Firmware Update Management Object

Approved Version 1.0 – 09 Feb 2007

Open Mobile Alliance

OMA-TS-DM-FUMO-V1_0-20070209-A

Contents

Figures

Tables

1. Scope

This document specifies management object(s) and their necessary behavior to support the updating of firmware in mobile devices. It leverages the OMA DM enabler [OMADM] and supports alternate download mechanisms (such as OMA Download [DLOTA]). This represents the interface between the client and server required to manage the update of a mobile device's firmware.

The content and format of the update package, and the process used to update firmware in the device, are implementation specific and are not covered by this specification.

2. References

2.1 Normative References

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

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

[IOPPROC] "OMA Interoperability Policy and Process", Version 1.13, Open Mobile Alliance™, OMA-IOP-Process-V1_13, URL:http://www.openmobilealliance.org/

[OMADM] *OMA Device Management, Version 1.2*. Open Mobile Alliance™. 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

[RFC2234] "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997, URL:http://www.ietf.org/rfc/rfc2234.txt

2.2 Informative References

[DLOTA] OMA Download, Version 1.0, Open Mobile Alliance™. URL:http://www.openmobilealliance.org/documents.html

[DMNOTI] "OMA Device Management Notification Initiated Session, Version 1.2". Open Mobile Alliance™. OMA-TS-DM-Notification-V1_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

[RFC2616] "Hypertext Transfer Protocol – HTTP/1.1". Network Working group. June 1999. URL: http://www.ietf.org/rfc/rfc2616.txt

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 also the DM Tree and Description [DMTND] document for definitions of terms related to the management tree.

3.3 Abbreviations

FUMO	Firmware Update Management Object
OMA	Open Mobile Alliance

4. Introduction

This specification provides information on management objects associated with firmware updates in OMA-DM based mobile devices and the behavior associated with the processing of the management objects. Also specified is behavior associated with the 'Exec' command and Generic Alerts.

The problem solved is the lack of an interoperable firmware update solution for mobile devices. This specification provides an interface between client and server to support this firmware update. This solution comprises the download of update package(s), the subsequent installation of the update packes(s) to update firmware, and the reporting of success or error results and associated status information.

This specification enables mobile operators, service providers, infrastructure manufacturers, device manufacturers, and software vendors to develop and deploy interoperable firmware update solutions.

The primary target audience for these management objects are engineers providing firmware update solutions and update package download solutions.

5. Firmware Update Management Object

The parameters associated with a single firmware update are assembled into a firmware update management object as shown in Figure 1. A firmware update management object may be either permanent or dynamic. There may be one or more such firmware update management objects in a device management tree. Only one update package, or reference to an update package, is associated with each such management object. The grouping or placement within the device management tree of multiple management objects, such as under a common node, is not addressed in this specification.

Management Object identifier: urn:oma:mo:oma-fumo:1.0

Protocol Compatibility: This object is compatible with OMA Device Management, version 1.2 [OMADM] and any later compatible versions of OMA Device Managment.

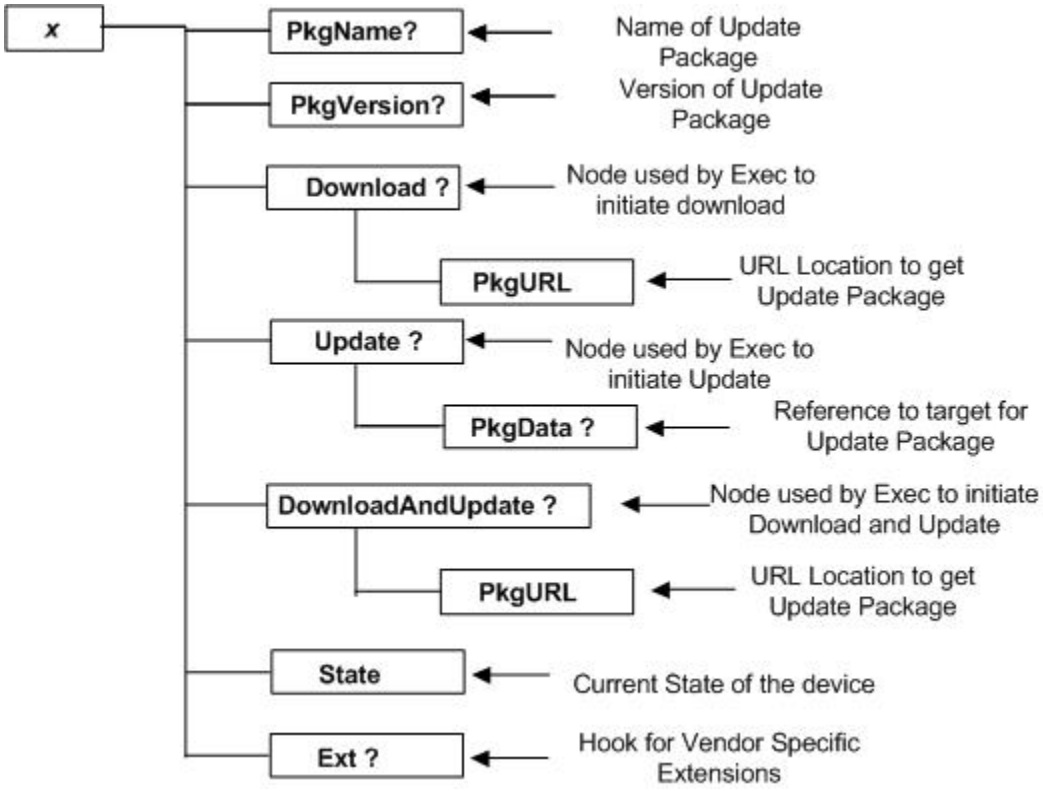

Figure 1: Firmware Update Mangement Object Structure (Informative)

5.1 Firmware Update Management Object Parameters

The following are the nodes of the Firmware Update management object.

