions are not intended to and do not necessarily reflect Defendants' interpretation of the true and proper scope of Plaintiff's claims, and Defendants reserve the right to adopt claim construction positions that differ from or even conflict with various positions put forth in this document.

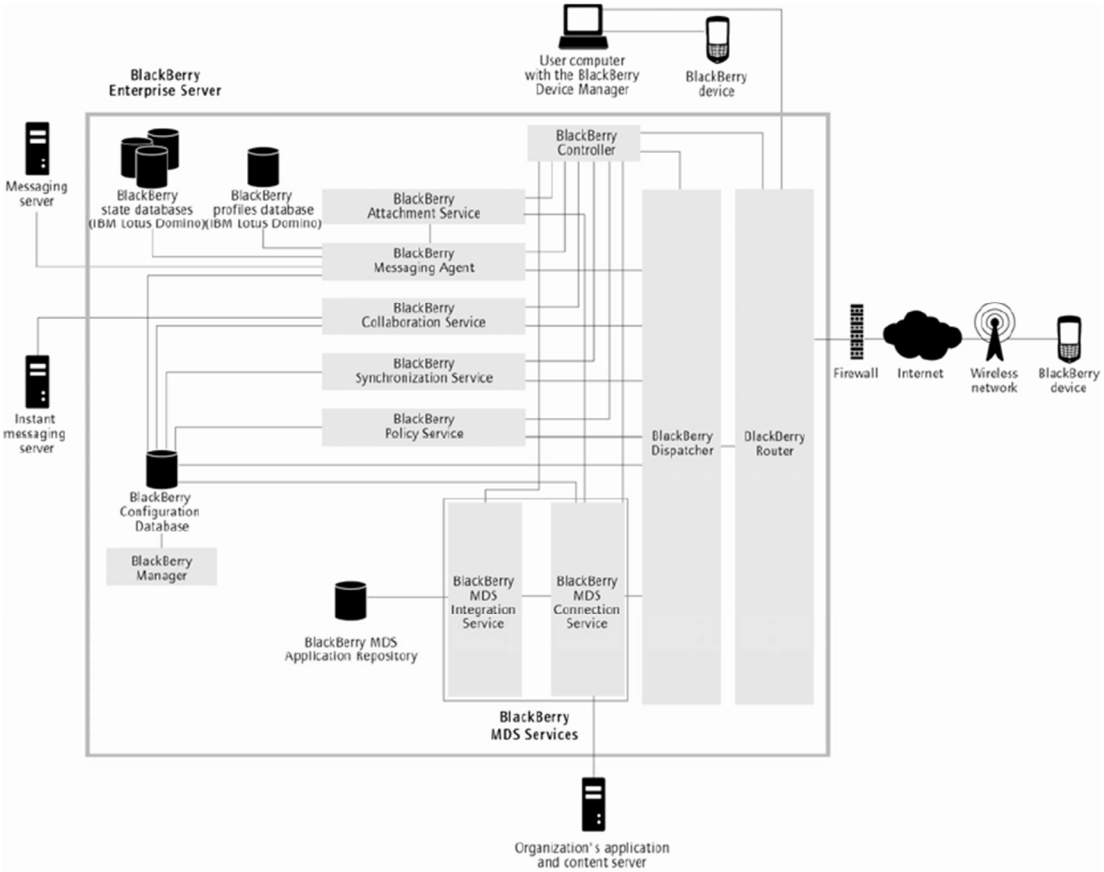
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 1 of 422

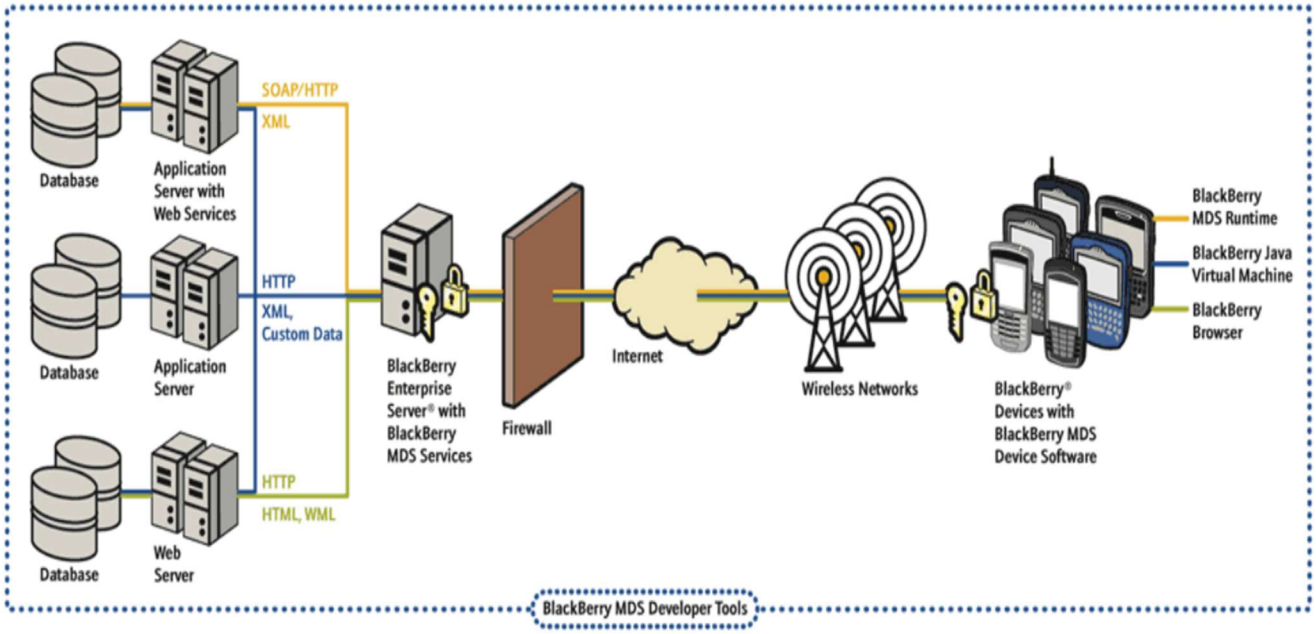
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
1[pre]: A networked system comprising:	Blackberry Push, as evidenced by the example citations below, discloses a networked system. Blackberry Push discloses a networked system comprising server-side message and application gateway components and device-side agents operating over wireless networks. *See, e.g.*, • BES Architecture Overview, p. 1: 

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 2 of 422

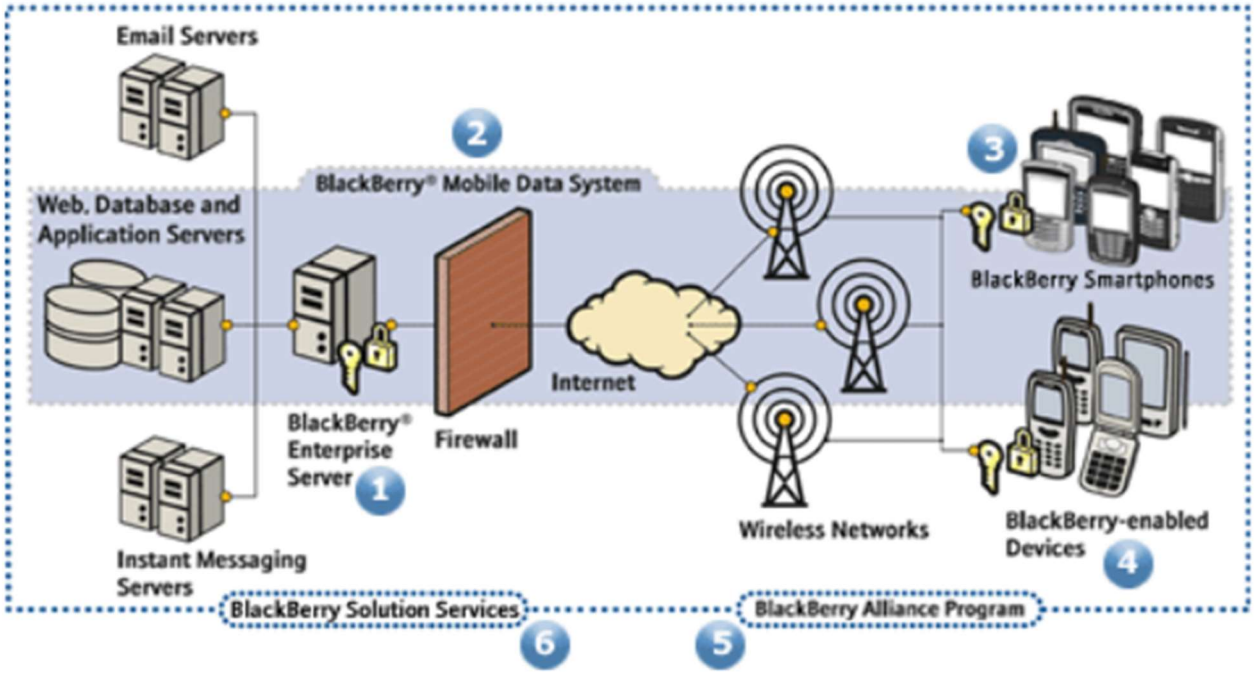
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	The key elements of the BlackBerry Enterprise Solution architecture include: 1. BlackBerry® Enterprise Server Robust software that acts as the centralized link between wireless devices, wireless networks and enterprise applications. The server integrates with enterprise messaging and collaboration systems to provide mobile users with access to email, enterprise instant messaging and personal information management tools. All data between applications and BlackBerry® smartphones flows centrally through the server. … 2. BlackBerry® Mobile Data System (BlackBerry MDS) An optimized framework for creating, deploying and managing applications for the BlackBerry Enterprise Solution. It provides essential components that enable applications beyond email to be deployed to mobile users, including developer tools, administrative services and BlackBerry® Device Software. It also uses the same proven BlackBerry push delivery model and advanced security features used for BlackBerry email. … 3. BlackBerry Smartphones Integrated wireless voice and data devices that are optimized to work with the BlackBerry Enterprise Solution. They provide push-based access to email and data from enterprise applications and systems in addition to web, MMS, SMS and organizer applications. • MDS Technical Series, p. 4: What is Mobile Data Series (MDS)? • A proxy server which is able to make TCP/IP connections on behalf of the BlackBerry handheld • Enables data sources, in addition to email, to be extended wirelessly with security • Open solution – transparent gateway – Provides the plumbing • Supports pull and push based content • Utilizes standard BlackBerry Enterprise Server™ services

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 3 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• BES 4.1 Technical Overview, p. 33:  BlackBerry Enterprise Server architecture

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS Overview, p. 1:  Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 5 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
1[a]: a network server system including	Blackberry Push, as evidenced by the example citations below, discloses a network server system. Blackberry Push discloses a networked system comprising server-side message and application gateway components and device-side agents operating over wireless networks. *See, e.g.*, • BES Architecture Overview, p. 1: 

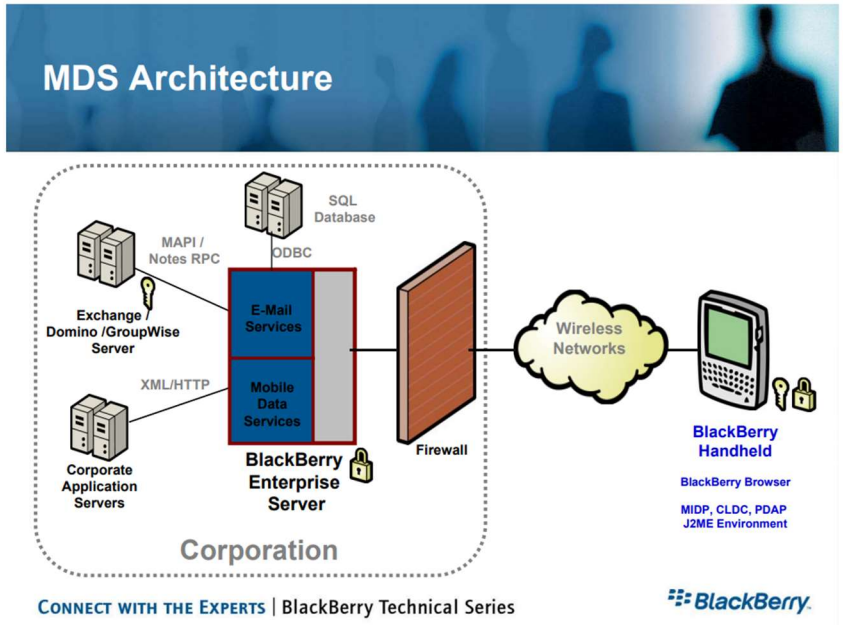
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 6 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	The key elements of the BlackBerry Enterprise Solution architecture include: 1. BlackBerry® Enterprise Server Robust software that acts as the centralized link between wireless devices, wireless networks and enterprise applications. The server integrates with enterprise messaging and collaboration systems to provide mobile users with access to email, enterprise instant messaging and personal information management tools. All data between applications and BlackBerry® smartphones flows centrally through the server. … 2. BlackBerry® Mobile Data System (BlackBerry MDS) An optimized framework for creating, deploying and managing applications for the BlackBerry Enterprise Solution. It provides essential components that enable applications beyond email to be deployed to mobile users, including developer tools, administrative services and BlackBerry® Device Software. It also uses the same proven BlackBerry push delivery model and advanced security features used for BlackBerry email. … 3. BlackBerry Smartphones Integrated wireless voice and data devices that are optimized to work with the BlackBerry Enterprise Solution. They provide push-based access to email and data from enterprise applications and systems in addition to web, MMS, SMS and organizer applications. • MDS Technical Series, p. 4: What is Mobile Data Series (MDS)? • A proxy server which is able to make TCP/IP connections on behalf of the BlackBerry handheld • Enables data sources, in addition to email, to be extended wirelessly with security • Open solution – transparent gateway – Provides the plumbing • Supports pull and push based content • Utilizes standard BlackBerry Enterprise Server™ services

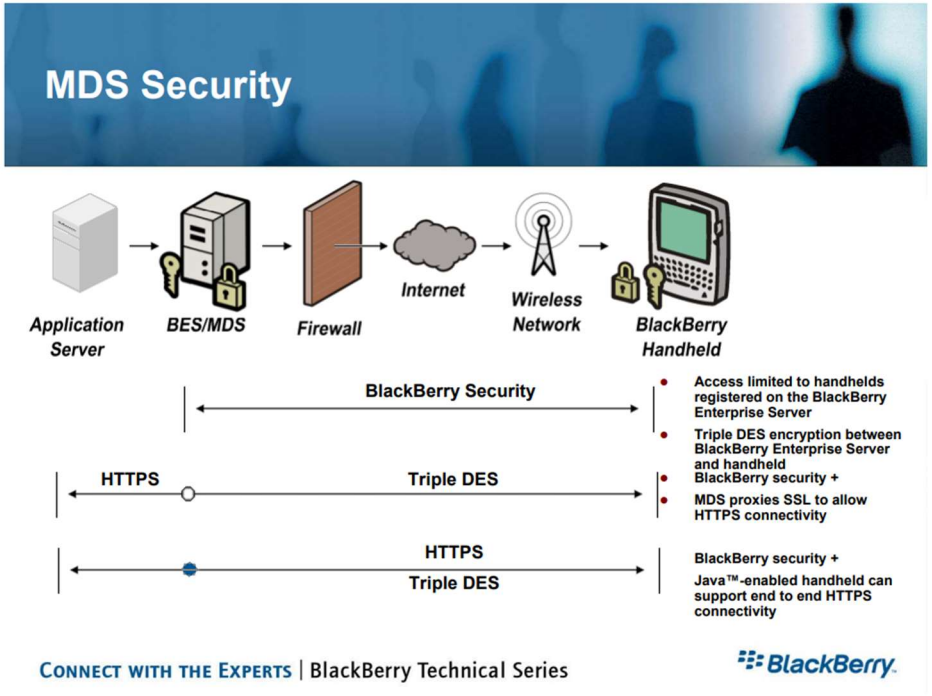
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 7 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.
1[b]: a link interface to maintain a respective secure Internet data message link between the link interface and a respective device link agent on each of a plurality of wireless end-user devices, each of the wireless end-user devices comprising multiple software components authorized to receive messages via the device link agent on that device;	Blackberry Push, as evidenced by the example citations below, discloses a link interface to maintain a respective secure Internet data message link between the link interface and a respective device link agent on each of a plurality of wireless end-user devices, each of the wireless end-user devices comprising multiple software components authorized to receive messages via the device link agent on that device. Blackberry Push implements secure, per-device links between the Blackberry Push server and device-side agents. The handheld communicates with Blackberry Push "using the existing secure 3DES (or AES) connection over port 3101," and uses the IP Proxy Protocol (IPPP) "service book" for device-MDS communications. *See, e.g.*, • MDS Technical Series, p. 8:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 8 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 • HTTP connection requests utilize the existing secure 3DES (or AES) connection over port 3101 • MDS communicates with web servers using protocol requested by handheld • MDS supports common protocols and mark up languages – HTTP, HTTPS, TCP, XML, HTML

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 9 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• Handheld and MDS communicate using IP Proxy Protocol – IPPP service book required on the handheld Blackberry Push provides end-to-end or proxied HTTPS. *See. e.g.*, • MDS Technical Series, p. 12: 

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 10 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	On each device, there is software receive push and pull data and dispatch to user agents and applications, including BlackBerry Browser and/or MDS Runtime, i.e. "device link agents." *See, e.g.*, • MDS 4.1 Overview, pp. 6-7: Push services can be used with any of the BlackBerry MDS application models. The BlackBerry Browser includes a listener for push connections. Application developers can also design BlackBerry MDS Java or BlackBerry MDS Studio applications that listen silently in the background for push requests. …

Application models and development environments

Application model	BlackBerry device software	Development environment
Browser model: BlackBerry MDS Browser applications	BlackBerry Browser	• To provide mobile access to intranet content and to create web-based wireless applications that BlackBerry users can access through the BlackBerry Browser, use existing web content development tools. Extend existing web-based applications wirelessly with little custom development. See "Making web content accessible wirelessly" on page 8 for more information. • To learn how to optimize new and existing web applications for the BlackBerry Browser using standard web development tools, use the *BlackBerry Wireless Handheld Browser Content Developer Guide*. • To create and test vector graphics and animations for BlackBerry devices, use the tools and simulators that the Plazmic Content Developer's Kit for BlackBerry (Plazmic CDK) provides.
Web service model: BlackBerry MDS Studio applications	BlackBerry MDS Runtime	To create lightweight applications that provide mobile access to enterprise web services, use the BlackBerry MDS Studio. The BlackBerry MDS Studio is a powerful visual application design and assembly tool that enables application developers to quickly create rich-client applications that integrate with enterprise web services using a component-based drag and drop approach. Application developers can create robust functionality without Java programming using the BlackBerry MDS Studio. See "Creating BlackBerry MDS Studio Applications" on page 11 for more information.
Traditional client/server model: BlackBerry MDS Java applications	BlackBerry JVM	To create custom Java applications for the BlackBerry device that provide a sophisticated user interface and navigation, robust data management, and flexible support for custom data formats, use the BlackBerry JDE. The BlackBerry JDE provides development and simulation tools that enable application developers to build rich-client Java Platform, Micro Edition (Java ME) applications, which integrate with standard application servers, for BlackBerry devices. See "Creating BlackBerry MDS Java applications" on page 15 for more information.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 11 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS Technical Series, p. 39: The BlackBerry Browser • Standard "web page-based" user interface and navigation • Able to display data formatted in standard mark-up – HTML, cHTML, xHTML and WML • BlackBerry Browser supports PUSH of content via MDS Push Applications: Push Types • BlackBerry-defined push types: – "Channel Push" • Push an icon to the Home screen along with a web page • The icon acts as a permanent 'channel' to the referenced URL – "Message Push" • Push a message to the user's inbox that automatically launches the browser to a specified page (cached or not) – "Cache Push" • Push a page to the browser's cache, but do not notify the user • MDS Studio 4.1, p. 8: BlackBerry MDS Runtime The BlackBerry MDS Runtime™ is the container-based execution environment for BlackBerry MDS Applications. The BlackBerry MDS Runtime contains a control center application that enables users to

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 12 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	search for and manage BlackBerry MDS Applications and view application information. The BlackBerry MDS Runtime is designed to provide the underlying services that are used by BlackBerry MDS Applications, including the user interface, data storage, and client-server communication services. Devices run multiple authorized software components (browser, PIM, third-party apps, MDS Runtime applications); Blackberry Push enforces access control and per-application authorization. *See, e.g.*, • MDS Technical Series, p. 23: Messaging model BlackBerry MDS Applications are designed to store and process data locally on the BlackBerry device, without the need for a continuously available network connection. … Similarly, inbound messages can be queued by the BlackBerry MDS Services and transmitted to the BlackBerry device only when the wireless connection becomes available. BlackBerry MDS Applications are designed to process inbound messages as asynchronous events. This asynchronous communication model is designed for applications that are inherently tolerant to sudden or unpredictable loss of connection. This feature, combined with the ability to store and process data locally, enables BlackBerry MDS Applications to be designed in such a way that application functionality and enterprise data are available, even when wireless connections are not available. • MDS Technical Series, p. 23: • Install BlackBerry Enterprise Server software – The default install will include MDS • IPPP service book will be pushed to handhelds OTA – Required for handheld to communicate with MDS • MDS is enabled by default

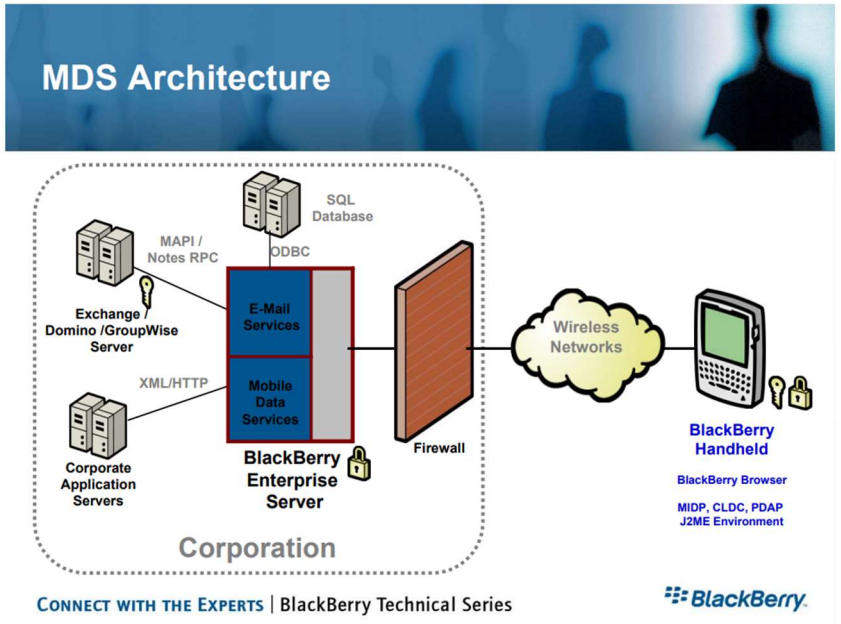
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 13 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	– BlackBerry Enterprise Server and users • If upgrading from a previous version of BlackBerry Enterprise Server the MDS settings do not change • Mens (2006), p. 2: What is MDS Push? The Mobile Data Service (MDS) is a software component of the BlackBerry Enterprise Server (BES) . I think of the MDS as a gateway between the wireless BlackBerry devices and the corporate network. The MDS can be used in the corporate environment to distribute (i.e. push) information and applications to devices utilizing the BlackBerry network infrastructure, and itMDS is used as a proxy between the BlackBerry browser(a.ka. MDS browser) on the devices and your corporate network and/or the Internet. • MDS Overview, p. 1: BlackBerry MDS Device Software BlackBerry MDS Device Software allows applications built with BlackBerry MDS Developer Tools to run on BlackBerry smartphones. • BES for MDS Application Overview, p. 1: 3. BlackBerry Smartphones Integrated wireless voice and data devices that are optimized to work with the BlackBerry Enterprise Solution. They provide push-based access to email and data from enterprise applications and systems in addition to web, MMS, SMS and organizer applications. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 14 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	knowledge of a person of ordinary skill in the art.
1[c]: a network interface to receive messages from a plurality of network elements, for delivery to respective ones of the software components identified in the messages, each network element authorized to send messages via the link interface to one or more of the software components on one or more of the wireless end-user devices; and	Blackberry Push, as evidenced by the example citations below, discloses a network interface to receive messages from a plurality of network elements, for delivery to respective ones of the software components identified in the messages, each network element authorized to send messages via the link interface to one or more of the software components on one or more of the wireless end-user devices. Blackberry Push discloses an interface to enterprise networks and web servers, acting as an HTTP/HTTPS/TCP proxy gateway allowing receipt of messages from external servers. *See, e.g.*, • MDS Technical Series, p. 8:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 15 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 • HTTP connection requests utilize the existing secure 3DES (or AES) connection over port 3101 • MDS communicates with web servers using protocol requested by handheld • MDS supports common protocols and mark up languages – HTTP, HTTPS, TCP, XML, HTML • Handheld and MDS communicate using IP Proxy Protocol – IPPP service book required on the handheld

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	The received messages include content and requests for delivery to user devices, including software component (application) addressing via headers/channel IDs. *See, e.g.*, • Mens (2006), p. 2: What is MDS Push? The Mobile Data Service (MDS) is a software component of the BlackBerry Enterprise Server (BES). I think of the MDS as a gateway between the wireless BlackBerry devices and the corporate network. The MDS can be used in the corporate environment to distribute (i.e. push) information and applications to devices utilizing the BlackBerry network infrastructure, and itMDS is used as a proxy between the BlackBerry browser(a.k.a. MDS browser) on the devices and your corporate network and/or the Internet. Push applications can be created using any language that can initiate a network connection to perform an HTTP request. A push message is made up of a request with a set of headers and a body (the page to be sent to the device). In essence, there are three type of Browser push that can be sent to a BlackBerry device (see Pushing Web Content to BlackBerry Wireless Devices for an more verbose description of the different kinds of browser push supported by the MDS) • Mens (2006), p. 6: pushlog Each push message sent to the MDS is recorded in the pushlog table, identifying the userid to whom the page was pushed. In addition, the appname is recorded as well as the (redundant) push destination pushdest. Each message pushed out is given a unique identifier called pushed; this is recorded as well. As soon as the MDS has delivered the message to the wireless BlackBerry device (or hasn't, in case of a failure), this table is updated to record that fact. The attribute status indicates success or failure of the message push. • MDS Studio 4.1, p. 12:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 17 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Message Message components are designed to communicate application data across the wireless network between the client application on the BlackBerry device and the data source. Using the BlackBerry MDS Studio, bindings are created that link specific message components to specific data source operations. An instance of a message component can be either an •inbound message: transmits data from the data source to the application on the BlackBerry device •outbound message: transmits data from the application on the BlackBerry device to the data source • MDS Studio 4.1, p. 10: Using SOAP SOAP is a protocol for exchanging information in a decentralized, distributed environment. It is an XML-based protocol that consists of three parts: •an envelope that defines a framework for describing what is in a message and how to process it •a set of encoding rules for expressing instances of application-defined data types •a convention for representing remote procedure calls and responses In the BlackBerry MDS Studio, application messages are designed to bind to SOAP operations. • MDS 4.1 Overview, p. 6: Data push The architecture that is used for BlackBerry wireless email is available for custom wireless application data. For Browser and Java application models, server-side applications, written in a variety of programming languages, retrieve content from a corporate web server, an application server, or a database. The applications then send HTTP POST or Push Access Protocol (PAP) requests to the connection service to send the content proactively to specific BlackBerry device users. BlackBerry MDS Studio applications use

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 18 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	web services eventing (WS-Eventing) to provide similar functionality. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.
1[d]: a message buffer system including a memory and logic,	Blackberry Push, as evidenced by the example citations below, discloses a message buffer system including a memory and logic. Blackberry Push includes memory and logic to stage, manage, and persist messages and delivery state. *See, e.g.*, • MDS Studio 4.1, p. 13: Messaging model BlackBerry MDS Applications are designed to store and process data locally on the BlackBerry device, without the need for a continuously available network connection. The communication model used to send and receive application messages is fully asynchronous. This means that if an outbound message is generated at a time when the BlackBerry device is outside of a wireless coverage area (for example, as a result of user input), the message is placed into an outbound queue until the wireless connection becomes available. When the message is placed into this queue, the application continues functioning, regardless of whether the message has been sent. Messages are transmitted from the queue in the background when the network connection becomes available. Similarly, inbound messages can be queued by the BlackBerry MDS Services and transmitted to the BlackBerry device only when the wireless connection becomes available. BlackBerry MDS Applications are designed to process inbound messages as asynchronous events. This asynchronous communication model is designed for applications that are inherently tolerant to sudden or unpredictable loss of connection.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 19 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	This feature, combined with the ability to store and process data locally, enables BlackBerry MDS Applications to be designed in such a way that application functionality and enterprise data are available, even when wireless connections are not available. In BlackBerry MDS Applications, the communication model consists of the following items: •delivery mode for application messages •handling messages when a BlackBerry device is outside of a wireless coverage area •message queue sizes •categories of notification messages •size limitations for messages • MDS Studio 4.1, p. 10: Using SOAP SOAP is a protocol for exchanging information in a decentralized, distributed environment. It is an XML-based protocol that consists of three parts: •an envelope that defines a framework for describing what is in a message and how to process it •a set of encoding rules for expressing instances of application-defined data types •a convention for representing remote procedure calls and responses In the BlackBerry MDS Studio, application messages are designed to bind to SOAP operations. • MDS Studio 4.1, p. 9:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 20 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push

BlackBerry MDS Services

The BlackBerry MDS Services are designed to provide connectivity between BlackBerry MDS Applications on BlackBerry devices and enterprise applications.

Service	Description
Connection Service	The Connection Service enables a system administrator to connect the BlackBerry MDS Services to the Mobile Data Service feature of the BlackBerry Enterprise Server®. The Connection Service is designed to accept and respond to push requests from server-side push applications when the application server is behind the corporate firewall. This service is designed to provide a link to standard servers on the corporate intranet or Internet using standard Internet protocol, such as HTTP or TCP/IP, and encrypt content using the same encryption standard that is used to encrypt BlackBerry messages and other BlackBerry data.
Application Integration Service	The Application Integration Service is designed to support web services and other standard mechanisms for integrating wireless applications with existing enterprise applications and systems. This service is designed to manage the transmission of application data messages between BlackBerry MDS Applications and data sources.
Provisioning Service	The Provisioning Service is designed to control which BlackBerry MDS Applications users can download to their BlackBerry devices, support application discovery from a BlackBerry device, and manage wireless transmission and on BlackBerry devices.
Data Optimization Service	The Data Optimization Service is designed to convert existing server-side content and data to enable wireless transmission and use on BlackBerry devices.
Administrative and Management Service	The Administrative and Management Service is designed to centralize the application lifecycle management, including centralized push installation, upgrade, and removal of applications from BlackBerry devices.

- MDS Technical Series, p. 49:

Centralized Push
• MDS traffic has highest priority
• Delivery retry is handled by the relay
• MDS does not retry if push / packet expires
• By default MDS supports 1000 connections
• Push has more of a performance impact than pull

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 21 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Additionally, Blackberry Push supported WAP 2.0 – PAP, which utilized priorities, deliver-before, and deliver-after functionality. *See, e.g.*, • MDS Technical Series, p. 44: • BlackBerry enables you to pro-actively PUSH new web pages and links to your users' handhelds • Why would I want to push web pages to my users?? – Alert them of new information based on an event that occurred on the server – 'Silently' update frequently accessed pages within the browser's local cache, so the user can access the most recent information even when out of coverage – Push a permanent link to a frequently accessed site (such as an internal portal or entry point to an application) • WAP 2.0 – PAP – Acknowledgments – Reliability • Java Development 4.0, pp. 121-22: Applications can push content to handhelds using one of two methods: • Push Access Protocol (PAP), which is part of the WAP 2.0 specification • RIM push Both push service implementations support the following tasks: • sending a server-side push submission • specifying a reliability mode for the push submission • In transport-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives acknowledgements for all sent packets. • In application-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives confirmation from the application that initiated the push. • specifying a deliver-before time-stamp for the push submission

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 22 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• requesting a result notification of the push submission The PAP implementation supports the following additional tasks: • specifying a deliver-after timestamp for the push submission • cancelling a push request submission • querying the status of a push request submission PAP pushes are stored in a database, whereas RIM pushes are stored in RAM. RIM pushes may be lost if the server is rebooted. A POSITA would have found it obvious that Blackberry Push inherently or obviously teaches the use of memory because its specifications disclose buffering content for transmission and reception, which necessarily requires temporary storage. Thus, a POSITA would recognize that Blackberry Push's buffering/ temporary storage of the undelivered message directly imply the presence and use of memory in the system. Further, it would have been obvious to one of ordinary skill in the art based on Blackberry Push alone or in combination with (i) the general knowledge of one of ordinary skill in the art and/or (ii) the teachings of any of the charted references with respect to this limitation, including, without limitation, for the following references: Lazaridis, Houghton, Munson, TS.23-140. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 23 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses a message buffer system, including a memory (e.g., primary memory store) and logic (e.g., the redirector). *See, e.g.*, "A system and method for pushing information from a host system to a mobile data communication device upon sensing a triggering event is disclosed. A redirector program operating at the host system enables a user to continuously redirect certain user-selected data items from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. The redirector program operates in connection with event generating applications and repackaging systems at the host system to configure and detect a particular user-defined event, and then to repackage the user-selected data items in an electronic wrapper prior to pushing the data items to the mobile device. The mobile device and the host system share a common electronic address so that messages generated at either the host system or the mobile data communication device are configured using the common electronic address." Lazaridis, Abstract.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 24 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Software executing on the mobile data communication device then transmits commands via the local communications link to the host system to cause the host to begin transmitting the user's data items for storage in a memory bank of the mobile device." *Id.*, 1:50-54. "Other sub-systems for creating triggering events and repackaging the user's data items could also be present at the host system. The host system also includes a primary memory store where the user's data items are normally stored." *Id.*, 3:4-8. "Once an event has triggered redirection of the user data items, the host system then repackages these items in a manner that is transparent to the mobile data communication device, so that information on the mobile device appears similar to information on the user's host system. The preferred repackaging method includes wrapping the user data items in an E-mail envelope that corresponds to the address of the mobile data communication device, although, alternatively, other repackaging methods could be used with the present invention, such as special-purpose TCP/IP wrapping techniques, or other methods of wrapping the user selected data items. The repackaging preferably results in E-mail messages generated by the user from the mobile device to be transmitted from the host system, thus enabling the user to appear to have a single E-mail address, such that the recipients of messages sent from the mobile communications device do not know where the user was physically located when the message was first sent. The repackaging also permits both messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed and decompressed." *Id.*, 3:66-4:19. "Message C in FIG. 1 represents an external message from a sender that is not directly connected to LAN 14, such as the user's mobile data communication device 24, some other user's mobile device (not shown), or any user connected to the Internet 18. Message C also represents a command message from the user's mobile data communication device 24 to the host system 10. As described in more detail in FIG. 3, the host system 10 preferably includes, along with the typical hardware and software associated with a workstation or desktop computer, the redirector program 12, a TCP/IP subsystem 43, a primary message store 40, an E-mail subsystem 44, a screen saver subsystem 48, and a keyboard subsystem" *Id.*, 5:44-56.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 25 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "FIG. 1 shows an E-mail message A being communicated over LAN 14 from computer 26 to the user's desktop system 10 (also shown in FIG. 1 is an external message C, which could be an Email message from an Internet user, or could be a command message from the user's mobile device 24). Once the message A (or C) reaches the primary message store of the host system 10, it can be detected and acted upon by the redirection software 12. The redirection software 12 can use many methods of detecting new messages." *Id.*, 7:31-38. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 26 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*8:32-55. "In the case where message C is representative of an external message from a computer on the Internet 18 to the host system 10 and the host has been configured to redirect messages of type C, then in a similar manner to message A, message C would be repackaged with an outer envelope B and transmitted to the user's mobile device 24." *Id.*, 8:56-61. "FIG. 2 is an alternative system diagram showing the redirection of user data items from a network server 11 to the user's mobile data communication device 24, where the redirector software 12 is operating at the server 11. This configuration is particularly advantageous for use with message servers such as Microsoft's® Exchange Server, which is normally operated so that all user messages are kept in one central location or mailbox store on the server instead of in a store within each user's desktop PC. This configuration has the additional advantage of allowing a single system administrator to configure and keep track of all users having messages redirected. If the system includes encryption keys, these too can be kept at one place for management and update purposes." *Id.*, 9:20-33. "If the message C is from an Internet computer to the user's desktop system and the user has redirection capabilities, then the server 11 detects the message C, repackages it using electronic envelope B, and redirects the repackaged message (C in B) to the user's mobile device 24. If the message C is a command message from the user's mobile device 24, then the server 11 simply acts upon the command message." *Id.*, 10:2-8. "The desktop system 10 is connected to LAN 14, and can send and receive data, messages, signals, event triggers, etc., to and from other systems connected to the LAN 14 and to external networks 18, 22, such as

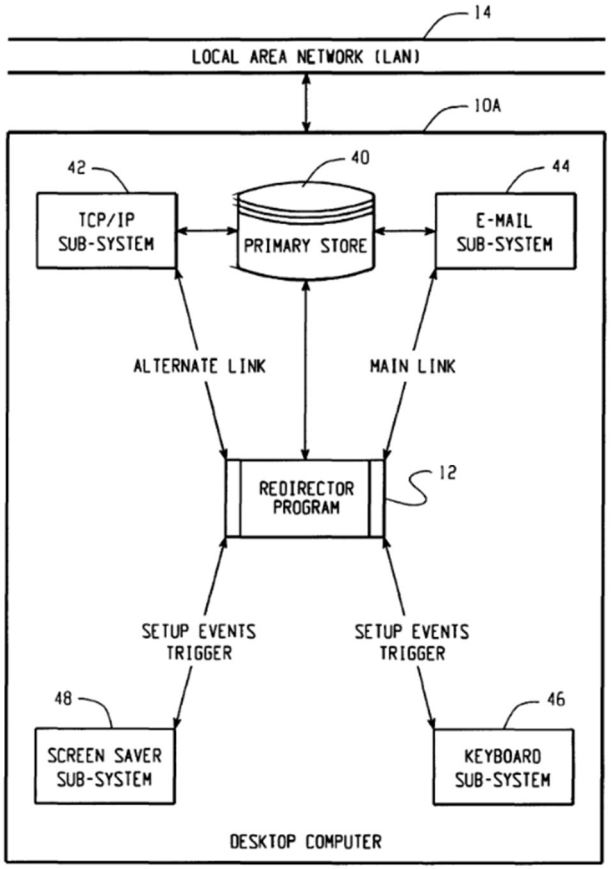
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 27 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the Internet or a wireless data network, which are also coupled to the LAN 14. In addition to the standard hardware, operating system, and application programs associated with a typical microcomputer or workstation, the desktop system 10 includes the redirector program 12, a TCP/IP sub-system 42, an E-mail sub-system 44, a primary data storage device 40, a screen saver sub-system 48, and a keyboard sub-system 46." *Id.*, 10:22-10:32. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. "When a message or signal is detected, the program 12 determines 58 whether it is one of the trigger events that has been configured by the user to signal redirection. If so, then at step 60 a trigger flag is set, indicating that subsequently received user data items (in the form of messages) that have been selected for redirection should be pushed to the user's mobile data communication device 24." *Id.*, 12:33-39. "If the trigger flag is set, then at step 70, the redirector 12 causes the repackaging system (E-Mail or TCP/IP) to add the outer envelope to the user data item, and at step 72 the repackaged data item is then redirected to the user's mobile data communication device 24 via LAN 14, WAN 18, wireless gateway 20 and wireless network 22." *Id.*, 12:50-56. "If, however, the redirection flag is on, then at step 110 the reply message is repackaged with the outer envelope having the addressing information of the user's desktop system 10, and the repackaged message is then transmitted to the desktop system 10 at step 106." *Id.*, 13:62-67.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 28 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"If, however, the mobile determines that the user is using the redirector software 12 at the desktop system 10, then control passes to step 110, where the outer envelope is added to the message. The repackaged original message is then transmitted to the desktop system 10 at step 106, which, as described previously, strips the outer envelope and routes the message to the correct destination. Following transmission of the message at step 106, control of the mobile returns to step 82 and waits for additional messages or signals." *Id.*, 14:14-23.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 29 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 3 *Id.*, Fig. 3.

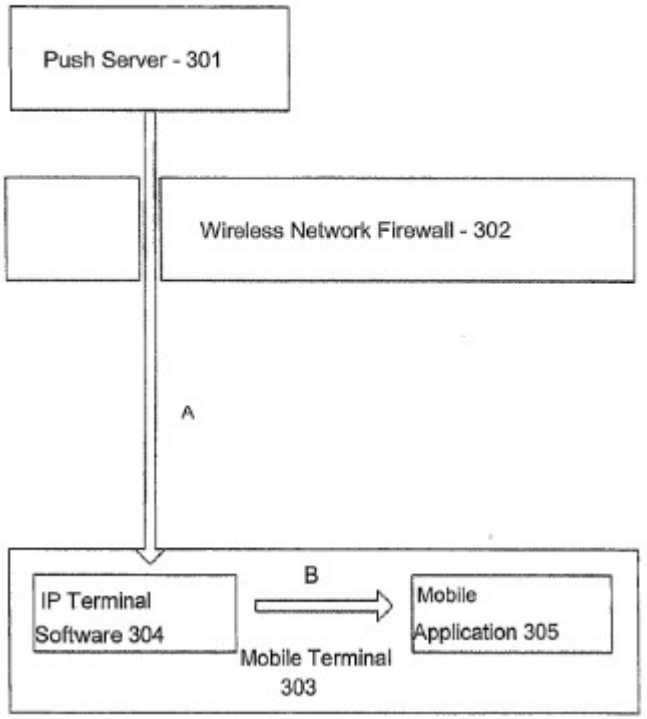
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 30 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Houghton Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, discloses a message buffer system including a

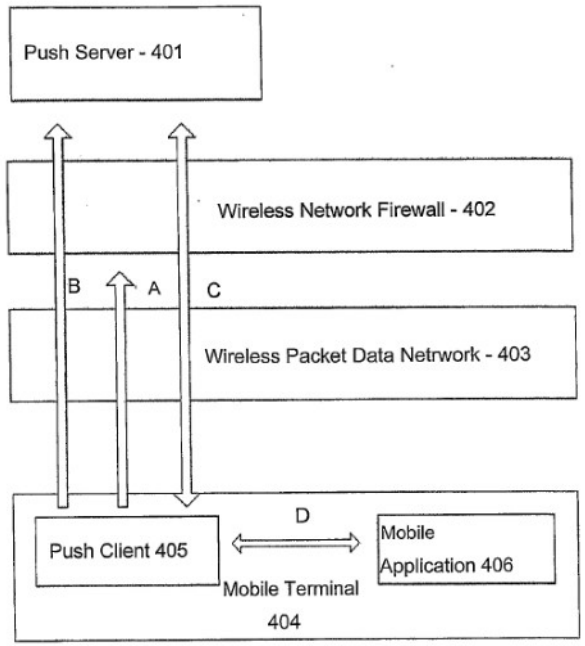
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 31 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	memory and logic. *See, e.g.*, "Push technology is characterized by the server proactively sending documents or information to a client or user. E-mail (Electronic mail) is actually a special kind of push technology that has been prevalent for a long time. The first and most common general category of mobile content push technologies is SMS (Short Message Service) based technologies. SMS and MMS (Multi-media Messaging Service) are the most common forms of 'push' messaging in the mobile industry. These store and forward messaging systems run by mobile telecommunications operators push all or part of the data to the mobile terminal over closed telecommunications signaling networks such as SS7 (Signaling System Seven)." Houghton, p. 3. "These server computers such as a telecommunications operator SIP Server or WAP Gateway is used to control and deliver messages and store messages when the mobile terminal is unavailable and deliver them on the senders behalf, possibly later then the terminal becomes available, or to forward them to a different gateway device, for example if the user is roaming to a different controlled mobile network ." *Id.*, p. 7 "When in proximity to a more capable network, if a connection can be established to this alternative broadband network access point, it is used preferentially and intelligence normally located in this broadband access point would route messages received on behalf of the wireless terminal to and from the server." *Id.*, pp. 13-14.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 32 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 3  *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 33 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 34 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Munson Munson and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Munson discloses a method of "pushing contents to client devices," including "group pushes" in which content is buffered and sent to multiple devices simultaneously, "serializ[ing]" content such that a series of messages are delivered to a particular device simultaneously. Munson, Abstract, ¶37, Fig. 4. A POSITA would have been motivated to combine Munson with Blackberry Push because a POSITA would have desired to support diverse client devices and network environments, as well as to leverage an existing, proven technology for wireless content delivery, e.g., Blackberry Push. Munson similarly teaches messages can be delivered from memory "according to a schedule of time or event" such as coordinating pushes "for a system maintenance purpose during off-peak hours." Munson, ¶¶40, 44. Thus, Munson provides additional details about how to implement a store-and-forward system by providing the computer memory at a server, and the computer programming to deliver messages from memory (including upon occurrence of certain events). *Id*., ¶¶36-38, 44, Fig. 4. Therefore, a POSITA would have been motivated to incorporate computer memory and programming into Blackberry Push to store and deliver messages from memory, consistent with Munson, to improve system flexibility and reliability. A POSITA would have had a reasonable expectation of success in combining Blackberry Push and Munson because Munson provides implementation details such push message system. *See, e.g.*, Munson, ¶¶36-38. A POSITA would have been motivated and had a reasonable expectation of success in incorporating conditions Munson teaches providing additional flexibility in how messages are delivered in PAP, used by Blackberry Push. This combination of Blackberry Push and Munson would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Munson, as the communication protocol because Blackberry Push was a well-known, industry-

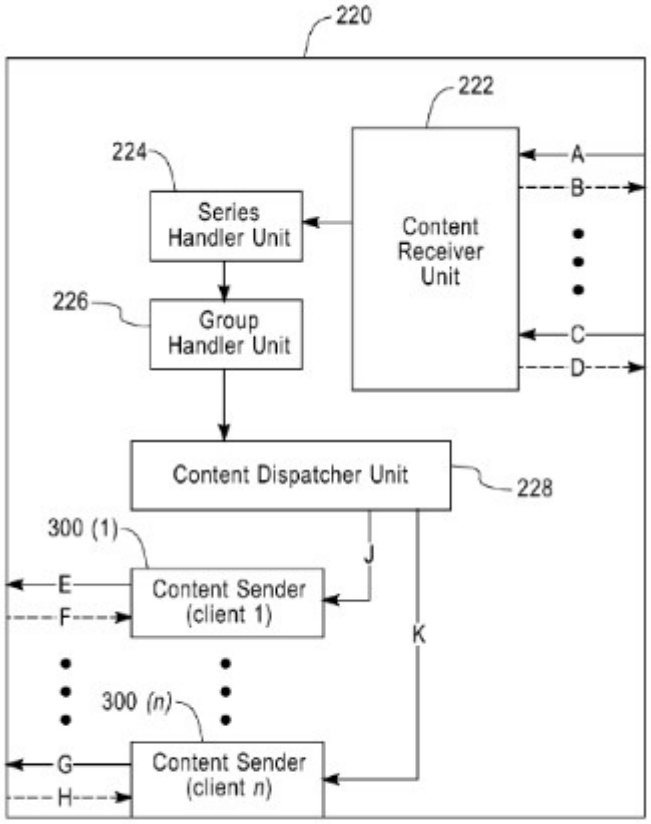
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 35 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	leading system. Munson, ¶¶35-36. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Munson push messaging system without requiring inventive skill. Munson, as evidenced by the example citations below, discloses a message buffer system (e.g., a series buffer), including a memory (e.g., Series Handler Unit) and logic. *See, e.g.*, "FIG. 3 illustrates an embodiment of the CPS 220. The CPS 220 includes a Content Receiver Unit 222, a Series Handler Unit 224, a Group Handler Unit 226, a Content Dispatcher Unit 228, and a plurality of Content Senders such as 300(1) and 300(*n*). The Content Receiver Unit 222 receives contents from application service providers 210, sends acknowledge receipts to application service providers 210, and forwards the contents to a Series Handler Unit 224. If the contents are received in a series, the Series Handler Unit 224 ensures the contents received as part of a series are pushed in the order specified by the application service providers. In one embodiment, the CPS 220 allows an application service provider to send contents to a group of client devices. Then, the Group Handler Unit 226 is responsible for looking up members of a group and performing an individual push on each member (i.e., pushing a content to each client device in the group). The Content Sender 300 performs actual transport of data of contents to a client device. There is one Content Sender 300 for each active client device. The Content Dispatcher Unit 228 maintains a list of the Content Senders 300. If a Content Sender is currently pushing a content to a client, the Content Dispatcher Unit 228 enqueues the content in the Content Sender. If a Content Sender is not pushing a content, the Content Dispatcher Unit 228 enqueues the content in the Content Sender and activates the Content Sender to start pushing." *Id*., ¶36. "FIG. 4 depicts a flow chart that the Series Handler Unit 224 utilizes. At step 225, the Series Handler Unit 224 receives contents from application service providers through the Content Receiver Unit 222. At Step 226, the Series Handler Unit checks if contents are in a series. If the contents are not serialized, at step 227, the contents are directly checked if the contents are delivered to a group. If the contents is are serialized, at step 228, a series buffer is provided to keep the contents until all contents in a series arrive. When pushing contents, the series buffer serializes the contents according to their sequence numbers in a series. At step

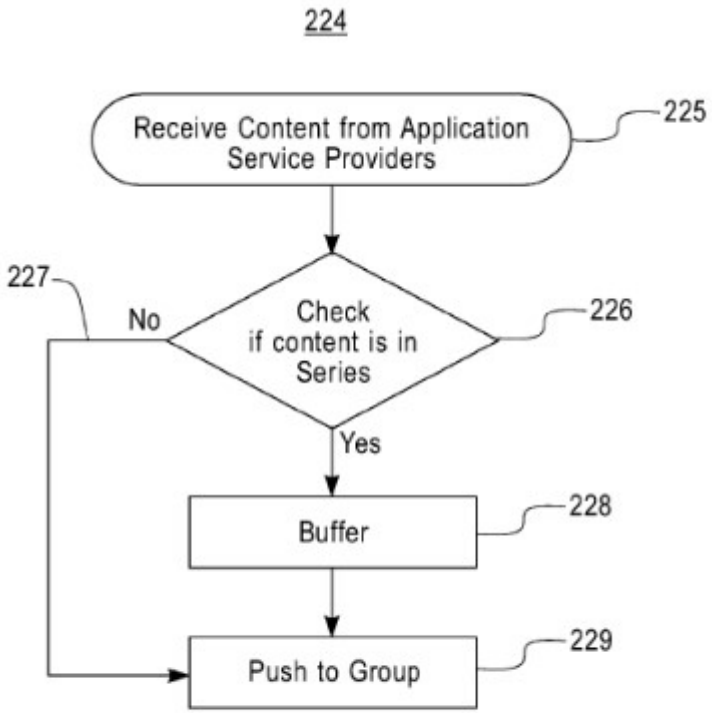
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 36 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	229, the pushed contents from the series buffer are forwarded to check if the contents are delivered to a group." *Id*., ¶37. "FIG. 5 depicts an embodiment of the Content Sender 300. The Content Sender 300 includes Queuer 320, a Queue Policy 328, Push Queues 330, DeQueuer 340, and Client and Network Specific Transport 350. After receiving contents from the Content Dispatcher Unit 228, Queuer checks urgency level of contents at step 322 (e.g., An urgent content is delivered ahead of a normal content, if they are arrived at same time.), checks customer level at step 324 (e.g., a content pushed from a "gold" customer is delivered ahead of a content from a 'silver' customer.), and checks dequeue policy at step 326 (e.g., If a content has been waiting for more than 3 seconds, the content should be ahead of any other contents, no matter what customer level.). The Content Sender employs Push Queues 330 that supports diverse priority levels. In one embodiment, the Push Queues 330 supports two different priority levels, an URGENT queue 332(1) and a NORMAL queue 332(n). DeQueuer 340 utilizes a scheduling algorithm that prevents starvation of a NORMAL queue 332(n), removes contents from top of queues, and forwards the contents to the Client and Network Specific Transport 350. The Push Queues 330 and DeQueuer 340 ensure that client devices receive contents one at a time. The Client and Network Specific Transport 350 applies different content delivery mechanisms for different clients. For example, to push contents to PC users, the Client and Network Specific Transport 350 may use TCP/IP. To push contents to cellular phone users, the Client and Network Specific Transport 350 may use carrier's own push services such as WAP (Wireless Application Protocol) Push. The Queue Policy 328 stores data related to policy (e.g., what is the level of a particular customer, how long a content may stay in a queue before it is dequeued.)" *Id*., ¶38. "In one embodiment, a content can be pushed according to a schedule of time or event. For example, when used with a group push, an application service provider can use pushes (i.e., pushing contents) for a system maintenance purpose during off-peak hours to take advantage of lower rates." *Id*., ¶44.

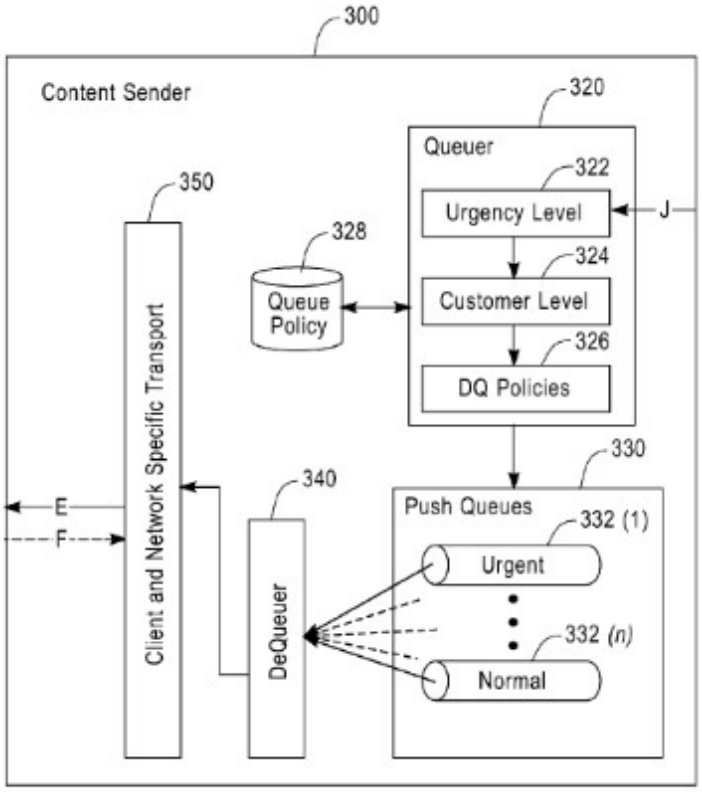
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 37 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 3 *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 38 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 4 *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 39 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 5 *Id.*, Fig. 5.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 40 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with TS.23-140 TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS.23-140, as evidenced by the citations below, discloses a message buffer system including a memory and

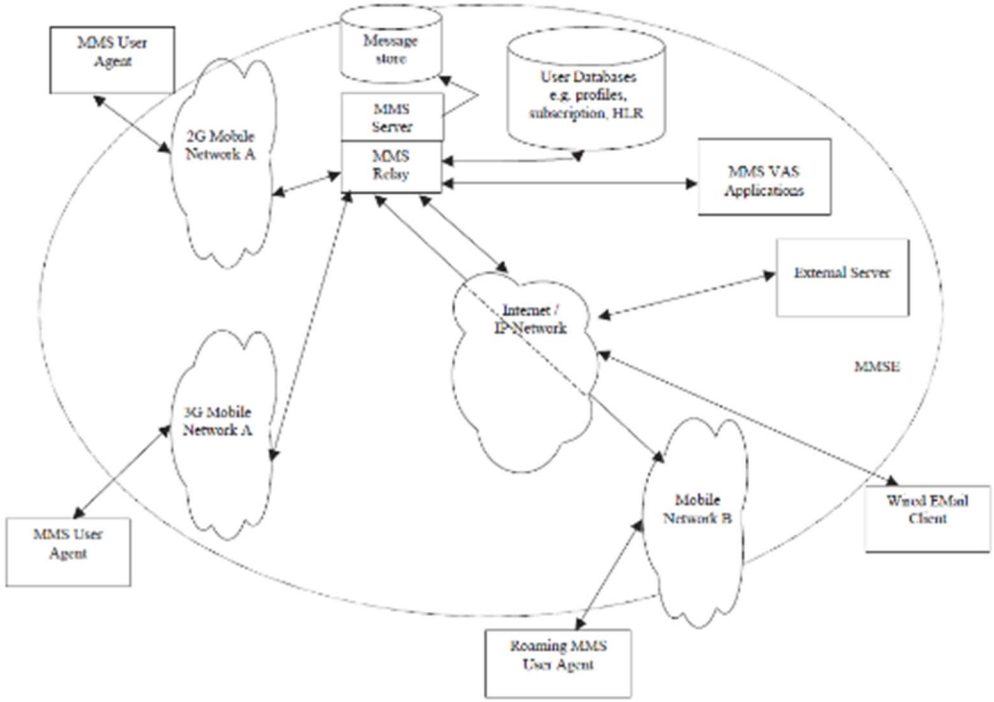
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 41 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	logic. *See, e.g.*, "The MMS Relay/Server is responsible for storage and handling of incoming and outgoing messages and for the transfer of messages between different messaging systems." TS.23-140, p. 17. "The MMS Relay/Server shall provide the following functionalities:. . . - temporary storage of messages; - ensuring that messages are not lost until successfully delivered to another MMSE element;" *Id*., p. 21. "The MMS Relay/Server may provide additional functionalities such as: . . . persistent storage of messages" *Id*. "An optional feature of MMS is the support of persistent, network-based storage, called an 'MMBox', a logical entity associated with the MMS Relay/Server into which Multimedia Messages (MMs) may be stored, retrieved, and deleted. Depending upon an operator's configuration, each subscriber may have her MMBox configured to automatically store incoming and submitted MMs, or, through supporting MMS User Agents, request that specific MMs be persistently stored on a case-by-case basis." *Id*., p. 22. "If a MMS User Agent supports submission of MMs the MMS User Agent may be able to: . . . Set the earliest desired time of delivery for the message;" *Id*., p. 26. "Upon reception of an MM from an originator MMS User Agent the originator MMS Relay/Server . . . shall retain the MM until the earliest desired time of delivery, if the optional feature of earliest time of delivery is supported by the originator MMS Relay/Server." *Id*., p. 27. "Upon reception of an MM the recipient MMS Relay/Server . . . • shall store the MM at least until o the associated time of expiry is reached,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 42 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	○ the MM is delivered, ○ the recipient MMS User Agent requests the MM to be routed forward or ○ the MM is rejected; • may store the MM into an MMBox. . . . • shall generate a notification to the recipient MMS User Agent." *Id.,* p. 28.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 43 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 2: MMS Architectural Elements** *Id.*, Fig. 2.
1[e]: the memory to buffer content from the received network element messages for which delivery is requested to any of the wireless end-user	Blackberry Push, as evidenced by the example citations below, discloses memory to buffer content from the received network element messages for which delivery is requested to any of the wireless end-user devices.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 44 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
devices,	Blackberry Push buffers content for delivery, including push content queued and retried subject to policy and connectivity. *See, e.g.*, • MDS Technical Series, p. 10: MDS Architecture • Maximum size of HTTP data that handheld can fetch from the Internet is by default 128KB (configurable) • Timeout of 2 min (configurable) if the access to the Internet/data source is not reachable or the handheld is out of coverage • MDS Technical Series, p. 47: Push Applications Push Types – Use Cases • Sample Use-Cases for Browser PUSH: – Channel Push • Push an icon that acts as a permanent link to an internal portal page or frequently visited application – Message Push • One-time or periodic pushes, such as pushing reports or alerts – Cache Push • Push other pages that are linked to a message push or cache push page • Ensures that all pages are available from local cache at a later time, even if the user is out of coverage) • MDS 4.1 Overview, p. 9: <u>Pending or offline content</u> If users are outside of a wireless coverage area, they have a number of options for managing web content requests:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 45 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• When a web page request cannot be completed, the user can save the pending request to the Messages screen. The BlackBerry Browser resends the request when the BlackBerry device is in a wireless coverage area and can notify the user when the page is available. • Users can define bookmarks for offline use, so they can view cached content without retrieving it from the server again. • Users can fill out offline forms and have the BlackBerry Browser queue and submit the forms in sequence when the BlackBerry device is in a wireless coverage area. • MDS Studio 4.1, p. 13: Messaging model BlackBerry MDS Applications are designed to store and process data locally on the BlackBerry device, without the need for a continuously available network connection. The communication model used to send and receive application messages is fully asynchronous. This means that if an outbound message is generated at a time when the BlackBerry device is outside of a wireless coverage area (for example, as a result of user input), the message is placed into an outbound queue until the wireless connection becomes available. When the message is placed into this queue, the application continues functioning, regardless of whether the message has been sent. Messages are transmitted from the queue in the background when the network connection becomes available. Similarly, inbound messages can be queued by the BlackBerry MDS Services and transmitted to the BlackBerry device only when the wireless connection becomes available. BlackBerry MDS Applications are designed to process inbound messages as asynchronous events. This asynchronous communication model is designed for applications that are inherently tolerant to sudden or unpredictable loss of connection. This feature, combined with the ability to store and process data locally, enables BlackBerry MDS Applications to be designed in such a way that application functionality and enterprise data are available, even when wireless connections are not available. Additionally, Blackberry Push supported WAP 2.0 – PAP, which utilized priorities, deliver-before, and deliver-after functionality. *See, e.g.*,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 46 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS Technical Series, p. 44: • BlackBerry enables you to pro-actively PUSH new web pages and links to your users' handhelds • Why would I want to push web pages to my users?? – Alert them of new information based on an event that occurred on the server – 'Silently' update frequently accessed pages within the browser's local cache, so the user can access the most recent information even when out of coverage – Push a permanent link to a frequently accessed site (such as an internal portal or entry point to an application) • WAP 2.0 – PAP – Acknowledgments – Reliability • Java Development 4.0, pp. 121-22: Applications can push content to handhelds using one of two methods: • Push Access Protocol (PAP), which is part of the WAP 2.0 specification • RIM push Both push service implementations support the following tasks: • sending a server-side push submission • specifying a reliability mode for the push submission • In transport-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives acknowledgements for all sent packets. • In application-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives confirmation from the application that initiated the push. • specifying a deliver-before time-stamp for the push submission • requesting a result notification of the push submission The PAP implementation supports the following additional tasks: • specifying a deliver-after timestamp for the push submission

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 47 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• cancelling a push request submission • querying the status of a push request submission PAP pushes are stored in a database, whereas RIM pushes are stored in RAM. RIM pushes may be lost if the server is rebooted. To the extent Plaintiff contends that Blackberry Push fails to disclose "memory," a POSITA would have found it obvious that Blackberry Push inherently or obviously teaches the use of memory because its specifications disclose buffering content for transmission and reception, which necessarily requires temporary storage. Thus, a POSITA would recognize that Blackberry Push's buffering/temporary storage of the undelivered message directly imply the presence and use of memory in the system. Further, it would have been obvious to one of ordinary skill in the art based on Blackberry Push alone or in combination with (i) the general knowledge of one of ordinary skill in the art and/or (ii) the teachings of any of the charted references with respect to this limitation, including, without limitation, for the following references: Lazaridis, Houghton, Munson, TS.23-140. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 48 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses a message buffer system with the memory (e.g., primary memory store) to buffer content (e.g., repackage content) from the received network element messages (e.g., messages from other systems connected to LAN and to external networks) for which delivery is requested to a given one of the wireless end-user devices (e.g., mobile data communication device). *See, e.g.*, "A system and method for pushing information from a host system to a mobile data communication device upon sensing a triggering event is disclosed. A redirector program operating at the host system enables a user to continuously redirect certain user-selected data items from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. The redirector program operates in connection with event generating applications and repackaging systems at the host system to configure and detect a particular user-defined event, and then to repackage the user-selected data items in an electronic wrapper prior to pushing the data items to the mobile device. The mobile device and the host system share a common electronic address so that messages generated at either the host system or the mobile data communication device are configured using the common electronic address." Lazaridis, Abstract.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 49 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Software executing on the mobile data communication device then transmits commands via the local communications link to the host system to cause the host to begin transmitting the user's data items for storage in a memory bank of the mobile device." *Id.*, 1:50-54. "Other sub-systems for creating triggering events and repackaging the user's data items could also be present at the host system. The host system also includes a primary memory store where the user's data items are normally stored." *Id.*, 3:4-8. "Once an event has triggered redirection of the user data items, the host system then repackages these items in a manner that is transparent to the mobile data communication device, so that information on the mobile device appears similar to information on the user's host system. The preferred repackaging method includes wrapping the user data items in an E-mail envelope that corresponds to the address of the mobile data communication device, although, alternatively, other repackaging methods could be used with the present invention, such as special-purpose TCP/IP wrapping techniques, or other methods of wrapping the user selected data items. The repackaging preferably results in E-mail messages generated by the user from the mobile device to be transmitted from the host system, thus enabling the user to appear to have a single E-mail address, such that the recipients of messages sent from the mobile communications device do not know where the user was physically located when the message was first sent. The repackaging also permits both messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed and decompressed." *Id.*, 3:66-4:19. "Message C in FIG. 1 represents an external message from a sender that is not directly connected to LAN 14, such as the user's mobile data communication device 24, some other user's mobile device (not shown), or any user connected to the Internet 18. Message C also represents a command message from the user's mobile data communication device 24 to the host system 10. As described in more detail in FIG. 3, the host system 10 preferably includes, along with the typical hardware and software associated with a workstation

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 50 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	or desktop computer, the redirector program 12, a TCP/IP subsystem 43, a primary message store 40, an E-mail subsystem 44, a screen saver subsystem 48, and a keyboard subsystem" *Id.*, 5:44-56. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "FIG. 1 shows an E-mail message A being communicated over LAN 14 from computer 26 to the user's desktop system 10 (also shown in FIG. 1 is an external message C, which could be an Email message from an Internet user, or could be a command message from the user's mobile device 24). Once the message A (or C) reaches the primary message store of the host system 10, it can be detected and acted upon by the redirection software 12. The redirection software 12 can use many methods of detecting new messages." *Id.*, 7:31-38. "Messaging API (MAPI), in which programs, such as the redirector program 12, register for notifications or `advise syncs` when changes to a mailbox take place. Other methods of detecting new messages could also be used with the present invention. Assuming that the redirector program 12 is activated, and has been configured by the user (either through the sensing of an internal, network or external event) to replicate certain user data items (including messages of type A or C) to the mobile device 24, when the message A is received at the host system 10, the redirector program 12 detects its presence and prepares the message for

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 51 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	redirection to the mobile device 24. In preparing the message for redirection, the redirector program 12 could compress the original message A, could compress the message header, or could encrypt the entire message A to create a secure link to the mobile device 24. Also programmed into the redirector 12 is the address of the user's mobile data communication device 24, the type of device, and whether the device 24 can accept certain types of attachments, such as word processing or voice attachments." *Id.*, 7:41-7:60. After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*8:32-55. "In the case where message C is representative of an external message from a computer on the Internet 18 to the host system 10 and the host has been configured to redirect messages of type C, then in a similar manner to message A, message C would be repackaged with an outer envelope B and transmitted to the user's mobile device 24." *Id.*, 8: 56-61. "FIG. 2 is an alternative system diagram showing the redirection of user data items from a network server 11 to the user's mobile data communication device 24, where the redirector software 12 is operating at the server 11. This configuration is particularly advantageous for use with message servers such as Microsoft's®

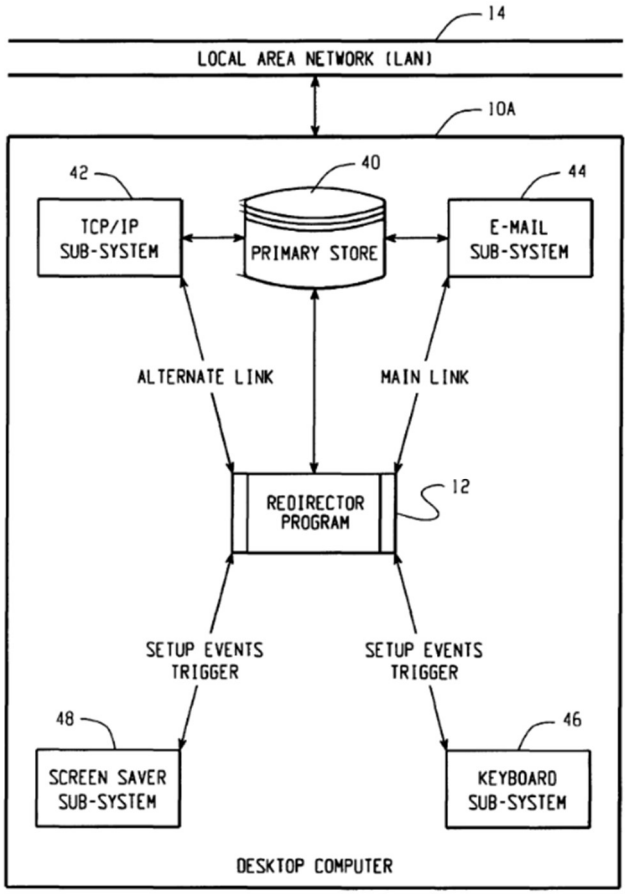
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 52 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Exchange Server, which is normally operated so that all user messages are kept in one central location or mailbox store on the server instead of in a store within each user's desktop PC. This configuration has the additional advantage of allowing a single system administrator to configure and keep track of all users having messages redirected. If the system includes encryption keys, these too can be kept at one place for management and update purposes." *Id.*, 9:20-33. "If the message C is from an Internet computer to the user's desktop system and the user has redirection capabilities, then the server 11 detects the message C, repackages it using electronic envelope B, and redirects the repackaged message (C in B) to the user's mobile device 24. If the message C is a command message from the user's mobile device 24, then the server 11 simply acts upon the command message." *Id.*, 10:2-8. "The desktop system 10 is connected to LAN 14, and can send and receive data, messages, signals, event triggers, etc., to and from other systems connected to the LAN 14 and to external networks 18, 22, such as the Internet or a wireless data network, which are also coupled to the LAN 14. In addition to the standard hardware, operating system, and application programs associated with a typical microcomputer or workstation, the desktop system 10 includes the redirector program 12, a TCP/IP sub-system 42, an E-mail sub-system 44, a primary data storage device 40, a screen saver sub-system 48, and a keyboard sub-system 46. The TCP/IP and E-mail subsystems 42, 44 are examples of repackaging systems that can be used to achieve the transparency of the present invention, and the screen saver and keyboard sub-systems 46, 48 are examples of event generating systems that can be configured to generate event messages or signals that trigger redirection of the user selected data items." *Id.*, 10:21-38. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 53 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. "When a message or signal is detected, the program 12 determines 58 whether it is one of the trigger events that has been configured by the user to signal redirection. If so, then at step 60 a trigger flag is set, indicating that subsequently received user data items (in the form of messages) that have been selected for redirection should be pushed to the user's mobile data communication device 24." *Id.*, 12:33-39. "If the trigger flag is set, then at step 70, the redirector 12 causes the repackaging system (E-Mail or TCP/IP) to add the outer envelope to the user data item, and at step 72 the repackaged data item is then redirected to the user's mobile data communication device 24 via LAN 14, WAN 18, wireless gateway 20 and wireless network 22." *Id.*, 12:50-56. "If, however, the redirection flag is on, then at step 110 the reply message is repackaged with the outer envelope having the addressing information of the user's desktop system 10, and the repackaged message is then transmitted to the desktop system 10 at step 106." *Id.*, 13:62-67. "If, however, the mobile determines that the user is using the redirector software 12 at the desktop system 10, then control passes to step 110, where the outer envelope is added to the message. The repackaged original message is then transmitted to the desktop system 10 at step 106, which, as described previously, strips the outer envelope and routes the message to the correct destination. Following transmission of the message at step 106, control of the mobile returns to step 82 and waits for additional messages or signals." *Id.*, 14:14-23.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 54 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 3 *Id.*, Fig. 3.

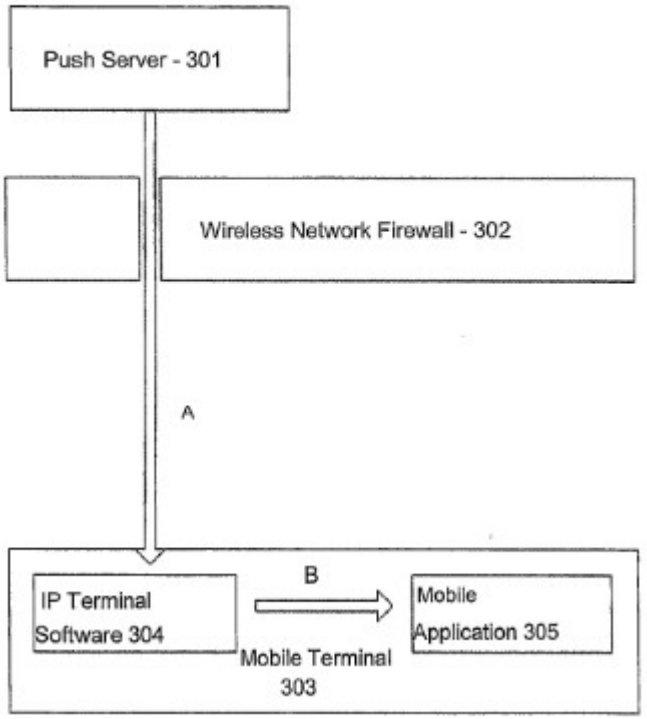
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 55 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Houghton Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill.

