

---

# WAP 1.2.1 PAP Push Initiator Interface Specification

---

Document Number 11457  
Revision 0.6  
Revision Date 03/25/02

This document contains trade secrets and proprietary information of AT&T Wireless Services, Inc. No use or disclosure of the information contained herein is permitted without prior written consent.

© 2002 AT&T Wireless Services, Inc.  
All rights reserved.

#### Revision History

Date	Revision	Description
12/17/01	0.1	This document is currently in Draft stage.
01/14/02	0.2	Revised sample message used in the document.
02/18/02	0.3	Added sample XML document, Sample Client Address, etc.
03/06/02	0.4	Added SIA as push content type.
03/20/02	0.5	Some minor editing.
03/25/02	0.6	Copy edited entire document, applied template styles, added table and figure captions.

**Project Sponsor: Richard Zaffino, Network Development, MMS**

For questions about this document, please contact:

**Jie McKnight**  
**Mobile Multimedia Subsidiary**  
**AT&T Wireless Services, Inc.**  
**P.O. Box 97061**  
**Redmond, WA 98073**  
**425-580-4663**  
**jie.mcknight@attws.com**

## Contents

<b>1. Introduction.....</b>	<b>5</b>
1.1. About This Document .....	5
1.1.1. Scope .....	5
1.1.2. Audience.....	5
1.2. Acknowledgements .....	5
1.3. References .....	6
1.4. Acronyms.....	6
<b>2. WAP 1.2.1 PAP Support .....</b>	<b>7</b>
2.1. Push Submission .....	8
2.1.1. Push Content Types .....	9
2.1.1.1. Service Indication (SI) .....	9
2.1.1.2. Service Loading (SL).....	9
2.1.1.3. Cache Operation (CO).....	9
2.1.1.4. Session Initiation Application (SIA).....	9
2.1.1.5. Other Push Content Types .....	10
2.1.2. Control Entity.....	12
2.1.3. Content Entity.....	15
2.1.4. Capability Entity.....	17
2.2. Result Notification .....	17
2.3. Push Cancellation .....	18
2.4. Status Query .....	19
2.5. Client Capability Query .....	20
<b>3. Configuration Parameters.....</b>	<b>21</b>
<b>Appendix A. PPG Code Samples.....</b>	<b>23</b>
A.1. AWS WAP Push Domain URL .....	23
A.1.1. Non-secure PAP .....	23
A.1.2. Secure PAP.....	23
A.2. Sample XML Document for Push Message .....	24
A.3. Sample Client Address Format .....	25
A.3.1. Using Subscriber ID.....	25

A.3.2.	Using MSISDN.....	25
A.4.	Sample HTTP Response and Error Codes.....	26
A.5.	Sample PPG Responses and Error Codes .....	27
A.5.1.	Successful Push Response.....	27
A.5.2.	Bad Address Value Response.....	27
A.5.3.	Capabilities Supported by Targeted Device Response	28
A.5.4.	Client Capability Query Failure Response.....	28
A.5.5.	Status Query Response.....	29
A.5.6.	Pending Message Response.....	29
A.5.7.	Expired Message Response.....	30

## Figures

Figure 1	Push-Message with Service Indication as the Content Type.....	11
Figure 2	Push Message Control Entity Example .....	12
Figure 3	Push Message Content Entity Example .....	15

## Tables

Table 1	Acronyms.....	6
Table 2	AWS Push Support .....	8
Table 3	PPG Phase 1 Elements and Attributes.....	13
Table 4	Result Notification Elements and Attributes .....	17
Table 5	Cancel Message Elements and Attributes .....	18
Table 6	Cancel Response Elements and Attributes .....	19
Table 7	Cancel-Result Elements and Attributes .....	19
Table 8	Status Query Message Elements and Attributes .....	19
Table 9	Status Query Response Elements and Attributes .....	20
Table 10	Status Query Result Elements and Attributes .....	20
Table 11	CCQ Message Elements and Attributes .....	20
Table 12	CCQ Response Elements and Attributes .....	21
Table 13	PPG Push Initiator Specific Configuration Parameters.....	21

## **1. Introduction**

### **1.1. About This Document**

This document describes the WAP Push Initiator interface for Phase 1 of the AT&T Wireless WAP Services Platform over the 2.5G Network — the GPRS Network. The content of this document is subject to change without advanced notice.

