SIP Based Push Requirements

Candidate Version 1.0 – 30 Jan 2005

Open Mobile Alliance

OMA-RD-SIP_Push-V1_0-20050130-C

Contents

Figures

Tables

1. Scope (Informative)

This document defines the use-cases and requirements for SIP based Push as a transport bearer for OMA Push. The document will maintain the compatibility with the existing WAP Push architecture; push message definitions, service definitions, charging [OMACharging].

Push Proxy Gateway Service specification [PPGService] defines PPG operations, which defines each push submission processing, before sending the push content to the client. It is out of the scope of SIP Based Push RD to define any new push submission processing.

2. References

2.1 Normative References

[IMSARCH]	IMS in OMA Architecture Document; URL:http://www.openmobilealliance.org/
[OMACharging]	"OMA Charging Requirements" URL:http://www.openmobilealliance.org/
[RFC2119]	"Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt
[RFC3261]	"SIP: Session Initiation Protocol", RFC3261, URL: http://www.ietf.org/rfc/rfc3261.txt?number=3261
[PPGService]	Push Proxy Gateway Service, WAP Forum™, WAP-249-PPGService, URL:http://www.openmobilealliance.org/
[Privacy]	OMA Privacy Requirements for Mobile Services: OMA-RD_Privacy-V1_0_0-20031104-A http://www.openmobilealliance.org/ftp/PD/OMA-RD_Privacy-V1_0_0-20031104-A.zip
[PushArch]	Push Architectural Overview, WAP Forum™, WAP-250-PushArchOverview, URL:http://www.openmobilealliance.org/
[PushMsg]	Push Message, WAP Forum™, WAP-251-PushMessage, URL:http://www.openmobilealliance.org/
[PushOTA]	"Push OTA Protocol Specification", WAP Forum™, WAP-235-PushOTA, URL:http://www.openmobilealliance.org/
[PushPAP]	"Push Access Protocol Specification", WAP Forum™, WAP-247-PAP, URL:http://www.openmobilealliance.org/
[PushSEC]	"WAP Push Security Requirements", version 1.0, URL:http://www.openmobilealliance.org/

2.2 Informative References

[OMADM]	OMA Device Management Specification™ version 1.1.2, URL:http://www.openmobilealliance.org/
[OMADictionary]	Dictionary for OMA Specifications V3.0.0, URL:http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

Push Client	See [OMADictionary]
Content	Subject matter (data) stored or generated at an origin server. Content is typically displayed or interpreted by a user agent on a client. Content can both be returned in response to a user request, or pushed directly to a client.
Device	Is a network entity that is capable of sending and/or receiving packets of information and has a unique device address. A device can act as either a client or a server within a given context or across multiple contexts. For example, a device can service a number of clients (as a server) while being a client to another server.
MMS Client	The MMS service endpoint located on the WAP client device.
MMS Proxy-Relay	This is the system element that the MMS Client interacts with. It provides access to the components that provide message storage services, and it is responsible for messaging activities with other available messaging systems. Some implementations may combine this component with the MMS Server.
Push Framework	The entire WAP push system. The push framework encompasses the protocols, service interfaces, and software entities that provide the means to push data to user agents in the WAP client.
Push Initiator	The entity that originates push content and submits it to the push framework for delivery to a user agent on a client.
Push OTA Protocol	A protocol used for conveying content between a Push Proxy Gateway and a certain user agent on a client.
Push Proxy Gateway	A proxy gateway that provides push proxy services.
Push Session	A WSP session that is capable of conducting push operations.
Server	A device (or service) that passively waits for connection requests from one or more clients. A server may accept or reject a connection request from a client. A server may initiate a connection to a client as part of a service (push).
Terminal	See " Push Client".
User	See [OMADictionary]
User agent	A user agent (or content interpreter) is any software or device that interprets resources. This may include textual browsers, voice browsers, search engines, etc.