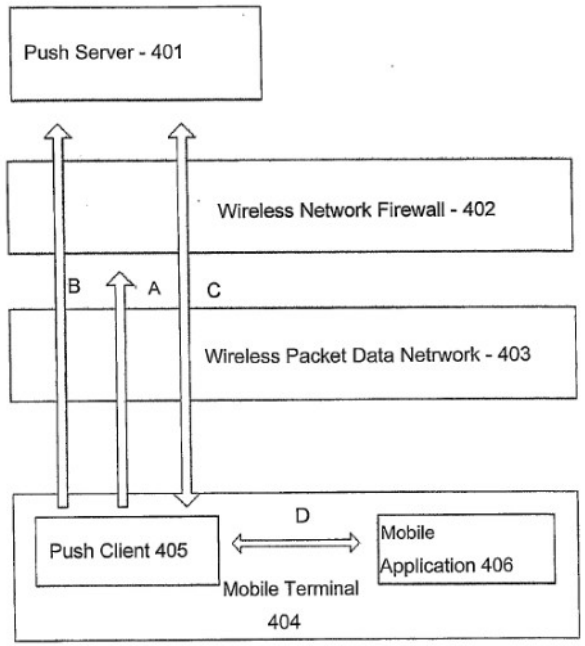
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 56 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Houghton, as evidenced by the example citations below, discloses the memory to buffer content from the received network element messages for which delivery is requested to a given one of the wireless end-user devices. *See, e.g.*, "Push technology is characterized by the server proactively sending documents or information to a client or user. E-mail (Electronic mail) is actually a special kind of push technology that has been prevalent for a long time. The first and most common general category of mobile content push technologies is SMS (Short Message Service) based technologies. SMS and MMS (Multi-media Messaging Service) are the most common forms of 'push' messaging in the mobile industry. These store and forward messaging systems run by mobile telecommunications operators push all or part of the data to the mobile terminal over closed telecommunications signaling networks such as SS7 (Signaling System Seven)." Houghton, p. 3. "These server computers such as a telecommunications operator SIP Server or WAP Gateway is used to control and deliver messages and store messages when the mobile terminal is unavailable and deliver them on the senders behalf, possibly later then the terminal becomes available, or to forward them to a different gateway device, for example if the user is roaming to a different controlled mobile network ." *Id.*, p. 7. "When in proximity to a more capable network, if a connection can be established to this alternative broadband network access point, it is used preferentially and intelligence normally located in this broadband access point would route messages received on behalf of the wireless terminal to and from the server." *Id.*, pp. 13-14.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 57 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 3  *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 58 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 59 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Munson Munson and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Munson discloses a method of "pushing contents to client devices," including "group pushes" in which content is buffered and sent to multiple devices simultaneously, "serializ[ing]" content such that a series of messages are delivered to a particular device simultaneously. Munson, Abstract, ¶37, Fig. 4. A POSITA would have been motivated to combine Munson with Blackberry Push because a POSITA would have desired to support diverse client devices and network environments, as well as to leverage an existing, proven technology for wireless content delivery, e.g., Blackberry Push. Munson similarly teaches messages can be delivered from memory "according to a schedule of time or event" such as coordinating pushes "for a system maintenance purpose during off-peak hours." Munson, ¶¶40, 44. Thus, Munson provides additional details about how to implement a store-and-forward system by providing the computer memory at a server, and the computer programming to deliver messages from memory (including upon occurrence of certain events). *Id*., ¶¶36-38, 44, Fig. 4. Therefore, a POSITA would have been motivated to incorporate computer memory and programming into Blackberry Push to store and deliver messages from memory, consistent with Munson, to improve system flexibility and reliability. A POSITA would have had a reasonable expectation of success in combining Blackberry Push and Munson because Munson provides implementation details such push message system. *See, e.g.*, Munson, ¶¶36-38. A POSITA would have been motivated and had a reasonable expectation of success in incorporating conditions Munson teaches providing additional flexibility in how messages are delivered in PAP, used by Blackberry Push. This combination of Blackberry Push and Munson would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Munson, as the communication protocol because Blackberry Push was a well-known, industry-

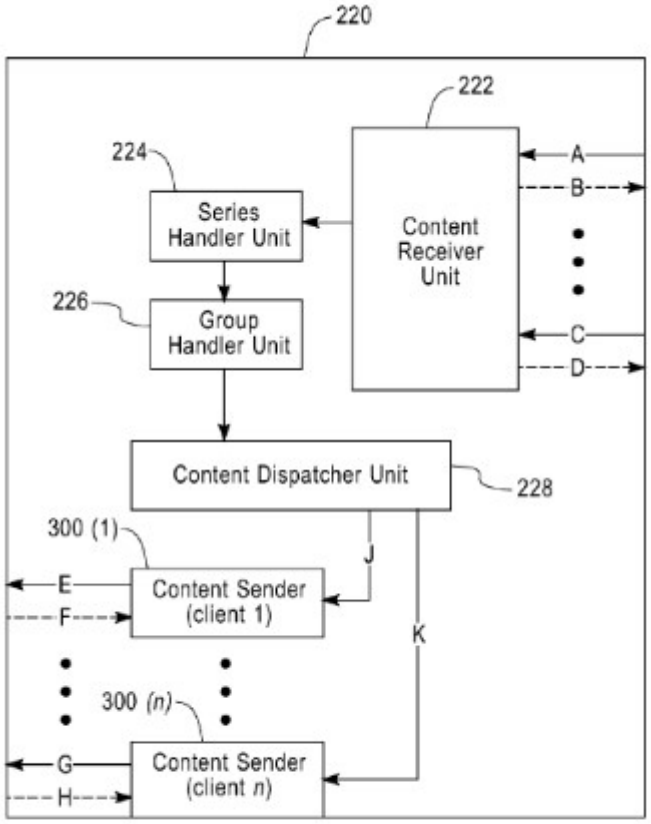
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 60 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	leading system. Munson, ¶¶35-36. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Munson push messaging system without requiring inventive skill. Munson, as evidenced by the example citations below, discloses the memory to buffer content from the received network element messages for which delivery is requested to a given one of the wireless end-user devices. *See, e.g.*, "FIG. 3 illustrates an embodiment of the CPS 220. The CPS 220 includes a Content Receiver Unit 222, a Series Handler Unit 224, a Group Handler Unit 226, a Content Dispatcher Unit 228, and a plurality of Content Senders such as 300(1) and 300(*n*). The Content Receiver Unit 222 receives contents from application service providers 210, sends acknowledge receipts to application service providers 210, and forwards the contents to a Series Handler Unit 224. If the contents are received in a series, the Series Handler Unit 224 ensures the contents received as part of a series are pushed in the order specified by the application service providers. In one embodiment, the CPS 220 allows an application service provider to send contents to a group of client devices. Then, the Group Handler Unit 226 is responsible for looking up members of a group and performing an individual push on each member (i.e., pushing a content to each client device in the group). The Content Sender 300 performs actual transport of data of contents to a client device. There is one Content Sender 300 for each active client device. The Content Dispatcher Unit 228 maintains a list of the Content Senders 300. If a Content Sender is currently pushing a content to a client, the Content Dispatcher Unit 228 enqueues the content in the Content Sender. If a Content Sender is not pushing a content, the Content Dispatcher Unit 228 enqueues the content in the Content Sender and activates the Content Sender to start pushing." *Id*., ¶36. "FIG. 4 depicts a flow chart that the Series Handler Unit 224 utilizes. At step 225, the Series Handler Unit 224 receives contents from application service providers through the Content Receiver Unit 222. At Step 226, the Series Handler Unit checks if contents are in a series. If the contents are not serialized, at step 227, the contents are directly checked if the contents are delivered to a group. If the contents is are serialized, at step 228, a series buffer is provided to keep the contents until all contents in a series arrive. When pushing

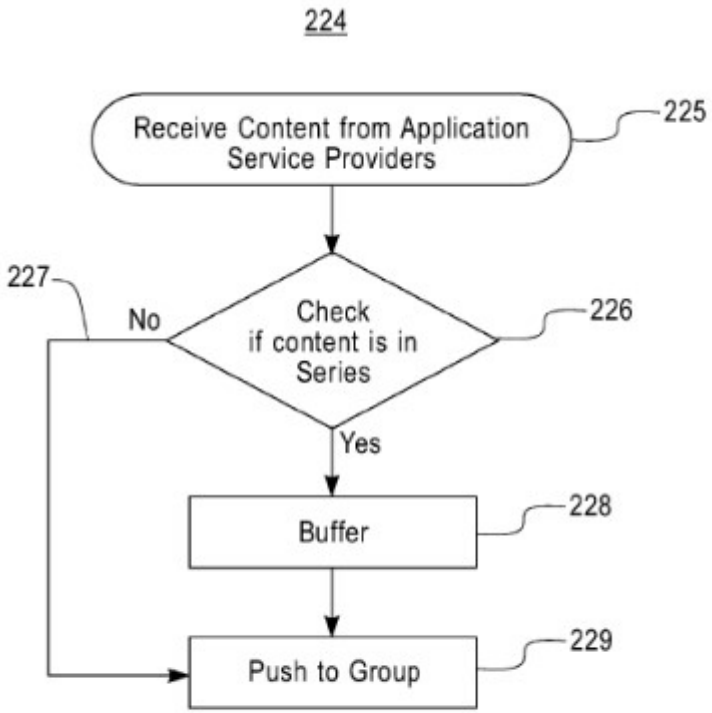
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 61 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	contents, the series buffer serializes the contents according to their sequence numbers in a series. At step 229, the pushed contents from the series buffer are forwarded to check if the contents are delivered to a group." *Id*., ¶37. "FIG. 5 depicts an embodiment of the Content Sender 300. The Content Sender 300 includes Queuer 320, a Queue Policy 328, Push Queues 330, DeQueuer 340, and Client and Network Specific Transport 350. After receiving contents from the Content Dispatcher Unit 228, Queuer checks urgency level of contents at step 322 (e.g., An urgent content is delivered ahead of a normal content, if they are arrived at same time.), checks customer level at step 324 (e.g., a content pushed from a "gold" customer is delivered ahead of a content from a 'silver' customer.), and checks dequeue policy at step 326 (e.g., If a content has been waiting for more than 3 seconds, the content should be ahead of any other contents, no matter what customer level.). The Content Sender employs Push Queues 330 that supports diverse priority levels. In one embodiment, the Push Queues 330 supports two different priority levels, an URGENT queue 332(1) and a NORMAL queue 332(n). DeQueuer 340 utilizes a scheduling algorithm that prevents starvation of a NORMAL queue 332(n), removes contents from top of queues, and forwards the contents to the Client and Network Specific Transport 350. The Push Queues 330 and DeQueuer 340 ensure that client devices receive contents one at a time. The Client and Network Specific Transport 350 applies different content delivery mechanisms for different clients. For example, to push contents to PC users, the Client and Network Specific Transport 350 may use TCP/IP. To push contents to cellular phone users, the Client and Network Specific Transport 350 may use carrier's own push services such as WAP (Wireless Application Protocol) Push. The Queue Policy 328 stores data related to policy (e.g., what is the level of a particular customer, how long a content may stay in a queue before it is dequeued.)" *Id*., ¶38. "In one embodiment, a content can be pushed according to a schedule of time or event. For example, when used with a group push, an application service provider can use pushes (i.e., pushing contents) for a system maintenance purpose during off-peak hours to take advantage of lower rates." *Id*., ¶44.

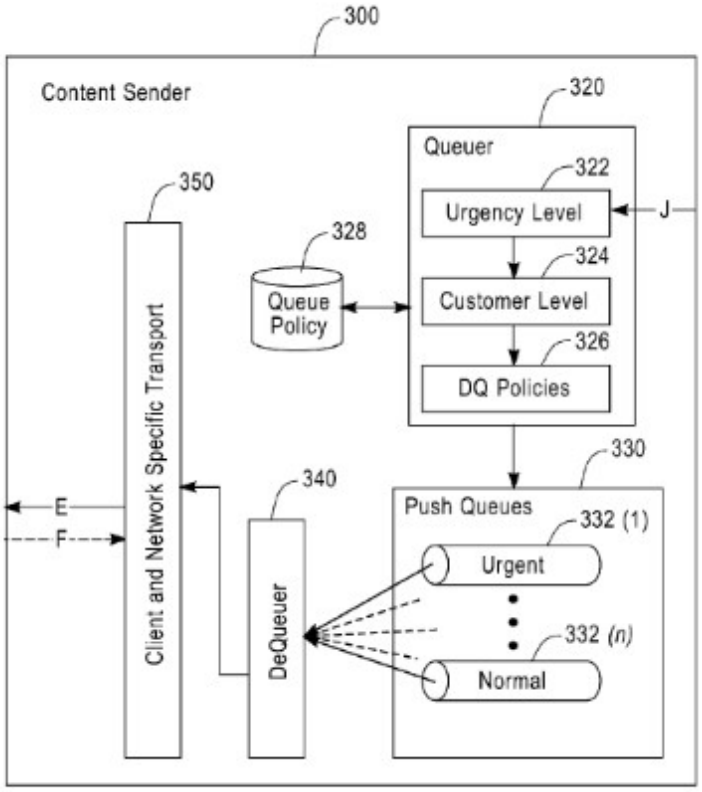
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 62 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 3 *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 63 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 4 *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 64 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 5 *Id.*, Fig. 5.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 65 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with TS.23-140 TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS.23-140, as evidenced by the citations below, discloses the memory (e.g., message store and/or MMBox)

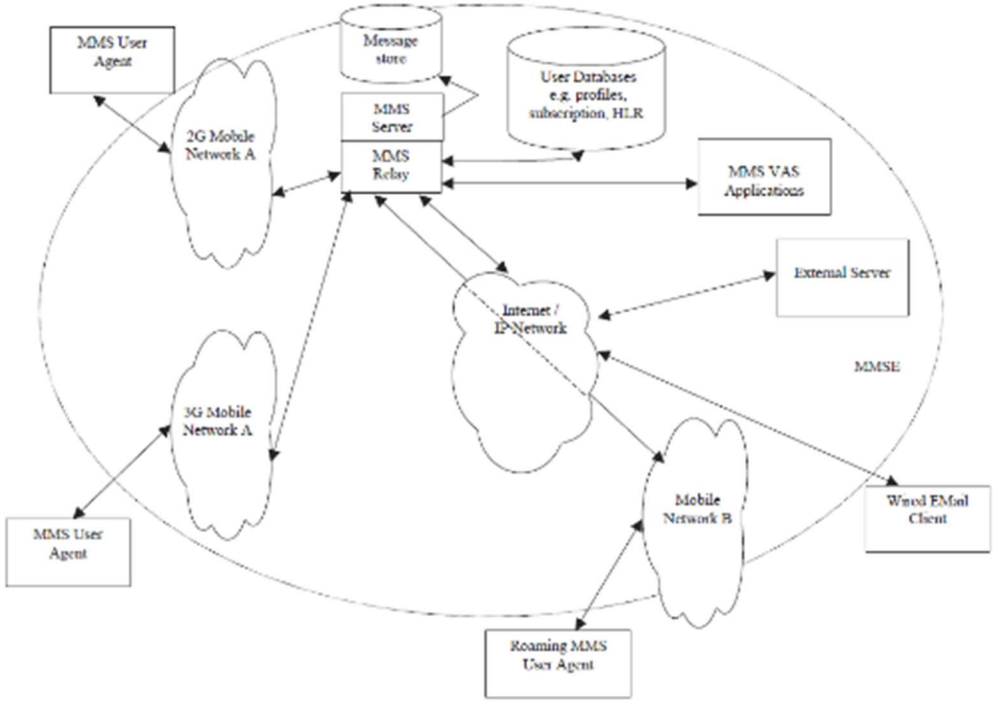
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 66 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	to buffer content from the received network element messages for which delivery is requested to a given one of the wireless end-user devices (e.g., storing the received messages before delivery to another network element such as a UE/mobile device). *See, e.g.*, "The MMS Relay/Server is responsible for storage and handling of incoming and outgoing messages and for the transfer of messages between different messaging systems." TS.23-140, p. 17. "The MMS Relay/Server shall provide the following functionalities:. . . - temporary storage of messages; - ensuring that messages are not lost until successfully delivered to another MMSE element;" *Id*., p. 21. "The MMS Relay/Server may provide additional functionalities such as: . . . persistent storage of messages" *Id*. "An optional feature of MMS is the support of persistent, network-based storage, called an 'MMBox', a logical entity associated with the MMS Relay/Server into which Multimedia Messages (MMs) may be stored, retrieved, and deleted. Depending upon an operator's configuration, each subscriber may have her MMBox configured to automatically store incoming and submitted MMs, or, through supporting MMS User Agents, request that specific MMs be persistently stored on a case-by-case basis." *Id*., p. 22. "If a MMS User Agent supports submission of MMs the MMS User Agent may be able to: . . . Set the earliest desired time of delivery for the message;" *Id*., p. 26. "Upon reception of an MM from an originator MMS User Agent the originator MMS Relay/Server . . . shall retain the MM until the earliest desired time of delivery, if the optional feature of earliest time of delivery is supported by the originator MMS Relay/Server." *Id*., p. 27. "Upon reception of an MM the recipient MMS Relay/Server . . . • shall store the MM at least until

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 67 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: **Blackberry Push**
	○ the associated time of expiry is reached, ○ the MM is delivered, ○ the recipient MMS User Agent requests the MM to be routed forward or ○ the MM is rejected; • may store the MM into an MMBox. . . . • shall generate a notification to the recipient MMS User Agent." *Id.,* p. 28.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 68 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 2: MMS Architectural Elements** *Id.*, Fig. 2.
1[f]: the logic to determine when one of a plurality of message delivery triggers for a given one of the wireless end-user devices has occurred,	Blackberry Push, as evidenced by the example citations below, discloses logic to determine when one of a plurality of message delivery triggers for a given one of the wireless end-user devices has occurred, wherein for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a message delivery trigger, and for at least one of the message delivery triggers, the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 69 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
wherein for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a message delivery trigger, and for at least one of the message delivery triggers, the trigger is an occurrence of an asynchronous event with time-critical messaging needs, and	trigger is an occurrence of an asynchronous event with time-critical messaging needs. Blackberry Push implements triggers and conditions for delivery other than message receipt at the server, including asynchronous events. *See, e.g.*, • MDS Technical Series, p. 47: Push Applications Push Types – Use Cases • Sample Use-Cases for Browser PUSH: – Channel Push • Push an icon that acts as a permanent link to an internal portal page or frequently visited application – Message Push • One-time or periodic pushes, such as pushing reports or alerts – Cache Push • Push other pages that are linked to a message push or cache push page • Ensures that all pages are available from local cache at a later time, even if the user is out of coverage) • MDS Studio 4.1, pp. 13-14: Messaging model BlackBerry MDS Applications are designed to store and process data locally on the BlackBerry device, without the need for a continuously available network connection. The communication model used to send and receive application messages is fully asynchronous. This means that if an outbound message is generated at a time when the BlackBerry device is outside of a wireless coverage area (for example, as a result of user input), the message is placed into an outbound queue until the wireless connection becomes available. When the message is placed into this queue, the application continues functioning, regardless of whether the message has been sent. Messages are transmitted from the queue in the background when the network connection becomes available.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 70 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Similarly, inbound messages can be queued by the BlackBerry MDS Services and transmitted to the BlackBerry device only when the wireless connection becomes available. BlackBerry MDS Applications are designed to process inbound messages as asynchronous events. This asynchronous communication model is designed for applications that are inherently tolerant to sudden or unpredictable loss of connection. This feature, combined with the ability to store and process data locally, enables BlackBerry MDS Applications to be designed in such a way that application functionality and enterprise data are available, even when wireless connections are not available. In BlackBerry MDS Applications, the communication model consists of the following items: • delivery mode for application messages • handling messages when a BlackBerry device is outside of a wireless coverage area • message queue sizes • categories of notification messages • size limitations for messages Delivery Modes Messages that pass between BlackBerry MDS Applications and the BlackBerry MDS Services are categorized into two delivery modes: best-effort and standard. The default delivery mode is standard. Best-effort The best-effort delivery mode is a fire-and-forget mode. Messages might not be delivered, and they might be delivered in a different order than they were sent from the application to the BlackBerry MDS Services or from the BlackBerry MDS Services to the application. Standard Messages sent from the application are designed to reach the BlackBerry device, even if the BlackBerry device spends time outside of a wireless coverage area or if the BlackBerry MDS Services stop. The sending side transmits the message when a connection becomes available. The application does not confirm receipt of messages, so the application might not process a message even after it reaches the BlackBerry device.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 71 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Similarly, the BlackBerry MDS Services might not process a message from the application even when the message reaches the BlackBerry MDS Services server. Standard mode messages are designed to be delivered and processed in the order in which they were generated. Handling messages when a BlackBerry device is outside of a wireless coverage area When a BlackBerry device is outside of a wireless coverage area, outbound standard and reliable application messages remain queued on the BlackBerry device until the BlackBerry device returns to a wireless coverage area, at which time the messages are transmitted. Best-effort messages are dropped when a BlackBerry device is outside of a wireless coverage area. Message queue sizes BlackBerry MDS Applications have restricted inbound and outbound message queue sizes. These restrictions are imposed by policies sent to the BlackBerry MDS Runtime software on the BlackBerry devices. The system administrator of the BlackBerry MDS Services can push new policies to the BlackBerry MDS Runtime, which might change the limitations of an application's inbound and outbound queue sizes. • MDS Technical Series, p. 10: MDS Architecture • Maximum size of HTTP data that handheld can fetch from the Internet is by default 128KB (configurable) • Timeout of 2 min (configurable) if the access to the Internet/data source is not reachable or the handheld is out of coverage • MDS Studio 4.1, p. 15: Message size limitations The maximum size for any inbound message is 32 KB. If the result of an operation performed by a web service requires a message size greater than 32 KB to be sent from the BlackBerry MDS Services to the BlackBerry MDS Application, the BlackBerry MDS Services send an error message to the application.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 72 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• Java Development 4.0, p. 123: The following headers are valid for RIM push requests: X-RIM-Push-ID: This header specifies a unique message ID, which can be used to cancel or check the status of a message. Typically, specify a URL in combination with a value, such as such as 123@rim.com. X-RIM-Push-NotifyURL This header specifies the URL to which a result notification is sent. X-RIM-Push-Reliability-Mode This header specifies the delivery reliability mode of the content-transport-level (TRANSPORT), application-level (APPLICATION) and application preferred (APPLICATIONPREFERRED). X-RIM-Push-Deliver-Before This header specifies the date and time by which the content must be delivered to the handheld. Content that has not been delivered before this date is not delivered. X-RIM-Push-Priority This header specifies the priority of channel push messages. Permitted strings include none (default), low, medium, and high. If the priority is low, medium or high, users receive notification of channel updates. If the priority is high, a status dialog accompanies the notification Additionally, Blackberry Push supported WAP 2.0 – PAP, which utilized priorities, deliver-before, and deliver-after functionality. *See, e.g.*, • MDS Technical Series, p. 44: • BlackBerry enables you to pro-actively PUSH new web pages and links to your users' handhelds • Why would I want to push web pages to my users?? – Alert them of new information based on an event that occurred on the server – 'Silently' update frequently accessed pages within the browser's local cache, so the user can access the most recent information even when out of coverage – Push a permanent link to a frequently accessed site (such as an internal portal or entry point to an application) • WAP 2.0 – PAP – Acknowledgments – Reliability

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 73 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• Java Development 4.0, pp. 121-22: Applications can push content to handhelds using one of two methods: • Push Access Protocol (PAP), which is part of the WAP 2.0 specification • RIM push Both push service implementations support the following tasks: • sending a server-side push submission • specifying a reliability mode for the push submission • In transport-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives acknowledgements for all sent packets. • In application-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives confirmation from the application that initiated the push. • specifying a deliver-before time-stamp for the push submission • requesting a result notification of the push submission The PAP implementation supports the following additional tasks: • specifying a deliver-after timestamp for the push submission • cancelling a push request submission • querying the status of a push request submission PAP pushes are stored in a database, whereas RIM pushes are stored in RAM. RIM pushes may be lost if the server is rebooted. To the extent Plaintiff contends that Blackberry Push fails to disclose this limitation, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Houghton, Munson, TS.23-140.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 74 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Lazaridis Lazaridis and Blackberry Push are analogous art to the ’320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet.” *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis’s push-type command to mobile communication devices by a remote server with Blackberry Push’s system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis’s push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the logic to determine when one of a plurality of message delivery triggers for the given one of the wireless end-user devices has occurred (e.g., the redirector detecting that a user-defined event trigger has taken place), wherein for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 75 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	message delivery trigger (e.g., calendar event, screen saver activation, keyboard timeout, programmable timer, and other event triggers), and for at least one of the message delivery triggers, the trigger is an occurrence of an asynchronous event with time-critical messaging needs (e.g., a gateway that forms a connection between the WAN 18 and some other type of network, such as an asynchronous connection). *See, e.g.*, "A system and method for pushing information from a host system to a mobile data communication device upon sensing a triggering event is disclosed. A redirector program operating at the host system enables a user to continuously redirect certain user-selected data items from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. The redirector program operates in connection with event generating applications and repackaging systems at the host system to configure and detect a particular user-defined event, and then to repackage the user-selected data items in an electronic wrapper prior to pushing the data items to the mobile device. The mobile device and the host system share a common electronic address so that messages generated at either the host system or the mobile data communication device are configured using the common electronic address." Lazaridis, Abstract. "Instead of warehousing (or storing) the user's data items at the host system and then 'synchronizing' the mobile data communication device to data items stored at the host system when the mobile device requests that such items of information be communicated to it, the present invention employs a "push" paradigm that continuously packages and retransmits the user-selected items of information to the mobile data communication device in response to a triggering event detected at the host system. Wireless mobile data communications devices, especially those that can return a confirmation signal to the host that the pushed data has been received are especially well suited for this type of push paradigm." *Id.*, 1:26-38. "There remains a more particular need for such a system and method where user-selected data items are continuously 'pushed' from the host system to the mobile data communication device upon the occurrence of one or more user defined triggering events. There remains an additional need for such a system and

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 76 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	method that provides flexibility in the types and quantities of user data items that are pushed from the host system to the mobile data communication device and that also provides flexibility in the configuration and types of events that can serve to trigger the redirection of the user data items" *Id.*, 2:27-38. "The present invention overcomes the problems noted above and satisfies the needs in this field for a system and method of pushing user-selected data items from a host system to a user's mobile data communication device upon detecting the occurrence of one or more user-defined event triggers." *Id.*, 2:48-53. "A redirector program operating at the host system enables the user to redirect or mirror certain user-selected data items (or parts of data items) from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. Also operating at the host system are various sub-systems that can be configured to create triggering events, such as a screen saver sub-system or a keyboard sub-system, as well as sub-systems for repackaging the user's data items for transparent delivery to the mobile data device, such as a TCP/IP sub-system or one or more E-Mail sub-systems. Other sub-systems for creating triggering events and repackaging the user's data items could also be present at the host system. The host system also includes a primary memory store where the user's data items are normally stored." *Id.*, 2:61-3:8. "Using the redirector program, the user can select certain data items for redirection, such as E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal reminders etc. Having selected the data items for redirection, the user can then configure one or more event triggers to be sensed by the redirector program to initiate redirection of the user data items. These user-defined trigger points (or event triggers) include external events, internal events and networked events. Examples of external events include: receiving a message from the user's mobile data communication device to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 77 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the examples of the types of user-defined events that can trigger the redirector program to push data items from the host to the mobile device. Although in the preferred embodiment it is anticipated that the configuration that specifies which data items will be redirected and in what form will be set at the host system, it is within the scope of this invention that such configuration may be set or modified through data sent from the mobile communications device." *Id.*, 3:9-35. "In an alternative system and method, the redirector program executes on a network server, and the server is programmed to detect numerous redirection event triggers over the network from multiple user desktop computers coupled to the server via a LAN. The server can receive internal event triggers from each of the user desktops via the network, and can also receive external event triggers, such as messages from the users' mobile data communication devices. In response to receiving one of these triggers, the server redirects the user's data items to the proper mobile data communication device." *Id.*, 4:19-29. "In another alternative configuration of the present invention, a redirector program operates at both the host system and at the user's mobile data communication device. In this configuration, the user's mobile device operates similarly to the host system described below, and is configured in a similar fashion to push certain user-selected data items from the mobile device to the user's host system (or some other computer) upon detecting an event trigger at the mobile device. This configuration provides two-way pushing of information from the host to the mobile device and from the mobile device to the host." *Id.*, 4:39-49. "The primary advantage of the present invention is that it provides a system and method for triggering the continuous and real-time redirection of user-selected data items from a host system to a mobile data communication device. Other advantages of the present invention include: (1) flexibility in defining the types of user data to redirect, and in defining a preferred list of message types that are to be redirected or preferred senders whose messages are to be redirected; (2) flexibility in configuring the system to respond to numerous internal, external and networked triggering events; (3) trans- parent repackaging of the user data items in a variety of ways such that the mobile data communication device appears as though it were

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 78 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the host system; (4) integration with other host system components such as E-mail, TCP/IP, keyboard, screen saver, webpages and certain programs that can either create user data items or be configured to provide trigger points; and (5) the ability to operate locally on a user's desktop system or at a distance via a network server. *Id.*, 4:50-67. "A gateway forms a connection or bridge between the WAN 18 and some other type of network, such as an RF wireless network, cellular network, satellite network, or other synchronous or asynchronous land-line connection." *Id.*, 6:2-6. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 79 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the events that could be used with the present invention to initiate replication of the user-selected data items from the host system 10 to the mobile device 24." *Id.*, 7:8-30. "The event triggers are preferably detected at the user's desktop system 10, 26, 28 and can be any of the external, internal or network events listed above. The desktop systems 10, 26, 28 preferably detect these events and then transmit a message to the server computer 11 via LAN 14 to initiate redirection." 9:40-46. "As described previously, the present invention can be triggered to begin redirection upon detecting numerous external, internal and networked events, or trigger points. Examples of external events include: receiving a command message from the user's mobile data communication device 24 to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer that is connected to the host system via a network to initiate redirection." *Id.*, 11:28-42. "Once the redirector program is configured 50, the trigger points (or event triggers) are enabled at step 52. The program 12 then waits 56 for messages and signals 54 to begin the redirection process. A message could be an E-Mail message or some other user data item than may have been selected for redirection, and a signal could be a trigger signal, or could be some other type of signal that has not been configured as an event trigger. When a message or signal is detected, the program determines 58 whether it is one of the trigger events that has been configured by the user to signal redirection. If so, then at step 60 a trigger flag is set, indicating that subsequently received user data items (in the form of messages) that have been selected for redirection should be pushed to the user's mobile data communication device 24. If the message or signal

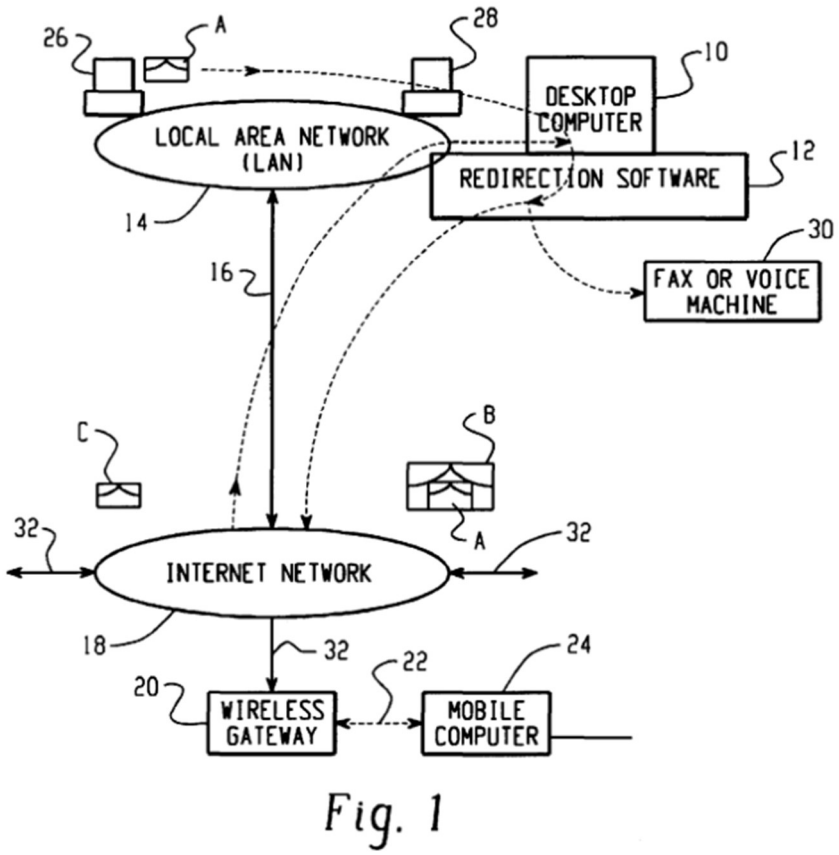
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 80 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	54 is not a trigger event, the program then determines at steps 62, 68 and 66 whether the message is, respectively, a system alarm 62, an E-Mail message 64, or some other type of information that has been selected for redirection. If the message or signal is none of these three items, then control returns to step 56, where the redirector waits for additional messages 54 to act upon. If, however the message is one of these three types of information, then the program 12 determines, at step 68, whether the trigger flag has been set, indicating that the user wants these items redirected to the mobile device." *Id.*, 12:26-51. "A user of the present invention can configure the redirector program 12 to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10, or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it. Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection." Lazaridis, 15:65-16:48.

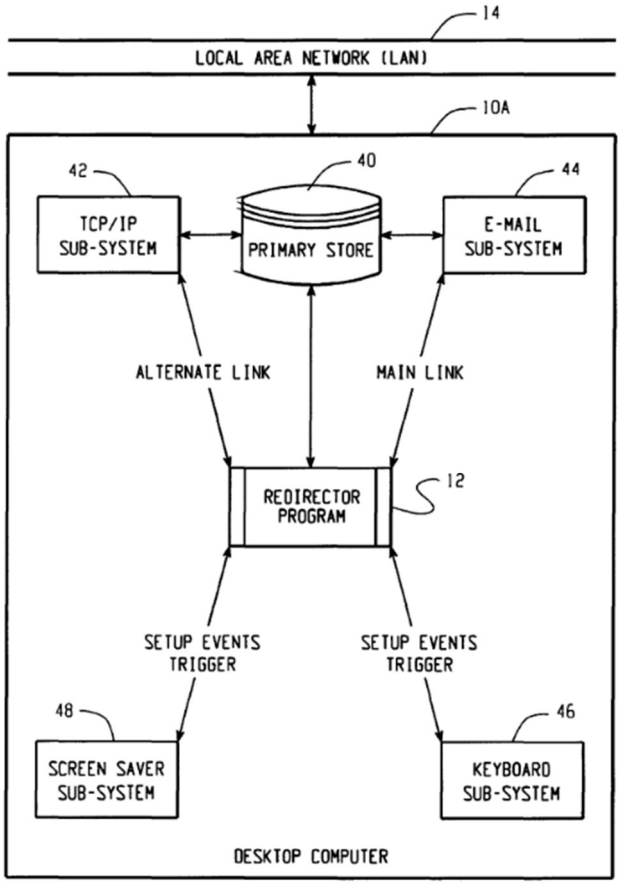
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 81 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	A method of redirecting messages between a host system and a mobile data communication device, comprising the steps of: configuring one or more redirection events at the host system; detecting that a redirection event has occurred at the host system and generating a redirection trigger; receiving messages directed to a first address at the host system from a plurality of message senders; in response to the redirection trigger, continuously redirecting the messages from the host system to the mobile data communication device; receiving the messages at the mobile data communication device; generating reply messages at the mobile data communication device to be sent to the plurality of message senders and transmitting the reply messages to the host system; receiving the reply messages at the host system and configuring address information of the reply messages such that the reply messages use the first address associated with the host system as the originating address, wherein messages generated at either the host system or the mobile data communication device share the first address; and transmitting the reply messages from the host system to the plurality of message senders." *Id.*, claim 1. "A message redirection method operating at a host system, comprising the steps of: A method of redirecting messages between a host system and a mobile data communication device, comprising the steps of: configuring one or more redirection events at the host system; detecting that a redirection event has occurred at the host system and generating a redirection trigger; receiving messages directed to a first address at the host system from a plurality of message senders; in response to the redirection trigger, continuously redirecting the messages from the host system to the mobile data communication device; receiving the messages at the mobile data communication device; generating reply messages at the mobile data communication device to be sent to the plurality of message senders and transmitting the reply messages to the host system; receiving the reply messages at the host system and configuring address information of the reply messages such that the reply messages use the first address associated with the host system as the originating address, wherein messages generated at either the host system or the mobile data communication device share the first address; and transmitting the reply messages from the host system to the plurality of message senders." *Id.*, claim 22.

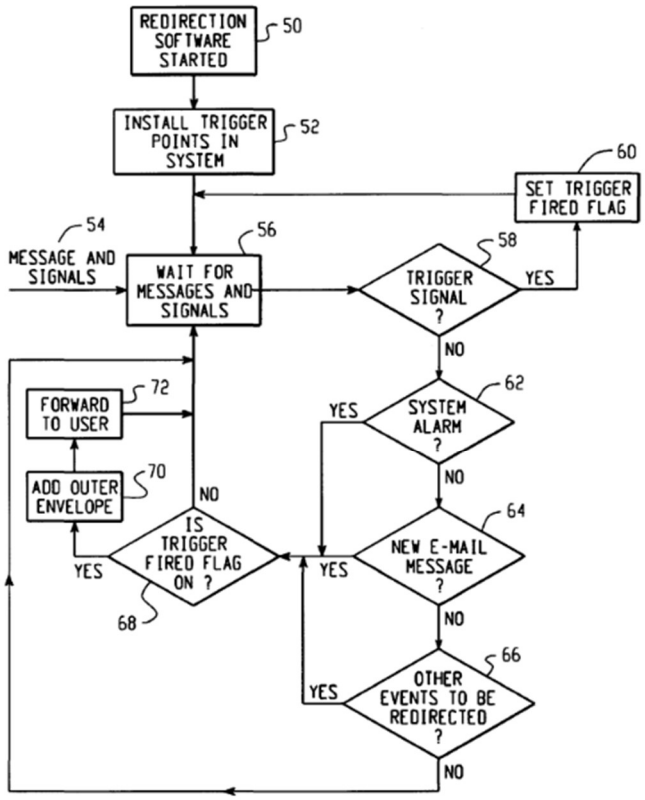
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 82 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 1 *Id.*, Fig. 1.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 83 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 3 *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 84 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 4 *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 85 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Houghton Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 86 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Houghton, as evidenced by the example citations below, discloses the logic to determine when one of a plurality of message delivery triggers for the given one of the wireless end-user devices has occurred (e.g., trigger events that can be local or remote), wherein for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a message delivery trigger (e.g., an alarm, notification, or measurement result received to the push server from another device or system), and for at least one of the message delivery triggers, the trigger is an occurrence of an asynchronous event with time-critical messaging needs (e.g., adjusting message delivery based on whether the message to be delivered is time-critical or non-time-critical). *See, e.g.*, "IP push command messages may be triggered either by events which the server is aware of, or events which a similar mobile application on another mobile terminal is aware of and which that push client makes the server aware of, the server in turn initiating a push command event. Such client-side events of which the server is informed for the possibility of delivering new web services include the making, receiving, and termination of telephone calls, the transition from presence-to-absence or absence-to-presence of other devices in a geographical area, metropolitan area, local area or personal area wireless communications network, mobile terminal creation of photographs video audio or other media, mobile terminal presence of files matching a given filtering criteria, changes in internal state of electronic equipment, taking of measurements, processing of such measurements to see if they match certain criteria, video game actions or events, application calculation events and messaging actions. In some cases a separate server such as a telecommunication service server, gaming environment server, and original equipment manufacturer device or business information system may notify the server directly of events or messages to be sent using an external application programming interface (API). Such a device connected by external API may also receive information including the results and responses of messages which the user has replied to by terminal software or web page interaction." *Id*., p. 14. "Such a selective use of this high priority, low latency communication channel from terminal 404 to server 401 gives the server the opportunity to in return push to the client 405 a service triggered by the event begin communicated by the client 405 as will be described in more detail below. The sending of such a return message is frequently time critical and the fastest available combination of techniques will be used." *Id*., pp.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 87 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	20-21. "The push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server. The trigger may, among others, include an alarm, notification, or measurement result received to the push server 401 from another device or system." *Id*., p. 21. "Figure 7 illustrates an example of an 'COMMAND PUSH' procedure wherein the push server 701 is triggered by a trigger event as for example from an application server 702 to push an application command message to the push client 704 and thereby initiate a mobile terminal client trigger event in a mobile application 705 from a plurality of such applications, program modules and user interface features on the terminal 705." *Id*., pp. 21-22. "Further, the push server 501 on becoming aware of such a situation may inform (arrow D) the application server 502 of the availability of a more desirable network connection (C plus A) to the client and on behalf of the application server 502 or independently decide to push additional commands and data to the wireless terminal using this connection. Further, the push client in the broadband access point has the additional capability to route such information to the wireless terminal. The result benefits customers using the wireless terminal in that data is managed and updated automatically for their mobile applications in the most cost efficient manner for non-time-critical capabilities such as lazy application code update and lazy application data update. While such alternative routing may also be achieved through normal IP routing and path cost without the presence in the broadband access point 509 of push-aware client software, the push server 501 and application server 502 would in this known-in-the-art implementation not be aware of when the low cost alternative route was available and thus able to at such time trigger the push of lazy application code and data installation and updates." *Id*., p. 24. "If a push message is received, the client 405 executes the corresponding function, such as launch or pass commands and data a mobile application (step 632). If required, the client 405 may also send a message to the server 401 (step 633), such as a command execution acknowledgement or result." *Id*., p. 26. "Referring to Figure 60, after the login and connection setup, the push client 405 may monitor the local

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 88 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	resources for a state change (a trigger event) in step 641. If a state change of other event is detected in step 642, the push client 405 may send a message to the server 401 (step 643)." *Id*. "A method as claimed in any one of claims 1-21, wherein said trigger events include one or more of: the making, receiving, and termination of telephone calls, the transition from presence-to-absence or absence-to presence of other devices in a physical contact, RF, near field, infrared, personal area, local area, metropolitan area or geographical area wireless communications network, an alarm, notification, videogame action or event, remote application user or application server event, remote user action or inaction, electronic or computing equipment change of state or measurement result." *Id*., claim 22. **Combination with Munson** Munson and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Munson discloses a method of "pushing contents to client devices," including "group pushes" in which content is buffered and sent to multiple devices simultaneously, "serializ[ing]" content such that a series of messages are delivered to a particular device simultaneously. Munson, Abstract, ¶37, Fig. 4. A POSITA would have been motivated to combine Munson with Blackberry Push because a POSITA would have desired to support diverse client devices and network environments, as well as to leverage an existing, proven technology for wireless content delivery, e.g., Blackberry Push. Munson similarly teaches messages can be delivered from memory "according to a schedule of time or event" such as coordinating pushes "for a system maintenance purpose during off-peak hours." Munson, ¶¶40, 44. Thus, Munson provides additional details about how to implement a store-and-forward system by providing the computer memory at a server, and the computer programming to deliver messages from memory (including upon occurrence of certain events). *Id*., ¶¶36-38, 44, Fig. 4. Therefore, a POSITA would have been motivated to incorporate computer memory and programming into Blackberry Push to store and deliver messages from memory, consistent

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 89 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	with Munson, to improve system flexibility and reliability. A POSITA would have had a reasonable expectation of success in combining Blackberry Push and Munson because Munson provides implementation details such push message system. *See, e.g.*, Munson, ¶¶36-38. A POSITA would have been motivated and had a reasonable expectation of success in incorporating conditions Munson teaches providing additional flexibility in how messages are delivered in PAP, used by Blackberry Push. This combination of Blackberry Push and Munson would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Munson, as the communication protocol because Blackberry Push was a well-known, industry-leading system. Munson, ¶¶35-36. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Munson push messaging system without requiring inventive skill. Munson, as evidenced by the example citations below, discloses the logic to determine when one of a plurality of message delivery triggers for the given one of the wireless end-user devices has occurred, wherein for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a message delivery trigger, and for at least one of the message delivery triggers, the trigger is an occurrence of an asynchronous event with time-critical messaging needs. *See, e.g.*, "The present invention guarantees reliable and asynchronous content push (i.e., pushing a content) to clients on diverse wireless networks by offering application services providers a single technology-independent interface. Applications are offered a 'send and forget' service. 'Send and forget' service means that after an application service provider delivers a content to the Content Push Service (CPS), the application service provider does not have to further manage delivering the content. The CPS will deal with all the details of getting the content to a client device. An application service provider is notified asynchronously, if a push (i.e., pushing a content) encounters a problem. This simplifies application programming because application service providers do not need to be concerned with delays in the sending (i.e., pushing a content) process."

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 90 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Munson, ¶40. "In one embodiment, a content can be pushed according to a schedule of time or event. For example, when used with a group push, an application service provider can use pushes (i.e., pushing contents) for a system maintenance purpose during off-peak hours to take advantage of lower rates." *Id*., ¶44. "The system according to claim 1, wherein the contents are pushed to client devices according to a schedule of time or event." *Id*., claim 8. "The method according to claim 13, further comprising: pushing the contents to the each member of the group according to a schedule of time or event." *Id*., claim 17. **Combination with TS.23-140** TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 91 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS.23-140, as evidenced by the citations below, discloses the logic to determine when one of a plurality of message delivery triggers for the given one of the wireless end-user devices has occurred, wherein for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a message delivery trigger, and for at least one of the message delivery triggers, the trigger is an occurrence of an asynchronous event with time-critical messaging needs. *See, e.g.*, "Upon reception of an MM from an originator MMS User Agent the originator MMS Relay/Server . . . shall retain the MM until the earliest desired time of delivery, if the optional feature of earliest time of delivery is supported by the originator MMS Relay/Server. If this feature is not supported then the MM is immediately routed forward" *Id*., p. 27. "Upon reception of an MM from an originator MMS User Agent the originator MMS Relay/Server . . . shall route the MM towards the MM recipients" *Id*. "Upon reception of an MM the recipient MMS Relay/Server . . . shall generate a notification to the recipient MMS User Agent." *Id*., p. 28. "With the MM notification the recipient MMS User Agent shall receive a message reference that can be used for retrieving the MM from the recipient MMS Relay/Server. The message reference that is conveyed in a notification shall at least be valid throughout the message expiry period, till the successful retrieval of

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 92 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the MM or until the MM was rejected. With the MM notification the recipient MMS User Agent may receive additional information on the MM." *Id*. "The MMS Relay/Server may include an indication in the MM notification recommending manual retrieval mode. This recommendation may be based on user settings in the User Profile." *Id*. "In a response to the notification the MMS User Agent shall be able to . . . retrieve the MM, either immediately or at a later time (i.e. defer the MM retrieval), either manually or automatically, as possibly determined by the operator configuration and user profile." *Id*. "The retrieval mode employed by the recipient MMS User Agent for a particular MM may be based either on the user settings in the terminal or on the recommendation carried in the MM notification. The recipient MMS User Agent may follow this recommendation to retrieve the MM, through manual retrieval. For any MM for which the retrieval has been deferred, the MMS User Agent may request deletion on the MMS Relay/Server instead of MM retrieval." *Id*., p. 29. "The recipient MMS User Agent shall be able to request retrieval of an MM from the recipient MMS Relay/Server based on the Message Reference received in a notification. If MMBoxes are supported, the MMS User Agent shall be able to request retrieval of an MM from the user's MMBox, based on a Message Reference received from a previous MMBox operation." *Id*. "Within a retrieval request the recipient MMS User Agent may indicate a size restriction of the returned MM (i.e., maximum size) that the MMS Relay/Server is to use in processing the retrieval request." *Id*. "Upon retrieval request the recipient MMS Relay/Server . . . should not deliver the MM (or any adaptation of the MM) to the recipient MMS User Agent unless the size restriction set by the MMS User Agent is met." *Id*. "Upon retrieval request the recipient MMS Relay/Server . . . shall store messages in the network until the recipient MMS User Agent becomes reachable (e.g. user moves back into coverage, switches MMS User Agent on) or until the MM expires" *Id*.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 93 of 422

<table>
<tr><th>ASSERTED CLAIMS</th><th>PRIOR ART REFERENCE:
Blackberry Push</th></tr>
<tr><td></td><td>“This part of the MMS service covers the notification about MM from the recipient MMS Relay/Server to the corresponding recipient MMS User Agent and involving abstract messages are outlined in the table below from type, and direction points of view.” Id., p. 66.

Table 7: abstract messages for notification of MM in MMS

<table><tr><th>Abstract message</th><th>Type</th><th>Direction</th></tr><tr><td>MM1_notification.REQ</td><td>Request</td><td>MMS Relay/Server -> MMS UA</td></tr><tr><td>MM1_notification.RES</td><td>Response</td><td>MMS UA -> MMS Relay/Server</td></tr></table>

Id., Table 7.

Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by TS-23.140, it would have been obvious in view of the disclosures in TS-23.140, or in combination with the knowledge of a person of ordinary skill in the art. Because the MMS Relay/Server must monitor when these events occur or conditions are satisfied, in my opinion, a POSITA would have understood or found it obvious that the MMS Relay/Server includes logic configured to determine when they have occurred for a given UE/mobile device. Also, a POSITA would have understood or found it obvious that at least some of the triggers occur upon an asynchronous event with time-critical messaging needs. For example, And the manual retrieval is a trigger occurring upon an asynchronous event with time-critical messaging needs because the user must manually cause the MMS User Agent to request delivery. TS-23.140, 20. The ’320 Patent itself discloses that a manual “user request” for delivery constitutes such a message-delivery trigger. ’320 Patent, 38:50-63.

Additionally, a POSITA would have understood that the disclosures above constitute triggers that are consistent with the ’320 Patent’s description of triggers. For instance, the ’320 Patent explains: “the service control server link 1638 can determine when to transmit based potentially on several parameters, such as one or more of: periodic timer trigger, waiting until a certain amount of service usage or traffic usage has</td></tr>
</table>

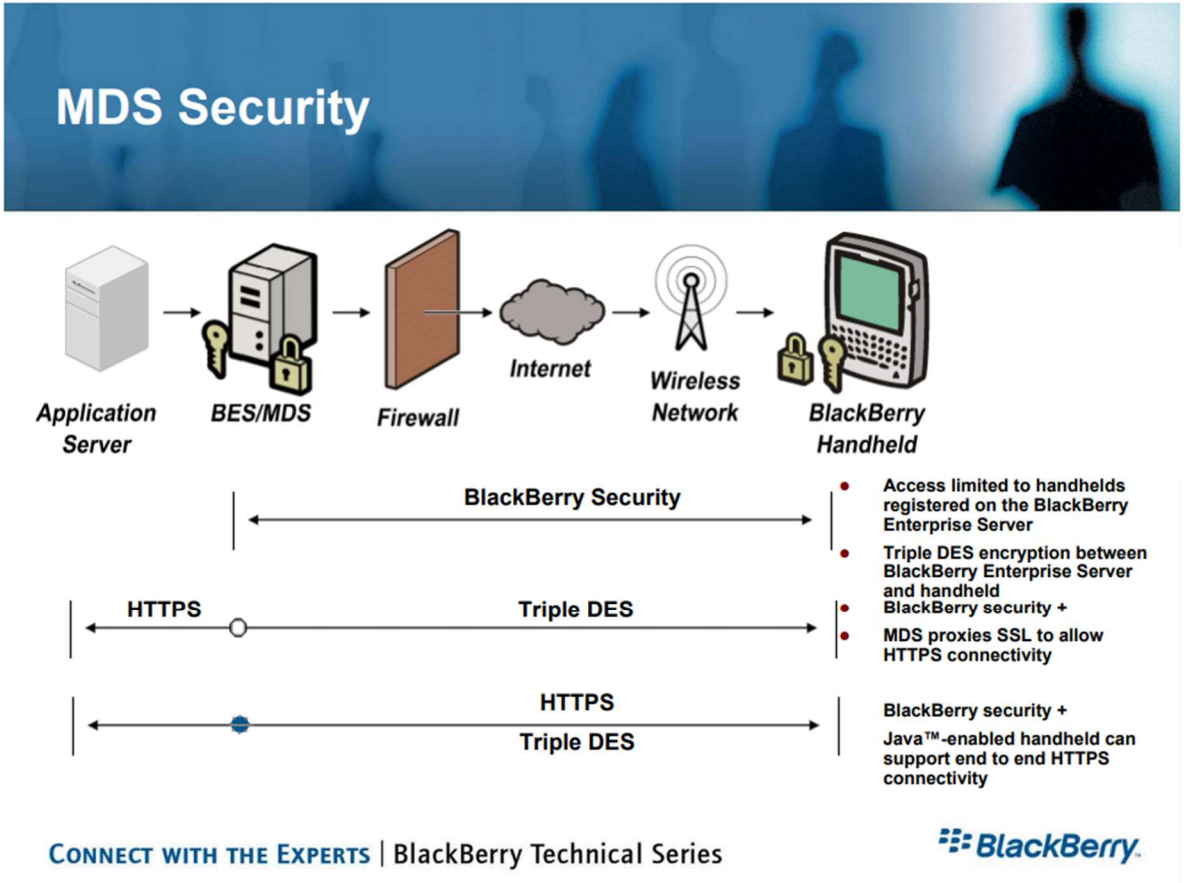
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 94 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	occurred, responding to a service agent message, responding to a service agent request, initiated by one or more servers, initiated by a verification error condition, and/or initiated by some other error condition." '320 Patent, 70:6-13. The '320 Patent further discloses: "In addition, the service control server link 1638 can provide for an efficient communication link based on various embodiments related to the timing of transmissions over the service control link, as similarly discussed above with respect to the service control device link 1691 description. For example, the timing functions, such as asynchronous messages or polling for messages, constant frequency transmission, transmission based on how much service usage or data traffic usage has taken place, transmission in response to device side control link message, service verification error events, other error events, and/or other message transmission trigger criteria can be determined, controlled and/or initiated by either the device side or the network side depending on the embodiment." *Id.*, 70:16-30. The '320 Patent makes clear that any component on the network (such as the server) and the (message-receiving) device can initiate the trigger. *Id.* ("For example, the timing functions, such as asynchronous messages or polling for messages, constant frequency transmission, transmission based on how much service usage or data traffic usage has taken place, transmission in response to device side control link message, service verification error events, other error events, and/or other message transmission trigger criteria can be determined, controlled and/or initiated by either the device side or the network side depending on the embodiment."). The '320 Patent discloses additional triggers for "the service control device link 1691," which are similar to the list of transmission/message delivery triggers described above. More specifically, the '320 specification states: "the service control device link 1691 determines when to transmit based potentially on several parameters including, for example, one or more of the following parameters: periodic timer trigger, waiting until a certain amount of service usage or traffic usage has occurred, responding to a service controller message, responding to a service controller request, initiated by one or more agents, initiated by a verification error condition, initiated by some other error or status condition." *Id.*, 38:30-38. Other identified examples include a "transmission trigger [that] is based on waiting for an amount of service usage, and also including a minimum transmission rate that triggers a transmission according to one or more

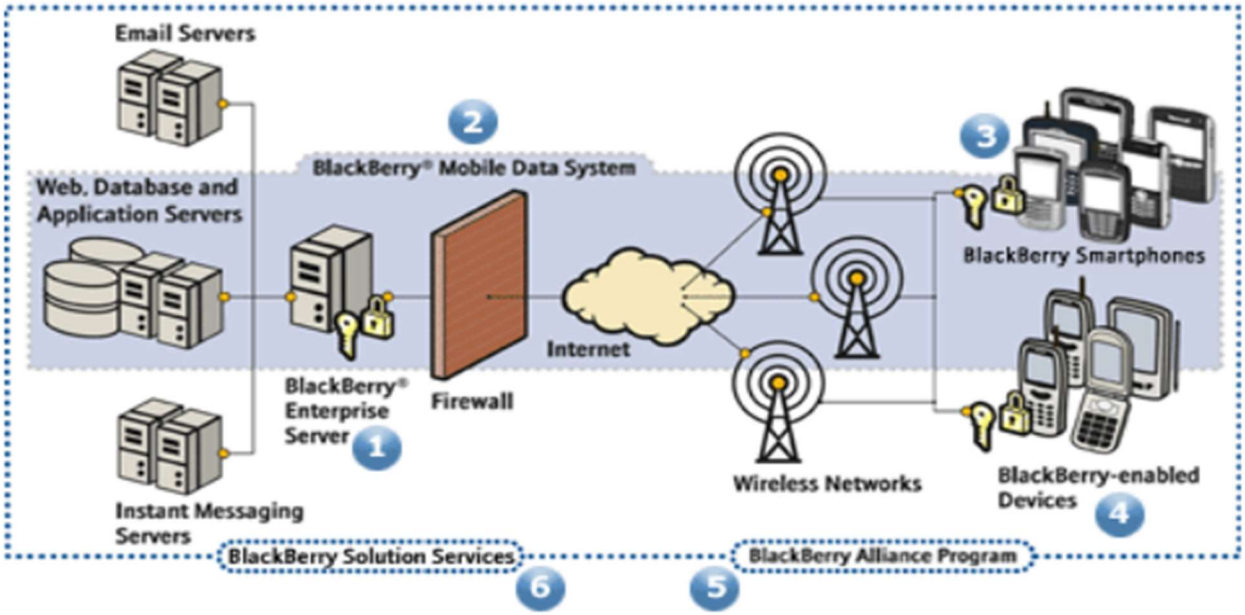
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 95 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	of the following parameters: a maximum time between transmissions clock to keep the service processor 115 in communication with the service controller 122 when little or no service usage is occurring, a polling request of some kind from the service controller 122, a response to a service controller heartbeat, a transmission generated by a service verification error event, or a transmission generated by some other asynchronous event with time critical service processor 115 (or service controller 122) messaging needs, such as a transaction or service billing event or a user request." *Id.*, 38:50-63. The '320 specification provides additional details of the "service usage" or "service activity trigger," noting that it can be "based on some other measure than traffic usage, such as a number of messages transacted, one or more billing events, number of files downloaded, number of applications run or time that an application has been running, usage of one or more specified applications, GPS coordinate changes, roaming event, an event related to another network connection to the device and/or other service related measures." *Id.*, 39:11-19.
1[g]: upon determining that one of the message delivery triggers has occurred for the given one of the wireless end-user devices, the logic further to supply one or more messages comprising the buffered content for the given one of the wireless end-user devices to the transport services stack for delivery on the secure message link maintained between the transport services stack and a device link agent on the given one of the wireless end-user devices; and	Blackberry Push, as evidenced by the example citations below, discloses, upon determining that one of the message delivery triggers has occurred for the given one of the wireless end-user devices, the logic further to supply one or more messages comprising the buffered content for the given one of the wireless end-user devices to the transport services stack for delivery on the secure message link maintained between the transport services stack and a device link agent on the given one of the wireless end-user devices. Blackberry Push supplies the messages/buffered content to the secure BES transport for delivery from the server to device link agents via secure channel. • MDS Technical Series, p. 12:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 96 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 97 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• BES Architecture Overview, p. 1:  • MDS Technical Series, p. 9: MDS Architecture • HTTP connection requests utilize the existing secure 3DES (or AES) connection over port 3101 • MDS communicates with web servers using protocol requested by handheld • MDS supports common protocols and mark up languages – HTTP, HTTPS, TCP, XML, HTML • Handheld and MDS communicate using IP Proxy Protocol – IPPP service book required on the handheld

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 98 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS Studio 4.1. pp. 14-15: Delivery Modes Messages that pass between BlackBerry MDS Applications and the BlackBerry MDS Services are categorized into two delivery modes: best-effort and standard. The default delivery mode is standard. Best-effort The best-effort delivery mode is a fire-and-forget mode. Messages might not be delivered, and they might be delivered in a different order than they were sent from the application to the BlackBerry MDS Services or from the BlackBerry MDS Services to the application. Standard Messages sent from the application are designed to reach the BlackBerry device, even if the BlackBerry device spends time outside of a wireless coverage area or if the BlackBerry MDS Services stop. The sending side transmits the message when a connection becomes available. The application does not confirm receipt of messages, so the application might not process a message even after it reaches the BlackBerry device. Similarly, the BlackBerry MDS Services might not process a message from the application even when the message reaches the BlackBerry MDS Services server. Standard mode messages are designed to be delivered and processed in the order in which they were generated. Handling messages when a BlackBerry device is outside of a wireless coverage area When a BlackBerry device is outside of a wireless coverage area, outbound standard and reliable application messages remain queued on the BlackBerry device until the BlackBerry device returns to a wireless coverage area, at which time the messages are transmitted. Best-effort messages are dropped when a BlackBerry device is outside of a wireless coverage area. Message queue sizes BlackBerry MDS Applications have restricted inbound and outbound message queue sizes. These restrictions are imposed by policies sent to the BlackBerry MDS Runtime software on the BlackBerry devices. The system administrator of the BlackBerry MDS Services can push new policies to the BlackBerry MDS Runtime, which might change the limitations of an application's inbound and outbound queue sizes.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 99 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Support for push events The push-based architecture that is used for BlackBerry email is available for custom BlackBerry MDS Applications. Using this push-based architecture, server-side applications can push new data, alerts, and notifications to users proactively. • Mens (2006), p. 12: HTTP_X_RIM_PUSH_ID This header contains the value specified in the X-RIM-Push-ID header upon submitting the push. In my programs, it is a unique ID which I use to correlate the push with the subsequent (and asynchronous) notification. HTTP_X_RIM_PUSH_DESTINATION This header contains the email address or the PIN used as destination address during the push. The notifiy.cgi program uses the push identifier supplied in the X-RIM- PUSH-ID header to correlate the notification with the original push request. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.
1[h]: the device link agents on the respective wireless end-user devices, each of the device link agents configured to	Blackberry Push, as evidenced by the example citations below, discloses device link agents on the respective wireless end-user devices. BlackBerry Browser, MDS Runtime, and Blackberry Java Virtual Machine are device link agents that handle push notifications over the Blackberry Push transport. *See, e.g.*,

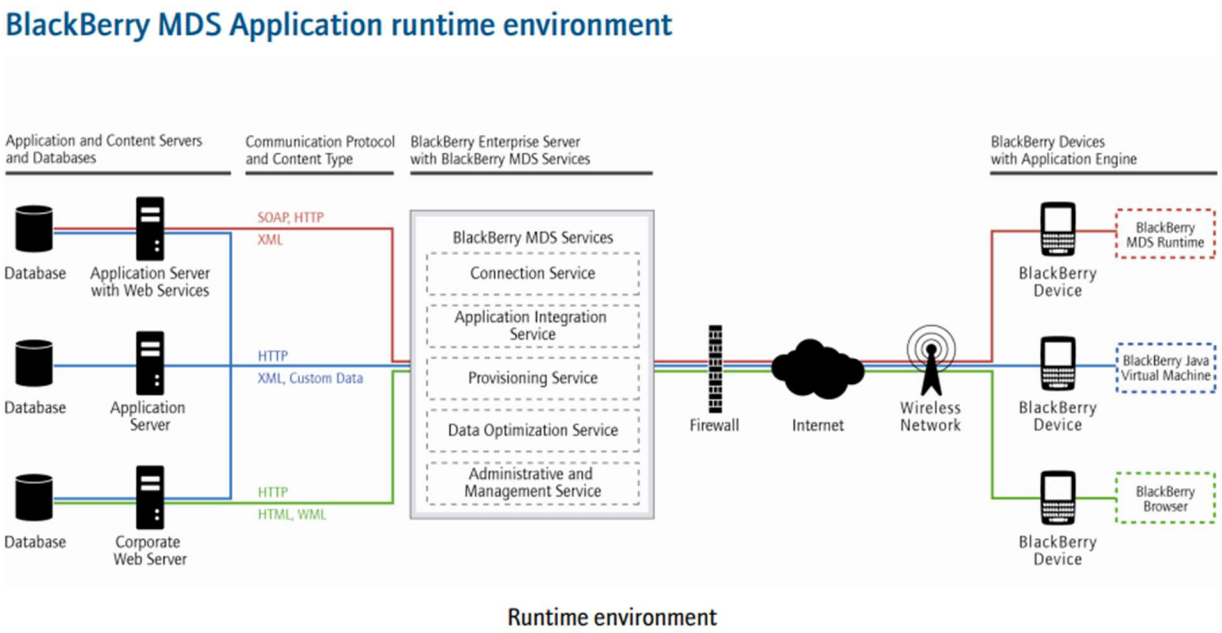
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 100 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS Technical Series, p. 39: The BlackBerry Browser • Standard "web page-based" user interface and navigation • Able to display data formatted in standard mark-up – HTML, cHTML, xHTML and WML • BlackBerry Browser supports PUSH of content via MDS • MDS Studio 4.1, p. 8: BlackBerry MDS Runtime The BlackBerry MDS Runtime™ is the container-based execution environment for BlackBerry MDS Applications. The BlackBerry MDS Runtime contains a control center application that enables users to search for and manage BlackBerry MDS Applications and view application information. The BlackBerry MDS Runtime is designed to provide the underlying services that are used by BlackBerry MDS Applications, including the user interface, data storage, and client-server communication services. • MDS Studio 4.1, p. 13: Messaging model BlackBerry MDS Applications are designed to store and process data locally on the BlackBerry device, without the need for a continuously available network connection. The communication model used to send and receive application messages is fully asynchronous. This means that if an outbound message is generated at a time when the BlackBerry device is outside of a wireless coverage area (for example, as a result of user input), the message is placed into an outbound queue until the wireless connection becomes available. When the message is placed into this queue, the application continues functioning, regardless of whether the message has been sent. Messages are transmitted from the queue in the background when the network connection becomes available.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 101 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Similarly, inbound messages can be queued by the BlackBerry MDS Services and transmitted to the BlackBerry device only when the wireless connection becomes available. BlackBerry MDS Applications are designed to process inbound messages as asynchronous events. This asynchronous communication model is designed for applications that are inherently tolerant to sudden or unpredictable loss of connection. This feature, combined with the ability to store and process data locally, enables BlackBerry MDS Applications to be designed in such a way that application functionality and enterprise data are available, even when wireless connections are not available. • MDS 4.1 Overview, p. 5: BlackBerry Mobile Data System device software The BlackBerry MDS uses the following device software to run BlackBerry MDS Applications on the device: • The BlackBerry MDS Runtime™ provides runtime services for applications that are developed with BlackBerry MDS Studio. Users must download the BlackBerry MDS Runtime to use BlackBerry MDS Studio applications on their BlackBerry devices. To download the BlackBerry MDS Runtime, visit http://www.blackberry.com. • The BlackBerry Java Virtual Machine™ (BlackBerry JVM) provides a secure environment on BlackBerry devices for running custom Java™ applications that are created with the BlackBerry Java Development Environment (JDE). • The BlackBerry Browser enables access to web-based applications and supports Internet standards, such as HTML, Wireless Markup Language (WML), Common Gateway Interface (CGI), Microsoft® Active Server Pages (ASP), and JavaServer Pages™ (JSP). The BlackBerry Browser is included with the BlackBerry Enterprise Solution and runs on top of the BlackBerry JVM. • MDS 4.1 Overview, p. 4:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 102 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.
1[i]: maintain the respective secure Internet data message link over a wireless network	Blackberry Push, as evidenced by the example citations below, discloses maintaining the respective secure Internet data message link over a wireless network to the link interface.

