OMA Push Management Object

Candidate Version 1.1 – 16 Mar 2010

Open Mobile Alliance

OMA-TS- Push_MO-V1_1-20100316-C

Contents

Figures

1. Scope

This document defines the OMA Push Management Object that manages the push whitelist.

2. References

2.1 Normative References

[PUSH22] "OMA Push Over The Air, Version 2.2 ". Open Mobile Alliance™ OMA-TS-PushOTA-V2_2. URL:http://www.openmobilealliance.org

[PUSH23] "OMA Push Over The Air, Version 2.3 ". Open Mobile Alliance™ OMA-TS-PushOTA-V2_2. URL:http://www.openmobilealliance.org

[DMSTDOBJ] "OMA Device Management Standardized Objects, Version 1.2". Open Mobile Alliance™ OMA-TS-DM-StdObj-V1_2. URL:http://www.openmobilealliance.org

[DM-TND-V1-2] "OMA Device Management Tree and Description, Version 1.2". Open Mobile Alliance™ OMA-TS-DM_TND-V1_2 URL:http://www.openmobilealliance.org

[DMBOOT] "OMA Device Management Bootstrap, Version 1.2". Open Mobile Alliance™. OMA-TS-DM_Bootstrap-V1_2. URL:http://www.openmobilealliance.org

[DMTNDS] "OMA Device Management Tree and Description Serialization Specification, Version 1.2". Open Mobile Alliance. OMA-TS-DM_TNDS-V1_2. URL:http://www.openmobilealliance.org

[RFC1918] Address Allocation for Private Internets http://www.rfc-editor.org/rfc/rfc1918.txt

[RFC791] RFC 791, Internet Protocol, DARPA, 1981, URL:http://www.ietf.org/rfc/rfc791.txt

[MediaType] RFC 2046 Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types, The Internet Society, 1996, URL: http://tools.ietf.org/rfc/rfc2046.txt

[RFC2141] RFC 2141 URN Syntax, The Internet Society, 1997, URL: http://tools.ietf.org/rfc/rfc2141.txt

[RFC3513] RFC 3513, Internet Protocol Version 6 (IPv6) Addressing Architecture, §§2.2, 2.3 The Internet Society, 2003, URL:http://www.ietf.org/rfc/rfc3513.txt

[RFC2373] IP Version 6 Addressing Architecture

http://www.ietf.org/rfc/rfc2373.txt

[RFC3986] URI Generic Syntax

http://rfc.net/rfc3986.html

[GENFORM] "WAP General Formats Document", WAP Forum_, WAP-188-WAPGenFormats, URL:

http://www.openmobilealliance.org/

2.2 Informative References

[OMNA] "Open Mobile Naming Authority", URL: http://www.openmobilealliance.org/Tech/OMNA.aspx

[PUSHMO-DDF] "Push Management Object Device Description Framework", URL: http://www.openmobilealliance.org/Tech/omna/omna-dm_mo-registry.aspx

[PushAppId] "OMNA PUSH Application ID", URL: http://www.openmobilealliance.org/Tech/omna/omna-push-app-id.aspx

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

See the DM Tree and Description [DM-TND-V1-2] document for definitions of terms related to the management tree.

3.3 Abbreviations

CBS	Cell Broadcast Service
OMA	Open Mobile Alliance
OMNA	Open Mobile Naming Authority
MBMS	Multimedia Broadcast/Multicast Service
MIME	Multipurpose Internet Mail Extensions
MO	Management Object
SMSC	Short Message Service Centre
PPG	Push Proxy Gateway
URN	Uniform Resource Name

4. Introduction

DM group has defined Management Objects (MO) where parameters can be easily managed and used by applications. This document describes the OMA Push MO syntax that provides the ability to manage push whitelists, which provide filtering capability for incoming push messages.

4.1 Version 1.0

The Push MO v1.0 addresses the push security requirements of the OMA Push v2.2 enabler release [PUSH22], including ability to define the allowed source addresses of:

- Push Proxy Gateways (PPG)
- SMS sources, e.g. SMSC

4.2 Version 1.1

The Push MO v1.1 adds the push security requirements of the OMA Push v2.3 enabler release [PUSH23], including ability to define the allowed source addresses, push applications (by Push Application ID), and content types from the following source types:

- PPG as in [PUSH22], with enhancements
    - definition of allowed push applications and content types
- SMS as in [PUSH22], with enhancements
    - definition of allowed push applications and content types
- Cell Broadcast Service (CBS)
- MBMS
- OMA BCAST

5. Standardized Management Objects

5.1 Introduction to Management Objects (Informative)

Management objects are the entities that can be manipulated by management actions carried over the OMA DM protocol. A Management Object can be as small as an integer or large and complex like a background picture, screen saver, or security certificate. The OMA DM protocol is neutral about the contents, or values, of the Management Objects and treats the node values as opaque data.

