EUROPEAN COMMISSION

DIGIT
Digital Europe Programme

Service Metadata Publisher

Administration Guide

SMP 4.2 RC1

Version [3.4]

Status [Final]

Date: 02/06/2022

Document Approver(s):

Approver Name	Role
Bogdan DUMITRIU	Project Manager

Document Reviewers:

Reviewer Name	Role
Joze Rihtarsic	Developer
Caroline AEBY	TESO Support

Summary of Changes:

Version	Date	Created by	Short Description of Changes
0.1	22/02/2016	DHENEIN Christophe	Initial Document Creation
1.0	15/06/2017	Chaouki BERRAH (CEFS), Christophe DHENEIN (CEFS)	Document restructuring and updating.
1.1	16/06/2016	Chaouki BERRAH (CEFS), Christophe DHENEIN (CEFS)	3.0.0-RC2 changed to 3.X Screens updated.
1.2	12/02/2018	Chaouki BERRAH	Version 4.0 RC1 changes added
1.3	16/02/2018	Chaouki BERRAH	Update after test installation
1.4	21/02/2018	Chaouki BERRAH	Pawel changes included.
1.5	26/02/2018	Chaouki BERRAH	Links added
1.6	28/02/2018	Chaouki BERRAH Caroline AEBY	Updates
1.7	20/03/2018	CEF Support	Reuse notice added
1.8	11/06/2018	Chaouki BERRAH	Document update
1.9	28/06/2018	CEF Support	References to nexus updated.
2.0	01/10/2018	Caroline AEBY	No more standby service
2.1	18/10/2018	Chaouki BERRAH	Updates
2.2	23/10/2018	Caroline AEBY	SMP 4.1 RC – SMP admin console
2.3	25/10/2018	Joze Rihtarsic	SMP 4.1 Updates
2.4	15/11/2018	Joze Rihtarsic Chaouki BERRAH	Oracle Script Changes for Migration purposes
2.5	28/11/2018	Joze Rihtarsic Chaouki BERRAH	SMP 4.1 Updates
2.6	04/12/2018	Chaouki BERRAH	Weblogic and user creation Updates
2.7	19/09/2019	Chaouki BERRAH Gowtham VAITHIAM	Document Update
2.8	08/10/2019	Joze Rihtarsic	SMP 4.1.1 Release updates
2.9	06/12/2019	Caroline AEBY	Typo
3.0	03/03/2020	Chaouki BERRAH Gowtham VAITHIAM	MySql Connector V8 and default keystore and trustore comments.
3.1	23/03/2021	Joze Rihtarsic	Tested on Oracle 19c

3.2	11/04/2022	Caroline AEBY	No more CEF
3.3	19/05/2022	Caroline AEBY	CEF eDelivery FMB change to EC-EDELIVERY
3.4	01/06/2022	Caroline AEBY Joer RIHTARSIC	SMP 4.2 RC1

Table of Contents

1. INTRODUCTION

This Administration Guide is intended for Administrators who are in charge of installing, managing and troubleshooting an eDelivery SMP (Service Metadata Publisher).

1.1. Purpose

The purpose of this guide is to provide detailed information on how to deploy and configure an SMP 4.X on either a WebLogic or Tomcat Application Server with either MySQL or Oracle database.

It also provides detailed descriptions of the related Security Configurations (Certificates).

There is also a section on the use of Soap UI to create, update and delete SMP Service Groups and Metadata.

Another section describes an alternative method to perform the creation, update and deletions using Swagger UI.

2. CONVENTION

The Commands and Configuration files listed in this document usually contain a mix of reserved words (commands, instructions and system-related special words), user-defined words (chosen by the user) as well as comments and default/preferred values for some fields or variables.

The conventions used in this document, to distinguish between them, are the following:

- **Bold** is used for "reserved" words and commands.
- *Normal italic* together with a short description of the argument is used for user-defined names (chosen by yourself to designate items like users, passwords, database etc.). It normally contains at least 2 words separated by "_".
- ***Bold and italic*** is used for advisable values which can be changed by the user depending on their infrastructure.
- Comments are sometimes added to describe the purpose of the commands, usually enclosed in brackets ().
- By default, non-OS specific paths will be described using Linux patterns.