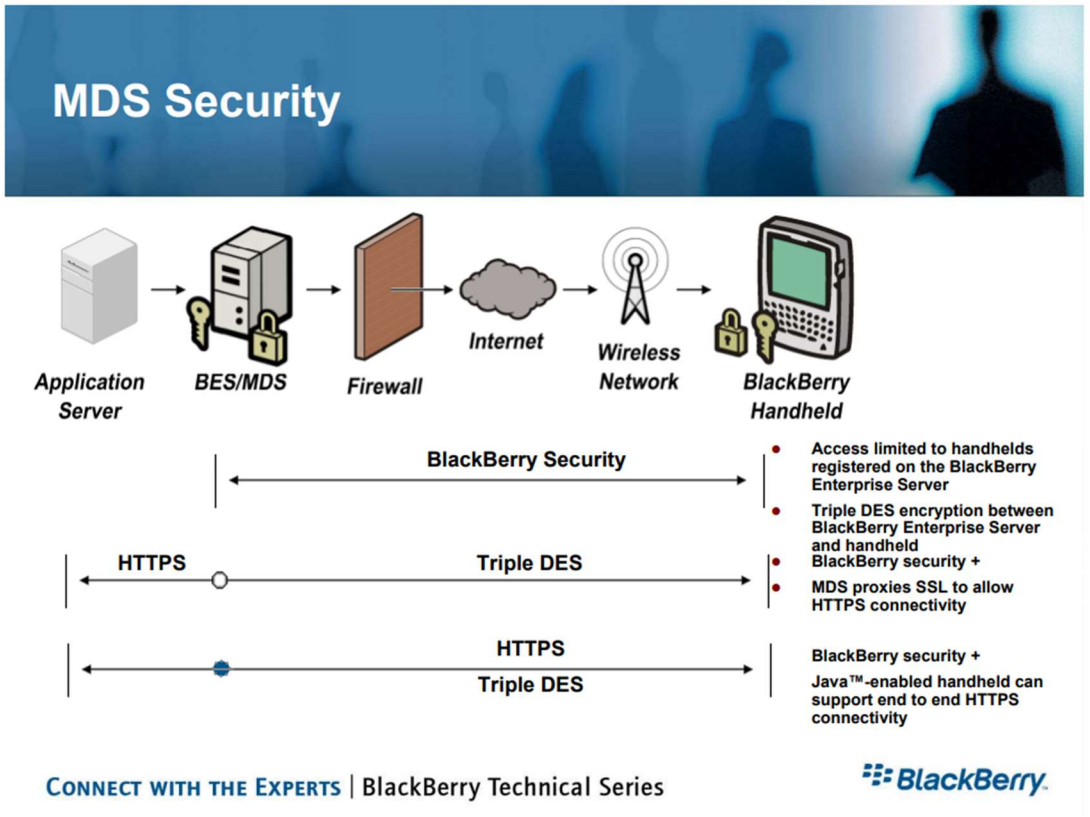
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 103 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
to the link interface,	BlackBerry device link agents maintain the secure link to Blackberry Push. *See, e.g.*, • MDS 4.1 Overview, p. 15: Creating BlackBerry MDS Java applications BlackBerry devices provide a Java ME wireless Java environment that is designed to enable developers to create sophisticated custom applications within the standard client/server model. Application developers can create custom Java applications with sophisticated user interfaces for data entry and searching, multithreading, internationalization, network communication, and local data storage. Applications can communicate with networks using standard TCP and HTTP connections, regardless of the underlying wireless network. The BlackBerry MDS Connection Service typically sets up the TCP connection on behalf of the application. Application developers can also create custom applications that integrate tightly with core BlackBerry device programs, such as the messages list, personal information management (PIM) programs, phone, and browser, for an essentially seamless user experience. • MDS 4.1 Overview, p. 16: Network transport Depending on the environment and the application requirements, BlackBerry devices can connect to both corporate intranets and the Internet through various gateways, including the BlackBerry Enterprise Server hosted by the enterprise or a gateway hosted by the wireless service provider. BlackBerry devices are designed to support a wide range of network transport protocols, including: • HTTP and HTTPS • TCP, SSL, and TLS sockets

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 104 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• email messages • User Datagram Protocol (UDP) datagrams • Short Message Service (SMS) • Multimedia Messaging Service (MMS) is supported on some BlackBerry devices • MDS Technical Series, p. 9: MDS Architecture • HTTP connection requests utilize the existing secure 3DES (or AES) connection over port 3101 • MDS communicates with web servers using protocol requested by handheld • MDS supports common protocols and mark up languages – HTTP, HTTPS, TCP, XML, HTML • Handheld and MDS communicate using IP Proxy Protocol – IPPP service book required on the handheld • MDS Studio 4.1, p. 8: BlackBerry MDS Runtime The BlackBerry MDS Runtime™ is the container-based execution environment for BlackBerry MDS Applications. The BlackBerry MDS Runtime contains a control center application that enables users to search for and manage BlackBerry MDS Applications and view application information. The BlackBerry MDS Runtime is designed to provide the underlying services that are used by BlackBerry MDS Applications, including the user interface, data storage, and client-server communication services.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 105 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS Technical Series, p. 12: 

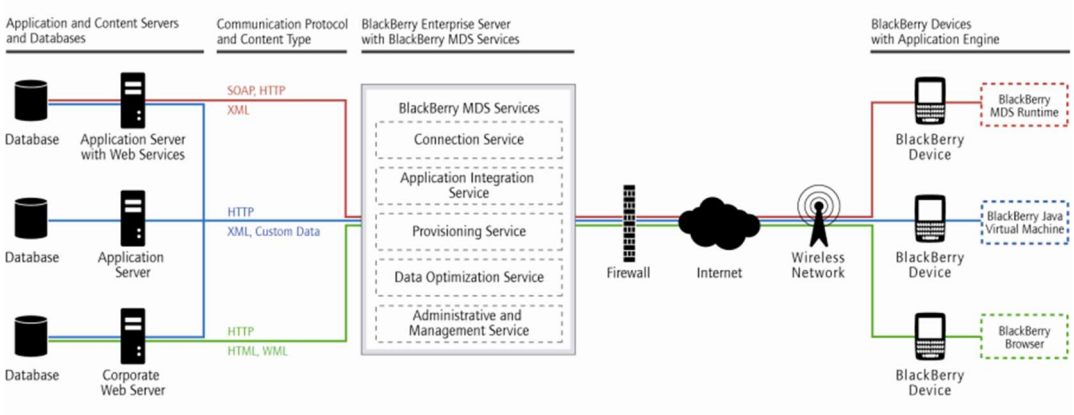
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 106 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.
1[j]: receive secure Internet data messages from the network server system over the respective secure Internet data message link, including messages collected from multiple ones of the network elements and messages corresponding to multiple ones of the software components authorized to receive messages via the device link agent on that respective device, wherein at least a first subset of the secure Internet data messages contain both a unique identifier for a corresponding one of the software agents and data to be consumed by that software component, the data supplied from a respective network element corresponding to that software component, and	Blackberry Push, as evidenced by the example citations below, discloses the ability to receive secure Internet data messages from the network server system over the respective secure Internet data message link, including messages collected from multiple ones of the network elements and messages corresponding to multiple ones of the software components authorized to receive messages via the device link agent on that respective device, wherein at least a first subset of the secure Internet data messages contain both a unique identifier for a corresponding one of the software agents and data to be consumed by that software component, the data supplied from a respective network element corresponding to that software component. The device link agents receive secure messages over the BES channel (3DES/AES) from MDS, which aggregates communications from multiple enterprise servers and can be sent to multiple applications. *See, e.g.*, • MDS Studio 4.1, p. 9: BlackBerry MDS Services The BlackBerry MDS Services are designed to provide connectivity between BlackBerry MDS Applications on BlackBerry devices and enterprise applications. Connection Service The Connection Service enables a system administrator to connect the BlackBerry MDS Services to the Mobile Data Service feature of the BlackBerry Enterprise Server®. The Connection Service is designed to accept and respond to push requests from server-side push applications when the application server is behind the corporate firewall. This service is designed to provide a link to standard servers on the corporate intranet or Internet using standard Internet protocol, such as HTTP or TCP/IP, and encrypt content using the same

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 107 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	encryption standard that is used to encrypt BlackBerry messages and other BlackBerry data. Application Integration Service The Application Integration Service is designed to support web services and other standard mechanisms for integrating wireless applications with existing enterprise applications and systems. This service is designed to manage the transmission of application data messages between BlackBerry MDS Applications and data sources. Provisioning Service The Provisioning Service is designed to control which BlackBerry MDS Applications users can download to their BlackBerry devices, support application discovery from a BlackBerry device, and manage wireless transmission and on BlackBerry devices. Data Optimization Service The Data Optimization Service is designed to convert existing server-side content and data to enable wireless transmission and use on BlackBerry devices. Administrative and Management Service The Administrative and Management Service is designed to centralize the application lifecycle management, including centralized push installation, upgrade, and removal of applications from BlackBerry devices.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 108 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS 4.1 Overview, p. 4:  Runtime environment • MDS 4.1 Overview, p. 13: Pushing content to users BlackBerry MDS Studio applications use WS-Eventing to send information proactively to the device. For push events, an outbound message component is designed to be bound at the time of design to the subscription web service that is associated with the event. An inbound message is also designed to be bound to the web service event. At runtime, the application can subscribe to receive the web service event (for example, through BlackBerry device user input that results in the outbound subscription message being sent). Web service events cause inbound messages to be sent to the application. BlackBerry MDS Studio applications can notify BlackBerry device users of inbound messages in several ways, including a flashing LED, a tune, or a vibration.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 109 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS 4.1 Overview, p. 17: Pushing content to users BlackBerry MDS Java applications can use HTTP POST or PAP requests to send information proactively to the BlackBerry device. The request headers include information about the content to be pushed, the destination BlackBerry device or user, and the BlackBerry device application for which the data is intended (such as the BlackBerry Browser or a custom application). The request body contains the content. When the connection service on the BlackBerry Enterprise Server receives the HTTP POST message, it initiates the connection to the BlackBerry devices. • Mens (2006), p. 9: The push program When a document is ready to be sent out, the pushout program is invoked with the tag identifying the document to send. Pushout scans the database looking for subscribers to that document tag-type. For each subscriber found, pushout invokes an HTTP request to the MDS to push the document to the user's devices. For each of the invocations, pushout constructs a list of headers containing information which is submitted to the MDS. These headers instruct the MDS to handle the push in a specific manner: Content-Location This header is passed directly to the BlackBerry browser and not otherwise used. If the user on the device looks at the page address, the value of this header is displayed. Similarly, if the browser is forced to refresh the page, it will use the value of the Content-Location header as the URL from which to refresh. If, for some reason, you want to hide the URL, you can set this to file://some/place, thereby effectively hiding the information and also effectively disabling a user from refreshing the page's content. • Java Development 4.0, p. 121: Client/server push applications: Data is sent to a custom Java application on the handheld. Client/server

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 110 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	push applications consist of a custom client application for the handheld and a server-side application that pushes content to it. This approach provides more control over the type of content that you can send out and how this data is processed and displayed on the handheld compared to browser push applications. Content and headers identify target applications/channels (unique identifiers) and carry payload for consumption by the addressed software component. *See, e.g.*, • MDS Technical Series, p. 45: Push Types • BlackBerry-defined push types: – "Channel Push" • Push an icon to the Home screen along with a web page • The icon acts as a permanent 'channel' to the referenced URL – "Message Push" • Push a message to the user's inbox that automatically launches the browser to a specified page (cached or not) – "Cache Push" • Push a page to the browser's cache, but do not notify the user • MDS Studio 4.1, p. 11: Data Data components are designed to represent application data structures. Data components are passed between mapped screen and message components and are instantiated from user actions or inbound messages from data sources. Data components store transient and persistent application data on the BlackBerry device. Transient data components are used when data is only needed temporarily by the application; persistent data components are used when data is stored between invocations of the application (including when the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 111 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	BlackBerry device is turned off). You define the structure of a data component from a set of available data types: •boolean •number •date •strings (including URL, password, phone number, email, and currency) •enumerated types •data component types •built-in standard types • Mens (2006), pp. 9-10: Content-Type This header specifies the MIME type of the document being sent to the browser. X-RIM-Push-Title The title of the browser message. For the messages we are pushing to the browser, the value of this header will be displayed in the message list. For push messages which install a new icon on the device home page, the icon will be named according to this value X-RIM-Push-Type The type of push message. Visit Pushing Web Content to BlackBerry Wireless Devices for a description of the four different kinds of browser push supported by the MDS X-RIM-Push-Channel-ID Each channel gets an identifier which can be used to delete the channel from a device. I will not be using this feature, but I specify a channel-id none the less X-RIM-Push-NotifyURL When the MDS pushes out a browser message, it will invoke this URL to post a notification regarding success or failure in pushing the message. This is optional, but I'm using the notification URL in order to determine whether a device received my page or not X-RIM-Push-ID

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 112 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	When using X-RIM-Push-NotifyURL, the MDS will pass the value of the this header to the notification URL. That is the only way to correlate a push with a subsequent asynchronous notification. If the programmer doesn't specify this value, the MDS constructs its own. X-Rim-Push-Use-Coverage This appears to be new with MDS 4.1; it ought to indicate via the notification URL whether the device is in coverage or not. I may have misunderstood this, but I cannot get the MDS to return anything but true. The pushout program is otherwise quite straightforward. To the extent this limitation is not disclosed by Blackberry Push, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, TS-23.140 **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 113 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the device link agents (e.g., the software program instructions) receiving secure Internet data messages from the network server system (e.g., secure messages) from the over the respective secure Internet data message link (e.g., secure connection/wireless link through the Internet), including messages collected from multiple ones of the network elements (e.g., other systems connected to the LAN) and messages corresponding to multiple ones of the software components (e.g., software) authorized to receive messages via the device link agent on that respective device wherein at least a first subset of the secure Internet data messages contain both a unique identifier for a corresponding one of the software agents (e.g., addressing information including a destination address) and data to be consumed by that software component (e.g., the message redirected), the data supplied from a respective network element corresponding to that software component. *See, e.g.*, "For example, if a user is traveling and receives a message with an attachment that the user's mobile device can process or display, the user may from a mobile communications device send a command message to the host system indicating that that attachment is to be sent to a fax machine at a hotel where the user will be spending the evening. This enables the user to receive important E-mail attachments as long as the host system is provided with sufficient information about the destination where the attachment is to be forwarded." Lazaridis, 3:56-65.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 114 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The repackaging also permits both messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed." *Id.*, 4:16-4:19. "Message C in FIG. 1 represents an external message from a sender that is not directly connected to LAN 14, such as the user's mobile data communication device 24, some other user's mobile device (not shown), or any user connected to the Internet 18. Message C also represents a command message from the user's mobile data communication device 24 to the host system 10. As described in more detail in FIG. 3, the host system 10 preferably includes, along with the typical hardware and software associated with a workstation or desktop computer, the redirector program 12, a TCP/IP subsystem 43, a primary message store 40, an E-mail subsystem 44, a screen saver subsystem 48, and a keyboard subsystem" Lazaridis, 5:44-56. "The mobile data communication device 24 includes software program instructions that work in conjunction with the redirector program 12 to enable the seamless, transparent redirection of user-selected data items. FIG. 4 describes the basic method steps of the redirector program 12, and FIG. 5 describes the steps of the corresponding program operating at the mobile device 24." *Id.*, 6:42-6:48. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 115 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"In preparing the message for redirection, the redirector program 12 could compress the original message A, could compress the message header, or could encrypt the entire message A to create a secure link to the mobile device 24. Also programmed into the redirector 12 is the address of the user's mobile data communication device 24, the type of device, and whether the device 24 can accept certain types of attachments, such as word processing or voice attachments." *Id.*, 7:53-7:60. "The redirector may also be programmed with a preferred list mode that is configured by the user either at the host system 10, or remotely from the user's mobile data communication device by transmitting a command message C. The preferred list contains a list of senders (other users) whose messages are to be redirected or a list of message characteristics that determine whether a message is to be redirected. If activated, the preferred list mode causes the redirector program 12 to operate like a filter, only redirecting certain user data items based on whether the data item was sent from a sender on the preferred list or has certain message characteristics that if present will trigger or suppress redirection of the message. In the example of FIG. 1, if desktop system 26 was operated by a user on the preferred list of host system 10, and the preferred list option was activated, then message A would be redirected. If, however, desktop 26 was operated by a user not on the host system's preferred list, then message A would not be redirected, even if the user of the host system had configured the redirector to push messages of type A The user of the host system 10 can configure the preferred list directly from the desktop system, or, alternatively, the user can then send a command message (such as C) from the mobile device 24 to the desktop system 10 to activate the preferred list mode, or to add or delete certain senders or message characteristics from the preferred list that was previously configured." *Id.*, 7:65-8:31. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 116 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "If the redirected user data item is an E-mail message, as described above, the user at the mobile device 24 sees the original subject, sender's address, destination address, carbon copy and blind carbon copy. When the user replies to this message, or when the user authors a new message, the software operating at the mobile device 24 adds a similar outer envelope to the reply message (or the new message) to cause the message to be routed first to the user's host system 10, which then removes the outer envelope and redirects the message to the final destination, such as back to computer 26. In the preferred embodiment, this results in the outgoing redirected message from the user's host system 10 being sent using the E-mail address of the host mailbox, rather than the address of the mobile device, so that it appears to the recipient of the message that the message originated from the user's desktop system 10 rather than the mobile data communication device. Any replies to the redirected message will then be sent to the desktop system 10, which if it is still in redirector mode, will repackage the reply and resend it to the user's mobile data device, as described above." *Id.*, 8:66-9:20. "If the message C is from an Internet computer to the user's desktop system and the user has redirection capabilities, then the server 11 detects the message C, repackages it using electronic envelope B, and redirects the repackaged message (C in B) to the user's mobile device 24. If the message C is a command message from the user's mobile device 24, then the server 11 simply acts upon the command message." *Id.*, 10:2-8.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 117 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The E-Mail sub-system 44 can receive messages A from external computers on the LAN 14, or can receive messages C from some external network such as the Internet 18 or a wireless data communication network 22, and stores these messages in the primary data store 40. Assuming that the redirector 12 has been triggered to redirect messages of this type, the redirector detects the presence of any new messages and instructs the E-Mail system 44 to repackage the message by placing an outer wrapper B about the original message A (or C), and by providing the addressing information of the mobile data communication device 24 on the outer wrapper B. As noted above, this outer wrapper B is removed by the mobile device 24, and the original message A (or C) is then recovered, thus making the mobile device 24 appear to be the desktop system 10." *Id.*, 10:58-11:5. "In addition, the E-Mail sub-system 44 receives messages back from the mobile device 24 having an outer wrapper with the addressing information of the desktop system 10, and strips this information away so that the message can be routed to the proper sender of the original message A (or C). The E-Mail sub-system also receives command messages C from the mobile device 24 that are directed to the desktop system 10 to trigger redirection or to carry out some other function. The functionality of the E-Mail sub-system 44 is controlled by the redirector program 12." Lazaridis, 11:6-15. "As described previously, the present invention can be triggered to begin redirection upon detecting numerous external, internal and networked events, or trigger points. Examples of external events include: receiving a command message from the user's mobile data communication device 24 to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer that is connected to the host system via a network to initiate redirection." *Id.*, 11:28-42.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 118 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"At step 82, the mobile device waits for messages and signals 84 to be generated or received. Assuming that the redirector software 12 operating at the user's desktop system 10 is configured to redirect upon receiving a message from the user's mobile device 24, at step 86, the user can decide to generate a command message that will start redirection. If the user does so, then at step 88 the redirection message is composed and sent to the desktop system 10 via the wireless network 22, through the wireless gateway 20, via the Internet 18 to the LAN 14, and is finally routed to the desktop machine 10. In this situation where the mobile device 24 is sending a message directly to the desktop system 10, no outer wrapper is added to the message (such as message C in FIGS. 1 and 2)." *Id.*, 13:6-19. "If, however, the redirection flag is on, then at step 110 the reply message is repackaged with the outer envelope having the addressing information of the user's desktop system 10, and the repackaged message is then transmitted to the desktop system 10 at step 106." *Id.*, 13:62-67. **Combination with TS-23.140** TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id.* A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 119 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS-23.140, as evidenced by the example citations below, discloses the device link agents (e.g., MMS User Agents) receiving secure Internet data messages (e.g., receiving secure messages) from the network server system over the respective secure Internet data message link (e.g., secure link utilizing MM1 Transport Protocol with TCP/IP and TLS), including messages collected from multiple ones of the network elements (e.g., a variety of other MMS User Agents, MMS VAS Applications, etc.) and messages corresponding to multiple ones of the software components authorized to receive messages via the device link agent on that respective device (e.g., authorized applications registered to receive messages via the MMS User Agent), wherein at least a first subset of the secure Internet data messages contain both a unique identifier for a corresponding one of the software agents (e.g., a destination application identifier) and data to be consumed by that software component (e.g., application data), the data supplied from a respective network element corresponding to that software component (e.g., data supplied by the originating network element such as a MMS User Agent or MMS VAS Application corresponding to the application). *See, e.g.*, "Abstract message: information which is transferred between two MMS entities used to convey an MM and/or associated control information between these two entities" TS-23.140, 14.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 120 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The MMSE is a collection of MMS-specific network elements under the control of a single administration." *Id.*, 17. "The MMS Relay/Server shall provide the following functionalities: receiving and sending MM" *Id.*, 21. "If an MMS User Agent supports submission of MMs the MMS User Agent shall be able to: Indicate the address of the MM recipient" *Id.*, 26. "The recipient MMS User Agent shall be able to request retrieval of an MM from the recipient MMS Relay/Server based on the Message Reference received in a notification. If MMBoxes are supported, the MMS User Agent shall be able to request retrieval of an MM from the user's MMBox, based on a Message Reference received from a previous MMBox operation." *Id.*, 29. "Within a request for delivery of an MM the recipient MMS User Agent shall be able to indicate its capabilities towards the recipient MMS Relay/Server. The recipient MMS User Agent may indicate its capabilities towards the recipient MMS Relay/Server by transmitting: a set of information describing the terminal's capabilities, including a binary flag indicating whether or not the terminal supports Application data." *Id.*, 30. "Apart from using MMS as a service for users to exchange messages, MMS may also be used to transport data specific to applications. Applications that intend to transport application specific data using MMS may either reside on an MMS User Agent or on an MMS VAS Application." *Id.*, 54. "Applications that want to transport data specific to applications other than MMS will initially need to register with the appropriate MMS User Agent or MMS VAS Application. During this registration process the application provisions an MMS User Agent or an MMS VAS Application with its application identification value and may negotiate with the MMS User Agent or MMS VAS Application the details (amount and format) of information to be exchanged between the two entities. The application registration

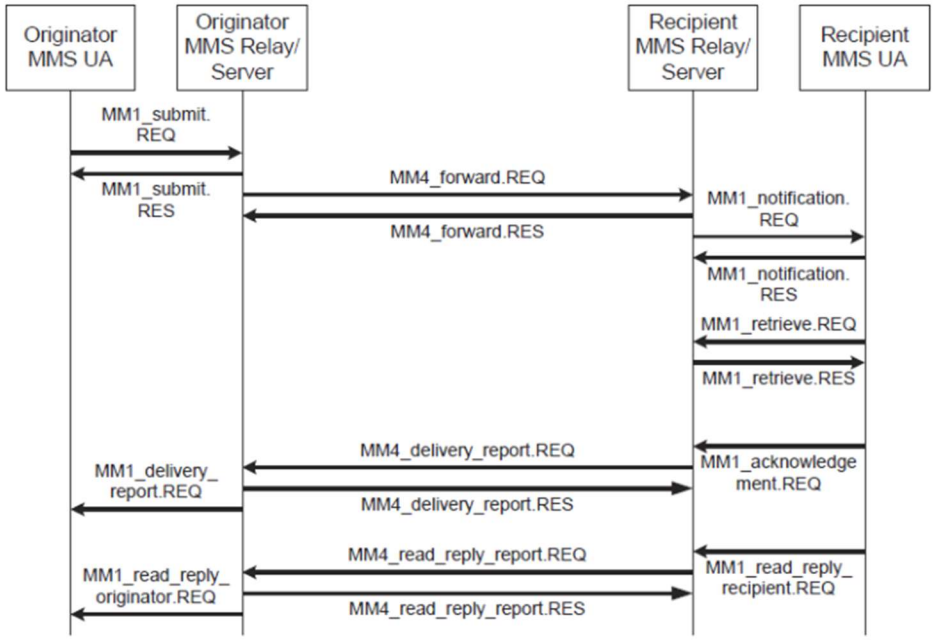
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 121 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	process is outside the scope of this specification. The registration may be an inherent process e.g., in the application's integration into a mobile phone. It may also be the initial step after the download of a downloadable application to a mobile phone. Whatever the details of the application registration process are, an MMS User Agent or an MMS VAS Application acts according to the negotiated results from the application registration process." *Id.* "MMS is used to transport data specific to applications between two MMS User Agents or an MMS User Agent and an MMS VAS Application (or vice versa)." *Id.*, 55. "The application identifier of the destination application shall be present in an abstract message, while the identifier of a 'reply-path' and some additional application/implementation specific control information may be present in an abstract message." *Id.* "The additional application/implementation specific control information shall be used for all future needs that are not supported by the application identifier of the destination application and the identifier of the originating application, such as specifying a particular logical channel in the application addressing method (e.g., 'discussion thread #05') or distinguishing between multiple instances of the same application (e.g., 'chess application #02')." *Id.* "Based on the negotiated details upon application registration process an application may trigger an MMS User Agent or an MMS VAS Application to submit certain abstract messages. Upon triggering an MMS User Agent or an MMS VAS Application to send an abstract message the MMS User Agent or MMS VAS Application may receive information from the application. The MMS User Agent or MMS VAS Application may insert this information in both the information elements and/or payload (if present) of the abstract message. The details for the above are according to the results of the application registration process. Abstract messages that are sent by an MMS User Agent or an MMS VAS Application on behalf of an originating application shall contain a destination application identifier. They may, in addition, contain an application identifier which is to be used in reply-MMs and they may contain additional application/implementation specific control information." *Id.*

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 122 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Abstract messages that are sent by an MMS User Agent or an MMS VAS Application on behalf of an originating application shall contain a destination application identifier. They may, in addition, contain an application identifier which is to be used in reply-MMs and they may contain additional application/implementation specific control information." *Id.* "If the destination application does not reside on the receiving MMS User Agent or on the USIM or MMS VAS Application, the MMS User Agent or MMS VAS Application shall discard the corresponding abstract message. In such a case the recipient MMS Relay/Server and recipient MMS User Agent or VAS application shall apply the normal reporting behaviour towards sending entities." *Id.*, 56. "If the MMS Relay/Server finds from the recipient MMS User Agent's capability indication (see clause 7.1.3.1) that the recipient MMS User Agent supports transport of application data, the MMS Relay/Server . . . shall pass on the destination application identifier, the 'reply-path' identifier (if present) and the additional application/implementation specific control information (if present) unaltered." *Id.*, 55-56. "Upon reception of an abstract message containing a destination application identifier (it can either be the MM1_notification.REQ, MM1_retrieve.RES or MM7_deliver.REQ transactions), the receiving MMS User Agent or MMS VAS Application shall first check if the destination application resides on it." *Id.*, 56. "[T]he originator MMS Relay/Server should use the recipient's address(es) as indicated in the corresponding MM to route the converted message towards its recipient(s). In addition to this, it may convey message class, priority and subject of the associated MM as part of the converted message." *Id.*, 90. "The recipient address over MM1 may or may not be mapped to recipient address over MM7. The recipient address over MM7 may also be independent of the recipient address over MM1." *Id.*, 190.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 123 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 6: Example Abstract Message Flow** *Id.*, Fig. 6.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 124 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
1[k]: for software components that are authorized to access messages received via the device link agent, cause messages with a unique identifier corresponding to a given one of those software applications to be securely delivered to a software process corresponding to the given software component.	Blackberry Push, as evidenced by the example citations below, discloses, for software components that are authorized to access messages received via the device link agent, cause messages with a unique identifier corresponding to a given one of those software applications to be securely delivered to a software process corresponding to the given software component. Blackberry Push enforces application authorization and dispatches messages to the indicated process/application via the device link agent, using application/channel identifiers. *See, e.g.*, • MDS Technical Series, p. 44: Push Applications • BlackBerry enables you to pro-actively PUSH new web pages and links to your users' handhelds • Why would I want to push web pages to my users?? – Alert them of new information based on an event that occurred on the server – 'Silently' update frequently accessed pages within the browser's local cache, so the user can access the most recent information even when out of coverage – Push a permanent link to a frequently accessed site (such as an internal portal or entry point to an application) • WAP 2.0 – PAP – Acknowledgments – Reliability • MDS Technical Series, p. 45: Push Types

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 125 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• BlackBerry-defined push types: – "Channel Push" • Push an icon to the Home screen along with a web page • The icon acts as a permanent 'channel' to the referenced URL – "Message Push" • Push a message to the user's inbox that automatically launches the browser to a specified page (cached or not) – "Cache Push" • Push a page to the browser's cache, but do not notify the user • MDS Studio 4.1, p. 12: Message Message components are designed to communicate application data across the wireless network between the client application on the BlackBerry device and the data source. Using the BlackBerry MDS Studio, bindings are created that link specific message components to specific data source operations. An instance of a message component can be either an •inbound message: transmits data from the data source to the application on the BlackBerry device •outbound message: transmits data from the application on the BlackBerry device to the data source • MDS 4.1 Overview, p. 13: Pushing content to users BlackBerry MDS Studio applications use WS-Eventing to send information proactively to the device. For push events, an outbound message component is designed to be bound at the time of design to the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 126 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	subscription web service that is associated with the event. An inbound message is also designed to be bound to the web service event. At runtime, the application can subscribe to receive the web service event (for example, through BlackBerry device user input that results in the outbound subscription message being sent). Web service events cause inbound messages to be sent to the application. BlackBerry MDS Studio applications can notify BlackBerry device users of inbound messages in several ways, including a flashing LED, a tune, or a vibration. • MDS 4.1 Overview, p. 17: Pushing content to users BlackBerry MDS Java applications can use HTTP POST or PAP requests to send information proactively to the BlackBerry device. The request headers include information about the content to be pushed, the destination BlackBerry device or user, and the BlackBerry device application for which the data is intended (such as the BlackBerry Browser or a custom application). The request body contains the content. When the connection service on the BlackBerry Enterprise Server receives the HTTP POST message, it initiates the connection to the BlackBerry devices. • Java Development 4.0, p. 121: Client/server push applications: Data is sent to a custom Java application on the handheld. Client/server push applications consist of a custom client application for the handheld and a server-side application that pushes content to it. This approach provides more control over the type of content that you can send out and how this data is processed and displayed on the handheld compared to browser push applications. To the extent this limitation is not disclosed by Blackberry Push, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 127 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, TS-23.140. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 128 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Lazaridis, as evidenced by the example citations below, discloses that for software components that are authorized to access messages received via the device link agent (e.g., messages received via mobile data communication device), the device link agents (e.g., the software program instructions) cause messages with a unique identifier corresponding to a given one of those software applications (e.g., messages with a destination address/addressing information) to be securely delivered (e.g., messages delivered via the secure connection) to a software process corresponding to the given software component (e.g., software). *See, e.g.*, "For example, if a user is traveling and receives a message with an attachment that the user's mobile device can process or display, the user may from a mobile communications device send a command message to the host system indicating that that attachment is to be sent to a fax machine at a hotel where the user will be spending the evening. This enables the user to receive important E-mail attachments as long as the host system is provided with sufficient information about the destination where the attachment is to be forwarded." Lazaridis, 3:56-65. "Message C in FIG. 1 represents an external message from a sender that is not directly connected to LAN 14, such as the user's mobile data communication device 24, some other user's mobile device (not shown), or any user connected to the Internet 18. Message C also represents a command message from the user's mobile data communication device 24 to the host system 10. As described in more detail in FIG. 3, the host system 10 preferably includes, along with the typical hardware and software associated with a workstation or desktop computer, the redirector program 12, a TCP/IP subsystem 43, a primary message store 40, an E-mail subsystem 44, a screen saver subsystem 48, and a keyboard subsystem" *Id.*, 5:44-56. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 129 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "The redirector may also be programmed with a preferred list mode that is configured by the user either at the host system 10, or remotely from the user's mobile data communication device by transmitting a command message C. The preferred list contains a list of senders (other users) whose messages are to be redirected or a list of message characteristics that determine whether a message is to be redirected. If activated, the preferred list mode causes the redirector program 12 to operate like a filter, only redirecting certain user data items based on whether the data item was sent from a sender on the preferred list or has certain message characteristics that if present will trigger or suppress redirection of the message. In the example of FIG. 1, if desktop system 26 was operated by a user on the preferred list of host system 10, and the preferred list option was activated, then message A would be redirected. If, however, desktop 26 was operated by a user not on the host system's preferred list, then message A would not be redirected, even if the user of the host system had configured the redirector to push messages of type A The user of the host system 10 can configure the preferred list directly from the desktop system, or, alternatively, the user can then send a command message (such as C) from the mobile device 24 to the desktop system 10 to activate the preferred list mode, or to add or delete certain senders or message characteristics from the preferred list that was previously configured." *Id.*, 7:65-8:31. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 130 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "If the redirected user data item is an E-mail message, as described above, the user at the mobile device 24 sees the original subject, sender's address, destination address, carbon copy and blind carbon copy. When the user replies to this message, or when the user authors a new message, the software operating at the mobile device 24 adds a similar outer envelope to the reply message (or the new message) to cause the message to be routed first to the user's host system 10, which then removes the outer envelope and redirects the message to the final destination, such as back to computer 26. In the preferred embodiment, this results in the outgoing redirected message from the user's host system 10 being sent using the E-mail address of the host mailbox, rather than the address of the mobile device, so that it appears to the recipient of the message that the message originated from the user's desktop system 10 rather than the mobile data communication device. Any replies to the redirected message will then be sent to the desktop system 10, which if it is still in redirector mode, will repackage the reply and resend it to the user's mobile data device, as described above." *Id.*, 8:66-9:20. "If the message C is from an Internet computer to the user's desktop system and the user has redirection capabilities, then the server 11 detects the message C, repackages it using electronic envelope B, and redirects the repackaged message (C in B) to the user's mobile device 24. If the message C is a command message from the user's mobile device 24, then the server 11 simply acts upon the command message." *Id.*, 10:2-8.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 131 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The E-Mail sub-system 44 can receive messages A from external computers on the LAN 14, or can receive messages C from some external network such as the Internet 18 or a wireless data communication network 22, and stores these messages in the primary data store 40. Assuming that the redirector 12 has been triggered to redirect messages of this type, the redirector detects the presence of any new messages and instructs the E-Mail system 44 to repackage the message by placing an outer wrapper B about the original message A (or C), and by providing the addressing information of the mobile data communication device 24 on the outer wrapper B. As noted above, this outer wrapper B is removed by the mobile device 24, and the original message A (or C) is then recovered, thus making the mobile device 24 appear to be the desktop system 10." *Id.*, 10:58-11:5. "In addition, the E-Mail sub-system 44 receives messages back from the mobile device 24 having an outer wrapper with the addressing information of the desktop system 10, and strips this information away so that the message can be routed to the proper sender of the original message A (or C). The E-Mail sub-system also receives command messages C from the mobile device 24 that are directed to the desktop system 10 to trigger redirection or to carry out some other function. The functionality of the E-Mail sub-system 44 is controlled by the redirector program 12." *Id.*, 11:6-15. "As described previously, the present invention can be triggered to begin redirection upon detecting numerous external, internal and networked events, or trigger points. Examples of external events include: receiving a command message from the user's mobile data communication device 24 to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer that is connected to the host system via a network to initiate redirection." *Id.*, 11:28-42.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 132 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"At step 82, the mobile device waits for messages and signals 84 to be generated or received. Assuming that the redirector software 12 operating at the user's desktop system 10 is configured to redirect upon receiving a message from the user's mobile device 24, at step 86, the user can decide to generate a command message that will start redirection. If the user does so, then at step 88 the redirection message is composed and sent to the desktop system 10 via the wireless network 22, through the wireless gateway 20, via the Internet 18 to the LAN 14, and is finally routed to the desktop machine 10. In this situation where the mobile device 24 is sending a message directly to the desktop system 10, no outer wrapper is added to the message (such as message C in FIGS. 1 and 2)." *Id.*, 13:6-19. "If, however, the redirection flag is on, then at step 110 the reply message is repackaged with the outer envelope having the addressing information of the user's desktop system 10, and the repackaged message is then transmitted to the desktop system 10 at step 106." *Id.*, 13:62-67. **Combination with TS-23.140** TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 133 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS-23.140, as evidenced by the example citations below, discloses for software components (e.g., applications) that are authorized (e.g., registered) to access messages received via the device link agent (e.g., MMS User Agent), the device link agents (e.g., MMS User Agents) causing messages with a unique identifier (e.g., MMs containing application identifiers) corresponding to a given one of those software applications to be securely delivered to a software process corresponding to the given software component (e.g., delivered using MM1 Transfer Protocols with TCP/IP and TLS for a secure link). *See, e.g.*, "Abstract message: information which is transferred between two MMS entities used to convey an MM and/or associated control information between these two entities" TS-23.140, 14. "The MMSE is a collection of MMS-specific network elements under the control of a single administration." *Id.*, 17. "The MMS Relay/Server shall provide the following functionalities: receiving and sending MM" *Id.*, 21.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 134 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"If an MMS User Agent supports submission of MMs the MMS User Agent shall be able to: Indicate the address of the MM recipient" *Id.*, 26. "The recipient MMS User Agent shall be able to request retrieval of an MM from the recipient MMS Relay/Server based on the Message Reference received in a notification. If MMBoxes are supported, the MMS User Agent shall be able to request retrieval of an MM from the user's MMBox, based on a Message Reference received from a previous MMBox operation." *Id.*, 29. "Within a request for delivery of an MM the recipient MMS User Agent shall be able to indicate its capabilities towards the recipient MMS Relay/Server. The recipient MMS User Agent may indicate its capabilities towards the recipient MMS Relay/Server by transmitting: a set of information describing the terminal's capabilities, including a binary flag indicating whether or not the terminal supports Application data." *Id.*, 30. "Apart from using MMS as a service for users to exchange messages, MMS may also be used to transport data specific to applications. Applications that intend to transport application specific data using MMS may either reside on an MMS User Agent or on an MMS VAS Application." *Id.*, 54. "Applications that want to transport data specific to applications other than MMS will initially need to register with the appropriate MMS User Agent or MMS VAS Application. During this registration process the application provisions an MMS User Agent or an MMS VAS Application with its application identification value and may negotiate with the MMS User Agent or MMS VAS Application the details (amount and format) of information to be exchanged between the two entities. The application registration process is outside the scope of this specification. The registration may be an inherent process e.g., in the application's integration into a mobile phone. It may also be the initial step after the download of a downloadable application to a mobile phone. Whatever the details of the application registration process

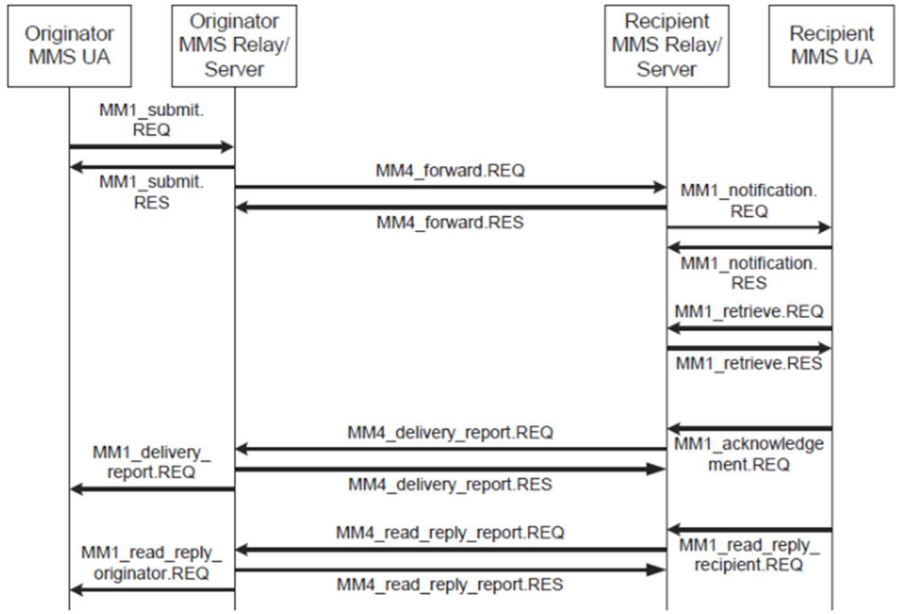
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 135 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	are, an MMS User Agent or an MMS VAS Application acts according to the negotiated results from the application registration process." *Id.* "MMS is used to transport data specific to applications between two MMS User Agents or an MMS User Agent and an MMS VAS Application (or vice versa)." *Id.*, 55. "The application identifier of the destination application shall be present in an abstract message, while the identifier of a 'reply-path' and some additional application/implementation specific control information may be present in an abstract message." *Id.* "The additional application/implementation specific control information shall be used for all future needs that are not supported by the application identifier of the destination application and the identifier of the originating application, such as specifying a particular logical channel in the application addressing method (e.g., 'discussion thread #05') or distinguishing between multiple instances of the same application (e.g., 'chess application #02')." *Id.* "Based on the negotiated details upon application registration process an application may trigger an MMS User Agent or an MMS VAS Application to submit certain abstract messages. Upon triggering an MMS User Agent or an MMS VAS Application to send an abstract message the MMS User Agent or MMS VAS Application may receive information from the application. The MMS User Agent or MMS VAS Application may insert this information in both the information elements and/or payload (if present) of the abstract message. The details for the above are according to the results of the application registration process. Abstract messages that are sent by an MMS User Agent or an MMS VAS Application on behalf of an originating application shall contain a destination application identifier. They may, in addition, contain an application identifier which is to be used in reply-MMs and they may contain additional application/implementation specific control information." *Id.*

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 136 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Abstract messages that are sent by an MMS User Agent or an MMS VAS Application on behalf of an originating application shall contain a destination application identifier. They may, in addition, contain an application identifier which is to be used in reply-MMs and they may contain additional application/implementation specific control information." *Id.* "If the destination application does not reside on the receiving MMS User Agent or on the USIM or MMS VAS Application, the MMS User Agent or MMS VAS Application shall discard the corresponding abstract message. In such a case the recipient MMS Relay/Server and recipient MMS User Agent or VAS application shall apply the normal reporting behaviour towards sending entities." *Id.*, 56. "If the MMS Relay/Server finds from the recipient MMS User Agent's capability indication (see clause 7.1.3.1) that the recipient MMS User Agent supports transport of application data, the MMS Relay/Server . . . shall pass on the destination application identifier, the 'reply-path' identifier (if present) and the additional application/implementation specific control information (if present) unaltered." *Id.*, 55-56. "Upon reception of an abstract message containing a destination application identifier (it can either be the MM1_notification.REQ, MM1_retrieve.RES or MM7_deliver.REQ transactions), the receiving MMS User Agent or MMS VAS Application shall first check if the destination application resides on it." *Id.*, 56. "[T]he originator MMS Relay/Server should use the recipient's address(es) as indicated in the corresponding MM to route the converted message towards its recipient(s). In addition to this, it may convey message class, priority and subject of the associated MM as part of the converted message." *Id.*, 90. "The recipient address over MM1 may or may not be mapped to recipient address over MM7. The recipient address over MM7 may also be independent of the recipient address over MM1." *Id.*, 190.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 137 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 6: Example Abstract Message Flow** *Id.*, Fig. 6.
2: The networked system of claim 1, the message buffer system logic to determine, for	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, the message buffer system logic to determine, for each of the wireless end-user devices, when one of a plurality of message delivery triggers has occurred, at least one of the triggers for each given device specific

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 138 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
each of the wireless end-user devices, when one of a plurality of message delivery triggers has occurred, at least one of the triggers for each given device specific to one or more states of that given device.	to one or more states of that given device. Blackberry Push evaluates device-specific states when determining delivery. *See, e.g.*, • MDS Studio 4.1, pp. 14-15: Delivery modes Messages that pass between BlackBerry MDS Applications and the BlackBerry MDS Services are categorized into two delivery modes: best-effort and standard. The default delivery mode is standard. Best-effort The best-effort delivery mode is a fire-and-forget mode. Messages might not be delivered, and they might be delivered in a different order than they were sent from the application to the BlackBerry MDS Services or from the BlackBerry MDS Services to the application. Standard Messages sent from the application are designed to reach the BlackBerry device, even if the BlackBerry device spends time outside of a wireless coverage area or if the BlackBerry MDS Services stop. The sending side transmits the message when a connection becomes available. The application does not confirm receipt of messages, so the application might not process a message even after it reaches the BlackBerry device. Similarly, the BlackBerry MDS Services might not process a message from the application even when the message reaches the BlackBerry MDS Services server. Standard mode messages are designed to be delivered and processed in the order in which they were generated. Handling messages when a BlackBerry device is outside of a wireless coverage area When a BlackBerry device is outside of a wireless coverage area, outbound standard and reliable application messages remain queued on the BlackBerry device until the BlackBerry device returns to a wireless coverage area, at which time the messages are transmitted. Best-effort messages are dropped when a BlackBerry device is outside of a wireless coverage area.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 139 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Message queue sizes BlackBerry MDS Applications have restricted inbound and outbound message queue sizes. These restrictions are imposed by policies sent to the BlackBerry MDS Runtime software on the BlackBerry devices. The system administrator of the BlackBerry MDS Services can push new policies to the BlackBerry MDS Runtime, which might change the limitations of an application's inbound and outbound queue sizes. Support for push events The push-based architecture that is used for BlackBerry email is available for custom BlackBerry MDS Applications. Using this push-based architecture, server-side applications can push new data, alerts, and notifications to users proactively. Message size limitations The maximum size for any inbound message is 32 KB. If the result of an operation performed by a web service requires a message size greater than 32 KB to be sent from the BlackBerry MDS Services to the BlackBerry MDS Application, the BlackBerry MDS Services send an error message to the application. Additionally, Blackberry Push supported WAP 2.0 – PAP, which utilized priorities, deliver-before, and deliver-after functionality. *See, e.g.*, • MDS Technical Series, p. 44: • BlackBerry enables you to pro-actively PUSH new web pages and links to your users' handhelds • Why would I want to push web pages to my users?? – Alert them of new information based on an event that occurred on the server – 'Silently' update frequently accessed pages within the browser's local cache, so the user can access the most recent information even when out of coverage – Push a permanent link to a frequently accessed site (such as an internal portal or entry point to an application) • WAP 2.0 – PAP

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 140 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	– Acknowledgments – Reliability • Java Development 4.0, pp. 121-22: Applications can push content to handhelds using one of two methods: • Push Access Protocol (PAP), which is part of the WAP 2.0 specification • RIM push Both push service implementations support the following tasks: • sending a server-side push submission • specifying a reliability mode for the push submission • In transport-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives acknowledgements for all sent packets. • In application-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives confirmation from the application that initiated the push. • specifying a deliver-before time-stamp for the push submission • requesting a result notification of the push submission The PAP implementation supports the following additional tasks: • specifying a deliver-after timestamp for the push submission • cancelling a push request submission • querying the status of a push request submission PAP pushes are stored in a database, whereas RIM pushes are stored in RAM. RIM pushes may be lost if the server is rebooted. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 141 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
3: The networked system of claim 1, wherein one of the message delivery triggers is the expiration of a periodic timer.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the message delivery triggers is the expiration of a periodic timer. Blackberry Push uses periodic timers in delivery control and system operation, including flow-control timeouts, HTTP connection timeouts (for handheld and server), and periodic logging/statistics intervals that govern processing cycles. *See, e.g.*, • MDS Technical Series, p. 10: MDS Architecture • Maximum size of HTTP data that handheld can fetch from the Internet is by default 128KB (configurable) • Timeout of 2 min (configurable) if the access to the Internet/data source is not reachable or the handheld is out of coverage Additionally, MDS allows HTTP configuration for authentication timeout, handheld connection timeout, server connection timeout, and a maximum number of redirects. See MDS Technical Series at p. 24. Additionally, Blackberry Push supported WAP 2.0 – PAP, which utilized priorities, deliver-before, and deliver-after functionality. *See, e.g.*, • MDS Technical Series, p. 44: • BlackBerry enables you to pro-actively PUSH new web pages and links to your users' handhelds • Why would I want to push web pages to my users?? – Alert them of new information based on an event that occurred on the server – 'Silently' update frequently accessed pages within the browser's local cache, so the user can access the most recent information even when out of coverage – Push a permanent link to a frequently accessed site (such as an internal portal or entry point to an application)

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 142 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• WAP 2.0 – PAP – Acknowledgments – Reliability • Java Development 4.0, pp. 121-22: Applications can push content to handhelds using one of two methods: • Push Access Protocol (PAP), which is part of the WAP 2.0 specification • RIM push Both push service implementations support the following tasks: • sending a server-side push submission • specifying a reliability mode for the push submission • In transport-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives acknowledgements for all sent packets. • In application-level reliability mode, messages are considered to be delivered when the Mobile Data Service receives confirmation from the application that initiated the push. • specifying a deliver-before time-stamp for the push submission • requesting a result notification of the push submission The PAP implementation supports the following additional tasks: • specifying a deliver-after timestamp for the push submission • cancelling a push request submission • querying the status of a push request submission PAP pushes are stored in a database, whereas RIM pushes are stored in RAM. RIM pushes may be lost if the server is rebooted. To the extent this limitation is not disclosed by Blackberry Push, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Adamczyk, Houghton, Munson, Herzog.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 143 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Lazaridis Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the message link server of claim 1, wherein one of the message delivery triggers is the expiration of a periodic timer (e.g., a programmable timer or calendar alarm). *See, e.g.*,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 144 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Using the redirector program, the user can select certain data items for redirection, such as E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal reminders etc. Having selected the data items for redirection, the user can then configure one or more event triggers to be sensed by the redirector program to initiate redirection of the user data items. These user-defined trigger points (or event triggers) include external events, internal events and networked events. Examples of external events include: receiving a message from the user's mobile data communication device to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the examples of the types of user-defined events that can trigger the redirector program to push data items from the host to the mobile device. Although in the preferred embodiment it is anticipated that the configuration that specifies which data items will be redirected and in what form will be set at the host system, it is within the scope of this invention that such configuration may be set or modified through data sent from the mobile communications device." Lazaridis, 3:9-35. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 145 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the events that could be used with the present invention to initiate replication of the user-selected data items from the host system 10 to the mobile device 24." *Id.*, 7:8-30. "As described previously, the present invention can be triggered to begin redirection upon detecting numerous external, internal and networked events, or trigger points. Examples of external events include: receiving a command message from the user's mobile data communication device 24 to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer that is connected to the host system via a network to initiate redirection." *Id.*, 11:28-42. "The screen saver and keyboard sub-systems 46, 48 are examples of systems that are capable of generating internal events. Functionally, the redirector program 12 provides the user with the ability to configure the screen saver and keyboard systems so that under certain conditions an event trigger will be generated that

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 146 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	can be detected by the redirector 12 to start the redirection process. For example, the screen saver system can be configured so that when the screen saver is activated, after, for example, 10 minutes of inactivity on the desktop system, an event trigger is transmitted to the redirector 12, which starts redirecting the previously selected user data items. In a similar manner the keyboard sub-system can be configured to generate event triggers when no key has been depressed for a particular period of time, thus indicating that redirection should commence. These are just two examples of the numerous application programs and hardware systems internal to the host system 10 that can be used to generate internal event triggers." *Id.*, 11:43-61. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Lazaridis, it would have been obvious in view of the disclosures in Lazaridis, or in combination with the knowledge of a person of ordinary skill in the art. Since Lazaridis discloses that an event trigger could be a programmable timer and a calendar alarm, it would have been obvious to a POSITA that a message delivery trigger could be a periodic timer. *Id.*, 3:22-24. **Combination with Adamczyk** Adamczyk and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Adamczyk is directed next generation networks (NGN) that encompass a wide range of access technologies and client devices, including both IP-based and non-IP (e.g., GSM, UMTS, POTS) endpoints. Adamczyk, ¶¶416, 432. Blackberry Push was a well-established system for enabling data and notification services on wireless devices, especially those with limited resources or non-IP connectivity. A POSITA would recognize that combining Adamczyk's notification middleware with Blackberry Push would allow the notification service to reach a broader set of mobile devices, including legacy and resource-constrained handsets.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 147 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Further, a POSITA would recognize that Adamczyk's architecture is designed to be compatible with a variety of industry standards, including 3GPP, IMS, and OSA/Parlay, and specifically references the use of "push services" and "notification service elements" in 3GPP and mobile environments *Id.*, ¶¶17, 385-388. Adamczyk references PAP as a standard for wireless notification delivery in mobile networks, which is utilized by Blackberry Push. Therefore, it would require no inventive skill to combine the disclosure of Adamczyk with Blackberry Push. A POSITA would have been motivated to implement the notification servers in the comparable message-transmission system of Adamczyk, Abstract, ¶¶7-9. Adamczyk's notification servers (like PPG) can be "configured to send [] notification messages to the recipients on demand, at a specific future time, and/or on a periodic schedule" including based on user preferences. Adamczyk, ¶10, 22. A POSITA would have been motivated to supplement Blackberry Push's triggers with an additional trigger like Adamczyk's periodic schedule trigger to provide users the option to specify a periodic schedule for message delivery. Adamczyk, as evidenced by the example citations below, discloses message delivery triggers being the expiration of a periodic timer. *See, e.g.*, "A system for supporting a notification capability for a plurality of applications utilizing a next generation network having a network layer includes an application services middleware between the applications and the network layer comprising a plurality of common infrastructure elements usable by the applications. The common infrastructure elements provide both services associated with use of the network and services that are not associated with use of the network. The plurality of common infrastructure elements includes a notification service that is configured to facilitate the sending of messages from authorized services to recipients." Adamczyk, Abstract. "According to some embodiments of the present invention, a system for supporting a notification capability for a plurality of applications utilizing a next generation network having a network layer includes an application services middleware between the applications and the network layer comprising a plurality of common infrastructure elements usable by the applications. The common infrastructure elements provide both services associated with use of the network and services that are not associated with use of the network.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 148 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	The plurality of common infrastructure elements includes a notification service that is configured to facilitate the sending of messages from authorized services to recipients. In other embodiments, the notification service is an Internet Protocol (IP) Multimedia Subsystem (IMS) network element. In still other embodiments, the notification service includes a notification server that is configured to receive the messages from the authorized services for delivery to the recipients and to send the messages to the recipients, an IMS Serving Call Session Control Function (S-CSCF) that is configured to maintain session state information between the notification server and the authorized services and between the notification server and the recipients, and a Home Subscriber Server (HSS) that is configured to store recipient profile and preference data. In still other embodiments, the notification server is configured to send the notification messages to the recipients on demand, at a specific future time, and/or on a periodic schedule." *Id*., ¶¶7-10. "In still further embodiments, applying the rules includes applying rules to the profile and preference data at the notification service to determine whether to send the message from one of the authorized services to one of the recipients on demand, at a specific future time, and/or on a periodic schedule." *Id*., ¶22. **Combination with Houghton** Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 149 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id*., p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id*., p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id*., p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, discloses wherein one of the message delivery triggers is the expiration of a periodic timer (e.g., a periodic message sent so that devices in the communication path do not time expire the connection). *See, e.g.*, "In an embodiment of the invention, the mobile terminal maintains contact with the server by periodically testing the connection with a suitable keepalive message, a 'ping', which resets timers in intervening operator stateful firewalls and/or NAT (network address translation) firewalls. The mobile terminal may also automatically re-establish the connection when the connection gets broken and adjust its connection settings based on past experience with failure to connect and such broken connections in a given mobile network.." Houghton, p. 11.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 150 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"IP push command messages may be triggered either by events which the server is aware of, or events which a similar mobile application on another mobile terminal is aware of and which that push client makes the server aware of, the server in turn initiating a push command event. Such client-side events of which the server is informed for the possibility of delivering new web services include the making, receiving, and termination of telephone calls, the transition from presence-to-absence or absence-to-presence of other devices in a geographical area, metropolitan area, local area or personal area wireless communications network, mobile terminal creation of photographs video audio or other media, mobile terminal presence of files matching a given filtering criteria, changes in internal state of electronic equipment, taking of measurements, processing of such measurements to see if they match certain criteria, video game actions or events, application calculation events and messaging actions. In some cases a separate server such as a telecommunication service server, gaming environment server, and original equipment manufacturer device or business information system may notify the server directly of events or messages to be sent using an external application programming interface (API). Such a device connected by external API may also receive information including the results and responses of messages which the user has replied to by terminal software or web page interaction." *Id*., p. 14. "Use of a periodic message from the client to the server or server to the client to the effect that client socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection" *Id*., p. 19. "Use of periodic messages initiated by the client 405 or by the server 401 to the effect that the receiving application is informed that their messages are continuing to be received, as indicated by response data from the receiving party to the sending party." *Id*. "Use of longer or shorter timing between such periodic messages to match the combined needs of all such network devices and protocols involved in transmitting communication, as adjusted based on the success or failure of earlier messages to successfully traverse the open network connection." *Id*. "In an embodiment of the invention, the mobile terminal 404 keeps a prioritized data structure listing the preferred communication protocols, server addresses, periodic message timings, network access settings,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 151 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	and alternate communication routes which it should use." *Id*., p. 21. "The push message from the push server 401 may be triggered by any trigger event, local or remote , defined at the server. The trigger may, among others, include an alarm, notification, or measurement result received to the push server 401 from another device or system." *Id*. "Referring to steps 620 and 621 in Figure 6B, after the login and connection setup in Figure 6A , the push client 405 may send a periodic message (the arrow C in Figure 4) from the client 405 to the server 401 to the effect that client socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection. The period of time (measured in seconds, minutes or hours and varying from mobile network to mobile network) between such periodic messages is set to match the combined needs of all such network devices and network protocols involved in transmitting communication (step 604)." *Id*., p. 26. "Referring to steps 1010 and 1011 in Figure 1 OB, after the login and connection setup, the push server 401 may receive a periodic message (the arrow C in Figure 4) from the client 405 to the effect that the server updates the respective server object associated with a given client so that the connection, server socket or client socket is not time expired as being stale. The server 401 may send a reply or acknowledgement in step 1012 to notify client 405 of successful network, firewall and security traversal." *Id*., p. 27. **Combination with Munson** Munson and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Munson discloses a method of "pushing contents to client devices," including "group pushes" in which content is buffered and sent to multiple devices simultaneously, "serializ[ing]" content such that a series of messages are delivered to a particular device simultaneously. Munson, Abstract, ¶37, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 152 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	A POSITA would have been motivated to combine Munson with Blackberry Push because a POSITA would have desired to support diverse client devices and network environments, as well as to leverage an existing, proven technology for wireless content delivery, e.g., Blackberry Push. Munson similarly teaches messages can be delivered from memory "according to a schedule of time or event" such as coordinating pushes "for a system maintenance purpose during off-peak hours." Munson, ¶¶40, 44. Thus, Munson provides additional details about how to implement a store-and-forward system by providing the computer memory at a server, and the computer programming to deliver messages from memory (including upon occurrence of certain events). *Id*., ¶¶36-38, 44, Fig. 4. Therefore, a POSITA would have been motivated to incorporate computer memory and programming into Blackberry Push to store and deliver messages from memory, consistent with Munson, to improve system flexibility and reliability. A POSITA would have had a reasonable expectation of success in combining Blackberry Push and Munson because Munson provides implementation details such push message system. *See, e.g.*, Munson, ¶¶36-38. A POSITA would have been motivated and had a reasonable expectation of success in incorporating conditions Munson teaches providing additional flexibility in how messages are delivered in PAP, used by Blackberry Push. This combination of Blackberry Push and Munson would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Munson, as the communication protocol because Blackberry Push was a well-known, industry-leading system. Munson, ¶¶35-36. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Munson push messaging system without requiring inventive skill. Munson, as evidenced by the example citations below, discloses one of the message delivery triggers is the expiration of a periodic timer (e.g., according to a schedule of time or event). *See, e.g.*, "In one embodiment, a content can be pushed according to a schedule of time or event. For example, when used with a group push, an application service provider can use pushes (i.e., pushing contents) for a system

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 153 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	maintenance purpose during off-peak hours to take advantage of lower rates." Munson, ¶44. "The system according to claim 1, wherein the contents are pushed to client devices according to a schedule of time or event." *Id*., claim 8. "The method according to claim 13, further comprising: pushing the contents to the each member of the group according to a schedule of time or event." *Id*., claim 17. **Combination with Herzog** Herzog and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Herzog is directed to solving the problem of delivering push messages to mobile clients that are non-addressable or unreachable from the public internet due to Network Address Translation, firewalls, or similar network restrictions, by using a gateway-based architecture with explicit client registration, persistent channels, and secure authorization mechanisms. Herzog, ¶¶1, 5, 18-26, 44-49. Blackberry Push was a well-established system designed to enable data and push notification services to mobile devices, particularly in wireless environments where direct addressing is often not possible due to the same NAT and firewall issues identified in Herzog. A POSITA would recognize that combining Herzog's gateway-based push messaging techniques with Blackberry Push would be an obvious solution to extend secure, reliable push messaging to a wide range of mobile devices operating in wireless network. A POSITA would be motivated to combine Herzog and Blackberry Push in order to deliver messages in asymmetric connectivity scenarios of Herzog. Therefore, it would require no inventive skill to combine the disclosures of Herzog and Blackberry Push. Herzog, as evidenced by the example citations below, discloses the message link server of claim 1, wherein one of the message delivery triggers is the expiration of a periodic timer (e.g., sending keep-alive messages

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 154 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	that are scheduled periodically). *See e.g.,* "In example embodiments, the server 130 is any type of server that is programmed to push content to the client 110. For example, the server 130 can be an email server, a sales server, a web server, a database server, a file server, a print server, or any other kind of line of business server. In one example, the server 130 is a Microsoft Exchange Server offered by Microsoft Corporation of Redmond, Wash. Other configurations are possible." Herzog, ¶24. "In one possible embodiment, the communication between the gateway 120 and server 130 is conducted using a Transport Layer Security (TLS) protocol or its predecessor, Secure Sockets Layer (SSL) protocol communication both of which provide secrecy as well as mutual authentication between the server 130 and the gateway 120 such that the gateway 120 securely knows the identity of the sending server 130. In alternative embodiments, the communication can be based on other protocols that provide secure communications on the Internet for Such activities as web browsing, e-mail, Internet faxing, instant messaging and other data transfers. The message can be in the form of Hypertext Transfer Protocol (HTTP) or Service Oriented Architecture Protocol (SOAP) message. In alternative embodiments, the message can be a message based on other protocols for exchanging messages over computer networks." *Id.*, ¶26. "Once the gateway 120 receives the message, the message can also be saved in a message buffer 127. In one possible embodiment, the gateway 120 operates in a mode queuing the message in the message buffer 127 until it can be delivered to the client. Preferably, the message buffer 127 is a persistent message storage such that messages are first stored in a shared message queue database, and any one of an array of the gateways can pick up messages for any of the clients 110 connected to it at the time. The advantages of this embodiment are that the messages survive long client device offline periods. In addition, the clients 110 can move between gateways 120 without losing their queued messages. Further, the gateways 120 can be recycled without any loss of messages." *Id.*, ¶31. "In one possible embodiment, the client push channel controller 316 maintains the open connection with the gateway 320 through keep-alive messages, which are periodic, small packets of data sent between the client 310 and the gateway 320. In other words, receiving the keep-alive messages indicates that the client is

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 155 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	connected and online. On the other hand, missing the keep-alive messages suggests that the client 310 is dis connected and offline." *Id.*, ¶36. "When there is a failure of connection between the client 310 and the gateway 320, the client push channel controller 316 supports recovering from the failure. The client push channel controller 316 can be configured to support various open channel types. For example, the client push channel controller 316 supports at least three different channel types: Virtual Private Network (VPN), long lived TCP and/or HTTP connection, and clearline of sight (non NAT-ed) registration of client Internet Protocol (IP) address (similar to dynamic DNS)." *Id.*, ¶37. "In example embodiments, the client 310 does not communicate directly with the server 330 to communicate its status since the direct communication consumes battery and overall air bandwidth. Instead, the server 330 gets the status information directly from the gateway 320, which gets the status information from the client 310. In one possible embodiment, the client registration and status module 324 tracks a communication link status of the client 310 based on keep-alive messages from the client 310. In another possible embodiment, the client registration and status module 324 can also learn about the client status from explicit disconnect signaling sent either from the client 310 or the radio controller or other networking component the client 310 is or was connected to. The client registration and status module 324 interacts with the gateway push channel controller 322 to figure out whether the client 310 is still connected, and to identify status change events (like disconnect and reconnect)." *Id.*, ¶41. "The client registration and status module 324 also distributes the communication link status of the client 310 to the server 330 when inquired or for subscribers when the communication link status changes. In particular, the server 330 can ask the gateway 320 with a one-time query for the client status. The server 330 can also subscribe to a client status 'news' channel which maintains the client status information. If servers 330 subscribes to the 'news'' channel, the gateway 320 will notify all the servers 330 on the subscription list when the gateway 320 learns about a client status. For example, the gateway 320 will notify the subscribers of the servers 330 when the client 310 changes connection status, i.e., comes online or goes offline.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 156 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Typically, with the absence of direct disconnect signaling, the gateway 320 may not know about the disconnection of the client 310 immediately. The gateway 320 learns about it implicitly when the next scheduled keep-alive message from the client 310 does not arrive, or when attempts to push down a message to the client 310 fail. In addition, when subscribing to the client status news channel, the server 330 may specify its tolerance for out-of-date information. The gateway 320 may adjust a keep-alive timer with the client 310 to comply with those tolerance requirements. For example, if the keep-alive timer is 30 minutes, but a server has a 10 minute tolerance, the keep-alive frequency may need to change to 10 minutes." *Id.*, ¶¶42-43.
4: The networked system of claim 3, wherein the period of the timer is fractionally shorter than a maximum data message interval beyond which the Internet data message link to the given device is taken down.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 3, wherein the period of the timer is fractionally shorter than a maximum data message interval beyond which the Internet data message link to the given device is taken down. As disclosed in claim 3, *supra*, periodic timers may be set for flow control, and PAP delivery allows a delivery to be set upon a certain timer. Also disclosed is a configurable 2-minute timeout. Given all these are configurable, a POSITA would understand that they may be set such that the period of the timer is shorter than any timeout value. This limitation would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Herzog, Houghton, Adamczyk. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network).

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 157 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the message link server of claim 9, wherein the period of the timer is fractionally shorter than a maximum data message interval beyond which the secure message link is taken down. *See, e.g.*, "Using the redirector program, the user can select certain data items for redirection, such as E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal reminders etc. Having selected the data items for redirection, the user can then configure one or more event triggers to be sensed by the redirector program to initiate redirection of the user data items. These user-defined trigger points (or event triggers) include external events, internal events and networked events. Examples of

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 158 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	external events include: receiving a message from the user's mobile data communication device to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the examples of the types of user-defined events that can trigger the redirector program to push data items from the host to the mobile device. Although in the preferred embodiment it is anticipated that the configuration that specifies which data items will be redirected and in what form will be set at the host system, it is within the scope of this invention that such configuration may be set or modified through data sent from the mobile communications device." Lazaridis, 3:9-35. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 159 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the events that could be used with the present invention to initiate replication of the user-selected data items from the host system 10 to the mobile device 24." *Id.*, 7:8-30. "As described previously, the present invention can be triggered to begin redirection upon detecting numerous external, internal and networked events, or trigger points. Examples of external events include: receiving a command message from the user's mobile data communication device 24 to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer that is connected to the host system via a network to initiate redirection." *Id.*, 11:28-42. "The screen saver and keyboard sub-systems 46, 48 are examples of systems that are capable of generating internal events. Functionally, the redirector program 12 provides the user with the ability to configure the screen saver and keyboard systems so that under certain conditions an event trigger will be generated that can be detected by the redirector 12 to start the redirection process. For example, the screen saver system can be configured so that when the screen saver is activated, after, for example, 10 minutes of inactivity on the desktop system, an event trigger is transmitted to the redirector 12, which starts redirecting the previously selected user data items. In a similar manner the keyboard sub-system can be configured to generate event triggers when no key has been depressed for a particular period of time, thus indicating that redirection

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 160 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	should commence. These are just two examples of the numerous application programs and hardware systems internal to the host system 10 that can be used to generate internal event triggers." *Id.*, 11:43-61. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Lazaridis, it would have been obvious in view of the disclosures in Lazaridis, or in combination with the knowledge of a person of ordinary skill in the art. Since Lazaridis discloses that an event trigger could be a programmable timer and a calendar alarm, it would have been obvious to a POSITA that a message delivery trigger could be a periodic timer. *Id.*, 3:22-24. And because the '320 Patent does not specify what constitutes "fractionally shorter," any degree shorter can be considered "fractionally shorter," as claimed, since it would be some "fraction" of the duration of the programmable timer. **Combination with Herzog** Herzog and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Herzog is directed to solving the problem of delivering push messages to mobile clients that are non-addressable or unreachable from the public internet due to Network Address Translation, firewalls, or similar network restrictions, by using a gateway-based architecture with explicit client registration, persistent channels, and secure authorization mechanisms. Herzog, ¶¶1, 5, 18-26, 44-49. Blackberry Push was a well-established system designed to enable data and push notification services to mobile devices, particularly in wireless environments where direct addressing is often not possible due to the same NAT and firewall issues identified in Herzog. A POSITA would recognize that combining Herzog's gateway-based push messaging techniques with Blackberry Push would be an obvious solution to extend secure, reliable push messaging to a wide range of mobile devices operating in wireless network. A POSITA would be motivated to combine Herzog and Blackberry Push in order to deliver messages in asymmetric connectivity scenarios of Herzog. Therefore, it would require no inventive skill to combine the disclosures of Herzog and

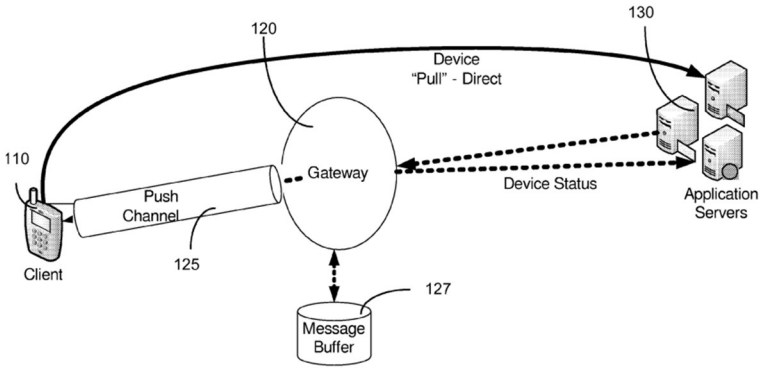
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 161 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Blackberry Push. Herzog, as evidenced by the example citations below, discloses a period of the timer is fractionally shorter than a maximum data message interval beyond which the secure message link is taken down. *See, e.g.*, "As discussed herein, an aspect of some embodiments relates to a method of authentication and authorization between a client, a server, and a gateway to facilitate communicating a message between the client and the server through the gateway, the client having a trusted relationship with each of the gateway and the server, the method including: registering the client with the gateway, including: forming a push channel between the client and the gateway to allow the gateway to communicate with the client, and receiving an address space identifying the gateway; constructing the address space identifying the gateway and the client; communicating the address space to the server; receiving an identity identifying the server; and if the client authorizes to receive a message from the server through the gateway, informing the authorization to the gateway, including: putting the identity identifying the server on a list of servers which are authorized to send messages to the client; and communicating the list of servers to the gateway." Herzog, ¶5. "In example embodiments, the server 130 is any type of server that is programmed to push content to the client 110. For example, the server 130 can be an email server, a sales server, a web server, a database server, a file server, a print server, or any other kind of line of business server. In one example, the server 130 is a Microsoft Exchange Server offered by Microsoft Corporation of Redmond, Wash. Other configurations are possible. When the server 130 wishes to send a message to the client 110, the server 130 sends the message to the gateway 120 first. After the gateway 120 receives the message from the server 130, the gateway 120 determines whether the gateway 120 should forward the message to the client 110 based on identifier (authentication and authorization) information in the message. The identifier information can be an address space such as that described above. The identifier information can also be a token, which is described further below in reference to FIGS. 8 and 9. Once the message is authorized by the gateway 120, the gateway 120 forwards the message to the client 110.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 162 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	In one possible embodiment, the communication between the gateway 120 and server 130 is conducted using a Transport Layer Security (TLS) protocol or its predecessor, Secure Sockets Layer (SSL) protocol communication both of which provide secrecy as well as mutual authentication between the server 130 and the gateway 120 such that the gateway 120 securely knows the identity of the sending server 130. In alternative embodiments, the communication can be based on other protocols that provide secure communications on the Internet for such activities as web browsing, e-mail, Internet faxing, instant messaging and other data transfers. The message can be in the form of Hypertext Transfer Protocol (HTTP) or Service Oriented Architecture Protocol (SOAP) message. In alternative embodiments, the message can be a message based on other protocols for exchanging messages over computer networks." *Id*., ¶¶24-26. "FIG. 2 is a block diagram illustrating communication modes that the server 130 may have with the client 110 in FIG. 1. The first mode is a pull mode. The pull mode is a network communication where the initial request for data originates from the client 110, and then is responded to by the server 130. The second mode is a push mode. In the push mode, the server 130 initiates a request to the client 110 through the gateway 120 and over a push channel 125. In other words, the server 130 sends a message to the client 110 through the gateway 120. In some embodiments, the message exchange between the server 130 and the gateway 120 is conducted using the address space or token described herein. Once the gateway 120 receives the message, the message can also be saved in a message buffer 127. In one possible embodiment, the gateway 120 operates in a mode queuing the message in the message buffer 127 until it can be delivered to the client. Preferably, the message buffer 127 is a persistent message storage such that messages are first stored in a shared message queue database, and any one of an array of the gateways can pick up messages for any of the clients 110 connected to it at the time. The advantages of this embodiment are that the messages survive long client device offline periods. In addition, the clients 110 can move between gateways 120 without losing their queued messages. Further, the gateways 120 can be recycled without any loss of messages. The third mode is the status query mode where a query is sent to the gateway 120 by the server 130. A response is sent back by the gateway 120 indicating the status of the client 110, i.e., whether or not the client

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 163 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	110 is offline or connected to the gateway 120." *Id.*, ¶¶30-32.  FIG. 2 *Id.*, Fig. 2. "The client push channel controller 316 is responsible for controlling the open channel 315 between the client 310 and the gateway 320. The client push channel controller 316 supports both connecting and disconnecting from the gateway 320. In addition, the client push channel controller 316 maintains an open connection with the gateway 320. In one possible embodiment, the client push channel controller 316 maintains the open connection with the gateway 320 through keep-alive messages, which are periodic, small packets of data sent between the client 310 and the gateway 320. In other words, receiving the keep-alive messages indicates that the client is connected and online. On the other hand, missing the keep-alive messages suggests that the client 310 is disconnected and offline. When there is a failure of connection between the client 310 and the gateway 320, the client push channel controller 316 supports recovering from the failure. The client push channel controller 316 can be configured

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 164 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	to support various open channel types. For example, the client push channel controller 316 supports at least three different channel types: Virtual Private Network (VPN), long lived TCP and/or HTTP connection, and clear line of sight (non NAT-ed) registration of client Internet Protocol (IP) address (similar to dynamic DNS)." *Id*., ¶¶36-37. "In example embodiments, the client 310 does not communicate directly with the server 330 to communicate its status since the direct communication consumes battery and overall air bandwidth. Instead, the server 330 gets the status information directly from the gateway 320, which gets the status information from the client 310. In one possible embodiment, the client registration and status module 324 tracks a communication link status of the client 310 based on keep-alive messages from the client 310. In another possible embodiment, the client registration and status module 324 can also learn about the client status from explicit disconnect signaling sent either from the client 310 or the radio controller or other networking component the client 310 is or was connected to. The client registration and status module 324 interacts with the gateway push channel controller 322 to figure out whether the client 310 is still connected, and to identify status change events (like disconnect and reconnect). The client registration and status module 324 also distributes the communication link status of the client 310 to the server 330 when inquired or for subscribers when the communication link status changes. In particular, the server 330 can ask the gateway 320 with a one-time query for the client status. The server 330 can also subscribe to a client status "news" channel which maintains the client status information. If servers 330 subscribes to the "news" channel, the gateway 320 will notify all the servers 330 on the subscription list when the gateway 320 learns about a client status. For example, the gateway 320 will notify the subscribers of the servers 330 when the client 310 changes connection status, i.e., comes online or goes offline. Typically, with the absence of direct disconnect signaling, the gateway 320 may not know about the disconnection of the client 310 immediately. The gateway 320 learns about it implicitly when the next scheduled keep-alive message from the client 310 does not arrive, or when attempts to push down a message to the client 310 fail. In addition, when subscribing to the client status news channel, the server 330 may specify its tolerance for out-of-date information. The gateway 320 may adjust a keep-alive timer with the client 310 to comply with those tolerance requirements. For example, if the keep-alive timer is 30 minutes, but a server has a 10 minute tolerance, the keep-alive frequency may need to change to 10 minutes." *Id*.,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 165 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	¶¶41-43. **Combination with Houghton** Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 166 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, discloses wherein the period of the timer is fractionally shorter than a maximum data message interval beyond which the secure message link is taken down (e.g., the periodic message sent so the connection does not expire, where, e.g., a POSITA would understand the period sending sets the maximum data message interval beyond which the secure message link is taken down because, otherwise, the connection expires). *See, e.g.*, "A second reason for automatically adjusting the network parameters is not simply establishing the connection but establishing the connection in the most cost effective or highest performance configuration available for the individual mobile terminal user account service level. Connection types differ in speed, customer expensive, server resource usage and hence scalability, security and ability to penetrate gateway security devices. Some customers may be offered more secure or higher performance service while others receive the least cost connectivity from either the mobile terminal's or the push server's standpoint. As an example, such network parameters may include but are not limited to one or more of the following - Use of a connection-oriented protocol suitable to the wireless network such as TCP/IP protocol, which is generally preferred in the common situation where an intermediate firewall, Network Address Translation (NAT) or gateway device 402 - Use of a connectionless protocol suitable to the wireless network 403, such as UDP/IP, which is generally preferred when no network device restricting terminal contact 402 is present as this reduces resource usage on the server when large number of mobile terminals is served - Use of an alternate connection-oriented protocol such as HTTP suitable for mobile push operation through more strict firewalls and proxy Internet connections - Use of a secure protocol such as HTTPS, IP-Sec, secure IP6 or a proprietary security protocol to identify the communicating parties, prevent message interception by 3rd parties, and prevent message modification by 3rd parties - Use of a periodic message from the client to the server or server to the client to the effect that client