5.1.1 Definition and description of Management Objects

OMA DM Management Objects are defined using the OMA DM Device Description Framework [DMTNDS], or DDF. The use of this description framework produces detailed information about the device in question. However, due to the high level of detail in these descriptions, they are sometimes hard for humans to digest and it can be a time consuming task to get an overview of a particular object's structure.

In order to make it easier to quickly get an overview of how a Management Object is organized and its intended use, a simplified graphical notation in the shape of a block diagram is used in this document. Even though the notation is graphical, it still uses some printable characters, e.g. to denote the number of occurrences of a node. These are mainly borrowed from the syntax of DTDs for XML. The characters and their meaning are defined in the following table.

Character	Meaning
+	one or many occurrences
*	zero or more occurrences
?	zero or one occurrences

If none of these characters is used the default occurrence is exactly once.

There is one more feature of the DDF that needs to have a corresponding graphical notation, the un-named block. These are blocks that act as placeholders in the description and are instantiated with information when the nodes are used at run-time. Un-named blocks in the description are represented by a lower case character in italics, e.g. *x*.

Each block in the graphical notation corresponds to a described node, and the text is the name of the node. If a block contains an *x*, it means that the name is not known in the description and that it will be assigned at run-time. The names of all ancestral nodes are used to construct the URI for each node in the Management Object. It is not possible to see the actual parameters, or data, stored in the nodes by looking at the graphical notation of a Management Object.

For a further introduction to this graphical notation, please refer to [DMStdObj].

5.2 DDF compliance

The Management Object descriptions in this document are normative. However, the descriptions also contain a number of informative aspects that could be included to enhance readability or serve as examples. Other informative aspects are, for instance, the ZeroOrMore and OneOrMore elements, where implementations may introduce restrictions. All these exceptions are listed here:

- All XML comments, e.g. "<!–some text →", are informative.
- The descriptions do not contain an RTProperties element, or any of its child elements, but a description of an actual implementation of this object MAY include these.
- If a default value for a leaf node is specified in a description, by the DefaultValue element, an implementation MUST supply its own appropriate value for this element. If the DefaultValue element is present in the description of a node, it MUST be present in the implementation, but MAY have a different value.

- The value of all Man, Mod, Description and DFTitle elements are informative and included only as examples.
- Below the interior nodes Ext and BearerParams, an implementation may add further nodes at will.
- The contents of the AccessType element MAY be extended by an implementation.
- If any of the following AccessType values are specified, they MUST NOT be removed in an implementation: Copy, Delete, Exec, Get, and Replace.
- If the AccessType value Add is specified it MAY be removed in an implementation if the implementation only supports a fixed number of child nodes.
- An implementation MAY replace the ZeroOrMore or OneOrMore elements with ZeroOrN or OneOrN respectively. An appropriate value for *N* must also be given with the …*OrN* elements.

5.2.1 Conformance Definitions

The status definition in the node definitions indicates if the client supports that node or not. If the status is “Required” then the client MUST support that node in the case the client supports the parent node.

6. Push Management Object

The Push MO is an object for OMA Push that defines rules for acceptance of push messages, depending on the source or type of content. The MO can be initially provisioned and is used for continuous provisioning to update service configurations.

If the Push MO is provisioned together with other management object(s) during bootstrap then [DMTNDS] and [DMBOOT] MUST be used.

The OMA Push Management Object consists of relevant parameters required by the PUSH enabler. It is compatible with OMA Device Management protocol specifications, version 1.2, and is defined using the OMA DM Device Description Framework as described in [DM-TND-V1-2] and [DMSTDOBJ].

The Management Object Identifier is: urn:oma:mo:oma-push:1.1

The Management Objects associated with OMA Push management are assembled under an unnamed interior node *x*, dynamically or statically created.

Protocol Compatibility: This object is compatible with OMA Device Management protocol specifications, version 1.2 [DMPRO].