Example 1: Sample Oracle Statement:

create user *smp_user* **identified by** *smp_password*;

grant all privileges to *smp_user*;

(Where *smp_user* and *smp_password* are names chosen by the user)

Example 2: Sample Configuration File:

jdbc.driver = com.mysql.jdbc.Driver

jdbc.url = jdbc:mysql://localhost:3306/*smp_database*

jdbc.user = smp_user

jdbc.password = *smp_password*

target-database = MySQL

(Where: *smp_user, smp_database* and *smp_password* are names chosen by the user.

localhost:3306 represents hostname:port parameters of the MySQL database.)

3. PREREQUISITES

Please install the following software on the target system. For further information and installation details, please refer to the software owner's documentation.

- Java runtime environment is now (JRE) 8 **only**: http://www.oracle.com/technetwork/java/javase/downloads/index.html
- **One** of the supported Database Management Systems:
  - MySQL 5.6 or above
  - Oracle 10g+
- **One** of the supported Application Servers:
  - WebLogic 12.2.1.x (WebLogic 12.1 is not supported from 4.1.1 release on)
  - Tomcat 8

3.1. Binaries repository

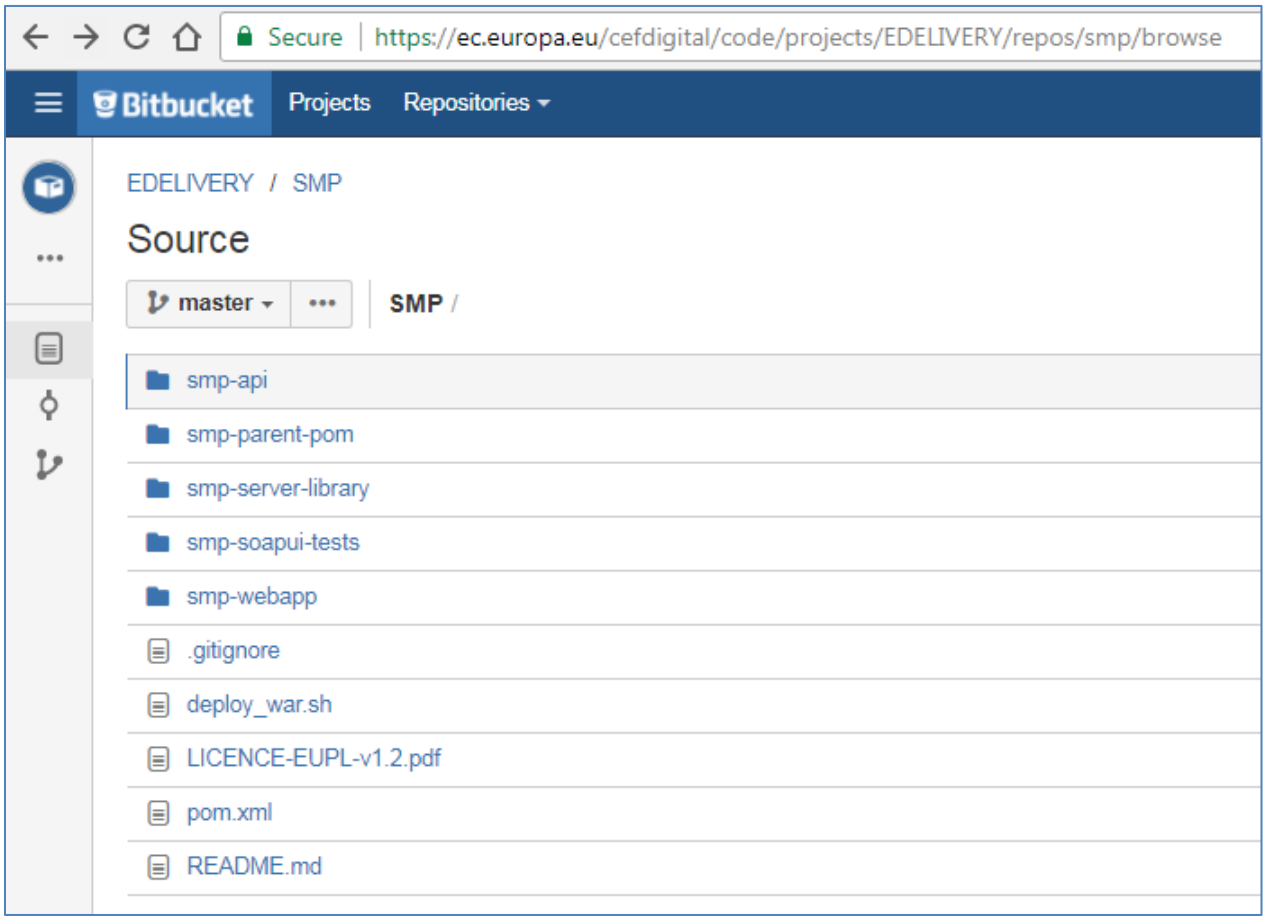
The SMP artefacts can be downloaded from the Digital site[1].

