



# Service User Profile Management Technical Specification

## LDAP binding for SUPM-1 interface

Approved Version 1.0 - 29 May 2012

---

**Open Mobile Alliance**

OMA-TS-Service\_User\_Profile\_Management\_LDAP\_Binding-V1\_0-  
20120529-A

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.

© 2012 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 .....	5
2. REFERENCES.....	6
2.1 NORMATIVE REFERENCES .....	6
2.2 INFORMATIVE REFERENCES .....	6
3. TERMINOLOGY AND CONVENTIONS .....	7
3.1 CONVENTIONS .....	7
3.2 DEFINITIONS.....	7
3.3 ABBREVIATIONS .....	7
4. INTRODUCTION .....	8
4.1 VERSION 1.0 .....	8
5. SUPM LDAP BINDING PRINCIPLES.....	9
6. OPERATIONS/MESSAGES .....	10
6.1 SERVICE USER PROFILE MANAGEMENT .....	10
6.1.1 Operation: createServiceUserProfileData .....	10
6.1.2 Operation: readServiceUserProfileData .....	10
6.1.3 Operation: updateServiceUserProfileData .....	11
6.1.4 Operation: deleteServiceUserProfileData .....	12
6.2 STATUS AND FAULT REPORTING.....	13
7. SPECIFIC SECURITY CONSIDERATIONS .....	14
7.1 AUTHENTICATION AND LDAP BIND OPERATION .....	14
APPENDIX A. CHANGE HISTORY (INFORMATIVE).....	15
A.1 APPROVED VERSION HISTORY .....	15
APPENDIX B. STATIC CONFORMANCE REQUIREMENTS (NORMATIVE).....	16
B.1 SCR FOR SUPM.LDAP SERVER.....	16
B.1.1 SCR for SUPM.LDAP.AttributesForUserProfile Server .....	16
B.1.2 SCR for SUPM.LDAP.IndividualAttributesOfUserProfile Server .....	16
B.1.3 SCR for SUPM.LDAP.DataViewsOfUserProfile Server.....	16
APPENDIX C. SUPM-1 LDAP SEQUENCE DIAGRAM (INFORMATIVE).....	17
APPENDIX D. DSML OPTION FOR LDAP (INFORMATIVE) .....	19
APPENDIX E. APPENDIX E. EXAMPLES (INFORMATIVE).....	20
E.1 OPERATION: CREATESERVICEUSERPROFILEDATA.....	20
E.2 OPERATION: READSERVICEUSERPROFILEDATA.....	20
E.3 OPERATION: UPDATESERVICEUSERPROFILEDATA .....	21
E.4 OPERATION: DELETESERVICEUSERPROFILEDATA .....	22

## Figures

Figure 1 - LDAP Message Call Flow .....	17
---	----

## Tables

Table 1: Input message AddRequest.....	10
Table 2: Output message AddResponse.....	10
Table 3: Input message SearchRequest .....	11
Table 4: Input message SearchResponse .....	11

**Table 5: Input message ModifyRequest.....12**  
**Table 6: Output message ModifyResponse.....12**  
**Table 7: Input message DelRequest .....12**  
**Table 8: Output message DelResponse .....13**

# 1. Scope

The scope of this document is to specify an LDAP protocol binding for the set of operations defined in [SUPM-TS].

