Enabler Validation Plan for Download OTA

Candidate Version 2.0 – 31 Mar 2008

Open Mobile Alliance

OMA-EVP-DLOTA-V2_0-20080331-C

Contents

Figures

Tables

1. Scope

This document details the Validation plan for the DLOTA 2.0 Enabler Release. The successful accomplishment of the validation activities will be required for the Enabler to be considered for Approved status.

The validation plan for the DLOTA 2.0 Enabler Release specifications is based on testing expectations in the Enabler Test Requirements (ETR). While the specific test activities to be performed are described in the Enabler Test Specification (ETS) the test environment is described in this plan. This test environment details infrastructure, operational and participation requirements identified for the needed testing activities.

1.1 Assumptions

None.

1.2 Exclusions

None.

2. References

2.1 Normative References

[IOPPROC] "OMA Interoperability Policy and Process", Version 1.5, Open Mobile Alliance™, OMA-ORG-IOP_Process-V1_5, URL:http://www.openmobilealliance.org/

[IOPTFG] "OMA TestFest Participation Guidelines", Version 1.3, Open Mobile AllianceTM, OMA-IOP-TestFest-Participation-Guidelines-V1_3, 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

[DLOTAv2.0] "OMA Enabler Release Definition for DLOTA Version 2.0". Open Mobile Alliance™, OMA-ERELD-DLOTA-V2_0, URL:http://www.openmobilealliance.org/

[IOPETR] "OMA Enabler Test Requirements for DLOTA Version 2.0". Open Mobile Alliance™, OMA-ETR-DLOTA-V2_0, URL:http://www.openmobilealliance.org/

[IOPETS] "OMA Enabler Test Specification for DLOTA Version 2.0". Open Mobile Alliance™, OMA-ETS-DLOTA-V2_0-20061219-C URL:http://www.openmobilealliance.org/

[IOPEICSClient] "OMA Enabler Implementation Compliance Statement for DLOTA Version 2.0". Open Mobile Alliance™, OMA-EICS-DLOTA_Client-V2_0, URL:http://www.openmobilealliance.org/

[IOPEICSServer] "Enabler Implementation Compliance Statement for DLOTA Version 2.0". Open Mobile Alliance™, OMA-EICS-DLOTA_Server-V2_0, URL:http://www.openmobilealliance.org/

2.2 Informative References

[OMADICT] "Dictionary for OMA Specifications", Version 2.6, Open Mobile Alliance™, OMA-ORG-Dictionary-V2_6, URL:http://www.openmobilealliance.org/

[BCAST] "OMA Mobile Broadcast", Version 1.0, Open Mobile Alliance™, OMA-ERP-BCAST-V1_0, URL:http://www.openmobilealliance.org/

[DLOTAv1.0] "OMA Download", Version 1.0, Open Mobile Alliance™, OMA-DL-V1_0, URL: http://www.openmobilealliance.org/

[DLREQ] "Download Over-The-Air Requirements", Version 2.0, Open Mobile Alliance™, OMA-RD-DLOTA-V2_0, URL:http://www.openmobilealliance.org/

[DRM1.0] "OMA Digital Rights Management", Version 1.0, Open Mobile Alliance™, OMA-DRM-V1_0, URL:http://www.openmobilealliance.org/

[DRM2.0] "OMA Digital Rights Management", Version 2.0, Open Mobile Alliance™, OMA-ERP-DRM-V2_0, URL:http://www.openmobilealliance.org/

[DVB-H] "Digital Video Broadcasting (DVB): Transmission System for Handheld Terminals (DVB-H)", ETSI EN 302 304. http://portal.etsi.org/

[HTTP] "Hypertext Transfer Protocol – HTTP/1.1", RFC2616, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999. http://www.ietf.org/rfc/rfc2616.txt

[MBMS] "Multimedia Broadcast/Multicast Service (MBMS); (Release 6)", 3GPP TS 22.146 v6.6.0. http://www.3gpp.org/ftp/Specs/html-info/22146.htm

"Multimedia Broadcast/Multicast Service (MBMS) User services; (Release 6)", 3GPP TS 22.246 v6.2.0. http://www.3gpp.org/ftp/Specs/html-info/22246.htm

[MIDP] "JSR-000118 Mobile Information Device Profile 2.0", Java Community Process, URL:

http://jcp.org/aboutJava/communityprocess/final/jsr118/

[MMS] "OMA Multimedia Messaging Service", Version 1.3, Open Mobile Alliance™, OMA-ERP-MMS-V1_3, URL:http://www.openmobilealliance.org/

[SSL] "The SSL 3.0 Protocol", Netscape Communications Corp., A. Frier, P. Karlton, and P. Kocher, Nov 18, 1996, URL:http://wp.netscape.com/

[TLS] "Transport Layer Security (TLS) Version 1.0", RFC 2246, T. Dierks and C. Allen, Jan 1999. URL:http://www.ietf.org/rfc/rfc2246.txt

[WAP Push] "Push Architectural Overview", WAP-250-PushArchOverview-20010703-a

[WSP] "Wireless Session Protocol Specification", WAP-230-WSP-20010705-

[WTLS] "Wireless Transport Layer Security", WAP-261-WTLS-20010406-a

3. Terminology and Conventions

3.1 Conventions

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

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

3.2 Definitions

Compound Objects	Content which is composed of one or more Media Objects which have relationship each other.
Content Provider	A human entity or a company which provides Media Objects to the User
Download Agent	A User agent in the device responsible for downloading a Media Object described by a download descriptor.
Download Descriptor	Metadata about a Media Object and instructions to the download agent for how to download it.
Download OTA Server	A logical entity which encompasses the all server components (i.e. Presentation Server, Download Server, and Status Report Server).
Download Server	A Web server hosting Media Objects available for download using the DLOTA protocol
Media Object	A resource on a Web server that can be downloaded.
Multiple Objects	Content which is composed of one or more Media Objects. However the Media Objects do not always have relationship each other.
Presentation Server	A Web server presenting a download service to the User.
Status Report Server	A Web server accepting status reports from the download agent.
User	An entity which uses services as defined in [DICTIONARY].

