Enabler Validation Plan for Push 2.2

Candidate Version 2.2 – 06 Nov 2007

Open Mobile Alliance

OMA-EVP-Push-V2_2-20071106-C

Contents

Figures

1. Scope

This document details the Validation plan for the Push V2.2 Enabler Release. The successful accomplishment of the validation activities will be required for the Enabler to be considered for Approved status.

The validation plan for the push Enabler Release specifications is based on testing expectations in the Enabler Test Requirements (ETR). While the specific test activities to be performed are described in the Enabler Test Specification (ETS) the test environment is described in this plan. This test environment details infrastructure, operational and participation requirements identified for the needed testing activities.

The list of specifications, defining the scope of Push 2.2, as stated in [ERELD] and [ETR] is according to the following:

- Push Architecture Wap 250- PushArchOverview
- Push Message Wap 251- Push Message
- Push PAP OMA-WAP-TS-PAP_V2_2
- Push OTA OMA-WAP-TS-PushOTA-V2-2

1.1   Assumptions

NONE

1.2   Exclusions

NONE

2. References

2.1 Normative References

[IOPPROC]	"OMA Interoperability Policy and Process", Version 1.5, Open Mobile Alliance™, OMA-ORG-IOP_Process-V1_5, URL:http://www.openmobilealliance.org/
[IOPTFG]	"OMA TestFest Participation Guidelines", Version 1.0, Open Mobile AllianceTM, OMA-IOP-TestFest-Participation-Guidelines-V1_0, 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

2.2 Informative References

[PushArch]	"WAP Push Architectural Overview", Open Mobile Alliance™, WAP-250-PushArchOverview, URL:http://www.openmobilealliance.org/
[PushMsg]	"Push Message Specification", Open Mobile Alliance™, WAP-251-PushMessage, URL: URL:http://www.openmobilealliance.org/
[PushOTA]	"Push OTA Protocol", Open Mobile Alliance™. OMA-WAP-TS-PushOTA-V2_2 URL:http//www.openmobilealliance.org/
[PushPAP]	"Push Access Protocol Specification". Open Mobile Alliance™. OMA-WAP-TS-PAP-V2_2 URL:http://www.openmobilealliance.org/
[RDF]	"Resource Description Framework (RDF) Model and Syntax Specification", W3C Recommendation, 22-February-1999. URL: http://www.w3.org/TR/REC-rdf-syntax
[RFC2246]	"The TLS Protocol Version 1.0", T. Dierks, C. Allen. January 1999, URL:http://www.ietf.org/rfc/rfc2246.txt
[RFC2396]	"Uniform Resource identifiers (URI)", T. Berners-Lee, et al., August 1998. URL: http://www.ietf.org/rfc/rfc2396.txt
[RFC822]	"Standard for the Format of ARPA Internet Text Messages", D. Crocker, August 1982. URL: http://www.ietf.org/rfc/rfc0822.txt
[WAPARCH]	"WAP Architecture". Open Mobile Alliance™. WAP-210-WAPArch. 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", are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

Application	A value-added data service provided to a Client. The application may utilise both push and pull data transfer to deliver content
Application Addressing	The ability to address a particular user agent on a WAP client.
Bearer Network	a network used to carry the messages of a transport-layer protocol between physical devices. Multiple bearer networks may be used over the life of a single push session.
Client	In the context of push, a client is a device (or service) that expects to receive push content from a server. In the context of pull a client, it is a device initiates a request to a server for content or data. See also "device"
Contact Point	Address information that describes how to reach a push proxy gateway, including transport protocol address and port of the push proxy gateway.
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.
Content Encoding	when used as a verb, content encoding indicates the act of converting a data object from one format to another. Typically the resulting format requires less physical space than the original, is easier to process or store, and/or is encrypted. When used as a noun, content encoding specifies a particular format or encoding standard or process.
Content Format	actual representation of content.
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.
End-user	See "user"
Multicast Message	a push message containing a single address which implicitly specifies more than one OTA client address.
Push Framework	The entire 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.
Registration	Refers to a procedure where the PPG becomes aware of the terminal's current capabilities and preferences.
Registration Context	A state where the PPG is aware of at least the last capabilities and preferences conveyed from

	the terminal.
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 "client".
Terminal-ID	An identifier that is used by a PPG to uniquely identify a terminal.
User	A user is a person who interacts with a user agent to view, hear, or otherwise use a rendered content. Also referred to as end-user
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