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 167 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection - Use of configuration messages to the firewall (FIG 4, A) using a firewall or other security configuration protocol, such as Universal Plug and Play, to open a path for server-initiated connection-oriented or connectionless messaging to the client - Use of periodic messages initiated by the client 405 or by the server 401 to the effect that the receiving application is informed that their messages are continuing to be received, as indicated by response data from the receiving party to the sending party - Use of longer or shorter timing between such periodic messages to match the combined needs of all such network devices and protocols involved in transmitting communication, as adjusted based on the success or failure of earlier messages to successfully traverse the open network connection - Use of an alternate push server network address such as an address accessible with fewer security and performance restrictions which may be more available, more accessible, lower cost, lower latency, less busy, higher bandwidth or otherwise preferred at the present time and from the present network location of mobile terminal 404. - Use of one or more alternative mobile terminal network connection settings, or variation of the parameters within the settings. - Use of an alternate connection route through a second mobile or fixed terminal accessible to the mobile terminal over an alternate wireless network or network connection and capable of communication with a push server. In an embodiment of the invention, the mobile terminal 404 keeps a prioritized data structure listing the preferred communication protocols, server addresses, periodic message timings, network access settings, and alternate communication routes which it should use. Automated and manually assisted use of this structure should then result in a connection to one or more push servers, however if one of these connections which the mobile terminal is programmed to expect is not available, the mobile terminal can choose the sleep that function of the client software and attempt re-connection at a later time, repeating in like fashion with the same or different network connection parameters as described above until a connection is

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 168 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	established or other user intervention. At any time after a connection is established between the client 405 and the server 401 , the client may push a message to the server or the server may send a push message to the client. This is illustrated by means of the arrow C in Figure 4. These messages through a previously established, connection-oriented protocol such as TCP/IP, SSL, HTTP or HTTPS arrive quite rapidly, more rapidly than if the connection where established only when needed. The associated protocol overhead such as 3-way handshake during connection establishment thus occurs before the actual need for message transmission rather than needing to occur at time or transmission and thus delaying the transmission." Houghton, pp. 19-20. "Referring to steps 620 and 621 in Figure 6B, after the login and connection setup in Figure 6A, the push client 405 may send a periodic message (the arrow C in Figure 4) from the client 405 to the server 401 to the effect that client socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection The period of time (measured in seconds, minutes or hours and varying from mobile network to mobile network) between such periodic messages is set to match the combined needs of all such network devices and network protocols involved in transmitting communication (step 604) The client 405 may then wait for a reply or acknowledgement from the server 401 , and if the acknowledgement is received (step 622), the client 405 returns to step 620 to await the next combined connection refresh and test If the no acknowledgement is received, the client 405 may return to step 604 in Figure 6A to possibly automatically adjust network settings or, if it deems the settings correct but the network not available, simply re-attempt login to the server 401" *Id.*, p. 26. **Combination with Adamczyk** Adamczyk and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Adamczyk is directed next generation networks (NGN) that encompass a wide range of access technologies

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 169 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	and client devices, including both IP-based and non-IP (e.g., GSM, UMTS, POTS) endpoints. Adamczyk, ¶¶416, 432. Blackberry Push was a well-established system for enabling data and notification services on wireless devices, especially those with limited resources or non-IP connectivity. A POSITA would recognize that combining Adamczyk's notification middleware with Blackberry Push would allow the notification service to reach a broader set of mobile devices, including legacy and resource-constrained handsets. Further, a POSITA would recognize that Adamczyk's architecture is designed to be compatible with a variety of industry standards, including 3GPP, IMS, and OSA/Parlay, and specifically references the use of "push services" and "notification service elements" in 3GPP and mobile environments *Id.*, ¶¶17, 385-388. Adamczyk references PAP as a standard for wireless notification delivery in mobile networks, which is utilized by Blackberry Push. Therefore, it would require no inventive skill to combine the disclosure of Adamczyk with Blackberry Push. A POSITA would have been motivated to implement the notification servers in the comparable message-transmission system of Adamczyk, Abstract, ¶¶7-9. Adamczyk's notification servers (like PPG) can be "configured to send [] notification messages to the recipients on demand, at a specific future time, and/or on a periodic schedule" including based on user preferences. Adamczyk, ¶10, 22. A POSITA would have been motivated to supplement Blackberry Push's triggers with an additional trigger like Adamczyk's periodic schedule trigger to provide users the option to specify a periodic schedule for message delivery. Adamczyk, as evidenced by the example citations below, discloses that the message link server of claim 9, wherein the period of the timer is fractionally shorter than a maximum data message interval beyond which the secure message link is taken down (e.g., sending a notification message on a periodic schedule that would be some degree shorter). *See, e.g.,* "In still other embodiments, the notification server is configured to send the notification messages to the recipients on demand, at a specific future time, and/or on a periodic schedule." *Id.*, ¶10. "In still further embodiments, applying the rules includes applying rules to the profile and preference data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 170 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	at the notification service to determine whether to send the message from one of the authorized services to one of the recipients on demand, at a specific future time, and/or on a periodic schedule." *Id.*, ¶22. "The system of claim 3, wherein the notification server is configured to send the notification messages to the recipients on demand, at a specific future time, and/or on a periodic schedule." *Id.*, cl. 4.
5: The networked system of claim 1, wherein one of the message delivery triggers is the receipt of a transmission on the respective secure Internet data message link from the device link agent of the given one of the wireless end-user devices, or a response generated to a transmission received from that device link agent.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the message delivery triggers is the receipt of a transmission on the respective secure Internet data message link from the device link agent of the given one of the wireless end-user devices, or a response generated to a transmission received from that device link agent. Blackberry Push supports request/response workflows and device-originated transmissions that trigger server-side delivery of queued content. *See, e.g.*, • MDS Studio 4.1, p. 13: Messaging model BlackBerry MDS Applications are designed to store and process data locally on the BlackBerry device, without the need for a continuously available network connection. The communication model used to send and receive application messages is fully asynchronous. This means that if an outbound message is generated at a time when the BlackBerry device is outside of a wireless coverage area (for example, as a result of user input), the message is placed into an outbound queue until the wireless connection becomes available. When the message is placed into this queue, the application continues functioning, regardless of whether the message has been sent. Messages are transmitted from the queue in the background when the network connection becomes available. Similarly, inbound messages can be queued by the BlackBerry MDS Services and transmitted to the BlackBerry device only when the wireless connection becomes available. BlackBerry MDS Applications are designed to process inbound messages as asynchronous events. This asynchronous communication

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 171 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	model is designed for applications that are inherently tolerant to sudden or unpredictable loss of connection. This feature, combined with the ability to store and process data locally, enables BlackBerry MDS Applications to be designed in such a way that application functionality and enterprise data are available, even when wireless connections are not available. • MDS Technical Series, p. 9: MDS Architecture • HTTP connection requests utilize the existing secure 3DES (or AES) connection over port 3101 • MDS communicates with web servers using protocol requested by handheld • MDS supports common protocols and mark up languages – HTTP, HTTPS, TCP, XML, HTML • Handheld and MDS communicate using IP Proxy Protocol – IPPP service book required on the handheld • Java Development 4.0, p. 122: Application-level acknowledgement When the push arrives at a handheld, the application acknowledges the content. The Mobile Data Service initiates a connection to the URL specified during the push request to inform the server of the delivery. If an error is encountered, the Mobile Data Service sends an error message to the server. Application-level acknowledgement is available in handheld software version 4.0. RIM push provides an application-preferred option, which uses application-level acknowledgement in handheld software version 4.0 and transport-level acknowledgement otherwise. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 172 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
6: The networked system of claim 1, wherein one of the message delivery triggers is a heartbeat message generated by the given device link agent, or a request received from the given device link agent.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the message delivery triggers is a heartbeat message generated by the given device link agent, or a request received from the given device link agent. Blackberry Push handles device requests and periodic operations that function as heartbeats and triggers. *See, e.g.*, • MDS Technical Series, p. 10: MDS Architecture • Maximum size of HTTP data that handheld can fetch from the Internet is by default 128KB (configurable) • Timeout of 2 min (configurable) if the access to the Internet/data source is not reachable or the handheld is out of coverage • MDS Technical Series, p. 49: Centralized Push • MDS traffic has highest priority • Delivery retry is handled by the relay • MDS does not retry if push / packet expires • By default MDS supports 1000 connections • Push has more of a performance impact than pull Additionally, MDS allows HTTP configuration for authentication timeout, handheld connection timeout, server connection timeout, and a maximum number of redirects. See MDS Technical Series at p. 24. Additionally, Blackberry Push supported WAP 2.0 – PAP, which utilized priorities, deliver-before, and deliver-after functionality. *See supra*.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 173 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	This limitation would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Shen, Houghton, Herzog. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 174 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the message delivery triggers is a heartbeat message generated by the given device link agent, or a request received from the given device link agent (e.g., message sent from the mobile data communication device). *See, e.g.*, "A system and method for pushing information from a host system to a mobile data communication device upon sensing a triggering event is disclosed. A redirector program operating at the host system enables a user to continuously redirect certain user-selected data items from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. The redirector program operates in connection with event generating applications and repackaging systems at the host system to configure and detect a particular user-defined event, and then to repackage the user-selected data items in an electronic wrapper prior to pushing the data items to the mobile device. The mobile device and the host system share a common electronic address so that messages generated at either the host system or the mobile data communication device are configured using the common electronic address." Lazaridis, Abstract. "The present invention overcomes the problems noted above and satisfies the needs in this field for a system and method of pushing user-selected data items from a host system to a user's mobile data communication device upon detecting the occurrence of one or more user-defined event triggers." *Id.*, 2:48-53. "A redirector program operating at the host system enables the user to redirect or mirror certain user-selected data items (or parts of data items) from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. Also operating at the host system are various sub-systems that can be configured to create triggering events, such as a screen saver sub-system or a keyboard sub-system, as well as sub-systems for repackaging the user's data items for transparent delivery to the mobile data device, such as a TCP/IP sub-system or one or more E-Mail sub-systems. Other

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 175 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	sub-systems for creating triggering events and repackaging the user's data items could also be present at the host system. The host system also includes a primary memory store where the user's data items are normally stored." *Id.*, 2:61-3:8. "Using the redirector program, the user can select certain data items for redirection, such as E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal reminders etc. Having selected the data items for redirection, the user can then configure one or more event triggers to be sensed by the redirector program to initiate redirection of the user data items. These user-defined trigger points (or event triggers) include external events, internal events and networked events. Examples of external events include: receiving a message from the user's mobile data communication device to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the examples of the types of user-defined events that can trigger the redirector program to push data items from the host to the mobile device. Although in the preferred embodiment it is anticipated that the configuration that specifies which data items will be redirected and in what form will be set at the host system, it is within the scope of this invention that such configuration may be set or modified through data sent from the mobile communications device." *Id.*, 3:9-35. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 176 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the events that could be used with the present invention to initiate replication of the user-selected data items from the host system 10 to the mobile device 24." *Id.*, 7:8-30. "In the case where message C is representative of a command message from the user's mobile device 24 to the host system 10, the command message C is not redirected, but is acted upon by the host system." *Id.*, 8:62-65. "The event triggers are preferably detected at the user's desktop system 10, 26, 28 and can be any of the external, internal or network events listed above. The desktop systems 10, 26, 28 preferably detect these events and then transmit a message to the server computer 11 via LAN 14 to initiate redirection." *Id.*, 9:40-46.

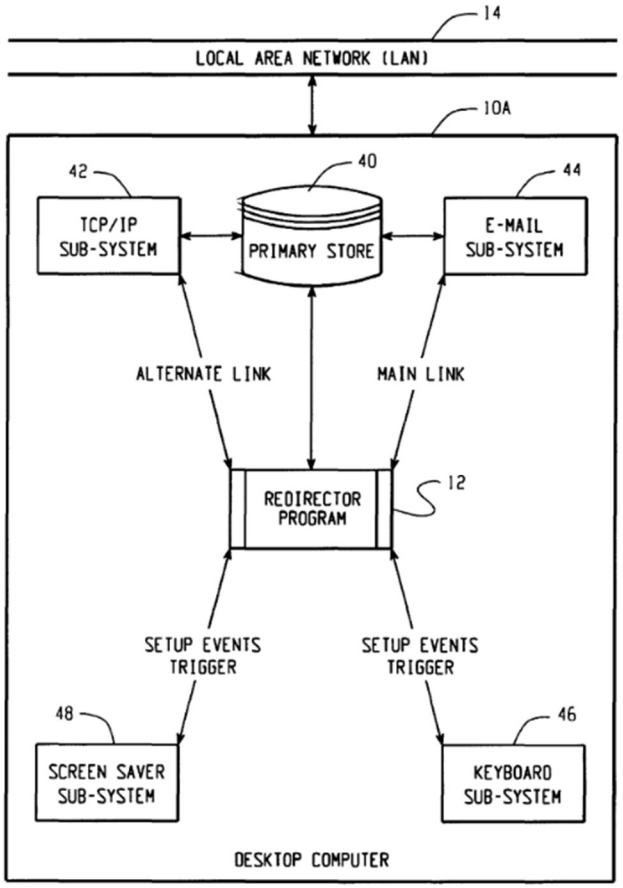
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 177 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"If the message C is a command message from the user's mobile device 24, then the server 11 simply acts upon the command message." *Id.*, 10:7-9. "The E-Mail sub-system also receives command messages C from the mobile device 24 that are directed to the desktop system 10 to trigger redirection or to carry out some other function." *Id.*, 11:11-14. "As described previously, the present invention can be triggered to begin redirection upon detecting numerous external, internal and networked events, or trigger points. Examples of external events include: receiving a command message from the user's mobile data communication device 24 to begin redirection; receiving a similar message from some external computer; sensing that the user is no longer in the vicinity of the host system; or any other event that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer that is connected to the host system via a network to initiate redirection." *Id.*, 11:28-42. "In addition to the redirection signal, the mobile device 24 could transmit any number of other commands to control the operation of the host system, and in particular the redirector program 12. For example, the mobile 24 could transmit a command to put the host system into the preferred list mode, and then could transmit additional commands to add or subtract certain senders from the preferred list." *Id.*, 13:20-26. "A method of redirecting messages between a host system and a mobile data communication device, comprising the steps of: configuring one or more redirection events at the host system; detecting that a redirection event has occurred at the host system and generating a redirection trigger; receiving messages directed to a first address at the host system from a plurality of message senders; in response to the redirection trigger, continuously redirecting the messages from the host system to the mobile data communication device; receiving the messages at the mobile data communication device; generating reply messages at the mobile data communication device to be sent to the plurality of message senders and

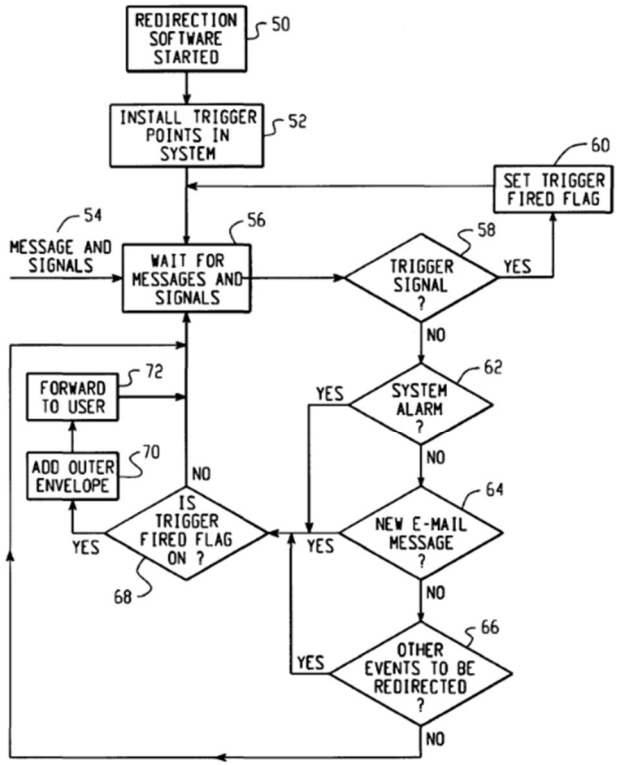
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 178 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	transmitting the reply messages to the host system; receiving the reply messages at the host system and configuring address information of the reply messages such that the reply messages use the first address associated with the host system as the originating address, wherein messages generated at either the host system or the mobile data communication device share the first address; and transmitting the reply messages from the host system to the plurality of message senders." *Id.*, claim 1.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 179 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 3 *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 180 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 4 *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 181 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Shen Shen and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Shen and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. Shen contemplates using existing or future protocols, such as HTTP or WAP, for sending and receiving MMS messages, making the integration of Blackberry Push a predictable solution to enable MMS over telecom networks. Shen, ¶42. A POSITA would have found it obvious to combine Shen's MMS system with Blackberry Push architecture. A POSITA would have been motivated to combine Shen's enterprise-focused MMS architecture (Shen, ¶2) with Blackberry Push's mobile communication system to support messaging across both wireless LAN and telecom networks. This combination of Blackberry Push and Shen would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Shen, as the communication protocol because Blackberry Push was a well-known, industry-leading system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Shen's MMS messaging system without requiring inventive skill. A POSITA would have been motivated to implement Shen's "heartbeat" as an additional trigger in the Blackberry Push's push notifications. Specifically, one Blackberry Push trigger occurs when a recipient device agent becomes available/reachable. Shen's heartbeat was one well-known way to determine whether device agent has become "reachable"—i.e., the device pings the gateway at intervals to indicate reachability/availability. Shen, as evidenced by the example citations below, discloses a heartbeat transmission message generated by the given device link agent, or a request received from the given device link agent. *See, e.g.*, "The invention also provides a MMS user device, a MMS gateway that may be used in above system, and

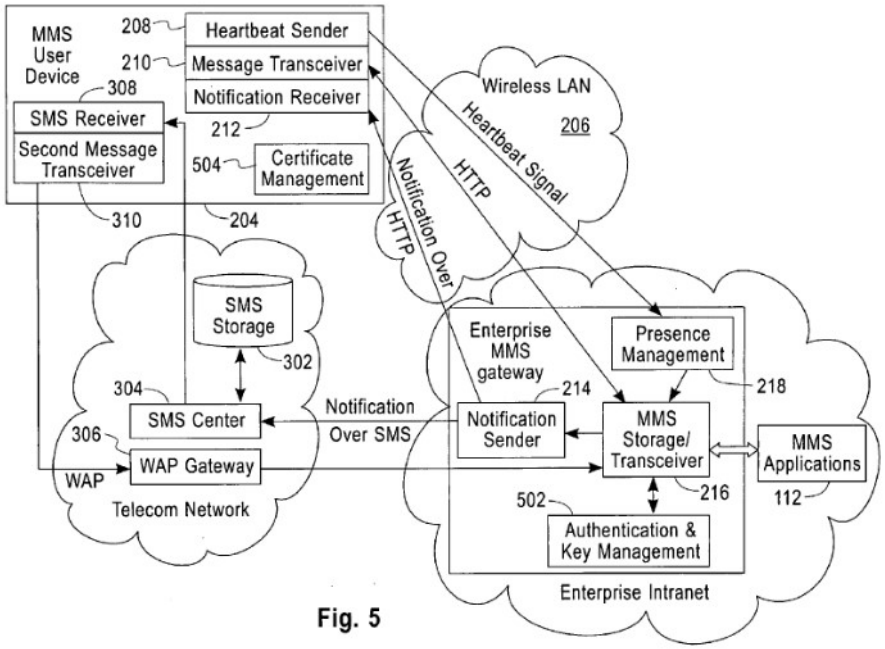
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 182 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	a method for receiving/sending MMS message. Specifically, the invention provides a MMS user device comprising: a heartbeat sender for sending heartbeat signals to be detected by a wireless LAN; a notification receiver for receiving from the wireless LAN a notification indicating a fact that there is a multimedia message to be received; a Message transceiver for sending multimedia message by sending a request carrying the multimedia message over the wireless LAN, and in response to the notification, sending a request carrying storage location information of the multimedia message over the wireless LAN so that the wireless LAN receives the multimedia message." Shen, ¶18. "FIG. 2 is a block diagram showing a first embodiment of a MMS system, a MMS gateway 202 and a user device 204 according to the present invention. In the following description to the MMS system, gateway and user device, a method for receiving and sending multimedia message of the present invention is implied. A concept of the present invention is to send and receive multimedia message over wireless LAN. That is, the multimedia messaging system of the present invention comprises a wireless LAN 206, a MMS user device 204 for sending and receiving multimedia messages over the wireless LAN, and a MMS gateway 202 for receiving, storing and sending multimedia messages over the wireless LAN. Thus, a direct MMS channel is established between the MMS user device 204 for sending and receiving multimedia messages over the wireless LAN and the MMS gateway 202 for receiving, storing and sending multimedia messages over the wireless LAN. As a result, security is ensured and cost will be reduced because of using the wireless LAN. Therefore, to an enterprise, its mobile employees may use MMS user devices 204, such as mobile phones supporting wireless LAN, to interact with MMS applications 112 (communicated with an enterprise MMS gateway) in the enterprise. Sending and receiving multimedia messages over the wireless LAN may be realized with any existing or future techniques. For example, the HTTP protocol, or any other protocols, may be used to send and/or receive multimedia messages. In the present specification including the accompanying drawings, as an example, the HTTP protocol is adopted in the wireless LAN. Obviously, this should not be interpreted as a limit to the scope of protection of the present invention.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 183 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	The MMS user device 204 may comprise a message transceiver 210 and a notification receiver 212, and the MMS gateway 202 may comprises a notification sender 214 and a multimedia message storage/transceiver 216 (hereinafter referred to as "MMS storage/transceiver"). The MMS storage/transceiver functions to receive, store and send multimedia messages and communicate with the backend MMS applications in the enterprise. The notification sender functions only to send MMS notifying message to the user device. When a MMS user device is to send a MMS message, the message transceiver sends a request (such as a HTTP request when the HTTP protocol is used) carrying the MMS message to the MMS storage/transceiver. When the MMS user device is to receive a MMS message, that is, when the MMS storage/transceiver is to send a message to the MMS user device, the MMS gateway first calls the notification sender 214 to send a MMS notifying message carrying location information of the MMS message to the MMS user device, the location information indicates the storage location of the MMS message in the MMS storage/transceiver, and is URL information when the HTTP protocol is used. Next, the MMS user device retrieves the location information such as URL from the MMS notifying message. Then the message transceiver sends a request (such as a HTTP request) carrying the location information to the MMS storage/transceiver. And last, the MMS storage/transceiver retrieves the MMS message and returns it to the MMS user device. The message is received by the message transceiver. In order that the MMS user device is able to determine whether a wireless LAN is available, or in order that the MMS gateway is able to determine whether a MMS user device is reachable, in a preferred embodiment, the MMS user device may further comprise a heartbeat sender, and the MMS gateway may further comprises a presence manager. In a wireless LAN environment, a MMS user device sends heartbeat signals periodically to the enterprise MMS gateway. The heartbeat signals contain information such as about user device number and IP address, which are also recorded in the database in the presence manager. If the gateway periodically receives the heartbeat signals, it means the corresponding user is located in the coverage of the wireless LAN, and the gateway may send a notification to the user when there is a message to be sent to that user; otherwise it means the user is located in the telecom network or offline, and the gateway does not send any notification

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 184 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	to the user even if there is a MMS message to be sent to that user." Shen, ¶¶29-34.  Fig. 5 *Id.*, Fig. 5.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 185 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Houghton Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, discloses wherein the transmission is a heartbeat

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 186 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	message generated by the given device link agent, or a request received from the given device link agent (e.g., a periodic message). *See, e.g.*, "IP push command messages may be triggered either by events which the server is aware of, or events which a similar mobile application on another mobile terminal is aware of and which that push client makes the server aware of, the server in turn initiating a push command event. Such client-side events of which the server is informed for the possibility of delivering new web services include the making, receiving, and termination of telephone calls, the transition from presence-to-absence or absence-to-presence of other devices in a geographical area, metropolitan area, local area or personal area wireless communications network, mobile terminal creation of photographs video audio or other media, mobile terminal presence of files matching a given filtering criteria, changes in internal state of electronic equipment, taking of measurements, processing of such measurements to see if they match certain criteria, video game actions or events, application calculation events and messaging actions. In some cases a separate server such as a telecommunication service server, gaming environment server, and original equipment manufacturer device or business information system may notify the server directly of events or messages to be sent using an external application programming interface (API). Such a device connected by external API may also receive information including the results and responses of messages which the user has replied to by terminal software or web page interaction." Houghton, p. 14. "Figure 4 illustrates an exemplary implementation of the system for pushing application commands to a mobile terminal based on server side and client side events in accordance with one embodiment of the present invention." *Id*., p. 16. "Use of a periodic message from the client to the server or server to the client to the effect that client socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection" *Id*., p. 19. "Use of periodic messages initiated by the client 405 or by the server 401 to the effect that the receiving application is informed that their messages are continuing to be received, as indicated by response data from

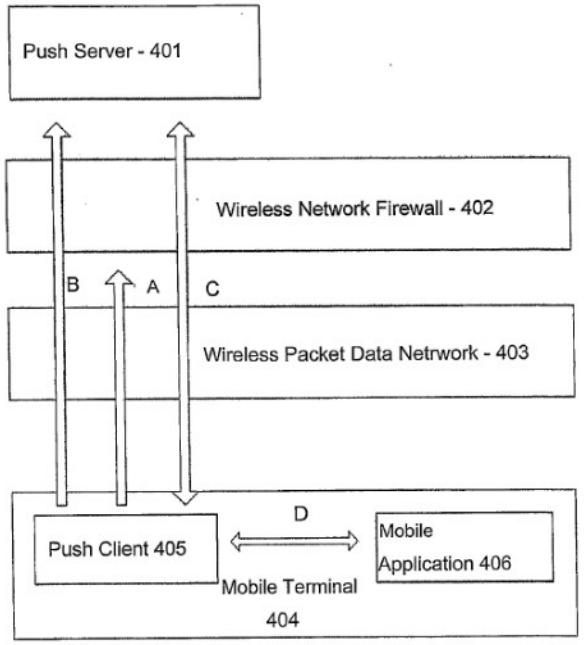
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 187 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the receiving party to the sending party." *Id*. "Use of longer or shorter timing between such periodic messages to match the combined needs of all such network devices and protocols involved in transmitting communication, as adjusted based on the success or failure of earlier messages to successfully traverse the open network connection." *Id*. "A connectionless protocol, such as UDP/IP, may also used in cases where the connection-oriented protocol is not possible or not desirable because of the push server resources used by every static instance of a connection-oriented protocol. Connectionless communication is also used when a message needs to get through to the server 401 quickly, such as when occurs when the client 405 notifies the server 401 of an terminal event, user interface event or application event but the connection-oriented protocol connection is not connected or is already busy transferring information" *Id*., p. 20. "Such a selective use of this high priority, low latency communication channel from terminal 404 to server 401 gives the server the opportunity to in return push to the client 405 a service triggered by the event begin communicated by the client 405 as will be described in more detail below. The sending of such a return message is frequently time critical and the fastest available combination of techniques will be used." *Id*., pp. 20-21. "The push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server. The trigger may, among others, include an alarm, notification, or measurement result received to the push server 401 from another device or system." *Id*., p. 21. "In an embodiment of the invention, the mobile terminal 404 keeps a prioritized data structure listing the preferred communication protocols, server addresses, periodic message timings, network access settings, and alternate communication routes which it should use." *Id*. "Referring to steps 620 and 621 in Figure 6B, after the login and connection setup in Figure 6A , the push client 405 may send a periodic message (the arrow C in Figure 4) from the client 405 to the server 401 to the effect that client socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection. The period of

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 188 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	time (measured in seconds, minutes or hours and varying from mobile network to mobile network) between such periodic messages is set to match the combined needs of all such network devices and network protocols involved in transmitting communication (step 604).” *Id*., p. 26. “If a push message is received, the client 405 executes the corresponding function, such as launch or pass commands and data a mobile application (step 632). If required, the client 405 may also send a message to the server 401 (step 633), such as a command execution acknowledgement or result.” *Id*. “Referring to Figure 60, after the login and connection setup, the push client 405 may monitor the local resources for a state change (a trigger event) in step 641. If a state change of other event is detected in step 642, the push client 405 may send a message to the server 401 (step 643).” *Id*. “Referring to steps 1010 and 1011 in Figure 1 OB, after the login and connection setup, the push server 401 may receive a periodic message (the arrow C in Figure 4) from the client 405 to the effect that the server updates the respective server object associated with a given client so that the connection, server socket or client socket is not time expired as being stale. The server 401 may send a reply or acknowledgement in step 1012 to notify client 405 of successful network, firewall and security traversal.” *Id*., p. 27.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 189 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id*. Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 190 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Herzog Herzog and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Herzog is directed to solving the problem of delivering push messages to mobile clients that are non-addressable or unreachable from the public internet due to Network Address Translation, firewalls, or similar network restrictions, by using a gateway-based architecture with explicit client registration, persistent channels, and secure authorization mechanisms. Herzog, ¶¶1, 5, 18-26, 44-49. Blackberry Push was a well-established system designed to enable data and push notification services to mobile devices, particularly in wireless environments where direct addressing is often not possible due to the same NAT and firewall issues identified in Herzog. A POSITA would recognize that combining Herzog's gateway-based push messaging techniques with Blackberry Push would be an obvious solution to extend secure, reliable push messaging to a wide range of mobile devices operating in wireless network. A POSITA would be motivated to combine Herzog and Blackberry Push in order to deliver messages in asymmetric connectivity scenarios of Herzog. Therefore, it would require no inventive skill to combine the disclosures of Herzog and Blackberry Push. Herzog, as evidenced by the example citations below, discloses wherein the transmission is a heartbeat message generated by the given device link agent, or a request received from the given device link agent. (e.g., sending keep-alive messages that are scheduled periodically). *See e.g.*, "In example embodiments, the server 130 is any type of server that is programmed to push content to the client 110. For example, the server 130 can be an email server, a sales server, a web server, a database server, a file server, a print server, or any other kind of line of business server. In one example, the server 130 is a Microsoft Exchange Server offered by Microsoft Corporation of Redmond, Wash. Other configurations are possible." Herzog, ¶24. "In one possible embodiment, the communication between the gateway 120 and server 130 is conducted

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 191 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	using a Transport Layer Security (TLS) protocol or its predecessor, Secure Sockets Layer (SSL) protocol communication both of which provide secrecy as well as mutual authentication between the server 130 and the gateway 120 such that the gateway 120 securely knows the identity of the sending server 130. In alternative embodiments, the communication can be based on other protocols that provide secure communications on the Internet for Such activities as web browsing, e-mail, Internet faxing, instant messaging and other data transfers. The message can be in the form of Hypertext Transfer Protocol (HTTP) or Service Oriented Architecture Protocol (SOAP) message. In alternative embodiments, the message can be a message based on other protocols for exchanging messages over computer networks." *Id*., ¶26. "Once the gateway 120 receives the message, the message can also be saved in a message buffer 127. In one possible embodiment, the gateway 120 operates in a mode queuing the message in the message buffer 127 until it can be delivered to the client. Preferably, the message buffer 127 is a persistent message storage such that messages are first stored in a shared message queue database, and any one of an array of the gateways can pick up messages for any of the clients 110 connected to it at the time. The advantages of this embodiment are that the messages survive long client device offline periods. In addition, the clients 110 can move between gateways 120 without losing their queued messages. Further, the gateways 120 can be recycled without any loss of messages." *Id*., ¶31. "In one possible embodiment, the client push channel controller 316 maintains the open connection with the gateway 320 through keep-alive messages, which are periodic, small packets of data sent between the client 310 and the gateway 320. In other words, receiving the keep-alive messages indicates that the client is connected and online. On the other hand, missing the keep-alive messages suggests that the client 310 is dis connected and offline. When there is a failure of connection between the client 310 and the gateway 320, the client push channel controller 316 supports recovering from the failure. The client push channel controller 316 can be configured to support various open channel types. For example, the client push channel controller 316 supports at least three different channel types: Virtual Private Network (VPN), long lived TCP and/or HTTP connection, and clearline of sight (non NAT-ed) registration of client Internet Protocol (IP) address (similar to dynamic

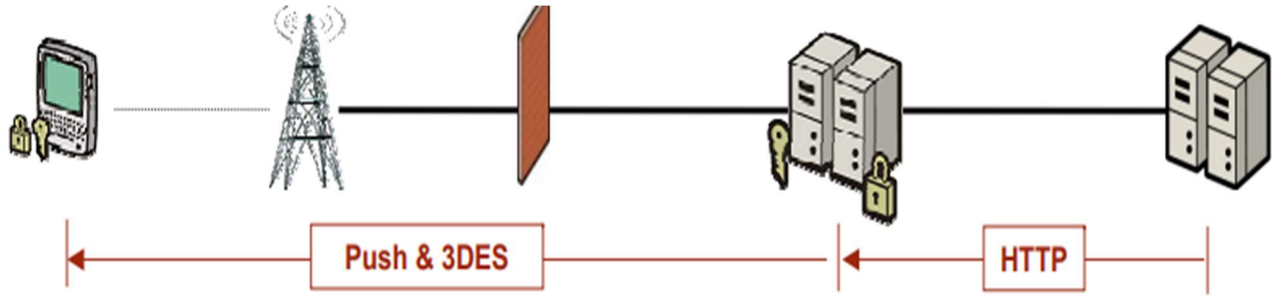
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 192 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	DNS).” *Id*., ¶¶36-37. “In example embodiments, the client 310 does not communicate directly with the server 330 to communicate its status since the direct communication consumes battery and overall air bandwidth. Instead, the server 330 gets the status information directly from the gateway 320, which gets the status information from the client 310. In one possible embodiment, the client registration and status module 324 tracks a communication link status of the client 310 based on keep-alive messages from the client 310. In another possible embodiment, the client registration and status module 324 can also learn about the client status from explicit disconnect signaling sent either from the client 310 or the radio controller or other networking component the client 310 is or was connected to. The client registration and status module 324 interacts with the gateway push channel controller 322 to figure out whether the client 310 is still connected, and to identify status change events (like disconnect and reconnect). The client registration and status module 324 also distributes the communication link status of the client 310 to the server 330 when inquired or for subscribers when the communication link status changes. In particular, the server 330 can ask the gateway 320 with a one-time query for the client status. The server 330 can also subscribe to a client status ‘news’ channel which maintains the client status information. If servers 330 subscribes to the ‘news’’ channel, the gateway 320 will notify all the servers 330 on the subscription list when the gateway 320 learns about a client status. For example, the gateway 320 will notify the subscribers of the servers 330 when the client 310 changes connection status, i.e., comes online or goes offline. Typically, with the absence of direct disconnect signaling, the gateway 320 may not know about the disconnection of the client 310 immediately. The gateway 320 learns about it implicitly when the next scheduled keep-alive message from the client 310 does not arrive, or when attempts to push down a message to the client 310 fail. In addition, when subscribing to the client status news channel, the server 330 may specify its tolerance for out-of-date information. The gateway 320 may adjust a keep-alive timer with the client 310 to comply with those tolerance requirements. For example, if the keep-alive timer is 30 minutes, but a server has a 10 minute tolerance, the keep-alive frequency may need to change to 10 minutes.” *Id*., ¶¶41-43.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 193 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
7: The networked system of claim 1, wherein one of the message delivery triggers is the receipt of a particular message from one of the network elements.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the message delivery triggers is the receipt of a particular message from one of the network elements. Blackberry Push delivers content in response to server-side events, including receipt of a message from an application server. *See* claim 1, *supra*. *See also, e.g.*, • MDS Technical Series, p. 45: BlackBerry-defined push types: – "Channel Push" • Push an icon to the Home screen along with a web page • The icon acts as a permanent 'channel' to the referenced URL – "Message Push" • Push a message to the user's inbox that automatically launches the browser to a specified page (cached or not) – "Cache Push" • Push a page to the browser's cache, but do not notify the user

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 194 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 • Mens (2006), p. 9: The push program When a document is ready to be sent out, the pushout program is invoked with the tag identifying the document to send. Pushout scans the database looking for subscribers to that document tag-type. For each subscriber found, pushout invokes an HTTP request to the MDS to push the document to the user's devices. For each of the invocations, pushout constructs a list of headers containing information which is submitted to the MDS. • MDS Studio 4.1, p. 12: Message Message components are designed to communicate application data across the wireless network between the client application on the BlackBerry device and the data source. Using the BlackBerry MDS Studio, bindings are created that link specific message components to specific data source operations. An instance of a message component can be either an •inbound message: transmits data from the data source to the application on the BlackBerry device •outbound message: transmits data from the application on the BlackBerry device to the data source

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 195 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.
8: The networked system of claim 1, further comprising a secure server to store a secure authorization list, the secure authorization list indicating the authorized software components and the authorized network elements that are allowed to communicate using the network server system.	Blackberry Push, as evidenced by the example citations below, discloses networked system of claim 1, further comprising a secure server to store a secure authorization list, the secure authorization list indicating the authorized software components and the authorized network elements that are allowed to communicate using the network server system. Blackberry Push provides secure authorization and policy signaling to devices specifying which applications/components are authorized to receive push/data. *See, e.g.*, • MDS Technical Series, p. 14: MDS Security Proxy Authentication • Automatic Proxy Authentication – Can specify user ID and password for MDS to login to proxy server – Reduces number of requests made from handheld • Using the handheld to authenticate – Going to the handheld for authentication is the default option – More secure option • The user has to enter the credentials in the following format: – username@domain + password or – domain\username + password – username + password (if a default_domain is set in MDSLogin.conf)

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 196 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS Technical Series, p. 19: • Push authentication – Push initiator has to be authenticated to MDS using HTTP basic authentication • MDS Technical Series, p. 48: Centralized Push • One MDS can be the Centralized Push Server, which can route push requests to users on other MDS servers • MDS uses the email or pin to forward the message to the correct device – MDS uses email-to-PIN mappings of users stored in a database – Must know both the address of the MDS and the email address or the PIN of the user • To push to a user, push request must be sent to the MDS that acts as the Centralized Push Server – User's MDS if non-centralized push architecture • Mens (2006), pp. 5-6: users The users table holds a touple consisting of a userid (a unique user identifier such as jdoe or J9001) and a push destination pushdest. This column contains either the unique device PIN or the user's BES email address to which the push messages should be directed. Subscribers Each user in the users table can be subscribed to one or more "pages", and this table records which "pages" the user has subscribed to. As mentioned above, in my case the administrator will be updating this table, but you could easily create a web page with which users can subscribe to pages themselves. • MDS 4.1 Overview, p. 10:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 197 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Web Access Control System administrators control wireless access to resources on the corporate network. • Network authentication: The connection service supports several types of network authentication, including HTTP Basic Authentication, NT LAN Manager (NTLM), and Kerberos, so system administrators can control wireless access to web content using existing network security mechanisms. The connection service also supports Lightweight Third-Party Authentication (LTPA) if cookie storage is enabled. • Access control: System administrators can assign roles to BlackBerry devices and push initiators to control their activity through the connection service. Push access control roles enable system administrators to limit push requests from push initiators to specific BlackBerry device users. Pull access control roles enable system administrators to permit or deny access to URLs that are requested by BlackBerry device users. To the extent Plaintiff contends that Blackberry Push does not explicitly teach certificate-based security to application-level authorization for message delivery, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Rakic, Houghton, Ellison, Fok. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 198 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 1, further comprising a secure server to store a secure authorization list, the secure authorization list indicating the authorized software components and the authorized network elements that are allowed to communicate using the network server system. (e.g., the host system, which can be a network server containing a preferred list of message types that are to be redirected or preferred senders whose messages are to be redirected). *See, e.g.*, "A system and method for pushing information from a host system to a mobile data communication device upon sensing a triggering event is disclosed. A redirector program operating at the host system enables a user to continuously redirect certain user-selected data items from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. The redirector program operates in connection with event generating applications and repackaging systems at the host system to configure and detect a particular user-defined event, and then to repackage the user-selected data items in an electronic wrapper prior to pushing the data items to the mobile device. The mobile device and the host system share a common electronic address so that messages generated at either the host

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 199 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	system or the mobile data communication device are configured using the common electronic address." Lazaridis, Abstract. "A redirector program operating at the host system enables the user to redirect or mirror certain user-selected data items (or parts of data items) from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. Also operating at the host system are various sub-systems that can be configured to create triggering events, such as a screen saver sub-system or a keyboard sub-system, as well as sub-systems for repackaging the user's data items for transparent delivery to the mobile data device, such as a TCP/IP sub-system or one or more E-Mail sub-systems. Other sub-systems for creating triggering events and repackaging the user's data items could also be present at the host system. The host system also includes a primary memory store where the user's data items are normally stored." *Id.*, 2:61-3:8. "In addition to the functionality noted above, the redirector program provides a set of software-implemented control functions for determining the type of mobile data communication device and its address, for programming a preferred list of message types that are to be redirected, and for determining whether the mobile device can receive and process certain types of message attachments, such as word processor or voice attachments." *Id.*, 3:38-45. "Once an event has triggered redirection of the user data items, the host system then repackages these items in a manner that is transparent to the mobile data communication device, so that information on the mobile device appears similar to information on the user's host system. The preferred repackaging method includes wrapping the user data items in an E-mail envelope that corresponds to the address of the mobile data communication device, although, alternatively, other repackaging methods could be used with the present invention, such as special-purpose TCP/IP wrapping techniques, or other methods of wrapping the user selected data items." *Id.*, 3:66-4:9.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 200 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The primary advantage of the present invention is that it provides a system and method for triggering the continuous and real-time redirection of user-selected data items from a host system to a mobile data communication device. Other advantages of the present invention include: (1) flexibility in defining the types of user data to redirect, and in defining a preferred list of message types that are to be redirected or preferred senders whose messages are to be redirected; (2) flexibility in configuring the system to respond to numerous internal, external and networked triggering events; (3) trans- parent repackaging of the user data items in a variety of ways such that the mobile data communication device appears as though it were the host system; (4) integration with other host system components such as E-mail, TCP/IP, keyboard, screen saver, webpages and certain programs that can either create user data items or be configured to provide trigger points; and (5) the ability to operate locally on a user's desktop system or at a distance via a network server. *Id.*, 4:50-67. "By way of example, consider an E-mail sent to a user that includes three attachments--a word processing document, a video clip and an audio clip. The redirection program could be configured to send the text of the E-mail to the remote device, to send the word processing document to a networked printer located near the user, to send the video clip to a store accessible through a secure connection through the internet and to send the audio clip to the user's voice mail system." *Id.*, 6:20-28. "Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 201 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	network to initiate redirection. These are just some of the events that could be used with the present invention to initiate replication of the user-selected data items from the host system 10 to the mobile device 24." *Id.*, 7:8-30. "Also programmed into the redirector 12 is the address of the user's mobile data communication device 24, the type of device, and whether the device 24 can accept certain types of attachments, such as word processing or voice attachments." *Id.*, 7:57-60. "The redirector may also be programmed with a preferred list mode that is configured by the user either at the host system 10, or remotely from the user's mobile data communication device by transmitting a command message C. The preferred list contains a list of senders (other users) whose messages are to be redirected or a list of message characteristics that determine whether a message is to be redirected. If activated, the preferred list mode causes the redirector program 12 to operate like a filter, only redirecting certain user data items based on whether the data item was sent from a sender on the preferred list or has certain message characteristics that if present will trigger or suppress redirection of the message. In the example of FIG. 1, if desktop system 26 was operated by a user on the preferred list of host system 10, and the preferred list option was activated, then message A would be redirected. If, however, desktop 26 was operated by a user not on the host system's preferred list, then message A would not be redirected, even if the user of the host system had configured the redirector to push messages of type A The user of the host system 10 can configure the preferred list directly from the desktop system, or, alternatively, the user can then send a command message (such as C) from the mobile device 24 to the desktop system 10 to activate the preferred list mode, or to add or delete certain senders or message characteristics from the preferred list that was previously configured." *Id.*, 7:65-8:31. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 202 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "If the redirected user data item is an E-mail message, as described above, the user at the mobile device 24 sees the original subject, sender's address, destination address, carbon copy and blind carbon copy." *Id.*, 8:66-9:2. "When the user replies to this message, or when the user authors a new message, the software operating at the mobile device 24 adds a similar outer envelope to the reply message (or the new message) to cause the message to be routed first to the user's host system 10, which then removes the outer envelope and redirects the message to the final destination, such as back to computer 26. In the preferred embodiment, this results in the outgoing redirected message from the user's host system 10 being sent using the E-mail address of the host mailbox, rather than the address of the mobile device, so that it appears to the recipient of the message that the message originated from the user's desktop system 10 rather than the mobile data communication device." *Id.*, 9:2-15. "In this alternative configuration, server 11 preferably maintains a user profile for each user's desktop system 10, 26, 28, including information such as whether a particular user can have data items redirected, which types of message and information to redirect, what events will trigger redirection, the address of the users' mobile data communication device 24, the type of mobile device, and the user's preferred list, if any." *Id.*, 9:34-41.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 203 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The method steps carried out by the redirector program 12 are described in more detail in FIG. 4. The basic functions of this program are: (1) configure and setup the user-defined event trigger points that will start redirection; (2) configure the types of user data items for redirection and optionally configure a preferred list of senders whose messages are to be redirected; (3) configure the type and capabilities of the user's mobile data communication device; (4) receive messages and signals from the repackaging systems and the event generating systems; and (5) command and control the redirection of the user-selected data items to the mobile data communication device via the repackaging systems. Other functions not specifically enumerated could be integrated into this program." *Id.*, 10:39-52. "The E-Mail sub-system 44 can receive messages A from external computers on the LAN 14, or can receive messages C from some external network such as the Internet 18 or a wireless data communication network 22, and stores these messages in the primary data store 40. Assuming that the redirector 12 has been triggered to redirect messages of this type, the redirector detects the presence of any new messages and instructs the E-Mail system 44 to repackage the message by placing an outer wrapper B about the original message A (or C), and by providing the addressing information of the mobile data communication device 24 on the outer wrapper B." *Id.*, 10:58-11:2. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. "Although not shown explicitly in FIG. 4, after step 68, the program could, if operating in the preferred list mode, determine whether the sender of a particular data item is on the preferred list, and if not, then the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 204 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	program would skip over steps 70 and 72 and proceed directly back to step 56. If the sender was on the preferred list, then control would similarly pass to steps 70 and 72 for repackaging and transmission of the message from the preferred list sender." *Id.*, 12:58-65. **Combination with Rakic** Rakic and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Blackberry Push and Rakic are directed to secure push messaging protocols in the wireless communications field and a POSITA would be familiar with both Blackberry Push and device management techniques described in Rakic. Rakic, ¶¶1, 46, 60. A POSITA would be motivated to combine Blackberry Push and Rakic based on the recognized need for secure, interoperable, and standardized remote device management in mobile networks, as discussed in Rakic. *Id.*, ¶6, 41-42. Given the detailed integration of PAP protocols, utilized by Blackberry Push, with secure push messaging in Rakic, a POSITA would have had a reasonable expectation of success in achieving a robust, secure, and compatible system for wireless device management. *Id.*, ¶¶44, 59, 79. Further, POSITA would have been motivated to implement a secure-push-message server and signature generation like Rakic with the disclosure of Blackberry Push to improve security for the client device and the underlying applications by enabling the client device to (1) "authenticate the sender" and (2) "verify that an intended recipient of the secure push message is the client device, and that the client device includes a correct component." *Id.*, ¶41. Rakic, as evidenced by the example citations below, discloses a secure server to provide secure authorization signatures to the given one of the wireless end-user devices, the secure authorization signatures indicating the authorized software components that are allowed to receive data from secure message link messages via

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 205 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the message link server. *See, e.g.*, "For the client device, using the codes to validate the secure push message may protect the client device in a number of ways. For example, using the codes, the client device may authenticate the sender. In another example, using the codes may verify that an intended recipient of the secure push message is the client device, and that the client device includes a correct component. For instance, using the codes may verify that the secure push message is intended for a device that includes a specific Subscriber Identity Module (SIM) card. In the above, for the client device, using the codes in the electronic signature of the secure push message may validate the secure push message in a number of ways. In this manner, the secure push message may provide a high level of security. The format of the secure push message may be simple and flexible, and may be easily modified to accommodate different types of applications/devices that receive/send secure push messages." Rakic, ¶¶41-42. "FIG. 1 illustrates the concepts described herein. As shown, network 100 may include a client device 102, an administrator device 104, a gateway 106, and a database device 108. Depending on the implementation, network 100 may include additional, fewer, or different devices than those illustrated in FIG. 1. For example, network 100 may include multiple client devices 102. Device 102 may include any of the following devices that have the ability to or are adapted to communicate and interact with another device, such as a radiotelephone or a mobile telephone with ultra wide band or Bluetooth communication capability; a personal communications system (PCS) terminal that may combine a cellular radiotelephone with data processing, facsimile, and/or data communications capabilities; an electronic notepad, a laptop, and/or a personal computer that communicate with wireless peripherals (e.g., a wireless keyboard, speakers, etc.); a personal digital assistant (PDA) that can include a telephone; a Global Positioning System device and/or another type of positioning device; a gaming device or console; a peripheral (e.g., wireless headphone); a digital camera; or another type of computational or communication device.

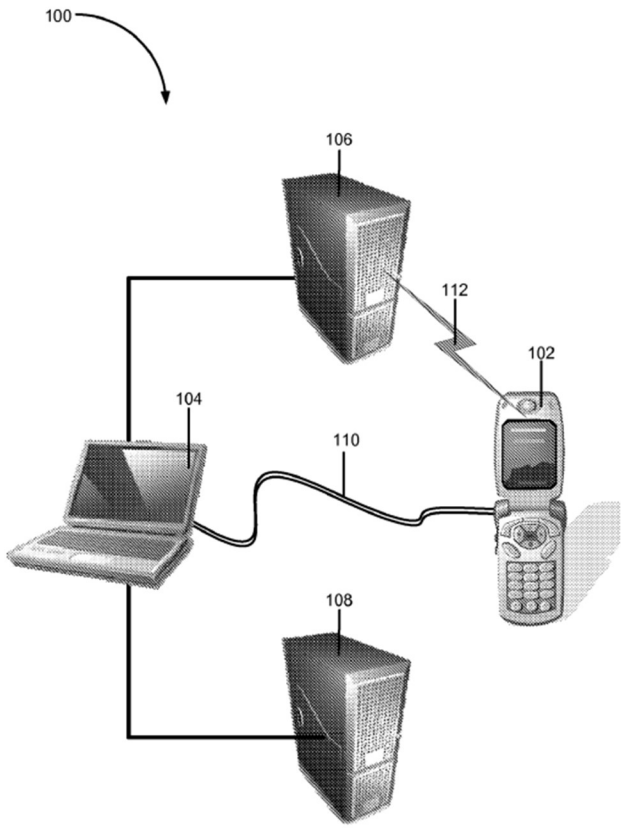
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 206 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Administrator device 104 may include any device that may be used to administer and/or control client device 102. Examples of administrator device 104 include a server, a personal computer, a laptop, and/or any other type of device with sufficient memory, processing speed, and/or bandwidth to administer/control one or more client devices 102. Gateway 106 may include a device via which client device 102 may communicate with devices in network 100. In one implementation, gateway 106 may receive push messages from administrator device 104 in accordance with the push access protocol, as specified by wireless application protocol (WAP) specification 164 (WAP-164) or WAP-247. In addition, gateway 106 may send the push messages from administrator device 104 to client device 102 in accordance with push over-the-air (OTA) protocol, as specified by WAP specification 189 (WAP-189) or WAP-235. Database device 108 may include a database of information about client device 102. In some implementations, the database may be included in administrator device 104, and in such instances, network 100 may not include a separate database device 108. In the above, administrator device 104 may configure client device 102 via a cable 110. Alternatively, administrator device 104 may configure client device 102 via wireless communications. During the configuration, administrator device 104 may obtain information that is specific to client device 102, store the information in a database on database device 108, and store an administrator code in client device 102. Once client device 102 is configured, administrator device 104 may retrieve the information related to client device 102 from database device 108, generate an electronic signature by signing a message based on the information, and combine the message with the electronic signature to produce a secure push message. Subsequently, administrator device 104 may send the secure push message to client device 102. When client device 102 receives the secure push message, client device 102 may validate the secure push message based on the electronic signature. If client device 102 is able to validate the secure push message, client device 102 may send a response to administrator device 104 to indicate that client device 102 has successfully received and/or validated the secure push message.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 207 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Depending on the implementation, client device 102 may perform a specific task in accordance with contents of the secure push message. For example, client device 102 may set parameters that are related to an application in client device 102, update firmware, reset parameters that are associated with subscription services, set security parameters, etc." *Id*., ¶¶43-51.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 208 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Fig. 1** *Id.*, Fig. 1.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 209 of 422

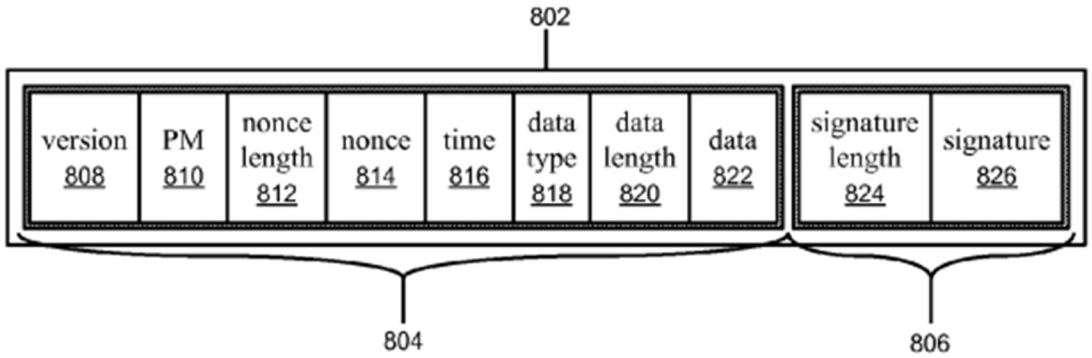
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"FIG. 5 is a functional block diagram of administrator device 104. As shown, administrator device 104 may include WAP logic 502 and a secure push message server 504. Although not illustrated in FIG. 5, administrator device 104 may include additional functional components, such as the components that are shown in FIG. 3, an operating system (e.g., Windows, Linux, Mac OS™, etc.), an application (e.g., an email server, an file transfer protocol (FTP) server, etc.), a database, logic for wireless or wire-line communication protocols other than the WAP, etc. WAP logic 502 may process messages that are sent or received over wireless communication links in accordance with the WAP. For example, when administrator device 104 and gateway 106 communicate through a wire-line network, administrator device 104 and gateway 106 may exchange messages in accordance with WAP-189 or WAP-235. Secure push message server 504 may send secure push messages to client device 102 via a wireless network or gateway 106. Given a message or a payload from an application, a user, or an administrator, secure push message server 504 may generate a key, an electronic signature, and a secure push message. Secure push message server 504 may generate the key based on codes (e.g., IMEI) that are obtained from client device 102, a component in client device 102 (e.g., removable memory 218), and an administrator. Depending on the implementation, additional codes or information may be used to generate the key. Once the key is generated, secure push message server 504 may generate an electronic signature based on the key and the message. In possession of the electronic signature and the key, secure push message server 504 may generate and send the secure push message to client device 102. In one implementation, secure push message server 504 may generate the secure push message by appending the electronic signature to the message and/or other information (e.g., a timestamp). In addition to generating/sending secure push messages to client device 102, secure push message server 504 may obtain and store information that is needed to generate the key. In one implementation, while configuring client device 102, secure push message server 504 may obtain a client device identifier from

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 210 of 422

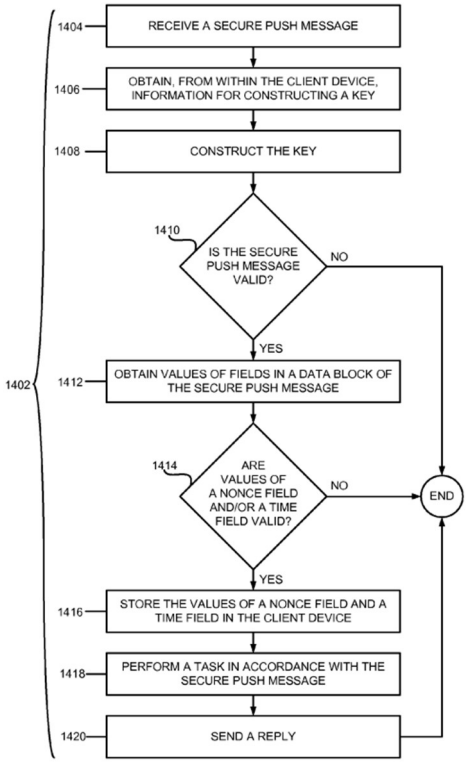
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	client device 102, subscriber information from removable memory 218 that is not yet installed in client device 102, and an administrator code from an administrator. Furthermore, secure push message server 504 may store the client device identifier, the information, and the administrator code in a database." *Id*., ¶¶63-68. "FIG. 14 is a flow diagram of an exemplary process 1402 for sending a secure push message. Assume that an administrator has configured client device 102 in accordance with process 902, and that administrator device 104 has created and sent a secure push message to client device 102. In one implementation, the secure push message may have been sent in accordance with a WAP. Process 1402 may begin at 1404, where the secure push message may be received (block 1404). For example, client device 102 may receive a secure push message that is sent from administrator device 104 at block 1318. Information for constructing a key may be obtained from within client device 102 (block 1406). In one implementation, secure push message client 404 may retrieve the administrator code, which has been stored in client device 102 at block 1108 (FIG. 11), an IMEI (e.g., a client device identifier) of client device 102, and the subscriber information from a component (e.g., SIM card 218 or 1002) within client device 102. The key may be constructed (block 1408). In some implementations, secure push message client 404 may construct the key by combining the administrator code, the subscriber information, and the client identifier. In one implementation, secure push message client 404 may construct the key in accordance with expression (3). It may be determined whether the secure push message is valid (block 1410). For example, to validate the secure push message, secure push message client 404 may obtain data block 804 and signature block 806 from the secure push message that is received at block 1404, sign obtained data block 804 based on the key constructed at block 1408, and compare the value of signature field 826 in signature block 804 to the signed data block 804. If the signed data block 804 is identical to the value of signature field 826, secure push message client 404 may determine that the secure push message is valid. Otherwise, secure push message

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	client 404 may determine that the secure push message is invalid. In the preceding, because the key is constructed based on information that could have been obtained only through a physical contact with client device 102 (e.g., the administrator code, IMSI, IMEI, etc.), validating the secure push message may indicate the authenticity of the sender. At block 1410, if the secure push message is determined as a valid message, process 1402 may proceed to block 1412. Otherwise, process 1402 may terminate. Values of fields 808-822 in data block 804 of the secure push message may be obtained (block 1412). In one implementation, secure push message client 404 may obtain each of the values in version field 808, PM field 810, nonce length field 812, nonce field 814, time field 816, data type field 818, data length field 820, and data field 822 from data block 804 in the secure push message. It may be determined whether the values of nonce field 814 and time field 816 are valid (block 1414). For example, secure push message client 404 may compare the nonce and the timestamp in nonce field 814 and the time field 816 to values of previous nonce field/time field values that are stored in client device 102. If the nonce and the timestamp matches any of the previous nonce field/time field values, the nonce and the timestamp may be determined as being invalid. That is, the same nonce and/or timestamp may indicate that the message is a duplicate message. By checking the nonce and the timestamp, secure push message client 404 may protect client device 102 against replay attacks or other types of attacks. If the value of nonce field 814 and/or time field 816 is valid, process 1402 may proceed to block 1416. Otherwise, process 1402 may terminate. At block 1416, the values of nonce field 814 and time field 816 may be stored in client device 102 (block 1416). Secure push message client 404 may later use the stored values to determine the validity of nonces/timestamps when client device 102 receives secure push messages. A task may be performed in accordance with the secure push message (block 1418). In one implementation, based on the value of data field 822, secure push message client 404 may perform a specific task. For instance, assume that the value of data type field 818 is XML_DATA_TYPE, and that the value of data field

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 212 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	822 is obtained from expression (5) illustrated above. Based on the values "set incoming mail server domain name" and "pop3.talktome.com" within <name/> and <data/> tags, secure push message client 404 may set the domain name of incoming mail server to pop3.talktome.com. In some instances, the data in data field 822 may include raw data. In such instances, secure push message client 404 may avoid memory intensive or processing intensive operations that are associated with scanning formatted data. A reply may be sent (block 1420). In one example, secure push message client 404 may send the value of nonce field 814 and a status code (e.g., a code to indicate whether the task has been successfully complete, whether the secure push message is valid, etc.) to administrator device 104."*Id.*, ¶¶108-21.  **Fig. 8** *Id.*, Fig. 8.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 213 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Fig. 14** *Id.*, Fig. 14.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 214 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Houghton Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, a secure server to provide secure authorization

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 215 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	signatures to the given one of the wireless end-user devices, the secure authorization signatures indicating the authorized software components that are allowed to receive data from secure message link messages via the message link server. *See, e.g.*, "When a remote program such as a push server 202 sitting inside the operator's firewall 201 and with permission to contact ports on the mobile terminal 203 sends an IP packet to the terminal (A), the IP terminal software 204 routes the packet with [its] commands or application-specific data to the mobile application 205 (B), launching the application if it is not already running." Houghton, p. 8. "Once such an IP connection is established between the server and one or more mobile terminals, the connection can be used to transfer messages to the mobile terminal. The most interesting such messages for the present invention are event trigger messages to launch or command a mobile application." *Id*., p. 9. "In an embodiment of the invention, the push messages may include, for example, a message stimulating, triggering or commanding the launch, display or shift to foreground of a user interface feature of the terminal such as an application which is part of the basic terminal software or and an additional installed application beyond the base software supplied with the mobile terminal." *Id*., p. 11. "In response to receiving such a message, a mobile terminal may spontaneously open the application, dynamically loaded module or user interface feature of the terminal thereof to a designed application view or location, present new information of the availability of new information, a new feature or operating state. If such application, module or user interface features is already operating in the mobile terminal, the action of such a message would instead stimulate the application or user interface feature to present itself to the foreground of the user interface and change state as specified in the application-specific message received from the server." *Id*., p. 12. "More specifically, Figure 4 illustrates a Push Client basic functionality 405 implemented in a mobile terminal 404 as part of a message passing system in cooperation with a Push Server 401. The push client 405 is preferably implemented in the form of a software run in a processor of the mobile terminal 404." *Id*., p. 16.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 216 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The push client software 405 preferably launches automatically when the mobile terminal 404 is turned on. The push client software 405 runs in the background and attempts to log into the push server 401 over a data network 403, normally a wireless network capable of transmitting Internet Protocol (IP) packets. The arrow A in Figure 4 illustrates an optional attempt by the push client 405 to communicate with the wireless network firewall 402 or other similar security or gateway device in order to open a connection for the push server 401 to directly contact the push client 401 as in FIG 3. If successful, operations can continue as described in FIG 3 or as below. Arrow B illustrates a login attempt." *Id*., p. 17. "As an example, such network parameters may include but are not limited to one or more of the following: - Use of a connection-oriented protocol suitable to the wireless network such as TCP/IP protocol, which is generally preferred in the common situation where an intermediate firewall, Network Address Translation (NAT) or gateway device 402. - Use of a connectionless protocol suitable to the wireless network 403, such as UDP/IP, which is generally preferred when no network device restricting terminal contact 402 is present as this reduces resource usage on the server when large number of mobile terminals is served. - Use of an alternate connection-oriented protocol such as HTTP suitable for mobile push operation through more strict firewalls and proxy Internet connections. - Use of a secure protocol such as HTTPS, IP-Sec, secure IP6 or a proprietary security protocol to identify the communicating parties, prevent message interception by 3rd parties, and prevent message modification by 3rd parties." *Id*., p. 19. "At any time after a connection is established between the client 405 and the server 401, the client may push a message to the server or the server may send a push message to the client. This is illustrated by means of the arrow C in Figure 4. These messages through a previously established, connection-oriented protocol such as TCP/IP, SSL, HTTP or HTTPS arrive quite rapidly, more rapidly than if the connection where established only when needed. The associated protocol overhead such as 3-way handshake during connection establishment thus occurs before the actual need for message transmission rather than needing to occur at time or transmission and thus delaying the transmission." *Id*., p. 20.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 217 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"A connectionless protocol, such as UDP/IP, may also used in cases where the connection-oriented protocol is not possible or not desirable because of the push server resources used by every static instance of a connection-oriented protocol. Connectionless communication is also used when a message needs to get through to the server 401 quickly, such as when occurs when the client 405 notifies the server 401 of an terminal event, user interface event or application event but the connection-oriented protocol connection is not connected or is already busy transferring information" *Id*., p. 20. "In the preferred embodiment of the invention, the most basic message pushed from the server 401 to the client 405 is a data packet containing no information. Additional binary or text information specifying which mobile application 406 from a plurality of such applications may optionally be included. Additional binary or text information to be passed to the default or specified mobile application may optionally be included. Additional binary or text information specifying how upon receiving such a message the client 405 will notify and optionally ask the user if the action should be performed may optionally be included." *Id*., p. 21. "The push client 405 may in turn at any time accept data D from each of a plurality of mobile applications 406 and direct such to the push server 401, possibly for redirection through push server API to other servers, mobile terminals, push clients and computing devices." *Id*. "Upon receiving such a message, appropriate automated actions and user interface media capabilities of the mobile terminal as specified in the command C from the server 401 are executed by the push client 405. The most common case is playing a sound, asking the user if the mobile application should be launched, and if affirmative passing the command details (arrow D) to the specified mobile application, module or additional feature within the same application suite, launching the application, module or feature if needed. Such action may include optionally proceed without audio, with network downloaded audio, or without first requesting confirmation from the user as specified by user preferences and command details." *Id*. "Figure 7 illustrates an example of an 'COMMAND PUSH' procedure wherein the push server 701 is triggered by a trigger event as for example from an application server 702 to push an application command message to the push client 704 and thereby initiate a mobile terminal client trigger event in a mobile application 705 from a plurality of such applications, program modules and user interface features on the

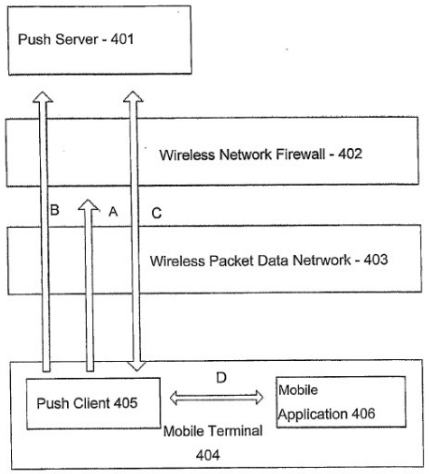
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 218 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	terminal 705. The trigger event from push client 704 to mobile application 705 may optionally include commands or data affecting the aspect of visual presentation of mobile application 705." *Id*., pp. 21-22. "The trigger mechanism may be between separate parts of a larger application or application suite, or may be an IP-triggered application launch and data transfer as in the case of mobile application launched by contact to a specified IP port." *Id*., p. 22. "The potential applications of the COMMAND PUSH include but are not limited to mobile applications for enterprise connectivity, back office integration, database access, scientific and industrial process and equipment monitoring and control, pictures transfer, mobile gaming, application data transfer, mobile electronic commerce , application alarms, reminders, multimedia interactivity, corporate calendaring, personal messaging and group messaging, etc." *Id*. "Figure 8 illustrates an extension of the procedure in Figure 7, one known as 'COMMAND PUSH WITH BIDIRECTIONAL DATA'. This embodiment of the invention includes all aspects of 'COMMAND PUSH' with the additional step that a data connection between application server 802 and mobile application 805 is established with the aid of the persistent managed, tested and configured data connection in Figure 4, arrow C between push server 401 /801 and push client 405/804. An IP or other API (application programming interface) connection between application server 802 and push server 801 simplifies the work and cost of creating application server 802 by taking care of network details already established for the COMMAND PUSH procedure." *Id*., 23. "Then at some later time the push server 901 is triggered by a trigger event of which the application server 902 is aware to send push message to the mobile terminal 905, and procedure steps A, B, C are traversed thereby pushing application launch commands, lazy application updates, and lazy application data updates through push clients 904 and 905 to mobile application 907 through local command and data path D (also Figure 5, arrow A)." *Id*., 24. "A method of pushing commands and data to a wireless terminal from a push server located outside a wireless system serving the wireless terminal, comprising the push client initiates and maintains over an

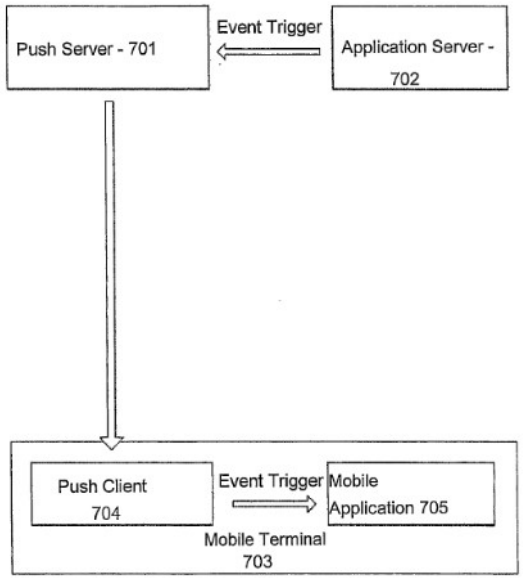
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 219 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	extended period of time a packet-based data connection through at least one communication security or inter-network connection device of the serving wireless system to the push server using Internet technologies; the push server sends, in response to predetermined trigger events, at any time push application installation, updates, commands or data messages to be directed to a push application or dynamically loaded module or user interface feature from among a plurality of such applications, modules and features by means of said client-initiated or client-opened and client-maintained permanent connection, whereby avoiding blocking of said server-originated push messages by the security device, and launching said application, application module or dynamically loaded module or user interface feature as needed, the push client, in response to receiving a push message containing application installation, updates, application commands or data from the push server, forwards such application commands or data or stores such application installation or updates within the mobile terminal to a mobile application from among a plurality of such applications so as to initiate launch optionally to a specified state or if already running initiate or change presentation to the user according to the push message." *Id*., claim 1. "A method as claimed in any one of claims 1-32, wherein the push client and push server establish a secure connection and unwrap such security on behalf of the mobile application." *Id*., claim 33. "A server comprising means for performing the server steps as claimed in any one of method claims 1-33." *Id*., claim 34.

