

Chapter 3

Subscriber Manager

This chapter contains information about Subscriber Manager, an SDX gateway application. This chapter contains the following sections:

Overview on page 45

Configuration and Installation Tasks on page 48

Sample Data on page 53

Testing Subscriber Manager on page 54

Developing Gateway Clients on page 54

Managing Gateway Clients on page 55

Overview

Subscriber Manager allows a business partner and its residential subscribers to access the SDX owner's equipment via the SDX gateway and to modify objects for residential subscribers in the SDX owner's directory. For an overview of the SDX gateway, see *Chapter 2, Dynamic Service Activator*. With Subscriber Manager, a business partner and its residential subscribers can perform the following functions:

Register subscribers.

Manage subscriber profiles.

Subscribe to services and unsubscribe from services.



NOTE: Subscriber Manager does not handle information related to billing.

The SDX owner is responsible for:

Installing one Subscriber Manager Web application for each business partner (see *Configuration and Installation Tasks* on page 48).

Configuring Subscriber Manager to interact with the business partner and optionally to start a workflow through the object state manager for the Web (OSMW). For more information on this topic, see *Configuration and Installation Tasks* on page 48).

The business partner is responsible for:

Creating the gateway clients that communicate with the gateway (see *Developing Gateway Clients* on page 54).

(Optional) Providing a way (for example, through the portal) for subscribers to register, manage their profiles, and manage services.

Subscriber Manager Operation

The following steps explain how Subscriber Manager interacts with other components to enable the gateway client to manage subscribers and services. Figure 3 illustrates this process.

1. The gateway client sends via HTTP a SOAP request to the Web application server at the SDX gateway.

The request includes:

Details about the operation that the gateway client wants to perform.

Information about the OSM that will perform workflow transactions.

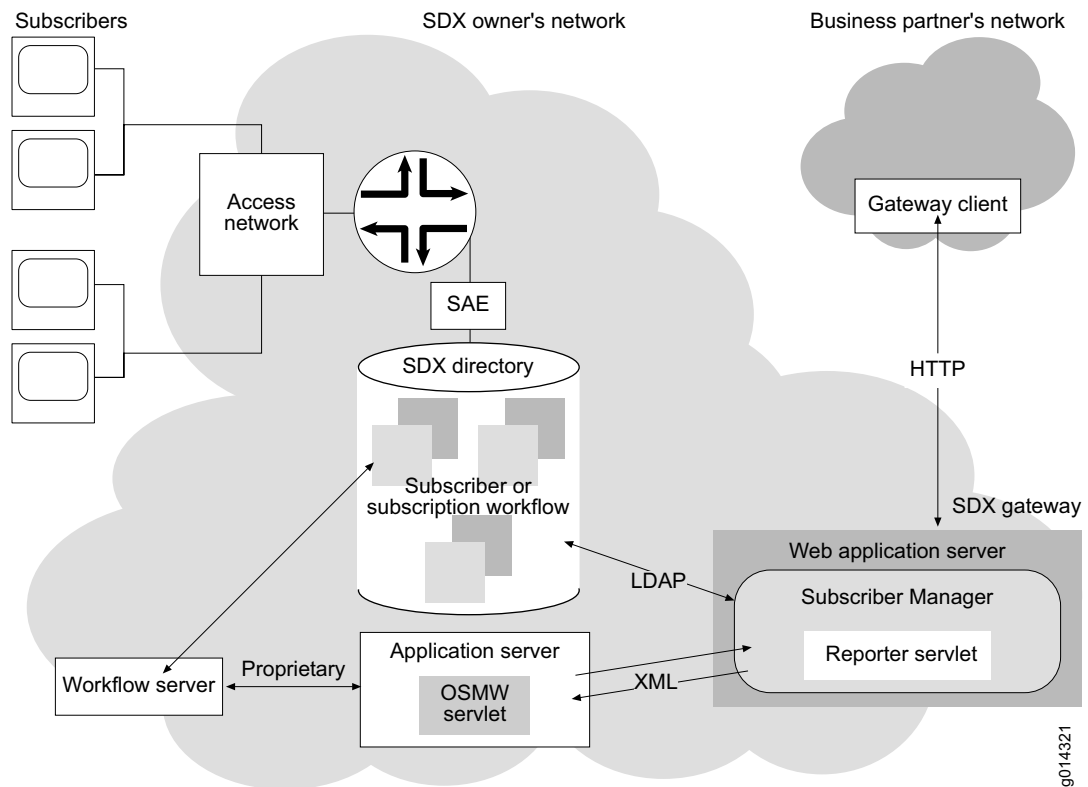
2. The Web application server authenticates the gateway client's identity.
3. The Web application server passes the SOAP request to Subscriber Manager.
4. Depending on the type of request, Subscriber Manager proceeds in one of the following ways.