3.3 Abbreviations

OMA	Open Mobile Alliance
GPRS	General Packet Radio Service
HTTP	Hypertext Transfer Protocol

IP	Internet Protocol
IMS	IP Multimedia Subsystem
MM	Multimedia Message
MMD	Multimedia Domain
OTA	Over The Air
OTA-HTTP	(Push) OTA over HTTP
OTA-HTTP-TLS	OTA-HTTP over TLS
OTA-SIP	(Push) OTA over SIP
OTA-WSP	(Push) OTA over WSP
PDP	Packet Data Protocol
PI	Push Initiator
PPG	Push Proxy Gateway
SIP	Session Initiated Protocol
SIMPLE	SIP for Instant Messaging and Presence Leveraging Extensions
SMS	Short Message Service
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WAP	Wireless Application Protocol
WSP	Wireless Session Protocol
WTLS	Wireless Transport Layer Security

4. Introduction (Informative)

Push Framework [PushArch] specification defines Push operation as client/server model, where there is no explicit request from the client before the server transmits its content. The Push Framework introduces a means to transmit information to a device without a user request. Where, the Push Initiator (PI) transmits the “push content” and “delivery instruction” to the Push Proxy Gateway (PPG). The PPG delivers the push content to the Push Client (henceforth referred to as "client" or "terminal") based on the delivery instruction. The Push Framework specifies the Push Access Protocol [PushPAP] and Push Over-the-Air Protocol [PushOTA], between PI and PPG, and between client and PPG, respectfully.

The IETF has defined the core set of technology for establishing multimedia session between two or more participants. Session Initiated Protocol (SIP) [RFC3261] provides session management and setup, user reachability and availability.

The SIP/IP Core Network provides a rich peer-to-peer media sessions, client-server sessions, and a smooth interaction of services (e.g. utilization of presence information by other services). In addition, it offers service control point, such as authentication and authorization for SIP clients and services, and centralized charging control point.

The work of OMA Push OTA over SIP (OTA-SIP) is to leverage these standards by transporting Push OTA announcements encapsulated in SIP messages. The restriction of the data size in SIP messages is defined in RFC3261 [RFC3261]. Figure 1 shows the Push Framework with OTA-SIP. The benefit of using OTA-SIP is lower cost of maintenance, higher interoperability, reuse of existing resources. High-level requirements can be summarize as the following:

- Enabling Push Proxy Gateway (PPG) to send Push content, and request delivery reports, when using a number of different accesses, or when roaming between different access networks, e.g., GPRS and WLAN
- Ability to have a single addressing scheme that is independent of any network access characteristics, i.e., a single SIP URI.
- Ability to advertise acceptance of certain push messages and route push messages from a single SIP address to a number of different locations, e.g., when roaming in a different network, using a different access technology, etc.

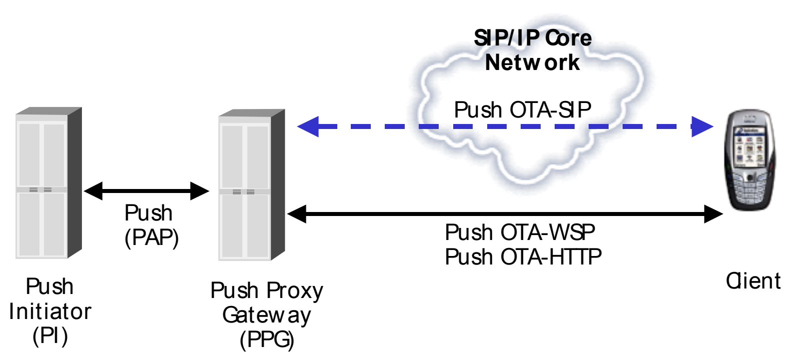

Figure 1: Push Framework

5. Use Cases (Informative)

This section identifies the Device Management, and MMS Notification use-cases for the OMA Push OTA over SIP (OTA-SIP). These use-cases are organized such that each enabler that is envisioned as using OTA-SIP is represented as a separate entity. This means that use-cases are drawn from each existing enabler that uses OMA push, as well as from future needs for OTA-SIP.

5.1 Use Case A, MMS

	Affected Areas				
	Device	**Connectivity**	**Enabling Services**	**Applications**	**Content**
Tickmarks (X)	X		X	X	
Additional Keywords					

Table 1: Affected Areas for Use Case A

5.1.1 Short Description

This use case illustrates the use of OTA-SIP in the MMS Notification functionality. The example describes the functions and roles of the various system elements in the MMS and Push frameworks.