6.1 Figure of the Management Object (Informative)

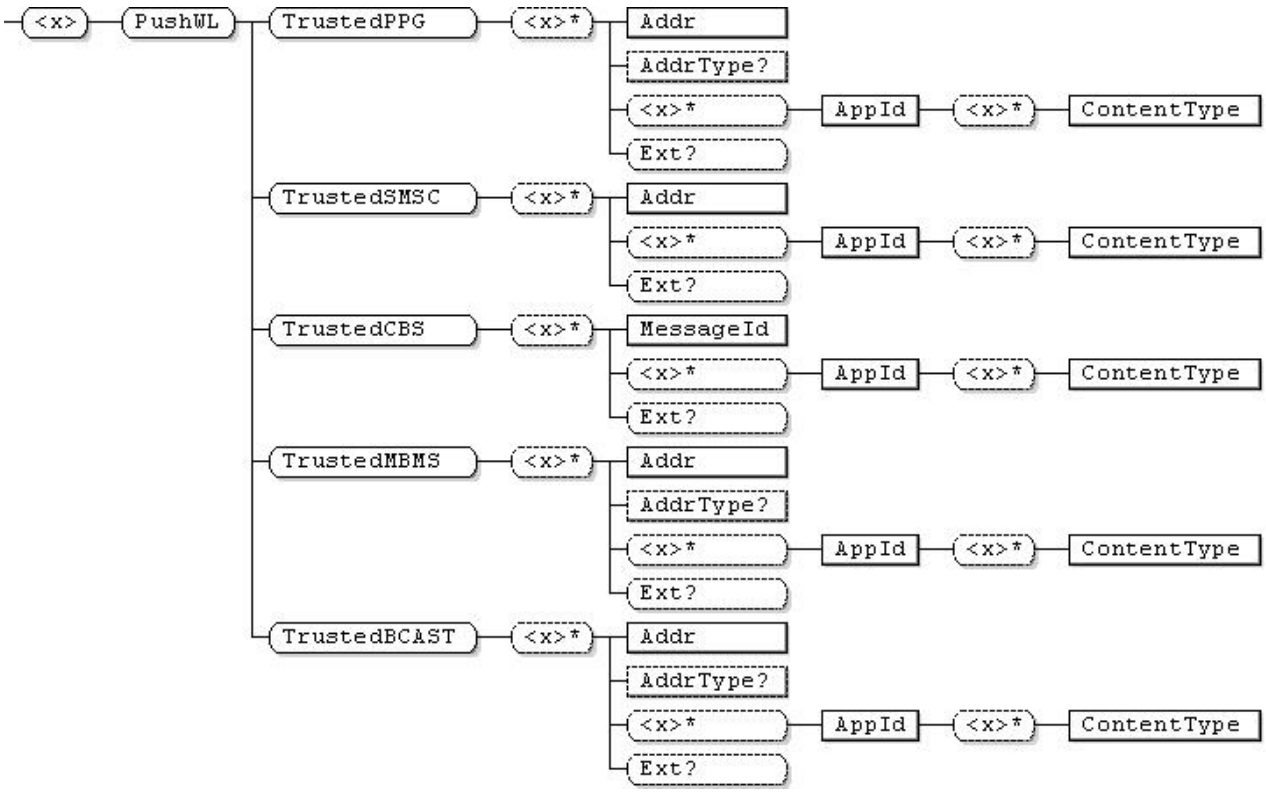

Figure 1: Push Management Object

6.2 Push Management Object Parameters

This section describes the parameters for the OMA PUSH MO. The procedure to validate whether the Push PDU originates from a trusted source or not is defined in [PUSH23].

<x>

Status	Tree Occurrence	Format	Min. Access Types
Required	One	node	Get

This interior node groups together the parameters of a Push Management Object. The purpose of this interior node is to group together the parameters of a single OMA PUSH object. The ancestor elements of this node define the position in the management tree of the OMA PUSH object. But the structure of the DM tree and hence positions in the tree of management objects is out of scope of this specification.

The type of this node MUST be the Push Management Object ID “urn:oma:mo:oma-push:1.1”.

PushWL

Status	Tree Occurrence	Format	Min. Access Types
Required	One	node	Get

The PushWL interior node is used to hold a list of PPG Addresses and/or a list of SMSCAddresses.

PushWL/TrustedPPG

Status	Tree Occurrence	Format	Min. Access Types
Required	One	node	Get

The TrustedPPG interior node is used to list the PPG addresses from which push message is trusted. It makes it possible to specify a plurality of addresses.

PushWL/ TrustedPPG/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized Push Proxy Gateway addresses.

PushWL/ TrustedPPG/<X>/Addr

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

The Addr node holds addresses of different types, for example, an IP address or an URI. The type of address in the field can be determined on the AddrType node.

PushWL/TrustedPPG/<X>/AddrType

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the type of NAP address supplied as the Addr leaf node value. If this node is omitted, the type of the Addr node value MUST be IPv4.