3.3 Abbreviations

AP	Access Point
DD	Download Descriptor
DLOTA	Download Over The Air
DRM	Digital Rights Management
DVB-H	Digital Video Broadcasting for Handheld
ETS	Enabler Test Specification
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
LAN	Local Access Network
MBMS	Multimedia Broadcast Multicast Service
MMS	Multimedia Messaging Service
MMSC	MMS Centre
OMA	Open Mobile Alliance
OTA	Over The Air
RADIUS	Remote Authentication Dial-In User Service
SMS	Short Message Service
SSL	Secure Socket Layer

TLS	Transport Layer Security
URI	Uniform Resource Identifier
URL	Uniform Resource Location
WAP	Wireless Access Protocol
WSP	Wireless Session Protocol
WTLS	Wireless Transport Layer Security
XHTML	Extended Hyper Text Markup Language

4. Enabler Validation Description

See section 5.

5. TestFest Activities

5.1 Enabler Test Guidelines

While OMA DLOTA version 1.0 [DLOTAv1.0] provides a general framework for downloading Media Objects, OMA DLOTA version 2.0 extends OMA DLOTA version 1.0 to support following new use cases.

- Functions that are already provided by OMA DLOTA version 1.0:
  1. Basic download
  2. Combined delivery
- New functions that extends OMA DLOTA version 1.0:
  3. Updating and removing Media Objects
  4. Download from multiple servers
  5. Download of compound objects and multiple objects
  6. Download of chunked Media Objects
  7. Control of User confirmation prompt
  8. Support for resumable download session
  9. Authentication of trusted entity and content integrity check
- New major functions that satisfy market's requirements:
  10. Pre-downloading of Media Objects
  11. Download timing reservation
  12. Progressive download
  13. Download OTA over broadcast protocols

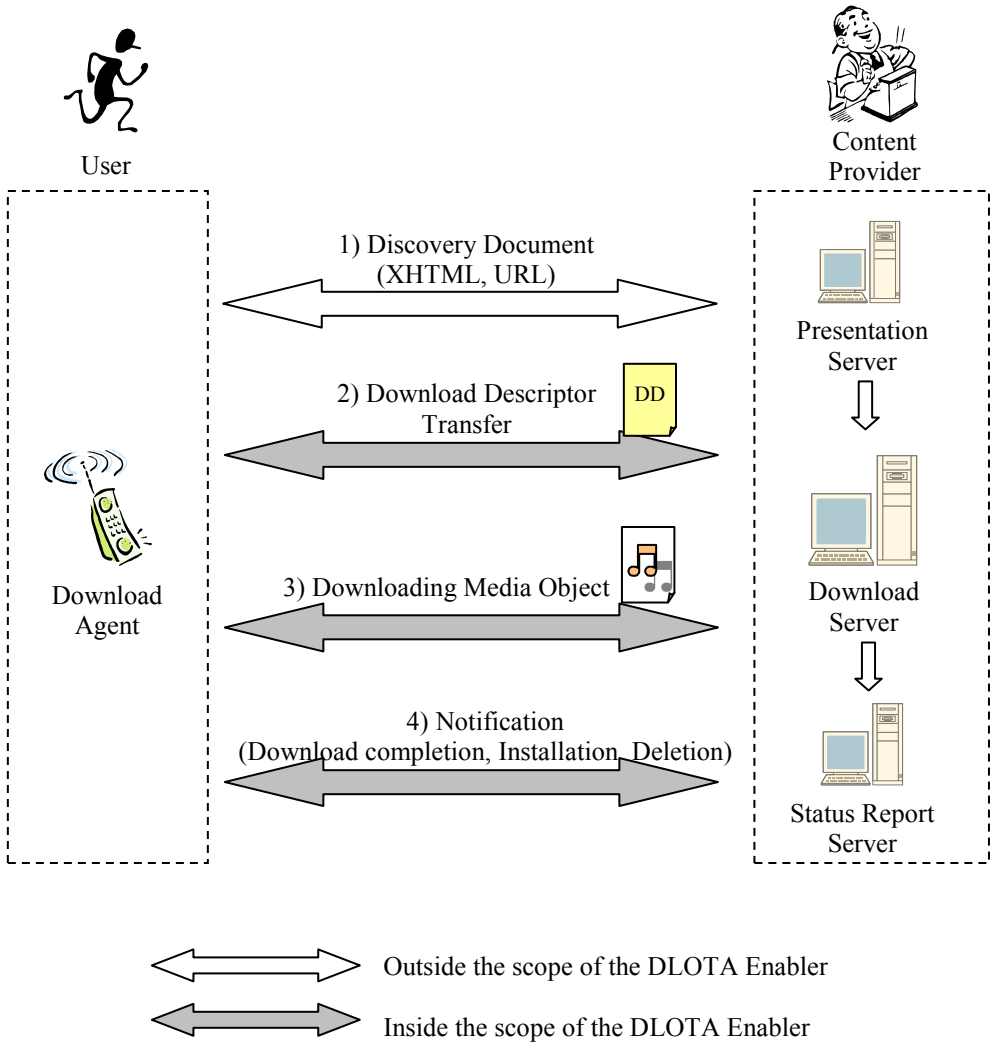

Figure 1: Functional architecture

5.1.1 Minimal Test Configuration

The minimal test configuration is one Server and one Client, interconnected via an IP network (using HTTP as the transport protocol). If the client resides in a mobile phone, a radio network (supporting the type of radio transmission used by the phone) is needed.