5.1.2 Actors

- User A is the originator of the MM
- User B is recipient of the MM
- MMS Proxy-Relay server interact with the user via Push Proxy Gateway

5.1.2.1 Actor Specific Issues

- User A is uncertain if User B will receives the MM
- User B is uncertain that all incoming MMs are actually received or if some are thrown away
- Push Proxy Gateway may have several Push OTA methods

5.1.2.2 Actor Specific Benefits

- User A is reasonably certain that the MM reaches User B.
- User B is able to receive the MM
- Push Proxy Gateway has efficient and more simple way to deliver the MM

5.1.3 Pre-conditions

- User B terminal must support TCP/IP, MMS, and SIP.
- User B is registered to the SIP/IP Core Network (e.g. IMS).
- Both User A and User B subscribe to the MMS service.

5.1.4 Post-conditions

- User B received and watched the MM that User A intended.

- User A is informed that User B successfully receive the MM

5.1.5 Normal Flow

1) User A activates MMS Client.
2) User A selects or enters User B address, and composes/edits MM to be sent.
3) User A requests that MM is sent to User B.
4) MMS Client submits the message to its associated MMS Proxy-Relay and request delivery report.
5) MMS Proxy-Relay resolves the User B address, and forwards the MM to user B MMS Proxy-Relay.
6) The MM is stored by the MMS Server associated with the User B MMS Proxy-Relay.
7) User B MMS Proxy-Relay sends a MMS Notification to User B MMS Client via Push Proxy Gateway (PPG).
8) Using Push OTA-SIP, the PPG forwards the MMS Notification User B MMS Client.
9) The User B MMS client requests the MMS Relay/Server to initiate the retrieval process.
10) The MMS Relay/Server send the respond, which contains MMS control information and the MM content.
11) After receiving the content, the User B MMS Client sends an acknowledgement to the corresponding MMS Relay/Server, if requested by the MMS Relay/Server.
12) Upon receiving the acknowledgement, the MMS Relay/server sends the delivery reports to the User A MMS Client, via PPG.
13) Using Push OTA-SIP, the PPG forwards the MMS delivery report User A MMS Client.

5.1.6 Alternative Flow

None

5.1.7 Operational and Quality of Experience Requirements

OTA-SIP must maintain at least same level of user experience as the existing Push OTA (Push OTA WAP235).

5.2 Use Case B, New Service Device Management Provisioning

	Affected Areas				
	Device	Connectivity	Enabling Services	Applications	Content
Tickmarks (X)	X		X	X	
Additional Keywords					

Table 2: Affected Areas for Use Case B

5.2.1 Short Description

A user orders a new service from a service provider. The method of ordering the service could be using a browser to request the service to be activated, traditional phone call to the customer service desk, or at a retail store. Let's assume the user is ordering an online game service via existing subscription service provider. The ordering method is using the browser to request the service to be activated.

5.2.2 Actors

- User owns a multimedia terminal device, which it is capable for online game service.
- Cellular service provider provides cellular access, Internet connection, SIP/IP Core Network services, and hosting online game.

5.2.2.1 Actor Specific Issues

- User prefers automated, quickly and easily configuration for online game service.
- User wants to play game with friends.
- User wants to use a unified way of addressing for all of the services.
- Service provider wants proper configuration based on company game policy.
- Service provider wants a unified way of addressing the user.
- Service provider wants to provide ease of use experience to the user.
- User and service provider want to have a mutual authentication.

5.2.2.2 Actor Specific Benefits

- User has seamless user service provisioning.
- Service provider has a lower total cost ownership (TCO), centralize policy and configuration, and simpler deployment (e.g. addressing).

5.2.3 Pre-conditions

- User terminal has successfully associated, authenticated, and IP connectivity to the network.
- User terminal must supports SIP protocols.
- User is registered to the SIP/IP Core Network [IMSARCH]. The SIP registration establishes a trusted relationship between the terminal and the SIP/IP Core Network.
- PPG has a relationship with SIP/IP Core Network.

- User terminal is not configured for the online game service.

5.2.4 Post-conditions

- The required service settings are configured and available to the application in the terminal.
- After successfully configured, User accesses online game.