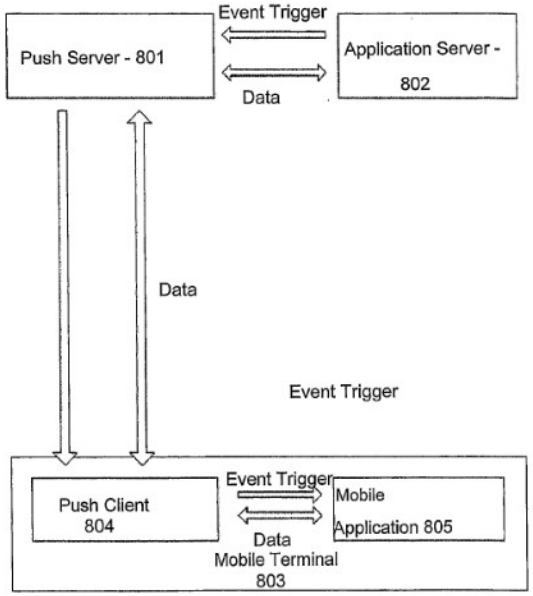
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 220 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 221 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 7  *Id.*, Fig. 7.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 222 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 8  *Id.*, Fig. 8.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 223 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Ellison Ellison and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Ellison addresses the need for robust processor security in the context of modern computing, especially for applications such as e-commerce and business-to-business (B2B) transactions, which are increasingly conducted over networked systems. Ellison, Background. Blackberry Push enables secure, efficient communication and content delivery to mobile and wireless devices. Since Ellison's architecture is intended to be broadly compatible and to protect platforms engaged in e-commerce and B2B transactions, a POSITA would find it obvious to secure communications over all relevant channels—including wireless networks. Blackberry Push was a widely adopted system for secure communication with mobile devices. No inventive skill would be required for a POSITA integrating Blackberry Push with Ellison's isolated execution architecture in order to extend the security benefits of Ellison's invention to mobile and wireless environments. Ellison discloses processors are vulnerable to malicious software attacks. Ellison, Abstract, 1:16-51. To inhibit this, Ellison isolates different software elements using a hierarchy of abstract rings and "nubs" to create an "isolated execution mode" (secure-execution environment, red) in which hardware/software access "is restricted" compared to a "normal execution mode" (outside of the secure-execution environment, green) that "operates in a non-secure environment." *Id.*, 4:65-5:1, 6:1-26, 8:25-32, Figs. 1A-1C. Ellison controls access using computer-generated keys. *Id.*, 8:66-9:40, 9:47-62, Fig. 2. A POSITA would have been motivated to utilize Ellison's hierarchical, nub-based normal and isolated execution environments to provide enhanced security for Blackberry Push clients. This would have involved using a known technique (isolation environments) to improve similar devices (BES networks) in the same way (securing content). Ellison, as evidenced by the example citations below, discloses a secure server to provide secure

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 224 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	authorization signatures (e.g., signatures of the subset of the software environment) to the given one of the wireless end-user devices, the secure authorization signatures indicating the authorized software components that are allowed to receive data from secure message link messages via the message link server (e.g., the signature being used to verify the subset). *See, e.g.*, "In the digital signature generation process, the protected private key 204 is used as an encryption key to encrypt a digest of a message producing a signature, and the public key 205 is used as a decryption key to decrypt the signature, revealing the digest value." Ellison, 8:60-65. "FIG. 3C is a diagram illustrating the usage protector 250 shown in FIG. 2 according to one embodiment of the invention. The usage protector 250 includes a decryptor 325, a signature generator 320, a storage medium 322, and a signature verifier 330." *Id*., 10: 63-67. "The decryptor 325 accepts the OS nub's encrypted private key (i.e., protected) 204, and decrypts it using the OSNK 203, exposing the private key 328 for use in the isolated environment. The signature generator 320 generates a signature 304 for the subset 230 using the private key 328. It is noted that the subset 230 may be compressed before input into the signature generator 320 to generate the signature 304. The signature algorithm used by the signature generator 320 may be public-key digital signature algorithm which makes use of a secret private key to generate the signature, and a public key to verify the signature. Example algorithms include ElGama, Schnorr and Digital Signature Algorithms schemes just to name a few. In one embodiment, the generation of the signature 304 includes hashing the subset 230 to generate a before hash value, which is then encrypted using the private key 328 to generate the signature 304. The signature 304 is then saved in a storage medium 322. At a later time, the signature is retrieved from the storage 322, and the retrieved signature 306 is used, along with public key 205, by the signature verifier 330 to verify the subset 230. The signature verifier 330 verifies whether the subset 230 has been modified, producing a modified/not-modified indicator 331. In one embodiment, the verification process includes decrypting the retrieved signature 306 using the public key 205 to expose the before hash value. The subset 230 is hashed to generate an after hash value. The before hash value is compared to the after hash value to detect whether the subset 230 has been modified. If the two hash values match, the subset 230 is the same as it was when the signature

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 225 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	was generated." *Id*., 11:1-30. "FIG. 3D is a diagram illustrating the usage protector 250 according to yet another embodiment of the invention. The usage protector 250 includes a decryptor 345, a manifest generator 335, a signature generator 340, a storage medium 349, a signature verifier 350, and a manifest verifier 355." *Id*., 11:31-35. "The manifest 307 is stored in a storage medium 349 for later use. The manifest 307 is also input to the signature generator 340 to generate a signature 308 over the manifest 307 using the private key 348. The generated signature 308 is also stored in a storage medium 349. At a later time, we desire to verify that the portions of the subset 230 described by the manifest have not changed. This requires verifying that the manifest itself has not been changed, and that each group in subset 230 described by the manifest has not been changed. The stored signature and manifest are retrieved from the storage medium 349. The retrieved signature 309, and the retrieved manifest 354, along with the public key 205, are used by the signature verifier 350 to test that the retrieved manifest 354 is unchanged from the original manifest 307. The signature verifier 350 produces a signature-verified flag 351, which is asserted only if the signature verifies that the manifest is unchanged." *Id*., 11:51-12:1. "In one embodiment, the verification process includes decrypting the retrieved signature 309 using the public key 205 to expose the before hash value. The retrieved manifest 354 is hashed to generate an after hash value. The before hash value is compared to the after hash value to detect whether the retrieved manifest 354 has been modified. If the two hash values match, the retrieved manifest 354 is the same as it was when the signature was generated. The retrieved manifest 354 is also supplied to the manifest verifier 355, which uses the descriptive information in the retrieved manifest 354 to selectively verify portions of subset 230. In a typical embodiment, this involves hashing each group in subset 230, where the group is identified by information in the retrieved manifest 354, and comparing the newly generated hash value against the hash value for the group stored in the retrieved manifest 354. The manifest verifier 355 produces a manifest-verified flag 356, which is asserted if all entries described by the retrieved manifest 354 are verified as unchanged. If both the manifest-verified flag and the signature-verified flag are asserted, then the overall verification process passes, and the selected portions of subset 230 are known to be the same as when the

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	signed manifest was originally generated, and a pass/fail flag 358 is asserted. Note that the signature verifier 350 and the manifest verifier 355 can be invoked in any order." *Id*., 12:1-26. "Upon START, the process 600 checks to see whether the subset is to be updated or to be tested (Block 605). If it is to be updated, the process 600 checks to see whether the request to be updated is authorized (Block 610). If it is not authorized, the process 600 is terminated. Otherwise, the process 600 obtains the OSNK, the protected private key and the subset (Block 615). The process 600 then decrypts the protected private using the OSNK (Block 620) and signs the subset using the private key (Block 625). The process 600 stores the signature in a storage (Block 630) and is terminated. If the subset is to be tested, the process 600 obtains the public key, the subset, and the signature (Block 650). The process 600 verifies the subset against the signature using the public key (Block 655). The process 600 then checks whether the subset is verified (Block 660). If the subset is verified, the process 600 clears 'modified' flag (Block 665) and is terminated. Otherwise, the process 600 sets "modified" flag (Block 670) and is terminated." *Id*., 13:46-64. "The apparatus of claim 1 wherein the usage protector comprises: a decryptor to decrypt a protected private key to generate a private key using the OSNK; a signature generator coupled to the decryptor to generate a signature of the subset of the software environment using the private key, the signature being stored in a storage; and a signature verifier to verify the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/not modified flag indicating whether the subset has been modified." *Id*., claim 6. "The apparatus of claim 1 wherein the usage protector comprises: a manifest generator to generate a manifest of the subset of the software environment, the manifest describing the subset of the software environment, the manifest being stored in a storage; a signature generator coupled to the manifest generator to generate a manifest signature using a private key, the private key being decrypted by a decryptor using the OSNK, the manifest signature being stored in the storage; a signature verifier to verify the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage; and a manifest verifier to verify the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested at a test center, the

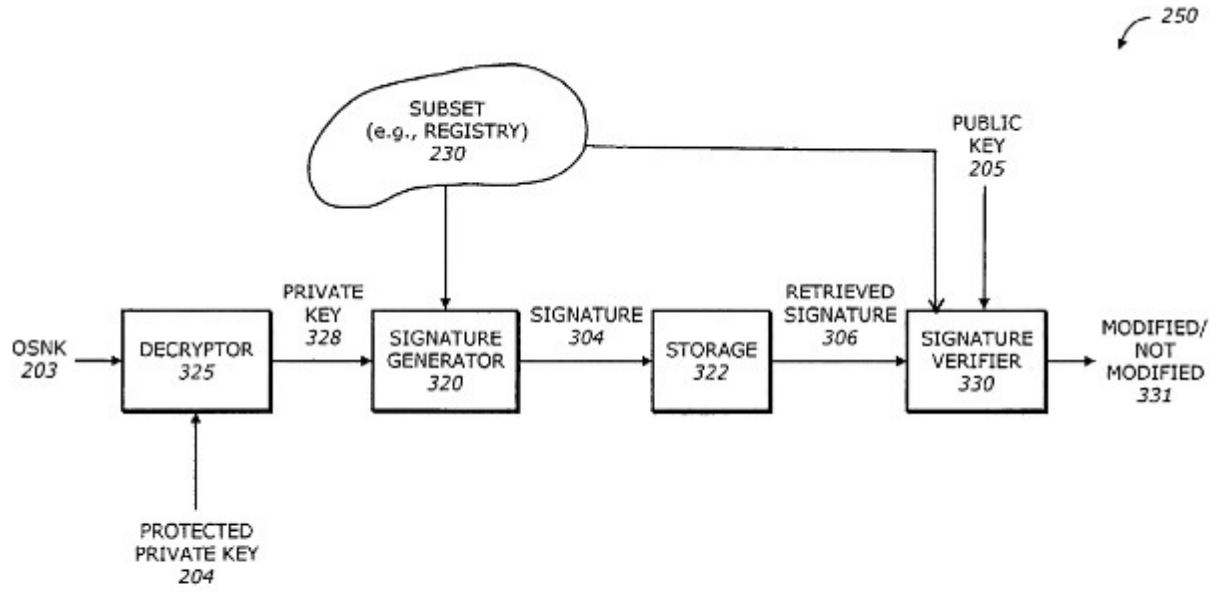
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 227 of 422

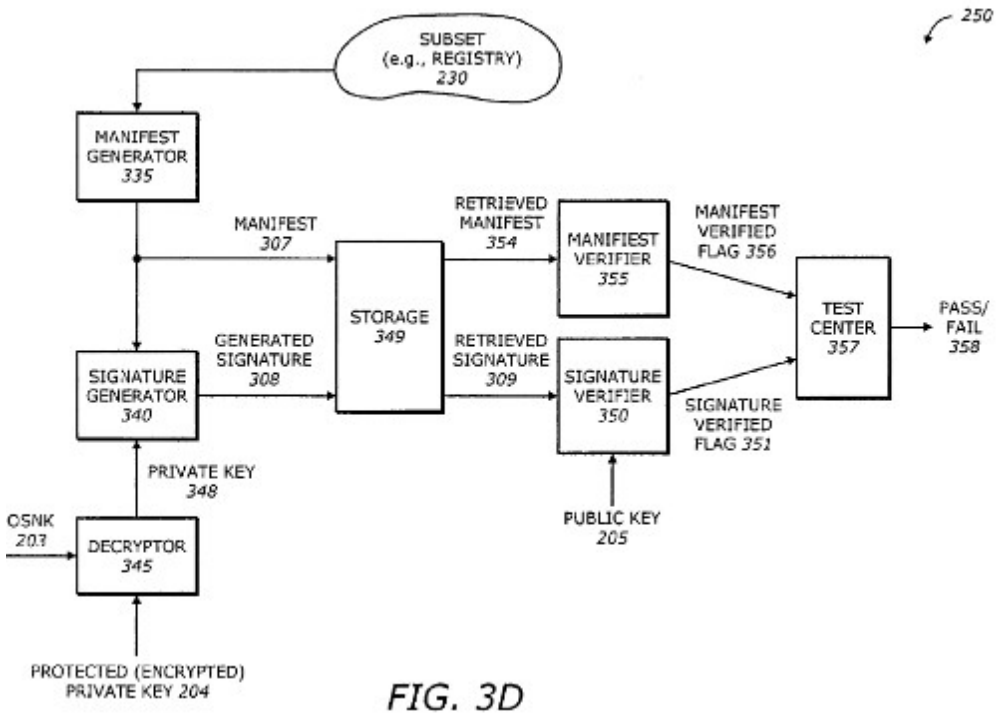
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	test center generating a pass/fail signal to indicate whether the subset has been modified." *Id*., claim 7. "The method of claim 10 wherein protecting usage comprises: decrypting a protected private key to generate a private key using the OSNK; generating a signature of the subset of the software environment using the private key, the signature being stored in a storage; and verifying the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/not modified flag indicating whether the subset of the software environment has been modified." *Id*., claim 15. "The method of claim 10 wherein detecting comprises: generating a manifest of the subset of the software environment, the manifest describing the subset of the software environment, the manifest being stored in a storage; generating a manifest signature of the manifest using a private key, the private key being decrypted using the OSNK, the manifest signature being stored in the storage; verifying the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage; and verifying the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested at a test center, the test center generating a pass/fail signal, the pass/fail signal indicating whether the subset of the software environment has been modified." *Id*., claim 16. "The computer program product of claim 19 wherein the computer readable program code for protecting usage comprises: computer readable program code for decrypting a protected private key to generate a private key using the OSNK; computer readable program code for generating a signature of the subset of the software environment using the private key, the signature being stored in a storage; and computer readable program code for verifying the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/not modified flag indicating whether the software environment has been modified." *Id*., claim 24. "The computer program product of claim 19 wherein the computer readable program code for protecting usage comprises: computer readable program code for generating a manifest of the subset of the software environment, the manifest being stored in a storage; computer readable program code for generating a manifest signature of the manifest using a private key, the private key being decrypted using the OSNK, the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 228 of 422

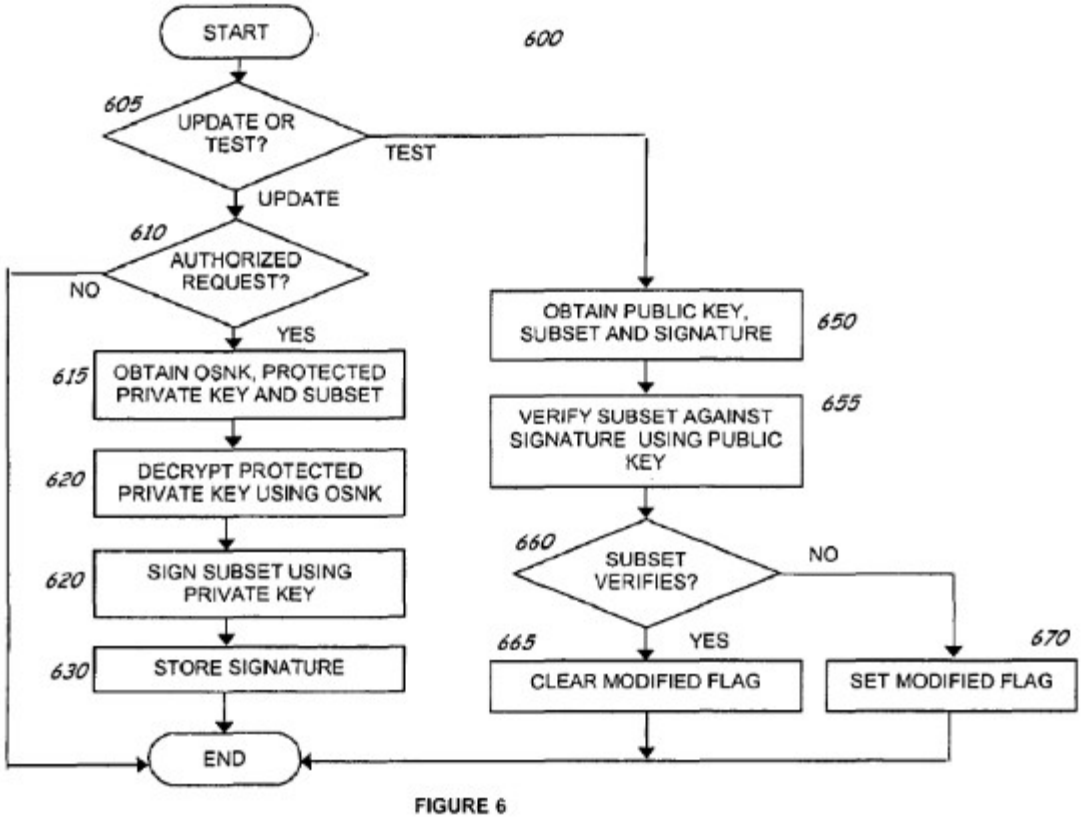
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	manifest signature being stored in the storage; computer readable program code for verifying the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage; and computer readable program code for verifying the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested at a test center, the test center generating a pass/fail signal, the pass/fail signal indicating whether the subset of the software environment has been modified." *Id*., claim 25. "The system of claim 28 wherein the usage protector comprises: a decryptor to decrypt a protected private key to generate a private key using the OSNK; a signature generator coupled to the decryptor to generate a signature of the subset of the software environment using the private key, the signature being stored in a storage; and a signature verifier to verify the signature to generate a modified/not modified flag using a public key, the signature being retrieved from the storage, the modified/not modified flag indicating whether the subset of the software environment has been modified." *Id*., claim 33. "The system of claim 28 wherein the usage protector comprises: a manifest generator to generate a manifest of the subset of the software environment, the manifest describing the subset of the software environment, the manifest being stored in a storage; a signature generator coupled to the manifest generator to generate a manifest signature of the manifest using a private key, the private key being decrypted using the OSNK, the manifest signature being stored in the storage; a signature verifier to verify the manifest signature to generate a signature verified flag using a public key, the manifest signature being retrieved from the storage; and a manifest verifier to verify the manifest to generate a manifest verified flag, the manifest being retrieved from the storage, the manifest verified flag and the signature verified flag being tested by a test center, the test center generating a pass/fail signal indicating whether the subset has been modified." *Id*., claim 34.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 229 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 3C *Id.*, Fig. 3C.

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 3D *Id.*, Fig. 3D.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 231 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIGURE 6 *Id.*, Fig. 6.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 232 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Fok Fok and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Blackberry Push and Fok are from the same field in that they disclose buffering and transmitting data using secured TCP/IP protocols. Fok, ¶¶9, 37, 56, 71, Figs. 2-3. Fok describes "secure inter-application communication" in a "mobile operating environment." *Id*., ¶¶36-37, 81, 106-108, Figs. 4, 8-9. Fok uses a "handshake" between a "primary" and one or more "recipient" applications to establish recipients are trusted using "a list of unique identifiers" before transmitting information between the two. *Id*., ¶¶47-53, 96-102, 106-108, Figs. 7-10. A POSITA would have been motivated to combine Blackberry Push and Fok to provide a "secure inter-application communication" like Fok's so that two or more applications residing on a single UE/wireless device may transmit application-specific MMs to one another through the MMS User Agent. Fok, as evidenced by the example citations below, discloses the message link server of claim 1, further comprising a secure server to provide secure authorization signatures to the given one of the wireless end-user devices (e.g., the unique identifiers), the secure authorization signatures indicating the authorized software components that are allowed to receive data from secure message link messages via the message link server (e.g., referencing the unique identifiers against a list of trusted applications). *See e.g.*, "Providing for secure and efficient communication for mobile applications executed in a mobile operating environment is described herein. As an example, a primary mobile application can initiate a handshake that includes a unique identifier of the primary application and a random number for signing and/or certifying responsive requests. A recipient application can reference the unique identifier with a list of certified primary applications to Verify the primary application. If verified, the recipient responds with the random number and a second random number that can sign and/or certify data requests sent by the primary application. According to some embodiments, random numbers can be hashed and/or truncated to provide low power encryption for such numbers. Further, round-trip policies can be enforced to provide reliable transmission

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 233 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	of data. Accordingly, reliable, secure and low power synchronous communication can be conducted in a mobile environment." Fok, Abstract. "According to other aspects, disclosed is a method for providing transmission of data between applications in a mobile operating environment. The method can include initiating a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake and receiving a response to the handshake that includes a first number and a second number, the second number is randomly generated by the mobile application. The method can further include verifying that the first number is the random number generated for the handshake and sending a data event to the mobile application that includes an increment of the second number if the first number is verified as the random number. Moreover, the method can also include waiting for a data event response from the mobile application that includes at least the increment of the second number prior to sending a second data event." *Id.*, ¶10. "According to one or more other aspects, described is a method for providing data transmission between applications in a mobile operating environment. The method can include receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application and referencing the unique ID against a list of trusted primary application identifiers. In addition, the method can include sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and receiving a data event request. Additionally, the method can further include responding to the data event request prior to receiving additional data event requests. According to additional aspects, disclosed is an apparatus that provides data transmission between applications in a mobile operating environment. The apparatus can include a security module configured to receive a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application. Further, the apparatus can include a reference module configure to verify the unique ID against a list of trusted primary application identifiers and an interface module configured to send a handshake response to the primary application if the unique ID

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 234 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	matches an ID of the list of trusted primary application identifiers and configured to receive a data event request. In addition to the foregoing, the apparatus can also include a management module configured require the interface to respond to the data event request prior to receiving additional data event requests. According to still other aspects, at least one processor that provides data transmission between applications in a mobile operating environment is described. The processor(s) can include a first module that receives a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application and a second module that references the unique ID against a list of trusted primary application identifiers. Further, the processor(s) can include a third module that sends a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and a fourth module that receives a data event request. Additionally, the processor(s) can include a fifth module that responds to the data event request prior to receiving additional data event requests. According to one or more additional aspects, an apparatus that provides data transmission between applications in a mobile operating environment. The apparatus can include means for receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application and means for referencing the unique ID against a list of trusted primary application identifiers. Further, the apparatus can include means for sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and means for receiving a data event request. Moreover, the apparatus can also include means for delaying receipt of additional data event requests until a response to the data event request is received." *Id*., ¶¶15-18. "A handshake can be initiated by the primary application 102 by sending a handshake request that includes at least a unique identifier of the primary application 102 or a random number generated for the handshake, or both. The unique identifier is an application ID associated with the primary application, that can be utilized by a recipient application to verify that the primary application is to be trusted. Typically, an operating system (106) can require certain standards be met for applications designed to execute on the operating system (106). One such requirement can be to incorporate a list of unique identifiers of trusted

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 235 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	primary applications. A recipient application 104 having such a list could then verify that a received handshake request is sent by an application authorized to make the request. The recipient (104) could then trust and engage in communication with such application. In addition to a unique identifier of a primary application, a handshake request can also include a random number (e.g., 32 bit, 16 bit, 64 bit, and so on) generated for the handshake. The random number can be sent to an intended recipient application and incorporated into responses sent to the primary application 104 by the recipient application 104. As a result, a response that includes the random number (or, e.g., a predictable variation thereof, discussed below) can be identified as originating from the intended recipient. According to additional aspects, the random number can be hashed utilizing a particular hash function (e.g., SHA-1, SHA-2, or the like) and/or truncated for transmission over a particular communication interface (106). A hashed random number can be much more difficult to decrypt, providing significant protection from spoofing attacks. Truncation can be utilized to transmit the random number utilizing various protocols and/or communication interfaces (106) that support different size numbers. For instance, if a generated random number identifying a handshake session is 32-bit, and a communication interface with a recipient application (104) enables only 16-bit data transfer, the 32-bit random number can be hashed and then truncated to 16-bit for transmission. The recipient can then hash and/or truncate the random number and compare it to the received hashed and/or truncated number to verify whether a sender of communication is the initiator (102) of the handshake. Recipient application 104 can receive a handshake request and verify whether a sender (102) is a trusted primary application. Particularly, a unique identifier can be extracted from the handshake request and compared with a list of trusted primary application identifiers. If the extracted identifier matches an identifier on the list, a handshake response can be sent to the sender (102), establishing a secure channel between sender (102) and recipient (104). If the extracted identifier does not match an identifier on the list, the handshake and any subsequent communication can be ignored until a verified handshake is received Additionally, the recipient application 104 can extract and store a random number included within the handshake request. As a specific example, the random number can be stored in temporary memory (not

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 236 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	depicted) to make it more difficult for malicious programs to extract the random number and spoof the recipient application's (104) identity. Upon verifying the unique identifier included within the handshake request, discussed above, the recipient application 104 can send a handshake response that includes the random number. Particularly, by including the random number the recipient application 104 can identify itself to the primary application 102. According to particular aspects, the recipient application 104 can hash and/or truncate the random number to make spoofing the recipient application's (104) identity (e.g., to the primary application 102 that generated and therefore is configured to trust the random number) more difficult and/or make the hashed number more compatible with particular communication interfaces (106), as discussed above." *Id*., ¶¶48-51. "Recipient application 302 can include at least a security module 304 configured to receive a handshake request from a primary application (not depicted, but see 202 at FIG. 2, supra). The handshake request can include a unique ID of the primary application and/or a random number generated by the primary application. Additionally, the recipient application 302 can include a reference module 306 configured to verify the unique ID of the primary application. For instance, the reference module can compare the unique ID to a list of trusted primary application identifiers 312. Such list (312) can include all unique IDs associated with applications that the recipient application 302 should trust and accept a handshake from. Thus, a primary application's unique ID can be verified by reference to the list 312." *Id*., ¶73. "Method 700, at 702, can initiate a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake. The primary application can be configured to communicate via a mobile operating system (e.g., BREW operating or the like) with the mobile application, utilizing a message passing architecture, or shared memory, or a suitable combination thereof or of like data transfer platforms. The unique identifier specified in the handshake can be utilized to identify the primary application as a trusted application. The random number, on the other hand, can be utilized by the mobile application to identify itself to the primary application. Consequently, communication received at the primary application that contains the random number (and/or, e.g., an appropriate increment thereof) can be associated with the mobile application. As a corollary to the foregoing, if a communication received at the primary application does not include the random number (and/or, e.g.,

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	an appropriate increment thereof), the communication can be assumed unassociated with the handshake and therefore not secure." *Id*., ¶96. "FIG. 10 illustrates a sample methodology 1000 for responding to secure communication between mobile applications in a mobile operating environment. At 1002, a handshake request can be received. The handshake request can include a unique identifier of an initiating application and/or a random number identifying the handshake, as described herein. At 1004, the unique identifier, if received, can be referenced against a list of trusted primary application IDs. If a match between a received unique identifier is found, method 900 can proceed to 1006. If not, method 900 can terminate (not depicted). At 1006, a response to the handshake can be sent if the unique ID is verified against the list of trusted applications. At 1008, a data event request can be received. The data event request can include, for instance, a second random number included within the handshake response to identify subsequent incoming communication associated with the handshake session (e.g., where such session is established by the handshake request received at reference number 1002 and handshake response sent at reference number 1006). At 1010, receipt of an additional data event request can be delayed at least until the data event request is responded to (e.g., a response to the data event request is sent prior to receiving additional data event requests). Accordingly, method 1000 can impose round-trip communication policies for secure communication, incorporating random numbers generated by initiating and recipient applications. Additionally, such random numbers can be hashed and truncated to provide additional security with low processing requirements and power cost. Therefore, method 1000 provides a mechanism to respond and participate to secure inter-application communication particularly beneficial for a mobile operating environment." *Id*., ¶109. "FIG. 12 illustrates an example methodology 1200 for receiving a re-initiated handshake in secure communication between mobile applications. At 1202, a secure mobile application handshake including at least one random number identifying parties to the communication (and, e.g., capable of identifying a subsequent round-trip communication) can be established, as described herein. Additionally, the hand shake can also include a unique ID of a primary application that can identify such application as a trusted entity. At 1204, a handshake re-initiation can be received that includes a new random number and the unique ID or a new unique ID. At 1206, the new random number can be stored in a temporary buffer until the new

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	random number is verified. Such verification can include receiving a data established by an application receiving the handshake re-initiation, or by receiving an acknowledgement response (ACK) for Successfully sent data that includes the new random number. Accordingly, the application receiving the handshake re-initiation can attempt to filter legitimate data from malicious or un-authorized data from a malicious and/or un-authorized source." *Id*., ¶111. "At 1208, the new unique ID can be verified, as described herein. For instance, the new unique ID can be reference with respect to a list of trusted primary application unique IDs. If a match is found, the new unique ID can then identify a particular trusted primary application. If a match is not found, the new unique ID can be assumed to be associated with an un-authorized application(s). At 1210, a determination is made as to whether the new unique ID is verified, as described above. If not, method 1200 proceeds to 1214 where the new random number is deleted and no handshake response is sent. If the new unique ID is verified, method 1200 can proceed to 1212 where the random number can be replaced with the new random number to identify subsequent communication between applications. As described, method 1200 provides a mechanism whereby a handshake can be verified prior to storing a new random number and unique ID and releasing and/or erasing a prior random number and/or unique ID. As a result, spoofing attacks that attempt to interrupt secure communication by replacing a random number with a false random number can be mitigated." *Id*., ¶112. "FIGS. 14 and 15 provide block diagrams of sample systems (1400, 1400) in accordance with particular aspects of the claimed subject matter. Specifically, system 1400 can provide for initiating and participating in secure inter-application communication in a mobile operating environment. For instance, system 1400 can include a module for initiating a handshake 1402. Specifically, the module 1402 can initiate a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake. Further, system 1400 can include a module for receiving a response 1404. The module 1404 can specifically receive a response to the handshake initiated by the module 1402, wherein the response includes a first number and a second number. The second number can, for example, be randomly generated by the mobile application (e.g., a determined target of the handshake) and can be utilized to sign and/or certify requests sent to the mobile application from the primary application." *Id*.,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 239 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	¶119. "In addition to the foregoing, system 1500 can include a module for receiving a handshake request 1502. Particularly, the module 1502 can receive a handshake request from a primary application that includes at least a unique ID of the primary application or a random number generated by the primary application, or both. Additionally, system 1500 can include a module for referencing an ID 1504 that can reference a received unique ID, if any, against a list of trusted primary application identifiers. Such a reference can be utilized to distinguish trusted applications from un-authorized or un-trusted applications. Furthermore, system 1500 can include a module for sending a handshake response 1506. The module 1506 can send a handshake response to the primary application if the unique ID matches a unique ID of the list of trusted primary application identifiers, as determined by the module 1504. In addition to the foregoing, system 1500 can also include a module for receiving a data event request 1508, and a module for delaying receipt of such request (e.g., where receipt can include Some form of data processing on received data aside from buffering and/or setting a process timer as described herein). Specifically, the module 1510 can delay receipt of additional data requests until a response to the data event request is initiated by system 1500. Accordingly, round-trip policies can be enforced, also enabling re-transmission of lost requests and/or re-initiation of handshake events in data transfer between mobile applications. As a result, an efficient data transfer architecture can be provided for synchronous communication in a packet loss environment where data is not lost or useless as a result of one or more lost packets." *Id*., ¶121. "A method for providing transmission of data between applications in a mobile operating environment, comprising: initiating a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake; receiving and Verifying a response to the handshake, the response includes at least a second number, sending a data event to the mobile application that includes the second number or a version of the second number, if the response to the handshake is verified; and freezing execution of the primary application until a data event response from the mobile application is received, the data event response includes at least the second number or the version of the second number." *Id*., claim 1.

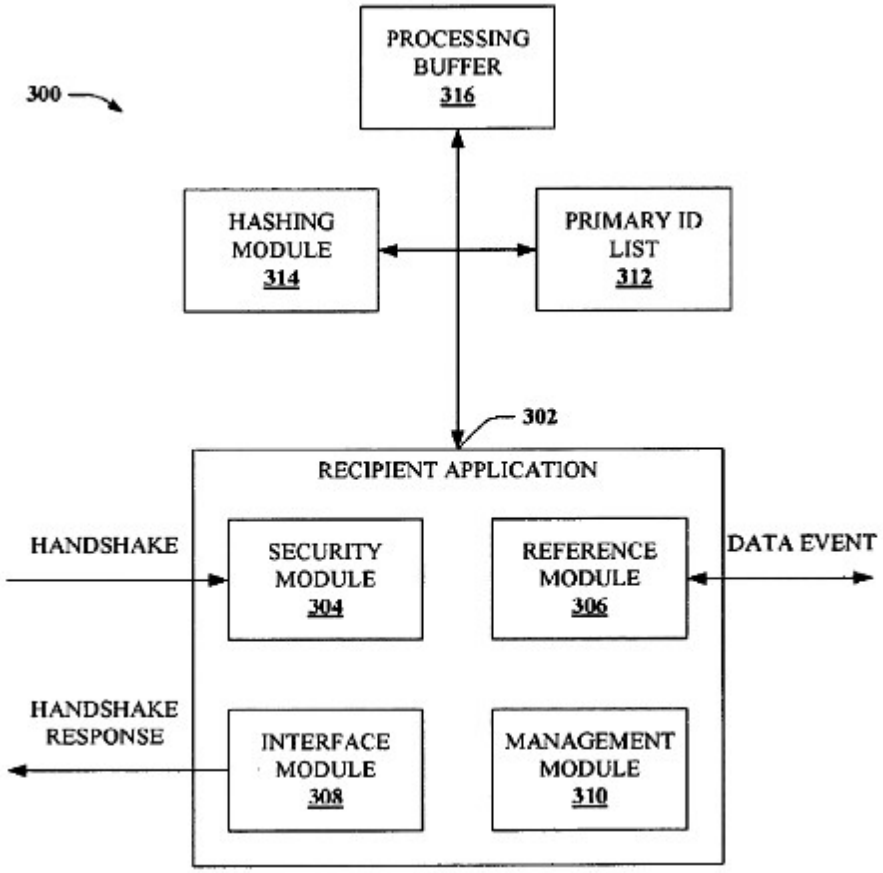
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 240 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"A method for providing data transmission between applications in a mobile operating environment, comprising: receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; referencing the unique ID against a list of trusted primary application identifiers; sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers; receiving a data event request; and responding to the data event request prior to receiving or processing additional data event requests." *Id*., claim 26. "The method of claim 26, further comprising: receiving a handshake re-initiation message that includes a new random number and either the unique ID or a second unique ID; storing the new random number in a temporary buffer; verifying the unique ID or the second unique ID matches an ID of the list of trusted primary application identifiers; and one of replacing the random number with the new random number if the unique ID or the second unique ID is verified or clearing the new random number from the temporary buffer if the unique ID or the second unique ID is not verified." *Id*., claim 33. "An apparatus that provides data transmission between applications in a mobile operating environment, comprising: a security module configured to receive a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; a reference module configure to verify the unique ID against a list of trusted primary application identifiers; an interface module configured to send a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and configured to receive a data event request; and a management module configured require the interface module to respond to the data event request prior to receiving additional data event requests." *Id*., claim 37. "The apparatus of claim 37, wherein the management module is further configured to at least one of: receive a handshake re-initiation message that includes a new random number and either the unique ID or a second unique ID; store the new random number in a temporary buffer; verify the unique ID or the second unique ID matches an ID of the list of trusted primary application identifiers; or one of replace the random number with the new random number if the unique ID or the second unique ID is verified or clear the new random

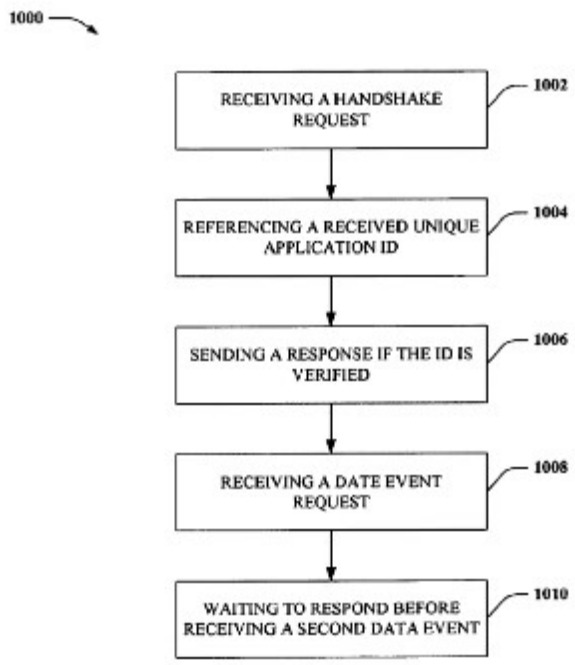
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 241 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	number from the temporary buffer if the unique ID or the second unique ID is not verified." *Id*., claim 44. "At least one processor that provides data transmission between applications in a mobile operating environment, comprising: a first module that receives a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; a second module that references the unique ID against a list of trusted primary application identifiers; a third module that sends a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers; a fourth module that receives a data event request; and a fifth module that responds to the data event request prior to receiving additional data event requests." *Id*., claim 48. "An apparatus that provides data transmission between applications in a mobile operating environment, comprising: means for receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; means for referencing the unique ID against a list of trusted primary application identifiers; means for sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers; means for receiving a data event request; and means for delaying receipt of additional data event requests until a response to the data event request is initiated." *Id*., claim 49.

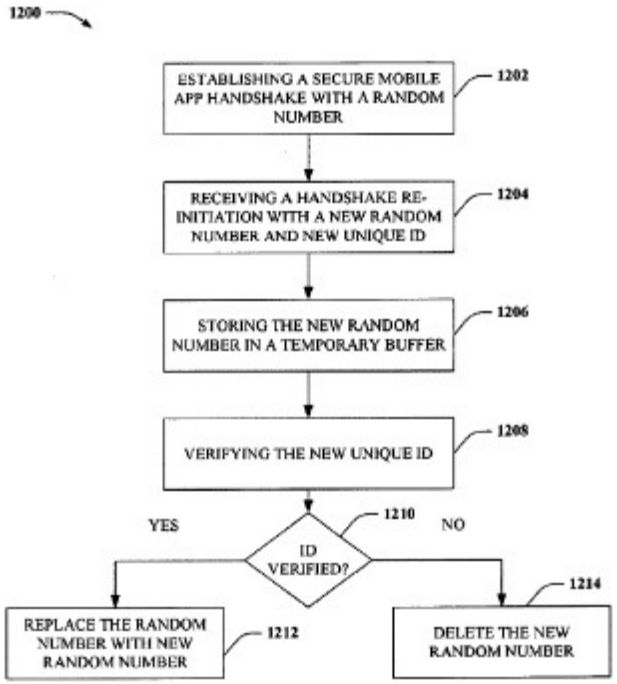
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 242 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 243 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **FIG. 10** *Id.*, Fig. 10.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 244 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **FIG. 12** *Id.*, Fig. 12.
9: The networked system of claim 8, further comprising	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 8, further comprising the respective device link agent on each wireless end-user device receiving access

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 245 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
the respective device link agent on each wireless end-user device receiving access authorization information from the secure server, the access authorization indicating, respectively for each wireless end-user device, the software components authorized to receive messages via the device link agent on that device.	authorization information from the secure server, the access authorization indicating, respectively for each wireless end-user device, the software components authorized to receive messages via the device link agent on that device. • MDS Technical Series, p. 14: Proxy Authentication • Automatic Proxy Authentication – Can specify user ID and password for MDS to login to proxy server – Reduces number of requests made from handheld • Using the handheld to authenticate – Going to the handheld for authentication is the default option – More secure option • The user has to enter the credentials in the following format: – username@domain + password or – domain\username + password – username + password (if a default_domain is set in MDSLogin.conf) • MDS 4.1 Overview, p. 10: Web access control System administrators control wireless access to resources on the corporate network. • Network authentication: The connection service supports several types of network authentication, including HTTP Basic Authentication, NT LAN Manager (NTLM), and Kerberos, so system administrators can control wireless access to web content using existing network security mechanisms. The connection

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 246 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	service also supports Lightweight Third-Party Authentication (LTPA) if cookie storage is enabled. • Access control: System administrators can assign roles to BlackBerry devices and push initiators to control their activity through the connection service. Push access control roles enable system administrators to limit push requests from push initiators to specific BlackBerry device users. Pull access control roles enable system administrators to permit or deny access to URLs that are requested by BlackBerry device users. • MDS 4.1 Overview, p. 12: Application registry and repository Application developers publish completed applications from the BlackBerry MDS Studio to an application repository, and system administrators register the applications with the BlackBerry MDS Services on the BlackBerry Enterprise Server. After the applications are published and registered, System administrators can make them available to BlackBerry device users. Organizations can distribute the applications using one of the following methods: • pushing BlackBerry MDS Studio applications to devices from a central console • enabling BlackBerry device users to search the registry through a device-side console and download applications MDS 4.1 Overview, p. 14: Security BlackBerry MDS Studio application data messages are sent using the data encryption technologies that are used by the BlackBerry Enterprise Server. Application data can also be encrypted on the device using the content protection mechanisms that are available on BlackBerry devices. See the *BlackBerry Enterprise Solution Security Technical Overview* for more information on BlackBerry security features. In addition to message and data content security, BlackBerry MDS Studio applications provide the following security-related features:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 247 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• Application signing: The BlackBerry MDS Studio applications can be signed using the industry-standard (RSA) security certificate. Access to the BlackBerry MDS Studio repository is password-protected. • Container-based execution environment: The BlackBerry MDS Runtime is a container-based execution environment that is designed to prevent unauthorized access to native APIs on the BlackBerry device. BlackBerry MDS Studio applications are prevented from accessing other BlackBerry MDS Studio application data, and one BlackBerry MDS Studio application cannot run or load another. • Secure connectivity to back-end systems: BlackBerry MDS Services support HTTPS for secure access to back-end web services. When this access is combined with the triple DES or AES encryption between the BlackBerry Enterprise Server and the BlackBerry device, a secure end-to-end link from the BlackBerry MDS Studio application to its related web services results. Deploying and managing applications wirelessly System administrators can wirelessly provision, upgrade, and remove BlackBerry MDS Studio applications using the BlackBerry Manager, which is the administration tool for the BlackBerry Enterprise Server. They can send a new application to BlackBerry device users or upgrade an existing application on BlackBerry devices in the field, without physically handling the BlackBerry devices. System administrators can also view the BlackBerry MDS Studio applications that are installed on individual BlackBerry devices. Depending on IT policy settings, BlackBerry device users can use the Control Centre application, which is part of the BlackBerry MDS Runtime, to search the BlackBerry MDS Studio application registry for applications (or application upgrades) that are available for download. Users can select these applications and provision them to their BlackBerry devices. To the extent this limitation is not disclosed by Blackberry Push, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Pazhyannur, Fok.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 248 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Lazaridis Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 8, further comprising the respective device link agent on each wireless end-user device receiving access authorization information from the secure server, the access authorization indicating, respectively for each wireless end-user device, the software components authorized to receive messages via the device link agent on that device (e.g., the host system, which can be a network server containing a preferred list of message types that are to

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 249 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	be redirected or preferred senders whose messages are to be redirected). *See, e.g.*, "A system and method for pushing information from a host system to a mobile data communication device upon sensing a triggering event is disclosed. A redirector program operating at the host system enables a user to continuously redirect certain user-selected data items from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. The redirector program operates in connection with event generating applications and repackaging systems at the host system to configure and detect a particular user-defined event, and then to repackage the user-selected data items in an electronic wrapper prior to pushing the data items to the mobile device. The mobile device and the host system share a common electronic address so that messages generated at either the host system or the mobile data communication device are configured using the common electronic address." Lazaridis, Abstract. "A redirector program operating at the host system enables the user to redirect or mirror certain user-selected data items (or parts of data items) from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. Also operating at the host system are various sub-systems that can be configured to create triggering events, such as a screen saver sub-system or a keyboard sub-system, as well as sub-systems for repackaging the user's data items for transparent delivery to the mobile data device, such as a TCP/IP sub-system or one or more E-Mail sub-systems. Other sub-systems for creating triggering events and repackaging the user's data items could also be present at the host system. The host system also includes a primary memory store where the user's data items are normally stored." *Id.*, 2:61-3:8. "In addition to the functionality noted above, the redirector program provides a set of software-implemented control functions for determining the type of mobile data communication device and its address, for programming a preferred list of message types that are to be redirected, and for determining whether the mobile device can receive and process certain types of message attachments, such as word processor or voice attachments." *Id.*, 3:38-45.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 250 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Once an event has triggered redirection of the user data items, the host system then repackages these items in a manner that is transparent to the mobile data communication device, so that information on the mobile device appears similar to information on the user's host system. The preferred repackaging method includes wrapping the user data items in an E-mail envelope that corresponds to the address of the mobile data communication device, although, alternatively, other repackaging methods could be used with the present invention, such as special-purpose TCP/IP wrapping techniques, or other methods of wrapping the user selected data items." *Id.*, 3:66-4:9. "The primary advantage of the present invention is that it provides a system and method for triggering the continuous and real-time redirection of user-selected data items from a host system to a mobile data communication device. Other advantages of the present invention include: (1) flexibility in defining the types of user data to redirect, and in defining a preferred list of message types that are to be redirected or preferred senders whose messages are to be redirected; (2) flexibility in configuring the system to respond to numerous internal, external and networked triggering events; (3) trans- parent repackaging of the user data items in a variety of ways such that the mobile data communication device appears as though it were the host system; (4) integration with other host system components such as E-mail, TCP/IP, keyboard, screen saver, webpages and certain programs that can either create user data items or be configured to provide trigger points; and (5) the ability to operate locally on a user's desktop system or at a distance via a network server. *Id.*, 4:50-67. "By way of example, consider an E-mail sent to a user that includes three attachments--a word processing document, a video clip and an audio clip. The redirection program could be configured to send the text of the E-mail to the remote device, to send the word processing document to a networked printer located near the user, to send the video clip to a store accessible through a secure connection through the internet and to send the audio clip to the user's voice mail system." *Id.*, 6:20-28.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 251 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a network to initiate redirection. These are just some of the events that could be used with the present invention to initiate replication of the user-selected data items from the host system 10 to the mobile device 24." *Id.*, 7:8-30. "Also programmed into the redirector 12 is the address of the user's mobile data communication device 24, the type of device, and whether the device 24 can accept certain types of attachments, such as word processing or voice attachments." *Id.*, 7:57-60. "The redirector may also be programmed with a preferred list mode that is configured by the user either at the host system 10, or remotely from the user's mobile data communication device by transmitting a command message C. The preferred list contains a list of senders (other users) whose messages are to be redirected or a list of message characteristics that determine whether a message is to be redirected. If activated, the preferred list mode causes the redirector program 12 to operate like a filter, only redirecting certain user data items based on whether the data item was sent from a sender on the preferred list or has certain message characteristics that if present will trigger or suppress redirection of the message. In the example of FIG. 1, if desktop system 26 was operated by a user on the preferred list of host system 10, and the preferred list option was activated, then message A would be redirected. If, however, desktop 26 was operated by a user not on the host system's preferred list, then message A would not be redirected, even if

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 252 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the user of the host system had configured the redirector to push messages of type A The user of the host system 10 can configure the preferred list directly from the desktop system, or, alternatively, the user can then send a command message (such as C) from the mobile device 24 to the desktop system 10 to activate the preferred list mode, or to add or delete certain senders or message characteristics from the preferred list that was previously configured." *Id.*, 7:65-8:31. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*8:32-55. "If the redirected user data item is an E-mail message, as described above, the user at the mobile device 24 sees the original subject, sender's address, destination address, carbon copy and blind carbon copy." 8:66-9:2. "When the user replies to this message, or when the user authors a new message, the software operating at the mobile device 24 adds a similar outer envelope to the reply message (or the new message) to cause the message to be routed first to the user's host system 10, which then removes the outer envelope and redirects the message to the final destination, such as back to computer 26. In the preferred embodiment, this results

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 253 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	in the outgoing redirected message from the user's host system 10 being sent using the E-mail address of the host mailbox, rather than the address of the mobile device, so that it appears to the recipient of the message that the message originated from the user's desktop system 10 rather than the mobile data communication device." *Id.*, 9:2-15. "In this alternative configuration, server 11 preferably maintains a user profile for each user's desktop system 10, 26, 28, including information such as whether a particular user can have data items redirected, which types of message and information to redirect, what events will trigger redirection, the address of the users' mobile data communication device 24, the type of mobile device, and the user's preferred list, if any." *Id.*, 9:34-41. "The method steps carried out by the redirector program 12 are described in more detail in FIG. 4. The basic functions of this program are: (1) configure and setup the user-defined event trigger points that will start redirection; (2) configure the types of user data items for redirection and optionally configure a preferred list of senders whose messages are to be redirected; (3) configure the type and capabilities of the user's mobile data communication device; (4) receive messages and signals from the repackaging systems and the event generating systems; and (5) command and control the redirection of the user-selected data items to the mobile data communication device via the repackaging systems. Other functions not specifically enumerated could be integrated into this program." *Id.*, 10:39-52. "The E-Mail sub-system 44 can receive messages A from external computers on the LAN 14, or can receive messages C from some external network such as the Internet 18 or a wireless data communication network 22, and stores these messages in the primary data store 40. Assuming that the redirector 12 has been triggered to redirect messages of this type, the redirector detects the presence of any new messages and instructs the E-Mail system 44 to repackage the message by placing an outer wrapper B about the original message A (or C), and by providing the addressing information of the mobile data communication device 24 on the outer wrapper B." *Id.*, 10:58-11:2.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 254 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. "Although not shown explicitly in FIG. 4, after step 68, the program could, if operating in the preferred list mode, determine whether the sender of a particular data item is on the preferred list, and if not, then the program would skip over steps 70 and 72 and proceed directly back to step 56. If the sender was on the preferred list, then control would similarly pass to steps 70 and 72 for repackaging and transmission of the message from the preferred list sender." *Id.*, 12:58-65. **Combination with Pazhyannur** Pazhyannur and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Pazhyannur discloses managing applications running on one or more user terminals using TCP/IP protocols. Pazhyannur, [0027]-[0029], [0038], [0040]. Moreover, Blackberry Push describes message delivery to specific applications on user devices. MDS 4.1 Overview, p. 13. Pazhyannur recognizes that users may register multiple devices associated with the user, and the system maintains a centralized user profile that lists all registered devices and applications to permit access to the applications on any of the user's registered devices. Pazhyannur, [0040]. This combination would involve only routine engineering leading to predictable results.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 255 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	A POSITA would have found it obvious to combine Pazhyannur's application management with Blackberry Push's system. This combination of Blackberry Push and Pazhyannur would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Pazhyannur, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Pazhyannur's push messaging system without requiring inventive skill. Pazhyannur, as evidenced by the example citations below, discloses the networked system of claim 8, further comprising the respective device link agent on each wireless end-user device receiving access authorization information from the secure server, the access authorization indicating, respectively for each wireless end-user device, the software components authorized to receive messages via the device link agent on that device. *See, e.g.*, "A method and a communication system having an Application Ensemble Service server is provided that manage an application ensemble having multiple applications running on one or more user terminals. The communication system allows a user to collectively transfer the ensemble from a source terminal or terminals to a target terminal or terminals, for example, from voice and data terminals to other voice and data terminals or to a single user terminal. The communication system further allows the user to selectively transfer applications of the ensemble, such as transferring only the voice session from the voice terminal to another voice or voice/data terminal, and to copy an ensemble, or a portion of an ensemble, from a source terminal to a target terminal, wherein the ensemble or portion of the ensemble is not removed from the source terminal. The communication system further allows a user to suspend and resume an applicationensemble." Pazhyannur, Abstract. "At least one memory device 404 further stores a profile for each user subscribed to the services of communication system 100. Each profile includes a list of user terminals registered by that user, for example, an identifier associated with each such device. For example, a profile of a user associated with user terminal

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 256 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	101 may further include identifiers associated with each of user terminals 102-105. The profile may further include a list of communication devices that the user has permission to access, for example, to copy applications to or from or to transfer applications to or from such devices. By storing associations among all user terminals registered to a user or that a user has permission to access, a user accessing AES server 124 via one of the devices registered to that user may be able to retrieve a list of application ensembles created by any of the other devices registered by, or accessible by, that user. In the sense of being associated with each other, such associated user terminals may also be thought of as related user terminals. In another embodiment of the present invention, the profiles of the users may be maintained in subscriber profile database 116 and may be downloaded from the subscriber profile database and stored by AES server 124 whenever needed." *Id.*, [0040]. **Combination with Fok** Fok and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Blackberry Push and Fok are from the same field in that they disclose buffering and transmitting data using secured TCP/IP protocols. Fok, ¶¶9, 37, 56, 71, Figs. 2-3. Fok describes "secure inter-application communication" in a "mobile operating environment." *Id*., ¶¶36-37, 81, 106-108, Figs. 4, 8-9. Fok uses a "handshake" between a "primary" and one or more "recipient" applications to establish recipients are trusted using "a list of unique identifiers" before transmitting information between the two. *Id*., ¶¶47-53, 96-102, 106-108, Figs. 7-10. Fok uses a "handshake" between a "primary" and one or more "recipient" applications to establish recipients are trusted using "a list of unique identifiers" before transmitting information between the two. *Id.*, [0047]-[0053], [0096]-[0102],[0106]-[0108], Figs. 7-10. This combination would involve only routing engineering leading to predictable results. A POSITA would have been motivated to combine Blackberry Push and Fok.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 257 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Fok, as evidenced by the example citations below, discloses the networked system of claim 8, further comprising the respective device link agent on each wireless end-user device receiving access authorization information from the secure server, the access authorization indicating, respectively for each wireless end-user device, the software components authorized to receive messages via the device link agent on that device (e.g., comparing the unique identifiers against a list of the trusted applications). *See e.g.*, "Providing for secure and efficient communication for mobile applications executed in a mobile operating environment is described herein. As an example, a primary mobile application can initiate a handshake that includes a unique identifier of the primary application and a random number for signing and/or certifying responsive requests. A recipient application can reference the unique identifier with a list of certified primary applications to Verify the primary application. If verified, the recipient responds with the random number and a second random number that can sign and/or certify data requests sent by the primary application. According to some embodiments, random numbers can be hashed and/or truncated to provide low power encryption for such numbers. Further, round-trip policies can be enforced to provide reliable transmission of data. Accordingly, reliable, secure and low power synchronous communication can be conducted in a mobile environment." Fok, Abstract. "According to other aspects, disclosed is a method for providing transmission of data between applications in a mobile operating environment. The method can include initiating a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake and receiving a response to the handshake that includes a first number and a second number, the second number is randomly generated by the mobile application. The method can further include verifying that the first number is the random number generated for the handshake and sending a data event to the mobile application that includes an increment of the second number if the first number is verified as the random number. Moreover, the method can also include waiting for a data event response from the mobile application that includes at least the increment of the second number prior to sending a second data event." *Id.*, [0010].

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 258 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"According to one or more other aspects, described is a method for providing data transmission between applications in a mobile operating environment. The method can include receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application and referencing the unique ID against a list of trusted primary application identifiers. In addition, the method can include sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and receiving a data event request. Additionally, the method can further include responding to the data event request prior to receiving additional data event requests." *Id.*, [0015]. "According to additional aspects, disclosed is an apparatus that provides data transmission between applications in a mobile operating environment. The apparatus can include a security module configured to receive a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application. Further, the apparatus can include a reference module configure to verify the unique ID against a list of trusted primary application identifiers and an interface module configured to send a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and configured to receive a data event request. In addition to the foregoing, the apparatus can also include a management module configured require the interface to respond to the data event request prior to receiving additional data event requests." *Id.*, [0016]. "According to still other aspects, at least one processor that provides data transmission between applications in a mobile operating environment is described. The processor(s) can include a first module that receives a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application and a second module that references the unique ID against a list of trusted primary application identifiers. Further, the processor(s) can include a third module that sends a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and a fourth module that receives a data event request. Additionally,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 259 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the processor(s) can include a fifth module that responds to the data event request prior to receiving additional data event requests." *Id.*, [0017]. "According to one or more additional aspects, an apparatus that provides data transmission between applications in a mobile operating environment. The apparatus can include means for receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application and means for referencing the unique ID against a list of trusted primary application identifiers. Further, the apparatus can include means for sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and means for receiving a data event request. Moreover, the apparatus can also include means for delaying receipt of additional data event requests until a response to the data event request is received." *Id.*, [0018]. "A handshake can be initiated by the primary application 102 by sending a handshake request that includes at least a unique identifier of the primary application 102 or a random number generated for the handshake, or both. The unique identifier is an application ID associated with the primary application, that can be utilized by a recipient application to verify that the primary application is to be trusted. Typically, an operating system (106) can require certain standards be met for applications designed to execute on the operating system (106). One such requirement can be to incorporate a list of unique identifiers of trusted primary applications. A recipient application 104 having such a list could then verify that a received handshake request is sent by an application authorized to make the request. The recipient (104) could then trust and engage in communication with such application." *Id.*, [0048]. "In addition to a unique identifier of a primary application, a handshake request can also include a random number (e.g., 32 bit, 16 bit, 64 bit, and so on) generated for the handshake. The random number can be sent to an intended recipient application and incorporated into responses sent to the primary application 104 by the recipient application 104. As a result, a response that includes the random number (or, e.g., a predictable variation thereof, discussed below) can be identified as originating from the intended recipient. According to additional aspects, the random number can be hashed utilizing a particular hash function (e.g., SHA-1,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 260 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	SHA-2, or the like) and/or truncated for transmission over a particular communication interface (106). A hashed random number can be much more difficult to decrypt, providing significant protection from spoofing attacks. Truncation can be utilized to transmit the random number utilizing various protocols and/or communication interfaces (106) that support different size numbers. For instance, if a generated random number identifying a handshake session is 32-bit, and a communication interface with a recipient application (104) enables only 16-bit data transfer, the 32-bit random number can be hashed and then truncated to 16-bit for transmission. The recipient can then hash and/or truncate the random number and compare it to the received hashed and/or truncated number to verify whether a sender of communication is the initiator (102) of the handshake." *Id.*, [0049]. "Recipient application 104 can receive a handshake request and verify whether a sender (102) is a trusted primary application. Particularly, a unique identifier can be extracted from the handshake request and compared with a list of trusted primary application identifiers. If the extracted identifier matches an identifier on the list, a handshake response can be sent to the sender (102), establishing a secure channel between sender (102) and recipient (104). If the extracted identifier does not match an identifier on the list, the handshake and any subsequent communication can be ignored until a verified handshake is received." *Id.*, [0050]. "Additionally, the recipient application 104 can extract and store a random number included within the handshake request. As a specific example, the random number can be stored in temporary memory (not depicted) to make it more difficult for malicious programs to extract the random number and spoof the recipient application's (104) identity. Upon verifying the unique identifier included within the handshake request, discussed above, the recipient application 104 can send a handshake response that includes the random number. Particularly, by including the random number the recipient application 104 can identify itself to the primary application 102. According to particular aspects, the recipient application 104 can hash and/or truncate the random number to make spoofing the recipient application's (104) identity (e.g., to the primary application 102 that generated and therefore is configured to trust the random number) more

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 261 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	difficult and/or make the hashed number more compatible with particular communication interfaces (106), as discussed above." *Id.*, [0051]. "Recipient application 302 can include at least a security module 304 configured to receive a handshake request from a primary application (not depicted, but see 202 at FIG. 2, supra). The handshake request can include a unique ID of the primary application and/or a random number generated by the primary application. Additionally, the recipient application 302 can include a reference module 306 configured to verify the unique ID of the primary application. For instance, the reference module can compare the unique ID to a list of trusted primary application identifiers 312. Such list (312) can include all unique IDs associated with applications that the recipient application 302 should trust and accept a handshake from. Thus, a primary application's unique ID can be verified by reference to the list 312." *Id.*, [0073]. "Method 700, at 702, can initiate a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake. The primary application can be configured to communicate via a mobile operating system (e.g., BREW operating or the like) with the mobile application, utilizing a message passing architecture, or shared memory, or a suitable combination thereof or of like data transfer platforms. The unique identifier specified in the handshake can be utilized to identify the primary application as a trusted application. The random number, on the other hand, can be utilized by the mobile application to identify itself to the primary application. Consequently, communication received at the primary application that contains the random number (and/or, e.g., an appropriate increment thereof) can be associated with the mobile application. As a corollary to the foregoing, if a communication received at the primary application does not include the random number (and/or, e.g., an appropriate increment thereof), the communication can be assumed unassociated with the handshake and therefore not secure." *Id.*, [0096]. "FIG. 10 illustrates a sample methodology 1000 for responding to secure communication between mobile applications in a mobile operating environment. At 1002, a handshake request can be received. The handshake request can include a unique identifier of an initiating application and/or a random number

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 262 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	identifying the handshake, as described herein. At 1004, the unique identifier, if received, can be referenced against a list of trusted primary application IDs. If a match between a received unique identifier is found, method 900 can proceed to 1006. If not, method 900 can terminate (not depicted). At 1006, a response to the handshake can be sent if the unique ID is verified against the list of trusted applications. At 1008, a data event request can be received. The data event request can include, for instance, a second random number included within the handshake response to identify subsequent incoming communication associated with the handshake session (e.g., where such session is established by the handshake request received at reference number 1002 and handshake response sent at reference number 1006). At 1010, receipt of an additional data event request can be delayed at least until the data event request is responded to (e.g., a response to the data event request is sent prior to receiving additional data event requests). Accordingly, method 1000 can impose round-trip communication policies for secure communication, incorporating random numbers generated by initiating and recipient applications. Additionally, such random numbers can be hashed and truncated to provide additional security with low processing requirements and power cost. Therefore, method 1000 provides a mechanism to respond and participate to secure inter-application communication particularly beneficial for a mobile operating environment." *Id.*, [0109]. "FIG. 12 illustrates an example methodology 1200 for receiving a re-initiated handshake in secure communication between mobile applications. At 1202, a secure mobile application handshake including at least one random number identifying parties to the communication (and, e.g., capable of identifying a subsequent round-trip communication) can be established, as described herein. Additionally, the hand shake can also include a unique ID of a primary application that can identify such application as a trusted entity. At 1204, a handshake re-initiation can be received that includes a new random number and the unique ID or a new unique ID. At 1206, the new random number can be stored in a temporary buffer until the new random number is verified. Such verification can include receiving a data established by an application receiving the handshake re-initiation, or by receiving an acknowledgement response (ACK) for Successfully sent data that includes the new random number. Accordingly, the application receiving the handshake re-initiation can attempt to filter legitimate data from malicious or un-authorized data from a malicious and/or un-authorized source." *Id.*, [0111].

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 263 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"At 1208, the new unique ID can be verified, as described herein. For instance, the new unique ID can be reference with respect to a list of trusted primary application unique IDs. If a match is found, the new unique ID can then identify a particular trusted primary application. If a match is not found, the new unique ID can be assumed to be associated with an un-authorized application(s). At 1210, a determination is made as to whether the new unique ID is verified, as described above. If not, method 1200 proceeds to 1214 where the new random number is deleted and no handshake response is sent. If the new unique ID is verified, method 1200 can proceed to 1212 where the random number can be replaced with the new random number to identify subsequent communication between applications. As described, method 1200 provides a mechanism whereby a handshake can be verified prior to storing a new random number and unique ID and releasing and/or erasing a prior random number and/or unique ID. As a result, spoofing attacks that attempt to interrupt secure communication by replacing a random number with a false random number can be mitigated." *Id.*, [0112]. "FIGS. 14 and 15 provide block diagrams of sample systems (1400, 1400) in accordance with particular aspects of the claimed subject matter. Specifically, system 1400 can provide for initiating and participating in secure inter-application communication in a mobile operating environment. For instance, system 1400 can include a module for initiating a handshake 1402. Specifically, the module 1402 can initiate a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake. Further, system 1400 can include a module for receiving a response 1404. The module 1404 can specifically receive a response to the handshake initiated by the module 1402, wherein the response includes a first number and a second number. The second number can, for example, be randomly generated by the mobile application (e.g., a determined target of the handshake) and can be utilized to sign and/or certify requests sent to the mobile application from the primary application." *Id.*, [0119]. "In addition to the foregoing, system 1500 can include a module for receiving a handshake request 1502. Particularly, the module 1502 can receive a handshake request from a primary application that includes at

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 264 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	least a unique ID of the primary application or a random number generated by the primary application, or both. Additionally, system 1500 can include a module for referencing an ID 1504 that can reference a received unique ID, if any, against a list of trusted primary application identifiers. Such a reference can be utilized to distinguish trusted applications from un-authorized or un-trusted applications. Furthermore, system 1500 can include a module for sending a handshake response 1506. The module 1506 can send a handshake response to the primary application if the unique ID matches a unique ID of the list of trusted primary application identifiers, as determined by the module 1504. In addition to the foregoing, system 1500 can also include a module for receiving a data event request 1508, and a module for delaying receipt of such request (e.g., where receipt can include Some form of data processing on received data aside from buffering and/or setting a process timer as described herein). Specifically, the module 1510 can delay receipt of additional data requests until a response to the data event request is initiated by system 1500. Accordingly, round-trip policies can be enforced, also enabling re-transmission of lost requests and/or re-initiation of handshake events in data transfer between mobile applications. As a result, an efficient data transfer architecture can be provided for synchronous communication in a packet loss environment where data is not lost or useless as a result of one or more lost packets." *Id.*, [0121]. "A method for providing transmission of data between applications in a mobile operating environment, comprising: initiating a handshake with a mobile application that includes a unique identifier of a primary mobile application and a random number generated for the handshake; receiving and Verifying a response to the handshake, the response includes at least a second number, sending a data event to the mobile application that includes the second number or a version of the second number, if the response to the handshake is verified; and freezing execution of the primary application until a data event response from the mobile application is received, the data event response includes at least the second number or the version of the second number." *Id.*, claim 1. "A method for providing data transmission between applications in a mobile operating environment, comprising: receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; referencing the unique ID

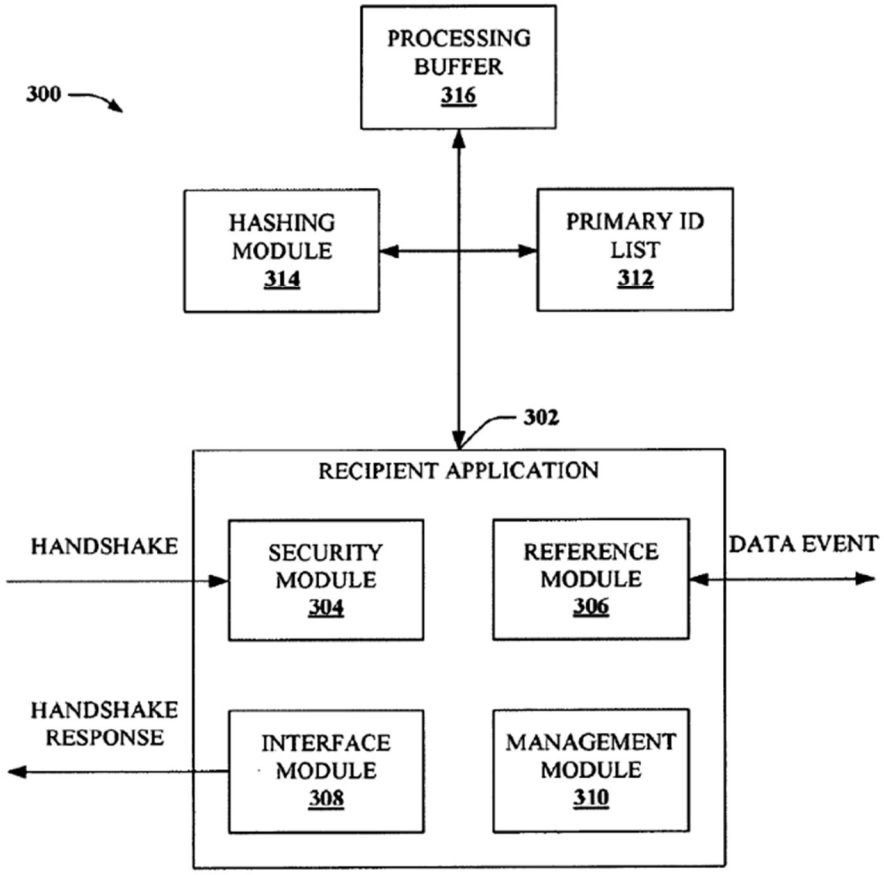
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 265 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	against a list of trusted primary application identifiers; sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers; receiving a data event request; and responding to the data event request prior to receiving or processing additional data event requests." *Id.*, claim 26. "The method of claim 26, further comprising: receiving a handshake re-initiation message that includes a new random number and either the unique ID or a second unique ID; storing the new random number in a temporary buffer; verifying the unique ID or the second unique ID matches an ID of the list of trusted primary application identifiers; and one of replacing the random number with the new random number if the unique ID or the second unique ID is verified or clearing the new random number from the temporary buffer if the unique ID or the second unique ID is not verified." *Id.*, claim 33. "An apparatus that provides data transmission between applications in a mobile operating environment, comprising: a security module configured to receive a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; a reference module configure to verify the unique ID against a list of trusted primary application identifiers; an interface module configured to send a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers and configured to receive a data event request; and a management module configured require the interface module to respond to the data event request prior to receiving additional data event requests." *Id.*, claim 37. "The apparatus of claim 37, wherein the management module is further configured to at least one of: receive a handshake re-initiation message that includes a new random number and either the unique ID or a second unique ID; store the new random number in a temporary buffer; verify the unique ID or the second unique ID matches an ID of the list of trusted primary application identifiers; or one of replace the random number with the new random number if the unique ID or the second unique ID is verified or clear the new random number from the temporary buffer if the unique ID or the second unique ID is not verified." *Id.*, claim 44.

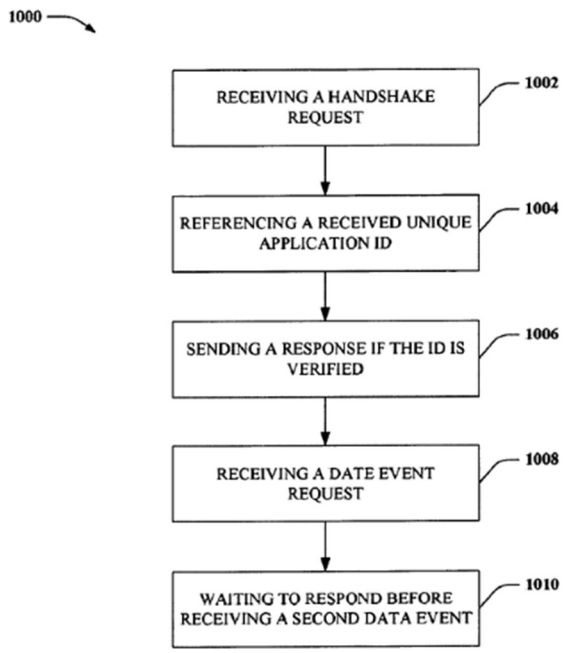
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 266 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"At least one processor that provides data transmission between applications in a mobile operating environment, comprising: a first module that receives a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; a second module that references the unique ID against a list of trusted primary application identifiers; a third module that sends a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers; a fourth module that receives a data event request; and a fifth module that responds to the data event request prior to receiving additional data event requests." *Id.*, claim 48. "An apparatus that provides data transmission between applications in a mobile operating environment, comprising: means for receiving a handshake request from a primary application that includes a unique ID of the primary application or a random number generated by the primary application; means for referencing the unique ID against a list of trusted primary application identifiers; means for sending a handshake response to the primary application if the unique ID matches an ID of the list of trusted primary application identifiers; means for receiving a data event request; and means for delaying receipt of additional data event requests until a response to the data event request is initiated." *Id.*, claim 49.

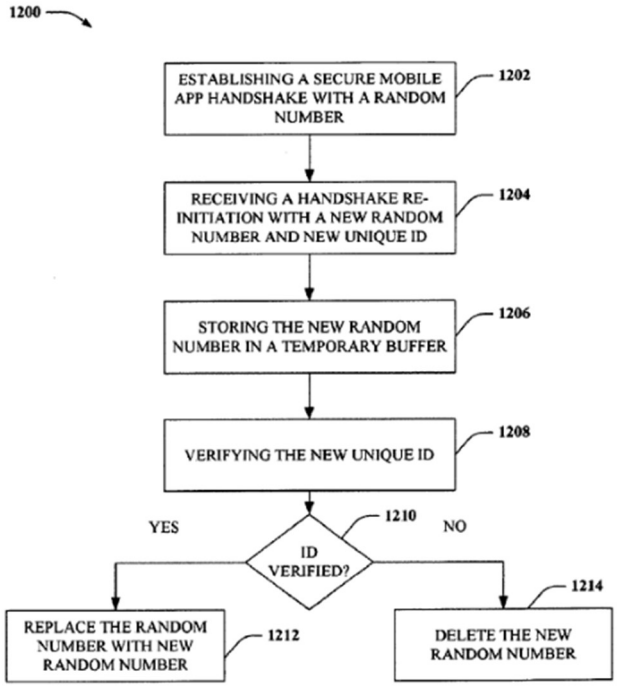
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 267 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 *Id.*, Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 268 of 422

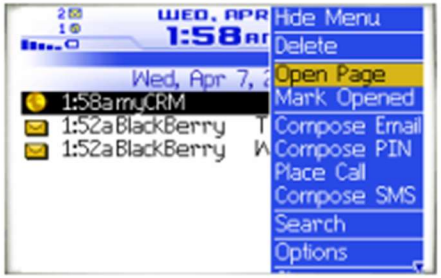
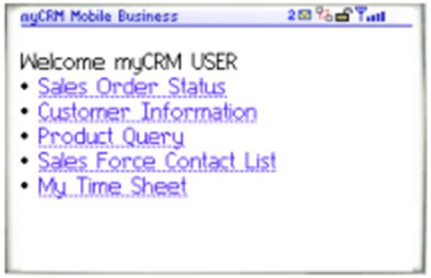
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **FIG. 10** *Id.*, Fig. 10.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 269 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **FIG. 12** *Id.*, Fig. 12.