3.2. Source Code Repository

The source code of eDelivery SMP is available in the **GIT** repository at the following location:

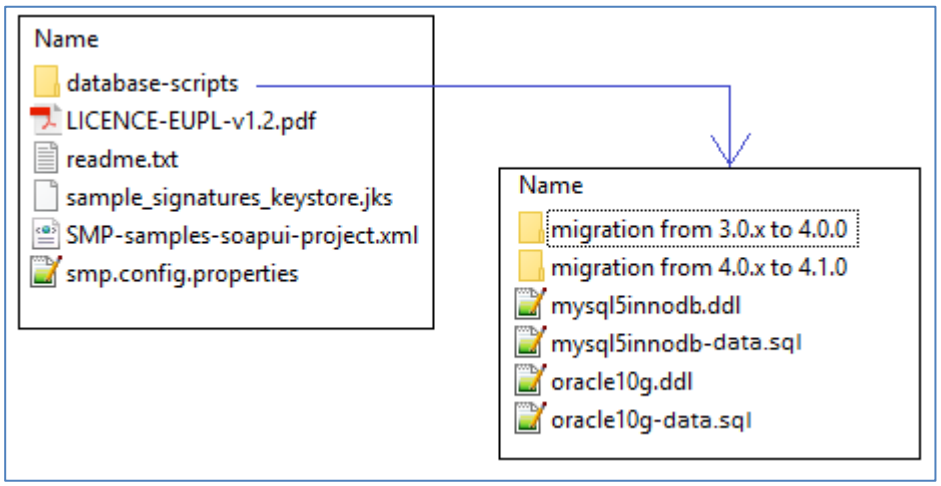
https://ec.europa.eu/digital-building-blocks/code/projects/EDELIVERY/repos/smp/browse

[1] https://ec.europa.eu/digital-building-blocks/wikis/display/DIGITAL/SMP+software



3.3. Database Scripts

The scripts to create (or migrate) the Oracle or MySQL databases are included in the following downloadable zip file from the Digital site (section §3.1): smp-4.x-setup.zip.



4. DEPLOYMENT

4.1. Deployment overview

As mentioned in the prerequisites, the deployment of the SMP is only supported on Tomcat or WebLogic application servers.

The deployment of the SMP on both platforms is almost identical but minor platform specific changes will be documented in a dedicated section of this manual.

The deployment of the SMP is summarized in the following mandatory steps:

- Database Configuration
- Application Server Preparation (Weblogic and Tomcat) for SMP
- SMP Initial Configuration
- SMP .WAR file Deployment

Remark:

*The environment variable, **cef_edelivery_path**, refers to the name of the folder where the SMP package is installed.*

*For Tomcat, it refers to **CATALINA_HOME**.*

*For Oracle WebLogic, it refers to **DOMAIN_HOME**.*

5. Database Creation

This section describes the steps necessary to create the database, tables and the SMP database user (**dbuser** used for database connection purpose).

It also includes the creation of an initial SMP user account that will be used by REST clients to connect to the SMP.

The SMP uses a direct connection to the database, which removes the need to configure a data source within WebLogic.

For this step you need to use the script included in the zip file downloaded in section §3.3.