AddrType	Value of Addr node
IPv4	An IPv4 address [RFC791] represented in string form dotted-decimal CIDR notation Subnetwork addressing using the CIDR notation is allowed (e.g. 12.11.10.9/15) [RFC1918] IPV4 is the default value of the AddrType node
IPv6	An IPv6 address represented in string form as in [RFC3513] Subnetwork addressing using the CIDR notation is allowed [RFC2373]
URI	URI formed as in [RFC3986]
E164	A phone number according to the E164 scheme [GENFORM]

PushWL/TrustedPPG/<X>/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push applications for a PPG source address.

PushWL/TrustedPPG/<X>/<X>/AppId

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a Push Application Identifier: a URN [RFC2141] for an OMNA-registered push application value as defined in [PushAppId], or an unregistered URN value. MAY include '*' as a wildcard in either the Namespace Identifier or Namespace Specific String part.

PushWL/TrustedPPG/<X>/<X>/AppId/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push content types for a push application.

PushWL/TrustedPPG/<X>/<X>/AppId/<X>/ContentType

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a content type (MIME media type [RFC2141]). MAY include '*' as a wildcard.

PushWL/TrustedPPG/<X>/Ext

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

The Ext is an interior node where the vendor-specific information about the OMA_OMA-PUSH MO is placed ("vendor" means application vendor, device vendor etc.). Usually the vendor extension is identified by a vendor-specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include a non-standardized sub-tree.

PushWL/TrustedSMSC

Status	Tree Occurrence	Format	Min. Access Types
Required	One	node	Get

The TrustedSMSC interior node is used to list the SMSC addresses from which push message content is trusted. It makes it possible to specify a plurality of addresses.

PushWL/ TrustedSMSC/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized SMSC addresses.

PushWL/TrustedSMSC/<X>/Addr

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

The Addr node holds the trusted SMSC E164 address.

PushWL/TrustedSMSC/<X>/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push applications for a SMSC source address.

PushWL/TrustedSMSC/<X>/<X>/AppId

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a Push Application Identifier: a URN [RFC2141] for an OMNA-registered push application value as defined in [PushAppId], or an unregistered URN value. MAY include '*' as a wildcard in either the Namespace Identifier or Namespace Specific String part.

PushWL/TrustedSMSC/<X>/<X>/AppId/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push content types for a push application.

PushWL/TrustedSMSC/<X>/<X>/AppId/<X>/ContentType

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a content type (MIME media type [RFC2141]). MAY include '*' as a wildcard.

PushWL/TrustedSMSC/<X>/Ext

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

The Ext is an interior node where the vendor-specific information about the OMA_OMA-PUSH MO is placed ("vendor" means application vendor, device vendor etc.). Usually the vendor extension is identified by a vendor-specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include a non-standardized sub-tree.

PushWL/TrustedCBS

Status	Tree Occurrence	Format	Min. Access Types
Required	One	node	Get

The TrustedCBS interior node is used to list the CBS message identifiers from which push message content is trusted. It makes it possible to specify a plurality of message identifiers.

PushWL/ TrustedCBS/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized CBS message identifiers.

PushWL/TrustedCBS/<X>/MessageId

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a CBS message identifier.

PushWL/TrustedCBS/<X>/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push applications for a CBS message identifier.

PushWL/TrustedCBS/<X>/<X>/AppId

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a Push Application Identifier: a URN [RFC2141] for an OMNA-registered push application value as defined in [PushAppId], or an unregistered URN value. MAY include '*' as a wildcard in either the Namespace Identifier or Namespace Specific String part.

PushWL/TrustedCBS/<X>/<X>/AppId/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push content types for a push application.

PushWL/TrustedCBS/<X>/<X>/AppId/<X>/ContentType

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a content type (MIME media type [RFC2141]). MAY include '*' as a wildcard.

PushWL/TrustedCBS/<X>/Ext

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

The Ext is an interior node where the vendor-specific information about the OMA-PUSH MO is placed ("vendor" means application vendor, device vendor etc.). Usually the vendor extension is identified by a vendor-specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include a non-standardized sub-tree.

PushWL/TrustedMBMS

Status	Tree Occurrence	Format	Min. Access Types
Required	One	node	Get

The TrustedMBMS interior node is used to list the MBMS source addresses from which push message content is trusted. It makes it possible to specify a plurality of addresses.

PushWL/ TrustedMBMS/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized MBMS source addresses.

PushWL/TrustedMBMS/<X>/Addr

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies addresses of different types, for example, an IP address or an URI. The type of address in the field can be determined on the AddrType node.