## 2. References

### 2.1 Normative References

- [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)
- [RFC4234] “Augmented BNF for Syntax Specifications: ABNF”. D. Crocker, Ed., P. Overell. October 2005, [URL:http://www.ietf.org/rfc/rfc4234.txt](http://www.ietf.org/rfc/rfc4234.txt)
- [RFC4511] “Lightweight Directory Access Protocol (LDAP): The Protocol”. J. Sermersheim, Ed. June 2006, [URL:http://www.ietf.org/rfc/rfc4511.txt](http://www.ietf.org/rfc/rfc4511.txt)
- [RFC4512] “Lightweight Directory Access Protocol (LDAP): Directory Information Models”. K. Zeilenga, Ed. June 2006, [URL:http://www.ietf.org/rfc/rfc4512.txt](http://www.ietf.org/rfc/rfc4512.txt)
- [RFC4513] “Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms”. R. Harrison, Ed. June 2006, [URL:http://www.ietf.org/rfc/rfc4513.txt](http://www.ietf.org/rfc/rfc4513.txt)
- [RFC4529] “Requesting Attributes by Object Class in the Lightweight Directory Access Protocol (LDAP)”. K. Zeilenga, Ed. June 2006, [URL:http://www.ietf.org/rfc/rfc4529.txt](http://www.ietf.org/rfc/rfc4529.txt)
- [SCRRULES] “SCR Rules and Procedures”, Open Mobile Alliance™, OMA-ORG-SCR\_Rules\_and\_Procedures, [URL:http://www.openmobilealliance.org/](http://www.openmobilealliance.org/)
- [SUPM-RD] “OMA Service User Profile Management Requirements”, Open Mobile Alliance™, OMA-RD-Service\_User\_Profile\_Management-V1\_0, [URL:http://www.openmobilealliance.org/](http://www.openmobilealliance.org/)
- [SUPM-AD] “OMA Service User Profile Management Architecture”, Open Mobile Alliance™, OMA-AD-Service\_User\_Profile\_Management-V1\_0, [URL:http://www.openmobilealliance.org/](http://www.openmobilealliance.org/)
- [SUPM-TS] “Service User Profile Management Technical Specification”, Open Mobile Alliance™, OMA-TS-Service User Profile Management-V1\_0, [URL:http://www.openmobilealliance.org/](http://www.openmobilealliance.org/)

### 2.2 Informative References

- [DSMLv2] “Directory Services Markup Language v2.0”, OASIS, [URL:http://www.oasis-open.org/committees/dsml/docs/DSMLv2.doc](http://www.oasis-open.org/committees/dsml/docs/DSMLv2.doc)
- [OMADICT] “Dictionary for OMA Specifications”, Version 2.8, Open Mobile Alliance™, OMA-ORG-Dictionary-V2\_8, [URL:http://www.openmobilealliance.org/](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.

### 3.2 Definitions

<b>Application</b>	See [OMADICT]
<b>Authorized Principal</b>	See [OMADICT]
<b>Principal</b>	See [OMADICT]
<b>Resource</b>	See [OMADICT]
<b>Service</b>	See [OMADICT]
<b>Service Provider</b>	See [OMADICT]
<b>Service User Profile</b>	See [SUPM-RD]
<b>User Profile</b>	See [OMADICT]
<b>User</b>	See [OMADICT]
<b>User’s Characteristic Description Information</b>	See [SUPM-RD]
<b>SUPM Data View</b>	See [SUPM-AD]

### 3.3 Abbreviations

<b>AD</b>	Architecture Document
<b>IP</b>	Internet Protocol
<b>LDAP</b>	Lightweight Directory Access Protocol
<b>OMA</b>	Open Mobile Alliance
<b>SUPM</b>	Service User Profile Management
<b>TCP</b>	Transmission Control Protocol

## 4. Introduction

The Service User Profile Management (SUPM) enabler provides a standardized interface to access and manage the data related to Service User Profile, with which applications and/or enablers can create, read, update and delete those data in order to support contextualization and personalization of the User's services.

This interface supports requests to create/read/update/delete Service User Profile data, and the corresponding responses.

This specification described the mapping of these operations for LDAP. The abstract definitions for the SUPM-1 interface are provided in [SUPM-TS].

### 4.1 Version 1.0

This document covers all requirements [SUPM-RD] of SUPM V 1.0.



## 5. SUPM LDAP Binding principles

SUPM-1 interface features specified in [SUPM-TS] are mapped to LDAP with following general principles:

- SUPM-1 operations `createServiceUserProfileData`, `readServiceUserProfileData`, `updateServiceUserProfileData`, `deleteServiceUserProfileData` are mapped to LDAP messages, respectively *AddRequest*, *SearchRequest*, *ModifyRequest*, *DelRequest* as defined in [RFC4511].
- Authentication of the data consumer is performed during the LDAP Bind message exchange as defined in [RFC4512].
- Data elements and their types are described in a LDAP Directory Schema as defined in [RFC4512].
- SUPM Data Views are mapped to object classes in LDAP as defined in [RFC4512]. LDAP object classes identify the set of attributes that “MUST” or “MAY” be present in entries that have this LDAP object class. [RFC4529] extends LDAP to support a mechanism that LDAP clients may use to request the return of all attributes of an LDAP object class.

## 6. Operations/messages

This section describes the SUPM operations/messages supported by the SUPM LDAP Binding.