5.1. MySQL

1. Download and copy the mysql5innodb-4.x.ddl script to cef_edelivery_path/sql-scripts
2. Open a command prompt and navigate to the cef_edelivery_path/sql-scripts folder
3. Execute the following MySQL commands:

```
mysql -h localhost -u root_user --password=root_password -e "drop schema if
exists smp_schema;create schema smp_schema;alter database smp_schema
charset=utf8; create user smp_dbuser@localhost identified by
'smp_password';grant all on smp_schema.* to smp_dbuser@localhost;"
```

This creates a *smp_schema* and an *smp_dbuser* with (all) privileges to the smp_schema**.**

Execute the following command to create the required objects (tables, etc.) in the database:

```
mysql -h localhost -u root_user -proot_password smp_schema < mysql5innodb.ddl
```

Execute the following command to fill initial test data:

```
mysql -h localhost -u root_user -proot_password smp_schema < mysql5innodb-
data.sql
```

5.2. Oracle Database

1. Download and copy the oracle10g.ddl **script** to *cef_edelivery_path***/sql-scripts**
2. Navigate to *cef_edelivery_path***/sql-scripts** directory
3. Execute the following commands :

```
sqlplus sys as sysdba   (password should be the one assigned during the Oracle
installation )

================================================================================
Once logged in Oracle:
create user smp_dbuser identified by smp_dbpassword;
```

```
grant all privileges to smp_dbuser;
connect smp_dbuser
show user;  (should return : smp_dbuser)
@oracle10g.ddl (run the scripts with the @ sign from the location of the scripts)

@oracle10g-data.ddl (Fill initial test data)

exit
================================================================================
```

6. ORACLE WEBLOGIC CONFIGURATION

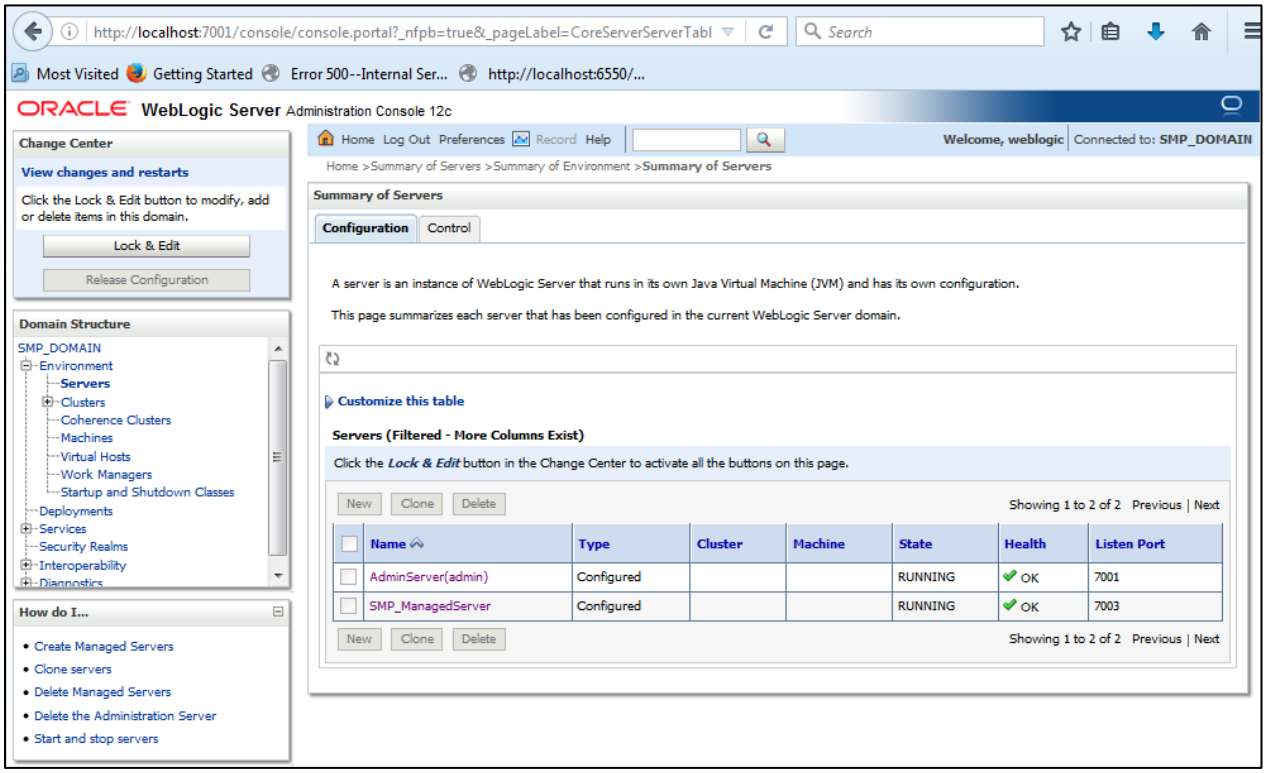
This section does not include the installation of a WebLogic application server. It is assumed that the WebLogic Server is installed and a WebLogic domain is created with an administration server and a managed server on which the SMP will be deployed.

