



# **OMA Device Management Tree and Description Serialization**

## **Candidate Version 1.2 – 07 Jun 2005**

---

**Open Mobile Alliance**  
OMA-TS-DM-TNDS-V1\_2-20050607-C

Use of this document is subject to all of the terms and conditions of the Use Agreement located at <http://www.openmobilealliance.org/UseAgreement.html>.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. Information contained in this document may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification. However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the “OMA IPR Declarations” list at <http://www.openmobilealliance.org/ipr.html>. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR'S REPRESENTED ON THE “OMA IPR DECLARATIONS” LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

© 2005 Open Mobile Alliance Ltd. All Rights Reserved.

Used with the permission of the Open Mobile Alliance Ltd. under the terms set forth above.

# Contents

- 1. SCOPE .....4
- 2. REFERENCES .....5
  - 2.1 NORMATIVE REFERENCES.....5
  - 2.2 INFORMATIVE REFERENCES.....5
- 3. TERMINOLOGY AND CONVENTIONS .....6
  - 3.1 CONVENTIONS.....6
  - 3.2 DEFINITIONS.....6
  - 3.3 ABBREVIATIONS.....6
- 4. INTRODUCTION .....7
- 5. TND SERIALIZATION DEFINITION .....8
  - 5.1 TNDS XML USAGE.....8
  - 5.2 GENERAL.....8
  - 5.3 DM COMMANDS WITH TNDS DATA.....8
- 6. TNDS SYNTAX .....11
  - 6.1 SUPPORTED PROPERTIES IN FILE .....11
  - 6.2 TNDS DTD .....11
  - 6.3 TNDS ELEMENTS.....11
    - 6.3.1 Structural elements.....11
- APPENDIX A. CHANGE HISTORY (INFORMATIVE).....17
  - A.1 APPROVED VERSION HISTORY .....17
  - A.2 DRAFT/CANDIDATE VERSION <CURRENT VERSION> HISTORY.....17
- APPENDIX B. STATIC CONFORMANCE REQUIREMENTS (NORMATIVE).....18
  - B.1 CLIENT SCR FOR TNDS.....18
  - B.2 SERVER SCR FOR TNDS .....18
- APPENDIX C. TYPE DEFINITIONS (NORMATIVE).....19
  - C.1 MIME MEDIA TYPE DEFINITION.....19

# Figures

- Figure 1: Example of management tree.....8

# Tables

- Table 1: Supported properties in the TNDS object byte stream .....11

# 1. Scope

This specification defines how to transform a runtime management tree into an xml or wbxml file. With this specification it is possible to transform a MO or complete/part of the Management Tree to or from an xml or wbxml file.

## 2. References

### 2.1 Normative References

- [DMREPU] “OMA Device Management Representation Protocol, Version 1.2”. Open Mobile Alliance™. OMA-TS-DM-RepPro-V1\_2. [URL:http://www.openmobilealliance.org](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](http://www.openmobilealliance.org)
- [IOPPROC] “OMA Interoperability Policy and Process”, Version 1.1, Open Mobile Alliance™, OMA-IOP-Process-V1\_1, [URL:http://www.openmobilealliance.org](http://www.openmobilealliance.org)
- [ISO8601] ISO 8601:2000, Data elements and interchange formats -- Information interchange -- Representation of dates and times  
[URL:http://www.iso.ch/](http://www.iso.ch/)
- [RFC2119] “Key words for use in RFCs to Indicate Requirement Levels”, S. Bradner, March 1997, [URL:http://www.ietf.org/rfc/rfc2119.txt](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](http://www.ietf.org/rfc/rfc2234.txt)
- [RFC2396] “Uniform Resource Identifiers (URI): Generic Syntax”. T. Berners-Lee, et al. August 1998. [URL:http://www.ietf.org/rfc/rfc2396.txt](http://www.ietf.org/rfc/rfc2396.txt)
- [XMLSCHEMADT] “XML Schema Part 2: Datatypes”, W3C Recommendation 02 May 2001, [URL:http://www.w3.org/XML/Schema/](http://www.w3.org/XML/Schema/)

### 2.2 Informative References

None.