ABNF	Augmented Backus-Naur Form
ABNF	Augmented Backus-Naur Form
CPI	Capability and Preference Information
CSD	Circuit Switched Data
DNS	Domain Name Server
DTD	Document Type Definition
ETR	Enabler Test Requirements
ETS	Enabler Test Specification
GPRS	General Packet Radio Service
HTTP	Hypertext Transfer Protocol
IANA	Internet Assigned Numbers Authority
IP	Internet Protocol
MS	Mobile Station
MSISDN	Mobile Station International Subscriber Directory Number
OMA	Open Mobile Alliance
OMA	Open Mobile Alliance
OMNA	Open Mobile Naming Authority
OTA	Over The Air
OTA-HTTP	(Push) OTA over HTTP
OTA-HTTP-TLS	OTA-HTTP over TLS
OTA-WSP	(Push) OTA over WSP
PAP	Push Access Protocol
PDP	Packet Data Protocol
PI	Push Initiator
PO-TCP	PPG Originated TCP connection establishment method
PPG	Push Proxy Gateway
QoS	Quality of Service

RADIUS	Remote Authentication Dial-In User Service
RFC	Request For Comments
SHA-1	Secure Hash Algorithm 1
SI	Service Indication
SIA	Session Initiation Application
SIR	Session Initiation Request
SL	Service Loading
SMS	Short Message Service
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TO-TCP	Terminal Originated TCP connection establishment method
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WAP	Wireless Application Protocol
WBXML	WAP Binary XML
WDP	Wireless Datagram Protocol
WINA	WAP Interim Naming Authority
WSP	Wireless Session Protocol
WTLS	Wireless Transport Layer Security

4. Enabler Validation Description

This document details the Validation plan for the Push V2.2 Enabler Release. The successful accomplishment of the validation activities will be required for the Enabler to be considered for Approved status. This Plan will detail the required testing environment and tools required to implement the testing successfully.

5. TestFest Activities

5.1 Enabler Test Guidelines

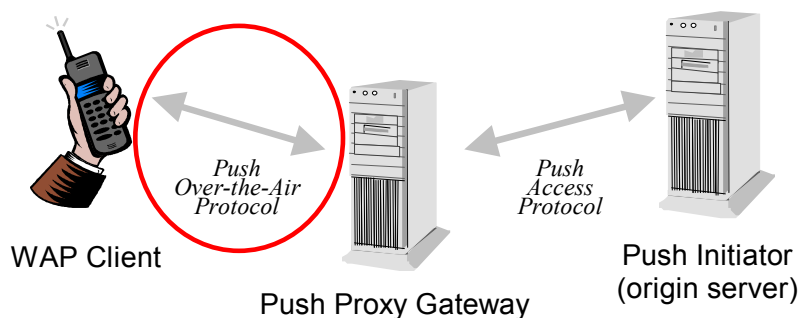A full description of Push 2.2 can be found in the ERELD and specifications.

The Push covers the following testing areas OMA Push OTA, Push SL, Push SI, Push CO, Push TO-TCP and Push PO-TCP Version 2.2.

- A Push initiator Tool is required to connect to the PPG for initiating the Push message.

- The Protocol Wap1 Push (WSP) is tested

- The Protocol Wap2 Push (HTTP) is tested

- The testing requires adequate Push Device and Server functionality.

- The testing requires adequate Push Device and Server Security Implementations.