Hereafter the domain location will be referred as *DOMAIN_HOME* (user-defined name).

In the examples below, we will use the following Domain and Server names:

- Domain Name : SMPDOMAIN
- Administration Server : AdminServer
- SMP Managed Server : SMP_ManagedServer

As shown below:



In order to deploy the SMP on the WebLogic Application Server platform, two preliminary steps need to be completed:

- Disabling the Authentication on the Weblogic Server,
- Configuring the Extra CLASSPATH for WebLogic,
- Setup sun HTTP Handler.

This is described in the following 2 sections.

6.1. Disabling the Authentication on the WebLogic

The eDelivery SMP has its own authentication mechanism that makes the WebLogic authentication redundant. It is therefore important to disable the WebLogic Authentication to stop it from interfering with the SMP authentication.

To do so, edit the config.xml file (under SMPDOMAIN/config) by adding the following tag before the `</security-configuration>` closing tag:

```
<enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-credentials>
```

Here is an example:

```
../
<enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-
credentials>
</security-configuration>
/..
```

6.2. Configuring the Extra CLASSPATH for WebLogic

Under the DOMAIN_HOME directory, create the following sub-directories:

- smp
- logs

Edit the WebLogic DOMAIN_HOME/bin/setDomainEnv.sh.

For Linux:

Add the **EXPORT CLASSPATH=${CLASSPATH}:${DOMAIN_HOME}/smp** statement at the end of the CLASSPATH definition as shown below:

```
../
if [ "${PRE_CLASSPATH}" != "" ] ; then
        CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${CLASSPATH}"
        export CLASSPATH
fi
        CLASSPATH=${CLASSPATH}:${DOMAIN_HOME}/smp
        export CLASSPATH
/..
```

For Windows:

```
../
If NOT "%PRE_CLASSPATH%"=="" (
      set CLASSPATH=%PRE_CLASSPATH%;%CLASSPATH%
)
set CLASSPATH=%CLASSPATH%;%DOMAIN_HOME%\smp
/..
```

6.3. Configuring Sun HTTP handler

Edit the WebLogic DOMAIN_HOME/bin/setDomainEnv.sh and add the following system parameter.

```
../
JAVA_OPTIONS=-DUseSunHttpHandler=true
export JAVA_OPTIONS
/..
```

7. TOMCAT CONFIGURATION

In order to deploy the SMP on Tomcat, the steps below need to be completed.

7.1. Configuring the Extra CLASSPATH for Tomcat

In this Tomcat example, a directory called **smp** will be created in the root path of the Tomcat installation (**CATALINA_HOME**) and the **CLASSPATH** modified to include this new directory using an existing Tomcat batch file (CATALINA_HOME/bin/setenv.[sh|bat]).

Create a **smp** directory in the **CATALINA_HOME** directory.

For Linux:

Edit the CATALINA_HOME/bin/setenv.sh file

```
#!/bin/sh
# Set CLASSPATH to include $CATALINA_HOME/smp
# where the smp 'smp.config.properties' is located
export CLASSPATH=$CATALINA_HOME/smp
```

For Windows:

Edit the %CATALINA_HOME%/bin/setenv.bat file

```
REM Set CLASSPATH to include $CATALINA_HOME/smp
REM where the 'smp.config.properties' is located
set classpath=%classpath%;%catalina_home%\smp
```

7.2. JDBC Driver

The JDBC driver needs to be downloaded from the manufacturer website:

- For Oracle Database : http://www.oracle.com/technetwork/apps-tech/jdbc-112010-090769.html
- For Mysql : https://www.mysql.com/products/connector/

The JDBC driver (.jar file) must be copied to the following directory: cef_edelivery_path/lib.

8. SMP CONFIGURATION

Since **SMP 4.1.0.FR**, configuration properties (except database connection configuration) are stored in the Database **SMP_CONFIGURATION** table. In order to make the migration easier for existing deployments, the initial database configuration is imported from the existing smp configuration file. Therefore, for a new installation, the initial procedure stays the same.