#### **1.1.1. Scope**

The WAP Push Initiator interface for the Phase 1 of the AT&T Wireless WAP Services Platform over the 2.5G Network is based on the WAP 1.2 Push Access Protocol (PAP) specification. This document describes which PAP operations are implemented in Phase 1 as well as providing sample solution-specific details needed by a Push Initiator.

#### **1.1.2. Audience**

This document is intended for developers who are participating in the AT&T Wireless Services Developer's Program. Users of this document should have basic knowledge of the GSM/GPRS network, the Internet, Hyper Text Transfer Protocol (HTTP), Wireless Application Protocol (WAP), and Push Access Protocol (PAP).

### **1.2. Acknowledgements**

The following individuals made significant contribution to this document.

Bryan Sullivan

Frank Herrgoss

### 1.3. References

- [R 1] *WAP Push Architectural Overview*, version 08\_Nov\_1999  
(<http://www.wapform.org.what/technical.htm>)
- [R 2] *WAP Push Access Protocol version WAP-164\_PAP*  
(<http://www.wapform.org.what/technical.htm>)
- [R 3] *WAP PPG Service version WAP-151\_PPGService*  
(<http://www.wapform.org.what/technical.htm>)
- [R 4] *WAP Push Message version WAP-145\_PushMessage*  
(<http://www.wapform.org.what/technical.htm>)
- [R 5] *WAP Push OTA version WAP-189\_PushOTA*  
(<http://www.wapform.org.what/technical.htm>)
- [R 6] *WAP Service Indication version WAP-167\_ServiceInd*  
(<http://www.wapform.org.what/technical.htm>)
- [R 7] *WAP Service Loading version WAP-168\_ServiceLoad*  
(<http://www.wapform.org.what/technical.htm>)
- [R 8] *Mobile Access Gateway Push System*, Openwave Release 5.0.1,  
September 2001

### 1.4. Acronyms

Table 1 identifies acronyms used in this document.

**Table 1 Acronyms**

Term	Definition
AWS	AT&T Wireless Services
CCQ	Client Capability Query
CO	Cache Operation
DTD	Document Type Definition
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications

Term	Definition
HTTP	HyperText Transfer Protocol
MSISDN	Mobile Subscriber Identification Number
OS	Operating System
PAP	Push Access Protocol
PDU	Protocol Data Unit
PI	Push Initiator
PPG	Push Proxy Gateway
SI	Service Indication
SIA	Session Initiation Application
SIR	Session Initiation Request
SL	Service Load
UAProf	User Agent Profile
URL	Uniform Resource Locator
WAP	Wireless Access Protocol
WML	Wireless Markup Language
XML	Extensible Markup language

## 2. WAP 1.2.1 PAP Support

The Push Access Protocol (PAP) provides a means of pushing content from the Internet to a mobile device, such as a WAP-enable mobile phone. PAP achieves this by managing the communication between the Internet server that does the push of the content, known in the context of this document as a Push Initiator (PI), and the Push Proxy Gateway (PPG), the server that pushes the content on to the target mobile device. The PPG is an intermediary that connects the external IP network and the AT&T Wireless Services (AWS) wireless network.

PAP makes possible the following operations:

- Push Submission
- Result Notification
- Push Cancellation
- Status Query

- Client Capability Query

Table 2 outlines PAP operations that are supported in Phase 1 of the AT&T Wireless WAP Services Platform over the 2.5G Network.

**Table 2 AWS Push Support**

PAP Message	Direction	Supported
Push-message	Push Initiator $\Rightarrow$ PPG	yes
Push-response	PPG $\Rightarrow$ Push Initiator	yes
cancel-message	Push Initiator $\Rightarrow$ PPG	yes
cancel-response	PPG $\Rightarrow$ Push Initiator	yes
resultnotification-message	PPG $\Rightarrow$ Push Initiator	yes
resultnotification-response	Push Initiator $\Rightarrow$ PPG	yes
statusquery-message	Push Initiator $\Rightarrow$ PPG	yes
statusquery-response	PPG $\Rightarrow$ Push Initiator	yes
ccq-message	Push Initiator $\Rightarrow$ PPG	yes
ccq-response	PPG $\Rightarrow$ Push Initiator	yes
badmessage-response	PPG $\Rightarrow$ Push Initiator	yes

## 2.1. Push Submission

From the PAP Specification: “The purpose of the Push Submission is to deliver a push-message from a Push Initiator to a PPG, which should then deliver the push-message to a user agent in a device on the wireless network.”