## 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

<b>Management tree</b>	The mechanism by which the management client interacts with the device, e.g. by storing and retrieving values from it and by manipulating the properties of it, for example the access control lists.
<b>Leaf node</b>	A node that can store a value, but cannot have child nodes. The Format property of a leaf node is not node.
<b>Interior node</b>	A node that may have child nodes, but cannot store any value. The Format property of an interior node is node.

### 3.3 Abbreviations

<b>OMA</b>	Open Mobile Alliance
<b>ACL</b>	Access Control List.
<b>DDF</b>	Device Description Framework
<b>TNDS</b>	Tree and Description Serialization

## 4. Introduction

This specification defines how to convert a runtime Management Tree or sub tree (i.e. an MO) into an xml or wbxml structure. This specification is useful whenever an MO needs to be created from one entity, moved outside of the DM Protocol session to another entity in an interoperable fashion. This specification is optional to support.

## 5. TND Serialization Definition

### 5.1 TNDS XML usage

This specification defines how to transform between a management sub-tree and xml structure. The DTD used for this is the same as defined in [DMTND]. A XML document complying with this specification has a similar but different content than the [DMTND] specification. Therefore a document complying with this specification **MUST** use a specific MIME-type to indicate it. It is also possible to use the WBXML encoding mechanism defined in [DMTND]. The two MIME-types, one for XML and one for WBXML are specified in Appendix C.

### 5.2 General

During runtime the client's management tree will contain interior nodes and leaf nodes. The client may take a snap-shot on the management tree or a subset of the management tree and convert all information of that part of the management tree into either an xml or a wxml stream. Some fields are optional and some are mandatory to support. The description for all properties is defined in [DMTND]. Properties which are optional in the management tree may not be stored in the file.

The DTD in [DMTND] defines the format of the xml or wxml file. This specification will not define any transport binding for the xml or wxml byte stream.

The following figure shows an example of a management tree. With this specification it is possible for the device to translate the E-Mail Management Object to or from an xml or wxml file.

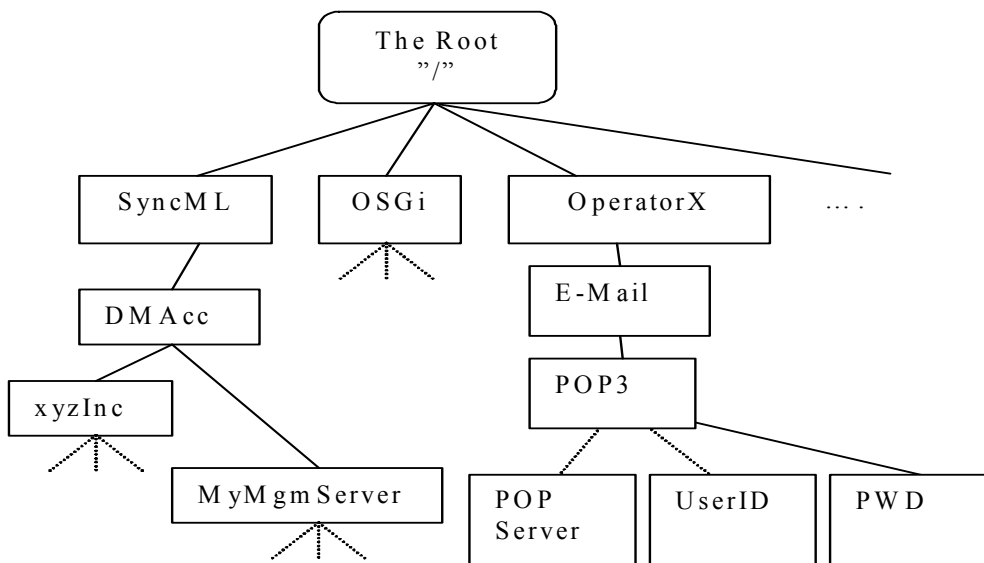


Figure 1: Example of management tree

### 5.3 DM Commands with TNDS data

The format of the TNDS object as byte stream has its own MIME-type. Therefore if the device and server support this MIME-type it is possible to use DM commands and operate on a complete sub tree in one command. The behavior for these commands are described in [DMREPPRO].

For example the server will be able to create a large complete structure including data with one single Add command.