5.1.1 Node: *x*

This interior node acts as a placeholder for a firmware upgrade package unique identifier. The node Type property MUST correspond to the management object identifier specified in section 5. The manufacturer MAY pre-create permanent nodes for *x*, allow update package nodes *x* to be created as needed, or a combination of these two methods. For example, permanent nodes might be created for all firmware packages known at the time of manufacture, with additional nodes added dynamically as new features become available. An example would be to include nodes labeled FWPkg1, FWPkg2..FWPkg*n*.

The DDF file provided by the device manufacturer MAY specify where the *x* node is to be located in the management tree of the device.

Occurrence: ZeroOrMore

Format: Node

Access Types: Get

Values: N/A

5.1.2 Node: *x*/PkgName

This optional node specifies the Name associated with the firmware update package.

Occurrence: ZeroOrOne

Format: Chr

Access Types: Get

Values: N/A

5.1.3 Node: *x*/PkgVersion

This optional node specifies the Version information for the firmware update package. The version information is device manufacture specific and can contain any data.

Occurrence: ZeroOrOne

Format: Chr

Access Types: Get

Values: N/A

5.1.4 Node: *x*/Download

This interior node is the target of an 'Exec' command in order to initiate a firmware download for the specified update package and is optional.

Occurrence: ZeroOrOne

Format: Node

Access Types: Exec, Get

Values: N/A

5.1.5 Node: *x*/Download/PkgURL

This node specifies the URL where the firmware update package or download descriptor is located. This URL is used for alternative download mechanisms (such as HTTP Get [RFC2616] or Descriptor Based Download [DLOTA]).

Occurrence: One

Format: Chr

Access Types: Get, Replace

Values: N/A

5.1.6 Node: *x*/Update

This interior node is a target of an 'Exec' command in order to initiate a firmware update for the specified update package and is optional.

Occurrence: ZeroOrOne

Format: Node

Access Types: Exec, Get

Values: N/A

5.1.7 Node: *x*/Update/PkgData

This node is the target of a 'Replace' command when DM is used to directly provide the binary firmware update package.

Occurrence: ZeroOrOne

Format: Bin

Access Types: Replace

Values: N/A

5.1.8 Node: *x*/DownloadAndUpdate

This interior node is the target of an 'Exec' command invoked to initiate a firmware download and update for the specified update package and is optional. The update MUST take place as soon as practical after download.

Occurrence: ZeroOrOne

Format: Node

Access Types: Exec, Get

Values: N/A

5.1.9 Node: *x*/DownloadAndUpdate/PkgURL

This node specifies the URL where the firmware update package or download descriptor is located, that is to be downloaded and installed at the next practical opportunity. This URL is used for alternative download mechanisms (such as HTTP Get [RFC2616] or Descriptor Based Download [DLOTA]).

Occurrence: One

Format: Chr

Access Types: Get, Replace

Values: N/A

5.1.10 Node: *x*/State

Contains a value indicating the current state of the mobile device with respect to this firmware update.

Occurrence: One

Format: Int

Access Types: Get

Values: See table below

The following state table enumerates the valid states:

Table 1: Valid States

State	Description	Integer Value
Idle / Start	No pending operation	10
Download Failed	Download failed	20
Download Progressing	Download has started	30
Download Complete	Download has been completed successfully	40
Ready to Update	Have data and awaiting command to start update	50
Update Progressing	Update has started	60

Update Failed / Have Data	Update failed but have update package	70
Update Failed / No Data	Update failed and no update package available	80
Update Successful / Have Data	Update complete and data still available	90
Update Successful / No Data	Data deleted or removed after a successful Update	100

5.1.11 Node: *x*/Ext

This is a node for supporting vendor specific extensions.

Occurrence: ZeroOrOne

Format: Node

Access Types: Get

Values: N/A

6. Behavior Associated with the Management Object

The following diagram shows the various valid states of the mobile device as they are related to firmware updates. It is possible that the server may not see some of these states.

