Device Management Meta Information

Candidate Version 1.3 – 25 May 2010

Open Mobile Alliance

OMA-TS-DM_MetaInfo-V1_3-20100525-C

Contents

1. Scope

This document outlines the DM Meta Information Specification and the respective conformance requirements for DM Clients and DM Servers.

2. References

2.1 Normative References

[DMREPRO] "DM Representation Protocol", Open Mobile Alliance™, OMA-TS-DM_RepPro-V1_3, URL:http://www.openmobilealliance.org/

[IOPPROC] "OMA Interoperability Policy and Process", Version 1.9, Open Mobile Alliance™, OMA-IOP-Process-V1_9, URL:http://www.openmobilealliance.org/

[ISO8601] "Data elements and interchange formats -- Information interchange -- Representation of dates and times", ISO 8601:2000, URL:http://www.iso.ch

[RFC2045] "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", N. Freed & N. Borenstein, November 1996, URL:http://www.ietf.org/rfc/rfc2045.txt

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt

[SYNCMLMETA] "SyncML Meta Information", Open Mobile Alliance™, OMA-TS-SyncML_Meta-V1_2, URL:http://www.openmobilealliance.org/

[WBXML] "WAP Binary XML Content Format Specification", WAP Forum™, WAP-154-WBXML, URL:http://www.openmobilealliance.org/

[XML] "Extensible Markup Language (XML) 1.0", World Wide Web Consortium Recommendation, URL:http://www.w3.org/TR/REC-xml

2.2 Informative References

[METADTD] "SyncML Meta Information, Document Type Definition", Open Mobile Alliance™, OMA-TS-SyncML_MetaInfo_DTD-V1_3, URL:http:www.openmobilealliance.org/

3. Terminology and Conventions

3.1 Conventions

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

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

Any reference to components of the DM DTD or XML snippets is specified in this `typeface`.

3.2 Definitions

Meta-Information	Parameter or attributes about the representation, state or type or content of an object or property.
Parser	Refers to an XML parser. An XML parser is not and absolutely needed to support SyncML. However, a SyncML implementation that integrates an XML parser might be easier to enhance. This document assumes that the reader has some familiarity with XML syntax and terminology.
Recipient	The network device that receives a SyncML request, processes the request and sends any resultant SyncML response.
Representation protocol	A well-defined format for exchanging a particular form of information. SyncML is a representation protocol for conveying data synchronization and device management operations.
Request	A message or a command sent from a device to another.

3.3 Abbreviations

DTD	Document Type Definition
EMI	Experimental Meta Information
OMA	Open Mobile Alliance
URI	Universal Resource Identifier
URN	Universal Resource Name
WBXML	WAP Binary XML
XML	Extensible Markup Language

4. Introduction

The meta-information associated with a DM command or data item is represented in a mark-up language defined by [XML]. The meta-information is identifiable as an XML name space. The DM Meta-Information DTD (Document Type Definition) defines the XML document type used to represent meta-information used in the DM representation protocol [DMREPRO].

5. Meta Information

5.1 XML Usage

The DM Meta-Information XML documents are specified using well-formed XML. However, they need not be valid XML. That is, they do not need to specify the XML prolog. They only need to specify properly identified name space element types from the DM Meta-Information DTD. This restriction allows for the DM Meta-Information to be specified with greater terseness than would be possible if a well-formed, valid XML document was REQUIRED.

This DTD makes heavy use of XML name spaces. Name spaces MUST be declared on the first element type that uses an element type from the name space. Element types from the DM Meta-Information DTD can be used in other XML documents, including a DM Message.

Names in XML are case sensitive. By convention in the DM Meta-Information DTD, the element type and attribute list names are specified with a "Hungarian" like notation of the first character in each word of the name in upper case text and remainder of the characters in each word of the names specified in lower case text. For example, `MetInf` for the Sync meta-information root element type or `Type` for the content type tag.

The element types in the DM Meta-Information DTD [METADTD] are defined within a namespace associated with the URI or the URN `syncml:metinf`.

DM also makes use of XML standard attributes, such as `xml:lang`. Any XML standard attribute can be used in a XML document conforming to this DTD.