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
10: The networked system of claim 1, wherein a second subset of the secure Internet data messages contain a user message from a network element, the user message intended for display on a user interface of a given one of the wireless end-user devices.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein a second subset of the secure Internet data messages contain a user message from a network element, the user message intended for display on a user interface of a given one of the wireless end-user devices. Blackberry Push supports user-facing push messages and browser content, including Message Push to display notifications/inbox items and Channel Push to create icons/home screen entries. *See, e.g.*, • MDS Technical Series, p. 45: Push Types • BlackBerry-defined push types: – "Channel Push" • Push an icon to the Home screen along with a web page • The icon acts as a permanent 'channel' to the referenced URL – "Message Push" • Push a message to the user's inbox that automatically launches the browser to a specified page (cached or not) – "Cache Push" • Push a page to the browser's cache, but do not notify the user • MDS 4.1 Overview, p. 10: Pushing content to the BlackBerry Browser The BlackBerry Browser provides built-in support for various types of push services, which provide

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 271 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	different notification options when the BlackBerry device receives pushed content. Each push service uses the same push mechanism through the connection service. Browser messages Applications can send web pages or alerts to the BlackBerry device so that they appear as browser message in the BlackBerry device messages list. The browser message includes the content to display or the URL of the page to fetch when the user requests it. This push method is useful for alerts to important information updates. Users set notification settings through the BlackBerry device profiles list.    Browser message from a Customer Relationship Management (CRM) system Browser channels

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Applications can push web pages to the BlackBerry device that create or update channels, which appear on the BlackBerry device Home screen with a custom icon. An updated icon appears when new content is available. This push service type effectively creates a browser-based application on the BlackBerry device that enables users to receive updates to certain types of data. A browser channel is an effective way to place a permanent icon on the Home screen that acts like a browser bookmark to important content.  Browser channel push from a CRM system Browser cache updates Applications can push web pages directly to the browser cache. The user receives no indication that the content is updated, but the next time that the user visits the specified URL, the browser retrieves the updated content from the cache. Browser cache updates can provide users with quick access to current data on their BlackBerry devices, even when they are not in a wireless coverage area. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 273 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
11: The networked system of claim 1, wherein one of the software components on a given one of the wireless end-user devices is a policy control agent, and the data to be consumed by the policy control agent comprises service settings and/or configuration information for the given device.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the software components on a given one of the wireless end-user devices is a policy control agent, and the data to be consumed by the policy control agent comprises service settings and/or configuration information for the given device. BlackBerry devices contain software components that receive and implement control information, e.g. "policy control agents." *See, e.g.*, • MDS Technical Overview, p. 23: • IPPP service book will be pushed to handhelds OTA – Required for handheld to communicate with MDS • MDS Technical Overview, p. 9: • Handheld and MDS communicate using IP Proxy Protocol – IPPP service book required on the handheld • Java Developers Guide 4.0, p. 7: net.rim.device.api.servicebook: This package enables applications to add, delete, and access service book entries. See the BlackBerry Application Developer Guide Volume 2: Advanced Topics for more information. • MDS 4.1 Overview, p. 12: Distributed processing resources

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 274 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	BlackBerry MDS Studio applications do not perform operations themselves; they define what to do. The programmatic behavior is distributed among multiple external components. • On the application server, one or more web services contain the code to complete the user tasks that the BlackBerry MDS Studio application provides and to support operations such as binding, discovery, provisioning, and notification. • On the BlackBerry device, the BlackBerry MDS Runtime provides a generic set of services such as screen handling, persistent data storage, and messaging for all of the BlackBerry MDS Studio applications that run on a BlackBerry device. • MDS 4.1 Overview, p. 14: Deploying and managing applications wirelessly System administrators can wirelessly provision, upgrade, and remove BlackBerry MDS Studio applications using the BlackBerry Manager, which is the administration tool for the BlackBerry Enterprise Server. They can send a new application to BlackBerry device users or upgrade an existing application on BlackBerry devices in the field, without physically handling the BlackBerry devices. System administrators can also view the BlackBerry MDS Studio applications that are installed on individual BlackBerry devices. Depending on IT policy settings, BlackBerry device users can use the Control Centre application, which is part of the BlackBerry MDS Runtime, to search the BlackBerry MDS Studio application registry for applications (or application upgrades) that are available for download. Users can select these applications and provision them to their BlackBerry devices. • MDS 4.1 Overview, p. 17: Security • Application control: System administrators can control whether users can install and run third-party applications, and they can view which applications have been installed on specific BlackBerry devices. • Access to memory: The BlackBerry Java implementation is designed to inhibit applications from causing

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 275 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	problems accidentally or maliciously in other applications or on the BlackBerry device. Java applications can write only to BlackBerry device memory that is allocated specifically for use by the BlackBerry JVM, and they cannot access the virtual memory or the persistent storage of other applications (unless they are specifically granted access to do so). Custom applications can only access persistent storage or user data, or communicate with other applications, through specific APIs. Research In Motion (RIM) must digitally sign applications that use certain BlackBerry APIs to provide an audit trail of applications that use sensitive APIs. To the extent this limitation is not disclosed by Blackberry Push, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Pazhyannur. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the ’320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet.” *Id.*, 4:19-38.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 276 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the software components on a given one of the wireless end-user devices (e.g., software-implemented control functions) is a policy control agent, and the data to be consumed by the policy control agent comprises service settings and/or configuration information for the given device (e.g., the network address of the mobile data communication device, the types of message attachments that the mobile data communication device can receive and process, and the type of mobile data communication device). *See, e.g.*, "In addition to the functionality noted above, the redirector program provides a set of software-implemented control functions for determining the type of mobile data communication device and its address, for programming a preferred list of message types that are to be redirected, and for determining whether the mobile device can receive and process certain types of message attachments, such as word processor or voice attachments." Lazaridis, 3:38-45. "The present invention includes the ability to redirect certain message attachments to such an external machine 30 if the redirector program configuration data reflects that the mobile device 24 cannot receive and process the attachments, or if the user has specified that certain attachments are not to be forwarded to mobile device 24, even if such device can process those attachments." *Id.*, 6:13-20.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 277 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The mobile data communication device 24 includes software program instructions that work in conjunction with the redirector program 12 to enable the seamless, transparent redirection of user-selected data items." *Id.*, 6:42-45. "In an alternative embodiment of the present invention, not explicitly shown in the drawings, the mobile device 24 also includes a redirector program. In this embodiment, user selected data items can be replicated from the host to the mobile device and vice versa. The configuration and operation of the mobile device 24 having a redirector program is similar to that described herein with respect to FIGS. 1-4." *Id.*, 6:49-55. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "As shown in FIG. 2, desktop system 26 generates a message A that is transmitted to and stored at the host system 11, which is the network server operating the redirector program 12. The message A is for desktop system 10, but in this embodiment, user messages are stored at the network server 11. When an event occurs at desktop system 10, an event trigger is generated and transmitted to the network server 11, which then determines who the trigger is from, whether that desktop has redirection capabilities, and if so, the server

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 278 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	(operating the redirector program) uses the stored configuration information to redirect message A to the mobile computer 24 associated with the user of desktop system 10." *Id.*, 9:52-64. "At step 80 the mobile software is started and the mobile device 24 is configured to operate with the system of the present invention, including, for example, storing the address of the user's desktop system 10." *Id.*, 13:1-5. "The method of claim 2, wherein the configuration information stored at the host system includes: (A) the network address of the mobile data communication device; and (B) an indication of the types of message attachments that the mobile data communication device can receive and process." *Id.*, claim 3. "The method of claim 3, wherein the configuration information further includes: (C) an indication of the type of mobile data communication device." *Id.*, claim 4. **Combination with Pazhyannur** Pazhyannur and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Pazhyannur discloses managing applications running on one or more user terminals using TCP/IP protocols. Pazhyannur, [0027]-[0029], [0038], [0040]. Moreover, Blackberry Push describes message delivery to specific applications on user devices. MDS 4.1 Overview, p. 13. Pazhyannur recognizes that users may register multiple devices associated with the user, and the system maintains a centralized user profile that lists all registered devices and applications to permit access to the applications on any of the user's registered

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 279 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	devices. Pazhyannur, [0040]. This combination would involve only routine engineering leading to predictable results. A POSITA would have found it obvious to combine Pazhyannur's application management with Blackberry Push's system. This combination of Blackberry Push and Pazhyannur would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Pazhyannur, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Pazhyannur's push messaging system without requiring inventive skill. To the extent that Blackberry Push does not disclose the networked system of claim 1, wherein one of the software components on a given one of the wireless end-user devices is a policy control agent, and the data to be consumed by the policy control agent comprises service settings and/or configuration information for the given device., it would have been obvious to a POSITA to combine Blackberry Push and Pazhyannur to meet this limitation. Blackberry Push and Pazhyannur disclose similar technologies addressing similar problems in the same field. Pazhyannur recognizes that users may register multiple devices associated with the user, and the system maintains a centralized user profile that lists all registered devices and applications to permit access to the applications on any of the user's registered devices. Pazhyannur, [0040]. This combination would involve only routine engineering leading to predictable results. Pazhyannur, as evidenced by the example citations below, discloses the networked system of claim 1, wherein one of the software components on a given one of the wireless end-user devices is a policy control agent (e.g., CSCF serving as the policy manager and policy enforcement point), and the data to be consumed by the policy control agent comprises service settings and/or configuration information for the given device (e.g., operator policies and user preferences). *See, e.g.*,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 280 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Infrastructure 110 is an Internet Protocol Multimedia Subsystem (IMS)-based system and includes an IMS core network 112 comprising a Call Session Control Function (CSCF) 114 that implements one or more of a Proxy CSCF (P-CSCF), a Serving CSCF (S-CSCF), and an Interrogating CSCF (I-CSCF). CSCF 114 serves as a centralized routing engine, policy manager, and policy enforcement point to facilitate the delivery of multiple real-time applications using IP transport." Pazhyannur, [0033]. "When an AES client requests to transfer an application ensemble, AES server 124 may modify the transfer, for example, by modifying a file associated with one or more applications being transferred or by blocking a transfer of one or more applications or files, based on policies stored on the server by a system operator or based on a profile of a user associated with a source user terminal or a target user terminal, which profiles may be downloaded by the S-CSCF from a subscriber profile database, such as subscriber profile database 130, during registration of the user terminal (via well-known registration procedures) and retrieved by AES server 124 from the S-CSCF or which profile may be downloaded directly by the AES server 124 during the registration." *Id.*, [0035]. "FIG. 4 is a block diagram of AES server 124 in accordance with an embodiment of the present invention. AES server 124 includes a processor 402, such as one or more microprocessors, microcontrollers, digital signal processors (DSPs), combinations thereof or such other devices known to those having ordinary skill in the art. AES server 124 further includes at least one memory device 404 associated with the processor, such as random access memory (RAM), dynamic random access memory (DRAM), and/or read only memory (ROM) or equivalents thereof, that store data and programs, such as Session Initiation Protocol (SIP)-related programs, that may be executed by the processor and that allow the AES server to perform all functions necessary to operate in communication system 100. At least one memory device 404 may further maintain operator policies concerning applications that may or may not be transferred among user terminals or any modifications that should be made to such applications before the applications are transferred. For example, when an ensemble is transferred from a laptop to a mobile station, and the ensemble contains a Microsoft® NetMeeting® application that is not supported by the mobile station, then the NetMeeting application should not be transferred." *Id.*, [0039].

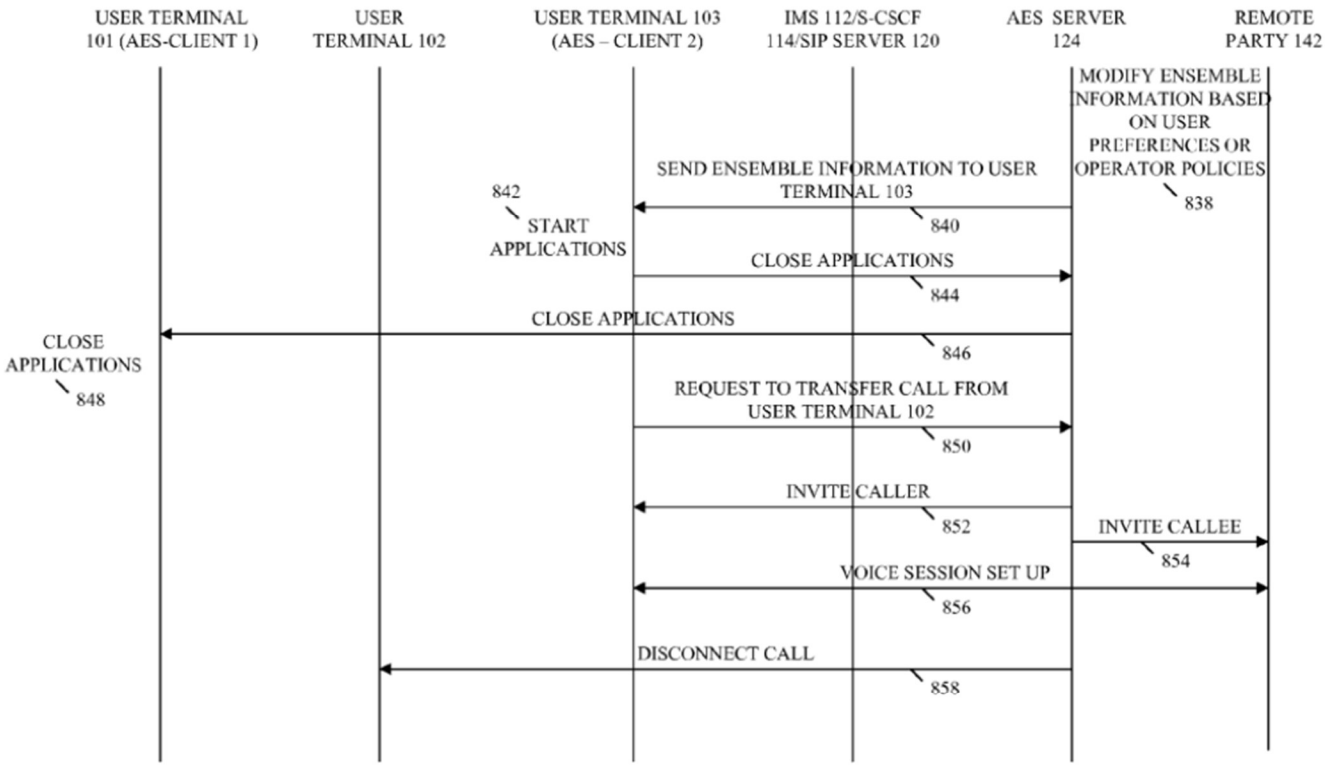
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 281 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"In response to receiving the ensemble information from user terminal 103, AES server 124 optionally may determine (838) whether to modify any of the ensemble information based on preferences of the user of user terminals 101, 102, and 103 or based on operator policies maintained in the at least one memory device 404 of AES server 124. The user preferences may be maintained by AES server 124 in a profile of the user or such user preferences may be maintained in a profile of the user at subscriber profile database 116 and downloaded from the database by AES server 124. For example, when an ensemble is transferred from a laptop to a mobile station, and the ensemble contains a Microsoft NetMeeting application that is not supported by the mobile station, then the NetMeeting application should not be transferred to the mobile station." *Id.*, [0068]. "After optionally modifying (838) the ensemble information in accordance with any determined user preferences or operator policies, AES server 124 conveys (840) the ensemble information, as so modified, to AES client 306 of user terminal 103. In response to receiving the ensemble information elements, AES client 306 of user terminal 103 instructs (842) the user terminal to start up the applications identified in the ensemble information and further to open, and download if necessary, the associated files based on the ensemble information elements." *Id.*, [0069]. "In response to receiving the ensemble information elements from user terminal 103, AES server 124 may optionally determine (1120) whether to modify any of the ensemble information elements based on preferences associated with user terminals 103 or 105 or based on operator policies stored in the at least one memory device 404 of AES server 124. The user preferences may be maintained by AES server 124 in the profile of the user or such user preferences may be maintained by subscriber profile database 116 and downloaded from the database by AES server 124." *Id.*, [0084]. "After optionally modifying (1120) the ensemble information elements in accordance with any determined user preferences or operator policies, AES server 124 conveys (1122) the ensemble information elements, as so modified, to AES client 306 of user terminal 105. In response to receiving the ensemble information

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 282 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	elements, AES client 306 of user terminal 105 instructs (1124) the user terminal to start up the applications identified in the ensemble information and to open, and to download if necessary, associated files based on the ensemble information elements." *Id.*, [0085].

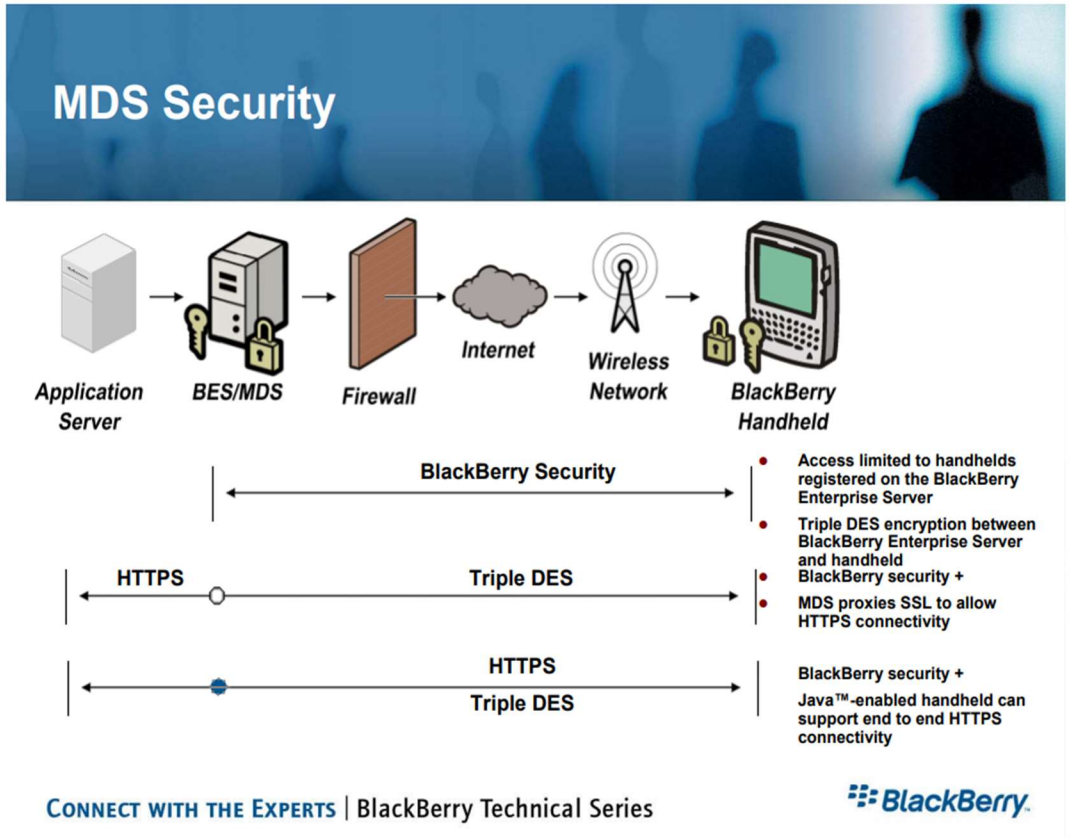
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 283 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 800 FIG. 8B *Id.*, Fig. 8B.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 284 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
12: The networked system of claim 1, the link interface to encrypt messages identified for delivery to each given one of the wireless end-user devices to create secure Internet data messages, the device link agent on each given device further configured to decrypt the received secure Internet data messages for that device prior to delivering those messages to a respective software process.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, the link interface to encrypt messages identified for delivery to each given one of the wireless end-user devices to create secure Internet data messages, the device link agent on each given device further configured to decrypt the received secure Internet data messages for that device prior to delivering those messages to a respective software process. Blackberry Push discloses encryption of messages on the secure link between the server and device. Specifically, Blackberry Push encrypts all device-bound traffic on the BES transport using 3DES/AES; MDS additionally supports HTTPS/TLS. Devices decrypt on receipt before dispatch to applications. *See, e.g.*, • MDS Technical Series, p. 9: MDS Architecture • HTTP connection requests utilize the existing secure 3DES (or AES) connection over port 3101 • MDS communicates with web servers using protocol requested by handheld • MDS supports common protocols and mark up languages – HTTP, HTTPS, TCP, XML, HTML • Handheld and MDS communicate using IP Proxy Protocol – IPPP service book required on the handheld • MDS Technical Series, p. 12:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 285 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 • BES Overview, p. 2: Security

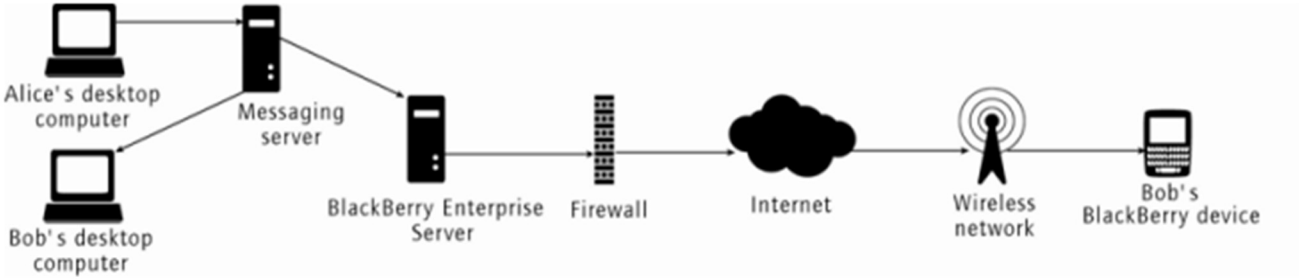
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 286 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	The BlackBerry® Enterprise Solution was created with corporate data security in mind. End-to-end AES or Triple DES* encryption – Helps ensure the confidentiality and integrity of wirelessly transmitted information from behind the firewall to wireless devices in the field. Over-the-air wireless IT policy enforcement and commands – Define and wirelessly enforce security settings on devices, as well as impose device lock-down or wipe data from lost or stolen devices. Learn more about BlackBerry® Enterprise Server IT policies Optional S/MIME** and PGP support – Wirelessly sign and encrypt messages between senders and recipients, providing writer to reader security functionality. Learn more about PGP support Available BlackBerry Smart Card Reader – Create a secure, two-factor authenticated environment for granting access to BlackBerry® and PKI applications. Learn more about the BlackBerry® Smart Card Reader FIPS 140-2 validated encryption technology – The BlackBerry Enterprise Solution meets strict U.S. government and military encryption standards. • MDS Studio 4.1, p. 9: Connection Service The Connection Service enables a system administrator to connect the BlackBerry MDS Services to the Mobile Data Service feature of the BlackBerry Enterprise Server®. The Connection Service is designed to accept and respond to push requests from server-side push applications when the application server is behind the corporate firewall. This service is designed to provide a link to standard servers on the corporate intranet or Internet using standard Internet protocol, such as HTTP or TCP/IP, and encrypt content using the same encryption standard that is used to encrypt BlackBerry messages and other BlackBerry data. • MDS 4.1 Overview, p. 14: Security BlackBerry MDS Studio application data messages are sent using the data encryption technologies that are used by the BlackBerry Enterprise Server. Application data can also be encrypted on the device using the content protection mechanisms that are available on BlackBerry devices.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 287 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	See the *BlackBerry Enterprise Solution Security Technical Overview* for more information on BlackBerry security features. In addition to message and data content security, BlackBerry MDS Studio applications provide the following security-related features: • Application signing: The BlackBerry MDS Studio applications can be signed using the industry-standard (RSA) security certificate. Access to the BlackBerry MDS Studio repository is password-protected. • Container-based execution environment: The BlackBerry MDS Runtime is a containerbased execution environment that is designed to prevent unauthorized access to native APIs on the BlackBerry device. BlackBerry MDS Studio applications are prevented from accessing other BlackBerry MDS Studio application data, and one BlackBerry MDS Studio application cannot run or load another. • Secure connectivity to back-end systems: BlackBerry MDS Services support HTTPS for secure access to back-end web services. When this access is combined with the triple DES or AES encryption between the BlackBerry Enterprise Server and the BlackBerry device, a secure end-to-end link from the BlackBerry MDS Studio application to its related web services results. • BES 4.1 Technical Overview, p. 17: Standard BlackBerry message encryption Standard BlackBerry encryption is designed to encrypt messages that the BlackBerry device sends or that the BlackBerry Enterprise Server forwards to the BlackBerry device. Standard BlackBerry encryption encrypts the message • from the time a user sends an email message from the BlackBerry device until the BlackBerry Enterprise Server receives the message • from the time the BlackBerry Enterprise Server receives a message sent to a BlackBerry device user until that user reads the message on the BlackBerry device When a user sends a message from the BlackBerry device, the BlackBerry Enterprise Server does not encrypt the message when it forwards the message to the message recipient unless the BlackBerry device

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 288 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	user installs additional secure messaging technology on the BlackBerry device and the BlackBerry Enterprise Server administrator has enabled the BlackBerry device to use that secure messaging technology to extend the messaging security. • BES 4.1 Technical Overview, pp. 17-18: Receiving an email message on the BlackBerry device  Sending a message from a computer to the BlackBerry device 1. Alice sends a message to Bob from her computer. Alice and Bob work at the same organization. 2. The messaging server receives the email message and notifies the BlackBerry Enterprise Server that the message has arrived. 3. The messaging server delivers the message to Bob's computer. 4. The BlackBerry Enterprise Server retrieves the message from the messaging server. 5. The BlackBerry Enterprise Server queries the messaging server for BlackBerry device user preferences to determine whether or not to forward the message to Bob's BlackBerry device. 6. The BlackBerry Enterprise Server compresses and encrypts the message.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 289 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	7. The BlackBerry Enterprise Server places the message in the outgoing queue. The BlackBerry Enterprise Server is designed to maintain a constant, direct outbound TCP/IP connection to the wireless network over the Internet through the firewall on port 3101 (or 4101 in the case of a BlackBerry device that supports implementation alongside an enterprise Wi-Fi network). This constant connection enables the efficient, continuous delivery of data to and from the BlackBerry device. 8. The wireless network routes and then delivers the encrypted message to Bob's BlackBerry device. 9. Bob's BlackBerry device receives the encrypted message. The BlackBerry device then decrypts and displays the message for Bob to read. To the extent Plaintiff contends that Blackberry Push fails to disclose this limitation, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Shen, Houghton, TS.23-140. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 290 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 1, the link interface to encrypt messages identified for delivery to each given one of the wireless end-user devices to create secure Internet data messages (e.g., encrypting the entire message to create a secure link to the mobile device), the device link agent on each given device further configured to decrypt the received secure Internet data messages (e.g., decrypting messages) for that device prior to delivering those messages to a respective software process. *See, e.g.*, The repackaging also permits both messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed and decompressed." Lazaridis, 4:16-19. "In preparing the message for redirection, the redirector program 12 could compress the original message A, could compress the message header, or could encrypt the entire message to create a secure link to the mobile device 24." *Id.*, 7:53-56. After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 291 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "If the system includes encryption keys, these too can be kept at one place for management and update purposes." *Id.*, 9:31-33. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. **Combination with Shen** Shen and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Blackberry Push discloses encrypting push application content over the network. MDS 4.1 Overview, p. 5-6. Blackberry Push also discloses the Blackberry Enterprise Solution provides end-to-end encryption. MDS

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 292 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Technical Solution, p. 12. Shen discloses an authentication/key-management module for encrypting MMS messages. Shen, ¶¶17, 54-60. Shen also discloses that messages can be "encode[d] and decode[d] to protect the privacy of these messages." *Id*., ¶54. This combination of Blackberry Push and Shen would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Shen, as the communication protocol because Blackberry Push was a well-known, industry-leading system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Shen's MMS messaging system without requiring inventive skill. Blackberry Push discloses that the message link server (e.g., BES) performs encoding operations on content before delivering it through the transport services stack to wireless devices, including content encoding and compression for efficiency. Blackberry Push also discloses remote delivery to clients, which relates to the secure message link and device link agent concepts. MDS 4.1 Overview, p. 5-6. Shen discloses an authentication/key-management module for encrypting MMS messages. Shen, ¶¶17, 54-60. Shen also discloses that messages can be "encode[d] and decode[d] to protect the privacy of these messages." *Id*., ¶54. A POSITA would have been motivated to utilize Shen's encryption/decryption scheme to improve security by preventing unauthorized access to the messages stored on BES servers. This would have involved using a known technique (encryption) to improve similar devices (BES networks) in the same way (securing content). This combination of Blackberry Push and Shen would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Shen, as the communication protocol because Blackberry Push was a well-known, industry-leading system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Shen's MMS messaging system without requiring inventive skill. Shen, as evidenced by the example citations below, discloses an encrypt function to encrypt one or more messages supplied to the transport services stack for delivery on the secure message link maintained between the message link server and the device link agent on the given one of the wireless end-user devices. *See, e.g.*, "Therefore there is a potential security problem in this kind of systems since it is not an end-to-end solution

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 293 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	for enterprise mobile applications. Another problem is the interface between the MMS center and enterprise applications. Currently different MMS centers provide different APIs, which brings some troubles for service providers to connect those different operators. Furthermore, if an enterprise wants to connect to an operator's MMS center, it has to sign a contract with the operator to follow the operator's price and security policy. Maybe this also brings some inconvenience for enterprises." Shen, ¶4. "The invention also provides a MMS system supporting both telecom network and wireless LAN and having improved security. Thus, the invention is to setup a direct communication channel between the MMS user device and the enterprise applications. Because there is a way to bypass the operator's MMS Center without storing MMS messages on the MMS Center, so we can encrypt the communication channel between the client and the backend MMS server with some security protocols, e.g. WTLS (Wireless Transport Layer Security), to enhance the security for MMS applications. When the user moves into a wireless LAN enabled office and the user's MMS device can access enterprise intranet via wireless LAN, the backend server can detect the user's roaming and switch its working model from the telecom model to WLAN model. Therefore the user could use an MMS device to access MMS applications via WLAN in an enterprise." *Id*., ¶17. "The invention can provide an end-to-end security solution for MMS applications, especially for enterprise applications." *Id*., ¶22. "FIG. 2 is a block diagram showing a first embodiment of a MMS system, a MMS gateway 202 and a user device 204 according to the present invention. In the following description to the MMS system, gateway and user device, a method for receiving and sending multimedia message of the present invention is implied." *Id*., ¶29. "A concept of the present invention is to send and receive multimedia message over wireless LAN. That is, the multimedia messaging system of the present invention comprises a wireless LAN 206, a MMS user device 204 for sending and receiving multimedia messages over the wireless LAN, and a MMS gateway 202 for receiving, storing and sending multimedia messages over the wireless LAN. Thus, a direct MMS channel is established between the MMS user device 204 for sending and receiving multimedia messages

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 294 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	over the wireless LAN and the MMS gateway 202 for receiving, storing and sending multimedia messages over the wireless LAN. As a result, security is ensured and cost will be reduced because of using the wireless LAN. Therefore, to an enterprise, its mobile employees may use MMS user devices 204, such as mobile phones supporting wireless LAN, to interact with MMS applications 112 (communicatedwith an enterprise MMS gateway) in the enterprise." *Id*., ¶30. "Sending and receiving multimedia messages over the wireless LAN may be realized with any existing or future techniques. For example, the HTTP protocol, or any other protocols, may be used to send and/or receive multimedia messages. In the present specification including the accompanying drawings, as an example, the HTTP protocol is adopted in the wireless LAN. Obviously, this should not be interpreted as a limit to the scope of protection of the present invention." *Id*., ¶31. "The MMS user device 204 may comprise a message transceiver 210 and a notification receiver 212, and the MMS gateway 202 may comprises a notification sender 214 and a multimedia message storage/transceiver 216 (herein after referred to as 'MMS storage/transceiver'). The MMS storage/transceiver functions to receive, store and send multimedia messages and communicate with the backend MMS applications in the enterprise. The notification sender functions only to send MMS notifying message to the user device. When a MMS user device is to send a MMS message, the message transceiver sends a request (such as a HTTP request when the HTTP protocol is used) carrying the MMS message to the MMS storage/transceiver. When the MMS user device is to receive a MMS message, that is, when the MMS storage/transceiver is to send a message to the MMS user device, the MMS gateway first calls the notification sender 214 to send a MMS notifying message carrying location information of the MMS message to the MMS user device, the location information indicates the storage location of the MMS message in the MMS storage/transceiver, and is URL information when the HTTP protocol is used. Next, the MMS user device retrieves the location information such as URL from the MMS notifying message. Then the message transceiver sends a request (such as a HTTP request) carrying the location information to the MMS storage/transceiver. And last, the MMS storage/transceiver retrieves the MMS message and returns it to the MMS user device. The message is received by the message transceiver." *Id*., ¶32.

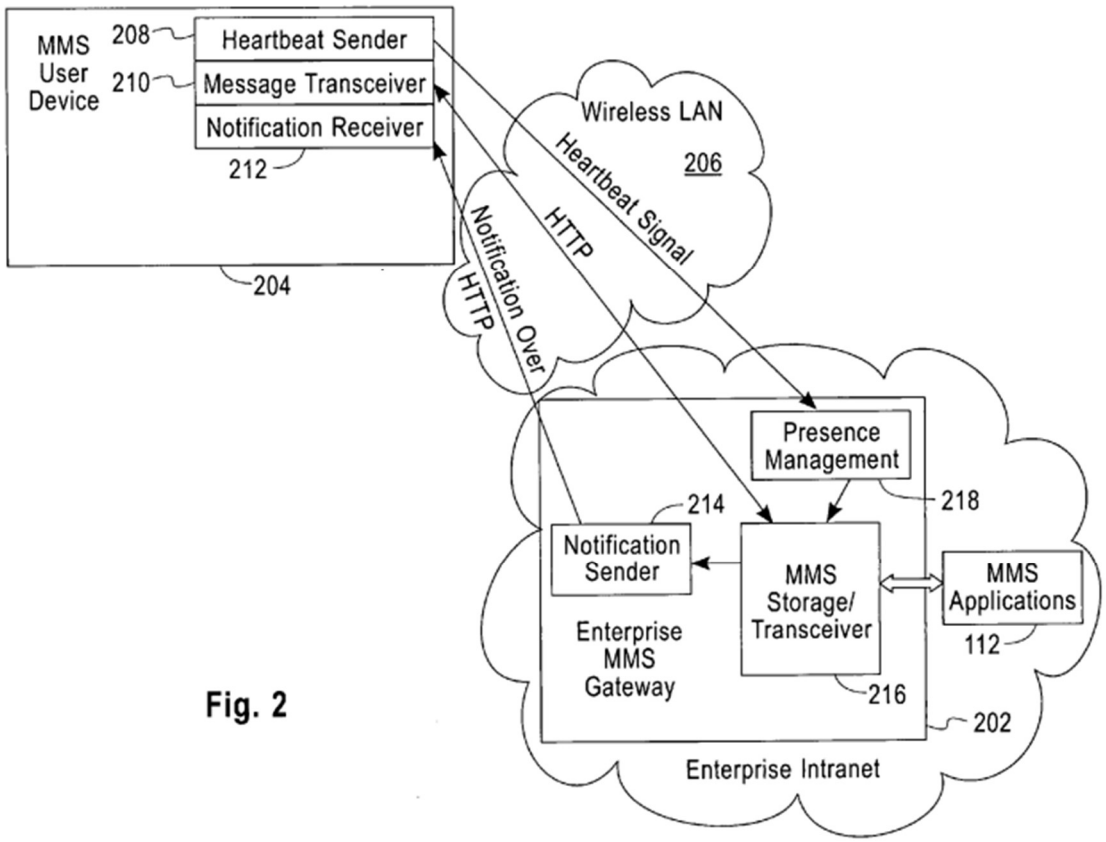
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 295 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"In order that the MMS user device is able to determine whether a wireless LAN is available, or in order that the MMS gateway is able to determine whether a MMS user device is reachable, in a preferred embodiment, the MMS user device may further comprise a heartbeat sender, and the MMS gateway may further comprises a presence manager." *Id*., ¶33. "In a wireless LAN environment, a MMS user device sends heartbeat signals periodically to the enterprise MMS gateway. The heartbeat signals contain information such as about user device number and IP address, which are also recorded in the database in the presence manager. If the gateway periodically receives the heartbeat signals, it means the corresponding user is located in the coverage of the wireless LAN, and the gateway may send a notification to the user when there is a message to be sent to that user; otherwise it means the user is located in the telecom network or offline, and the gateway does not send any notification to the user even if there is a MMS message to be sent to that user." *Id*., ¶34. "As shown in FIG. 5, in a fourth embodiment, an authentication and key management module 502 is added in the MMS gateway to distribute symmetric keys to users and a certificate management module 504 is added in the user device. Both the MMS gateway and the user device deploy the same symmetric cipher to encode and decode MMS messages to protect the privacy of these messages." *Id*., ¶54. "The encryption may be realized in any possible manner. The following is an example." *Id*., ¶55. "If the MMS user device has no certificate of the MMS gateway, it first sends a CA (certificate) requirement to the gateway. After the gateway confirms that the user has the access to send and receive MMS according to the user device number, the gateway's CA is sent to the user device. There are three methods to distribute a certificate to a user device: manually synchronize a certificate into the user device via an infrared port, or send a certificate via SMS or WAP response." *Id*., ¶56.

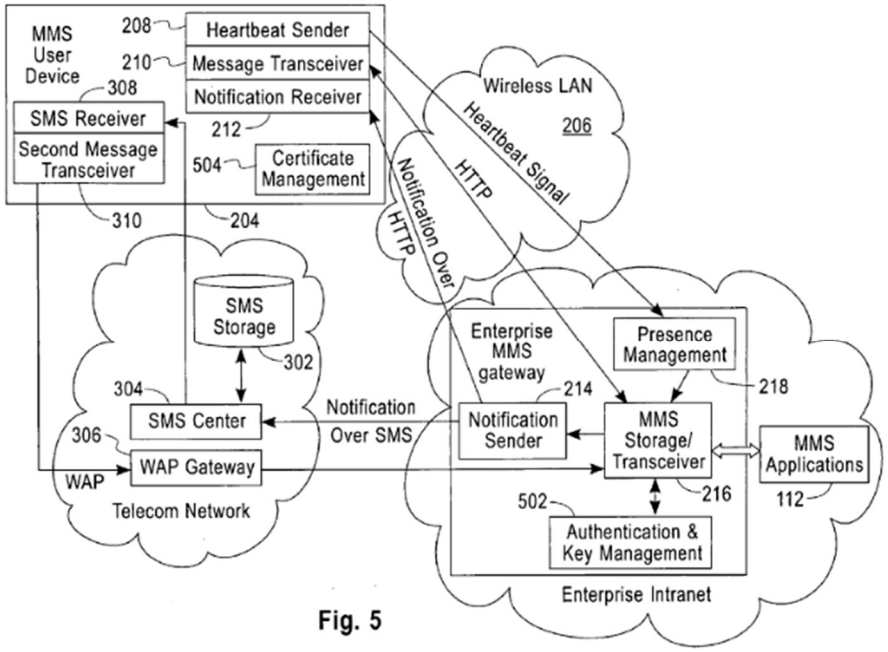
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 296 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The MMS user device sends a Master Key requirement to the gateway and the gateway creates a session ID and a master key, generally a random object, and sends them to the device via SMS or WAP response. The random object is encrypted by the gateway's private key." Shen, [0057]. "The random object is decoded by the user device and mixed with its phone number and session ID to generate a shared secret key. The mixing operation may be any operation, the simplest is XOR operation. At the same time, the gateway also generates a shared secret key with the same algorithm." *Id*., ¶58. "Both the MMS user device and the MMS gate way can secretly communicate with each other by encrypting MMS messages with the shared secret key." *Id*., ¶59. "In above encryption process, both the MMS user device and the MMS gateway can fresh the shared secret key by changing the session ID to prevent the threat of attacks stemming from key reuse." *Id*., ¶60.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 297 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 2 *Id.*, Fig. 2

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 298 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 5 *Id.*, Fig. 5.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 299 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Houghton Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id*., p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id*., p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id*., p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, discloses an encrypt function to encrypt one or

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 300 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	more messages supplied to the transport services stack for delivery on the secure message link maintained between the message link server and the device link agent on the given one of the wireless end-user devices. *See, e.g.* "A server outside a wireless network security entity is adapted to push messages to a mobile terminal located in the wireless network and thereby cause programs to start to a specified operating state or, if already started, change operating state to the state indicated by the message thereby presenting a new application view to the user of the mobile terminal. The mobile terminal is provided with means, such as software, which initiates and maintains contact with the server using Internet technologies. This mobile-initiated permanent connection allows the server to pass or push messages from outside the operator's firewall to the mobile terminal at any time." Houghton, Abstract. "The Internet is a collection of networks which exchange messages using IP protocol (Internet Protocol) and associated protocols which run on top of IP such as TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). Gateways between IP networks automatically route and transmit information over primarily the IP4 and IP6 protocols between two or more end points on any of the connected fixed and mobile networks. Each of these networks may also include or run on top of other network transport protocols such as ATM (Asynchronous Transfer Mode) or a wireless protocol which in turn carry the IP packets over that section of the Internet." *Id.*, p. 1. "Push technology is characterized by the server proactively sending documents or information to a client or user. E-mail (Electronic mail) is actually a special kind of push technology that has been prevalent for a long time. The first and most common general category of mobile content push technologies is SMS (Short Message Service) based technologies. SMS and MMS (Multi-media Messaging Service) are the most common forms of "push" messaging in the mobile industry. These store and forward messaging systems run by mobile telecommunications operators push all or part of the data to the mobile terminal over closed telecommunications signaling networks such as SS7 (Signaling System Seven). These signaling networks are closed; they are not part of the Internet. Each operator controls and encrypts this traffic, charging high

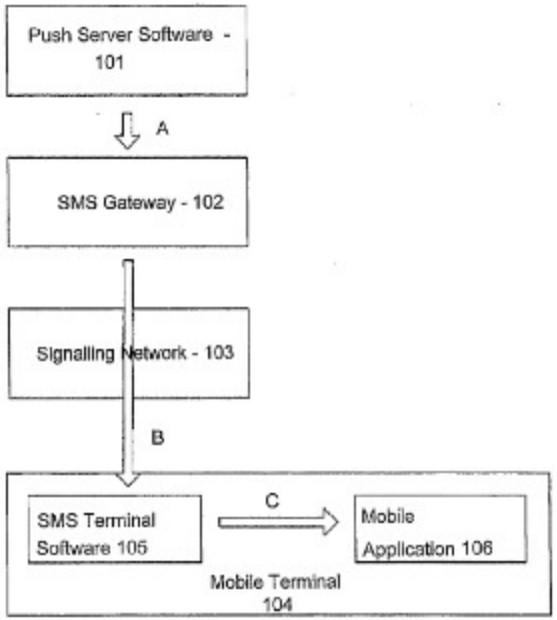
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 301 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	prices for their use and limiting their use and usefulness." *Id*., p. 3. "The mobile terminal is provided with means, such as software, which initiates and maintains contact with the server using Internet technologies, thereby avoiding cost, latency, contractual and firewall penetration issues of the server contacting the mobile terminals by telecommunications messaging or Internet technologies." *Id*., p. 11. "The push client software 405 preferably launches automatically when the mobile terminal 404 is turned on. The push client software 405 runs in the background and attempts to log into the push server 401 over a data network 403, normally a wireless network capable of transmitting Internet Protocol (IP) packets. The arrow A in Figure 4 illustrates an optional attempt by the push client 405 to communicate with the wireless network firewall 402 or other similar security or gateway device in order to open a connection for the push server 401 to directly contact the push client 401 as in FIG 3. If successful, operations can continue as described in FIG 3 or as below. Arrow B illustrates a login attempt." *Id*., p. 17. "Additionally, IP connection techniques, such as Internet Protocol 4 (IP4) network address translation and Internet Protocol 6 (IP6) security, encryption and/or authentication, may hide the actual address of the mobile terminal or make it otherwise difficult for a server to connect to directly to the mobile terminal." *Id*., p. 17. "As an example, such network parameters may include but are not limited to one or more of the following: - Use of a connection-oriented protocol suitable to the wireless network such as TCP/IP protocol, which is generally preferred in the common situation where an intermediate firewall, Network Address Translation (NAT) or gateway device 402. - Use of a connectionless protocol suitable to the wireless network 403, such as UDP/IP, which is generally preferred when no network device restricting terminal contact 402 is present as this reduces resource usage on the server when large number of mobile terminals is served. - Use of an alternate connection-oriented protocol such as HTTP suitable for mobile push operation through more strict firewalls and proxy Internet connections.

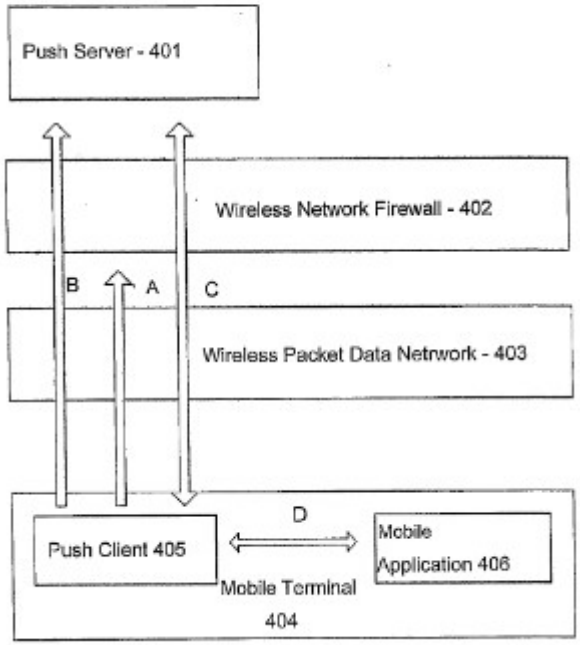
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 302 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	- Use of a secure protocol such as HTTPS, IP-Sec, secure IP6 or a proprietary security protocol to identify the communicating parties, prevent message interception by 3rd parties, and prevent message modification by 3rd parties." *Id*., p. 19. "At any time after a connection is established between the client 405 and the server 401, the client may push a message to the server or the server may send a push message to the client. This is illustrated by means of the arrow C in Figure 4. These messages through a previously established, connection-oriented protocol such as TCP/IP, SSL, HTTP or HTTPS arrive quite rapidly, more rapidly than if the connection where established only when needed. The associated protocol overhead such as 3-way handshake during connection establishment thus occurs before the actual need for message transmission rather than needing to occur at time or transmission and thus delaying the transmission." *Id*., p. 20. "A connectionless protocol, such as UDP/IP, may also used in cases where the connection-oriented protocol is not possible or not desirable because of the push server resources used by every static instance of a connection-oriented protocol. Connectionless communication is also used when a message needs to get through to the server 401 quickly, such as when occurs when the client 405 notifies the server 401 of an terminal event, user interface event or application event but the connection-oriented protocol connection is not connected or is already busy transferring information" *Id*., p. 20. "A method as claimed in claim 9, wherein said trying of one or more other techniques and/network parameters include one or more of using an alternate push server network address which may be more available or more accessible at the present time and from the present location, using one or more alternative mobile terminal network connection settings, or variation of the parameters within the settings, using an alternate connection route through a second wireless or fixed terminal accessible to the wireless terminal over a wireless network and capable of communication with the push server, using a TCP connection, using a UDP connection, using an SSL connection, using an HTTP or tunneling HTTP connection, using an SHTTP or tunneling HTTPS connection, using an XML-based connection over other transport protocol." *Id*., p. claim 10.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 303 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 1  *Id.*, Fig. 1.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 304 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id.*, Fig. 4.

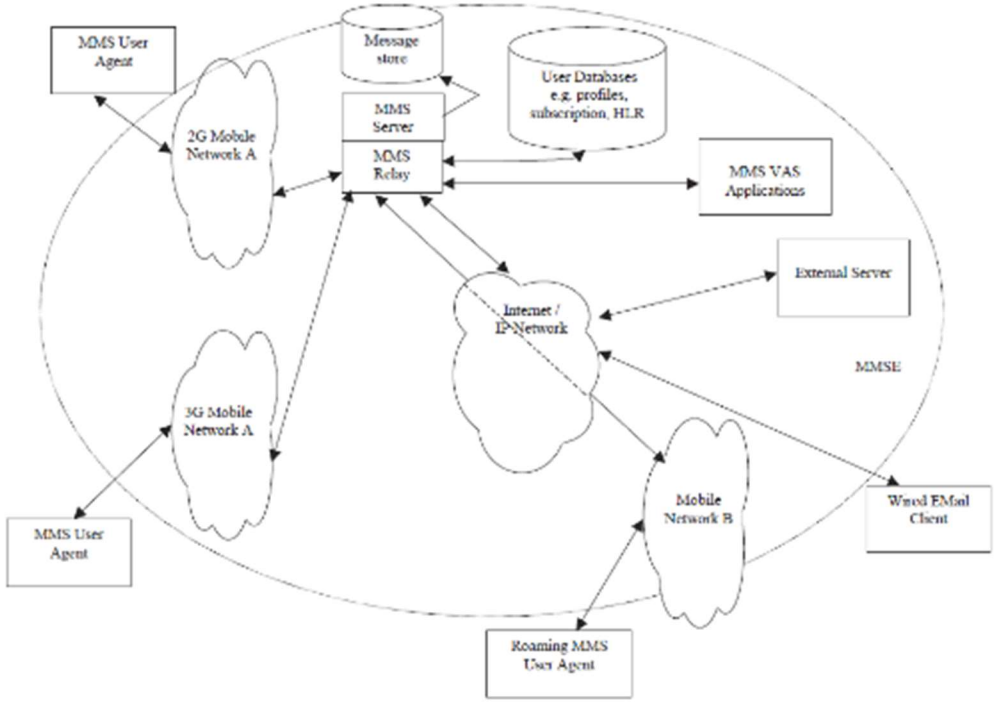
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 305 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with TS.23-140 TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS.23-140, as evidenced by the citations below, discloses an encrypt function to encrypt one or more

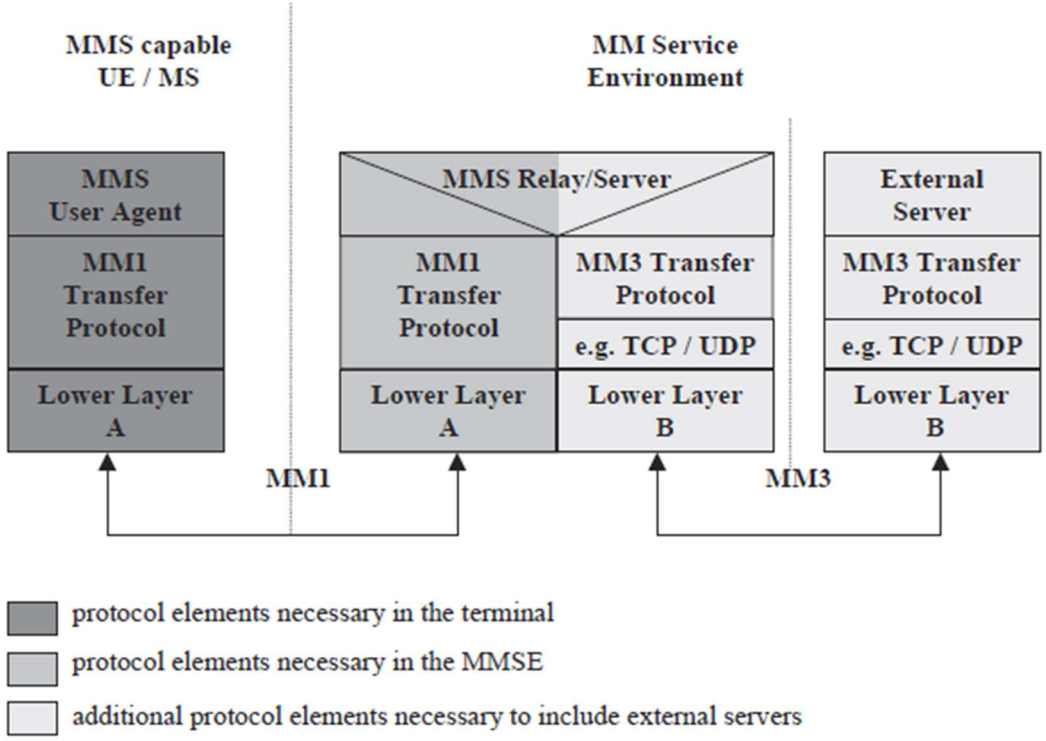
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 306 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	messages supplied to the transport services stack for delivery on the secure message link maintained between the message link server and the device link agent on the given one of the wireless end-user devices. *See e.g.*, "The MMS User Agent may provide additional application layer functionalities such as: . . . the decryption and encryption of an MM on an end-user to end-user basis" *Id*., p. 19. "Reference point MM1 is used to submit Multimedia Messages from MMS User Agent to MMS Relay/Server, to let the MMS User Agent pull MMs from the MMS Relay/Server, let the MMS Relay/Server push information about MMs to the MMS User Agent as part of an MM notification, and to exchange delivery reports between MMS Relay/Server and MMS User Agents." *Id*., p. 24. "The interface between MMS Relay/server and VASP may be carried over an encrypted and secure bearer, e.g. HTTP over SSL or TLS, or by use of application-layer encryption. This is an optional feature and may be further elaborated in future releases." *Id*., p. 41.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 307 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 2: MMS Architectural Elements** *Id.*, Fig. 2.

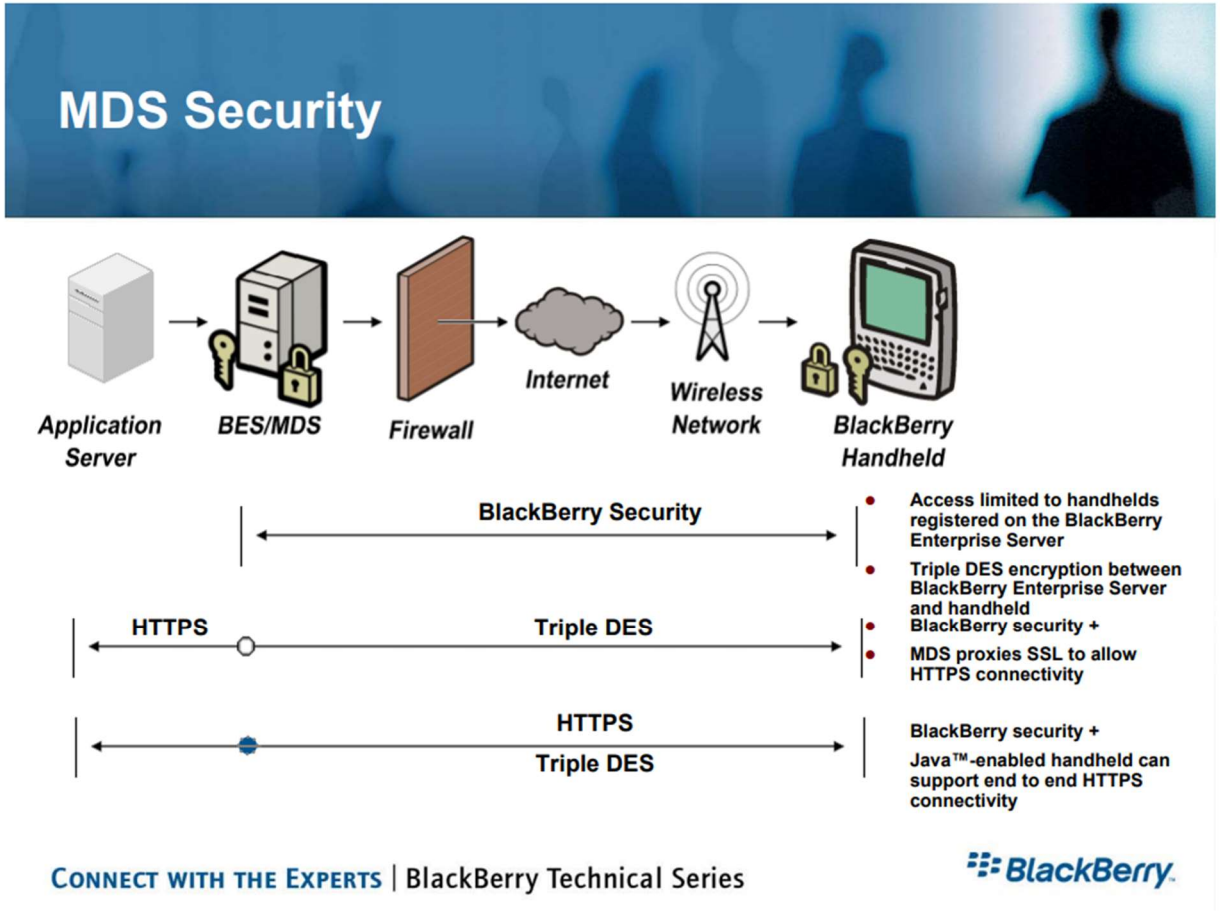
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 308 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 4: Protocol Framework to provide MMS** *Id.*, Fig. 4.
13: The networked system of	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 12,

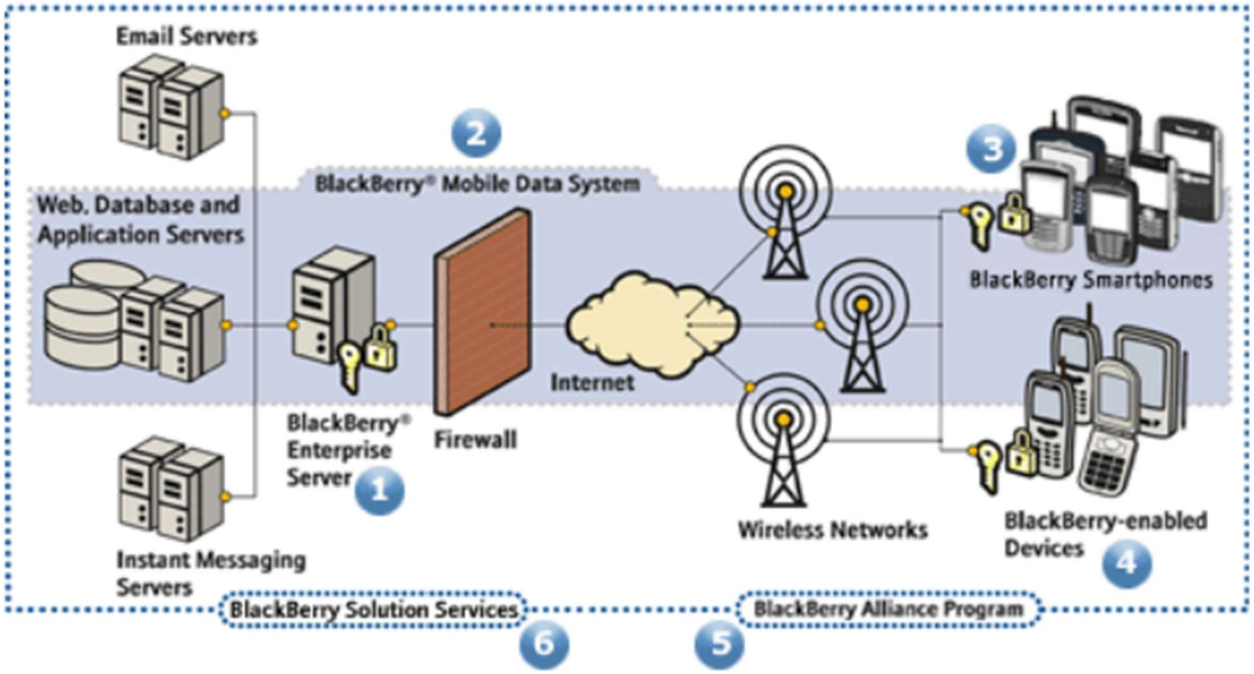
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 309 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
claim 12, wherein the encrypted messages are transported to the device link agent using one or more of encryption on a transport services stack, IP (Internet Protocol) layer encryption, and tunneling.	wherein the encrypted messages are transported to the device link agent using one or more of encryption on a transport services stack, IP (Internet Protocol) layer encryption, and tunneling. Blackberry Push transports encrypted messages to the device link agent on Blackberry devices by encrypting the transport services stack. Blackberry Push provides an SRP-based BES link with 3DES/AES and supports SSL/TLS for HTTP(S) sessions proxied by MDS and/or end-to-end HTTPS from device to server. *See, e.g.*, • MDS Technical Series, p. 12:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 310 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 311 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• BES Architecture Overview, p. 1:  *See also supra*, claims 2, 12. To the extent Plaintiff contends that Blackberry Push fails to disclose this limitation, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Shen, Houghton, TS.23-140.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 312 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Lazaridis Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the message link server of claim 2, wherein the encrypted one or more messages are transported to the device link agent on the given one of the wireless end-user devices using one or more of encryption on the transport services stack, IP (Internet Protocol) layer encryption, and tunneling (e.g., encrypting the entire message to create a secure link to the mobile device). *See, e.g.*,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 313 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The repackaging also permits both messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed and decompressed." Lazaridis, 4:16-19. "In preparing the message for redirection, the redirector program 12 could compress the original message A, could compress the message header, or could encrypt the entire message to create a secure link to the mobile device 24." *Id.*, 7:53-56. After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*8:32-55. "If the system includes encryption keys, these too can be kept at one place for management and update purposes." *Id.*, 9:31-33. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 314 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. **Combination with Shen** Shen and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Shen and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. Shen contemplates using existing or future protocols, such as HTTP or WAP, for sending and receiving MMS messages, making the integration of Blackberry Push a predictable solution to enable MMS over telecom networks. Shen, ¶42. A POSITA would have found it obvious to combine Shen's MMS system with Blackberry Push architecture. A POSITA would have been motivated to combine Shen's enterprise-focused MMS architecture (Shen, ¶2) with Blackberry Push's mobile communication system to support messaging across both wireless LAN and telecom networks. This combination of Blackberry Push and Shen would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Shen, as the communication protocol because Blackberry Push was a well-known, industry-leading system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Shen's MMS messaging system without requiring inventive skill. A POSITA would have been motivated to utilize Shen's encryption/decryption scheme to improve security by preventing unauthorized access to the messages stored on BES servers. This would have involved using a known technique (encryption) to improve similar devices (BES networks) in the same way (securing content). Shen, as evidenced by the example citations below, discloses an encrypt function to encrypt one or more

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 315 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	messages supplied to the transport services stack for delivery on the secure message link maintained between the message link server and the device link agent on the given one of the wireless end-user devices. *See, e.g.*, "Therefore there is a potential security problem in this kind of systems since it is not an end-to-end solution for enterprise mobile applications. Another problem is the interface between the MMS center and enterprise applications. Currently different MMS centers provide different APIs, which brings some troubles for service providers to connect those different operators. Furthermore, if an enterprise wants to connect to an operator's MMS center, it has to sign a contract with the operator to follow the operator's price and security policy. Maybe this also brings some inconvenience for enterprises." Shen, ¶4. "The invention also provides a MMS system supporting both telecom network and wireless LAN and having improved security. Thus, the invention is to setup a direct communication channel between the MMS user device and the enterprise applications. Because there is a way to bypass the operator's MMS Center without storing MMS messages on the MMS Center, so we can encrypt the communication channel between the client and the backend MMS server with some security protocols, e.g. WTLS (Wireless Transport Layer Security), to enhance the security for MMS applications. When the user moves into a wireless LAN enabled office and the user's MMS device can access enterprise intranet via wireless LAN, the backend server can detect the user's roaming and switch its working model from the telecom model to WLAN model. Therefore the user could use an MMS device to access MMS applications via WLAN in an enterprise." *Id*., ¶17. "FIG. 2 is a block diagram showing a first embodiment of a MMS system, a MMS gateway 202 and a user device 204 according to the present invention. In the following description to the MMS system, gateway and user device, a method for receiving and sending multimedia message of the present invention is implied. A concept of the present invention is to send and receive multimedia message over wireless LAN. That is, the multimedia messaging system of the present invention comprises a wireless LAN 206, a MMS user device 204 for sending and receiving multimedia messages over the wireless LAN, and a MMS gateway 202 for receiving, storing and sending multimedia messages over the wireless LAN. Thus, a direct MMS channel is established between the MMS user device 204 for sending and receiving multimedia messages

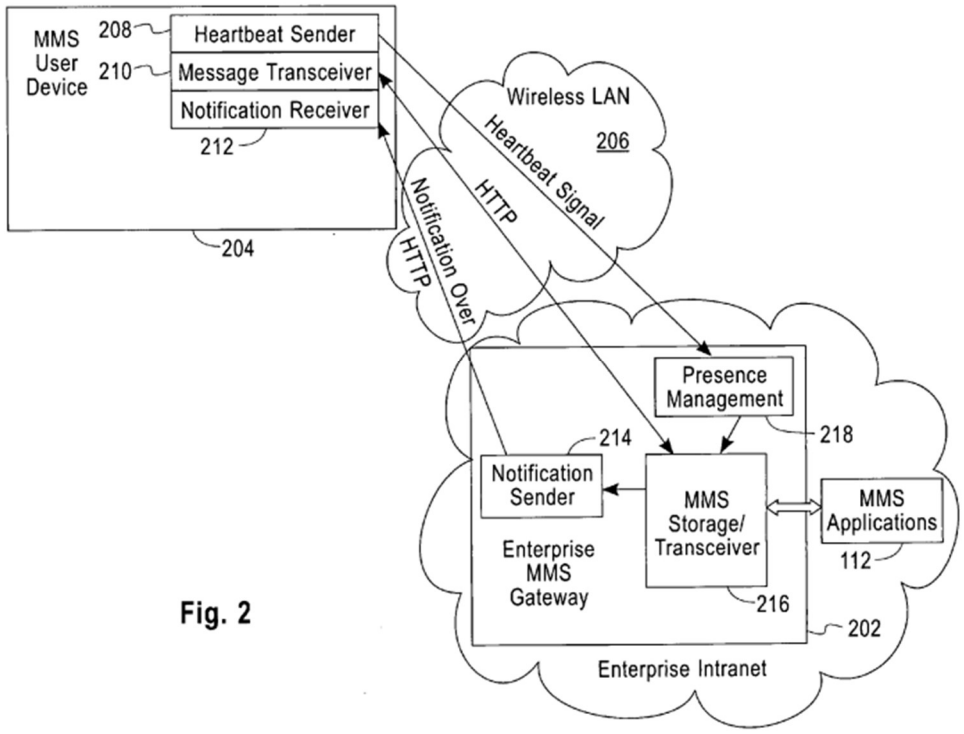
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 316 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	over the wireless LAN and the MMS gateway 202 for receiving, storing and sending multimedia messages over the wireless LAN. As a result, security is ensured and cost will be reduced because of using the wireless LAN. Therefore, to an enterprise, its mobile employees may use MMS user devices 204, such as mobile phones supporting wireless LAN, to interact with MMS applications 112 (communicated with an enterprise MMS gateway) in the enterprise. Sending and receiving multimedia messages over the wireless LAN may be realized with any existing or future techniques. For example, the HTTP protocol, or any other protocols, may be used to send and/or receive multimedia messages. In the present specification including the accompanying drawings, as an example, the HTTP protocol is adopted in the wireless LAN. Obviously, this should not be interpreted as a limit to the scope of protection of the present invention. The MMS user device 204 may comprise a message transceiver 210 and a notification receiver 212, and the MMS gateway 202 may comprises a notification sender 214 and a multimedia message storage/transceiver 216 (hereinafter referred to as "MMS storage/transceiver"). The MMS storage/transceiver functions to receive, store and send multimedia messages and communicate with the backend MMS applications in the enterprise. The notification sender functions only to send MMS notifying message to the user device. When a MMS user device is to send a MMS message, the message transceiver sends a request (such as a HTTP request when the HTTP protocol is used) carrying the MMS message to the MMS storage/transceiver. When the MMS user device is to receive a MMS message, that is, when the MMS storage/transceiver is to send a message to the MMS user device, the MMS gateway first calls the notification sender 214 to send a MMS notifying message carrying location information of the MMS message to the MMS user device, the location information indicates the storage location of the MMS message in the MMS storage/transceiver, and is URL information when the HTTP protocol is used. Next, the MMS user device retrieves the location information such as URL from the MMS notifying message. Then the message transceiver sends a request (such as a HTTP request) carrying the location information to the MMS storage/transceiver. And last, the MMS storage/transceiver retrieves the MMS message and returns it to the MMS user device. The message is received by the message transceiver.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 317 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	In order that the MMS user device is able to determine whether a wireless LAN is available, or in order that the MMS gateway is able to determine whether a MMS user device is reachable, in a preferred embodiment, the MMS user device may further comprise a heartbeat sender, and the MMS gateway may further comprises a presence manager. In a wireless LAN environment, a MMS user device sends heartbeat signals periodically to the enterprise MMS gateway. The heartbeat signals contain information such as about user device number and IP address, which are also recorded in the database in the presence manager. If the gateway periodically receives the heartbeat signals, it means the corresponding user is located in the coverage of the wireless LAN, and the gateway may send a notification to the user when there is a message to be sent to that user; otherwise it means the user is located in the telecom network or offline, and the gateway does not send any notification to the user even if there is a MMS message to be sent to that user." *Id*., ¶¶29-34.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 318 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 2 *Id.*, Fig. 2 "The encryption may be realized in any possible manner. The following is an example. 1. If the MMS user device has no certificate of the MMS gateway, it first sends a CA (certificate) requirement to the gateway. After the gateway confirms that the user has the access to send and receive

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 319 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	MMS according to the user device number, the gateway's CA is sent to the user device. There are three methods to distribute a certificate to a user device: manually synchronize a certificate into the user device via an infrared port, or send a certificate via SMS or WAP response. 2. The MMS user device sends a Master Key requirement to the gateway and the gateway creates a session ID and a master key, generally a random object, and sends them to the device via SMS or WAP response. The random object is encrypted by the gateway's private key. 3. The random object is decoded by the user device and mixed with its phone number and session ID to generate a shared secret key. The mixing operation may be any operation, the simplest is XOR operation. At the same time, the gateway also generates a shared secret key with the same algorithm. 4. Both the MMS user device and the MMS gateway can secretly communicate with each other by encrypting MMS messages with the shared secret key. In above encryption process, both the MMS user device and the MMS gateway can fresh the shared secret key by changing the session ID to prevent the threat of attacks stemming from key reuse." *Id*., ¶¶55-60. **Combination with Houghton** Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 320 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id*., p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id*., p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id*., p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, discloses the message link server of claim 2, wherein the encrypted one or more messages are transported to the device link agent (e.g., a push client) on the given one of the wireless end-user devices (e.g., mobile terminals) using one or more of encryption on the transport services stack, IP (Internet Protocol) layer encryption, and tunneling (e.g., the push server using TCP/IP transport protocols with SSL/HTTPS encryption). *See* disclosures for claim 2, above. In addition, *see, e.g.*, "A server outside a wireless network security entity is adapted to push messages to a mobile terminal located in the wireless network and thereby cause programs to start to a specified operating state or, if already started, change operating state to the state indicated by the message thereby presenting a new application view to the user of the mobile terminal. The mobile terminal is provided with means, such as software,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 321 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	which initiates and maintains contact with the server using Internet technologies. This mobile-initiated permanent connection allows the server to pass or push messages from outside the operator's firewall to the mobile terminal at any time." Houghton, Abstract. "The Internet is a collection of networks which exchange messages using IP protocol (Internet Protocol) and associated protocols which run on top of IP such as TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). Gateways between IP networks automatically route and transmit information over primarily the IP4 and IP6 protocols between two or more end points on any of the connected fixed and mobile networks. Each of these networks may also include or run on top of other network transport protocols such as ATM (Asynchronous Transfer Mode) or a wireless protocol which in turn carry the IP packets over that section of the Internet." *Id*., p. 1. "Push technology is characterized by the server proactively sending documents or information to a client or user. E-mail (Electronic mail) is actually a special kind of push technology that has been prevalent for a long time. The first and most common general category of mobile content push technologies is SMS (Short Message Service) based technologies. SMS and MMS (Multi-media Messaging Service) are the most common forms of "push" messaging in the mobile industry. These store and forward messaging systems run by mobile telecommunications operators push all or part of the data to the mobile terminal over closed telecommunications signaling networks such as SS7 (Signaling System Seven). These signaling networks are closed; they are not part of the Internet. Each operator controls and encrypts this traffic, charging high prices for their use and limiting their use and usefulness." *Id*., p. 3. "The mobile terminal is provided with means, such as software, which initiates and maintains contact with the server using Internet technologies, thereby avoiding cost, latency, contractual and firewall penetration issues of the server contacting the mobile terminals by telecommunications messaging or Internet technologies." *Id*., p. 11. "The push client software 405 preferably launches automatically when the mobile terminal 404 is turned on. The push client software 405 runs in the background and attempts to log into the push server 401 over a data network 403, normally a wireless network capable of transmitting Internet Protocol (IP) packets. The arrow

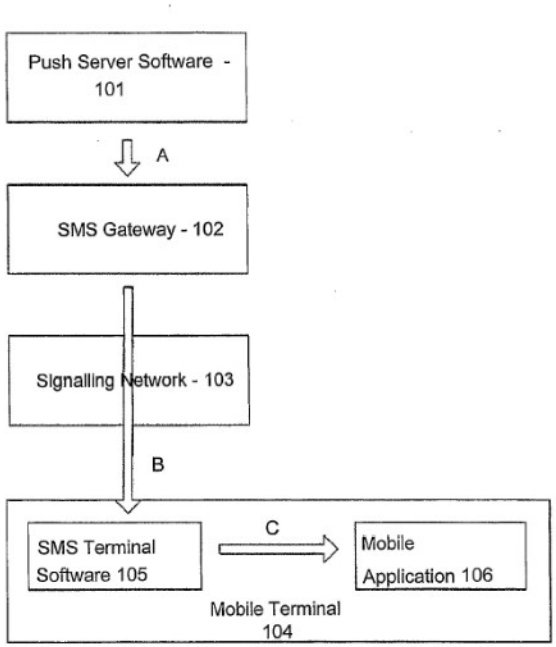
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 322 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	A in Figure 4 illustrates an optional attempt by the push client 405 to communicate with the wireless network firewall 402 or other similar security or gateway device in order to open a connection for the push server 401 to directly contact the push client 401 as in FIG 3. If successful, operations can continue as described in FIG 3 or as below. Arrow B illustrates a login attempt." *Id*., p. 17. "Additionally, IP connection techniques, such as Internet Protocol 4 (IP4) network address translation and Internet Protocol 6 (IP6) security, encryption and/or authentication, may hide the actual address of the mobile terminal or make it otherwise difficult for a server to connect to directly to the mobile terminal." *Id*., p. 17. "As an example, such network parameters may include but are not limited to one or more of the following: - Use of a connection-oriented protocol suitable to the wireless network such as TCP/IP protocol, which is generally preferred in the common situation where an intermediate firewall, Network Address Translation (NAT) or gateway device 402. - Use of a connectionless protocol suitable to the wireless network 403, such as UDP/IP, which is generally preferred when no network device restricting terminal contact 402 is present as this reduces resource usage on the server when large number of mobile terminals is served. - Use of an alternate connection-oriented protocol such as HTTP suitable for mobile push operation through more strict firewalls and proxy Internet connections. - Use of a secure protocol such as HTTPS, IP-Sec, secure IP6 or a proprietary security protocol to identify the communicating parties, prevent message interception by 3rd parties, and prevent message modification by 3rd parties." Houghton, 19. "At any time after a connection is established between the client 405 and the server 401, the client may push a message to the server or the server may send a push message to the client. This is illustrated by means of the arrow C in Figure 4. These messages through a previously established, connection-oriented protocol such as TCP/IP, SSL, HTTP or HTTPS arrive quite rapidly, more rapidly than if the connection where established only when needed. The associated protocol overhead such as 3-way handshake during connection establishment thus occurs before the actual need for message transmission rather than needing

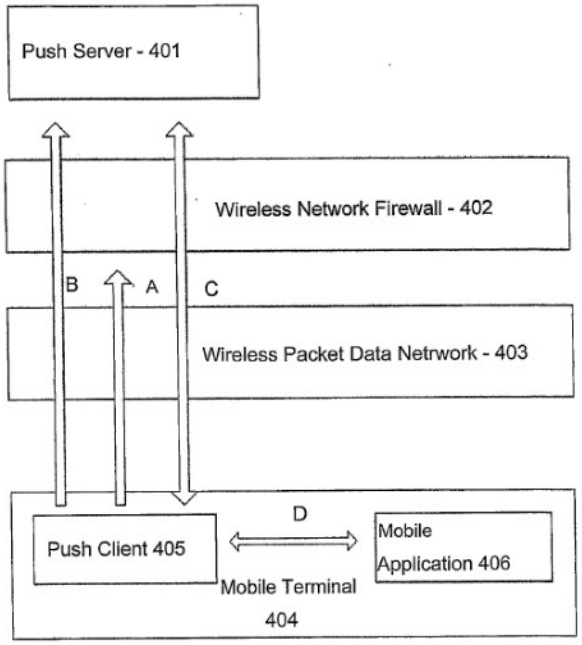
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 323 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	to occur at time or transmission and thus delaying the transmission." *Id*., p. 20. "A connectionless protocol, such as UDP/IP, may also used in cases where the connection-oriented protocol is not possible or not desirable because of the push server resources used by every static instance of a connection-oriented protocol. Connectionless communication is also used when a message needs to get through to the server 401 quickly, such as when occurs when the client 405 notifies the server 401 of an terminal event, user interface event or application event but the connection-oriented protocol connection is not connected or is already busy transferring information" *Id*., p. 20. "A method as claimed in claim 9, wherein said trying of one or more other techniques and/network parameters include one or more of using an alternate push server network address which may be more available or more accessible at the present time and from the present location, using one or more alternative mobile terminal network connection settings, or variation of the parameters within the settings, using an alternate connection route through a second wireless or fixed terminal accessible to the wireless terminal over a wireless network and capable of communication with the push server, using a TCP connection, using a UDP connection, using an SSL connection, using an HTTP or tunneling HTTP connection, using an SHTTP or tunneling HTTPS connection, using an XML-based connection over other transport protocol." *Id*., claim 10.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 324 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 1  *Id.*, Fig. 1.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 325 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id.*, Fig. 4.