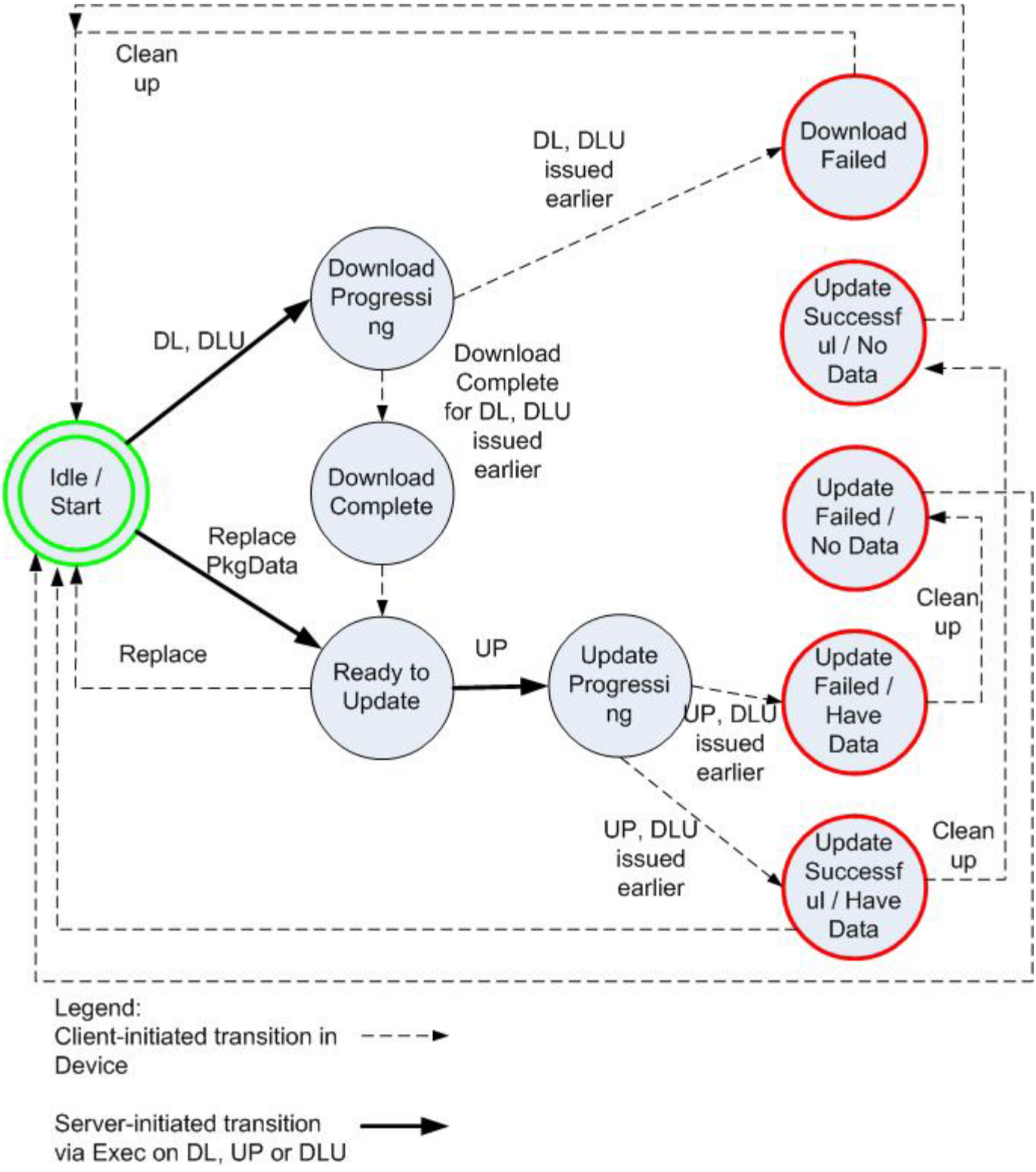

Figure 2: Firmware Update State Diagram

Typically, the starting state is 'Idle/Start' and the terminating states are one of:

- Update Failed / Have Data
- Update Failed / No Data
- Update Successful / Have Data
- Update Successful / No Data
- Download Failed

6.1 'Exec' Command

The 'Exec' command MUST be supported.

The server issues 'Exec' commands to initiate long running operations in the client, such as download and update. The result of the 'Exec' command, encoded as a ResultCode, is returned in a Generic Alert following completion of the operation. A correlator, if supplied with the 'Exec' command, is also returned in that Generic Alert. The client MUST send a status 202 (asynchronous) for the 'Exec' operation if the command is accepted for later processing.

Optionally a User Interaction Alert [DMPRO] can be used for soliciting user opt-in prior to the execution of the Exec command on the Download node.

In the case of a download of an update package employing the large-object download feature of the OMA-DM protocol [DMPRO], the 'Replace' command is used by the DM server to initiate the download, prior to the invocation of an 'Exec' command to invoke the update activity.

The State element in the management object is updated to indicate the state the client reached during the corresponding Exec initiated update or download activity (See Chapter 6.).

6.1.1 Exec Command Semantics for Alternate Download

The server issues an 'Exec' command targeting the *x*/Download node. The client initiates an alternate download operation from the URL identified in the *x*/Download/PkgURL value. When the download operation is complete, the client issues a Generic Alert indicating the result of the download operation.

6.1.1.1 Example of 'Exec' Command for Alternate Download

Pre-Condition: The following element needs to be set with an appropriate value:

- *x*/Download/PkgURL is set

Example of 'Exec' command:

```
<Exec>
    <CmdID>3</CmdID>
    <Item>
        <Target>
            <LocURI>x/Download</LocURI>
        </Target>
    </Item>
</Exec>
```

6.1.2 'Exec' Command Semantics for Update

The server issues an 'Exec' command targeting the *x*/Update node. The client applies the previously received update package. When the update operation is complete, the client issues a Generic Alert indicating the result of the update operation.