5.2.5 Normal Flow

1. The user orders online game service from service provider by using the browser to request the service to be activated.
2. The service provider accepts the service request from the user. The online gaming service requests the Device Management Server to configure the service.
3. The Device Management server, acting as a PI, sends an initial device management bootstrap message (OMA Client Provisioning message) to the Push Proxy Gateway (PPG) using PAP.
4. Using Push OTA-SIP, the PPG forwards the device management bootstrap message to the terminal, via SIP/IP Core Network.
5. The SIP/IP Core Network forwards the device management bootstrap message to the terminal.
6. Terminal receives and handles the Device Management bootstrap message, which contains the initial setting to use the device management server.
7. To start the configuration session the Device Management Server sends Device Management session initiation message via PPG. The PPG uses Push OTA-SIP to forward the message to the SIP/IP Core Network.
8. The SIP/IP Core Network forwards message to the terminal. Terminal responds and the Device Management configuration will continue in normal [OMADM] standard way.
9. After Device Management session is done, the user uses the new configuration setting to play online game.

5.2.6 Alternative Flow

None

5.2.7 Operational and Quality of Experience Requirements

OTA-SIP must maintain at least same level of user experience as the existing Push OTA [Push OTA].

5.3 Use Case C, Continuous Device Management Notification Support

	Affected Areas				
	Device	Connectivity	Enabling Services	Applications	Content
Tickmarks (X)	X		X	X	
Additional Keywords					

Table 3: Affected Areas for Use Case C

5.3.1 Short Description

After the service configuration is installed successfully, the Device Management Server may manage the settings by using OMA Device Management. The Device Management session startup notification message can be sent using OTA-SIP.

5.3.2 Actors

- User would like to be notified when the user online game service configuration has changed. User terminal is configured for continuous device management.
- Service provider Device Management Server provides a notification capability to the subscribed user.

5.3.2.1 Actor Specific Issues

- When online game service configuration changed in the network, user wants to have a capability to receive the notification before accepting the changes.
- Service provider wants user terminal up to date configuration.

5.3.2.2 Actor Specific Benefits

- User has the latest supporting service configuration.
- Service provider has a lower total cost ownership (TCO), centralize policy and configuration, and simpler deployment.

5.3.3 Pre-conditions

- User terminal has successfully associated, authenticated, and IP connectivity to the network.
- User terminal must supports SIP protocol.
- User registers to the SIP/IP Core Network [IMSARCH]. The SIP registration establishes a trusted relationship between the terminal and the SIP IP Core Network.
- PPG has a relationship with SIP/IP Core Network.
- User registers to the Device Management Server.
- The Device Management Server knows the address of the terminal.
- The online game service configuration is installed successfully to the user terminal.

5.3.4 Post-conditions

- The updated service configurations are configured and available to the application in the terminal, provided that the user accepts the changes.

5.3.5 Normal Flow

1. The service provider made changes to the user online game service. The online game service server places a device management change order to the Device Management Server for specific user.
2. The Device Management Server sends the notification to the Push Proxy Gateway (PPG). The notification contains information for terminal to start device management session with the Device Management server.
3. Using Push OTA-SIP, the PPG forwards the device management notification message to the terminal, via SIP/IP Core network.
4. The SIP/IP Core Network forwards the Device Management notification message to the terminal.
5. Upon receiving the notification, the terminal initiates device management session. After initiation of the device management session, the user will be presented with the option of accepting or rejecting the configuration change for the gaming service. This is no longer under the control of the push service. Upon acceptance of the configuration change, the user configuration is updated and the online game is enabled.

5.3.6 Alternative Flow

None

5.3.7 Operational and Quality of Experience Requirements

None

5.4 Use Case D: Service wishes to use a push without a receipt confirmation of delivery

	Affected Areas				
	Device	**Connectivity**	**Enabling Services**	**Applications**	**Content**
Tickmarks (X)	X		X	X	
Additional Keywords					

Table 4: Affected Areas Use Case D

5.4.1 Short Description

A low value application, such as advertising, wishes to communicate content to a user of a device. The application, acting as a push initiator, will use push, to initiate the communication of an event to the mobile customer. The user may have a device and a connectivity constraint that means that the cost and bandwidth on the 'push' channel is constrained and one-way. In this case confirmation of delivery, by a receipt notification from the mobile device, is not required.

The application wish to use a content type & size that is limited, but not constrained by, the MTU of the bearer network used to communicate to the device. In other words, the content used by the service to notify the customer may be any MIME type that is mutually understood by the client (and user agent) and the application.