Because the Push Submission contains address, content, and optional capabilities entities, the format of the push message is in the form of a multipart/related document. For the PPG in Phase 1 of the AT&T Wireless WAP Services Platform over the 2.5G Network, the multipart/related document, known in the context of this document as the push-message is transported via HTTP/HTTPs Post to the PPG.



### 2.1.1. Push Content Types

The sections that follow include WAP-specific push content types that can currently be delivered using PAP.

#### 2.1.1.1. Service Indication (SI)

This content type consists of asynchronous notifications about news headlines, change in selected stock prices, advertisements, reminders, and so on. An SI, at its most basic, contains a brief message and a URL specifying a service. The wireless client can either start the service immediately or store it for later action.

#### 2.1.1.2. Service Loading (SL)

This content type allows a user agent on a client device to load and execute a service, specified by a URL without user intervention. For example, an SL can notify a mobile device of a low prepaid service balance and require user action by loading a WML deck that presents a variety of options.

#### 2.1.1.3. Cache Operation (CO)

This content type makes it possible to invalidate content objects in the wireless client's cache. All invalidated content objects must be reloaded from the server on which they originated the next time they are accessed. Use CO if an application cannot predict when content that it creates will expire. For instance, use a CO to ensure that a mobile device always loads the most current contents from a mailbox application. A CO is an XML document that consists of a URL and one of the following operation types:

1. **Invalidate object:** Invalidates the specific object that the URL indicates.
2. **Invalidate service:** Invalidates all objects that share the same URL prefix.

#### 2.1.1.4. Session Initiation Application (SIA)

Since content push is asynchronous by nature, it is possible that no push session exists, that no active TCP connection has been established, or that the desired bearer is not utilized when content is about to be pushed from

the PPG to the mobile client. The PPG on AWS 2.5G Network provides a push framework that will circumvent the limitation.

When the PPG gets a request from the PI for a confirmed push to a client, if it cannot deliver the push content because there are no active sessions available to the client, the PPG sends a Session Initiation Request (SIR) via a Session Initiation Application (SIA) content push to the client. If the client supports SIA, it accepts the SIR from the PPG and establishes push session or an active TCP connection, using a specific bearer.

The content type, *application/vnd.wap.sia*, is defined and encoded based on a set of protocol data units (PDUs). For detailed the information about this particular content definition and encoding schema, please refer to WAP-235-PushOTA-20010425-a of the Wireless Application Protocol.

#### **2.1.1.5. Other Push Content Types**

Besides the content types that are defined in WAP, the push content type can be any content as long as it is described as MIME type.

To illustrate the PAP interface provided by AWS PPG, Figure 1 shows an example of a push-message with Service Indication as the content type.

From the example shown in Figure 1, the push-message contains a control entity, a content entity, and a capability entity. For the PPG in Phase 1 of the AT&T Wireless WAP Services Platform over the 2.5G Network, the capabilities entity is optional.

*Figure 1 Push-Message with Service Indication as the Content Type*

```

POST /pap HTTP/1.1
Content-Length: 1513
Content-Type: multipart/related; boundary=asdlfkjiurwghasf; type="application/xml"

--asdlfkjiurwghasf
Content-Type: application/xml; charset=UTF-8

<?xml version="1.0"?>
<!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
    "http://www.wapforum.org/DTD/pap_1.0.dtd">
<pap product-name="My Push Initiator">
  <push-message
    push-id="9fjeo39jf084@PIDomain.com "
    deliver-before-timestamp="2001-04-30T10:45:00Z"
    deliver-after-timestamp="2001-04-30T06:45:00Z"
    source-reference="IM Daemon"
    ppg-notify-requested-to="http://im.PIDomain.com/pap/results/.cgi"
    progress-notes-requested="true">
    <address address-
value="WAPPUSH=10125_mag.nwest.attws.com/TYPE=USER@pn.attwireless.net"/>
    <quality-of-service priority="medium" delivery-method="unconfirmed"/>
  </push-message>
</pap>

--asdlfkjiurwghasf
Content-Type: text/vnd.wap.si; charset=UTF-8

<?xml version="1.0"?>
<!DOCTYPE si PUBLIC "-//WAPFORUM//DTD SI 1.0//EN"
    "http://www.wapforum.org/DTD/si.dtd">

<si>
  <indication href =" http://www.PIDomain.com/im?SubID=john.doe_subID "
    action="signal- Medium">
    Welcome to PIDOMAIN. Your Buddy A just send you an instant message
  </indication>
</si>

--asdlfkjiurwghasf