If the gateway client sends a request for information, Subscriber Manager queries the directory and returns the information as the result of the request.

If the gateway client sends a request for updates to information, Subscriber Manager creates or changes entries in the directory. If Subscriber Manager is configured to use the OSMW, OSMW transactions control updates to objects in the directory. SDX owners can configure transactions to start workflows that perform associated operations. For example, if the OSMW suspends a subscription, this transaction can trigger a workflow that sends a letter to the subscriber. For details about how Subscriber Manager interacts with the OSMW, see *Interactions Between Subscriber Manager and OSMW* on page 47.

5. Subscriber Manager returns a result to the gateway client via a SOAP response.

Figure 3: Subscriber Manager operation



Interactions Between Subscriber Manager and OSMW

When you configure Subscriber Manager to use the OSMW, some types of requests from Subscriber Manager initiate OSMW actions that change the state of an object. The OSMW executes a transaction to manage the change of state. When it executes a transaction, the OSMW:

1. Locks the object so that it cannot be changed by another component simultaneously.
2. Checks that the requested action is valid for the object's current state, and cancels the transaction if it is not valid.
3. Executes a workflow if one is associated with the action, and cancels the transaction if the workflow fails.
4. Performs an action that changes the state of the object.
5. Unlocks the object.

Table 9 shows the actions that the OSMW performs when it receives a specific type of request from Subscriber Manager.

Table 9: Request types for OSMW actions

Request Type (Operation)	OSMW Action
CreateSubscriber	create
CreateSubscriberExt	create
ModifySubscriber	modify
DeleteSubscriber	delete
SuspendSubscriber	suspend
ResumeSubscriber	resume
FixSubscriber	fix
ChangeBasicSubscription	chgbasic
CreateSubscription	create
ModifySubscription	modify
DeleteSubscription	delete
SuspendSubscription	suspend
ResumeSubscription	resume

Configuration and Installation Tasks

This section lists the main tasks involved with configuring and installing Subscriber Manager. Refer to the following sections for more detailed information on these tasks.

To configure and install Subscriber Manager:

1. Complete the prerequisite tasks.
2. Configure one instance of Subscriber Manager for each business partner who uses the gateway.
3. Install the instance of Subscriber manager.

Prerequisite Tasks

Before you use Subscriber Manager, you must:

Deploy a working SDX network.

Determine a method for distinguishing each business partner's instance of Subscriber Manager. The methods you can use depend on your Web application server. The simplest method is to use a different filename for each instance of Subscriber Manager.

If you want to configure Subscriber Manager to use workflows, you must install and configure the workflow engine and the OSMW. For information about installing the Workflow application, see *Chapter 1, Installing the SDX Applications*. For information about configuring the Workflow application, see *Chapter 5, Configuring the Workflow Application*.

Configuring Subscriber Manager

To install and configure Subscriber Manager:

1. Copy the *sub.war* file from the *webapp* folder on the application library CD to a temporary folder on your workstation.

```
mkdir tmp
cd tmp
cp /cdrom/cdrom0/webapp/sub.war .
```

2. Extract the *WEB-INF/web.xml* file from the *sub.war* file.

```
jar xvf sub.war WEB-INF/web.xml
```

3. Edit the *WEB-INF/web.xml* file with any text editor.

This XML file provides information about how Subscriber Manager communicates with the LDAP directory and the OSMW. The default file shows examples of the settings for the properties. See the descriptions following this procedure for information about the properties you must configure in this file.



NOTE: Residential subscribers can subscribe only to RADIUS services and value-added services. However, Subscriber Manager does not reject requests to subscribe residential subscribers either to access services or to outsourced services. If you want to block requests for these services, you must specify the objects *umcAccessServiceProfile* and *umcOutsourceServiceProfile* as *LdapExcludedClasses* in the file *web.xml*:

4. Repack the *sub.war* file.

```
jar uvf sub.war WEB-INF/web.xml
```

LdapHostUrl

Specifies the location of the primary directory

Value – URL of the directory that stores configuration information in the format:
ldap://< host> :< portNumber>

< host> – IP address or name of host that supports the Web application

< portNumber> – number of the TCP port

LdapBindDn

Specifies the DN that contains the username that the Web application server uses to connect to the directory

Value – < DN>

Default – *cn=umcadmin, o=umc*

LdapPassword

Specifies the password that the Web application server uses to connect to the directory

Value – text string

Default – admin123

LdapBaseDN

Specifies the root location in the directory

Value – < DN>

Default – *o=umc*

Retailer

Specifies the name of the business partner whose subscribers are managed by the gateway client

Value – text string

Guidelines – if one company provides both the retail and wholesale functions, use the name *default*, the default setting.