6.1.2.1 Example of 'Exec' Command for Update

Pre-Condition:

- The firmware update package must be available on the device.

Example of 'Exec' command:

```
<Exec>
    <CmdID>3</CmdID>
    <Item>
        <Target>
            <LocURI>x/Update</LocURI>
        </Target>
    </Item>
</Exec>
```

6.1.3 Exec Command Semantics for DownloadAndUpdate

The server issues an 'Exec' command targeting the *x*/DownloadAndUpdate node. The client initiates an alternate download operation from the URL identified in the *x*/DownloadAndUpdate/PkgURL value. When the download operation is completed successfully, the client applies the received update package without further server intervention. When the update operation is complete, the client issues a Generic Alert indicating the result of the update operation. In the event that the download fails, the client issues a Generic Alert indicating the failure of the download operation.

The Update activity is launched at the next practical opportunity.

6.1.3.1 Example 'Exec' Command for DownloadAndUpdate

Pre-Condition:

The following object needs to be set with an appropriate value:

- *x*/DownAndUpdate/PkgURL

Example of Exec command:

```
<Exec>
    <CmdID>3</CmdID>
    <Item>
        <Target>
            <LocURI>x/DownloadAndUpdate</LocURI>
        </Target>
    </Item>
</Exec>
```

6.2 Use of Generic Alert for Notifications

At the end of the operation invoked by the Exec commands in section 6.1, the device MUST send a notification to the DM server via a Generic Alert [DMPRO] message. The alert message includes the following data:

- An integer result code – Used to report status of the operation
- The URI of the Firmware Update Management Object – Used to identify the source
- An alert type – Used to identify the operation
- Correlator – Used by the server and passed as part of the Exec command

Alerts that are reporting an error or failure condition SHOULD report a severity other than Informational in the Mark field of the Meta information.

Once the Generic Alert message is sent to the DM Server at the end of the operation, the device MUST NOT retry the operation invoked via the Exec command by the DM Server without further server intervention.

NOTE: If the server needs to retrieve additional information, such as State, then the server MAY query the device for those specific nodes.

6.2.1 URI of Firmware Update Management Object

The URI of the Firmware Update Management Object MUST be sent as the source of the Generic Alert [DMPRO] message. This allows the Management Server to identify the origin of the alert.

6.2.2 Firmware Update Alert Types

One of the following alert types MUST be used in a Generic Alert [DMPRO] message originating from a Firmware Update Management Object. The alert types are used to identify the Exec operation that was performed on the device.

- The alert type “org.openmobilealliance.dm.firmwareupdate.download” MUST be used in response to the completion of a Download operation.
- The alert type “org.openmobilealliance.dm.firmwareupdate.update” MUST be used in response to the completion of an Update operation
- The alert type “org.openmobilealliance.dm.firmwareupdate.downloadandupdate” MUST be used in response to the completion of a DownloadAndUpdate operation

6.2.3 Correlator

If the server passes a correlator to the client in the Exec command of a Firmware Update Operation, the client MUST return the same value to the server in the correlator field of the Generic Alert [DMPRO] message.

If the server does not pass a correlator to the client in the Exec command of a Firmware Update Operation, the client MUST NOT send a correlator to the server in the correlator field of the Generic Alert [DMPRO] message.

6.2.4 Result Code

The result code of the operation MUST be sent as an integer value in the Data element of the GenericAlert [DMPRO] message. The ResultCode MUST be one of the values defined below:

Table 2: Result Code

Result Code	Meaning	Usage
200	Successful	Successful - The Request has Succeeded
250 -299	Successful – Vendor Specified	Successful Operation with Vendor Specified ResultCode
400	Management Client Error	Management Client error – based on User or Device behavior
401	User Cancelled	User chose not to accept the operation when prompted
402	Corrupted Firmware Update Package	Corrupted firmware update package, did not store correctly. Detected, for example, by mismatched CRCs between actual and expected.
403	Firmware Update Package – Device Mismatch	Wrong Firmware Update Package delivered to device based on current device characteristics
404	Failed Firmware Update Package Validation	Failure to positively validate digital signature of firmware update package
405	Firmware Update Package Not Acceptable	Firmware Update Package is Not Acceptable
406	Alternate Download Authentication Failure	Authentication was Required but Authentication Failure was encountered when downloading Firmware Update Package
407	Alternate Download Request Time-Out	Client has encountered a time-out when downloading Firmware Update Package
408	Not Implemented	The device does not support the requested operation.
409	Undefined Error	Indicates failure not defined by any other error code
410	Firmware Update Failed	Firmware Update operation failed in device
411	Malformed or Bad URL	The URL provided for alternate download is bad
412	Alternate Download Server Unavailable	The Alternate Download Server is Unavailable or Does not Respond
450 -499	Client Error – Vendor Specified	Client Error encountered for Operation with Vendor Specified ResultCode
500	Alternate Download Server Error	Alternate Download Server Error Encountered
501	Download fails due to device is out of memory	The download fails due insufficient memory in the device to save the firmware update package.
502	Firmware update fails due to device out of memory	The update fails because there isn't sufficient memory to update the device.
503	Download fails due to network issues	The download fails due to network/transport level errors
550 -599	Alternate Download Server Error – Vendor Specified	Alternate Download Server Error encountered for Operation with Vendor Specified ResultCode

In the above table, the series 2xx result codes indicate successful outcome. The series 4xx and 5xx result codes, which indicate unsuccessful outcome, provide indication of failure conditions that resulted in the end of the firmware update activity in the device.