This is an example for creating an E-Mail object:



```

<Add>
  <CmdID>4</CmdID>
  <Item>
    <Target>
      <LocURI>/OperatorX/E-Mail</LocURI>
    </Target>
    <Meta>
      <Format xmlns=' syncml:metinf'>xml</Format>
      <Type xmlns=' syncml:metinf'>
        application/vnd.syncml.dmtnds+xml
      </Type>
    </Meta>
    <Data>
      <![CDATA[
        <SyncML xmlns=' syncml:dmddf1.2'>
          <MgmtTree>
            <VerDTD>1.2</VerDTD>
            <Node>
              <Path>/OperatorX</Path>
              <NodeName>E-Mail</NodeName>
              <RTProperties >
                <Format>
                  <node/>
                </Format>
                <Type>com.operatorX.dm/1.0/EMail</Type>
              </RTProperties>
              <Node>
                <NodeName>POP3</NodeName>
                <Node>
                  <NodeName>POPServer</NodeName>
                  <RTProperties >
                    <Format>
                      <chr/>
                    </Format>
                    <Type >text/plain</Type>
                  </RTProperties >
                  <Value >mail.Operatorx.com</Value>
                </Node>
                <Node>
                  <NodeName>UserID</NodeName>
                  <RTProperties>
                    <Format>
                      <chr/>
                    </Format>
                    <Type>text/plain</Type>
                  </RTProperties>
                  <Value>UserName</Value>
                </Node>
                <Node>
                  <NodeName>PWD</NodeName>
                  <RTProperties>
                    <Format>
                      <chr/>
                    </Format>
                    <Type>text/plain</Type>
                  </RTProperties>
                  <Value>4571F7C34A9876B3</Value>
                </Node>
              </Node>
            </MgmtTree>
          </SyncML>
        ]>
      ]>
    </Data>
  </Item>
</Add>

```

```
        </Node>
      </Node>
    </MgmtTree>
  </SyncML>
}
</Data>
</Item>
<Add>
```

## 6. TNDS Syntax

### 6.1 Supported properties in file

All properties are defined and described in [DMTND]. Only a subset of these properties are valid in the file. The following table defines the subset of valid properties together with the requirements level for how it must be supported depending on if an entity is decoding or encoding a TNDS object:

Property	Decode	Encode
MgmtTree	MUST	MUST
VerDTD	MUST	MUST
Man	MAY	MAY
Mod	MAY	MAY
Node	MUST	MUST
NodeName	MUST	MUST
Path	MUST	MAY
Value	MUST	MUST
RTProperties	MUST	MUST
ACL	MUST	MAY
Format	MUST	MAY
Title	MAY	MAY
TStamp	MAY	MAY
Type	MUST	MUST
VerNo	MAY	MAY

Table 1: Supported properties in the TNDS object byte stream

### 6.2 TNDS DTD

The DTD is specified in [DMTND]. All extra behavior and restriction compared to the DTD definitions is defined later in this chapter. A compliant TNDS object MUST follow all rules in this specification. The TNDS object byte stream can be encoded as xml or as wxml. Two different MIME-types are defined to indicate what encoding type the byte stream uses.

### 6.3 TNDS Elements

This section defines the elements that are allowed in the TNDS object.

#### 6.3.1 Structural elements

These elements provide various kinds of structural information of the described management object.

##### 6.3.1.1 MgmtTree

Usage: Container for one or more described management objects.

Parent Elements: none

Restrictions: This MUST be the root element of all descriptions.

### 6.3.1.2 VerDTD

Usage: Specifies the major and minor version identifier of the SyncML DM Description Framework specification used to represent the SyncML DM description.

Parent Elements: MgmtTree

Restrictions: Major revisions of the specification create incompatible changes which will generally require a new parser. Minor revisions involve changes that do not impact basic compatibility of the parser. When the XML document conforms to this revision of the SyncML Tree and Description specification [DMTND] the value MUST be 1.2. The element type MUST be included in the MgmtTree.

### 6.3.1.3 Man

Usage: Specifies the device manufacturer name of the device the TNDS object is created from.

Parent Elements: MgmtTree

Restrictions: This element is OPTIONAL.

### 6.3.1.4 Mod

Usage: Specifies the device manufacturer's model number of the device the TNDS object is created from.

Parent Elements: MgmtTree

Restrictions: This element is OPTIONAL.

### 6.3.1.5 Node

Usage: Specifies a node.