- The proper Push capability definition as defined in the UAProfile or the Accept Headers

- Telephony Mobile network supporting the open Push ports

- OTA Device Parameter Provisioning Capability

5.1.1 Minimal Test Configuration

The minimal test configuration of Push shall include:
- A Client implementation
- A Push Proxy Server implementation
- A Push Initiator Tool
- An SMS network for Over the Air Push notification delivery
- A Mobile WAP1/WAP2 telephony network for Session based push
- A protocol Analyser to monitor and debug the Push protocols

WAP Client   Push Proxy Gateway   Push Initiator (origin server)

Push Over-the-Air Protocol

Push Access Protocol

5.1.2 Minimal Participation Guidelines

Minimum

- • 3 client implementations and

- • 1 PPG / WAP GW/Proxies are needed

5.1.3 Optimal TestFest Achievement Guidelines

The ETS Test Cases listed below represent a subset of all the Test Cases for the Enabler that it is thought can be executed in a test session at an OMA TestFest. This list is intended to facilitate maximum test coverage of the functionality of the enabler within a test session. It is not intended to be the only tests executed at a TestFest, and teams are encouraged to execute more tests if they are able to do in the time allowed.

The list includes:

Client settings shall be in accordance with the network parameters provided by the TestFest host.

- • Gateway and proxy configuration shall be in accordance with the information provided by the TestFest host for serving all clients participating in the TestFest.

- • Initiator Push tool to create the Push requests .that will be submitted to the Push Proxy Gateway used during the TestFest.

- • The SMS-C configuration shall provide the agreed interface to the gateways and SMS service to the clients participating in the TestFest.

- • Configuration of the AP and the RADIUS shall provide RADIUS ACCOUNTING to the gateways/proxies requiring it for MSISDN forwarding. Necessary configuration parameters shall be in advance documented.

5.2 Enabler Test Requirements

5.2.1 Test Infrastructure Requirements

The testing shall be performed as end-to-end testing. Most likely the client participants will be in one place, while the participant gateways and proxies will be located in member companies premises, accessible to the rest of the test fest environment. Such a "distributed" test fest environment puts effort on the test fest host and requires detailed documented configuration.

The Network Elements involved in Push Testing are:

- • PLMN (GSM/GPRS)

- • An SMS-C Supporting Binary Encoding

- • A Gateway supporting Push 2.0 or higher

- • Clients supporting Push 2.0 or higher

- • Initiator Push Tool to create the pushes submitted to the Push Proxy Gateway.

- • A Provision Server to pre provision the Push security parameters if required.

A Network Analyzer for Network Monitoring/Protocol Analyzing is also useful

5.2.2   Enabler Execution Flow

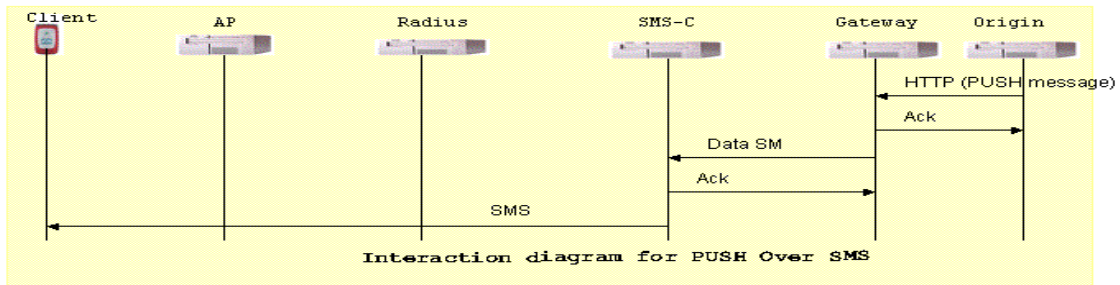**Figure 1: Interaction Diagram for Push Over the Air**

The above configuration is used for sending the Connectionless Push notifications This uses an SMSC to send the notifications over the air to the receiving device.