6.3 Support for update package download and firmware update activities

The FUMO 1.0 compliant client MUST support at least one download mechanism to download update packages. This download mechanism MUST be OMA DM based transfer or an alternate download mechanism, such as OMA Download [DLOTA]. In addition, to successfully conduct a firmware update, at least one of the following two activities MUST be supported:

a) Exec on x/Update node

b) Exec on x/DownloadAndUpdate node

7. Client Initiated Firmware Update (Normative)

7.1 General

Firmware Upgrade is in its nature device dependent. This feature is therefore an optional feature for client devices and for servers. This section only defines the format of the client initiated message and not in which circumstances the client device will send it. This message is an information message to the server so that the server should investigate if an update is needed. The device can send two different Firmware Update requests dependent on if it is originated from the Device or a User. The client SHOULD NOT expect any specific Firmware Update action from the server. If the Alert Type is "User Initiated", then the server MAY send User Interaction Commands in the same session to inform the user how the server will handle the firmware update request. The Server may investigate if update is needed in the same session, but may also inform the user that the server will investigate this later on.

The Generic Alert format is used for this notification. The following client requirements MUST be supported if Client Initiated Firmware Update is implemented:

7.1.1 Generic Alert

The message MUST follow the Generic Alert format

7.1.2 Alert Type

The message MUST use the alert type: "org.openmobilealliance.dm.firmwareupdate.devicerequest" for device initiated firmware update and alert type: "org.openmobilealliance.dm.firmwareupdate.userrequest" for user initiated firmware update.

7.1.3 URI

The URI in the alert, if specified, MUST point to the dynamic node (e.g., the <x>) representing a single firmware update management object in the tree. When present, the server investigates the availability of updates for the firmware indicated by the management object. It is further suggested when not present, the server investigates the availability of all relevant firmware updates for the terminal originating the alert. The server MAY query the contents of the management object specified by a URI. The server SHOULD initiate a firmware update if any relevant updates are discovered.

7.1.4 Data

The Data element MUST be included. A client vendor MAY use the data field to supply implementation specific data. In case there is no implementation specific data the value needs to be left empty

8. FUMO Usage (Informative)

8.1 Firmware Update Protocol Overview

The Firmware Update Protocol specifies a set of standard commands with associated parameters and management objects that shall be used for OTA firmware updates. OTA Firmware updates require special attention to handle the discovery, security, download and installation.

OMA DM is the leading standards initiative that focuses on device management for wireless devices. The OMA Download [DLOTA] specification provides a flexible protocol for enabling the download of generic content, controlled through the use of a separate download descriptor. By drawing on elements of these protocols and adding new elements, an effective protocol is constructed that combines Device Management [DMPRO] for controlling the main device management functions and provides for the use of descriptor-based download mechanisms to download larger binary objects such as firmware updates. The download process is abstracted to allow the use of either OMA DM (E.g., Add/Replace) or any suitable alternative download mechanisms (for example, a descriptor based download protocol such as OMA Download [DLOTA].).

The protocol shall support the following process steps in order to achieve an OTA firmware update:

1. Firmware Update Step 1: Firmware Update Initiation
2. Firmware Update Step 2: Device Information Exchange
3. Firmware Update Step 3: Firmware Download
4. Firmware Update Step 4: Firmware Installation
5. Firmware Update Step 5: Notification of Firmware Update

8.1.1 Scenario 1: Firmware Update via OMA DM Download (Replace)

The following example shows how OMA DM is used directly to move a firmware update package to the device using a DM "Replace" command to access a management object representing the actual firmware binary package data:

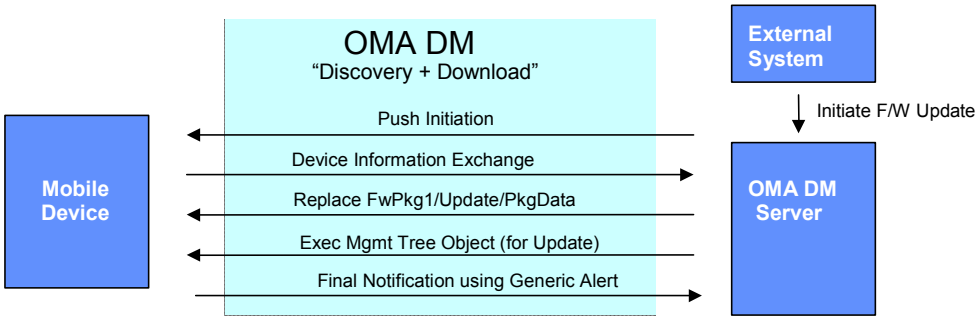

Figure 3: Firmware Update via OMA DM Download

8.1.2 Scenario 2: Firmware Update through an Alternative Download Mechanism

The following examples shows how OMA DM is used to invoke an external download method, using a DM "Replace" command to specify the URL of the download descriptor that describes further details concerning the firmware package and the download method to be used:

8.1.2.1 Example 1: 'Exec' on x/Download node + 'Exec' on x/Update node

The server issues an 'Exec' command targeting the x/Download node. If needed, server will issue 'Exec' command targeting the x/Update node after client sends download notification using Generic Alert to server.