```

### 2.1.2. Control Entity

The first type of push-message entity is the control entity. The control entity identifies the target mobile terminal and contains delivery instructions such as time delivery restrictions.

**Figure 2 Push Message Control Entity Example**

```
POST /pap HTTP/1.1
Content-Length: 1513
Content-Type: multipart/related; boundary=asdlfkjiurwghasf;
type="application/xml"

--asdlfkjiurwghasf
Content-Type: application/xml; charset=UTF-8

<?xml version="1.0"?>
<!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
    "http://www.wapforum.org/DTD/pap_1.0.dtd">
<pap product-name="My Push Initiator">
<push-message
    push-id="9fjeo39jf084@PIDomain.com "
    deliver-before-timestamp="2001-04-30T10:45:00Z"
    deliver-after-timestamp="2001-04-30T06:45:00Z"
    source-reference="IM Daemon"
    ppg-notify-requested-to="http://im.PIDomain.com/pap/results/.cgi"
    progress-notes-requested="true">

    <address address-
value="WAPPUSH=10125_mag.nwest.attws.com/TYPE=USER@pn.attwireless.net"/>
    <quality-of-service priority="medium" delivery-method="unconfirmed"/>
    </push-message>
</pap>

--asdlfkjiurwghasf
Content-Type: text/vnd.wap.si; charset=UTF-8

<?xml version="1.0"?>
<!DOCTYPE si PUBLIC "-//WAPFORUM//DTD SI 1.0//EN"
    "http://www.wapforum.org/DTD/si.dtd">

<si>
    <indication href =" http://www.PIDomain.com/im?SubID=john.doe_subID"
        action="signal- Medium">

        Welcome to PIDOMAIN. Your Buddy A just send you an instant message
    </indication>
</si>

--asdlfkjiurwghasf
```

Table 3 outlines the elements and attributes that are supported by the PPG in Phase 1.

**Table 3 PPG Phase 1 Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
push-message	yes
push-id	yes This must be in the following format: unique_pushid@Pushinitiator_Domain
deliver-before-timestamp	yes
deliver-after-timestamp	yes
source-reference	yes
ppg-notify-requested-to	yes
progress-notes-requested	no
address	yes
address-value	yes
quality-of-service	yes

The PPG validates the control entity using an XML validator before returning the push response to the Push Initiator. There are four major data items that the AWS PPG uses from the control entity - the push-id, the ppg-notify-requested-to value, the address-value and source-reference:

1. The push-id is necessary to determine which push initiator is sending the push message and for tracking the progress of the request. The push-id must be of the format “unique\_pushid@Pushinitiator\_Domain.” As a syntax requirement by AWS, The Pushinitiator\_Domain must indicate which Push Initiator sent the push-message. This value must be agreed upon with AT&T Wireless Services network as it is used to configure Push-Initiator-specific parameters. An example would be “company.com.” Further, the “unique\_pushid” portion of the push-id

*must* be a globally unique number. The following is an example of a globally unique push-id:

<running number>.<timestamp>.<host-ip>@< Pushinitiator\_Domain >

2. The ppg-notify-requested-to value is used to determine whether or not to send a resultnotification-message and the address to send it to. If the ppg-notify-requested-to attribute is present, the PPG will use its attribute value as the address to which the resultnotification-message will be sent.
3. If the Push Initiator application does not implement ppg-notify-requested-to, it is highly recommended that the application should at least read the HTTP response from PPG. In this way, the application can determine an error occurring at the HTTP level to help troubleshooting during integration of the application to the AT&T Wireless Services network.
4. The [WAPPUSH=john.doe\\_subID/TYPE=USER@pn.attwireless.net](#) of the address-value (in the <address> tag) is used to determine which mobile user should receive the instant message.
5. It is highly recommended that the Push Initiator application should provide a source-reference as one of the push-message attributes. Source-reference will provide a means for AT&T Wireless Services operators to easily query all push messages that are related to the particular Push Initiator.