Note: The data consumer (Originator part name) is explicitly identified during the LDAP Session establishment (LDAP BindRequest) and is therefore not present in each of the operations described below.

### 6.1 Service User Profile Management

#### 6.1.1 Operation: createServiceUserProfileData

This operation allows the creation of a Service User Profile for a specific user. Initial attribute values are pre-populated as part of the profile creation.

This operation is mapped to LDAP *AddRequest* message, and corresponding response message as specified in [RFC4511].

##### 6.1.1.1 Input message: AddRequest

Abstract part name	LDAP part name	Optional	Description
Userld	entry	No	Identification of the user for whom the profile is created.
AttributeValue Pairs	attributes	No	List of attribute names and attribute values.

**Table 1: Input message AddRequest**

##### 6.1.1.2 Output message: AddResponse

Abstract part name	LDAP part name	Optional	Description
ResultCode	resultCode	No	Success indication or error code
ErrorMessage	diagnosticMessage	Yes	Optional error diagnostic description.

**Table 2: Output message AddResponse**

#### 6.1.2 Operation: readServiceUserProfileData

This operation allows the access to Service User Profile.

This operation is mapped to LDAP *SearchRequest* message, and corresponding response messages as specified in [RFC4511].

### 6.1.2.1 Input message: SearchRequest

Abstract part name	LDAP part name	Optional	Description
Userld	baseObject	No	Identification of the user that is used for the read operation.
AttributeNames	attributes	No	List of attributes that are requested. This list may include LDAP attributes or LDAP object class identifiers.  As defined in [RFC4529], if an LDAP object class identifier is requested all attributes belonging to this object class are returned. The "@" character is used to distinguish an object class identifier from an attribute descriptions.
N/A	filter	No	Filter checking for the presence (filter.present [value = 7]) of the 'objectClass' attribute. Note: This is the standard LDAP mechanism to emulate a read operation. This filter is always true as all entries have an objectClass.
N/A	scope	No	baseObject.[value = 0]

**Table 3: Input message SearchRequest**

### 6.1.2.2 Output message: SearchResponse

Upon receipt of a SearchRequest, the server attempts to perform the necessary modifications to the Service User Profile and returns the result in a SearchResponse, defined as follows:

Abstract part name	LDAP part name	Optional	Description
ResultEntries	SearchResultEntry	Yes	If successful it may return one SearchResultEntry which is the request Service User Profile as LDAP objectName (UserID) and the list of requested LDAP attributes. It may include the LDAP objectClass attribute if an object class has been requested in the searchRequest [RFC4529].
ResultCode	resultCode	No	Success indication or error code
ErrorMessage	diagnosticMessage	Yes	Optional error diagnostic description.

**Table 4: Input message SearchResponse**

## 6.1.3 Operation: updateServiceUserData

This operation allows to update individual attributes of a Service User Profile for a specific user. It is required to use AddRequest (respectively DelRequest) operation to create (respectively delete) a complete Service User Profile for a specific user.

This operation is mapped to LDAP *ModifyRequest* message, and corresponding response message as specified in [RFC4511].

### 6.1.3.1 Input message: ModifyRequest

Abstract part name	LDAP part name	Optional	Description
Userld	object	No	Identification of the user for whom the profile is updated.
AttributeValue Pairs	changes	No	A sequence of {operations, and modifications}:
N/A	operation	No	Used to specify the type of modification being performed. Each operation type acts on the following modification. The replace operation SHALL be supported.  replace: replace all existing values of the modification attribute with the new values listed, creating the attribute if it did not already exist. A replace with no value will delete the entire attribute if it exists, and it is ignored if the attribute does not exist.
N/A	modification	No	The attribute being modified and the value to be used in the modification. Value is optional as it is not needed for deletion

**Table 5: Input message ModifyRequest**

### 6.1.3.2 Output message: ModifyResponse

Upon receipt of a ModifyRequest, the server attempts to perform the necessary modifications to the Service User Profile and returns the result in a ModifyResponse, defined as follows:

Abstract part name	LDAP part name	Optional	Description
ResultCode	resultCode	No	Success indication or error code
ErrorMessage	diagnosticMessage	Yes	Optional error diagnostic description.

**Table 6: Output message ModifyResponse**

## 6.1.4 Operation: deleteServiceUserProfileData

This operation allows the deletion of the Service User Profile for a specific user. It is required to use the update operation to delete individual attributes of a Service User Profile.