The application may wish to address a specific user agent that is resident in the device; in this case, the mobile device may have multiple user-agents that may be active and the reception of a content type on the push channel may cause the activation of a device resident application; in this case, application level addressing will be used.

5.4.2 Actors

- Push Initiator, Push Proxy Gateway, User, Client.

5.4.2.1 Actor Specific Issues

- Push Initiator does not require receipt confirmation of delivery.

5.4.2.2 Actor Specific Benefits

- Legacy Push use case.

5.4.3 Pre-conditions

- The Push Initiator wishes to deliver low-value content to a user via a client in a device.
- User has a device that is constrained in terms of its support for push, in that there is reduced bandwidth and one-way connectivity.

5.4.4 Post-conditions

- Client receives push message and it is presented to user.

5.4.5 Normal Flow

1. Push Initiator submits PAP message to Push Proxy Gateway; User is identified as per the addressing scheme defined in [PushPAP]. The push initiator may wish to address a specific user agent. This should be conveyed to the device; such that at step 6 the client may direct the request appropriately.
2. The Push Proxy gateway determines the delivery mechanism based on
   a. PAP Quality of Service
   b. User identity & device capability
   c. User availability
3. Using Push OTA-SIP, the PPG forwards the content to the client without explicit delivery confirmation.
4. Push Proxy gateway indicates status of push submission to push initiator using PAP.
5. Some time later, the push message physically arrives at the device and is processed by the client.
6. Using the semantics of the content and push meta data the pushed message is presented via the client user-agent.

5.4.6 Alternative Flow

1. At step 3, the content may be segmented for delivery if the bearer or underlying delivery mechanism is constrained e.g. if SMS is used as a fallback;
2. At step 6, the content must be reconstituted by the client prior to transfer to the device user-agent.

5.4.7 Operational and Quality of Experience Requirements

None.

5.5 Use Case E: Service wishes to use a push that requires receipt confirmation of delivery

	Affected Areas				
	Device	Connectivity	Enabling Services	Applications	Content
Tickmarks (X)	X		X	X	
Additional Keywords					

Table 5: Affected Areas Use Case E

5.5.1 Short Description

A high value application, such as a stock ticker, wishes to push content to a user. The application will use a push proxy, to initiate the communication of the content to the user's Device. The content being communicated necessitates that the push proxy and mobile device may have to establish common communication context such that confirmation of successful transport of the content is confirmed and subsequently notified back to the originator of the message, the application.

The application may wish to use a content type & size that is limited, but not constrained by, the MTU of the bearer network used to communicate to the device. In other words, the content used by the service to notify the customer may be any MIME type that is mutually understood by the client (and user agent) and the application.

The application may wish to address a specific user agent that is resident in the device; in this case, the mobile device may have multiple user-agents that may be active and the reception of a content type on the push channel may cause the activation of a device resident application; in this case, application level addressing will be used

5.5.2 Actors

- Push Initiator, Push Proxy Gateway, User, Client.

5.5.2.1 Actor Specific Issues

- Push Initiator requires receipt confirmation of delivery.

5.5.2.2 Actor Specific Benefits

- Legacy Push use case.

5.5.3 Pre-conditions

- The Push Initiator wishes to deliver content to a user via a client in a device.
- User has a device that is capable to support confirmed push.

5.5.4 Post-conditions

- Client receives push message and it is presented to user
- Push initiator receives application level delivery confirmation from Push Proxy Gateway

5.5.5 Normal Flow

1. Push Initiator submits PAP message to Push Proxy Gateway; User is identified as per the addressing scheme defined in [PushPAP]. Push Initiator requests confirmed delivery and result notification via PAP Quality of Service
2. Push Proxy gateway determines mechanism for delivery based on

a) PAP Quality of Service

b) User identity & device capability

c) User availability

3. Using Push OTA-SIP, the PPG forwards the content to the client with explicit delivery confirmation
4. Client acknowledges receipt of delivery to push proxy gateway
5. Push Proxy gateway indicates status of push submission to push initiator using PAP
6. Using the semantics of the content and push meta-data the pushed message is presented via the client user-agent.