- Client settings shall be in accordance with the network parameters provided by the TestFest host.
- Gateway and proxy configuration (if applicable) shall be in accordance with the information provided by the TestFest host for serving all clients participating in the TestFest.
- Configuration of the AP and the RADIUS shall provide RADIUS ACCOUNTING to the gateways/proxies requiring it for MSISDN forwarding. Necessary configuration parameters shall be in advance documented.

5.1.2 Minimal Participation Guidelines

A minimum of **3** clients and **3** servers are needed for meaningful testing at an OMA Test Fest.

5.1.3 Optimal TestFest Achievement Guidelines

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

The list includes:

All ETS Test Cases.

5.2 Enabler Test Requirements

5.2.1 Test Infrastructure Requirements

The following infrastructure elements are required for interoperability testing at a testfest:

- Download Agent(s) Test object
- Download Server(s) Test Object
- Infrastructure to interconnect Agents and Servers. This may be a LAN (for IP access), a radio network with access points etc. depending on the needs of the tested terminals housing the Download Agents.

Optionally, the following equipment may be used:

- MMSC and means to send MMS notifications to the Download Agent (e.g. SMS support)
- Push support for Push of DDs and/or Media objects
- A Broadband transport network
- WAP Gateway (when WSP used as the transport protocol)

For Security tests:

- Security enabled servers

5.2.2 Enabler Execution Flow

5.2.2.1 Basic Download

This use case describes the basic functionality. The User gets a Download Descriptor, downloads a Media Object, and the Status Report Server is notified when the download is complete.

Figure 2 shows an example of the download process of the basic download by using a pull-pull method (see **Error! Reference source not found.** for a description of download methods).

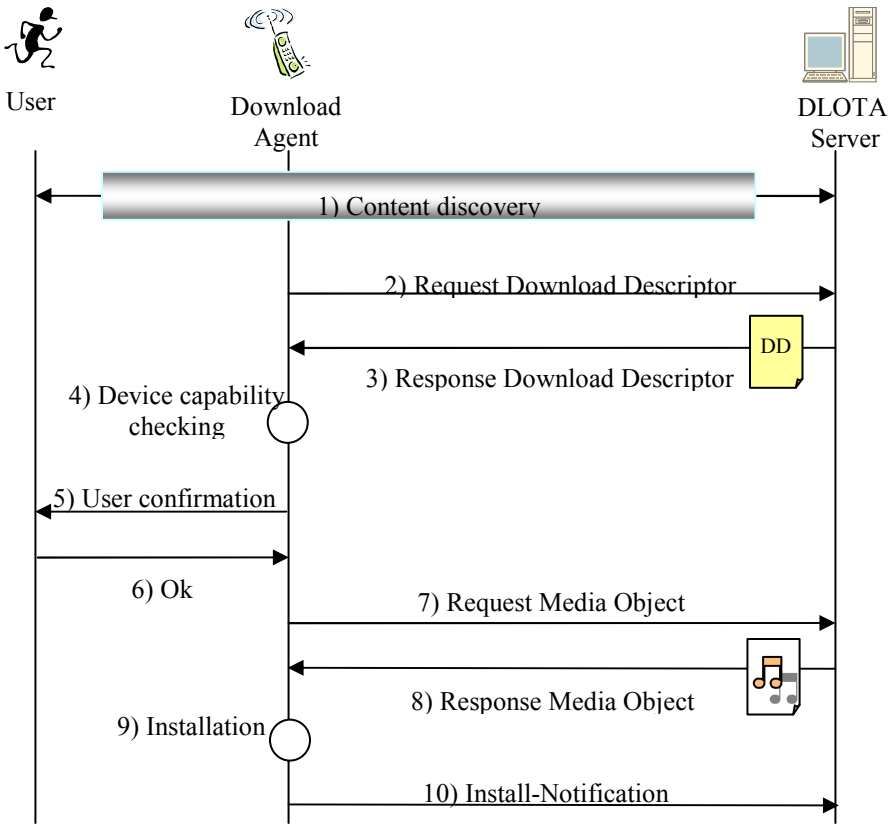

Figure 2: Example Call Flow

1. While using Discovery Application, the User is typically presented with a reference to the Download Descriptor. The reference may be on a Web page, or inside an email or MMS message, or stored in memory or in an accessory attached to the phone. Content discovery and how to find the URI to the Download Descriptor, is outside the scope of DLOTA.
2. The Download Agent selects a URI that points to the Download Descriptor in the DLOTA Server.
3. The Download Descriptor is delivered to the Download Agent. The Download Descriptor includes a URI that points to the Media Object.
4. The Download Agent analyses the Download Descriptor and checks the capability of device (e.g. available memory size, content type of the Media Object, etc.).
5. Using the information included in the Download Descriptor, the Download Agent requests the User to confirm whether to proceed with the download transaction or not.
6. The User decides to proceed with the download transaction.
7. The Download Agent selects the URI that points to the Media Object and the Download Agent proceeds with the download transaction.
8. The Download Agent retrieves the Media Object from the DLOTA Server.
9. The Download Agent installs the Media Object
10. The Download Agent successfully reports the status of the download transaction to the DLOTA Server and makes the content available to the User.