XML can be viewed as more verbose than alternative binary representations. This is often cited as a reason why it might not be appropriate for low bandwidth network protocols. In most cases, this DTD uses shortened element type and attribute names. This provides a minor reduction in verbosity. Additionally, the DM Meta-Information can be encoded in a tokenized, binary format defined by [WBXML]. The use of [WBXML] format is external to specification of the DTD and transparent to any XML application

One of the main advantages of XML is that it is a widely accepted International recommendation for text document mark-up. It provides for both human readability and machine processability. In addition, XML allows the originator to capture the structure of a document, not just its content. This is extremely useful for applications such as data synchronization, where not just content, but structure semantics is often exchanged.

5.2 Element Type Descriptions

5.2.1 Anchor

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.2 EMI

Usage: Specifies the non-standard, experimental meta information (EMI) extensions supported by the device. The extensions are specified in terms of the XML element type name and the value.

Parent Elements: `MetInf`

Restrictions: The EMI element type MUST specify the extension element name. It MAY also specify one or more enumerated values. Multiple non-standard extensions can be specified by specifying the EMI element type multiple times. This element type is OPTIONAL.

Content Model:

```
(#PCDATA)
```

Attributes: None.

Example: The following example specifies a non-standard extension with a value of "Confidential".

```
<EMI xmlns='syncml:metinf'>Confidential</EMI>
```

5.2.3 FieldLevel

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.4 Format

Usage: Specifies the encoding format of the content information in the Data element.

Parent Elements: `MetInf`

Restrictions: The value of this element SHOULD be one of bin, bool, b64, chr, int, node, null, xml, date, time, or float. If the element type is missing, the default value is chr. If the value is bin, then the format of the content is binary data. If the value is bool, then the encoding of the content is either case sensitive "true" or case sensitive "false." If the value is b64, then the format of the content information is binary data that has been character encoded using the Base64 transfer encoding defined by [RFC2045]. If the value is chr, then the format of the content information is clear-text in the character set specified on either the transport protocol, the MIME content type header or the XML prolog. If the value is int, then the format of the content information is numeric text representing the integer. If the value is node, then the content represents an interior object in the management tree. If the value is null, then there is no content information. This value is used by some synchronization data models to delete the content, but not the presence of the property. If the value is xml, then the format of the content information is XML structured mark-up data. If the value is date, then the format of the content is in ISO 8601 format with the century being included in the year [ISO8601]. If the value is time, then the format of the content is in ISO 8601 format. If the value is float, then the format of the content is standard concept of real numbers corresponding to a single precision 32 bit floating point type as defined in XML Schema 1.0 as the `float` primitive type [XMLSCHEMADT].

In case a `Meta` element containing a `Format` element contains meta-information about a `Data` object, this `Meta` element MUST have the same parent as the `Data` object it refers to.

The target object is the one in which the meta-information appears.

Content Model:

```
(#PCDATA)
```

Attributes: None.

Example: The following example illustrates how the element type is used within the DM DTD to specify format meta-information for data in the Item element type.

```
<Item>
  <Meta>
    <Format xmlns='syncml:metinf'>int</Format>
  </Meta>
  <Data>1024</Data>
</Item>
```

5.2.5 FreeID

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.6 FreeMem

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.7 Last

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.8 Mark

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.9 MaxMsgSize

Usage: Specifies the maximum byte size of any response message to a given SyncML request.

Parent Elements: `MetInf.`

Restrictions: The element type appears in the Meta element in the SyncHdr of a SyncML request to specify the maximum size of any subsequent response messages. The element type is usually specified by a SyncML client, but can also be specified by a SyncML server.

This element type value is applicable for the remainder of the synchronization session, unless it is specified again.

The element type value is the text string representation of the maximum, decimal byte size of any response message.

In order to use the elements from the MetInf name space, the root element does not need to be specified.

Content Model:

```
(#PCDATA)
```

Attributes: None.

Example: Normally, the root element type does not appear in a SyncML Meta element type.

```
<MaxMsgSize xmlns='syncml:metinf'>1023</MaxMsgSize>
```

5.2.10 MaxObjSize

Usage: Specifies the maximum size in bytes of a data object that the device is able to receive.

Parent Elements: `MetInf.`

Restrictions: The element type appears in the Meta element of a SyncML request to specify the maximum size of the largest object it is capable of receiving in any subsequent response messages. This element type value is applicable for the remainder of the synchronization session.