5.5.6 Alternative Flow

1. At step 3 the device may not be available, or the request to send a push message may expire leading to the following alternate scenarios:
   a. This will result in the non-delivery of the push message
   b. An intermediate 'progress – note', as defined in [PAP] may be delivered by the push proxy gateway to the push initiator
2. At step 3 the Push Proxy gateway may request a capability information for the device:
   a. This may result in the delivery of the push message not being attempted in step 4 – if the user-agent is considered not compatible with the content.
   b. The push proxy gateway will then indicate delivery status to push initiator as defined in [PAP]

5.5.7 Operational and Quality of Experience Requirements

None.

6. Requirements

6.1 High-Level System Requirements (Normative)

REQ1	Push OTA-SIP SHALL provide a unique way to identify the user, the device and the application within the device.
REQ2	The application protocol (i.e. OTA-SIP) SHALL be based on SIP [IMSARCH].
REQ3	Push OTA-SIP SHALL comply to Push Architecture [PushArch]
REQ4	Push OTA-SIP SHALL reuse existing standard protocols and data formats to the extent possible.
REQ5	It SHOULD be possible to support OTA-SIP, whilst roaming using SIP/IP Core Network roaming capability.
REQ6	Push OTA-SIP SHALL be agnostic to the underlying network.
REQ7	Push OTA-SIP MUST be capable of providing an unconfirmed delivery of content to the device via the [IMSARCH].
REQ8	Push OTA-SIP MUST be capable of providing a confirmed delivery of content to the device, via an acknowledgement of successful transport, using the [IMSARCH].
REQ9	Push OTA-SIP MUST support the transport of push meta data, including application identifiers, to the device. The push meta data is defined in the WAP Push Message Specification [PushMsg]
REQ10	Push OTA-SIP MUST support all WAP Push defined content types
REQ11	Push OTA-SIP SHOULD support all MIME content types as per [PushArch]
REQ12	Push OTA-SIP MUST provide a transport to the device via [IMSARCH] such that the size of content being transferred is comparable to other, pre-existing, push OTA mechanisms as defined in [PushOTA]
REQ13	Push OTA-SIP SHOULD support the same set of PAP operations that are currently defined in the WAP Push PAP Specification [PushPAP] and managed by the PPG [PPGService]
REQ14	Push OTA-SIP MAY provide quality of service options, exposed via PAP, to the Push Initiator
REQ15	Push OTA-SIP SHOULD provide a mapping, where possible, of pre-existing PAP Quality of Service options (as defined in [PushPAP]) to the OTA interface.
REQ16	Push OTA-SIP MUST not interfere with the content presentation semantics of pre-existing push content types.
REQ17	OTA-SIP SHALL treat the SIP/IP Core Network [IMSARCH], as a network capability providing both IP transport and other functions as described in the above paragraphs.
REQ18	OTA-SIP MUST be able to interface to the SIP/IP Core Network [IMSARCH].

Table 4: High-Level System Requirements

6.1.1 Security

REQ19	The user SHALL be able to use the OTA-SIP in an authorized and authenticated manner.
REQ20	The confidentiality and integrity of SIP-Push message SHALL be ensured.
REQ21	Security requirements SHALL be compliant with requirements stated in [PushSEC] RD

Table 5: High-Level System Requirements – Security Items

6.1.2 Charging

None

6.1.3 Administration and Configuration

None

6.1.4 Usability

None

6.1.5 Interoperability

None

6.1.6 Privacy

REQ22	Privacy requirements SHALL be compliant with requirements stated in [Privacy].

Table 6: High-Level System Requirements – Privacy Items

6.2 System Elements (Informative)

A push operation [PushArch] is accomplished by allowing a Push Initiator (PI) to transmit push content and delivery instructions to a Push Proxy Gateway (PPG), which then delivers the push content to the push client (henceforth referred to as "client" or "terminal") according to the delivery instructions.

The PI is typically an application that runs on an ordinary web server. It communicates with the PPG using the Push Access Protocol (PAP) over HTTP. The PPG [PPGservice] uses the Push Over-The-Air (OTA) Protocol to deliver the push content to the client. Figure 1 illustrates the Push Framework.

PAP [PushPAP] is based on standard Internet protocols; XML is used to express the delivery instructions, and the push content can be any MIME media type.