5.2.2.2 DLOTAv2.0 Delivery Methods

Since DLOTA is a bearer independent protocol, the Download Descriptor and the Media Object can be delivered to the Download Agent using any underlying transport protocol such as HTTP [HTTP] and WSP [WSP]. The Download Descriptor and the Media Object can also be delivered using push protocols such as MMS [MMS], WAP Push [WAP Push] and MBMS [MBMS]. Therefore; there are four different scenarios where the Download Descriptor and the Media Object are provided to the Download Agent (Table 1).

Table 1: DLOTAv2.0 Delivery Methods

		Media Object delivery method	
		Pull	Push
DD delivery method	Pull	Case 1: Pull-Pull Scenario	Case 2: Pull-Push Scenario
	Push	Case 3: Push-Pull Scenario	Case 4: Push-Push Scenario

- Case1: Pull-Pull Scenario

  This is the most typical scenario where the Download Descriptor and the Media Object are delivered by using a pull method such as HTTP and WSP.

- Case2: Pull-Push Scenario

  The Download Descriptor is delivered by using a pull method such as HTTP and WSP, and then the Media Object is delivered by using a push method such as WAP Push and MBMS. In MBMS, Media Objects are delivered over a broadcast bearer. This scenario can be cost effective if the size of the Media Object is large and only limited bandwidth is available.

- Case3: Push-Pull Scenario

  The Download Descriptor is pushed to the Download Agent, and then the Media Object is delivered by using a pull method such as HTTP and WSP. This scenario can used for the situation where the content provider wants to notify the end User that a Media Object is available for download, User, upon reception of the Download Descriptor the User can then decide to download the Media Object using a pull method to download it.

- Case4: Push-Push Scenario

  The Download Descriptor and the Media Object are pushed to the Download Agent. This scenario is often used if the network bearer only has a unidirectional bearer. MBMS is often used over the unidirectional bearer.

5.2.3 Test Content Requirements

Reference content should be used in the tests. This can be downloaded from the OMA test server:
(http://testfest.openmobilealliance.org)

5.2.4 Test Limitations

5.2.4.1 Physical

None.

5.2.4.2 Resources

None.

5.2.5 Test Restrictions

5.2.5.1 Dependencies

OMA DLOTA version 2.0 provides a flexible mechanism for downloading Media Objects of any type and size from a network. Therefore, OMA DLOTA version 2.0 can be used by the other OMA enablers to download Media Objects. Examples of such enablers are OMA Digital Rights Management [DRM1.0] [DRM2.0] and Device Management [OMADM].

On the other hand, since OMA DLOTA version 2.0 is a bearer independent protocol, any underlying protocols such as HTTP [HTTP] and WSP [WSP] can be used. Security protocols such as SSL [SSL]/TLS [TLS] or WTLS [WTLS] can also be used to download Media Objects in a secure manner. If a broadcast network such as MBMS [MBMS] and DVB-H [DVB-H] is used to download Media Objects, OMA DLOTA version 2.0 may need to rely on OMA BCAST enabler release [BCAST] for service discovery, content protection and so on.

An example of the relationship between the enablers is illustrated in Figure 3.

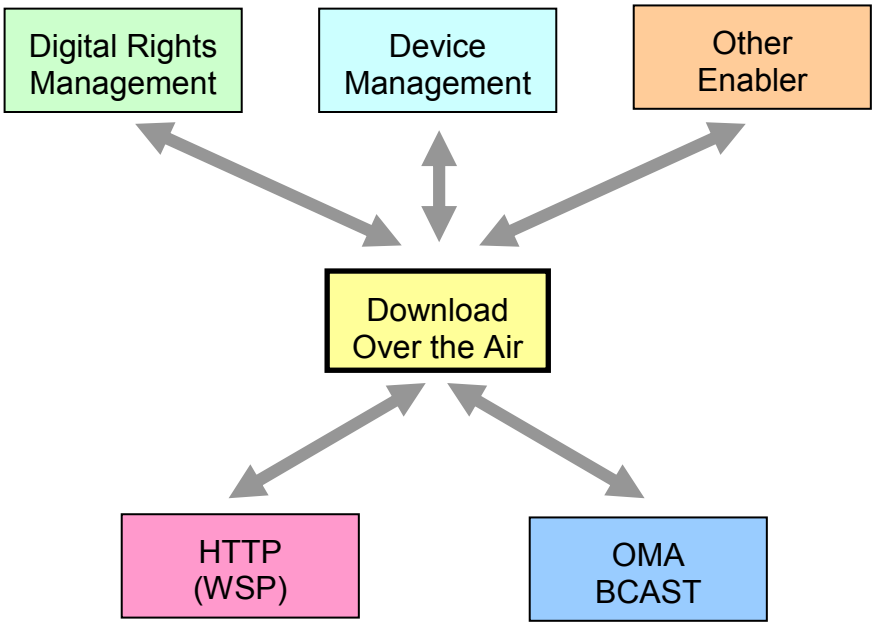

Figure 3: Relationship between OMA Download OTA and other OMA Enablers

5.2.6 Test Tools

5.2.6.1 Existing Tools to be Used

A test tool is available for Conformance testing of DLOTA 2.0 Clients. The tool consists of test cases on the OMA test server (http://testfest.openmobilealliance.org), which can be accessed by all OMA members.