LdapServiceTreeRdn

Specifies the DN, relative to the value of the LdapBaseDN property, of the object that stores service templates

Value – < DN>

Default – *o=Services*

LdapSubscriberTreeRdn

Specifies the DN, relative to the value of the LdapBaseDN property, of the object that stores subscriber profiles

Value – < DN>

Default – *o=Users*

LdapStateManagerTreeRdn

Specifies the DN, relative to the value of the LdapBaseDN property, of the object that stores workflows

Value – < DN>

Default – *o= Workflows*

LdapServicePrefix

Specifies the prefix that is used to construct service profile DNs

Value – comma-separated list of services in the format:
serviceName= < serviceName>

< serviceName> – name of service in the directory

LdapRetailerPrefix

Specifies the prefix that is used to construct DNs for business partners

Value – retailername= < retailerName>

< retailerName> – name of business partner in the directory

LdapLocationPrefix

Specifies the prefix that is used to construct DNs of subscriber folders in the business partner's subtree.

Value – name of subscriber folder

Default – *ou= < siteName>*

LdapSubscriberPrefix

Specifies the prefix that is used to construct subscriber DNs.

Value – list of services separated by commands in the format:
uniqueID= < subscriberName>

< subscriberName> – name of subscriber in the directory

LdapStateMachinePrefix

Specifies the prefix that is used to construct RDNs of state machines

Value – stateMachineName= < stateMachineName>

< stateMachineName> – name of a state machine that the OSMW uses to control transactions

LdapExcludedClasses

Specifies object classes in the directory that Subscriber Manager cannot change

Value – comma-separated list of object classes

Default – TransactionalObjectAuxClass, parameterAuxClass,
umcOutsourceService, umcAccessServiceProfile, umcOutsourceServiceProfile

LdapExcludedAttributes

Attributes in the directory that Subscriber Manager cannot change

Value – comma-separated list of attribute names

Default is objectClass

OsmintHostUrl

Specifies the URL of the servlet that processes workflow requests (see *Chapter 11, Object State Manager for the Web*)

Value – URL in the format: http://< host> :< portNumber> /osmint

< host> – IP address or name of host that supports the OSMW

< portNumber> – number of the TCP port

OsmintSubscriberSM

Specifies the subscriber state machine

Value – name of the subscriber state machine

OsmintSubscriberSMInitState

Specifies the initial state of the subscriber state machine

Initial

OsmintSubscriptionSM

Specifies the subscription state machine

Value – name of the subscription state machine

OsmintSubscriptionSMInitState

Specifies the initial state of the subscription state machine

Default – Initial

ReporterServletUrl

Specifies the URL of the reporting function for the workflow utility (see *Chapter 13, Workflow Engine Functionality*).

Value – URL in the format: http://< host> :< portNumber> /subs/report

< host> – IP address or name of host on which Subscriber Manager is installed

< portNumber> – number of the TCP port

Guidelines – Add two entries for this property in the web.xml file (see the default *web.xml* file for examples).

WkfXmlDtdFile

Specifies the URL of the template that the workflow utility uses to create the XML document that lists the workflow (see *Chapter 7, Work Item Library*).

Value – URL in the format: `http:// < host> :< portNumber> /Wf-XML-1_0.dtd`

`< host>` – IP address or name of host specified for the `OsmintHostUrl` property

`< portNumber>` – number of the TCP port

Installing Subscriber Manager

To install Subscriber Manager:

1. Depending on the method you have chosen, identify the unique instance of Subscriber Manager. For example:

```
cp sub.war subPartner1.war
```

2. Deploy the `sub.war` file by following the procedure appropriate for the Web application server you are using.

```
cp subPartner1.war /opt/UMC/jboss/server/default/deploy
```

Sample Data

You can view sample state machines and workflows that you can use with Subscriber Manager in `o= Workflows, o= umc`. Table 10 lists the relevant state machines and workflows.

Table 10: Sample state machines and workflows for use with subscriber manager

Object Name	statemachineName= umcSubscriber
Description	State machine that manages subscribers
Object Name	statemachineName= umcSubscription
Description	State machine that manages subscriptions
Object Name	workflowName= GW< operation> < object> Workflow < operation> – one of the following operations: ChgBasic (subscription only) Create Modify Delete Fix (subscriber only) Suspend Resume < object> – Subscriber or Subscription
Description	Workflow that performs an operation specified by the variable < operation> on the object specified by the variable < object>
Example	workflowName= GWCreateSubscriberWorkflow

Testing Subscriber Manager

You can experiment with this application by installing Subscriber Manager and by writing a gateway client (see *Developing Gateway Clients* on page 54).