This operation is mapped to LDAP *DelRequest* message, and corresponding response message as specified in [RFC4511].

### 6.1.4.1 Input message: DelRequest

Abstract part name	LDAP part name	Optional	Description
Userld	entry	No	Identification of the user for whom the profile is deleted.

**Table 7: Input message DelRequest**

### 6.1.4.2 Output message: DelResponse

Upon receipt of a DelRequest, the SUPM Component will attempt to perform the requested removal of Service User Profile belonging to a specific user and return the result in the DelResponse defined as follows:

Abstract part name	LDAP part name	Optional	Description
ResultCode	resultCode	No	Success indication or error code
ErrorMessage	diagnosticMessage	Yes	Optional error diagnostic description.

**Table 8: Output message DelResponse**

## 6.2 Status and Fault reporting

The rules for status and fault reporting in [RFC4511] SHALL be applied. In particular the rules for LDAP result codes are defined in [RFC4511] chapter 4.1.9.

Note: The diagnostic messages is a human readable string and is not standardized, implementations MUST NOT rely on the values returned. Diagnostic messages typically supplement the resultCode with additional information.

## 7. Specific Security Considerations

### 7.1 Authentication and LDAP Bind operation

The Bind operation as defined in [RFC4511] allows authentication information to be exchanged between the LDAP client (data consumer) and the LDAP server (SUPM component).

A data consumer shall first establish a transport connection (TCP) with the SUPM component. After establishment of the transport connection the data consumer shall initiate a LDAP session using the LDAP *BindRequest* operation. The data consumer is identified in the Bind Request message. The *BindRequest* operation shall be sent before any other SUPM operation.

The SUPM component shall support the “name/password authentication mechanism of Simple Bind” of “Simple Authentication Method” as defined in [RFC4513].

The LDAP session is closed by the data consumer by sending an *UnbindRequest* operation as defined in [RFC4511].

## Appendix A. Change History

(Informative)

### A.1 Approved Version History

Reference	Date	Description
OMA-TS-Service_User_Profile_Management_LDAP_Binding-V1_0-20120529-A	29 May 2012	Status changed to Approved by TP Ref TP Doc # OMA-TP-2012-0204-INP_SUPM_V1_0_for_Final_Approval

## Appendix B. Static Conformance Requirements (Normative)

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

### B.1 SCR for SUPM.LDAP Server

Item	Function	Reference	Requirement
SUPM-LDAP-S-001-M	Support for the LDAP binding of SUPM-1 interface	5	

#### B.1.1 SCR for SUPM.LDAP.AttributesForUserProfile Server

Item	Function	Reference	Requirement
SUPM-LDAP-ATTRIB-S-001-M	Management support for all attributes for a given user profile.	6.1	
SUPM-LDAP-ATTRIB-S-002-M	This operation creates a Service User profile - AddRequest	6.1.1	
SUPM-LDAP-ATTRIB-S-003-M	This operation returns all attributes - SearchRequest	6.1.2	
SUPM-LDAP-ATTRIB-S-004-M	This operation updates all the attributes - ModifyRequest	6.1.3	
SUPM-LDAP-ATTRIB-S-005-M	This operation deletes a Service User Profile- DelRequest	6.1.4	