For this step, use the following resources delivered within the zip file downloaded in section §3.3:

- An smp config file named **smp.config.properties** located in the CLASSPATH

The **smp.config.properties** file must be copied to the CLASSPATH folder configured in chapter (see $7.1).

8.1. Database configuration

The eDelivery SMP database back-end configuration is performed within the eDelivery SMP configuration file (**smp.config.properties** file).

Depending on the selected database back-end, modify the **smp.config.properties** files as indicated below. Smp database connection can configured in property file or can use application server datasource configuration by JNDI.

8.1.1. *Oracle Database:*

- Datasource configured from property file:

```
../
## Sample for Oracle
jdbc.driver=oracle.jdbc.driver.OracleDriver
jdbc.url=jdbc:oracle:thin:@localhost:1521/xe
jdbc.user=smp
jdbc.password=secret123
hibernate.dialect=org.hibernate.dialect.Oracle10gDialect
/..
```

- Datasource configured on the application server

```
../
hibernate.dialect=org.hibernate.dialect.Oracle10gDialect
# weblogic datasource JNDI example
# datasource.jndi=jdbc/edeliverySmpDS
# tomcat datasource JNDI example
datasource.jndi=java:comp/env/jdbc/edeliverySmpDS
```

8.1.2. *MySQL:*

- Datasource configured from property file

```
../
## Database access
# For mysql connector v8
#jdbc.driver = com.mysql.cj.jdbc.Driver
# For mysql connector v5
jdbc.driver=com.mysql.jdbc.Driver
jdbc.url=jdbc:mysql://localhost:3306/smp
jdbc.user=smp
jdbc.password=secret123
hibernate.dialect =org.hibernate.dialect.MySQL5InnoDBDialect
/..
```

- Datasource configured on the application server

```
../
hibernate.dialect=org.hibernate.dialect.Oracle10gDialect
# weblogic datasource JNDI example
# datasource.jndi=jdbc/edeliverySmpDS
# tomcat datasource JNDI example
datasource.jndi=java:comp/env/jdbc/edeliverySmpDS
/..
```

8.2. SMP Keystore

eDelivery SMP uses keystore for storing keys for the two different purposes:

- One **mandatory** key is used for signing the responses to **GET** requests (XMLDSIG response signing)
- One **optional** key is used to authenticate SMP using 2-way-SSL when it is calling SML via HTTPS.

The keystore is automatically created, and a sample key is generated at the initial SMP start-up. The keystore can be managed with a System admin account using the UI under the page Domains/Edit keystore.

8.3. SMP Truststore

eDelivery SMP uses truststore for storing trusted X509Certificates for the WebService 2-way-SSL authentication and for storing the SML server certificate. The truststore is automatically created at the initial

SMP start-up. The truststore can be managed with a System admin account using the UI tools under the page Users/Edit truststore.

9. SMP .WAR FILE DEPLOYMENT

The eDelivery SMP is deployed using the steps described in the next sections.

9.1.1. *Tomcat*

Copy the cef_edelivery/smp/temp/ smp-4.X.war file to the Tomcat **webapps** directory (cef_edelivery/webapps/smp.war).

9.1.2. *Oracle WebLogic*

Deploy the **.war** file within WebLogic using the Oracle Weblogic deployer feature or using the Weblogic Administration Console.