PushWL/TrustedMBMS/<X>/AddrType

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the type of NAP address supplied as the **Addr** leaf node value. If this node is omitted, the type of the Addr node value MUST be IPv4.

AddrType	**Value of Addr node**
IPv4	An IPv4 address [RFC791] represented in string form dotted-decimal CIDR notation Subnetwork addressing using the CIDR notation is allowed (e.g. 12.11.10.9/15) [RFC1918] IPV4 is the default value of the AddrType node
IPv6	An IPv6 address represented in string form as in [RFC3513] Subnetwork addressing using the CIDR notation is allowed [RFC2373]
URI	URI formed as in [RFC3986]

PushWL/TrustedMBMS/<X>/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push applications for an MBMS source address.

PushWL/TrustedMBMS/<X>/<X>/AppId

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a Push Application Identifier: a URN [RFC2141] for an OMNA-registered push application value as defined in [PushAppId], or an unregistered URN value. MAY include '*' as a wildcard in either the Namespace Identifier or Namespace Specific String part.

PushWL/TrustedMBMS/<X>/<X>/AppId/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push content types for a push application.

PushWL/TrustedMBMS/<X>/<X>/AppId/<X>/ContentType

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a content type (MIME media type [RFC2141]). MAY include '*' as a wildcard.

PushWL/TrustedMBMS/<X>/Ext

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

The Ext is an interior node where the vendor-specific information about the OMA-PUSH MO is placed ("vendor" means application vendor, device vendor etc.). Usually the vendor extension is identified by a vendor-specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include a non-standardized sub-tree.

PushWL/TrustedBCAST

Status	Tree Occurrence	Format	Min. Access Types
Required	One	node	Get

The TrustedBCAST interior node is used to list the OMA BCAST source addresses from which push message content is trusted. It makes it possible to specify a plurality of addresses.

PushWL/ TrustedBCAST/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized OMA BCAST source addresses.

PushWL/TrustedBCAST/<X>/Addr

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies addresses of different types, for example, an IP address or an URI. The type of address in the field can be determined on the AddrType node.

PushWL/TrustedBCAST/<X>/AddrType

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the type of NAP address supplied as the **Addr** leaf node value. If this node is omitted, the type of the Addr node value MUST be IPv4.

AddrType	**Value of Addr node**
IPv4	An IPv4 address [RFC791] represented in string form dotted-decimal CIDR notation Subnetwork addressing using the CIDR notation is allowed (e.g. 12.11.10.9/15) [RFC1918] IPV4 is the default value of the AddrType node
IPv6	An IPv6 address represented in string form as in [RFC3513] Subnetwork addressing using the CIDR notation is allowed [RFC2373]
URI	URI formed as in [RFC3986]

PushWL/TrustedBCAST/<X>/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push applications for a BCAST source address.

PushWL/TrustedBCAST/<X>/<X>/AppId

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a Push Application Identifier: a URN [RFC2141] for an OMNA-registered push application value as defined in [PushAppId], or an unregistered URN value. MAY include '*' as a wildcard in either the Namespace Identifier or Namespace Specific String part.

PushWL/TrustedBCAST/<X>/<X>/AppId/<X>

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This node contains the instances of authorized push content types for a push application.

PushWL/TrustedBCAST/<X>/<X>/AppId/<X>/ContentType

Status	Tree Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a content type (MIME media type [RFC2141]). MAY include '*' as a wildcard.

PushWL/TrustedBCAST/<X>/Ext

Status	Tree Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

The Ext is an interior node where the vendor-specific information about the OMA-PUSH MO is placed ("vendor" means application vendor, device vendor etc.). Usually the vendor extension is identified by a vendor-specific name under the ext node. The tree structure under the vendor identified is not defined and can therefore include a non-standardized sub-tree.

The complete Device Description Framework of this Push management object can be found in [PUSHMO-DDF].

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a		

A.2 Draft/Candidate Version 1.1 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS-Push_MO-V1_1-	07 Oct 2009	All	Initial version for Push 2.3, based upon OMA-TS-Push_MO-V1_0-20071002-C, as agreed in OMA-CD-PUSH-2009-0095R01-INP_Push2.3_Push_MO_Baseline
	20 Jan 2009	All	Updated per agreed CR: OMA-CD-PUSH-2010-0012R01-CR_2.3_CONRR_Push_MO
Candidate Versions OMA-TS-Push_MO-V1_1-	16 Mar 2010	All	Status changed to Candidate by TP: OMA-TP-2010-0106-INP_PUSH_V2_3_ERP_for_Candidate_Approval