#### B.1.2 SCR for SUPM.LDAP.IndividualAttributesOfUserProfile Server

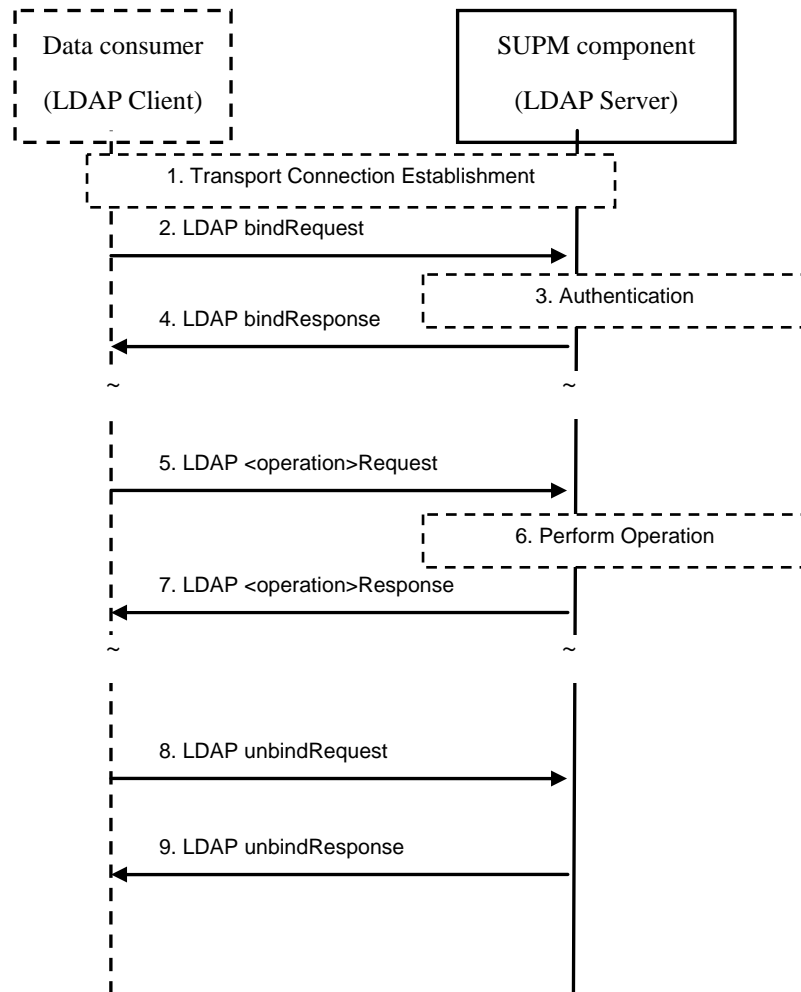
Item	Function	Reference	Requirement
SUPM-LDAP-IND-ATTRIB-S-001-M	Support for management (create, update, retrieve and delete) of individual attributes for a user profile	6.1	
SUPM—LDAP-IND-ATTRIB-S-002-M	This operation returns the value of user profile attributes - SearchRequest	6.1.2	
SUPM—LDAP-IND-ATTRIB-S-003-M	This operation creates, updates or deletes attributes of an user profile - ModifyRequest	6.1.3	

#### B.1.3 SCR for SUPM.LDAP.DataViewsOfUserProfile Server

Item	Function	Reference	Requirement
SUPM-LDAP-DV-S-001-O	Support for operations (retrieve) for a Data View of an user profile		SUPM-LDAP-DV-S-002-O
SUPM-LDAP-DV-S-002-O	This operation returns the list of attributes and values of a Data View of an user profile – SearchRequest	6.1.2	



## Appendix C. SUPM-1 LDAP sequence diagram (Informative)



**Figure 1 - LDAP Message Call Flow**

1. Data consumer establishes a transport connection (TCP/IP) towards the SUPM component on a well known LDAP port or any agreed TCP port.
2. Data consumer issues LDAP *bindRequest* with its credentials.
3. SUPM component performs data consumer authentication.
4. SUPM component issues a LDAP *bindResponse*. The data consumer shall not issue any LDAP operation before receiving the LDAP bindResponse.
5. SUPM component sends an LDAP operation Request (LDAP *addRequest*, *searchRequest*, *modifyRequest*, *delRequest*).
6. SUPM component performs the operation for the identified Service User Profile data. SUPM component determines based on configuration at deployment time, where the actual data specified in the request resides and reads (or creates, deletes, modifies) the data from the data source.

7. SUPM component returns the result of the operation to the data consumer using the corresponding LDAP operation Response.

NOTE: Step 5 to 7 are repeated as necessary until session termination.

8. To end the LDAP session, the data consumer issues a LDAP *unbindRequest*.
9. SUPM component sends a LDAP *unbindResponse* to confirm session termination.

## Appendix D. DSML option for LDAP (Informative)

SUPM deployments that implement the LDAP binding that require a SOAP/XML interface with the same level of functionalities and LDAP data concepts may use DSMLv2 specification with SOAP/XML normative binding as defined in section 6 of [DSMLv2].

DSMLv2 expresses LDAP requests and responses as XML document fragments. For the most part DSMLv2 is a systematic translation of LDAP's ASN.1 grammar into XML-Schema. Thus, when a DSMLv2 element name matches an identifier in LDAP's ASN.1 grammar, the named element means the same thing in DSMLv2 and in LDAP.