Parent Elements: MgmtTree

Restrictions: This element is recursive. A Node with a Value element MUST always terminate the recursion. It is possible for a Node to omit both the next recursive Node and a Value, this means that the hierarchy of Nodes continues elsewhere. This can be used to increase readability of very deep trees. In the continuation, the Path element MUST contain a full URI that specifies the insertion point in the tree.

Example: The following XML is a description of a number of nodes that form the URI Vendor/ISP/GWInfo/GWName.

```
<MgmtTree>
  <Node>
    <NodeName>Vendor</NodeName>
    <RTProperties>...</RTProperties>
    <Node>
      <NodeName>ISP</NodeName>
      <RTProperties>...</RTProperties>
      <Node>
        <NodeName>GWInfo</NodeName>
        <RTProperties>...</RTProperties>
        <Node>
          <NodeName>GWName</NodeName>
          <RTProperties>...</RTProperties>
          <Value>gw.yyy.se</Value>
        </Node>
      </Node>
    </Node>
  </Node>
</MgmtTree>
```

### 6.3.1.6 NodeName

Usage: Specifies the name of the described node.

Parent Elements: Node

Restrictions: See [RFC2396] for general restrictions on URI. The NodeName element MAY be empty. If empty, this means that the name of the node MUST be assigned when the node is created. When the node name is assigned at node creation time, the value for the name is set to the last segment of the URI specified as Target for the command this results in the node being created. See also in [DMTND].

If NodeName is omitted then the client may use the Type to decide where in the tree the structure should be created.

### 6.3.1.7 Path

Usage: Specifies the URI up to, but not including, the described node.

Parent Elements: Node

Restrictions: OPTIONAL element when creating a TNDS object and MANDATORY when importing nodes from a TNDS object. If omitted, the URI for the node MUST be constructed by concatenating all ancestral NodeName and Path values. This concatenated value MUST form the correct URI. Path SHOULD only be used inside Node elements that are child elements to the MgmtTree element. For general restrictions, see [RFC2396].

Example: The following XML is an alternative way to describe the same management objects as in the example in section 6.3.1.5. This description specifies the same URI as the other example: Vendor/ISP/GWInfo/GWName .

```
<MgmtTree>
  <Node>
    <NodeName>Vendor</NodeName>
    <RTProperties>...</RTProperties>
  </Node>
  <Node>
    <NodeName>ISP</NodeName>
    <Path>Vendor</Path>
    <RTProperties>...</RTProperties>
  </Node>
  <Node>
    <NodeName>GWInfo</NodeName>
    <Path>Vendor/ISP</Path>
    <RTProperties>...</RTProperties>
  </Node>
  <Node>
    <NodeName>GWName</NodeName>
    <Path>Vendor/ISP/GWInfo</Path>
    <RTProperties>...</RTProperties>
    <Value>gw.yyy.se</Value>
  </Node>
</MgmtTree>
```

### 6.3.1.8 Value

Usage: Specifies the value for nodes that are instantiated using the current description.

Parent Elements: Node

Restrictions: OPTIONAL element. If omitted, the node description does not specify any default value for the node. In this case the initial value of new nodes are undefined.

### 6.3.1.9 RTProperties

Usage: Aggregating element for run-time properties, i.e. properties that the nodes have in a device at run-time. Used to specify which properties the described node supports at run-time. Can also be used to supply default values for supported run-time properties.

Parent Elements: Node

Restrictions: MANDATORY element.

### 6.3.1.10 ACL

Usage: Specifies the ACL value. This MAY be used to specify a different ACL value than the default value for this node.

Parent Elements: RTProperties

Restrictions: If a value is specified it MUST be formatted according to definition in [DMTND].

### 6.3.1.11 Format

Usage: Specifies the Format value.

Parent Elements: RTProperties

Restrictions: If a value is specified it MUST specify the correct format of the node according to section definition in [DMTND].

### 6.3.1.12 TStamp

Usage: Specifies the TStamp value.

Parent Elements: RTProperties

Restrictions: If a value is specified it MUST be formatted according to section definition in [DMTND].

### 6.3.1.13 Type

Usage: Specifies the Type value.

Parent Elements: RTProperties

Restrictions: For leaf nodes, the Type property MUST be used to specify the correct MIME type of the node value. For interior nodes the value MUST specify a valid identifier of a DDF document, or be empty according to section definition in [DMTND].

### 6.3.1.14 VerNo

Usage: Specifies the VerNo value.