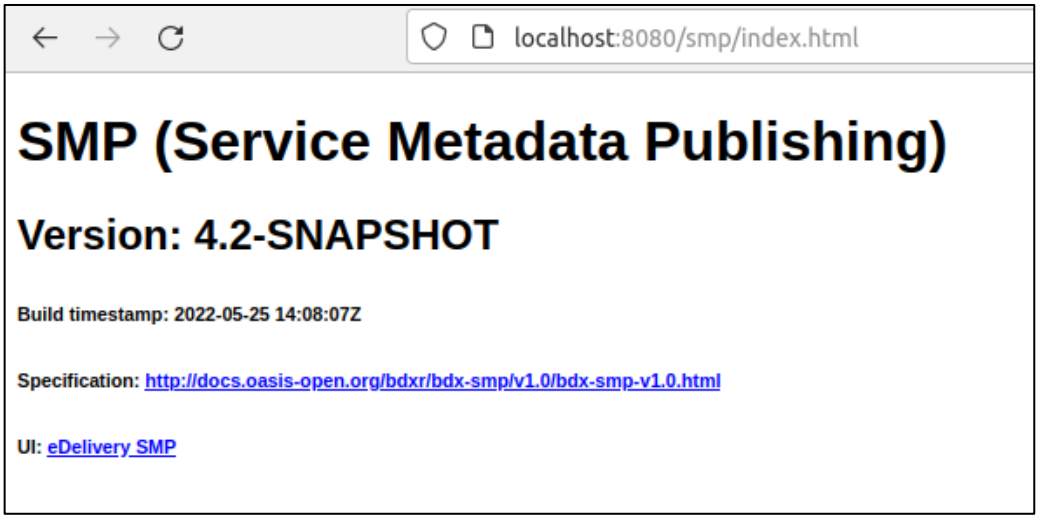
An example of using the Oracle the **weblogic.deployer,** is shown below:

```
java weblogic.Deployer -adminurl
t3://${WebLogicAdminServerListenAddress}:${WebLogicAdminServerPort} \
-username ${WebLogicAdminUserName} \
-password ${WebLogicAdminUserPassword} \
-deploy -name smp.war \
-targets ${SMP_ManagedServer} \
-source $TEMP_DIR/ smp.war
```

9.1.3. *Verification of the Installation*

Verify the installation by navigating with your browser to the following address:
http://*[hostname]*:*[port]*/smp

If the deployment is successful, the following page is displayed:



10. CONFIGURING THE EDELIVERY SMP FOR USE WITH AN SML

The eDelivery SMP can be registered in an SML, using two identification mechanisms:

- Using HTTP and plain text with metadata embedded into the header of the REST request,
- Using HTTPS/TLS and a keystore containing a certificate.

10.1. Configuring the BDMSL Integration

The first step is to configure the SMP so that it can be used with an SML.

This is achieved by setting the database property: **bdmsl.integration.enabled** parameter to **true**, default is **false**.

```
Execute the following sql command for Oracle database:UPDATE
SMP_CONFIGURATION SET VALUE='true', LAST_UPDATED_ON=systemdate WHERE
PROPERTY='bdmsl.integration.enabled';
```

10.2. Configuring the SML URL

The configuration of the SML URL endpoint is achieved by configuring the **bdmsl.integration.url** property in the **smp.config.property** file as follows:

```
UPDATE SMP_CONFIGURATION SET VALUE= http://localhost:8080/edelivery-sml/,
LAST_UPDATED_ON=systemdate WHERE PROPERTY= bdmsl.integration.url';
```

10.3. SMP authentication to an SML

Once registered in an SML, the SMP needs to authenticate against the SML during normal operations. For authentication, the client key/certificate must be installed in the keystore and an alias defined for the domain. The configuration is either done using the eDelivery SMP Admin console (see also §17) or by entering the values directly into the database.

SML supports two ways of authentication

- BlueCoat: HTTP client certificate header. This is used behind a bluecoat reverse proxy.
- HTTPS/TLS: Standard mutual TLS authentication.

10.3.1. *Plain Text HTTP*

SML_CLIENT_CERT_HEADER contains the SMP certificate needed if accessing SML directly through HTTP. The configured "Client-Cert" HTTP header will be added to each SML request bypassing SSL certificate verification made normally by SSL terminators.

SML_CLIENT_CERT_HEADER is the configuration attribute (column) in the **SMP_DOMAIN** table described in section §16.2.2.

10.3.2. *HTTPS/TLS*

When using HTTPS/TLS, as mentioned in the **bdmsl.integration.url** covered earlier, the SML Certificate server must be stored in the truststore. And the SMP TLS client key/certificate must be added to the Keystore. The TLS client key/certificate can be assigned to the domain using the UI tools under the Domain page.

11. SMP User Management

Only **Admin SMP** and Admin **ServiceGroup** users, who connect to the eDelivery SMP, need to be created in the SMP database.

Anonymous users or public users can access the SMP to retrieve only. They are not registered and therefore not added to the database.

There are no restrictions on the number of users that can be created to access the eDelivery SMP.