The mapping of the abstract operations and parameters defined for LDAP is applicable for DSML.

## Appendix E. Appendix E. Examples (Informative)

The following examples illustrate the use of DSML for the SUPM operations. The examples illustrate also the use of native LDAP.

Examples are based on the data elements and SUPM Data Views defined in [SUPM\_TS].

### E.1 Operation: CreateServiceUserProfileData

The following is an example of a createServiceUserProfileData operation mapped on *addRequest*.

```
<?xml version="1.0" encoding="UTF-8"?>
<batchRequest xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<addRequest dn="UserID= mailto:bob@example.com,DC=example,DC=com">
  <attr name="objectclass"><value>top</value></attr>
  <attr name="objectclass"><value>name</value></attr>
  <attr name="objectclass"><value>address</value></attr>
  <attr name="title"><value>Mr</value></attr>
  <attr name="given"><value>Bob</value></attr>
  <attr name="country"><value>France</value></attr>
  <attr name="locality"><value>Paris</value></attr>
</addRequest>
</batchRequest>
```

Example of *addResponse*:

```
<?xml version="1.0" encoding="UTF-8"?>
<batchResponse xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<addResponse>
  <resultCode code="0"/>
  <errorMessage>completed</errorMessage>
</addResponse>
</batchResponse>
```

### E.2 Operation: ReadServiceUserProfileData

The following is an example of a readServiceUserProfileData operation mapped on *searchRequest*. This example asks for the attribute "given".

```
<?xml version="1.0" encoding="UTF-8"?>
<batchRequest xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<searchRequest dn="UserID= mailto:bob@example.com,DC=example,DC=com" scope="baseObject" derefAliases="derefAlways">
  <filter> <present name="objectclass"/></filter>
  <attributes>
    <attribute name="given"/>
  </attributes>
</searchRequest>
</batchRequest>
```

The *searchResponse* example shows the returned attribute “given”:

```
<?xml version="1.0" encoding="UTF-8"?>
<batchResponse xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<searchResponse>
  <searchResultEntry dn="UserID= mailto:bob@example.com,DC=example,DC=com">
    <attr name="given">
      <value>Bob</value>
    </attr>
  </searchResultEntry>
</searchResponse>
</batchResponse>
```

The following is an example of a read request asking for a whole user profile:

```
<?xml version="1.0" encoding="UTF-8"?>
<batchRequest xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<searchRequest dn="UserID= mailto:bob@example.com,DC=example,DC=com " scope="baseObject" derefAliases="derefAlways">
  <filter><present name="objectClass"/></filter>
</searchRequest>
</batchRequest>
```

## E.3 Operation: UpdateServiceUserData

The following example is based on the *updateServiceUserData* operation mapping on *modifyRequest* defined in chapter 6.1.3

Example of *modifyRequest*:

```
<?xml version="1.0" encoding="UTF-8"?>
<batchRequest xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<modifyRequest dn="UserID= mailto:bob@example.com,DC=example,DC=com">
  <modification name="country" operation="replace">
    <value>Austria</value>
  </modification>
  <modification name="locality" operation="replace">
    <value>Vienna</value>
  </modification>
  <modification name="title" operation="replace">
    <value>Mr</value>
  </modification>
</modifyRequest>
</batchRequest>
```

Example of *modifyResponse*:

```
<?xml version="1.0" encoding="UTF-8"?>
<batchResponse xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<modifyResponse>
  <resultCode code="53" descr="unwillingToPerform"/>
  <errorMessage>System Attribute may not be modified</errorMessage>
</modifyResponse>
</batchResponse>
```

## E.4 Operation: DeleteServiceUserProfileData

The following is an example of a deleteServiceUserProfileData operation mapped on *delRequest*

Examples of *delRequest*:

```
<?xml version="1.0" encoding="UTF-8"?>
<batchRequest xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
  <delRequest dn="UserID= mailto:bob@example.com,DC=example,DC=com " />
</batchRequest>
```

Example of *delResponse*:

```
<?xml version="1.0" encoding="UTF-8"?>
<batchResponse xmlns="urn:oasis:names:tc:DSML:2:0:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:DSML:2:0:core http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd">
<delResponse>
  <resultCode code="32" descr="noSuchObject"/>
  <errorMessage>DSDEL::230234</errorMessage>
</delResponse>
</batchResponse>
```