The push-id is an attribute of the <push message> tag. In the example above, the push-id is “9fjeo39jf084@PIdomain.com.” The WAPPUSH is found in the address-value attribute of the <address> tag. In the example, it is 10125\_mag.nwest.attws.com/TYPE=USER@pn.attwireless.net. If these two items are not correct, a 2000 (bad request) error code is returned in the push response.

Finally, in the example, the ppg-notify-requested-to attribute value is present; therefore, a resultnotification-message shall be sent. The ppg-notify-requested-to value must represent an active web server address which is ready to receive HTTP POST messages from the PPG.

### 2.1.3. Content Entity

The second entity type is the content entity. The content entity contains content destined for the target mobile terminal and is only used for push submissions.

*Figure 3 Push Message Content Entity Example*

```
POST /pap HTTP/1.1
Content-Length: 1513
Content-Type: multipart/related; boundary=asdlfkjiurwghasf;
type="application/xml"

--asdlfkjiurwghasf
Content-Type: application/xml; charset=UTF-8

<?xml version="1.0"?>
<!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
    "http://www.wapforum.org/DTD/pap_1.0.dtd">
<pap product-name="My Push Initiator">
<push-message
    push-id="9fjeo39jf084@PIDomain.com "
    deliver-before-timestamp="2001-04-30T10:45:00Z"
    deliver-after-timestamp="2001-04-30T06:45:00Z"
    source-reference="IM Daemon"
    ppg-notify-requested-to="http://im.PIDomain.com/pap/results/.cgi"
    progress-notes-requested="true">

    <address address-
value="WAPPUSH=10125_mag.nwest.attws.com/TYPE=USER@pn.attwireless.net"/>
    <quality-of-service priority="medium" delivery-method="unconfirmed"/>
    </push-message>
</pap>

--asdlfkjiurwghasf
Content-Type: text/vnd.wap.si; charset=UTF-8

<?xml version="1.0"?>
<!DOCTYPE si PUBLIC "-//WAPFORUM//DTD SI 1.0//EN"
    "http://www.wapforum.org/DTD/si.dtd">

<si>
    <indication href =" http://www.PIDomain.com/im?SubID=john.doe_subID"
        action="signal- Medium">

        Welcome to PIDOMAIN. Your Buddy A just send you an instant message
    </indication>
</si>

--asdlfkjiurwghasf
```

In this example, the push initiator must send the content in an XML document that conforms to the SI DTD that can be referenced at: <http://www.wapforum.org/DTD/>. The PPG will validate the content entity using an XML validator. If the content entity is invalid, the PPG will send a push-response with the appropriate error code. See Section A.5 for push-response samples.



#### 2.1.4. Capability Entity

For the PPG in Phase 1 of the AT&T Wireless WAP Services Platform over the 2.5G Network, the capabilities entity is optional. The capability entity contains the client capabilities for which the message is formatted. It replies on the WAP User Agent Profile feature. This entity is only used in push submissions.

If the push initiator includes a client capabilities entity in the push submissions, the PPG will compare the following properties in the handset's UAProf:

- Browser Name
- Browser Version
- OS Name
- OS Version

If the push initiator specifies a capability different from that indicated by the handset's UAProf, the PPG rejects the push.

#### 2.2. Result Notification

If the ppg-notify-requested-to attribute is present in the control entity of the push-message, the PPG must send a resultnotification-message to the Push Initiator to indicate success or failure of the push message request. The PPG implements the resultnotification-message elements and attributes as shown in Table 4.