The element type value is the text string representation of the maximum, decimal byte size without leading zeroes of any object.

In order to use the elements from the MetInf name space, the root element does not need to be specified.

Content Model:

```
(#PCDATA)
```

Attributes: None.

Example: Device that can receive a maximum object of 10K bytes.

```
<MaxObjSize xmlns='syncml:metinf'>10240</MaxObjSize>
```

5.2.11 Mem

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.12 MetInf

Usage: Specifies the root element for the SyncML meta-information document.

Parent Elements: `Root element type.`

Restrictions: In order to use the elements from the MetInf name space, the root element does not need to be specified. The element type can appear in the Meta element of a SyncML document to allow for declaring a default name space.

Content Model:

```
MetInf (FieldLevel?, Format?, Type?, Size?, Anchor?, Version?, NextNonce?,
MaxMsgSize?, MaxObjSize?, EMI*, Mem?)
```

Attributes: None.

Example: Normally, the root element type does not appear in a SyncML Meta element type.

```
<MetInf xmlns='syncml:metinf'>
  <FieldLevel/>
  <Type xmlns='syncml:metinf'>x-type/x-subtype</Type>
  <Format xmlns='syncml:metinf'>chr</Format>
  <Size xmlns='syncml:metinf'>877566</Size>
  <Version xmlns='syncml:metinf'>20000714T082300Z</Version>
</MetInf>
```

5.2.13 Next

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.14 NextNonce

Usage: Specifies the nonce string to be used in any subsequent communication.

Parent Elements: `MetInf`

Restrictions: The nonce string MUST be further re-formatted using the Base64 algorithm. Terminators or length of Nonce String MUST NOT be included in this re-formatting. The Nonce string MUST be treated as opaque data.

This element type is used to specify the next nonce string that is to be used in any subsequent SyncML message. For example, a SyncML server specifies this element type to tell the SyncML client to change its nonce to a new value.

Nonce strings are used in the SyncML "MD5 Digest" scheme of authentication credentials.

Content Model:

```
(#PCDATA)
```

Attributes: None.

Example:

```
<Meta>
  <NextNonce
xmlns='syncml:metinf'>QWxhZGRpbjpvcGVuIHNlc2FtZQ==</NextNonce> </Meta>
```

5.2.15 SharedMem

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

5.2.16 Size

Usage: Specifies the byte size of a data object.

Parent Elements: `MetInf`

Restrictions: The byte size is specified as the numeric text equivalent of the byte count of the data object. In case a `Meta` element containing a `Size` element contains meta-information about a `Data` object, this `Meta` element MUST have the same parent as the `Data` object it refers to

Content Model:

```
(#PCDATA)
```

Attributes: None

Example: The following example illustrates how the element type is used within the SyncML DTD to specify meta-information about the byte size of the Item element type.

```
<Item>
  <Target><LocURI>4</LocURI>
  <Meta>
    <Size xmlns='syncml:metinf'>10</Size>
  </Meta>
  <Data>John Smith</Data>
</Item>
```

5.2.17 Type

Usage: Specifies the media type of the content information in the Data element.

Parent Elements: `MetInf`

Restrictions: If this element is missing, then the default content-type is text/plain. The content information for this element type SHOULD BE a registered MIME content-type. Alternatively, a URN can be used to specify the media type. In case a `Meta` element containing a `Type` element contains meta-information about a `Data` object, this `Meta` element MUST have the same parent as the `Data` object it refers to.

Content Model:

```
(#PCDATA)
```

Attributes: None

Example: The following example illustrates how the element type is used within a SyncML message to specify meta-information about the media type of the content information in the Item element type.

```
<Item>
  <Target><LocURI>3</LocURI></Target>
  <Meta>
    <Type xmlns='syncml:metinf'>text/directory;profile=vCard</Type>
  </Meta>
  <Data>BEGIN:VCARD
VERSION:3.0 FN:Jim Smith N:Smith;Jim TEL;TYPE=WORK,VOICE,FAX:+1-919-555-
1234 EMAIL;TYPE=INTERNET,WORK:Jim_Smith@mail.host.com
END:VCARD
  </Data>
</Item>
```

5.2.18 Version

Restrictions by DM: This element is not used in OMA Device Management Protocol. Please refer to [SYNCMLMETA] for detail information.