The use of this tool is COMPULSORY for DLOTA 2.0 Clients, where test cases exists and are applicable.

No tool exists for Conformance tests of DLOTA 2.0 Servers.

5.2.6.2 Test Tool Requirements

None.

5.2.7 Resources Required

One tester should be able to run all interoperability test cases within the 3 hour time slot provided at the test fest.

5.3 Tests to be Performed

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

5.3.1 Entry Criteria for TestFest

The following tests need to be performed and passed by implementations by members wishing to participate in the TestFest. This ensures minimal requisite capability of the implementations. The tests are defined in the ETS [IOPETS] and any special comments are noted.

The following sub set of test cases from the ETS should be used in order to verify connectivity setup:

1. TC "Separate Delivery with Status Report" (6.2.3 in [IOPETS])

If possible, the set of Conformance test cases in the ETS should be run. For Client tests, subsets of these are available on the OMA server (http://testfest.openmobilealliance.org) and can be run prior to arriving at the test fest. Alternatively, they can be run as part of the test fest pre-test.

For Clients, if conformance test cases cannot be run, and for Servers, interoperability test cases testing mandatory SCRs should be run as a pre-test. A subset of these constitutes a set, which could be used during the first part of the TestFest.

5.3.2 Testing to be Performed at TestFest

The following tests need to be performed to fully cover the range of capabilities of the enabler and defined protocols. These tests are to be covered in the TestFest. The tests are defined in the ETS [XXXETS] and any special comments are noted.

Test Case Id	Special Conditions
DL-OTA-2.0-con-001	• Download Agent must be able to download the download descriptor. • Download Server MUST include installSize element in the DD. • Download Agent device has sufficient storage resources – at least installSize number of bytes free • Download Agent is capable of handling Media Object's type specified in type element
DL-OTA-2.0-con-002	• The user device is almost full, lack of capability. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-con-003	• The Download Descriptor and the Media Object are delivered separately. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the Download Descriptor. • In the Download Descriptor suppressUserConfirmation element MUST BE equal to 'Never'.

Test Case Id	Special Conditions
DL-OTA-2.0-con-004	• The Download Descriptor and the Media Object are delivered separately. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the Download Descriptor. • In the Download Descriptor suppressUserConfirmation element MUST BE equal to 'Always'. • The Download Server MUST BE authorized by the Download Agent.
DL-OTA-2.0-con-005	• The Download Descriptor and the Media Object are delivered separately. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the Download Descriptor. • In the Download Descriptor suppressUserConfirmation element MUST BE equal to 'Never'
DL-OTA-2.0-con-006	• The Media Object MUST BE executable content (i.e., content that needs to be installed or rendered to be used). • InstallNotifyURI element is included in the Dowload Descriptor for sending the Server the Install Notification. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the Download Descriptor.
DL-OTA-2.0-con-007	• The client device must be able to download the download descriptor.
DL-OTA-2.0-con-008	• The client device must be able to download the download descriptor.
DL-OTA-2.0-con-009	• There is NO server configured in the White List of the client
DL-OTA-2.0-con-010	• The used server IS configured in the White List of the client
DL-OTA-2.0-con-011	• The Client supports handling DDs that comes in an MMS
DL-OTA-2.0-con-012	• A newer version of the Media Object than the version present in the client is available on to the server. • The InstallNotifyURI element is included in the Download Descriptor • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-con-013	• The Content Portal knows the servers where the Media Objects are hosted. • The Content Portal creates the Donwload Descriptor, with the same Media Objects located in different Donwload Servers. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to every server the Download Notification. • The content portal MUST include both objectID and object Version elements in the media Object element of the DD.
DL-OTA-2.0-con-014	• `objectURI` element is included in the Download Descriptor for the Download Agent to locate the Media Objects. • Multiple `server` elements are included in the `objectURI` element of the Download Descriptor. • Media Objects must be available in all specified Servers.

Test Case Id	Special Conditions
DL-OTA-2.0-con-015	• Download Descriptor has a 'license' element for the Media Object. • Download Descriptor has an 'order' element equal to 'post'. • InstallNotifyURI element is included in the Download Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Download Descriptor for sending to the Server the Download Notification. • DRM Agent is present in the Terminal and it can successfully retrieve licenses. • DRM Agent communicates the status of the license retrieval to the Download Agent.
DL-OTA-2.0-con-016	• Download Descriptor has a 'license' element for the Media Object. • Download Descriptor has an 'order' element equal to 'post'. • DownloadNotifyURI is included in the Download Descriptor for sending to the Server the Download Notification. • DRM Agent is present in the Terminal and it can successfully retrieve licenses. • DRM Agent communicates the status of the license retrieval to the Download Agent.
DL-OTA-2.0-con-017	• Media Object or Product includes `license` element. • Device supports OMA DRMv2 and functionality associated to `license` element. • Both License element and Media Object are retrieved in a sequential way.
DL-OTA-2.0-con-018	• Media Object or Product includes `license` element. • Device supports OMA DRMv2 and functionality associated to `license` element. • Both License element and Media Object are retrieved in parallel.
DL-OTA-2.0-con-019	• The client device must be able to download the download descriptor. • The client device supports NextURL functionality.
DL-OTA-2.0-con-020	• The client device must be able to download the download descriptor. • The client device supports SuportURI functionality.
DL-OTA-2.0-con-021	• Progressive download is supported by the device, the terminal (Download Agent) can start rendering the content while the download is in progress. • The Dowload Descriptor and the Media Object are delivered separately. • The progressiveDownloadFlag is set to a true value. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.