As mentioned, the PPG is responsible for delivering the push content to the client. In doing so it potentially may need to translate the client address provided by the PI into a format understood by the mobile network, store the content if the client is currently unavailable, etc. The PPG does more than deliver messages. For example, it may notify the PI about the final outcome of a push submission and optionally handle cancellation, replace, or client capability requests from the PI.

The Push Over-The-Air (OTA) [PushOTA] protocol is the part of the Push Framework that is responsible for transporting content from the PPG to the client and its user agents. It is designed to run on top of HTTP (OTA-HTTP), WSP (OTA-WSP), or SIP (OTA-SIP).

OTA-SIP utilizes IETF SIP protocol and extension for over-the-air communication between the PPG and the client. It is primarily to be used in conjunction with SIP/IP Core Network. Push content is delivered using the IETF SIP and SIMPLE protocols and SIP/IP Core Network.

The SIP/IP Core Network is a Session Initiation Protocol (SIP) based IP multimedia infrastructure.

6.2.1 Push Proxy Gateway

The Push Proxy Gateway (PPG) is the entity that does most of the work in the Push framework. Its responsibilities include acting as an access point for content pushes from the Internet to the mobile network, and everything associated therewith (authentication, address resolution, etc).

As the PPG is the entry point to a mobile network, it may implement network access-control policies about who is able to gain access to the network, i.e. who is able to push content and who is not, and under which circumstances, etc.

6.2.2 SIP/IP Core Network

3GPP and 3GPP2 have defined a SIP/IP Core Network as an architectural solution in 3GPP and 3GPP2 networks [IMSARCH]. As defined in [IMSARCH] the SIP/IP Core Network is a Session Initiation Protocol (SIP) based IP multimedia infrastructure that provides a complete architecture and framework for providing multimedia services. This includes security functions (e.g. authentication, authorization), routing, charging, and default codecs. Thus, it provides a platform for globally interoperable IP multimedia services - especially in the mobile environment. The SIP/IP Core Network service capabilities consist of session management, user data access, event subscription and notification, messaging, data manipulation, and conference control.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version

A.2 Draft/Candidate Version 1_0 History

Document Identifier	Date	Sections	Description
Draft Version OMA-RD_SIP_Push-V1_0	25 Jan 2004	1, 5	Added initial baseline structure and introductory text
Draft Version OMA-RD_SIP_Push-V1_1	April 26 2004		Added sections: 1, 3.3, 5.2, 5.3, and 6.1.
Draft Version OMA-RD_SIP_Push-V1_2	May 27, 2004		Input comments from Munich Meeting (April 26, 2004)
Draft Version OMA-RD_SIP_Push-V1_4	July 19, 2004		Input comments from Bangkok meeting (June 22, 2004)
Draft Version OMA-RD_SIP_Push-V1_5	September 8, 2004		Input comments from Hawaii (August 17): OMA-BAC-PUSH-2004-0029 (Only the Section 2, 3, 4 and MMS use case)
Draft Version OMA-RD_SIP_Push-V1_6	September 27, 2004		Input contributions from Orlando (September 27, 2004) OMA-BAC-PUSH-2004-0016R05 OMA-BAC-PUSH-2004-0031R1 OMA-BAC-PUSH-2004-0032R1 OMA-BAC-PUSH-2004-0033R1
Draft Version OMA-RD-SIP_Push-V1_0_0	Oct 5th 2004		Updated document in OMA – Push Permanent doc repository
Draft Version OMA-RD-SIP_Push_V1_0_1	Oct 15th 2004	References, cover, pagination, 6.2	Updated after informal review by OMA Requirements Group.
Draft Version OMA-RD-SIP_Push_V1_0_2	November 18, 2004		Updated after formal review by OMA Requirement Group. OMA-REQ-2004-1064R01-RDRR-SIPPush-V1_0_1.doc
Draft Version OMA-RD-SIP_Push_V1_1	November 29. 2004		Updated after formal review by OMA Requirement Group. OMA-REQ-2004-1064R02-RDRR-SIPPush-V1_1.doc
Candidate Version OMA-RD-SIP_Push-V1_0-20050130-C	30 Jan 2005	n/a	Status changed to Candidate by TP TP ref # OMA-TP-2005-0019R05-WID-054-SIPPush-RD-for-TP-Approval