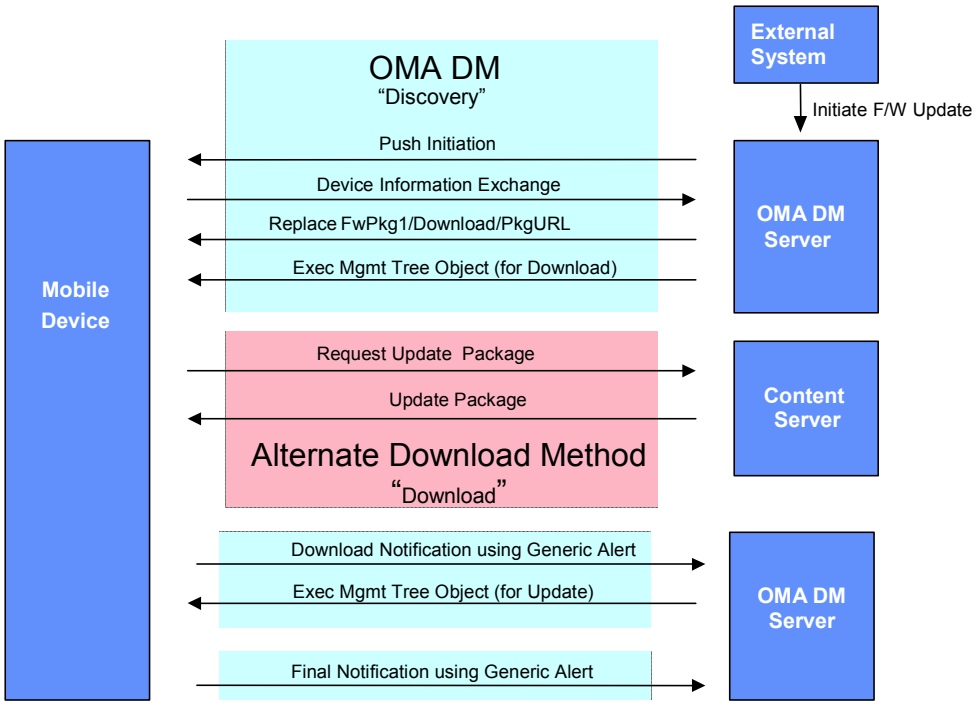

Figure 4: 'Exec' on x/Download and 'Exec' on x/Update

8.1.2.2 Example 2: 'EXEC' on x/DownloadAndUpdate node

The server issues an 'Exec' command targeting the x/DownloadAndUpdate node. Client sends final notification to server after the completion of DownloadAndUpdate operation.

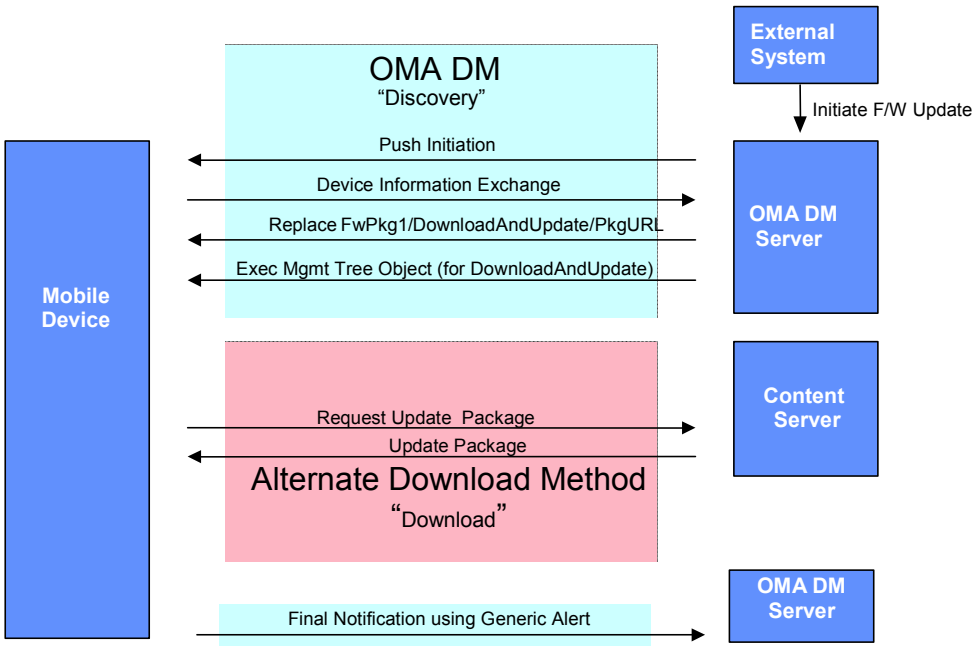

Figure 5: 'Exec' on x/DownloadAndUpdate

8.2 Protocol Definition

8.2.1 Firmware Update Step1: Firmware Update Initiation

In order to begin any kind of firmware update, the device is required to open a data connection to the server. The following mechanisms could be supported to initiate the firmware update process:

- User initiated
- Network initiated

The subscriber experience for user-initiated updates are not addressed in this specification as it does not require specific standardization. Recommended approaches are through menu items and service codes on the device. The user initiated update process would simply launch an OMA DM session.