Test Case Id	Special Conditions
DL-OTA-2.0-con-022	• `The environment` element is included in the Download Descriptor. • The `envtype` attribute of the environment element is supported by the Download Agent. • The `envtype` unambiguously identifies the information set that can be included inside the environment element and the content handler of the Media Object. • The possible values of `envtype`, syntax and semantics of the internal meta data structures depend on separate environment specific standards. • The retrieval of all associated licenses is already done. • `installNotifyURI` element is present in the Download Descriptor. • The Device supports MIDP Content, this is, it is capable of install and execute MIDP Java code.
DL-OTA-2.0-con-101	• The Server contains a complex Product, containing several Media Objects, at least one of which has an associated License • The Download Descriptor used, includes the request for status report (installation notification requested)
DL-OTA-2.0-con-102	• The Download Descriptor does NOT contain a `supressUserConfirmation` element or it is NOT set to 'Always'. • `NextURL` element is NOT included in the Download Descriptor
DL-OTA-2.0-con-103	• The Download Descriptor does NOT contain a `supressUserConfirmation` element or it is NOT set to 'Always'. • The Download Descriptor includes `nextURL` element.
DL-OTA-2.0-con-201	• The client device must be able to download the Download Descriptor. There are seven Download Descriptor files available. Theycontain many of the attributes specified in [DLOTAv2-TS] . • The DDs are taken from the examples in Appendix D of [DLOTAv2-TS] and are slightly modified/corrected
DL-OTA-2.0-con-202	• The Server uses HTTP as a transport protocol when communicating with the Client. • A protocol analysis tool is available and it is possible to analyse the traffic between server and client.
DL-OTA-2.0-con-401	• installSize element is included in the Download Descriptor. • suppressUserConfirmation element is included in the Download Descriptor and is set to 'Always' • Download Agent device has insufficient storage resources to install the Media Object – storage available is less than installSize number of bytes free. Download Server name is present in Download Agent's white list (Download Server can be authorized).
DL-OTA-2.0-con-402	• The client device must be able to download the download descriptor. • The client device is not compatible with the "major" version of the download descriptor, as indicated in the attribute Version (that is a parameter to the attribute Media).

Test Case Id	Special Conditions
DL-OTA-2.0-con-403	• The client device must be able to download the download descriptor. • There is not enough storage space in the client device for the media object i.e. the media object file is bigger than the size of the available memory.
DL-OTA-2.0-con-404	• InstallNotifyURI element is included in the Download Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Download Descriptor for sending to the Server the Download Notification. • The Media Object is a corrupted content making the Terminal to abort the installation process.
DL-OTA-2.0-con-405	• A new version of the Media Object has to be available in to the server. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-con-406	• The media object is available in a different Download Server. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-con-407	• A new version of the Media Object has to be available in to the server. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-con-408	• The client device must be able to download the download descriptor. • The TYPE field in the download descriptor indicates an unknown media object type (e.g. something gibberish).

Test Case Id	Special Conditions
DL-OTA-2.0-con-409	• The Media Object MUST BE executable content (ie., content that needs to be installed or rendered to be used). • The advertised content-type of the Media Object MUST be supported by the Device. • The format of the Media Object MUST BE non-usable with the application in the Device that MUST execute the Media Object. • For example, a given Media Object claims to be a JPEG file (“file.jpg”) and claims to use the proper content-type (“image/jpeg”) so it is accepted by the Device because it has the proper codecs. But that same Media Object is in fact a GIF file (i.e. it uses the encoding used in GIF files) so it will not be usable by the application trying to render a JPEG file from GIF content. • InstallNotifyURI element is included in the Download Descriptor for sending the Server the Install Notification. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the Download Descriptor.
DL-OTA-2.0-con-410	• The client device must be able to download the download descriptor. • The download descriptor contains syntactic errors e.g. o Some of the mandatory attributes are missing (e.g. Type). • The DD file does not comply with the XML Schema (e.g. there are end tags without start tags)
DL-OTA-2.0-con-411	• Progressive download is not supported by the device. • The Media Object has to be completly downloaded to render. • The progressiveDownloadFlag is set to a true value. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-con-412	• The `environment` element is included in the Download Descriptor. • The `envtype` unambiguously identifies the information set that can be included inside the `environment` element and the content handler of the Media Object. • The possible values of `envtype`, syntax and semantics of the internal meta data structures depend on separate environment specific standards • The content handler is unknown for the Client, this is, the Client is not able to use the specified environment. • `installNotifyURI` element is present is the Download Descriptor)