Developing Gateway Clients

When you have installed Subscriber Manager, you can access a Web Services Description Language (WSDL) file for the application. The WSDL file defines the public SOAP interface for Subscriber Manager. SDX owners and business partners can use this interface to develop gateway clients for Subscriber Manager.

The URL for the WSDL file is:

`http://< host> :< portNumber> /sub/services/SubscriberManagement?wsdl`

< host> – IP address or name of the host that supports Subscriber Manager

< portNumber> – number of the TCP port

Trace Requests

Gateway clients can also submit trace requests to OSMW via Subscriber Manager after OSMW has executed an action. The trace requests do not initiate OSMW actions. Table 11 lists the trace requests and their functions.

Table 11: Trace requests for Subscriber Manager

Trace request (operation)	Function
checkTransaction	Check whether the action for the previous request is complete.
abortTransaction	Cancel the action for the previous request.
traceTransaction	Trace all histories associated with the transaction ID for the previous request.
traceSubscriber	Trace all actions performed for a subscriber.
traceSubscription	Trace all actions performed for a subscription.

Managing Gateway Clients

This section contains information that SDX owners and their business partners need to manage gateway clients and their interactions with the gateway.

Error Codes That Subscriber Manager Returns

When Subscriber Manager receives a SOAP request it cannot handle, it returns an error code to the gateway client. Administrators of the gateway client can use this information to fix a problem with the client or to notify the gateway administrator of a problem with Subscriber Manager. Subscriber Manager returns error codes as strings to the gateway client. See Table 12.

Table 12: Error codes for Subscriber Manager

Error Code	Meaning	Action
SubscriberMgmt.SUBSMGMT_API_BAD_ARGUMENT	Gateway client passed an invalid argument.	Check that all arguments are correct.
SubscriberMgmt.SUBSMGMT_API_FORBIDDEN	Requested operation is forbidden. For example, a client requested to create a new subscriber whose unique ID is already used by another subscriber.	Check that the values are correct. If they are, contact the gateway administrator.
SubscriberMgmt.SUBSMGMT_API_INVALID_DATA	Gateway client passed an invalid value for an LDAP attribute. For example, the gateway client provided an incorrect string for a subscriber's IP address.	Check that attribute values are correct.
SubscriberMgmt.SUBSMGMT_API_INVALID_DN	Requested ID, such as subscriber ID or service ID, is invalid.	Check that the values are correct. If they are, contact the gateway administrator.
SubscriberMgmt.SUBSMGMT_API_LDAP_UNAVAILABLE	LDAP server is not available.	Contact the gateway administrator.
SubscriberMgmt.SUBSMGMT_API_LOCKED	Requested LDAP object is currently locked and cannot perform the requested operation.	Request the LDAP object later.
SubscriberMgmt.SUBSMGMT_API_TRANSPORT_FAILURE	OSMW is not available.	Contact the gateway administrator.
SubscriberMgmt.SUBSMGMT_API_BAD_TARGET	OSMW could not start a transaction for a given target.	Contact the gateway administrator.
SubscriberMgmt.SUBSMGMT_API_NOT_FOUND	Requested transactionID is not found in transaction history.	Contact the gateway administrator.

Results from OSMW Actions

Although the business partner does not know whether requests involve OSMW actions, Subscriber Manager does return the results of OSMW operations to the gateway client. Each result contains the following information:

Boolean completion indicator (operation completed or not)

Boolean success indicator (request successful or not)

Either a transactionID value (OSMW started action) or an errorMsg value

Table 13 shows the results that Subscriber Manager can return to the gateway client.

Table 13: Results that Subscriber Manager returns to gateway client

Completion Indicator	Success Indicator	Item Returned	Meaning
False	True	transactionID	OSMW action is executing the transaction.
False	False	errorMsg	Request failed.
True	False	errorMsg	Request failed and the OSMW action is not being executed, or subscriber manager is not configured to use the OSMW.
True	True	Nothing	Request is successful and completed, and subscriber manager is not configured to use the OSMW.

Error Codes for Trace Requests

When a gateway client submits a trace request, Subscriber Manager returns an error code of the type int error, and a string that describes the error code. Table 14 lists the error codes and their meanings.

Table 14: Error codes for trace requests

Error Code	Meaning
TraceResult.TRACE_RESULT_NO_ERROR	Trace was successful.
TraceResult.TRACE_RESULT_NO_SUCH_DATA	Given transactionID is not found in histories.
TraceResult.TRACE_RESULT_NO_SUCH_OBJ	Subscriber or subscription is not found in the trace history.
TraceResult.TRACE_RESULT_ABORT	Trace operation cancelled because of the reason shown in errorMsg