For network initiated updates, OMA Device Management provides a framework by which clients can be sent a "Notification Initiation Alert" to trigger the client to start the data session. It is the intent of the Firmware Update Protocol to leverage General Package#0 as specified in the OMA DM Notification Initiation Session document [DMNOTI]. OMA DM specifies that WAP Push can be used for this purpose and specifies a format acceptable for the purpose of firmware upgrade initiation.

8.2.2 Firmware Update Step 2: Device Information Exchange

In order to provide a device with the appropriate firmware update, a minimum set of selection criteria is sent by the device to the server. For the purpose of the firmware updates, the minimum set of criteria is the required DevInfo parameters that are mandatory for each OMA DM management session as specified in [DMSTDOBJ, Section 5].

NOTE: The device information exchange can be followed by an optional user interaction prior to the setup of the firmware download process described in the next section. An OMA DM "Alert" command can be used to seek user confirmation.

8.2.3 Firmware Update Step 3: Firmware Download

The firmware download could occur either through a OMA DM 'Replace' command or via an alternative (external descriptor-based download) method by initiating the download process with the appropriate behavioral parameters. There are advantages and disadvantages to each process and it is at the discretion of the operators and manufacturers to select a preferred implementation. The protocol proposed here allows for either download method to be used.

Please refer to Appendix D for further details.

8.2.4 Firmware Update Step 4: Firmware Installation

It is anticipated that there will be multiple methods available on the market to process firmware updates within a device. The firmware update specification's intent is to standardize interoperability between devices and wireless network solutions. Therefore, this specification does not address the method of how a device must process the actual firmware update when it is independent of the network. Instead, this specification provides requirements to achieve an acceptable user experience for the firmware installation.

8.2.4.1 Firmware Install after OMA DM Download

The following Exec command initiates the update process in cases where the firmware update is downloaded into the Pkgdata element using OMA DM Replace:

```
<Exec>
    <CmdID>3</CmdID>
    <Item>
        <Target>
            <LocURI>./x/Update</LocURI>
        </Target>
    </Item>
</Exec>
```

8.2.4.2 Firmware Installation Alternative Download

For alternative download (such as Descriptor based OMA v1.0 Download), it is recommended that the appropriate install parameters are provided prior to the download initiation Exec command. Please reference ' Normative - Exec Command ' section above for the appropriate Exec command.

The update package could be deleted from object storage at the completion of the update when it is no longer needed. The management client could choose to delete it as soon as an update is successfully or unsuccessfully terminated, or whenever prompted to do so by the DM server.

8.2.5 Firmware Update Step 5: Notification of Firmware Update Status

At the completion of the firmware update process, the device notifies the server of the resulting status of the firmware update. This is accomplished through a subsequent client or server initiated OMA DM session to assure that the management server is informed of the final result.

A ResultCode is provided to the DM server via a Generic Alert [DMPRO] notification.

The Generic Alert [DMPRO] command provided by the OMA-DM protocols is to be used by the OMA-DM client to communicate the ResultCode value to the DM Server.

8.2.5.1 Non-Fatal Result Codes

For non-fatal update failures, the end-user can be provided an indication of the failure and return the phone to an operational mode.

In addition to the result codes listed in the ResultCodes table, the ./FwUpdate/x/State element provides additional detailed information regarding the state in which the mobile device is left in at the termination (successful or otherwise) Exec operations invoked on the Update or Download elements.

8.2.5.2 Fatal Failures

Fatal failure will likely render the device inoperable. Therefore, it will not be possible to provide indication to the user or notify the OMA DM server of the failure. For this reason, implementations that do not provide a high degree of fault tolerance are not likely to be used for the purpose of updating firmware.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
OMA-TS-DM-FUMO-V1_0-20070209-A	09 Feb 2007	Status changed to Approved by TP TP Doc ref# OMA-TP-2007-0076R01-INP_ERP_FUMO_V1.0_for_Final_Approval

Appendix B. Static Conformance Requirements (Normative)

The notation used in this appendix is specified in [IOPPROC].

B.1 SCR for FUMO Tree Structure

Item	Function	Reference	Status	Requirement
FWUPDATE-T-001	Use of appropriate management object identifier for the FUMO node	Section 5	M	
FWUPDATE-T-002	Support for x/Download	Section 5.1.4	O	FWUPDATE-T-010
FWUPDATE-T-003	Support for x/DownloadAndUpdate	Section 5.1.8	O	FWUPDATE-T-009
FWUPDATE-T-004	Support for x/Update	Section 5.1.6	O	FWUPDATE-T-011
FWUPDATE-T-005	Support for PkgName	Section 5.1.2	O	
FWUPDATE-T-006	Support for PkgVersion	Section 5.1.3	O	
FWUPDATE-T-007	Support for Ext	Section 5.1.11	O	
FWUPDATE-T-008	Support for State	Section 5.1.10	M	
FWUPDATE-T-009	Support for x/DownloadAndUpdate/PkgURL	Section 5.1.9	O	
FWUPDATE-T-010	Support for x/Download/PkgURL	Section 5.1.5	O	
FWUPDATE-T-011	Support for x/Update/PkgData	Section 5.1.7	O	FWUPDATE-T-004