Test Case Id	Special Conditions
DL-OTA-2.0-con-413	• `The environment` element is included in the Download Descriptor. • The `envtype` attribute of the environment element is supported by the Download Agent. • The `envtype` unambiguously identifies the information set that can be included inside the environment element and the content handler of the Media Object. • The possible values of `envtype`, syntax and semantics of the internal meta data structures depend on separate environment specific standards. • The retrieval of all associated licenses is already done. • `installNotifyURI` element is present in the Download Descriptor. • The Device supports the use of the `environment` element but it does NOT support the installation of MIDP Content.
DL-OTA-2.0-con-501	• The Server supports both DLOTA v1 and DLOTAv2 • The server supports the feature to adjust to the version of DLOTA advertised by the client • The test Client supports only DLOTA v1 • The test Client advertises support for only DLOTA v1
DL-OTA-2.0-con-502	• The Server uses HTTP as a transport protocol when communicating with the Client. • A protocol analysis tool is available and it is possible to analyse the traffic between server and client.
DL-OTA-2.0-con-503	• None
DL-OTA-2.0-int-001	• The Server contains a Download Descriptor for DLOTA v1.0
DL-OTA-2.0-int-002	• The Server contains a Download Descriptor for DLOTA v1.0
DL-OTA-2.0-int-003	• The Server contains a Download Descriptor for DLOTA v1.0
DL-OTA-2.0-int-004	• The Server supports both DLOTA v1 and DLOTAv2 • The Client supports both DLOTA v1 and DLOTAv2 • The Client advertises support for both DLOTA v1 and DLOTAv2
DL-OTA-2.0-int-101	• The download descriptor includes InstallNotifyURI
DL-OTA-2.0-int-102	• The Dowload Descriptor and the Media Object are co-delivered. • InstallNotifyURI element is included in the Download Descriptor for sending the Server the Install Notification. • DownloadNotifyURI is included in the Download Descriptor for sending the Server the Download Notification. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-103	• None
DL-OTA-2.0-int-104	• The download descriptor includes DownloadNotifyURI
DL-OTA-2.0-int-105	• The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.

Test Case Id	Special Conditions
DL-OTA-2.0-int-106	• The Download Server has capability of time management functionality, and it gives a list of candidate time for downloading. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to every server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD. • The server includes the donwloadTime element in the DD.
DL-OTA-2.0-int-107	• The Download Server has capability of time management functionality, and it gives a list of candidate time for downloading. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to every server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD. • The server includes the donwloadTime element in the DD.
DL-OTA-2.0-int-108	• The Download Server has capability of time management functionality, and it gives a list of candidate time for downloading. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to every server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD. • The server includes the donwloadTime element in the DD and the reservationNotifyURI.
DL-OTA-2.0-int-109	• The Download Server has capability of time management functionality, and it gives a list of candidate time for downloading. • The Media Object is updated in the server between the user request and the time selected to download.
DL-OTA-2.0-int-110	• A new version of the Media Object has to be available in to the server. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-111	• A new version of the Media Object has to be available in to the server. • The Dowload Descriptor and the Media Object are deliveried in a combined manner. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server SHOULD include the ETag element in DD and the client agent MAY support this perform.

Test Case Id	Special Conditions
DL-OTA-2.0-int-112	• The Content Portal creates the Download Descriptor for the Product Object, with 10 Media Objects • The Download Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Download Descriptor • the installNotifyURI has been defined for the whole Product and not for one of the Media Objects
DL-OTA-2.0-int-113	• The Content Portal creates the Donwload Descriptor for the Compound Object, with several Media Objects of different types. • The user does not mind that the object is compound. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to every server the Download Notification. • The conent portal MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-114	• The Server contains a Download Descriptor with at least 2 Products, each containing several Media Objects • The Download Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Download Descriptor, for each Product • the installNotifyURI has been defined for the whole Products and not for one of the Media Objects within the Products
DL-OTA-2.0-int-115	• The Content Portal knows where are hosted each Media Object. • The Content Portal creates the Donwload Descriptor for the Compound Object, with several Media Objects from different Donwload Servers. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to every server the Download Notification. • The conent portal MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-116	• A new version of the Media Object has to be available in to the server. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server SHOULD include the ETag element in DD and the client agent MAY support this perform.
DL-OTA-2.0-int-117	• The Server supports (HTTP) Chunked delivery and it is enabled on the server. • InstallNotifyURI element is included in the Download Descriptor, • The Media Object is large enough to warrant the use of Chunked Delivery

Test Case Id	Special Conditions
DL-OTA-2.0-int-118	• The Download Server allows the continuation of an interrupted donwload session. • The Donwload Agent supports resume of a download transaction from the point it was interrupted. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-119	• The Download Server allows the continuation of an interrupted donwload session. • The Donwload Agent supports resume of a download transaction from the point it was interrupted. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-120	• The server device must be able to push the download descriptor through a Push Gateway/Proxy. • The user must subscribe the service that push Media Objects to the terminal. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD
DL-OTA-2.0-int-121	• The Server contains a Download Descriptor • The download descriptor includes InstallNotifyURI • The client supports WSP • The client is configured to use a WAP Gateway
DL-OTA-2.0-int-122	• The Server contains a Download Descriptor • The download descriptor includes InstallNotifyURI • The client supports WTLS, class 1 • The client is configured to use a WAP Gateway
DL-OTA-2.0-int-123	• The Server contains two Download Descriptors that can be sent together • Both download descriptors includes InstallNotifyURI
DL-OTA-2.0-int-124	• The Client supports handling DDs that comes in an WAP Push • The Server can send DDs in WAP Push messages