The following diagrams provide a high level overview of the message exchanges between the involved elements of the Push test environment based on the type of bearers, i.e. WSP or WHTTP stack.
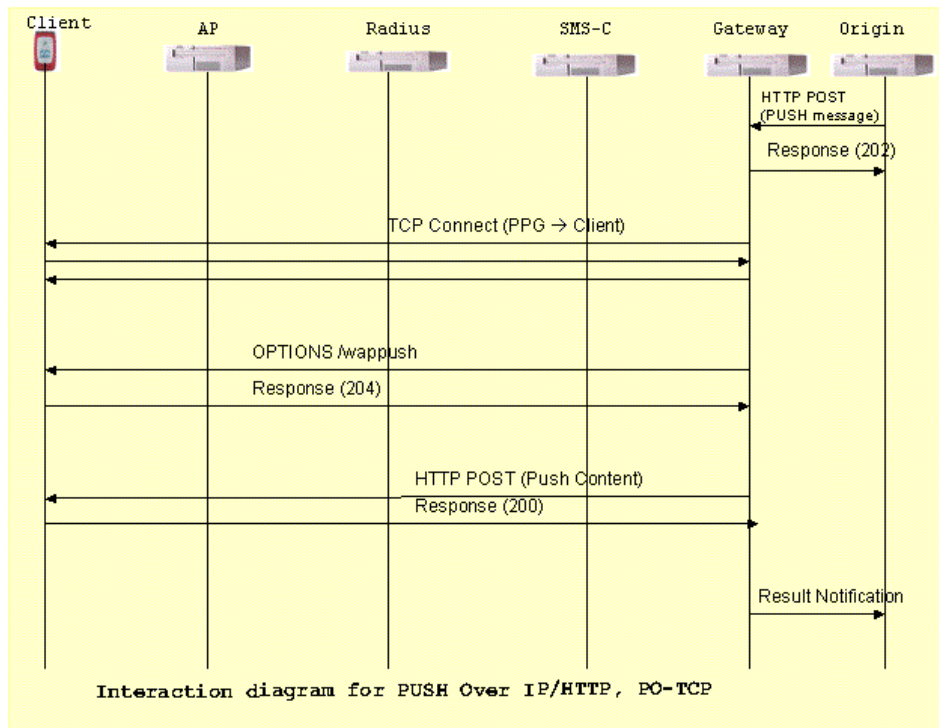
Figure2: Interaction Diagram for Push Server Originating (PO)

The above configuration is used for sending the Connection Orientated Push notifications This uses an established TCP connection between the device and the Server for sending the notifications via this channel. In this case the Server is the sender and the client is the receiver .
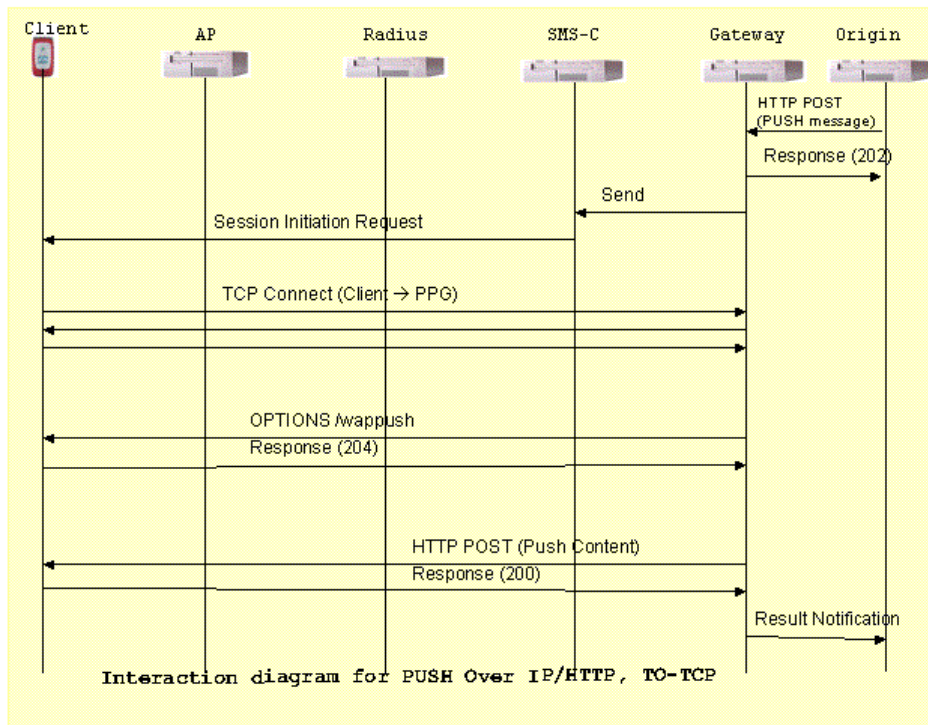
Figure 3 **Interaction Diagram for Terminal Originating Push (TO)**