6. DTD Definition

```
<!--
OMA DM Meta Information (SYNCML-METINF) V1.2 Document Type


This DTD defines a sequence of meta-information that is used within
the DM Representation Protocol DTD. Typical usage:
   <!DOCTYPE MetInf PUBLIC "-//OMA//DTD SYNCML-METINF 1.2//EN"
             "http://www.openmobilealliance.org/tech/DTD/OMA-TS-
DM_MetaInfo_DTD-V1_3.dtd"
             [<?oma-syncml-metinf-ver supported-versions="1.2"?>]>
   <element>
      ...
   </element>


Terms and conditions of use are available from the
Open Mobile Alliance Ltd. web site at
http://www.openmobilealliance.org/useterms.html
-->


<!-- Root Element -->
<!ELEMENT MetInf (FieldLevel?, Format?, Type?, Mark?, Size?, Anchor?,
Version?, NextNonce?, MaxMsgSize?, MaxObjSize?, EMI*, Mem?)>
<!-- FieldLevel change flag -->
<!ELEMENT FieldLevel EMPTY>
<!-- Format or encoding type -->
<!ELEMENT Format (#PCDATA)>
<!-- Element specific type specification -->
<!ELEMENT Type (#PCDATA)>
<!-- Mark -->
<!ELEMENT Mark (#PCDATA)>
<!-- Byte count -->
```

```
<!ELEMENT Size (#PCDATA)>
<!-- Data versioning info -->
<!ELEMENT Anchor (Last?, Next)>
<!ELEMENT Last (#PCDATA)>
<!ELEMENT Next (#PCDATA)>
<!ELEMENT Version (#PCDATA)>
<!ELEMENT NextNonce (#PCDATA)>
<!ELEMENT MaxMsgSize (#PCDATA)>
<!ELEMENT MaxObjSize (#PCDATA)>
<!-- Experimental Meta Information extension -->
<!ELEMENT EMI (#PCDATA)>
<!-- Dynamic Memory -->
<!ELEMENT Mem (SharedMem?, FreeMem, FreeID)>
<!-- Free Memory in the number of identifiers -->
<!ELEMENT FreeID (#PCDATA)>
<!-- Free Memory in bytes -->
<!ELEMENT FreeMem (#PCDATA)>
<!-- Shared Memory -->
<!ELEMENT SharedMem EMPTY>
<!-- End of DTD -->
```

7. WBXML Definition

The following tables define the token assignments for the mapping of the Meta-Information DTD element types into WBXML as defined by [WBXML].

7.1 Code Space and Code Page Definitions

This version of the SyncML representation protocol specification maps all the SyncML related DTDs into WBXML code spaces.

DTD Name	WBXML PUBLICID Token (Hex Value)	Formal public Identifier
SyncML1.2	0x01201	-//SYNCML//DTD SyncML 1.2/EN

The SyncML DTD is assigned the WBXML document public identifier (i.e., the "publicid" WBXML BNF production) associated with the `0x1201 t`oken.

The element types from the Meta-Information DTD utilize the code page x01 (one) within the SyncML Code Space.

DTD Name	WBXML Code Page Token (Hex Value)	Formal Public Identifier
SyncML	00	-//SYNCML//DTD SyncML 1.2//EN
MetaInf	01	-//SYNCML//DTD MetInf 1.2//EN

7.2 Token Definitions

The following WBXML token codes represent element types (i.e., tags) form code page x01 (one), Meta-Information DTD.

Element Type Name	WBXML Tag Token (Hex Value)
Anchor	05
EMI	06
Format	07
FreeID	08
FreeMem	09
Last	0A
Mark	0B
MaxMsgSize	0C
Mem	0D
MetInf	0E
Next	0F
NextNonce	10
SharedMem	11
Size	12
Type	13
Version	14
MaxObjSize	15

FieldLevel	16

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
N/A	N/A	No prior 1.3 version