B.2 SCR for FUMO Client

Item	Function	Reference	Status	Requirement
FWUPDATE-C-001	Support for Package Download Operation	Section 6.3	M	FWUPDATE-C-003 OR FWUPDATE-C-010
FWUPDATE-C-002	Support for Exec	Section 6.3	M	
FWUPDATE-C-003	Support for Alternative Download of Update	Section 6.3,	O	FWUPDATE-C-004 OR FWUPDATE-C-007

Item	Function	Reference	Status	Requirement
	Package			
FWUPDATE-C-004	Support for Exec on x/Download	Section 6.1.1	O	FWUPDATE-T-002 AND (FWUPDATE-C-005 OR FWUPDATE-C-006)
FWUPDATE-C-005	Support for Add of x/Download/PkgURL	Section 5.1.5	O	
FWUPDATE-C-006	Support for Replace of x/Download/PkgURL	Section 5.1.5	O	
FWUPDATE-C-007	Support for Exec on x/DownloadAndUpdate	Section 5.1.8	O	FWUPDATE-T-003 AND (FWUPDATE-C-008 OR FWUPDATE-C-009)
FWUPDATE-C-008	Support for Add of x/DownloadAndUpdate/PkgURL	Section 5.1.9	O	
FWUPDATE-C-009	Support for Replace of x/DownloadAndUpdate/PkgURL	Section 5.1.9	O	
FWUPDATE-C-010	Support for OMA DM Based Package Download	Section 6.3	O	FWUPDATE-T-004 AND (FWUPDATE-C-011 OR FWUPDATE-C-012)
FWUPDATE-C-011	Support for Add of x/Update/PkgData	Section 5.1.7	O	
FWUPDATE-C-012	Support for Replace of x/Update/PkgData	Section 5.1.7	O	
FWUPDATE-C-013	Support for Update Operation	Section 6.1.2, 5.1.6	M	FWUPDATE-C-007 OR FWUPDATE-C-014
FWUPDATE-C-014	Support for Exec on x/Update	Section 5.1.6	O	FWUPDATE-C-004 OR FWUPDATE-C-010
FWUPDATE-C-015	Support for Generic Alert for result reporting	Section 6.2	M	
FWUPDATE-C-016	Use FUMO URI for result reporting	Section 6.2.1	M	
FWUPDATE-C-017	Use predefined result codes for result reporting	Section 6.2.4	M	
FWUPDATE-C-018	Use predefined alert types for result reporting	Section 6.2.2	M	
FWUPDATE-C-019	Support for Correlator	Section 6.2.3	M	
FWUPDATE-C-020	Use alert severities for result reporting	Section 6.2	O	
FWUPDATE-C-021	Support for Client Initiated Firmware Update	Section 7	O	FWUPDATE-C-022 AND FWUPDATE-C-023 AND FWUPDATE-C-025 AND FWUPDATE-C-026
FWUPDATE-C-022	Support for Generic Alert for Client Initiated Firmware Update	Section 7.1.1	O	
FWUPDATE-C-	Use of the predefined Alert	Section	O	

Item	Function	Reference	Status	Requirement
023	Types for Client Initiated Firmware Update	7.1.2		
FWUPDATE-C-024	Use of the FUMO URI	Section 7.1.3	O	
FWUPDATE-C-025	Use of String as Data Type	Section 7.1.4	O	
FWUPDATE-C-026	Use of User Interaction Alert prior to update	Section 6.1	O	

B.3 SCR for FUMO Server

Item	Function	Reference	Status	Requirement
FWUPDATE-S-001	Support for the Firmware Update Management Object	Section 5	M	
FWUPDATE-S-002	Support for receiving Generic Alert	Section 6.2	M	
FWUPDATE-S-003	Support for Correlator	Section 6.2.3	O	
FWUPDATE-S-004	Support for Exec	Section 6.3	M	

Appendix C. Recommendations (Informative)

a) It is anticipated that updating the firmware will require that the device become inoperable during the time of the firmware upgrade. It is also likely that the update process will not be capable of immediately returning to operating condition in the case of an interruption. In the case that the device cannot immediately return to operating condition, it is recommended that the end-user be presented with a warning that the device will be offline and the approximate time of the update prior to beginning the firmware installation.

b) Deletion of the Update package under the /*x*/ node is an activity that needs to be conducted after successful update, after an unsuccessful update attempt, after an aborted update attempt. However, any specification when such deletes should occur is not addressed in this specification. The management client could choose to delete it as soon as an update is successfully or unsuccessfully terminated, or whenever prompted to do so by the DM server.

c) PkgURL and PkgData are mutually exclusive. Only one of them needs to be set.

d) The PkgData node contains the actual binary firmware upgrade package. Once the package is installed, the client could remove the data to save space, leaving the node empty. Similarly, and the end of an update activity, the client could remove the update package downloaded from a server specified by a PkgURL.