The above configuration is used for sending the Connection Orientated Push notifications This uses an established TCP connection between the device and the Server for sending the notifications via this channel. In this case the Client is the sender and the Server is the receiver .

The Session Initiation Request (SIR) may be sent out to the device for a response connection to be setup.

5.2.3 Test Content Requirements

Service Load - Content is required which can be a URL of a Website

Service Indication – A message is required to be composed and then sent as Push and rendered by the device

Cache Operation – A cache operation is sent to the device with the new URL to be loaded thus overwriting the old one.

Pap Content – The Push content for the Push message is in the correct format acceptable by the Push server.

5.2.4 Test Limitations

5.2.4.1 Physical

NONE

5.2.4.2   Resources

NONE

5.2.5   Test Restrictions

NONE

5.2.6   Test Tools

5.2.6.1   Existing Tools to be Used

The test tools needed to carry out the conformance and interoperability test cases are listed as:

- SMSC
- PPG
- Push Initiator Tool
- Provisioning Server
- Mobile Device supporting Push

5.2.6.2   Test Tool Requirements

NONE

5.2.7   Resources Required

All test cases can be run in approx 8 hours . However, comparative testing may require more time, depending on the number of devices being compared.

5.3   Tests to be Performed

The following sections describe the tests related to the formal TestFest validation activities.

5.3.1   Entry Criteria for TestFest

There are no specific tests that need to be performed and passed by implementations by members wishing to participate in the TestFest.

5.3.2   Testing to be performed at TestFest

The following tests need to be performed to fully cover the range of capabilities of the enabler and defined protocols. These tests are to be covered in the TestFest.

5.4   Enabler Test Reporting

5.4.1   Problem Reporting Requirements

Normal Reporting Tool to be used, no special reporting required.

5.4.2   Enabler Test Requirements

As per the [ETR]

6. Alternative Validation Activities

NONE

7. Approval Criteria

The Push 2.2 Enabler can be put in the Approved state when:

- The Enabler has been tested successfully at 2 Test Fests or

- 3 Companies have successfully run Bi-lateral tests sessions towards a Push server and has reported results and any issues to OMA

- No open PRs exists.

Appendix A.   Change History   (Informative)

A.1   Approved Version History

Reference	Date	Description
n/a	n/a	No prior version –or- No previous version within OMA

A.2   Draft/Candidate Version 2.2 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-EVP-Push-V2_2	03 Dec 2006	All	New document, based on OMA-IOP-BRO-2007-0244
	22 Jul 2007	4, 5.1	Addition of Provisioning Server
	19 Aug 2007	All	Migrated ETG to EVP Template
	22 Oct 2007	n/a	IOP WG agreed, ref # OMA-IOP-2007-0211-INP_Push_2.2_EVP
Candidate Version OMA-EVP-Push-V2_2	06 Nov 2007	n/a	Status changed to candidate. TP R&A 2007-10-24 to 2007-11-06, TP doc. ref # OMA-TP-2007-0442-INP_EVP_Push_V2_2_for_candidate_ approval