A.2 Draft/Candidate Version 1.3 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS-SyncML_MetaInfo-V1_3	27 Aug 2009	All	New baseline using OMA-TS-SyncML-MetaInfo-V1_2_1
	25 Nov 2009	All	Applied OMA-DM-DM13-2009-0099R02-CR_MetaInfo_cleanup
Draft Versions OMA-TS-DM_MetaInfo-V1_3	08 Jan 2010	All	General editorial clean-up by DSO
	04 May 2010	5.1, App B	Correction of a mispelling error. Fixed one reference in App B
	05 May 2010	All	Changed snippets front from Courrier to Courrier New to harmonize with the rest f the document Fixed broken cross-references
Candidate Version OMA-TS-DM_MetaInfo-V1_3	25 May 2010	N/A	Status changed to Candidate by TP Ref # OMA-TP-2010-0221-INP_DM_V1.3_ERP_and_ETR_for_Candidate_approval

Appendix B. Static Conformance Requirements (Normative)

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

B.1 Client Meta Information

Item	Function	Ref.	Status	Requirement
DSDM-METINF-C-001	Support for Anchor element	5.2.1	M	DSDM-METINF-C-007 AND DSDM-METINF-C-013
DSDM-METINF-C-002	Support for EMI element	5.2.2	O	
DSDM-METINF-C-003	Support for FieldLevel element	5.2.3	O	
DSDM-METINF-C-004	Support for Format element	5.2.4	M	
DSDM-METINF-C-005	Support for FreeID element	5.2.5	O	
DSDM-METINF-C-006	Support for FreeMem element	5.2.6	O	
DSDM-METINF-C-007	Support for Last element	5.2.7	M	
DSDM-METINF-C-008	Support for Mark element	5.2.8	O	
DSDM-METINF-C-009	Support for MaxMsgSize element	5.2.9	O	
DSDM-METINF-C-010	Support for MaxObjSize element	5.2.10	O	
DSDM-METINF-C-011	Support for Mem element	5.2.11	O	DSDM-METINF-C-005 AND DSDM-METINF-C-006 AND DSDM-METINF-C-015
DSDM-METINF-C-012	Support for MetInf element	5.2.12	M	DSDM-METINF-C-001 AND DSDM-METINF-C-004 AND DSDM-METINF-C-014 AND DSDM-METINF-C-017 AND DSDM-METINF-C-018
DSDM-METINF-C-013	Support for Next element	5.2.13	M	
DSDM-METINF-C-014	Support for NextNonce element	5.2.14	M	
DSDM-METINF-C-015	Support for SharedMem element	5.2.15	O	
DSDM-METINF-C-016	Support for Size element	5.2.16	O	
DSDM-METINF-C-017	Support for Type element	5.2.17	M	
DSDM-METINF-C-018	Support for Version element	5.2.18	M	

B.2 Server Meta Information

Item	Function	Ref.	Status	Requirement
DSDM-METINF-S-001	Support for Anchor element	5.2.1	M	DSDM-METINF-S-007 AND DSDM-METINF-S-013
DSDM-METINF-S-002	Support for EMI element	5.2.2	O	
DSDM-METINF-S-003	Support for FieldLevel element	5.2.3	O	
DSDM-METINF-S-004	Support for Format element	5.2.4	M	
DSDM-METINF-S-005	Support for FreeID element	5.2.5	M	
DSDM-METINF-S-006	Support for FreeMem element	5.2.6	M	
DSDM-METINF-S-007	Support for Last element	5.2.7	M	
DSDM-METINF-S-008	Support for Mark element	5.2.8	O	
DSDM-METINF-S-009	Support for MaxMsgSize element	5.2.9	M	
DSDM-METINF-S-010	Support for MaxObjSize element	5.2.10	M	
DSDM-METINF-S-011	Support for Mem element	5.2.11	M	DSDM-METINF-S-005 AND DSDM-METINF-S-006 AND DSDM-METINF-S-015
DSDM-METINF-S-012	Support for MetInf element	5.2.12	M	DSDM-METINF-S-001 AND DSDM-METINF-S-004 AND DSDM-METINF-S-009 AND DSDM-METINF-S-010 AND DSDM-METINF-S-011 AND DSDM-METINF-S-014 AND DSDM-METINF-S-017 AND DSDM-METINF-S-018
DSDM-METINF-S-013	Support for Next element	5.2.13	M	
DSDM-METINF-S-014	Support for NextNonce element	5.2.14	M	
DSDM-METINF-S-015	Support for SharedMem element	5.2.15	M	
DSDM-METINF-S-016	Support for Size element	5.2.16	O	
DSDM-METINF-S-017	Support for Type element	5.2.17	M	
DSDM-METINF-S-018	Support for Version element	5.2.18	M	