**Table 4 Result Notification Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
resultnotification-message	yes
push-id	yes
sender-address	yes
sender-name	yes
received-time	yes
event-time	yes

ELEMENT/ATTRIBUTE	Support
message-state	yes
code	yes
desc	yes
address	yes
address-value	yes
quality-of-service	yes

According to the WAP 1.2 PAP, upon receiving the resultnotification-message, the Push Initiator should return a resultnotification-response indicating either of two codes. (See Section 4 Appendix for Sample push-response).

- 1001-OK
- 2000-Bad Request

### 2.3. Push Cancellation

From the PAP Specification: “The purpose of the Push Cancellation is to allow the Push Initiator to attempt to cancel a previously submitted push message.” The push initiator initiates this operation and the PPG responds with an indication of whether the request was successful or not.

The cancel message (cancel-message) and the result message (result-message) are XML documents. If a message for which a result notification was requested is cancelled, the result notification **MUST** be sent and *must* report a message-state of “cancelled.”

The PPG implements the cancel-message elements and attributes as shown in Table 5.

**Table 5 Cancel Message Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
cancel-message	yes
push-id	yes

The PPG implements the cancel-response elements and attributes as shown in Table 6.

**Table 6 Cancel Response Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
cancel-response	yes
push-id	yes
cancel-result	yes
code	yes
desc	Yes

The PPG implements the cancel-result elements and attributes as shown in Table 7.

**Table 7 Cancel-Result Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
cancel-result	yes
code	yes
desc	yes

## 2.4. Status Query

The status query operation allows the push initiator to request the current status of a message that has been previously submitted.

The PPG implements the statusquery-message elements and attributes as shown in Table 8.

**Table 8 Status Query Message Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
statusquery-message	yes
push-id	yes

The PPG implements the statusquery-response elements and attributes as shown in Table 9.

**Table 9 Status Query Response Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
statusquery-response	yes
push-id	yes
statusquery-result	yes
event-time	yes
message-state	yes
code	yes
desc	Yes

The PPG implements the statusquery-result elements and attributes as shown in Table 10.

**Table 10 Status Query Result Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
statusquery-result	yes
event-time	yes
message-state	yes
code	yes
desc	yes

## 2.5. Client Capability Query

The Client Capability Query operation allows the push initiator to query the PPG for the capabilities of a specific device.

The PPG implements the ccq-message elements and attributes as shown in Table 11.

**Table 11 CCQ Message Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
ccq-message	yes
query-id	yes
app-id	yes

The PPG implements the ccq-response elements and attributes as shown in Table 12.

**Table 12 CCQ Response Elements and Attributes**

ELEMENT/ATTRIBUTE	Support
ccq-response	yes
query-id	yes
code	yes
desc	yes

### 3. Configuration Parameters

There are a few configuration parameters stored on AWS PPG that are specific to the push initiator applications in the Internet domain. These configuration parameters can be over-written by the equivalent parameter values set by push initiators if the application developer chooses to do so.

The following table lists the push-initiator-specific configuration parameters that are set in the PPG:

**Table 13 PPG Push Initiator Specific Configuration Parameters**

Configuration Parameter	Format	Description
MaxDeliverAfterTime	Integer Default: 604800 This parameter is optional.	Maximum length of deliver-after period if specified in push message (in seconds).
MaxPendingPushQSize	Integer Example: 10 This parameter is optional.	Maximum size of the PPG push pending queues.
MaxPushContentLen	Integer Example: 1024 This parameter is optional.	Maximum size of push content (bytes).

Configuration Parameter	Format	Description
MinDeliverBeforeTime	Integer Example: 1800 This parameter is optional.	Minimum length of deliver-before period if specified in push message (in seconds).
MinDeliveryPeriod	Integer Default: 1800 This parameter is optional.	Minimum delivery period for a push message (in seconds).
PendingPushExpiry	Integer Default: 259200 This parameter is optional.	Maximum period that a push can remain pending before it expires (in seconds).