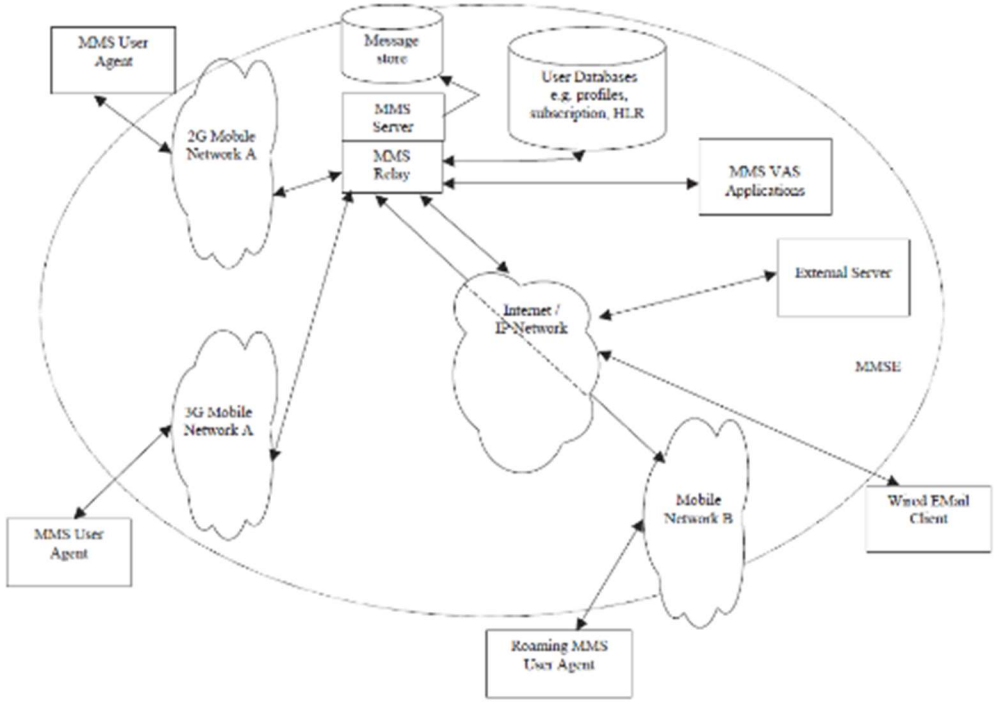
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 326 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with TS.23-140 TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS.23-140, as evidenced by the citations below, discloses wherein the encrypted one or more messages are

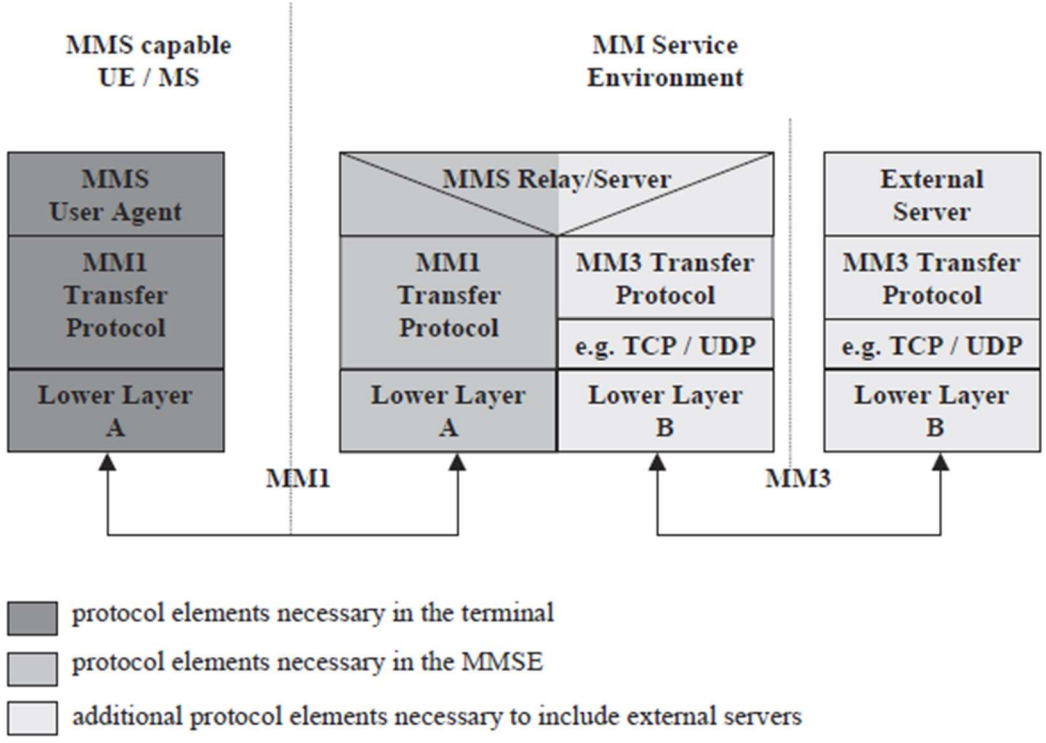
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 327 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	transported to the device link agent on the given one of the wireless end-user devices using one or more of encryption on the transport services stack, IP (Internet Protocol) layer encryption, and tunneling (e.g., using MM1 Transfer Protocol with TCP/IP and TLS). *See, e.g.*, "The MMS User Agent may provide additional application layer functionalities such as: . . . the decryption and encryption of an MM on an end-user to end-user basis" *Id*., p. 19. "Reference point MM1 is used to submit Multimedia Messages from MMS User Agent to MMS Relay/Server, to let the MMS User Agent pull MMs from the MMS Relay/Server, let the MMS Relay/Server push information about MMs to the MMS User Agent as part of an MM notification, and to exchange delivery reports between MMS Relay/Server and MMS User Agents." *Id*., p. 24. "The interface between MMS Relay/server and VASP may be carried over an encrypted and secure bearer, e.g. HTTP over SSL or TLS, or by use of application-layer encryption. This is an optional feature and may be further elaborated in future releases." *Id*., p. 41.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 328 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 2: MMS Architectural Elements** *Id.*, Fig. 2.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 329 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 4: Protocol Framework to provide MMS** *Id.*, Fig. 4.
14: The networked system of claim 1, wherein the device link agent on a given device	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein the device link agent on a given device executes in a secure execution environment, and at least one

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 330 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
executes in a secure execution environment, and at least one of the software components on that device executes outside of the secure execution environment.	of the software components on that device executes outside of the secure execution environment. Blackberry Push separates the secure transport/agent layer from application components. The device link agents operate under BlackBerry's secure device environment, while third-party applications and MDS Applications execute as software components authorized by IT policy. *See, e.g.*, • MDS Studio 4.1, p. 9: Connection Service The Connection Service enables a system administrator to connect the BlackBerry MDS Services to the Mobile Data Service feature of the BlackBerry Enterprise Server®. The Connection Service is designed to accept and respond to push requests from server-side push applications when the application server is behind the corporate firewall. This service is designed to provide a link to standard servers on the corporate intranet or Internet using standard Internet protocol, such as HTTP or TCP/IP, and encrypt content using the same encryption standard that is used to encrypt BlackBerry messages and other BlackBerry data. Application Integration Service The Application Integration Service is designed to support web services and other standard mechanisms for integrating wireless applications with existing enterprise applications and systems. This service is designed to manage the transmission of application data messages between BlackBerry MDS Applications and data sources. Provisioning Service The Provisioning Service is designed to control which BlackBerry MDS Applications users can download to their BlackBerry devices, support application discovery from a BlackBerry device, and manage wireless transmission and on BlackBerry devices. • MDS Applications, p. 1: More flexible deployments

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 331 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	With BlackBerry Enterprise Server for MDS Applications, you can now choose how to make the most efficient and effective use of your resources by deploying both wireless email and applications or wireless applications only. BlackBerry Enterprise Server for MDS Applications also provides you with the flexibility to deploy BlackBerry smartphones to business partners and customers that need to use your applications but shouldn't be tied to your corporate email server. … Features Functionally robust BlackBerry® Enterprise Server for MDS Applications enables IT departments to manage and control the deployment of mobile applications with a centralized administration console. BlackBerry Enterprise Server for MDS Applications includes the following: Wireless transport and synchronization: BlackBerry Enterprise Server for MDS Applications acts as the centralized link between wireless BlackBerry smartphones, wireless networks and enterprise applications, including all the standard BlackBerry® Mobile Data System (BlackBerry MDS) features Document viewing: Enabled by the BlackBerry® Browser IT policies: Control which applications users can access and implement IT policies on security and manageability Security: BlackBerry Enterprise Server for MDS Applications leverages the same accredited security model as the traditional BlackBerry® Enterprise Server. It creates an end-to-end authenticated, outbound-initiated connection that encrypts all data Push services: Increase the accessibility of enterprise applications and provide real-time data to mobile workforces by proactively sending data to them • MDS Technical Series, p. 6: MDS can be used by: – 3rd party applications and browsers – ISVs, SIs – Enterprise developers

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 332 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	To the extent Plaintiff contends that Blackberry Push fails to disclose this limitation, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Houghton, TS.23-140, Ellison. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 333 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 1, wherein the device link agent on a given device executes in a secure execution environment (e.g., encrypting the messages to create a secure link to the mobile device), and at least one of the software components executes outside of the secure execution environment on that device. *See, e.g.*, "There remains still another need for such a system and method that provides for secure, transparent delivery of the user-selected data items from the host system to the mobile device." Lazaridis, 2:42-45. "In an alternative system and method, the redirector program executes on a network server, and the server is programmed to detect numerous redirection event triggers over the network from multiple user desktop computers coupled to the server via a LAN. The server can receive internal event triggers from each of the user desktops via the network, and can also receive external event triggers, such as messages from the users' mobile data communication devices. In response to receiving one of these triggers, the server redirects the user's data items to the proper mobile data communication device. The user data items and addressing information for a particular mobile device can be stored at the server or at the user's PC. Using this alternative configuration, one redirector program can serve a plurality of users. This alternative configuration could also include an internet- or intranet-based redirector program that could be accessible through a secure webpage or other user interface. The redirector program could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. "The redirection program could be configured to send the text of the E-mail to the remote device, to send the word processing document to a networked printer located near the user, to send the video clip to a store accessible through a secure connection through the internet and to send the audio clip to the user's voice mail system." *Id.*, 6:22-29.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 334 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The repackaging also permits both messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed and decompressed." *Id.*, 4:16-19. "In preparing the message for redirection, the redirector program 12 could compress the original message A, could compress the message header, or could encrypt the entire message to create a secure link to the mobile device 24." *Id.*, 7:53-56. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "If the system includes encryption keys, these too can be kept at one place for management and update purposes." *Id.*, 9:31-33. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 335 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. **Combination with Houghton** Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 336 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. Houghton, as evidenced by the example citations below, discloses wherein the device link agent executes in a secure execution environment on at least one of the devices (e.g., secure push server protocols such as SSL and HTTPS), and at least one of the software components executes outside of the secure execution environment on that device. *See, e.g.*, "A server outside a wireless network security entity is adapted to push messages to a mobile terminal located in the wireless network and thereby cause programs to start to a specified operating state or, if already started, change operating state to the state indicated by the message thereby presenting a new application view to the user of the mobile terminal. The mobile terminal is provided with means, such as software, which initiates and maintains contact with the server using Internet technologies. This mobile-initiated permanent connection allows the server to pass or push messages from outside the operator's firewall to the mobile terminal at any time." Houghton, Abstract. "The Internet is a collection of networks which exchange messages using IP protocol (Internet Protocol) and associated protocols which run on top of IP such as TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). Gateways between IP networks automatically route and transmit information over primarily the IP4 and IP6 protocols between two or more end points on any of the connected fixed and mobile networks. Each of these networks may also include or run on top of other network transport protocols such as ATM (Asynchronous Transfer Mode) or a wireless protocol which in turn carry the IP packets over that section of the Internet." *Id*., p. 1. "The mobile terminal is provided with means, such as software, which initiates and maintains contact with the server using Internet technologies, thereby avoiding cost, latency, contractual and firewall penetration issues of the server contacting the mobile terminals by telecommunications messaging or Internet

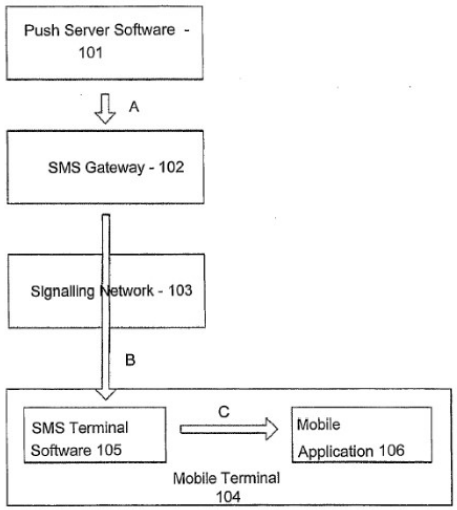
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 337 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	technologies." *Id*., p. 11. "Figure 4 illustrates an exemplary implementation of the system for pushing application commands to a mobile terminal based on server side and client side events in accordance with one embodiment of the present invention. More specifically, Figure 4 illustrates a Push Client basic functionality 405 implemented in a mobile terminal 404 as part of a message passing system in cooperation with a Push Server 401. The push client 405 is preferably implemented in the form of a software run in a processor of the mobile terminal 404. The mobile terminal 404 is preferably a commercially available mobile terminal, such as a portable phone or appliance capable of running software or personal digital assistant, which provides a platform for downloading application software over air or from a personal computer, installing the software in the terminal, and running the software in the terminal." *Id*., p. 16. "The push client software 405 preferably launches automatically when the mobile terminal 404 is turned on. The push client software 405 runs in the background and attempts to log into the push server 401 over a data network 403, normally a wireless network capable of transmitting Internet Protocol (IP) packets. The arrow A in Figure 4 illustrates an optional attempt by the push client 405 to communicate with the wireless network firewall 402 or other similar security or gateway device in order to open a connection for the push server 401 to directly contact the push client 401 as in FIG 3. If successful, operations can continue as described in FIG 3 or as below. Arrow B illustrates a login attempt." *Id*., p. 17. "As an example, such network parameters may include but are not limited to one or more of the following: - Use of a connection-oriented protocol suitable to the wireless network such as TCP/IP protocol, which is generally preferred in the common situation where an intermediate firewall, Network Address Translation (NAT) or gateway device 402. - Use of a connectionless protocol suitable to the wireless network 403, such as UDP/IP, which is generally preferred when no network device restricting terminal contact 402 is present as this reduces resource usage on the server when large number of mobile terminals is served. - Use of an alternate connection-oriented protocol such as HTTP suitable for mobile push operation through more strict firewalls and proxy Internet connections.

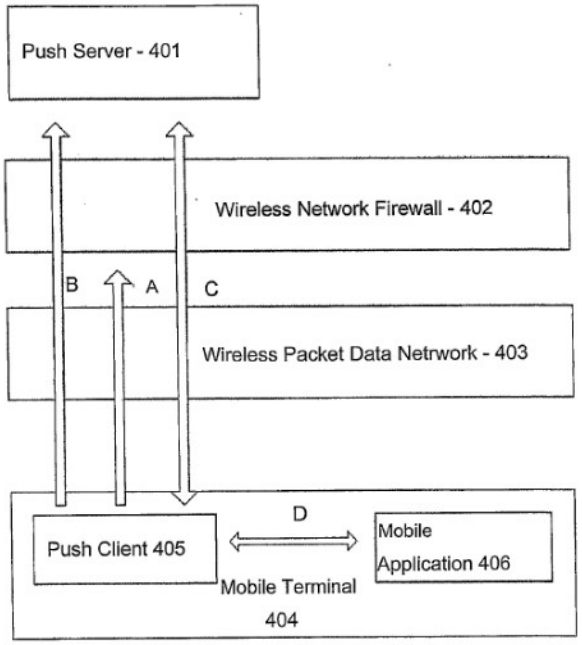
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 338 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	- Use of a secure protocol such as HTTPS, IP-Sec, secure IP6 or a proprietary security protocol to identify the communicating parties, prevent message interception by 3rd parties, and prevent message modification by 3rd parties." *Id*., p. 19. "At any time after a connection is established between the client 405 and the server 401, the client may push a message to the server or the server may send a push message to the client. This is illustrated by means of the arrow C in Figure 4. These messages through a previously established, connection-oriented protocol such as TCP/IP, SSL, HTTP or HTTPS arrive quite rapidly, more rapidly than if the connection where established only when needed. The associated protocol overhead such as 3-way handshake during connection establishment thus occurs before the actual need for message transmission rather than needing to occur at time or transmission and thus delaying the transmission." *Id*., p. 20. "A connectionless protocol, such as UDP/IP, may also used in cases where the connection-oriented protocol is not possible or not desirable because of the push server resources used by every static instance of a connection-oriented protocol. Connectionless communication is also used when a message needs to get through to the server 401 quickly, such as when occurs when the client 405 notifies the server 401 of an terminal event, user interface event or application event but the connection-oriented protocol connection is not connected or is already busy transferring information" *Id*., p. 20. "A method as claimed in claim 9, wherein said trying of one or more other techniques and/network parameters include one or more of using an alternate push server network address which may be more available or more accessible at the present time and from the present location, using one or more alternative mobile terminal network connection settings, or variation of the parameters within the settings, using an alternate connection route through a second wireless or fixed terminal accessible to the wireless terminal over a wireless network and capable of communication with the push server, using a TCP connection, using a UDP connection, using an SSL connection, using an HTTP or tunneling HTTP connection, using an SHTTP or tunneling HTTPS connection, using an XML-based connection over other transport protocol." *Id.*, claim 10.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 339 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 1  *Id.*, Fig. 1.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 340 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 341 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Ellison Ellison and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Ellison addresses the need for robust processor security in the context of modern computing, especially for applications such as e-commerce and business-to-business (B2B) transactions, which are increasingly conducted over networked systems. Ellison, Background. Blackberry Push enables secure, efficient communication and content delivery to mobile and wireless devices. Since Ellison's architecture is intended to be broadly compatible and to protect platforms engaged in e-commerce and B2B transactions, a POSITA would find it obvious to secure communications over all relevant channels—including wireless networks. Blackberry Push was a widely adopted system for secure communication with mobile devices. No inventive skill would be required for a POSITA integrating Blackberry Push with Ellison's isolated execution architecture in order to extend the security benefits of Ellison's invention to mobile and wireless environments. Ellison discloses processors are vulnerable to malicious software attacks. Ellison, Abstract, 1:16-51. To inhibit this, Ellison isolates different software elements using a hierarchy of abstract rings and "nubs" to create an "isolated execution mode" (secure-execution environment, red) in which hardware/software access "is restricted" compared to a "normal execution mode" (outside of the secure-execution environment, green) that "operates in a non-secure environment." *Id.*, 4:65-5:1, 6:1-26, 8:25-32, Figs. 1A-1C. Ellison controls access using computer-generated keys. *Id.*, 8:66-9:40, 9:47-62, Fig. 2. A POSITA would have been motivated to utilize Ellison's hierarchical, nub-based normal and isolated execution environments to provide enhanced security for Blackberry Push clients. This would have involved using a known technique (isolation environments) to improve similar devices (BES networks) in the same way (securing content). Ellison, as evidenced by the example citations below, discloses that a device link agent executes in a secure

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 342 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	execution environment on at least one of the devices, and at least one of the software components executes outside of the secure execution environment on that device. *See, e.g.*, "An operating system and the processor may have several levels of hierarchy, referred to as rings, corresponding to various operational modes. A ring is a logical division of hardware and software components that are designed to perform dedicated tasks within the operating system. The division is typically based on the degree or level of privilege, namely, the ability to make changes to the platform. For example, a ring-0 is the innermost ring, being at the highest level of the hierarchy. Ring-0 encompasses the most critical, privileged components. In addition, modules in Ring-0 can also access to lesser privileged data, but not vice versa. Ring-3 is the outermost ring, being at the lowest level of the hierarchy. Ring-3 typically encompasses users or applications level and has the least privilege. Ring-1 and ring-2 represent the intermediate rings with decreasing levels of privilege. FIG. 1A is a diagram illustrating a logical operating architecture 50 according to one embodiment of the invention. The logical operating architecture 50 is an abstraction of the components of an operating system and the processor. The logical operating architecture 50 includes ring-0 10, ring-1 20, ring-2 30, ring-3 40, and a processor nub loader 52. The processor nub loader 52 is an instance of an processor executive (PE) handler. The PE handler is used to handle and/or manage a processor executive (PE) as will be discussed later. The logical operating architecture 50 has two modes of operation: normal execution mode and isolated execution mode. Each ring in the logical operating architecture 50 can operate in both modes. The processor nub loader 52 operates only in the isolated execution mode. Ring-0 10 includes two portions: a normal execution Ring-011 and an isolated execution Ring-015. The normal execution Ring-0 11 includes software modules that are critical for the operating system, usually referred to as kernel. These software modules include primary operating system (e.g., kernel) 12, software drivers 13, and hardware drivers 14. The isolated execution Ring-0 15 includes an operating system (OS) nub 16 and a processor nub 18. The OS nub 16 and the processor nub 18 are instances of an OS executive (OSE) and processor executive (PE), respectively. The OSE and the PE are part of executive entities that operate in a secure environment associated with the isolated area 70 and the isolated execution mode. The

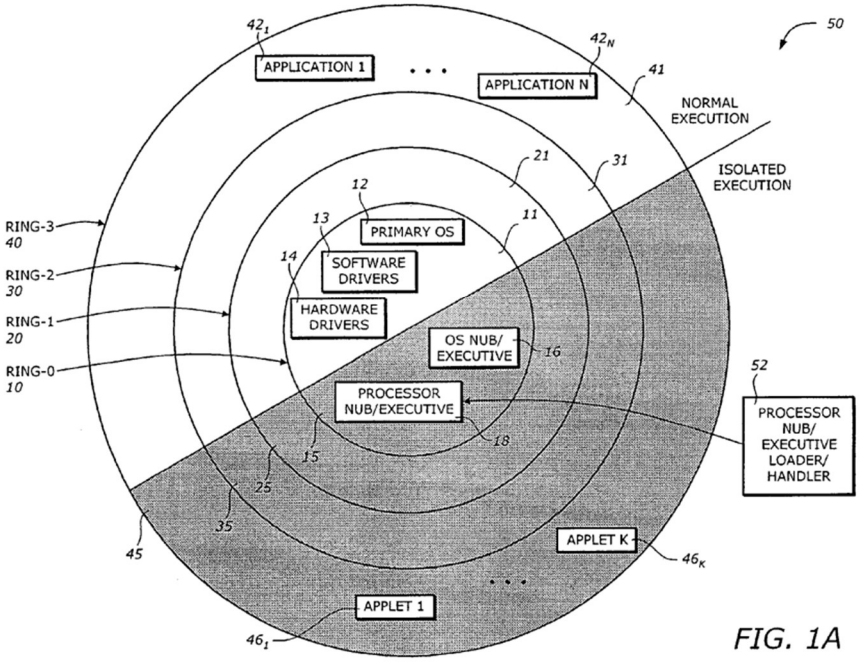
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 343 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	processor nub loader 52 is a protected bootstrap loader code held within a chipset in the system and is responsible for loading the processor nub 18 from the processor or chipset into an isolated area as will be explained later. Similarly, ring-1 20, ring-2 30, and ring-3 40 include normal execution ring-1 21, ring-2 31, ring-3 41, and isolated execution ring-1 25, ring-2 35, and ring-3 45, respectively. In particular, normal execution ring-3 includes N applications 42 to 42 and isolated execution ring-3 includes K applets 46 to 46. One concept of the isolated execution architecture is the creation of an isolated region in the system memory, referred to as an isolated area, which is protected by both the processor and chipset in the computer system. The isolated region may also be in cache memory, protected by a translation look aside (TLB) access check. Access to this isolated region is permitted only from a front side bus (FSB) of the processor, using special bus (e.g., memory read and write) cycles, referred to as isolated read and write cycles. The special bus cycles are also used for Snooping. The isolated read and write cycles are issued by the processor executing in an isolated execution mode. The isolated execution mode is initialized using a privileged instruction in the processor, combined with the processor nub loader 52. The processor nub loader 52 verifies and loads a ring-0 nub software module (e.g., processor nub 18) into the isolated area. The processor nub 18 provides hardware-related services for the isolated execution." Ellison, 2:46-3:48. "The processor 110 includes a normal execution mode 112 and an isolated execution circuit 115. The normal execution mode 112 is the mode in which the processor 110 operates in a non-secure environment, or a normal environment without the security isolated execution circuit 115 provides a mechanism to allow the processor 110 to operate in an isolated execution mode. The isolated execution circuit 115 provides hardware and software support for the isolated execution mode. This Support includes configuration for isolated execution, definition of an isolated area, definition (e.g., decoding and execution) of isolated instructions, generation of isolated access bus cycles, and generation of isolated mode interrupts. In one embodiment, the computer system 100 can be a single processor System, Such as a desktop computer, which has only one main central processing unit, e.g. processor 110. In other embodiments, the computer system 100 can include multiple processors, e.g. processors 110, 110a, 110b, etc., as shown in FIG. 1C."

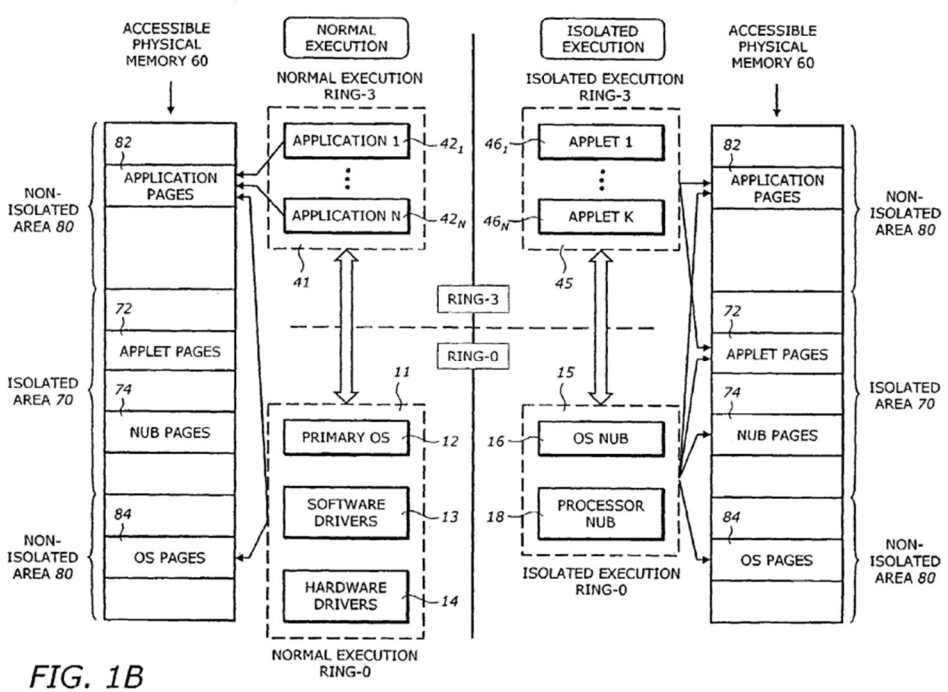
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 344 of 422

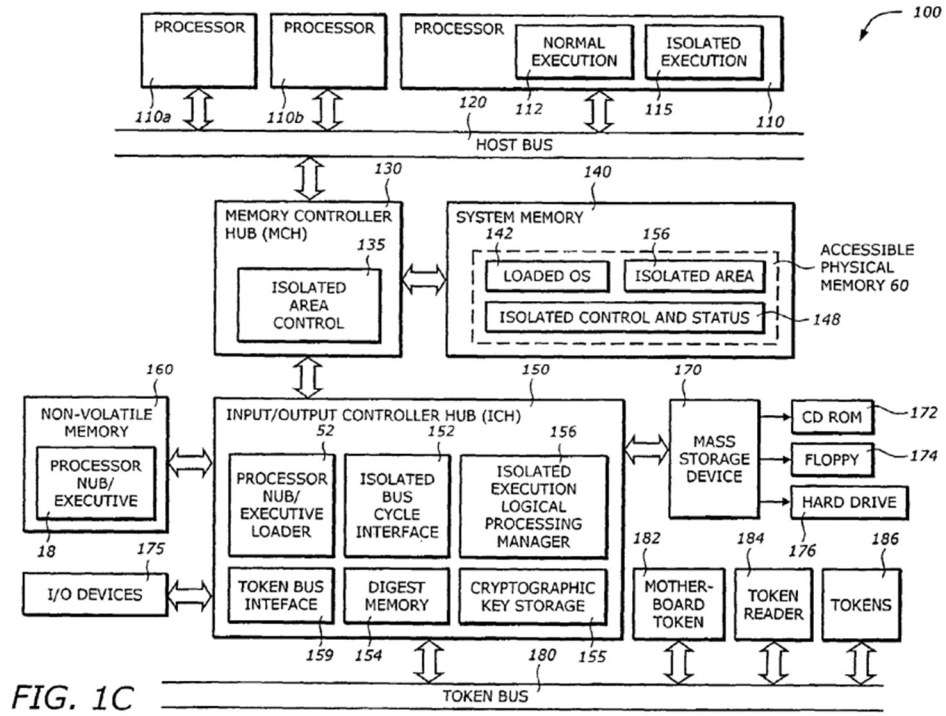
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Id., 4:63-5:16. "FIG. 2 is a diagram illustrating a secure platform 200 according to one embodiment of the invention. The secure platform 200 includes the OS nub 16, the processor nub 18, a key generator 240, a hashing function 220, and a usage protector 250, all operating within the isolated execution environment, as well as a software environment 210 that may exist either inside or outside the isolated execution environment." *Id*., 8:25-32. "The processor nub 18 includes a master binding key (BK0) 202. The BK0 202 is generated at random when the processor nub 18 is first invoked, i.e., when it is first executed on the secure platform 200. The key generator 240 generates a key operating system nub key (OSNK) 203 which is provided only to the OS Nub 16. The OS nub 16 may supply the OSNK 203 to trusted agents, such as the usage protector 250."*Id*., 8:66-9:6. "The Software environment 210 may include a plurality of subsets (e.g., subset 230). The usage of the software environment 210 or the usage of the subset 230 is protected by the usage protector 250. The usage protector 250 uses the OSNK 203 to protect the usage of the subset 230. The software environment 210 may include an operating system (e.g., a Windows operating system, a Windows 95 operating system, a Windows 98 operating system, a Windows NT operating system, Windows 2000 operating system) or a database. The subset 230 may be a registry in the Windows operating system or a Subset of a database. Elements can be implemented in hardware or software. The subset 230 is hashed by the hashing function 220 to produce a first hash value 206 and a second hash value 312. One way to detect intrusion or modification of the subset 230 is to compare the state of the subset before and after a time period. The first and second hash values 206 and 312 are typically generated at different times and/or at different places. The usage protector 250 is coupled to the key generator 240 to protect usage of the software environment 210 or the subset 230, using the OSNK 203. The usage protection includes protection against unauthorized reads, and detection of intrusion, tampering or unauthorized modification. If the two hash values are not the same, then the usage protector 250 knows that there is a change in the subset 230. If it is known that this

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 345 of 422

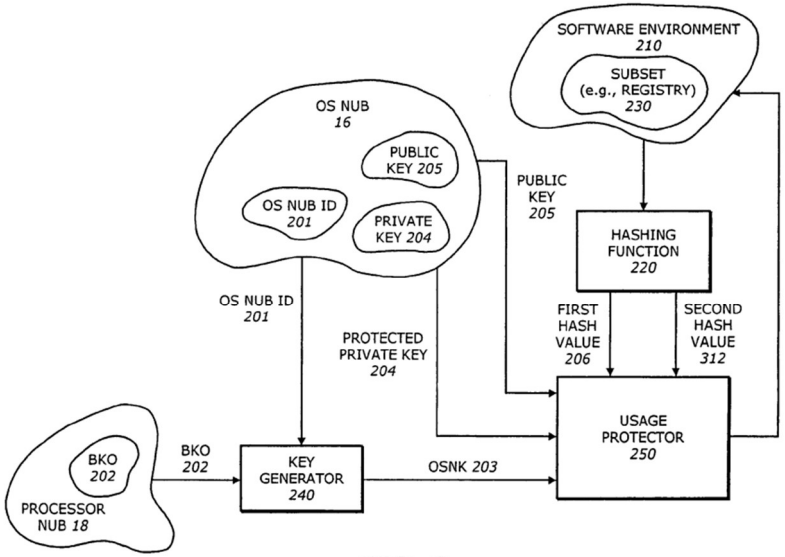
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	change is authorized and an updated hash value has been provided, the usage protector 250 would merely report the result. Otherwise, the usage protector 250 may generate an error or a fault function. When a user is notified of the error, fault condition, he or she would know that the subset 230 has been tampered, modified. The user may take appropriate action. Therefore, the usage of the subset 230 is protected." *Id.*, 9:28-62.  FIG. 1A *Id.*, Fig. 1A.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 346 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 1B *Id.*, Fig. 1B.

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 1C *Id.*, Fig. 1C.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 348 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 FIG. 2 *Id.*, Fig. 2. **Combination with TS.23-140** TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 349 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS.23-140, as evidenced by the citations below, discloses wherein the device link agent executes in a secure execution environment on at least one of the devices, and at least one of the software components executes outside of the secure execution environment on that device. *See, e.g.*, "The MMS Relay/Server is responsible for storage and handling of incoming and outgoing messages and for the transfer of messages between different messaging systems. Depending on the business model, the MMS Relay/Server may be a single logical element or may be separated into MMS Relay and MMS Server

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 350 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	elements. These may be distributed across different domains." TS-23.140, p. 17. "The MMS User Agent resides on a UE, an MS or on an external device connected to a UE/MS. It is an application layer function that provides the users with the ability to view, compose and handle MMs (e.g. submitting, receiving, deleting of MMs)." *Id.*, p. 18. "The MMS VAS Applications offer Value Added Services to MMS users. There could be several MMS VAS Applications included in or connected to an MMSE. MMS VAS Applications may be able to generate CDRs." *Id.* "The MMS VAS Applications provide value added services to the MMS users. In many ways MMS VAS Applications behave like a fixed MMS User Agent. However, MMS VAS Applications may provide some additional features like MM recall between MMS VAS Applications and MMS Relay/Server which are not available for MMS User Agents." *Id.*, p. 23. "The interface between MMS Relay/server and VASP may be carried over an encrypted and secure bearer, e.g. HTTP over SSL or TLS, or by use of application-layer encryption. This is an optional feature and may be further elaborated in future releases." *Id.*, p. 41.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 351 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push

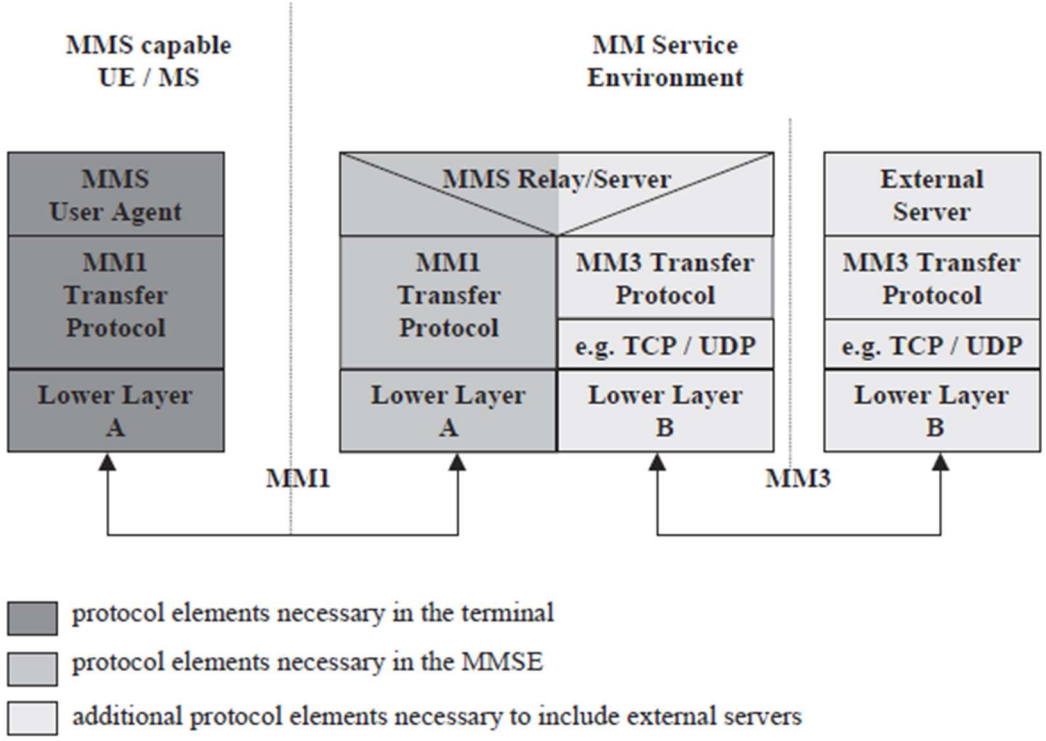


Figure 4: Protocol Framework to provide MMS

Id., Fig. 4.

Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push and/or TS-23.140, it would have been obvious in view of the disclosures in TS-23.140, or in combination with the knowledge of a person of ordinary skill in the art. Given that TS-23.140 discloses secure protocols, such as SSL and TLS, that create a secure execution environment, a POSITA would have

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 352 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	found it obvious that some applications (e.g., software components) would execute outside of the secure execution environment when these protocols are not used.
15: The networked system of claim 1, wherein the device link agent on a given one of the devices is further configured to receive, from one or more of the software components on that device, upload messages, each of the upload messages identifying a corresponding one of the network elements to which the device respective software component has requested delivery, the device link agent on that device transmitting the upload messages to the network message server over the respective secure Internet data message link, for delivery by the network message server to the respective identified network elements.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein the device link agent on a given one of the devices is further configured to receive, from one or more of the software components on that device, upload messages, each of the upload messages identifying a corresponding one of the network elements to which the device respective software component has requested delivery, the device link agent on that device transmitting the upload messages to the network message server over the respective secure Internet data message link, for delivery by the network message server to the respective identified network elements. Blackberry Push supports device-originated data uploads from applications over the secure link to MDS, which sends them to identified network elements (e.g., intranet web servers, application servers) using HTTP/TCP. *See, e.g.*, • MDS Studio 4.1, p. 12: Message Message components are designed to communicate application data across the wireless network between the client application on the BlackBerry device and the data source. Using the BlackBerry MDS Studio, bindings are created that link specific message components to specific data source operations. An instance of a message component can be either an •inbound message: transmits data from the data source to the application on the BlackBerry device •outbound message: transmits data from the application on the BlackBerry device to the data source • MDS Studio 4.1, p. 13: Messaging model

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 353 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	BlackBerry MDS Applications are designed to store and process data locally on the BlackBerry device, without the need for a continuously available network connection. The communication model used to send and receive application messages is fully asynchronous. This means that if an outbound message is generated at a time when the BlackBerry device is outside of a wireless coverage area (for example, as a result of user input), the message is placed into an outbound queue until the wireless connection becomes available. When the message is placed into this queue, the application continues functioning, regardless of whether the message has been sent. Messages are transmitted from the queue in the background when the network connection becomes available. To the extent Plaintiff contends that Blackberry Push fails to disclose this limitation, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Houghton, TS.23-140. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 354 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 1, wherein the device link agent on a given one of the devices is configured to receive, from one or more of the software components on the device, upload messages. *See, e.g.*, "A general problem with these synchronization systems is that the only time that the user data items are replicated between the host system and the mobile data communication device is when the user commands the mobile device to download or pull the user data from the host system." Lazaridis, 1:61-66. "In the case where message C is representative of a command message from the user's mobile device 24 to the host system 10, the command message C is not redirected, but is acted upon by the host system." *Id.*, 8:62-65. "If the redirected user data item is an E-mail message, as described above, the user at the mobile device 24 sees the original subject, sender's address, destination address, carbon copy and blind carbon copy. When the user replies to this message, or when the user authors a new message, the software operating at the mobile device 24 adds a similar outer envelope to the reply message (or the new message) to cause the message to be routed first to the user's host system 10, which then removes the outer envelope and redirects

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 355 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the message to the final destination, such as back to computer 26. In the preferred embodiment, this results in the outgoing redirected message from the user's host system 10 being sent using the E-mail address of the host mailbox, rather than the address of the mobile device, so that it appears to the recipient of the message that the message originated from the user's desktop system 10 rather than the mobile data communication device. Any replies to the redirected message will then be sent to the desktop system 10, which if it is still in redirector mode, will repackage the reply and resend it to the user's mobile data device, as described above." *Id.*, 8:66-9:20. "If the message C is a command message from the user's mobile device 24, then the server 11 simply acts upon the command message." *Id.*, 10:7-9. "The E-Mail sub-system also receives command messages C from the mobile device 24 that are directed to the desktop system 10 to trigger redirection or to carry out some other function." *Id.*, 11:11-14. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. "At step 82, the mobile device waits for messages and signals 84 to be generated or received. Assuming that the redirector software 12 operating at the user's desktop system 10 is configured to redirect upon receiving a message from the user's mobile device 24, at step 86, the user can decide to generate a command message that will start redirection. If the user does so, then at step 88 the redirection message is composed and sent to the desktop system 10 via the wireless network 22, through the wireless gateway 20, via the Internet 18 to the LAN 14, and is finally routed to the desktop machine 10. In this situation where the mobile device 24

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 356 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	is sending a message directly to the desktop system 10, no outer wrapper is added to the message (such as message C in FIGS. 1 and 2)." *Id.*, 13:6-19. "In addition to the redirection signal, the mobile device 24 could transmit any number of other commands to control the operation of the host system, and in particular the redirector program 12. For example, the mobile 24 could transmit a command to put the host system into the preferred list mode, and then could transmit additional commands to add or subtract certain senders from the preferred list. In this manner, the mobile device 24 can dynamically limit the amount of information being redirected to it by minimizing the number of senders on the preferred list. Other example commands include: (1) a message to change the configuration of the host system to enable the mobile device 24 to receive and process certain attachments; and (2) a message to instruct the host system to redirect an entire data item to the mobile device in the situation where only a portion of a particular data item has been redirected." Lazaridis, 13:20-35. "If, however, the redirection flag is on, then at step 110 the reply message is repackaged with the outer envelope having the addressing information of the user's desktop system 10, and the repackaged message is then transmitted to the desktop system 10 at step 106. As described above, the redirector program 12 executing at the desktop system then strips the outer envelope and routes the reply message to the appropriate destination address using the address of the desktop system as the "from" field, so that to the recipient of the redirected message, it appears as though it originated from the user's desktop system rather than the mobile data communication device." *Id.*, 13:59-14:3. Lazaridis, as evidenced by the example citations below, discloses that each of the upload messages identifies a corresponding one of the network elements (e.g., other systems connected to the LAN) to which the device respective software component has requested delivery. *See, e.g.*, "In an alternative system and method, the redirector program executes on a network server, and the server is programmed to detect numerous redirection event triggers over the network from multiple user desktop computers coupled to the server via a LAN. The server can receive internal event triggers from each of the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 357 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	user desktops via the network, and can also receive external event triggers, such as messages from the users' mobile data communication devices. In response to receiving one of these triggers, the server redirects the user's data items to the proper mobile data communication device. The user data items and addressing information for a particular mobile device can be stored at the server or at the user's PC. Using this alternative configuration, one redirector program can serve a plurality of users. This alternative configuration could also include an internet- or intranet-based redirector program that could be accessible through a secure webpage or other user interface. The redirector program could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "If the redirected user data item is an E-mail message, as described above, the user at the mobile device 24 sees the original subject, sender's address, destination address, carbon copy and blind carbon copy. When the user replies to this message, or when the user authors a new message, the software operating at the mobile device 24 adds a similar outer envelope to the reply message (or the new message) to cause the message to be routed first to the user's host system 10, which then removes the outer envelope and redirects

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 358 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	the message to the final destination, such as back to computer 26. In the preferred embodiment, this results in the outgoing redirected message from the user's host system 10 being sent using the E-mail address of the host mailbox, rather than the address of the mobile device, so that it appears to the recipient of the message that the message originated from the user's desktop system 10 rather than the mobile data communication device. Any replies to the redirected message will then be sent to the desktop system 10, which if it is still in redirector mode, will repackage the reply and resend it to the user's mobile data device, as described above." *Id.*, 8:66-9:20. "The E-Mail sub-system 44 can receive messages A from external computers on the LAN 14, or can receive messages C from some external network such as the Internet 18 or a wireless data communication network 22, and stores these messages in the primary data store 40. Assuming that the redirector 12 has been triggered to redirect messages of this type, the redirector detects the presence of any new messages and instructs the E-Mail system 44 to repackage the message by placing an outer wrapper B about the original message A (or C), and by providing the addressing information of the mobile data communication device 24 on the outer wrapper B. As noted above, this outer wrapper B is removed by the mobile device 24, and the original message A (or C) is then recovered, thus making the mobile device 24 appear to be the desktop system 10." *Id.*, 10:58-11:5. "In addition, the E-Mail sub-system 44 receives messages back from the mobile device 24 having an outer wrapper with the addressing information of the desktop system 10, and strips this information away so that the message can be routed to the proper sender of the original message A (or C). The E-Mail sub-system also receives command messages C from the mobile device 24 that are directed to the desktop system 10 to trigger redirection or to carry out some other function. The functionality of the E-Mail sub-system 44 is controlled by the redirector program 12." *Id.*, 11:6-15. "If, however, the redirection flag is on, then at step 110 the reply message is repackaged with the outer envelope having the addressing information of the user's desktop system 10, and the repackaged message is then transmitted to the desktop system 10 at step 106." *Id.*, 13:62-67.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 359 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Lazaridis, as evidenced by the example citations below, discloses the device link agent on that device transmitting the upload messages to the network message server over the respective secure Internet data message link, for delivery by the network message server to the respective identified network elements. *See, e.g.*, "Present systems and methods for replicating information from a host system to a user's mobile data communication device are typically 'synchronization' systems in which the user's data items are warehoused (or stored) at the host system for an indefinite period of time and then transmitted in bulk only in response to a user request. In these types of systems and methods, when replication of the warehoused data items to the mobile device is desired, the user typically places the mobile device in an interface cradle that is electrically connected to the host system via some form of local, dedicated communication, such as a serial cable or an infrared or other type of wireless link." *Id.*, 1:39-50. "In an alternative system and method, the redirector program executes on a network server, and the server is programmed to detect numerous redirection event triggers over the network from multiple user desktop computers coupled to the server via a LAN. The server can receive internal event triggers from each of the user desktops via the network, and can also receive external event triggers, such as messages from the users' mobile data communication devices. In response to receiving one of these triggers, the server redirects the user's data items to the proper mobile data communication device. The user data items and addressing information for a particular mobile device can be stored at the server or at the user's PC. Using this alternative configuration, one redirector program can serve a plurality of users. This alternative configuration could also include an internet- or intranet-based redirector program that could be accessible through a secure webpage or other user interface. The redirector program could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. "FIG. 5 is a flow chart showing the steps carried out by the mobile data communication device to interface with the redirector software operating at the host system." *Id.*, 5:31-33.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 360 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"Message C in FIG. 1 represents an external message from a sender that is not directly connected to LAN 14, such as the user's mobile data communication device 24, some other user's mobile device (not shown), or any user connected to the Internet 18. Message C also represents a command message from the user's mobile data communication device 24 to the host system 10. As described in more detail in FIG. 3, the host system 10 preferably includes, along with the typical hardware and software associated with a workstation or desktop computer, the redirector program 12, a TCP/IP subsystem 43, a primary message store 40, an E-mail subsystem 44, a screen saver subsystem 48, and a keyboard subsystem" *Id.*, 5:44-56. "In FIG. 1, the host system is the user's desktop system, typically located in the user's office. The host system is connected to a LAN 14, which also connects to other computers 26, 28 that may be in the user's office or elsewhere. The LAN 14, in turn, is connected to a wide area network ("WAN") 18, preferably the Internet, which is defined by the use of the Transmission Control Protocol/ Internet Protocol ("TCP/IP") to exchange information, but which, alternatively could be any other type of WAN. The connection of the LAN 14 to the WAN 18 is via high bandwidth link 16, typically a Tl or T3 connection. The WAN 18 in turn is connected to a variety of gateways 20, via connections 32. A gateway forms a connection or bridge between the WAN 18 and some other type of network, such as an RF wireless network, cellular network, satellite network, or other synchronous or asynchronous land-line connection." *Id.*, 5:57-6:6. "Among the user-defined event triggers that can be detected by the redirector program 12 are in the preferred embodiment external events, internal events and networked events. External events preferably include: (1) receiving a command message (such as message C) from the user's mobile data communication device to begin redirection, or to execute some other command at the host, such as a command to enable the preferred list mode, or to add or subtract a particular sender from the preferred list; (2) receiving a similar message from some external computer; and (3) sensing that the user is no longer in the vicinity of the host system; although, alternatively, an external event can be any other detectable occurrence that is external to the host system. Internal events could be a calendar alarm, screen saver activation, keyboard timeout, programmable timer, or any other user-defined event that is internal to the host system. Networked events are user-defined messages that are transmitted to the host system from another computer coupled to the host system via a

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 361 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	network to initiate redirection. These are just some of the events that could be used with the present invention to initiate replication of the user-selected data items from the host system 10 to the mobile device 24." *Id.*, 7:8-30. "Turning now to FIG. 3, a block diagram showing the interaction of the redirector software 12 with additional components of the host system 10 of FIG. l (the desktop PC) to enable more fully the pushing of information from the host system 10 to the user's mobile data communication device 24 is set forth. These additional components are illustrative of the type of event-generating systems that can be configured and used with the redirector software 12, and of the type of repackaging systems that can be used to interface with the mobile communication device 24 to make it appear transparent to the user." *Id.*, 10:10-20. "The desktop system 10 is connected to LAN 14, and can send and receive data, messages, signals, event triggers, etc., to and from other systems connected to the LAN 14 and to external networks 18, 22, such as the Internet or a wireless data network, which are also coupled to the LAN 14. In addition to the standard hardware, operating system, and application programs associated with a typical microcomputer or workstation, the desktop system 10 includes the redirector program 12, a TCP/IP sub-system 42, an E-mail sub-system 44, a primary data storage device 40, a screen saver sub-system 48, and a keyboard sub-system 46. The TCP/IP and E-mail subsystems 42, 44 are examples of repackaging systems that can be used to achieve the transparency of the present invention, and the screen saver and keyboard sub-systems 46, 48 are examples of event generating systems that can be configured to generate event messages or signals that trigger redirection of the user selected data items." *Id.*, 10:21-38. "FIGS. 4 and 5, set forth, respectively, flow charts showing the steps carried out by the 15 redirector software 12 operating at the host system 10, and the steps carried out by the mobile data communication device 24 in order to interface with the host system." *Id.*, 11:63-67. "FIG. 5 sets forth the method steps carried out by the user's mobile data communication device 24 in order to interface to the redirector program 12 of the present invention. At step 80 the mobile software is started

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 362 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	and the mobile device 24 is configured to operate with the system of the present invention, including, for example, storing the address of the user's desktop system 10." *Id.*, 12:65-13:5. **Combination with Houghton** Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id.*, p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id.*, p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id.*, p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 363 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill. A POSITA would have been motivated to incorporate computer memory and programming in Houghton to deliver messages from memory in the same way as disclosed by Blackberry Push. Houghton, as evidenced by the example citations below, discloses a transport services stack configured receive, over each of the respective secure message links, upload messages forwarded by the respective device link agents from at least a subset of the device software components, each of the upload messages identifying a corresponding one of the network elements to which the device respective software component has requested delivery, the network server system using the interface to a network to deliver content from the upload messages to the respective identified network elements. *See, e.g.*, "Figure 4 illustrates an exemplary implementation of the system for pushing application commands to a mobile terminal based on server side and client side events in accordance with one embodiment of the present invention. More specifically, Figure 4 illustrates a Push Client basic functionality 405 implemented in a mobile terminal 404 as part of a message passing system in cooperation with a Push Server 401. The push client 405 is preferably implemented in the form of a software run in a processor of the mobile terminal 404. The mobile terminal 404 is preferably a commercially available mobile terminal, such as a portable phone or appliance capable of running software or personal digital assistant, which provides a platform for downloading application software over air or from a personal computer, installing the software in the terminal, and running the software in the terminal. For example, the Symbian operating system is an open operating system, application framework, and application suite optimised for the needs of wireless information devices such as intelligent phones, smartphones and communicators, and open operating system and drives standards for the interoperation of data-enabled mobile phones with mobile networks, content applications, and services. The Symbian operating system also includes connectivity software for synchronisation with data on personal computers and servers. The application may be in an appropriate programming language, such as C++ or Java, which is an object-oriented programming language and virtual machine platform developed by Sun Microsystems. The push client software 405 preferably launches automatically when the mobile terminal 404 is turned on. The push client software 405 runs in the background and attempts to log into the push server 401 over a data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 364 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	network 403, normally a wireless network capable of transmitting Internet Protocol (IP) packets. The arrow A in Figure 4 illustrates an optional attempt by the push client 405 to communicate with the wireless network firewall 402 or other similar security or gateway device in order to open a connection for the push server 401 to directly contact the push client 401 as in FIG 3. If successful, operations can continue as described in FIG 3 or as below. Arrow B illustrates a login attempt. The wireless network 403 may be protected by means of a firewall or other appropriate security entity 402, which connects the wireless network to the public networks and filters and selectively discards the data packets entering and exiting the wireless network according to predefined rules. Thus, a firewall is a gateway device which operates at the same time as a connector and a separator between networks in a sense that it keeps track of the traffic that passes through it from one network to another and restricts connections and packets that are defined as unwanted by the administrator of the system. Additionally, IP connection techniques, such as Internet Protocol 4 (IP4) network address translation and Internet Protocol 6 (IP6) security, encryption and/or authentication, may hide the actual address of the mobile terminal or make it otherwise difficult for a server to connect to directly to the mobile terminal. The public telecommunications operator, company, organization, or private individual setting up this mobile Internet space with one or more users having greater ability to contact one another and included servers does this to protect the mobile terminals from hostile or unwanted contact common on the public Internet. They may also do so to reduce background traffic, protect their control over this space and limit outside or undesired service providers through technical connection policies set in the firewall, gateway or other security device. Physically a firewall may be a machine with appropriate software to do the task assigned to it. 402 can be a firewall, router, a personal computer, a gateway server or whatever similar computing device, infrastructure or secure protocol such as IP6 can be used for such purposes. Consequently to the existence and configuration of 402, if a fixed or mobile computing device from outside the controlled mobile network tries to contact a mobile terminal 404 inside the controlled mobile network, the incoming connection is not generally permitted unless the organization or individual controlling the firewall, gateway server or security device 24 makes an adjustment to their security policy and explicitly allows the connection. From point of view of the wireless packet data network 23, the login attempt B and associated connection establishment C is one or more normal IP connections established at the request of the mobile terminal. If optional attempt A to open a path for the push server 401 to initiate the IP connection now or in the future, use from that point forward may proceed as in FIG 3. However this firewall and/or security device or protocol opening A is unsuccessful, the login attempt B is in either case normally allowed to pass the firewall

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 365 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	or security infrastructure 402 since the connection is from the inside the firewall to outside. After attempting login, if the login is not successful the client 405 may automatically or under supervision of the user of the mobile terminal 404 try several different techniques for contacting the server 401. Since there are numerous variations between mobile terminals and in network connectivity, more than one connection attempt may be needed to contact the push server 401. Differences in wireless network terminal characteristics, individual push server availability and accessibility, firewall and gateway permissions, connection timeouts, connection level stateful (such as TCP) and stateless (such as UDP) transport protocol availability (such as IP4, IP6, SSL, HTTP, HTTPS) and alternate mobile terminal network settings mean that a sequence of connection establishment attempts and connection maintaining steps may required to be made using variations in relevant network parameters. A second reason for automatically adjusting the network parameters is not simply establishing the connection but establishing the connection in the most cost effective or highest performance configuration available for the individual mobile terminal user account service level. Connection types differ in speed, customer expensive, server resource usage and hence scalability, security and ability to penetrate gateway security devices. Some customers may be offered more secure or higher performance service while others receive the least cost connectivity from either the mobile terminal's or the push server's standpoint. As an example, such network parameters may include but are not limited to one or more of the following - Use of a connection-oriented protocol suitable to the wireless network such as TCP/IP protocol, which is generally preferred in the common situation where an intermediate firewall, Network Address Translation (NAT) or gateway device 402 - Use of a connectionless protocol suitable to the wireless network 403, such as UDP/IP, which is generally preferred when no network device restricting terminal contact 402 is present as this reduces resource usage on the server when large number of mobile terminals is served - Use of an alternate connection-oriented protocol such as HTTP suitable for mobile push operation through more strict firewalls and proxy Internet connections - Use of a secure protocol such as HTTPS, IP-Sec, secure IP6 or a proprietary security protocol to identify the communicating parties, prevent message interception by 3rd parties, and prevent message modification by 3rd parties

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 366 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	- Use of a periodic message from the client to the server or server to the client to the effect that client socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection - Use of configuration messages to the firewall (FIG 4, A) using a firewall or other security configuration protocol, such as Universal Plug and Play, to open a path for server-initiated connection-oriented or connectionless messaging to the client - Use of periodic messages initiated by the client 405 or by the server 401 to the effect that the receiving application is informed that their messages are continuing to be received, as indicated by response data from the receiving party to the sending party - Use of longer or shorter timing between such periodic messages to match the combined needs of all such network devices and protocols involved in transmitting communication, as adjusted based on the success or failure of earlier messages to successfully traverse the open network connection - Use of an alternate push server network address such as an address accessible with fewer security and performance restrictions which may be more available, more accessible, lower cost, lower latency, less busy, higher bandwidth or otherwise preferred at the present time and from the present network location of mobile terminal 404. - Use of one or more alternative mobile terminal network connection settings, or variation of the parameters within the settings. - Use of an alternate connection route through a second mobile or fixed terminal accessible to the mobile terminal over an alternate wireless network or network connection and capable of communication with a push server. In an embodiment of the invention, the mobile terminal 404 keeps a prioritized data structure listing the preferred communication protocols, server addresses, periodic message timings, network access settings, and alternate communication routes which it should use. Automated and manually assisted use of this structure should then result in a connection to one or more push servers, however if one of these connections which the mobile terminal is programmed to expect is not available, the mobile terminal can choose the

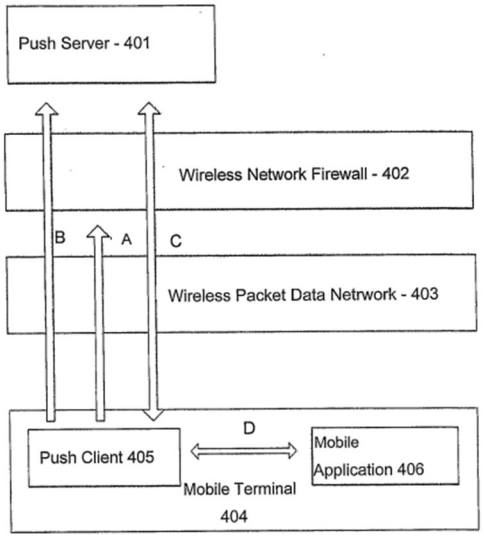
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 367 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	sleep that function of the client software and attempt re-connection at a later time, repeating in like fashion with the same or different network connection parameters as described above until a connection is established or other user intervention. At any time after a connection is established between the client 405 and the server 401 , the client may push a message to the server or the server may send a push message to the client. This is illustrated by means of the arrow C in Figure 4. These messages through a previously established, connection-oriented protocol such as TCP/IP, SSL, HTTP or HTTPS arrive quite rapidly, more rapidly than if the connection where established only when needed. The associated protocol overhead such as 3-way handshake during connection establishment thus occurs before the actual need for message transmission rather than needing to occur at time or transmission and thus delaying the transmission. A connectionless protocol, such as UDP/IP, may also used in cases where the connection-oriented protocol is not possible or not desirable because of the push server resources used by every static instance of a connection-oriented protocol. Connectionless communication is also used when a message needs to get through to the server 401 quickly, such as when occurs when the client 405 notifies the server 401 of an terminal event, user interface event or application event but the connection-oriented protocol connection is not connected or is already busy transferring information. Such a selective use of this high priority, low latency communication channel from terminal 404 to server 401 gives the server the opportunity to in return push to the client 405 a service triggered by the event begin communicated by the client 405 as will be described in more detail below. The sending of such a return message is frequently time critical and the fastest available combination of techniques will be used. In the preferred embodiment of the invention, the most basic message pushed from the server 401 to the client 405 is a data packet containing no information. Additional binary or text information specifying which mobile application 406 from a plurality of such applications may optionally be included. Additional binary or text information to be passed to the default or specified mobile application may optionally be included. Additional binary or text information specifying how upon receiving such a message the client 405 will notify and optionally ask the user if the action should be performed may optionally be included. Upon receiving such a message, appropriate automated actions and user interface media capabilities of the mobile terminal as specified in the command C from the server 401 are executed by the push client 405. The most common case is playing a sound, asking the user if the mobile application should be launched, and if affirmative passing the command details (arrow D) to the specified mobile application, module or additional

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 368 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	feature within the same application suite, launching the application, module or feature if needed. Such action may include optionally proceed without audio, with network downloaded audio, or without first requesting confirmation from the user as specified by user preferences and command details. The push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server. The trigger may, among others, include an alarm, notification, or measurement result received to the push server 401 from another device or system. With a push message, the server 401 may direct the mobile application 406 to make use of additional network resources either on the push server or other server. The push client 405 may in turn at any time accept data D from each of a plurality of mobile applications 406 and direct such to the push server 401 , possibly for redirection through push server API to other servers, mobile terminals, push clients and computing devices." Houghton, pp. 16-21.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 369 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 4  *Id.*, Fig. 4 "In the preferred embodiment of the invention, the most basic message pushed from the server 401 to the client 405 is a data packet containing no information. Additional binary or text information specifying which mobile application 406 from a plurality of such applications may optionally be included. Additional binary or text information to be passed to the default or specified mobile application may optionally be included. Additional binary or text information specifying how upon receiving such a message the client 405 will notify and optionally ask the user if the action should be performed may optionally be included. Upon receiving such a message, appropriate automated actions and user interface media capabilities of the mobile

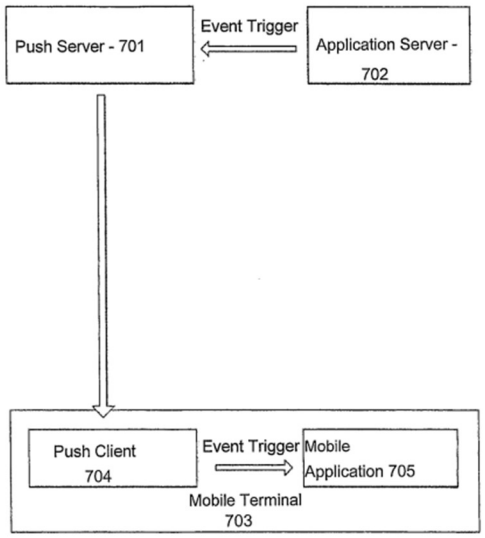
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 370 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	terminal as specified in the command C from the server 401 are executed by the push client 405. The most common case is playing a sound, asking the user if the mobile application should be launched, and if affirmative passing the command details (arrow D) to the specified mobile application, module or additional feature within the same application suite, launching the application, module or feature if needed. Such action may include optionally proceed without audio, with network downloaded audio, or without first requesting confirmation from the user as specified by user preferences and command details. The push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server. The trigger may, among others, include an alarm, notification, or measurement result received to the push server 401 from another device or system. With a push message, the server 401 may direct the mobile application 406 to make use of additional network resources either on the push server or other server. The push client 405 may in turn at any time accept data D from each of a plurality of mobile applications 406 and direct such to the push server 401 , possibly for redirection through push server API to other servers, mobile terminals, push clients and computing devices." *Id*., p. 21. "Figure 7 illustrates an example of an "COMMAND PUSH" procedure wherein the push server 701 is triggered by a trigger event as for example from an application server 702 to push an application command message to the push client 704 and thereby initiate a mobile terminal client trigger event in a mobile application 705 from a plurality of such applications, program modules and user interface features on the terminal 705, The trigger event from push client 704 to mobile application 705 may optionally include commands or data affecting the aspect of visual presentation of mobile application 705. The even trigger mechanism may be any electronic process mechanism as for example computer software calls to load or configure a class, library, dynamic link library, M I Diet, Symbian application, Linux daemon or other executable object. The trigger mechanism may be between separate parts of a larger application or application suite, or may be an IP-triggered application launch and data transfer as in the case of mobile application launched by contact to a specified IP port. The potential applications of the COMMAND PUSH include but are not limited to mobile applications for enterprise connectivity, back office integration, database access, scientific and industrial process and equipment monitoring and control, pictures transfer, mobile gaming, application data transfer, mobile electronic commerce, application alarms, reminders,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 371 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	multimedia interactivity, corporate calendaring, personal messaging and group messaging, etc. The push server 701 may in combination with user preferences stored in either the push server 701 or push client 704, optionally request the push client 704 to display or otherwise as appropriate for the mobile terminal or appliance notify the user asking permission for launching mobile application 705 prior to application launch. The preferred embodiment is for the behavior, language and content of this information to be configured on the server 701 based on the service and preferences of the individual receiving the message. In such cases an optional informational message will be presented to the user and such information messages are associated for instance with the menu option transmitted to the client 704 by the server 701. In an embodiment of the invention, the client 704 is also capable of receiving and storing variables which configure behavior of the client 704. The server 701 may thus specify the network connection preferences, notification sounds, and other behavior of the client 704. These changes facilitate the efficient and manageable operation of the entire system and can be used for automated remote maintenance of push client 704, mobile applications as 705, and configurable aspects of underlying mobile terminal 703, for example changing the username, password, and security certificates stored in the mobile terminal." *Id.*, pp. 21-23.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 372 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 7  *Id.*, Fig. 7. **Combination with TS.23-140** TS-23.140 and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network).