Parent Elements: RTProperties

Restrictions: If a value is specified it MUST be formatted according to definition in [DMTND].

### 6.3.1.15 b64

Usage: OMA DM format description. Specifies that the node value is Base64 encoded.

Parent Elements: Format

Restrictions: None.

### **6.3.1.16 bin**

Usage: OMA DM format description. Specifies that the node value is binary data.

Parent Elements: Format

Restrictions: None.

### **6.3.1.17 bool**

Usage: OMA DM format description. Specifies that the node value is a Boolean.

Parent Elements: Format

Restrictions: None.

### **6.3.1.18 chr**

Usage: OMA DM format description. Specifies that the node value is text.

Parent Elements: Format

Restrictions: The character set used is specified either by the transport protocol, MIME content type header or XML prologue.

### **6.3.1.19 int**

Usage: OMA DM format description. Specifies that the node value is an integer.

Parent Elements: Format

Restrictions: None.

### **6.3.1.20 node**

Usage: OMA DM format description. Specifies that the node is an interior node.

Parent Elements: Format

Restrictions: This Format MUST only be used for interior nodes.

### **6.3.1.21 null**

Usage: OMA DM format description. Specifies that the node value is null.

Parent Elements: Format

Restrictions: None.

### **6.3.1.22 xml**

Usage: OMA DM format description. Specifies that the node value is XML data.

Parent Elements: Format

Restrictions: None.

### **6.3.1.23 date**

Usage: OMA DM format description. Specifies that the node value is a date in ISO 8601 format with the century being included in the year [ISO8601].

Parent Elements: `Format`

Restrictions: None.

#### **6.3.1.24 time**

Usage: OMA DM format description. Specifies that the node value is a time in ISO 8601 format [ISO8601].

Parent Elements: `Format`

Restrictions: None.

#### **6.3.1.25 float**

Usage: OMA DM format description. Specifies that the node value is a real number corresponding to a single precision 32 bit floating point type as defined in XML Schema 1.0 as the `float` primitive type [XMLSCHEMADT].

Parent Elements: `Format`

Restrictions: None.



## Appendix A. Change History

(Informative)

### A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior versions

### A.2 Draft/Candidate Version 1.2 History

Document Identifier	Date	Sections	Description
OMA-DM-BLOB-MO-V1_0	02 Nov 2004	All	First draft
OMA-DM-BLOB-MO-V1_0	05 Nov 2004	All	Language corrections + example correction
OMA-OS_DM-DMTNDS-V1_2	23 Nov 2004	5	Updates after Barcelona meeting + Name change
OMA-DM-TNDS-V1_2	16 Dec 2004	5.2, 6.3.1	Included CR 2004-0321
OMA-DM-TNDS-V1_2	15 Dec 2004	5.2, 5.3	OMA-DM-2004-0348-CR-TNDS-Example-with-CDATA
OMA-DM-TNDS-V1_2	5 Jan 2005	A.2 B.1 & B.2	Version missing in header Renamed header
OMA-TS-DM-TNDS-V1_2	31 Jan 2005	All	Updated filenames.
OMA-TS-DM-TNDS-V1_2	03 May 2005	Filename Title page References	Changed version from 1.2.0 to 1.2
Candidate Versions OMA-TS-DM-TNDS-V1_2	07 Jun 2005	n/a	Candidate version approved by TP R&A OMA-TP-2005-0137R01-DM-V1_2-for-Candidate-approval

## Appendix B. Static Conformance Requirements (Normative)

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

### B.1 SCR for DM Client

Item	Function	Reference	Status	Requirement
DM-TNDS-C-001	Support of Encoding a TNS object	Section 6	O	
DM-TNDS-C-002	Support of Decoding a TNS object	Section 6	O	

### B.2 SCR for DM Server

Item	Function	Reference	Status	Requirement
DM-TNDS-S-001	Support of Encoding a TNS object	Section 6	O	
DM-TNDS-S-002	Support of Decoding a TNS object	Section 6	O	

## Appendix C. Type definitions

(Informative)

### C.1 MIME Media Type Definition

MIME Type	Description
application/vnd.syncml.dmtnds+xml	XML encoded Serialized MO that comply with this specification
application/vnd.syncml.dmtnds+wxml	WXML encoded Serialized MO that comply with this specification and WXML encoded as specified in [DMTND]