## **Appendix A. PPG Code Samples**

### **A.1. AWS WAP Push Domain URL**

#### **A.1.1. Non-secure PAP**

The following link is for the non-secure PAP:

<http://pn.attwireless.net:9002/pap>

#### **A.1.2. Secure PAP**

The following link is for the secure PAP:

<http://pn.attwireless.net:9003/pap>

## A.2. Sample XML Document for Push Message

```
POST /pap HTTP/1.0
Content-Length: 815
Content-Type: multipart/related; boundary=asdlfkjiurwghasf;
type="application/xml"

--asdlfkjiurwghasf
Content-Type: application/xml

<?xml version="1.0"?>
<!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
"http://www.wapforum.org/DTD/pap_1.0.dtd">
<pap><push-message push-id="23@xyz.com">
<address address-value="WAPPUSH=1012516088-
28_mag.nwest.attws.com/TYPE=USER@pn.attwireless.net"/>
<quality-of-service priority="high" delivery-method="unconfirmed"
bearer="SMS" bearer-required="true" network="GSM" network-
required="true"/>
</push-message></pap>

--asdlfkjiurwghasf
Content-Type: text/vnd.wap.si

<?xml version="1.0"?>
<!DOCTYPE si PUBLIC "-//WAPFORUM//DTD SI 1.0//EN"
"http://www.wapforum.org/DTD/si.dtd">
<si><indication
href="http://10.111.111.25:80/?Cmd=List&R=99999&service=attws"
action="signal-high">You've got new instant message.</indication></si>

--asdlfkjiurwghasf
```



### **A.3. Sample Client Address Format**

#### **A.3.1. Using Subscriber ID**

address-value="WAPPUSH=1012516088  
28\_mag.nwest.attws.com/TYPE=USER@ppg.attwireless.com"

#### **A.3.2. Using MSISDN**

address-value=[WAPPUSH=+12066662901/TYPE=PLMN@ppg.attwireless.com](#)

## A.4. Sample HTTP Response and Error Codes

```
<response-result code="1001" desc="The request has been  
accepted for processing"/>
```

```
<response-result code="2000" desc="Bad Request. Not  
understood due to malformed syntax. Error 0x10, Line 5 : PAP Syntax  
Error, Required attribute(s) missing from element or invalid attributes  
specified."/>
```

```
<response-result code="2000" desc="Bad Request. Not  
understood due to malformed syntax. Error 0x12, Line 7 : PAP Syntax  
Error, Attribute value for element is invalid."/>
```

```
<response-result code="2000" desc="Bad Request. Not  
understood due to malformed syntax. Error 0x13, Line 6 : PAP Syntax  
Error, Suspected syntax error in format of attribute value."/>
```

```
<response-result code="2000" desc="Bad Request. Not  
understood due to malformed syntax. Error 0x5 : HTTP Syntax Error,  
HTTP/MIME Content-type header value not valid."/>
```

```
<response-result code="2000" desc="Bad Request. Not  
understood due to malformed syntax. Error 0x9, Line 4 : PAP Syntax  
Error, Incorrect element type."/>
```

```
<response-result code="2000" desc="Bad Request. Not  
understood due to malformed syntax. Error 0xC, Line 5 : PAP Syntax  
Error, Content expected for parent element."/>
```

```
<response-result code="2000" desc="Bad Request. Not  
understood due to malformed syntax. Error 0xE, Line 4 : PAP Syntax  
Error, Invalid child element within the parent element."/>
```

```
<response-result code="2002" desc="The client specified was not  
recognised."/>
```

```
<response-result code="2007" desc="The push-id supplied is not  
unique within the PPG."/>
```

## A.5. Sample PPG Responses and Error Codes

The response from the PPG is actually an XML document. The following are sample PPG responses that are generated using the Openwave Push Initiator Simulator.

### A.5.1. Successful Push Response

The following is an example of a successful push response:

```
<?xml version="1.0"?>
  <!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
    "http://www.wapforum.org/DTD/pap_1.0.dtd">
  <pap>
    <resultnotification-message push-id="Test 13" sender-
address="mag.nwest.attws.com" sender-name="Openwave Push Proxy Gateway"
received-time="
2002-03-05T18:51:33Z" event-time="2002-03-05T18:51:34Z" message-
state="delivered" code="1000" desc="The request succeeded.">
      <address address-value="WAPPUSH=214-
17341_mag.nwest.attws.com/type=USER@pn.attwireless.net"/>
      <quality-of-service priority="medium" delivery-
method="confirmed" network="GSM" network-required="true" bearer="GPRS"
bearer-required="true"/>
    </resultnotification-message>
  </pap>
```