Test Case Id	Special Conditions
DL-OTA-2.0-int-201	• The Download Descriptor used, includes the request for status reports: o `reservationNotiryURI` element ▪ "DownloadTime" is set to 5 minutes into the future. • `installNotifyURI` element
DL-OTA-2.0-int-202	• The Download Descriptor used, includes the request for status reports: • `deleteNotifyURI` element
DL-OTA-2.0-int-203	• DownloadNotifyURI is included in the Download Descriptor for sending the Server the Download Notification.
DL-OTA-2.0-int-301	• The Dowload Descriptor and the Media Object are co-delivered. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the Server the Download Notification. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the DD. • The Download Server initiates authentication within a HTTP session by sending a HTTP Unauthorized message
DL-OTA-2.0-int-302	• The Server must accept the https:// prefix as a means to set up a TLS connection • The Server certificate is downloaded into the client before the test case is executed
DL-OTA-2.0-int-303	• DownloadNotifyURI is included in the Download Descriptor for sending the Server the Download Notification. • The Client and Server MUST support Server authentication. • A TLS tunnel MUST be established by the Client using a HTTP CONNECT method. • A TLS connection MUST be established with a TLS message exchange prior the Media Object is requested. • The Download Server MUST implement the following cipher suites in coding the Download Descriptor and the Media Object: TLS_RSA_WITH_3DES_EDE_CBC_SHA. • The Download Agent MUST implement the following cipher suites: TLS_RSA_WITH_3DES_EDE_CBC_SHA. • Note: As a TLS connection is used for Server Authentication, the authenticated download must also be confidentiality and integrity protected.

Test Case Id	Special Conditions
DL-OTA-2.0-int-304	• DownloadNotifyURI is included in the Download Descriptor for sending the Server the Download Notification. • The Download Descriptor and the Media Object are co-delivered. • The Client and Server MUST support Server authentication. • A TLS tunnel MUST be established by the client using a HTTP CONNECT method. • A TLS connection MUST be established with a TLS message exchange prior the Media Object was requested. • The Download Server MUST implement the following cipher suites in coding the Download Descriptor and the Media Object: TLS_RSA_WITH_AES_128_CBC_SHA [RFC3268]. • The Download Server SHOULD implement the following cipher suites: TLS_RSA_WITH_AES_128_CBC_SHA [RFC3268]. • Note: As a TLS connection is used for Server Authentication, the authenticated download must also be confidentiality and integrity protected.
DL-OTA-2.0-int-305	• The Download Descriptor and the Media Object are co-delivered. • InstallNotifyURI element is included in the Download Descriptor for sending the Server the Install Notification. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the Download Descriptor. • The Download Agent MUST send an HTTP request with a constant string “3gpp-gba” in the HTTP header. • The Download Server MUST send the HTTP WWW-Authenticate message with a constant string “3gpp-bootstraping” in the HTTP header.
DL-OTA-2.0-int-306	• A new version of the Media Object has to be available in to the server. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD. • Download Agent and Download Server MUST support HTTP and TLS protocols. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification.
DL-OTA-2.0-int-307	• A new version of the Media Object has to be available in to the server. • The Dowload Descriptor and the Media Object are delivered separately. • Download Agent and Download Server MUST support HTTP and TLS protocols. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-401	• The client device must be able to download the download descriptor.

Test Case Id	Special Conditions
DL-OTA-2.0-int-402	• A new version of the Media Object has to be available in to the server. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-403	• The user must subscribe the service that push Media Objects to the terminal. • The server device must be able to push the Download Descriptor through a Push Gateway/Proxy. • The Media Object is not available on the Donwload Server, but the Download Descriptor MUST send to the terminal. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the server the Download Notification. • The server MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-404	• The Content Portal creates the Donwload Descriptor for the Compound Object, with several Media Objects from different Donwload Servers. • One Single Object is missed in one of the Servers. • The user do not mind that the object is compound. • The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Content Portal the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending to every server the Download Notification. • The conent portal MUST include both objectID and objectVersion elements in the mediaObject element of the DD.
DL-OTA-2.0-int-405	• The Dowload Descriptor and the Media Object are delivered separately. • InstallNotifyURI element is included in the Download Descriptor for sending the Server the Install Notification. • DownloadNotifyURI is included in the Download Descriptor for sending the Server the Download Notification. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the DD. • The Content-Type parameter in the transport protocol MUST NOT be included.
DL-OTA-2.0-int-406	• The Dowload Descriptor and the Media Object are co-delivered. • InstallNotifyURI element is included in the Dowload Descriptor for sending the Server the Install Notification. • DownloadNotifyURI is included in the Dowload Descriptor for sending the Server the Download Notification. • The Server MUST include both objectID and objectVersion elements in the mediaObject element of the DD. • The Download Server initiates authentication with in a HTTP session by sending a HTTP Unauthorized message.

Table 2: Listing of Tests to be Performed at TestFest

5.4 Enabler Test Reporting

5.4.1 Problem Reporting Requirements

Normal Reporting, no special reporting required.

5.4.2 Enabler Test Requirements

Normal Reporting, no special reporting required.

6. Alternative Validation Activities

None.

7. Approval Criteria

The DLOTA 2.0 Enabler can be put in the Approved state when:

- The Enabler has been tested successfully , meaning that:
  - Test cases testing Mandatory features have been run >= 9 times each.
  - A set of test cases that together test > 80% of all Mandatory SCRs and 40% of all Optional SCRs have been run. A minimum of 60% of these test cases yielded the result “PASS”.

No open PRs exist.

Appendix A. Change History (Informative)

A.1 Approved Version History

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

A.2 Draft/Candidate Version 2.0 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-EVP-DLOTA-V2.0	14 Feb 2008	All	First version (converted ETG to EVP) CR implemented: OMA-IOP-BRO-2007-0185
Candidate Versions OMA-EVP-DLOTA-V2.0	31 Mar 2008	n/a	Prepared for TP Candidate approval TP# OMA-TP-2008-0123-INP_DLOTA_2.0_EVP_for_Candidate_approval