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 373 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	TS-23.140 is a technical specification issued by the 3GPP standards body to standardize multimedia transmissions through an MMS. TS-23.140, Figs. 3, pp. 10, 23. An "MMS Relay/Server" ("Relay/Server") coordinates storage, notification, reporting, and handling of multimedia messages ("MM[s]"). *Id*., pp. 21, 23-24. To transmit messages, Relay/Server may coordinate between MMS User Agents ("User Agents") on end-user devices, out-of-network MMS Relay/Servers, and Value Added Services ("VAS") Applications. *Id*. A POSITA would recognize that both TS.23-140 and Blackberry Push describe mobile data and network communications: TS.23-140 for MMS and Blackberry Push for a wider range of mobile internet and data communication. A POSITA would find it obvious that MMS relies on underlying transport protocols for message delivery, and that Blackberry Push utilized the same or similar methods for data exchange over mobile networks. The combination of MMS and Blackberry Push was well-understood and predictable, as TS.23-140 was designed for interoperability in mobile networks including Blackberry Push. Indeed, in TS-23.140, MM1 Transfer Protocol may implement the PAP defined by the Open Mobile Alliance ("OMA"), which was utilized by Blackberry Push. TS-23.140, pp. 13, 162. A POSITA would have recognized the benefits of combining these protocols to enable MMS services over mobile networks, and the standards themselves provided guidance for such integration. This combination of Blackberry Push and TS.23-140 would involve only routine engineering leading to predictable results because both were designed for interoperability in mobile networks. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with TS.23-140 MMS without requiring inventive skill. TS.23-140, as evidenced by the citations below, discloses the transport services stack (e.g., MM1 Transfer Protocol with TCP/IP and TLS) is further to receive, over each of the respective secure message links, upload messages (e.g., application-specific MM) forwarded by the respective device link agents (e.g., User Agent) from at least a subset of the device software components (e.g., messages submitted by/forwarded from a User Agent corresponding to at least one of the application registered therewith), each of the upload messages identifying a corresponding one of the network elements to which the device respective software

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 374 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	component has requested delivery (e.g., the messages include information identifying the recipient application and/or the recipient UE/mobile device), the network server system (e.g., Relay/Server) using the interface (e.g., MM1, MM2, MM4, and MM7 interfaces) to a network (e.g., 2G, 3G, and IP/internet networks) to deliver content from the upload messages to the respective identified network elements (e.g., destinations). *See, e.g.*, "MMS User Agent: application residing on a UE, an MS or an external device that performs MMS-specific operations on a user's behalf and/or on another application's behalf." *Id*., p. 14. "Figure 2 shows that multimedia messaging may encompass many different network types. The basis of connectivity between these different networks shall be provided by the Internet protocol and its associated set of messaging protocols. This approach enables messaging in 2G and 3G wireless networks to be compatible with messaging systems found on the Internet." *Id*., p. 17. "The MMS Relay/Server should provide additional functionalities such as: . . . transport of application data." *Id*., p. 21. "The MMS Relay/Server is responsible for storage and notification, reports, and general handling of messages." *Id*. "The MMS Relay/Server shall provide the following functionalities: receiving and sending MM." *Id*. "The interfaces in the MMS Reference Architecture are: MM1: The reference point between the MMS User Agent and the MMS Relay/Server. MM2: The reference point between the MMS Relay and the MMS Server. MM3: The reference point between the MMS Relay/Server and external (legacy) messaging systems. MM4: The reference point between the MMS Relay/Server and another MMS Relay/Server that is within another MMSE. MM5: The reference point between the MMS Relay/Server and the Home Location Register (HLR). MM6: The reference point between the MMS Relay/Server and the MMS User Databases. MM7: The reference point between the MMS Relay/Server and MMS VAS Applications. MM8: The reference point between the MMS Relay/Server and the post-processing system. MM9: The reference point between the MMS Relay/Server and the online charging system. MM10: The reference point between the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 375 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	MMS Relay/Server and a Messaging Service Control Function (MSCF)." *Id*., pp. 23-24. "Reference point MM1 is used to submit Multimedia Messages from MMS User Agent to MMS Relay/Server, to let the MMS User Agent pull MMs from the MMS Relay/Server, let the MMS Relay/Server push information about MMs to the MMS User Agent as part of an MM notification, and to exchange delivery reports between MMS Relay/Server and MMS User Agents." *Id*., p. 24. "If an MMS User Agent supports submission of MMs the MMS User Agent shall be able to: Indicate the address of the MM recipient" *Id*., p. 26. "Upon reception of a read-reply report from a recipient MMS User Agent the recipient MMS Relay/Server . . . shall route the read-reply report towards the originator MMS User Agent or originator VASP of the original MM." *Id*., p. 35. "Upon retrieve request, the recipient MMS Relay/Server shall deliver a modified MM with one or several presentation descriptions as defined in [41], as one or several MM elements, in place of the corresponding streamable MM contents to the recipient MMS User Agent, if it has made the MM contents available for streaming retrieval." *Id*. "The MMS Relay/Server shall store an MM into an MMBox under the following conditions: . . . • MMS User Agent submitting an MM, which also includes a store request; • MMS User Agent forwarding an MM, which also includes a store request; • MMS User Agent uploading an MM for storage into the MMBox" *Id*., p. 40. "Apart from using MMS as a service for users to exchange messages, MMS may also be used to transport data specific to applications. Applications that intend to transport application specific data using MMS may either reside on an MMS User Agent or on an MMS VAS Application." *Id*., p. 54. "Applications that want to transport data specific to applications other than MMS will initially need to

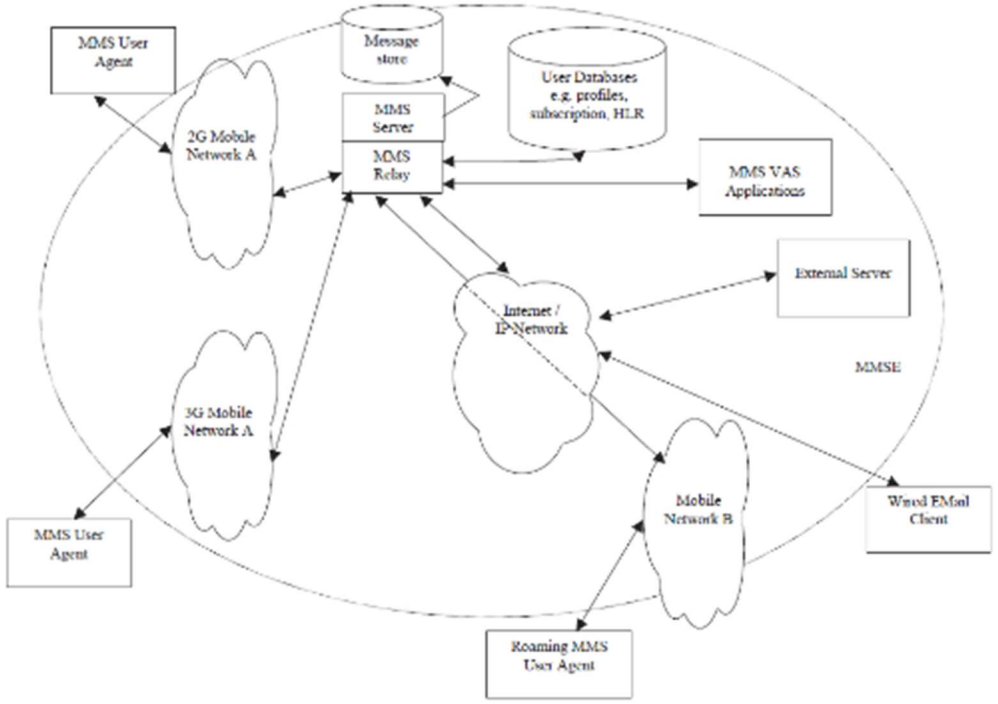
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 376 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	register with the appropriate MMS User Agent or MMS VAS Application. During this registration process the application provisions an MMS User Agent or an MMS VAS Application with its application identification value and may negotiate with the MMS User Agent or MMS VAS Application the details (amount and format) of information to be exchanged between the two entities. The application registration process is outside the scope of this specification. The registration may be an inherent process e.g., in the application's integration into a mobile phone. It may also be the initial step after the download of a downloadable application to a mobile phone. Whatever the details of the application registration process are, an MMS User Agent or an MMS VAS Application acts according to the negotiated results from the application registration process." *Id*. "MMS is used to transport data specific to applications between two MMS User Agents or an MMS User Agent and an MMS VAS Application (or vice versa)." *Id*., p. 55. "Based on the negotiated details upon application registration process an application may trigger an MMS User Agent or an MMS VAS Application to submit certain abstract messages. Upon triggering an MMS User Agent or an MMS VAS Application to send an abstract message the MMS User Agent or MMS VAS Application may receive information from the application. The MMS User Agent or MMS VAS Application may insert this information in both the information elements and/or payload (if present) of the abstract message. The details for the above are according to the results of the application registration process. Abstract messages that are sent by an MMS User Agent or an MMS VAS Application on behalf of an originating application shall contain a destination application identifier. They may, in addition, contain an application identifier which is to be used in reply-MMs and they may contain additional application/implementation specific control information." *Id*. "If the MMS Relay/Server finds from the recipient MMS User Agent's capability indication (see clause 7.1.3.1) that the recipient MMS User Agent supports transport of application data, the MMS Relay/Server . . . shall pass on the destination application identifier, the 'reply-path' identifier (if present) and the additional application/implementation specific control information (if present) unaltered." *Id*., pp. 55-56.

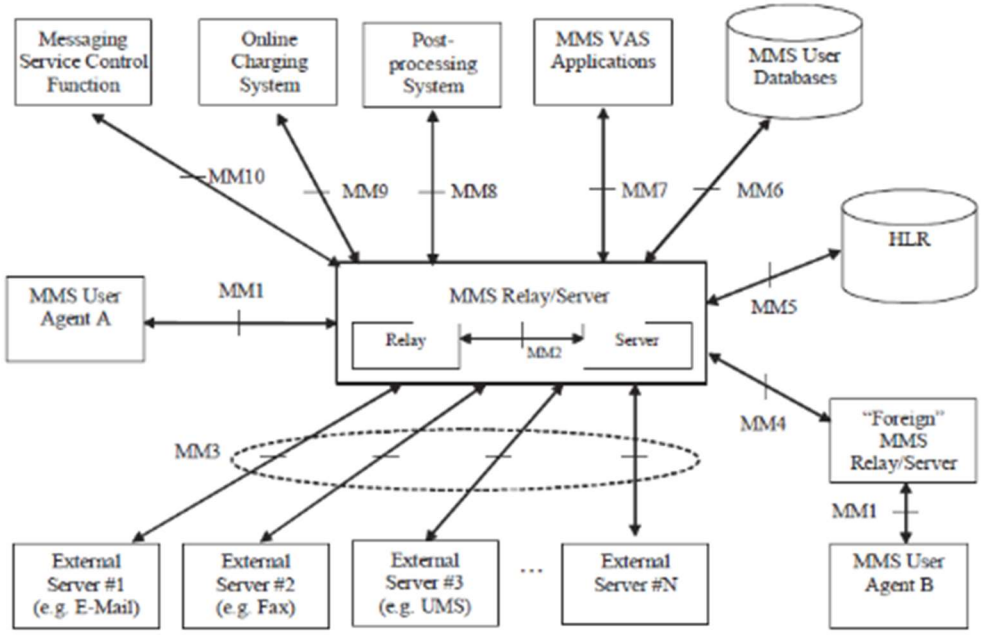
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 377 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The MMS addressing model on MM1 contains three addresses: the address of the MMS Relay/Server, the address of the recipient and the address of the originator. The address of the MMS Relay/Server shall be the URI of the MMS Relay/Server given by the MMS service provider. Thus, the URI needs to be configurable in the MMS User Agent." *Id*., p. 57. "The originator's a address could be either a user's address or a user's terminal address. The recipient's address can be a user's address, a user's terminal address, or a short code. For this release the user's terminal addresses (e.g. terminal IP addresses) are not supported. The MMS User Agent's responsibility is to format these addresses before it submits the message to the originator MMS Relay/Server." *Id*. "[T]he originator MMS User Agent may specify each of the address fields in one of the following formats: 1. RFC 2822 address (FQDN or unqualified) 2. PLMN address: ['+' | '*' | '#'] [digit | '*' | '#'] … ['/TYPE= PLMN'] 3. Other '/TYPE= ' The '/TYPE=' field specifies the address type. When PLMN format is used the type is optional. The '/TYPE='convention provides flexibility for future enhancements. When the '/TYPE=' qualifier is absent, the MMS Relay/Server should resolve potential ambiguities by applying the following logic to the address in the following order: 1. if it contains the '@' character, the address should be interpreted as an FQDN RFC2822 address 2. if it is completely numeric, except possibly including '+', '*', or '#', it should be interpreted as '/TYPE=PLMN', e.g. an E.164 address, a local telephone number, or a numeric short code, 3. otherwise, it should be interpreted as an unqualified RFC2822 address (alphanumeric short code)" *Id*., pp. 57-58. "[T]he originator MMS Relay/Server should use the recipient's address(es) as indicated in the corresponding MM to route the converted message towards its recipient(s). In addition to this, it may convey message class, priority and subject of the associated MM as part of the converted message." *Id*., p., 90.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 378 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"The recipient address over MM1 may or may not be mapped to recipient address over MM7. The recipient address over MM7 may also be independent of the recipient address over MM1." *Id.*, p. 190.  **Figure 2: MMS Architectural Elements** *Id.*, Fig. 2.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 379 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **Figure 3: MMS Reference Architecture** *Id*. Fig. 3.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 380 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push

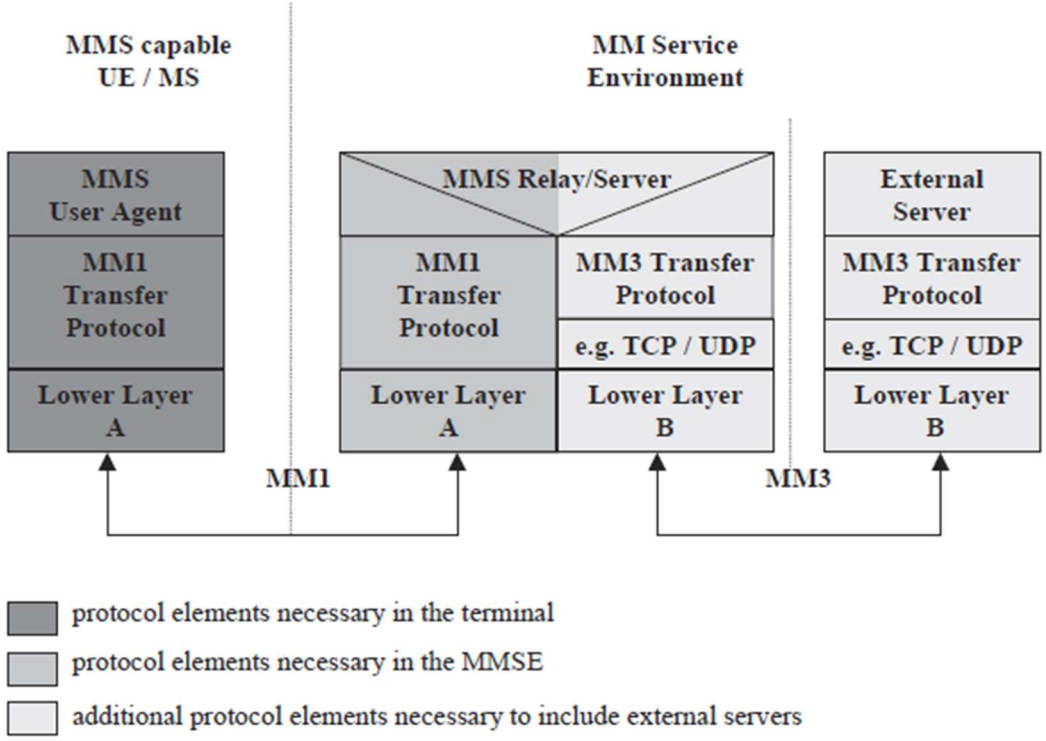


Figure 4: Protocol Framework to provide MMS

Id., Fig. 4.

Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push and/or TS-23.140, it would have been obvious in view of the disclosures in TS-23.140, or in combination with the knowledge of a person of ordinary skill in the art. A POSITA would have understood that an application receiving application data using MMS can also send application data to an application

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 381 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	resident on another device (e.g., another device with an MMS User Agent or an MMS VAS Application). Moreover, in the MMS Environment contemplated by TS-23.140 (*see* Fig. 2), a POSITA would have understood that multiple different devices, including end-user devices/terminal, would be able to use an MMS Relay/Server for transmission of data to multiple other devices/server/network elements. In such an environment, a POSITA would have found it obvious for the MMS Relay/Server to receive upload messages (including application data) from multiple MMS User Agents and their respective applications requesting delivery of such data, and such communications would happen over the dedicated TLS-based links between each such MMS User Agent and the MMS Relay/Server.
16: The networked system of claim 15, the device link agent on the given device further configured to buffer one or more of the upload messages for transmission to the network message server at a time selected by a heartbeat mechanism.	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 15, the device link agent on the given device further configured to buffer one or more of the upload messages for transmission to the network message server at a time selected by a heartbeat mechanism. Blackberry Push devices queue outbound messages when out of coverage and transmit when connectivity resumes—functioning equivalently to a heartbeat-triggered release. *See. e.g.*, • MDS Studio 4.1, p. 14: Messages that pass between BlackBerry MDS Applications and the BlackBerry MDS Services are categorized into two delivery modes: best-effort and standard. The default delivery mode is standard. Best-effort The best-effort delivery mode is a fire-and-forget mode. Messages might not be delivered, and they might be delivered in a different order than they were sent from the application to the BlackBerry MDS Services or from the BlackBerry MDS Services to the application. Standard Messages sent from the application are designed to reach the BlackBerry device, even if the BlackBerry device spends time outside of a wireless coverage area or if the BlackBerry MDS Services stop. The sending

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 382 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	side transmits the message when a connection becomes available. The application does not confirm receipt of messages, so the application might not process a message even after it reaches the BlackBerry device. Similarly, the BlackBerry MDS Services might not process a message from the application even when the message reaches the BlackBerry MDS Services server. Standard mode messages are designed to be delivered and processed in the order in which they were generated. Handling messages when a BlackBerry device is outside of a wireless coverage area When a BlackBerry device is outside of a wireless coverage area, outbound standard and reliable application messages remain queued on the BlackBerry device until the BlackBerry device returns to a wireless coverage area, at which time the messages are transmitted. Best-effort messages are dropped when a BlackBerry device is outside of a wireless coverage area. Message queue sizes BlackBerry MDS Applications have restricted inbound and outbound message queue sizes. These restrictions are imposed by policies sent to the BlackBerry MDS Runtime software on the BlackBerry devices. The system administrator of the BlackBerry MDS Services can push new policies to the BlackBerry MDS Runtime, which might change the limitations of an application's inbound and outbound queue sizes. To the extent this limitation is not disclosed by Blackberry Push, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis, Houghton. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 383 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill. Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 15, the device link agent on the given device further configured to buffer one or more of the upload messages for transmission to the network message server at a time selected by a heartbeat mechanism. *See, e.g.*, "A system and method for pushing information from a host system to a mobile data communication device upon sensing a triggering event is disclosed. A redirector program operating at the host system enables a user to continuously redirect certain user-selected data items from the host system to the user's mobile data communication device upon detecting that one or more user-defined triggering events has occurred. The redirector program operates in connection with event generating applications and repackaging systems at the host system to configure and detect a particular user-defined event, and then to repackage the user-selected data items in an electronic wrapper prior to pushing the data items to the mobile device. The mobile

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 384 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	device and the host system share a common electronic address so that messages generated at either the host system or the mobile data communication device are configured using the common electronic address." Lazaridis, Abstract. "Software executing on the mobile data communication device then transmits commands via the local communications link to the host system to cause the host to begin transmitting the user's data items for storage in a memory bank of the mobile device." *Id.*, 1:50-54. "Other sub-systems for creating triggering events and repackaging the user's data items could also be present at the host system. The host system also includes a primary memory store where the user's data items are normally stored." *Id.*, 3:4-8. "Once an event has triggered redirection of the user data items, the host system then repackages these items in a manner that is transparent to the mobile data communication device, so that information on the mobile device appears similar to information on the user's host system. The preferred repackaging method includes wrapping the user data items in an E-mail envelope that corresponds to the address of the mobile data communication device, although, alternatively, other repackaging methods could be used with the present invention, such as special-purpose TCP/IP wrapping techniques, or other methods of wrapping the user selected data items. The repackaging preferably results in E-mail messages generated by the user from the mobile device to be transmitted from the host system, thus enabling the user to appear to have a single E-mail address, such that the recipients of messages sent from the mobile communications device do not know where the user was physically located when the message was first sent. The repackaging also permits both messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed and decompressed." *Id.*, 3:66-4:19. "Message C in FIG. 1 represents an external message from a sender that is not directly connected to LAN 14, such as the user's mobile data communication device 24, some other user's mobile device (not shown), or any user connected to the Internet 18. Message C also represents a command message from the user's

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 385 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	mobile data communication device 24 to the host system 10. As described in more detail in FIG. 3, the host system 10 preferably includes, along with the typical hardware and software associated with a workstation or desktop computer, the redirector program 12, a TCP/IP subsystem 43, a primary message store 40, an E-mail subsystem 44, a screen saver subsystem 48, and a keyboard subsystem" *Id.*, 5:44-56. "A user of the present invention can configure the redirector 12 program to push certain user-selected data items to the user's mobile data communication device 24 when the redirector 12 detects that a particular user-defined event trigger (or trigger point) has taken place. User-selected data items preferably include E-mail messages, calendar events, meeting notifications, address entries, journal entries, personal alerts, alarms, warnings, stock quotes, news bulletins, etc., but could, alternatively, include any other type of message that is transmitted to the host system 10 or that the host system 10 acquires through the use of intelligent agents, such as data that is received after the host system 10 initiates a search of a database or a website or a bulletin board. In some instances, only a portion of the data item is transmitted to the mobile device 24 in order to minimize the amount of data transmitted via the wireless network 22. In these instances, the mobile device 24 can optionally send a command message to the host system to receive more or all of the data item if the user desires to receive it." *Id.*, 6:56-7:7. "FIG. 1 shows an E-mail message A being communicated over LAN 14 from computer 26 to the user's desktop system 10 (also shown in FIG. 1 is an external message C, which could be an Email message from an Internet user, or could be a command message from the user's mobile device 24). Once the message A (or C) reaches the primary message store of the host system 10, it can be detected and acted upon by the redirection software 12. The redirection software 12 can use many methods of detecting new messages." *Id.*, 7:31-38. After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 386 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "In the case where message C is representative of an external message from a computer on the Internet 18 to the host system 10 and the host has been configured to redirect messages of type C, then in a similar manner to message A, message C would be repackaged with an outer envelope B and transmitted to the user's mobile device 24." *Id.*, 8: 56-61. "FIG. 2 is an alternative system diagram showing the redirection of user data items from a network server 11 to the user's mobile data communication device 24, where the redirector software 12 is operating at the server 11. This configuration is particularly advantageous for use with message servers such as Microsoft's® Exchange Server, which is normally operated so that all user messages are kept in one central location or mailbox store on the server instead of in a store within each user's desktop PC. This configuration has the additional advantage of allowing a single system administrator to configure and keep track of all users having messages redirected. If the system includes encryption keys, these too can be kept at one place for management and update purposes." *Id.*, 9:20-33. "If the message C is from an Internet computer to the user's desktop system and the user has redirection capabilities, then the server 11 detects the message C, repackages it using electronic envelope B, and redirects the repackaged message (C in B) to the user's mobile device 24. If the message C is a command

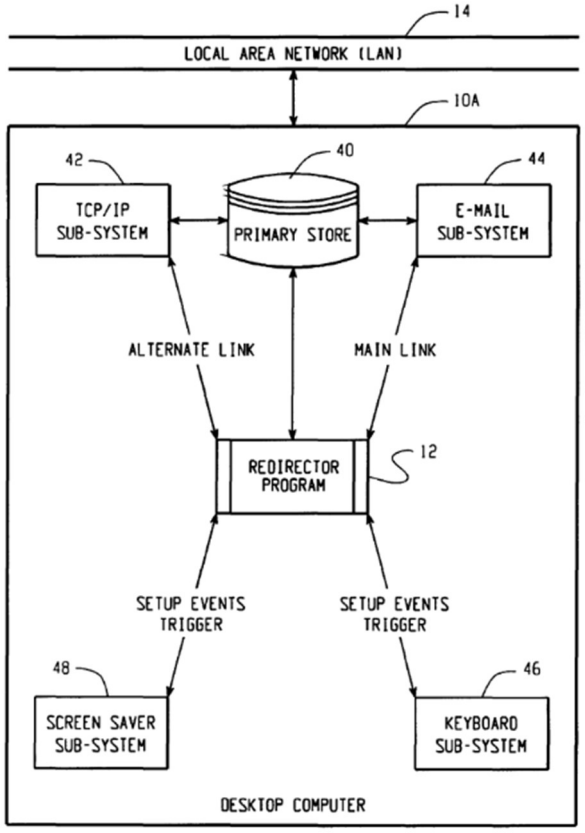
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 387 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	message from the user's mobile device 24, then the server 11 simply acts upon the command message." *Id.*, 10:2-8. "The desktop system 10 is connected to LAN 14, and can send and receive data, messages, signals, event triggers, etc., to and from other systems connected to the LAN 14 and to external networks 18, 22, such as the Internet or a wireless data network, which are also coupled to the LAN 14. In addition to the standard hardware, operating system, and application programs associated with a typical microcomputer or workstation, the desktop system 10 includes the redirector program 12, a TCP/IP sub-system 42, an E-mail sub-system 44, a primary data storage device 40, a screen saver sub-system 48, and a keyboard sub-system 46. The TCP/IP and E-mail subsystems 42, 44 are examples of repackaging systems that can be used to achieve the transparency of the present invention, and the screen saver and keyboard sub-systems 46, 48 are examples of event generating systems that can be configured to generate event messages or signals that trigger redirection of the user selected data items." *Id.*, 10:21-38. "The TCP/IP sub-system 42 is an alternative repackaging system. It includes all of the functionality of the E-Mail sub-system 44, but instead of repackaging the user-selected data items as standard E-mail messages, this system repackages the data items using special-purpose TCP/IP packaging techniques. This type of special-purpose sub-system is useful in situations where security and improved speed are important to the user. The provision of a special-purpose wrapper that can only be removed by special software on the mobile device 24 provides the added security, and the bypassing of E-mail store and forward systems can improve speed and real-time delivery." *Id.*, 11:16-27. "When a message or signal is detected, the program 12 determines 58 whether it is one of the trigger events that has been configured by the user to signal redirection. If so, then at step 60 a trigger flag is set, indicating that subsequently received user data items (in the form of messages) that have been selected for redirection should be pushed to the user's mobile data communication device 24." *Id.*, 12:33-39.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 388 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"If the trigger flag is set, then at step 70, the redirector 12 causes the repackaging system (E-Mail or TCP/IP) to add the outer envelope to the user data item, and at step 72 the repackaged data item is then redirected to the user's mobile data communication device 24 via LAN 14, WAN 18, wireless gateway 20 and wireless network 22." *Id.*, 12:50-56. "If, however, the redirection flag is on, then at step 110 the reply message is repackaged with the outer envelope having the addressing information of the user's desktop system 10, and the repackaged message is then transmitted to the desktop system 10 at step 106." *Id.*, 13:62-67. "If, however, the mobile determines that the user is using the redirector software 12 at the desktop system 10, then control passes to step 110, where the outer envelope is added to the message. The repackaged original message is then transmitted to the desktop system 10 at step 106, which, as described previously, strips the outer envelope and routes the message to the correct destination. Following transmission of the message at step 106, control of the mobile returns to step 82 and waits for additional messages or signals." *Id.*, 14:14-23.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 389 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 3 *Id.*, Fig. 3.

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Houghton Houghton and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Houghton and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Houghton attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Houghton, p. 10; MDS Studio 4.1, p. 15. Houghton discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Houghton, Abstract, p. 11. Further, Houghton discloses a network comparable to Blackberry Push's system for pushing messages to a client on a mobile terminal in a wireless network. Houghton, *Id*., p. 11. Push messages "may be triggered by any trigger event, local or remote, defined at the server," including an "alarm, notification, or measurement result received to the push server 401 from another device or system." *Id*., p. 21. Upon triggering, messages are transmitted over a network using secure protocols, e.g., "HTTPS, IP-Sec, secure IP6 or a proprietary security protocol." *Id*., p. 19. A POSITA would have found it obvious to combine Houghton's push-type command of applications on a mobile terminal by a remote server computer with Blackberry Push's system. This combination of Blackberry Push and Houghton would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Houghton, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Houghton's push messaging system without requiring inventive skill.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 391 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	A POSITA would have been motivated to incorporate computer memory and programming in Houghton to deliver messages from memory in the same way as disclosed by Blackberry Push. Houghton, as evidenced by the example citations below, discloses the networked system of claim 15, the device link agent on the given device further configured to buffer one or more of the upload messages for transmission to the network message server at a time selected by a heartbeat mechanism.. *See, e.g.*, "Figure 4 illustrates an exemplary implementation of the system for pushing application commands to a mobile terminal based on server side and client side events in accordance with one embodiment of the present invention. More specifically, Figure 4 illustrates a Push Client basic functionality 405 implemented in a mobile terminal 404 as part of a message passing system in cooperation with a Push Server 401. The push client 405 is preferably implemented in the form of a software run in a processor of the mobile terminal 404. The mobile terminal 404 is preferably a commercially available mobile terminal, such as a portable phone or appliance capable of running software or personal digital assistant, which provides a platform for downloading application software over air or from a personal computer, installing the software in the terminal, and running the software in the terminal. For example, the Symbian operating system is an open operating system, application framework, and application suite optimised for the needs of wireless information devices such as intelligent phones, smartphones and communicators, and open operating system and drives standards for the interoperation of data-enabled mobile phones with mobile networks, content applications, and services. The Symbian operating system also includes connectivity software for synchronisation with data on personal computers and servers. The application may be in an appropriate programming language, such as C++ or Java, which is an object-oriented programming language and virtual machine platform developed by Sun Microsystems. The push client software 405 preferably launches automatically when the mobile terminal 404 is turned on. The push client software 405 runs in the background and attempts to log into the push server 401 over a data network 403, normally a wireless network capable of transmitting Internet Protocol (IP) packets. The arrow A in Figure 4 illustrates an optional attempt by the push client 405 to communicate with the wireless network firewall 402 or other similar security or gateway device in order to open a connection for the push server 401 to directly contact the push client 401 as in FIG 3. If successful, operations can continue as described in FIG 3 or as below. Arrow B illustrates a login attempt. The wireless network 403 may be protected by means of a firewall or other appropriate security entity 402, which connects the wireless network to the public networks and filters and selectively discards the data

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 392 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	packets entering and exiting the wireless network according to predefined rules. Thus, a firewall is a gateway device which operates at the same time as a connector and a separator between networks in a sense that it keeps track of the traffic that passes through it from one network to another and restricts connections and packets that are defined as unwanted by the administrator of the system. Additionally, IP connection techniques, such as Internet Protocol 4 (IP4) network address translation and Internet Protocol 6 (IP6) security, encryption and/or authentication, may hide the actual address of the mobile terminal or make it otherwise difficult for a server to connect to directly to the mobile terminal. The public telecommunications operator, company, organization, or private individual setting up this mobile Internet space with one or more users having greater ability to contact one another and included servers does this to protect the mobile terminals from hostile or unwanted contact common on the public Internet. They may also do so to reduce background traffic, protect their control over this space and limit outside or undesired service providers through technical connection policies set in the firewall, gateway or other security device. Physically a firewall may be a machine with appropriate software to do the task assigned to it. 402 can be a firewall, router, a personal computer, a gateway server or whatever similar computing device, infrastructure or secure protocol such as IP6 can be used for such purposes. Consequently to the existence and configuration of 402, if a fixed or mobile computing device from outside the controlled mobile network tries to contact a mobile terminal 404 inside the controlled mobile network, the incoming connection is not generally permitted unless the organization or individual controlling the firewall, gateway server or security device 24 makes an adjustment to their security policy and explicitly allows the connection. From point of view of the wireless packet data network 23, the login attempt B and associated connection establishment C is one or more normal IP connections established at the request of the mobile terminal. If optional attempt A to open a path for the push server 401 to initiate the IP connection now or in the future, use from that point forward may proceed as in FIG 3. However this firewall and/or security device or protocol opening A is unsuccessful, the login attempt B is in either case normally allowed to pass the firewall or security infrastructure 402 since the connection is from the inside the firewall to outside. After attempting login, if the login is not successful the client 405 may automatically or under supervision of the user of the mobile terminal 404 try several different techniques for contacting the server 401. Since there are numerous variations between mobile terminals and in network connectivity, more than one connection attempt may be needed to contact the push server 401. Differences in wireless network terminal characteristics, individual push server availability and accessibility, firewall and gateway permissions, connection timeouts, connection level stateful (such as TCP) and stateless (such as UDP) transport protocol

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 393 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	availability (such as IP4, IP6, SSL, HTTP, HTTPS) and alternate mobile terminal network settings mean that a sequence of connection establishment attempts and connection maintaining steps may required to be made using variations in relevant network parameters. A second reason for automatically adjusting the network parameters is not simply establishing the connection but establishing the connection in the most cost effective or highest performance configuration available for the individual mobile terminal user account service level. Connection types differ in speed, customer expensive, server resource usage and hence scalability, security and ability to penetrate gateway security devices. Some customers may be offered more secure or higher performance service while others receive the least cost connectivity from either the mobile terminal's or the push server's standpoint. As an example, such network parameters may include but are not limited to one or more of the following - Use of a connection-oriented protocol suitable to the wireless network such as TCP/IP protocol, which is generally preferred in the common situation where an intermediate firewall, Network Address Translation (NAT) or gateway device 402 - Use of a connectionless protocol suitable to the wireless network 403, such as UDP/IP, which is generally preferred when no network device restricting terminal contact 402 is present as this reduces resource usage on the server when large number of mobile terminals is served - Use of an alternate connection-oriented protocol such as HTTP suitable for mobile push operation through more strict firewalls and proxy Internet connections - Use of a secure protocol such as HTTPS, IP-Sec, secure IP6 or a proprietary security protocol to identify the communicating parties, prevent message interception by 3rd parties, and prevent message modification by 3rd parties - Use of a periodic message from the client to the server or server to the client to the effect that client socket, server socket, intermediate Network Address Translation (NAT) devices, intermediate firewalls, and intermediate gateway devices do not time expire the connection - Use of configuration messages to the firewall (FIG 4, A) using a firewall or other security configuration protocol, such as Universal Plug and Play, to open a path for server-initiated connection-oriented or connectionless messaging to the client

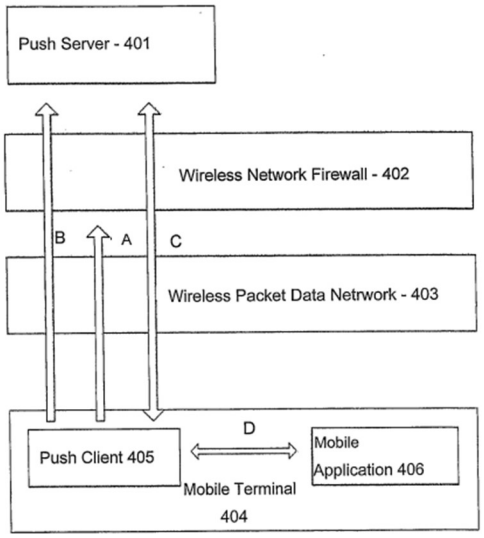
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 394 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	- Use of periodic messages initiated by the client 405 or by the server 401 to the effect that the receiving application is informed that their messages are continuing to be received, as indicated by response data from the receiving party to the sending party - Use of longer or shorter timing between such periodic messages to match the combined needs of all such network devices and protocols involved in transmitting communication, as adjusted based on the success or failure of earlier messages to successfully traverse the open network connection - Use of an alternate push server network address such as an address accessible with fewer security and performance restrictions which may be more available, more accessible, lower cost, lower latency, less busy, higher bandwidth or otherwise preferred at the present time and from the present network location of mobile terminal 404. - Use of one or more alternative mobile terminal network connection settings, or variation of the parameters within the settings. - Use of an alternate connection route through a second mobile or fixed terminal accessible to the mobile terminal over an alternate wireless network or network connection and capable of communication with a push server. In an embodiment of the invention, the mobile terminal 404 keeps a prioritized data structure listing the preferred communication protocols, server addresses, periodic message timings, network access settings, and alternate communication routes which it should use. Automated and manually assisted use of this structure should then result in a connection to one or more push servers, however if one of these connections which the mobile terminal is programmed to expect is not available, the mobile terminal can choose the sleep that function of the client software and attempt re-connection at a later time, repeating in like fashion with the same or different network connection parameters as described above until a connection is established or other user intervention. At any time after a connection is established between the client 405 and the server 401 , the client may push a message to the server or the server may send a push message to the client. This is illustrated by means of the arrow C in Figure 4. These messages through a previously established, connection-oriented protocol

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 395 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	such as TCP/IP, SSL, HTTP or HTTPS arrive quite rapidly, more rapidly than if the connection where established only when needed. The associated protocol overhead such as 3-way handshake during connection establishment thus occurs before the actual need for message transmission rather than needing to occur at time or transmission and thus delaying the transmission. A connectionless protocol, such as UDP/IP, may also used in cases where the connection-oriented protocol is not possible or not desirable because of the push server resources used by every static instance of a connection-oriented protocol. Connectionless communication is also used when a message needs to get through to the server 401 quickly, such as when occurs when the client 405 notifies the server 401 of an terminal event, user interface event or application event but the connection-oriented protocol connection is not connected or is already busy transferring information. Such a selective use of this high priority, low latency communication channel from terminal 404 to server 401 gives the server the opportunity to in return push to the client 405 a service triggered by the event begin communicated by the client 405 as will be described in more detail below. The sending of such a return message is frequently time critical and the fastest available combination of techniques will be used. In the preferred embodiment of the invention, the most basic message pushed from the server 401 to the client 405 is a data packet containing no information. Additional binary or text information specifying which mobile application 406 from a plurality of such applications may optionally be included. Additional binary or text information to be passed to the default or specified mobile application may optionally be included. Additional binary or text information specifying how upon receiving such a message the client 405 will notify and optionally ask the user if the action should be performed may optionally be included. Upon receiving such a message, appropriate automated actions and user interface media capabilities of the mobile terminal as specified in the command C from the server 401 are executed by the push client 405. The most common case is playing a sound, asking the user if the mobile application should be launched, and if affirmative passing the command details (arrow D) to the specified mobile application, module or additional feature within the same application suite, launching the application, module or feature if needed. Such action may include optionally proceed without audio, with network downloaded audio, or without first requesting confirmation from the user as specified by user preferences and command details. The push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server. The trigger may, among others, include an alarm, notification, or measurement result received to the push server 401 from another device or system. With a push message, the server 401 may direct the

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 396 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	mobile application 406 to make use of additional network resources either on the push server or other server. The push client 405 may in turn at any time accept data D from each of a plurality of mobile applications 406 and direct such to the push server 401 , possibly for redirection through push server API to other servers, mobile terminals, push clients and computing devices." Houghton, pp. 16-21. Figure 4  *Id.*, Fig. 4

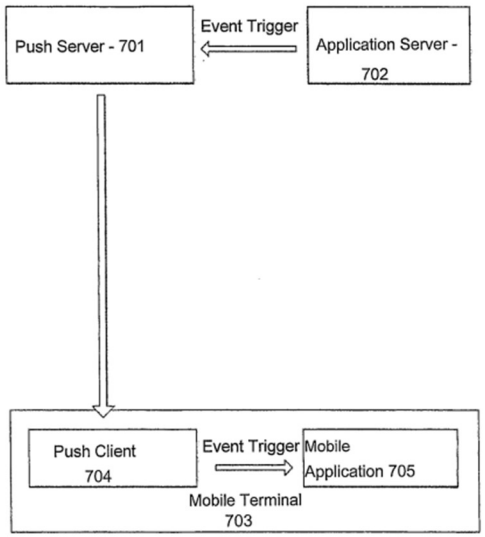
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 397 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	"In the preferred embodiment of the invention, the most basic message pushed from the server 401 to the client 405 is a data packet containing no information. Additional binary or text information specifying which mobile application 406 from a plurality of such applications may optionally be included. Additional binary or text information to be passed to the default or specified mobile application may optionally be included. Additional binary or text information specifying how upon receiving such a message the client 405 will notify and optionally ask the user if the action should be performed may optionally be included. Upon receiving such a message, appropriate automated actions and user interface media capabilities of the mobile terminal as specified in the command C from the server 401 are executed by the push client 405. The most common case is playing a sound, asking the user if the mobile application should be launched, and if affirmative passing the command details (arrow D) to the specified mobile application, module or additional feature within the same application suite, launching the application, module or feature if needed. Such action may include optionally proceed without audio, with network downloaded audio, or without first requesting confirmation from the user as specified by user preferences and command details. The push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server. The trigger may, among others, include an alarm, notification, or measurement result received to the push server 401 from another device or system. With a push message, the server 401 may direct the mobile application 406 to make use of additional network resources either on the push server or other server. The push client 405 may in turn at any time accept data D from each of a plurality of mobile applications 406 and direct such to the push server 401 , possibly for redirection through push server API to other servers, mobile terminals, push clients and computing devices." *Id*., p. 21. "Figure 7 illustrates an example of an "COMMAND PUSH" procedure wherein the push server 701 is triggered by a trigger event as for example from an application server 702 to push an application command message to the push client 704 and thereby initiate a mobile terminal client trigger event in a mobile application 705 from a plurality of such applications, program modules and user interface features on the terminal 705, The trigger event from push client 704 to mobile application 705 may optionally include commands or data affecting the aspect of visual presentation of mobile application 705. The even trigger mechanism may be any electronic process mechanism as for example computer software calls to load or

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 398 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	configure a class, library, dynamic link library, M I Diet, Symbian application, Linux daemon or other executable object. The trigger mechanism may be between separate parts of a larger application or application suite, or may be an IP-triggered application launch and data transfer as in the case of mobile application launched by contact to a specified IP port. The potential applications of the COMMAND PUSH include but are not limited to mobile applications for enterprise connectivity, back office integration, database access, scientific and industrial process and equipment monitoring and control, pictures transfer, mobile gaming, application data transfer, mobile electronic commerce, application alarms, reminders, multimedia interactivity, corporate calendaring, personal messaging and group messaging, etc. The push server 701 may in combination with user preferences stored in either the push server 701 or push client 704, optionally request the push client 704 to display or otherwise as appropriate for the mobile terminal or appliance notify the user asking permission for launching mobile application 705 prior to application launch. The preferred embodiment is for the behavior, language and content of this information to be configured on the server 701 based on the service and preferences of the individual receiving the message. In such cases an optional informational message will be presented to the user and such information messages are associated for instance with the menu option transmitted to the client 704 by the server 701. In an embodiment of the invention, the client 704 is also capable of receiving and storing variables which configure behavior of the client 704. The server 701 may thus specify the network connection preferences, notification sounds, and other behavior of the client 704. These changes facilitate the efficient and manageable operation of the entire system and can be used for automated remote maintenance of push client 704, mobile applications as 705, and configurable aspects of underlying mobile terminal 703, for example changing the username, password, and security certificates stored in the mobile terminal." *Id.*, pp. 21-23.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 399 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Figure 7  *Id.*, Fig. 7. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, and/or Houghton, it would have been obvious in view of the disclosures in Blackberry Push and/or Houghton in combination with the knowledge of a person of ordinary skill in the art.
17: The networked system of claim 1, at least a given one of	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, at least a given one of the devices further comprising a secure interprocess communication service separately

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 400 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
the devices further comprising a secure interprocess communication service separately secured from the secure Internet data message link, the device link agent for the given device causing messages to be securely delivered to a software process by initiating delivery of each such message on the secure interprocess communication service.	secured from the secure Internet data message link, the device link agent for the given device causing messages to be securely delivered to a software process by initiating delivery of each such message on the secure interprocess communication service. Blackberry Push employs on-device delivery through a device link agent, e.g. MDS runtime, to target specific applications separate from the network-channel encryption (BES Transport/HTTPS). *See, e.g.*, • MDS Studio 4.1, p. 8: BlackBerry MDS Runtime The BlackBerry MDS Runtime™ is the container-based execution environment for BlackBerry MDS Applications. The BlackBerry MDS Runtime contains a control center application that enables users to search for and manage BlackBerry MDS Applications and view application information. The BlackBerry MDS Runtime is designed to provide the underlying services that are used by BlackBerry MDS Applications, including the user interface, data storage, and client-server communication services. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art. • MDS 4.1 Overview, p. 9: The BlackBerry MDS Application Integration Service supports web services and other standard mechanisms for integrating mobile applications with enterprise applications. The application integration service also manages transmitting application data messages between BlackBerry MDS Studio applications and back. To the extent this limitation is not disclosed by Blackberry Push, it would have been obvious to a POSITA based on Blackberry Push alone or in combination with (i) the general knowledge of a POSITA at the time of application and/or (ii) the teachings of any one or combination of the following references: Lazaridis,

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 401 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Fok. **Combination with Lazaridis** Lazaridis and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Both Lazaridis and Blackberry Push disclose the transmission of multimedia messages over wireless networks and seek to improve interoperability and security for mobile communications. For example, both Blackberry Push and Lazaridis attempt to solve the problem of reducing the need for manual user intervention, allowing applications to respond automatically to network events or server commands. Lazaridis, Abstract; MDS Studio 4.1, p. 15. Lazaridis discloses a server that pushes messages to a client on a mobile terminal in a wireless network. Lazaridis, 5:20-23. Further, Lazaridis discloses a network server designed to redirect user data to the mobile data communication devices in response to triggering events. *Id.* Lazaridis teaches the network server could be located on an Internet Service Provider's system and accessible only through the Internet." *Id.*, 4:19-38. A POSITA would have found it obvious to combine Lazaridis's push-type command to mobile communication devices by a remote server with Blackberry Push's system. This combination of Blackberry Push and Lazaridis would involve only routine engineering leading to predictable results. Additionally, a POSITA would have found it obvious to use Blackberry Push, as disclosed by Lazaridis, as the communication protocol because Blackberry Push was a well-known, industry-leading push system. The widespread adoption and documentation of Blackberry Push would have motivated the use of Blackberry Push with Lazaridis's push messaging system without requiring inventive skill.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 402 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Lazaridis, as evidenced by the example citations below, discloses the networked system of claim 1, at least a given one of the devices further comprising a secure interprocess communication service separately secured from the secure Internet data message link, the device link agent for the given device causing messages to be securely delivered to a software process by initiating delivery of each such message on the secure interprocess communication service. *See, e.g.*, "In addition to the functionality noted above, the redirector program provides a set of software-implemented control functions for determining the type of mobile data communication device and its address, for programming a preferred list of message types that are to be redirected, and for determining whether the mobile device can receive and process certain types of message attachments, such as word processor or voice attachments." Lazaridis, 3:36-42. "If the redirector is configured so that the mobile data device cannot receive and process word processor or voice attachments, then the redirector routes these attachments to an external machine that is compatible with the particular attachment, such as an attached printer or networked fax machine or telephone. Other types of attachments could be redirected to other types of external machines in a similar fashion, depending upon the capabilities of the mobile device." *Id.*, 3:50-58. "Once an event has triggered redirection of the user data items, the host system then repackages these items in a manner that is transparent to the mobile data communication device, so that information on the mobile device appears similar to information on the user's host system. The preferred repackaging method includes wrapping the user data items in an E-mail envelope that corresponds to the address of the mobile data communication device, although, alternatively, other repackaging methods could be used with the present invention, such as special-purpose TCP/IP wrapping techniques, or other methods of wrapping the user selected data items. The repackaging preferably results in E-mail messages generated by the user from the mobile device to be transmitted from the host system, thus enabling the user to appear to have a single E-mail address, such that the recipients of messages sent from the mobile communications device do not know where the user was physically located when the message was first sent. The repackaging also permits both

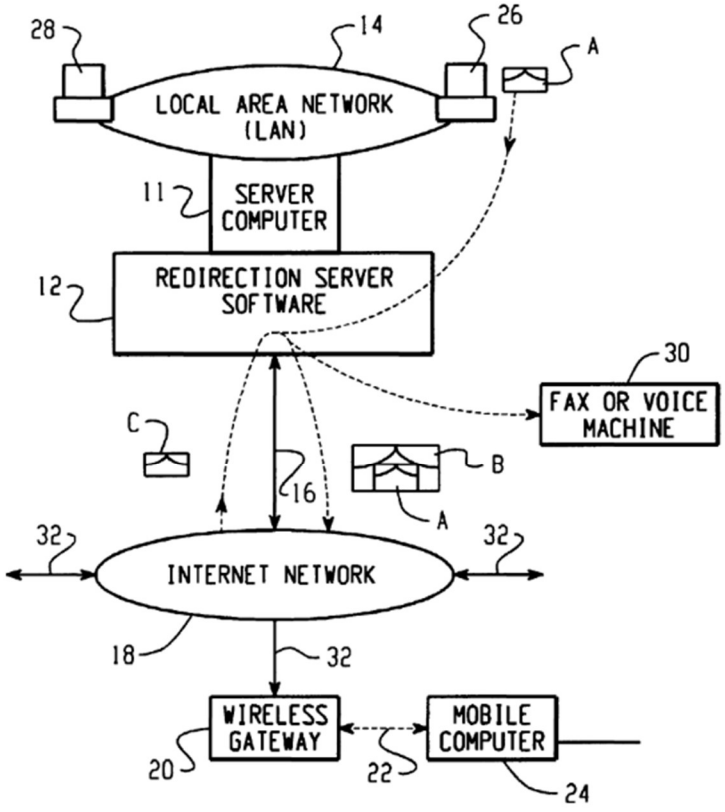
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 403 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	messages to the mobile device and sent from the mobile device to be encrypted and decrypted as well as compressed and decompressed." *Id.*, 3:66-4:19. "FIG. 2 is a system diagram showing the redirection of user data items from a network server (host system) to the user's mobile data communication device, where the redirector software is operating at the server. FIG. 3 is a block diagram showing the interaction of the redirector software with other components of the host system in FIG. 1 (the user's desktop PC) to enable the pushing of information from the host system to the user's mobile data communication device." *Id.*, 5:20-28. "The redirection program could be configured to send the text of the E-mail to the remote device, to send the word processing document to a networked printer located near the user, to send the video clip to a store accessible through a secure connection through the internet and to send the audio clip to the user's voice mail system." *Id.*, 6:22-28. "The mobile data communication device 24 includes software program instructions that work in conjunction with the redirector program 12 to enable the seamless, transparent redirection of user-selected data items. FIG. 4 describes the basic method steps of the redirector program 12, and FIG. 5 describes the steps of the corresponding program operating at the mobile device 24. In an alternative embodiment of the present invention, not explicitly shown in the drawings, the mobile device 24 also includes a redirector program. In this embodiment, user selected data items can be replicated from the host to the mobile device and vice versa. The configuration and operation of the mobile device 24 having a redirector program is similar to that described herein with respect to FIGS. 1-4." *Id.*, 6:42-55. "After the redirector has determined that a particular message should be redirected, and it has prepared the message for redirection, the software 12 then sends the message A to a secondary memory store located in the mobile device 24, using whatever means are necessary. In the preferred embodiment this method is to send the message back over the LAN 14, WAN 18, and through the wireless gateway 20 to the mobile data communication 24. In doing so, the redirector preferably repackages message A as an E-mail, with an outer

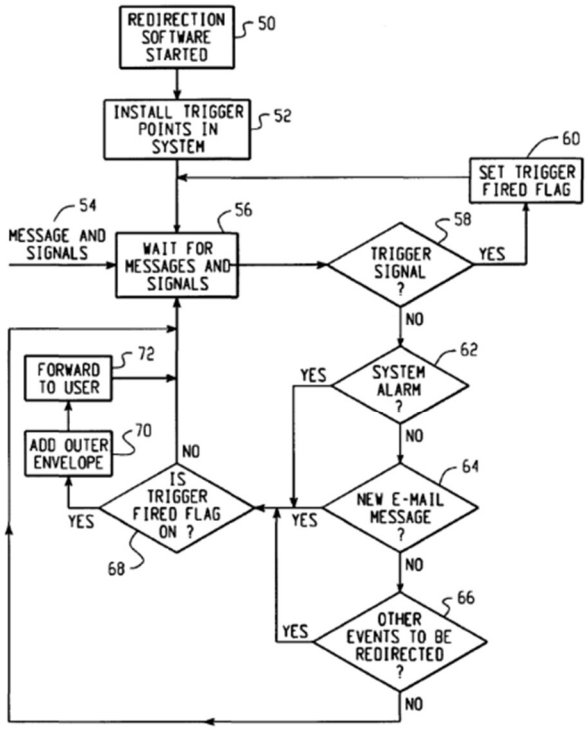
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 404 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	envelope B that contains the addressing information of the mobile device 24, although alternative repackaging techniques and protocols could be used, such as a TCP/IP repackaging and delivery method (most commonly used in the alternative server configuration shown in FIG. 2). The wireless gateway 20 requires this outer envelope information B in order to know where to send the redirected message A. Once the message (A in B) is received by the mobile device 24, the outer envelope B is removed and the original message A is placed in the secondary memory store within the mobile device 24. By repackaging and removing the outer envelope in this manner, the present invention causes the mobile computer 24 to appear to be at the same physical location as the host system 10, thus creating a transparent system." *Id.*, 8:32-55. "When the user replies to this message, or when the user authors a new message, the software operating at the mobile device 24 adds a similar outer envelope to the reply message (or the new message) to cause the message to be routed first to the user's host system 10, which then removes the outer envelope and redirects the message to the final destination, such as back to computer 26." *Id.*, 9:2-9. "At step 82, the mobile device waits for messages and signals 84 to be generated or received. Assuming that the redirector software 12 operating at the user's desktop system 10 is configured to redirect upon receiving a message from the user's mobile device 24, at step 86, the user can decide to generate a command message that will start redirection. If the user does so, then at step 88 the redirection message is composed and sent to the desktop system 10 via the wireless network 22, through the wireless gateway 20, via the Internet 18 to the LAN 14, and is finally routed to the desktop machine 10. In this situation where the mobile device 24 is sending a message directly to the desktop system 10, no outer wrapper is added to the message (such as message C in FIGS. 1 and 2)." *Id.*, 13:6-19.

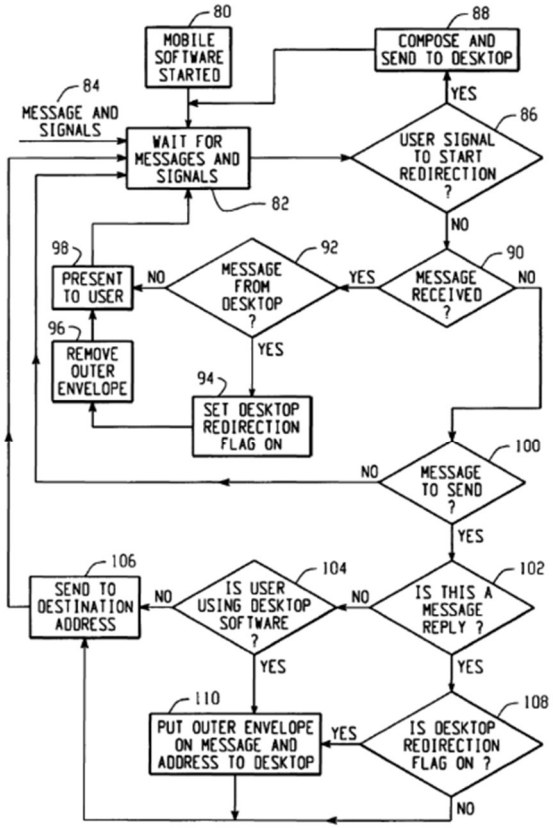
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 405 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 2 *Id.*, Fig. 2.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 406 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 4 *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 407 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 Fig. 5 *Id.*, Fig. 5.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 408 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	Combination with Fok Fok and Blackberry Push are analogous art to the '320 patent and each other; they are in the same field of endeavor (e.g., computer systems and/or networks) and/or reasonably pertinent to the problems faced by the inventor (e.g., secure computer messaging across a network). Blackberry Push and Fok are from the same field in that they disclose buffering and transmitting data using secured TCP/IP protocols. Fok, ¶¶9, 37, 56, 71, Figs. 2-3. Fok describes "secure inter-application communication" in a "mobile operating environment." *Id*., ¶¶36-37, 81, 106-108, Figs. 4, 8-9. Fok uses a "handshake" between a "primary" and one or more "recipient" applications to establish recipients are trusted using "a list of unique identifiers" before transmitting information between the two. *Id*., ¶¶47-53, 96-102, 106-108, Figs. 7-10. A POSITA would have been motivated to combine Blackberry Push and Fok to provide a "secure inter-application communication" like Fok's so that two or more applications residing on a single UE/wireless device may transmit application-specific MMs to one another through the MMS User Agent. Fok, as evidenced by the example citations below, discloses the networked system of claim 1, at least a given one of the devices further comprising a secure interprocess communication service separately secured from the secure Internet data message link, the device link agent for the given device causing messages to be securely delivered to a software process by initiating delivery of each such message on the secure interprocess communication service. *See e.g.*, "The subject disclosure provides for secure efficient and computationally light communication between applications in a mobile environment. Various conventional mechanism for securing electronic communication exist, however, such conventional mechanisms typically utilize complex and computationally intensive encryption and decryption algorithms. Such mechanisms consume a relatively large amount of electrical power due to the computation requirements. Mobile operating environments, however, benefit from low power consumption due to reliance on battery power. Although some applications have been modified for the low power consumption in the mobile environment, such

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 409 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	modifications have been lax in securing communications. One reason is due to advances and anticipated advances in system processing and memory technology. For instance, such advances have led many application researchers to avoid modifying communication protocols to reduce power consumption, in favor of anticipated advances in processor size, parallel processing, memory size, and the like." Fok, [0036]. "Also disclosed is inter-application communication utilizing a round-trip concept. Specifically, applications can wait for a concurrent request or response to be completed prior to initiating and/or processing subsequent requests/responses. Round-trip communication can be particularly efficient in a synchronous environment. Common conventional communications, such as transmission control protocol/Internet protocol (TCP/IP), typically transmit a string of data packets prior to awaiting a response from a target application. In addition, in secured TCP/IP communications, decryption and/or understanding received data often depends on receiving all data packets in a transmission. If one data packet is lost, much of the transmitted data can be unusable. Instead, the subject disclosure provides for re-transmission of requests and/or responses if data or a security identifier is lost by an initiating or recipient application. Such a system is dramatically different from the widely successful TCIP/IP architecture utilized in the Internet and most private intranets as well as other message passing and/or shared memory environments. As a result, the round-trip and re-transmission architecture described herein is contrary and distinct from conventional communication architectures, which, because of their success and popularity, have led research away from efficient, synchronous and low power secure data exchange described herein." Fok, [0037]. "In addition to the foregoing, disclosed is an inter-application communication architecture that employs random numbers, generated by particular applications, to identify and/or authenticate communication provided by another application. For instance, an initiating mobile application can generate a random number and provide that random number to a recipient application. Subsequent responses to the initiating mobile application should include the random number, or an increment thereof identifying a particular round-trip, to identify communication as coming from the recipient, and not a spoofing program or other malware. According to further aspects, the random number(s) can be encrypted utilizing a computationally un-intensive hash function (e.g., SHA-1, SHA-224, SHA-256, SHA-386, SHA-512, SHA-2, message digest

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 410 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	algorithm 5 MD5 or the like) to obfuscate the random numbers without requiring intensive and high power encryption/decryption algorithms. Additionally, the hashed random number(s) can be truncated as required by a particular communication interface between the initiating and recipient applications. Accordingly, disclosed is a mechanism to provide secure inter-application communication beneficial for a mobile environment." *Id.*, [0038]. "Primary application 102 can initiate a handshake that establishes a secure communication session with at least one other application (104). Although much of the subject disclosure involves a single communication between two applications, the application (102) can maintain multiple secure communication sessions with multiple other applications (104) concurrently. The handshake can be utilized to create a trusted "channel" between the primary application 102 and a recipient application 104. As a result, arbitrary data (e.g., binary, etc.) can be exchanged between the primary application and the recipient application over the trusted channel utilizing, for instance, address pointers that identify data, instructions, and so on." *Id.*, [0047]. "A handshake can be initiated by the primary application 102 by sending a handshake request that includes at least a unique identifier of the primary application 102 or a random number generated for the handshake, or both. The unique identifier is an application ID associated with the primary application, that can be utilized by a recipient application to verify that the primary application is to be trusted. Typically, an operating system (106) can require certain standards be met for applications designed to execute on the operating system (106). One such requirement can be to incorporate a list of unique identifiers of trusted primary applications. A recipient application 104 having such a list could then verify that a received handshake request is sent by an application authorized to make the request. The recipient (104) could then trust and engage in communication with such application." *Id.*, [0048]. "In addition to a unique identifier of a primary application, a handshake request can also include a random number (e.g., 32 bit, 16 bit, 64 bit, and so on) generated for the handshake. The random number can be sent to an intended recipient application and incorporated into responses sent to the primary application 104 by the recipient application 104. As a result, a response that includes the random number (or, e.g., a predictable

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 411 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	variation thereof, discussed below) can be identified as originating from the intended recipient. According to additional aspects, the random number can be hashed utilizing a particular hash function (e.g., SHA-1, SHA-2, or the like) and/or truncated for transmission over a particular communication interface (106). A hashed random number can be much more difficult to decrypt, providing significant protection from spoofing attacks. Truncation can be utilized to transmit the random number utilizing various protocols and/or communication interfaces (106) that support different size numbers. For instance, if a generated random number identifying a handshake session is 32-bit, and a communication interface with a recipient application (104) enables only 16-bit data transfer, the 32-bit random number can be hashed and then truncated to 16-bit for transmission. The recipient can then hash and/or truncate the random number and compare it to the received hashed and/or truncated number to verify whether a sender of communication is the initiator (102) of the handshake." *Id.*, [0049]. "Recipient application 104 can receive a handshake request and verify whether a sender (102) is a trusted primary application. Particularly, a unique identifier can be extracted from the handshake request and compared with a list of trusted primary application identifiers. If the extracted identifier matches an identifier on the list, a handshake response can be sent to the sender (102), establishing a secure channel between sender (102) and recipient (104). If the extracted identifier does not match an identifier on the list, the hand shake and any subsequent communication can be ignored until a verified handshake is received." *Id.*, [0050]. "Additionally, the recipient application 104 can extract and store a random number included within the handshake request. As a specific example, the random number can be stored in temporary memory (not depicted) to make it more difficult for malicious programs to extract the random number and spoof the recipient application's (104) identity. Upon verifying the unique identifier included within the handshake request, discussed above, the recipient application 104 can send a handshake response that includes the random number. Particularly, by including the random number the recipient application 104 can identify itself to the primary application 102. According to particular aspects, the recipient application 104 can hash and/or truncate the random number to make spoofing the recipient application's (104) identity (e.g., to the primary application 102 that generated and therefore is configured to trust the random number) more

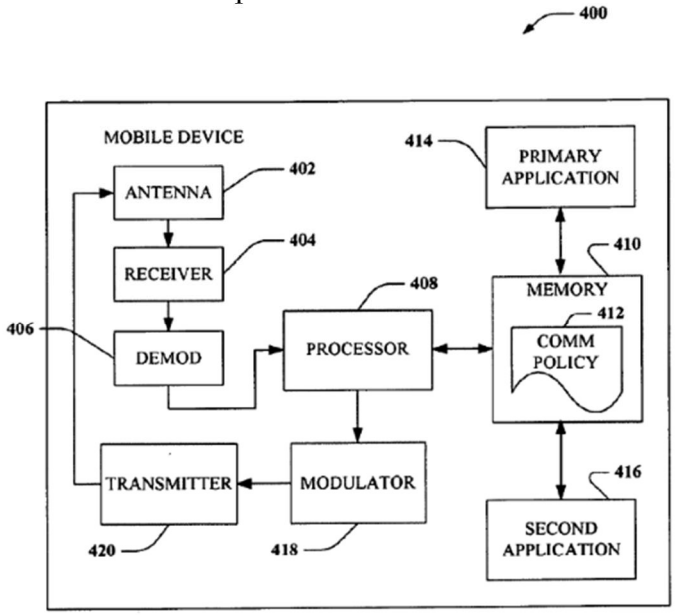
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 412 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	difficult and/or make the hashed number more compatible with particular communication interfaces (106), as discussed above." *Id.*, [0051]. "Furthermore, the recipient application 104 can also generate and include a second random number into the handshake response. The second random number can be utilized by the primary application 102 to identify itself in subsequent communication with the recipient application 104. Subsequent communication by the primary application 102 can hash and/or truncate the second random number, as discussed above, to protect integrity of the second random number." *Id.*, [0052]. "According to further aspects, subsequent receiving a handshake response from the recipient application 104, primary application 102 can send a data event to the recipient application 104. The data event can, for instance, involve any data or data structure suitable to a mobile environment, incorporate a request for data, or a request to carry out an instruction (e.g., gathering diagnostic information, incorporate a configuration parameter, and so on), or a combination thereof or of the like. The request can include data within the request itself, or can utilize address pointers to identify data in shared memory, and so on. In addition to data/instructions, the data event can incorporate the second random number generated by the recipient application (e.g., where the second random number is hashed and/or truncated) to identify and/or certify that a sender of the data event is the primary application 102." *Id.*, [0053]. "FIG. 4 illustrates an example mobile device 400 that can execute mobile applications and provide secure and low power transmission between the applications. Particularly, mobile device 400 can provide a common platform (408, 410, 412) facilitating communication between mobile applications (414, 416), and/or can provide a platform for a first application (414, 416) to wirelessly interface with a remote device (not depicted, but see FIG. 5, infra) containing a second application (414, 416), where an intervening radio network, core mobile network and/or IP network, or the like, provides a communication interface between the remote devices (400, 504a, 504b). Accordingly, system 400 can provide secure communication between mobile applications coupled by a processing and communication architecture (e.g., processor 408, memory

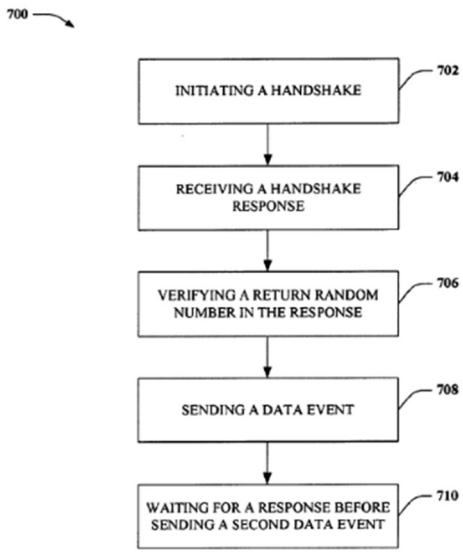
Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 413 of 422

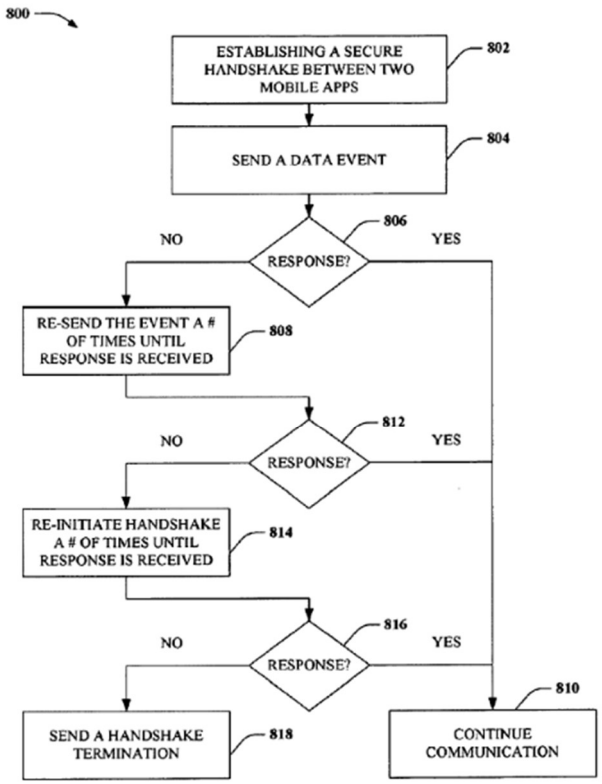
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	410, communication policies 412, and suitable bus structures) of a single device (400) or of remote, networked devices (400, 504a, 504b)." *Id.*, [0081]. "FIG. 9 depicts an example methodology 900 for incrementing round-trip communication in a secure communication between mobile applications. At 902, method 900 can send a handshake initiation including at least a random number. The random number can be utilized by a recipient application to securely identify communication as originating from such recipient (e.g., because the random number was sent only to the recipient application). The handshake can further include a unique ID of an initiating application, as described herein." *Id.*, [0106]. "At 904, a response to the handshake can be received. At 906, a determination can be made as to whether the response includes the random number or an appropriate version thereof (e.g., a hashed and/or truncated version). If not, method 900 can proceed to 910 where a handshake termination is sent, and optionally all data related to the handshake session, including the random number, received numbers, data transmitted, and so on, can be deleted. If an appropriate number is included within the response received at reference number 904, method 900 can proceed to 908 where the received random number can be incremented and incorporated within a data event sent to the recipient application. The incremented random number can identify a concurrent round trip communication session, as described herein." *Id.*, [0107]. "At 912, a determination is made as to whether a response to the data event, if received, includes an appropriate increment of the random number. If not, method 900 proceeds to 910 where a handshake termination can be initiated. Particularly, it can be assumed that if an improper increment is received, the data event response was sent by an unauthorized application (e.g., the random number was compromised) or the secure communication policies of the recipient application have become corrupt, or the like. If, at reference number 912, an appropriate increment to the random number is received, method 900 can proceed to 914 where an additional increment to the random number is included within a subsequent data event. At 916, another determination can be made as to whether a response to the Subsequent data event, if received, includes an appropriate additional increment of the random number. If so, method 900 can proceed to 918

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 414 of 422

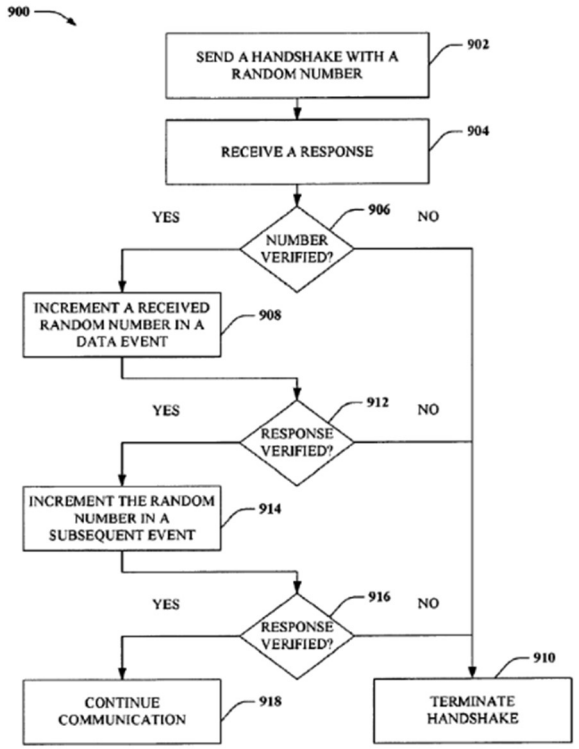
ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	where communication is continued in the fashion described by method 900 and/or as described elsewhere herein. If an appropriate increment is not determined at 916, method 900 can proceed to 910 where a handshake termination, described above, can be initiated. As described, method 900 can provide round-by-round analysis of incremented random numbers identifying a handshake session as well as an appropriate data event round-trip to secure communication between mobile applications." *Id.*, [0108].  *Id.*, Fig. 4.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 415 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **FIG. 7** *Id.*, Fig. 7.

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **FIG. 8** *Id.*, Fig. 8.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 417 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	 **FIG. 9** *Id.*, Fig. 9.
18: The networked system of claim 1, wherein at least one	Blackberry Push, as evidenced by the example citations below, discloses the networked system of claim 1, wherein at least one of the secure Internet data messages comprises multiple identifier/data pairs.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 418 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
of the secure Internet data messages comprises multiple identifier/data pairs.	Blackberry Push's push and application messages comprise structured headers/fields (identifier/data pairs). *See, e.g.*, Blackberry Push discloses sending a message with destination application's identifier and application data (identifier/data pair). *See* claim 1, *supra*. Blackberry Push supported custom application builds from developers, allowing BlackBerry apps using BlackBerry MDS Studio or Java APIs. Such applications were capable of aggregated data from multiple sources and presented it as a single message or notification. *See, e.g.*, • MDS 4.1 Overview, p. 7:

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 419 of 422

Application models and development environments

Application model	BlackBerry device software	Development environment
Browser model: BlackBerry MDS Browser applications	BlackBerry Browser	• To provide mobile access to intranet content and to create web-based wireless applications that BlackBerry users can access through the BlackBerry Browser, use existing web content development tools. Extend existing web-based applications wirelessly with little custom development. See "Making web content accessible wirelessly" on page 8 for more information. • To learn how to optimize new and existing web applications for the BlackBerry Browser using standard web development tools, use the *BlackBerry Wireless Handheld Browser Content Developer Guide*. • To create and test vector graphics and animations for BlackBerry devices, use the tools and simulators that the Plazmic Content Developer's Kit for BlackBerry (Plazmic CDK) provides.
Web service model: BlackBerry MDS Studio applications	BlackBerry MDS Runtime	To create lightweight applications that provide mobile access to enterprise web services, use the BlackBerry MDS Studio. The BlackBerry MDS Studio is a powerful visual application design and assembly tool that enables application developers to quickly create rich-client applications that integrate with enterprise web services using a component-based drag and drop approach. Application developers can create robust functionality without Java programming using the BlackBerry MDS Studio. See "Creating BlackBerry MDS Studio Applications" on page 11 for more information.
Traditional client/server model: BlackBerry MDS Java applications	BlackBerry JVM	To create custom Java applications for the BlackBerry device that provide a sophisticated user interface and navigation, robust data management, and flexible support for custom data formats, use the BlackBerry JDE. The BlackBerry JDE provides development and simulation tools that enable application developers to build rich-client Java Platform, Micro Edition (Java ME) applications, which integrate with standard application servers, for BlackBerry devices. See "Creating BlackBerry MDS Java applications" on page 15 for more information.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 420 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	• MDS 4.1 Overview, p. 11: Creating BlackBerry MDS Studio Applications BlackBerry MDS Studio applications combine the light-weight, programmatic simplicity of BlackBerry MDS Browser applications with the functional sophistication of BlackBerry MDS Java applications. BlackBerry MDS Studio applications enable BlackBerry device users to access the functionality of remote web services from their BlackBerry devices. Web services can be located on a server that resides either inside or outside of a corporate firewall; they are web-accessible application components that provide a set of well-defined operations (for example, a conversion utility). BlackBerry MDS Studio applications access these web services wirelessly through HTTP or HTTPS connections and communicate with them using SOAP. BlackBerry MDS Studio applications use the operations performed by web services. Applications can be designed to access some or all the operations of a single web service, or they can be designed to access operations from multiple web services from within a single BlackBerry MDS Studio application. The web service development model gives application developers design flexibility. BlackBerry MDS Studio applications also require less coding sophistication than traditional client/server application models because the web service, the BlackBerry MDS Runtime, and the BlackBerry MDS Services provide most of the complex programmatic functionality. The reduced development complexity and the connection service push capabilities enable application developers to produce applications with the same performance, responsiveness, and user experience of traditional Java client applications, but in less time, without the need for extensive coding knowledge, and with fewer demands placed on the processing and memory constraints of the BlackBerry device. Java Development 4.0, p. 121: There are two types of push applications: • Browser push applications: Web content is sent to the browser on the handheld. The BlackBerry Browser configuration supports Mobile Data Service push applications. The WAP Browser configuration supports WAP push applications. The Internet Browser configuration does not support push applications. See the BlackBerry Browser Developer Guide for information on writing a browser push application. • Client/server push applications: Data is sent to a custom Java application on the handheld. Client/server push applications consist of a custom client application for the handheld and a server-side application that pushes content to it. This approach provides more control over the type of content that you can send out and

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 421 of 422

ASSERTED CLAIMS	PRIOR ART REFERENCE: Blackberry Push
	how this data is processed and displayed on the handheld compared to browser push applications. Further, a POSITA would have understood that multiple Blackberry applications on a user's device receive messages with application data from the server. A POSITA would have understood, in some situations, application data/identifiers for multiple applications should be consolidated in a single message. For example, when a user's device becomes available and connects to the network, multiple messages may have been queued. Consolidating application data/identifiers for multiple applications into a single message for delivery (multiple identifier/data pairs) would provide network efficiencies over sending multiple distinct messages for each application. Further, to the extent Plaintiff contends this element is not expressly or inherently disclosed by Blackberry Push, it would have been obvious in view of the disclosures in Blackberry Push, or in combination with the knowledge of a person of ordinary skill in the art.

Headwater Research LLC
Ex. 2007, IPR2026-00106
Page 422 of 422