### A.5.2. Bad Address Value Response

The following is an example of a response with a bad address-value in a push message:

```
<?xml version="1.0"?>
<!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
"http://www.wapforum.org/DTD/pap_1.0.dtd">

<pap product-name="Openwave Push Proxy Gateway">
  <push-response push-id="Test 1" sender-address="141.204.142.36"
sender-name="mag.nwest.attws.com" reply-time="2002-03-05T18:14:45Z">
    <response-result code="2002" desc="The client specified was
not recognised."/>
  </push-response>
</pap>
```

### A.5.3. Capabilities Supported by Targeted Device Response

The following is an example of a response when capabilities or quality of service is supported by the targeted device:

```
<?xml version="1.0"?>
<!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
"http://www.wapforum.org/DTD/pap_1.0.dtd">

<pap product-name="Openwave Push Proxy Gateway">
  <push-response push-id="Test 5" sender-address="141.204.142.36"
sender-name="mag.nwest.attws.com" reply-time="2002-03-05T18:21:30Z">
    <response-result code="3006" desc="The PPG was unable to
perform a transformation on the message."/>
  </push-response>
</pap>
```

### A.5.4. Client Capability Query Failure Response

The following is an example of a response when client capability query (CCQ) fails:

```
<?xml version="1.0"?>
<!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
"http://www.wapforum.org/DTD/pap_1.0.dtd">

<pap product-name="Openwave Push Proxy Gateway">
  <ccq-response code="3008" desc="Capabilities Not Available.
Client capabilities for the specified client are not available. Error
0x1F : PAP Operation
Error, Either no profile is explicitly defined or no URL to profile for
specified WAP address." >
    <address address-value="WAPPUSH=1011637127-
22_mag.nwest.attws.com/TYPE=USER@pn.attwireless.net"/>
  </ccq-response>
</pap>
```

### A.5.5. Status Query Response

The following is an example of a response to a Status Query:

```
<?xml version="1.0"?>
  <!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
    "http://www.wapforum.org/DTD/pap_1.0.dtd">

  <pap product-name="Openwave Push Proxy Gateway">
    <statusquery-response push-id="Test 13">
      <statusquery-result event-time="2002-03-05T18:51:34Z"
message-state="delivered" code="1000" desc="OK. The request
succeeded.">
        <address address-value="WAPPUSH=214-
17341_mag.nwest.attws.com/type=USER@pn.attwireless.net"/>
        <quality-of-service priority="medium" delivery-
method="confirmed" bearer="GPRS" bearer-required="true" network="GSM"
network-required="true"/>
      </statusquery-result>
    </statusquery-response>
  </pap>
```

### A.5.6. Pending Message Response

The following is an example of a response to a pending message:

```
<?xml version="1.0"?>
  <!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
    "http://www.wapforum.org/DTD/pap_1.0.dtd">

  <pap product-name="Openwave Push Proxy Gateway">
    <statusquery-response push-id="Test 15">
      <statusquery-result event-time="2002-03-05T19:01:20Z"
message-state="pending" code="1000" desc="OK. The request succeeded.">
        <address address-value="WAPPUSH=214-
17341_mag.nwest.attws.com/type=USER@pn.attwireless.net"/>
        <quality-of-service priority="medium" delivery-
method="confirmed" bearer="GPRS" bearer-required="true" network="GSM"
network-required="true"/>
      </statusquery-result>
    </statusquery-response>
  </pap>
```

### A.5.7. Expired Message Response

The following is an example of a response to an expired message:

```
<?xml version="1.0"?>
  <!DOCTYPE pap PUBLIC "-//WAPFORUM//DTD PAP 1.0//EN"
    "http://www.wapforum.org/DTD/pap_1.0.dtd">

  <pap product-name="Openwave Push Proxy Gateway">
    <statusquery-response push-id="Test 15">
      <statusquery-result event-time="2002-03-05T19:00:29Z"
message-state="unknown" code="3003" desc="Not Possible. Action not
possible because message is no longer available.">
        <address address-value="WAPPUSH=214-
17341_mag.nwest.attws.com/type=USER@pn.attwireless.net"/>
      </statusquery-result>
    </statusquery-response>
  </pap>
```