The database script creates the following users:

User name	Role	Default password [2]
system	SYSTEM_ADMIN	123456
smp	SMP_ADMIN	123456
user	SERVICE_GROUP_ADMIN	123456

11.1. User Roles

The eDeliery SMP users can be of three types, as briefly described below:

Actor	UC	Short description	Oper.	Data
Admin SMP	Create or Update Service Group	Create a new ServiceGroup for a new receiver participant. This service stores the Service Group and links it to the specified duplet participantIdentifier + participantIndentifierScheme. Information is stored into ServiceGroup table. This same service is used to create and update a ServiceGroup.	PUT	ServiceGroup
Admin SMP	Erase Service Group	Erases the service group definition AND the list of services for the specified receiver participant.	DELETE	ServiceGroup
Admin Service Group	Create or Update Service	Publish detailed information about one specific document service (multiple processes and endpoints). This same service is used to create and	PUT	ServiceMetadata

[2] to change immediately for security reasons

Actor	UC	Short description	Oper.	Data
	Metadata	update ServiceMetaData.		
Admin Service Group	Erase Service Metadata	Remove all information about one specific service (i.e. all related processes and endpoints definitions).	DELETE	ServiceMetadata
User	Retrieve Service Group	Obtain the list of services provided by a specific receiver participant (collection of references to the ServiceMetaData's). This service provides the information related to the Service Group according to the input duplet participantIdentifier + participantIndentifierScheme. Returns information from the ServiceMetadata table only (references to actual MetaData).	GET	ServiceGroup
User	Retrieve Service Metadata	Obtain detailed definition about one specific service of a specific participant for all supported transports. This service retrieves the SignedServiceMetadata according to the input quadruplet participantIdentifier+participantIndentifierScheme+ documentIdentifier+documentIdentifierScheme. Returns information from the Endpoint table.	GET	SignedServiceMetadata
System admin		Create, modify and delete users and domains. System admin can be only used in the edelivery SMP UI.		

Note: For a complete description of the SMP user management, please consult the SMP Interface Control Document (ICD) document available at https://ec.europa.eu/digital-building-blocks/wikis/display/DIGITAL/SMP.

Users can be added, modified and deleted using the SMP Admin console or directly by executing sql commands. Below are instructions on how to modify users in the database.

11.2. BCRYPT password generation

Since SMP 4.1.x users can be managed by SMP Admin console. Following procedure can be used for creating first system user.

The SMP v4.X uses the BCRYPT algorithm to hash users' passwords. A BCRYPT-hashing tool is bundled into the SMP WAR file. To get the hashing code, follow the steps below.

Place a copy of the **smp-4.X.war** file into a temporary directory of your choice.

Extract the war file using the **jar** command:

```
jar -xvf smp-4.X.war
```

Obtain one or multiple hashes at once, using the following command:

```
java -cp "WEB-INF/lib/*"
eu.europa.ec.edelivery.smp.utils.BCryptPasswordHash password_to_be_hashed
```

The result will be a BCRYPT hash of the specified password (listed below in italic):

```
java -cp "WEB-INF/lib/*"
eu.europa.ec.edelivery.smp.utils.BCryptPasswordHash 123456
Gives:
$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa
```

The next command shows the hashing of several passwords at once, separated by a space in the command.

```
java -cp "WEB-INF/lib/*" eu.europa.ec.edelivery.smp.BCryptPasswordHash
password_to_be_hashed_1 password_to_be_hashed_2
$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa
$2a$107zNzSeZpxiHeqY2BRKkHE.HknfIe3aiu6XzU.qHHnnPbUHKtfcmDG
```

11.3. SMP Database User Creation

Adding an SMP user is done by adding a new entry in the SMP database **SMP_USER** table either directly or via the Administration console.

The User role is set in the SMP_USER table ROLE column as follows:

User Role	Role value
Admin SMP	SMP_ADMIN
Admin Service Group	SERVICE_GROUP_ADMIN
System Administrator	SYSTEM_ADMIN
AnonymousUser (Not defined in the SMP User database)	N/A

In the following two examples, an **Admin SMP** and an **Admin ServiceGroup** users are created.

11.3.1. *SYSTEM_ADMIN SMP User creation*

Remark:

- In order to logon on the Administration Console **for the first time**, it is necessary to, first create a user with **SYSTEM_ADMIN** privileges by entering the details directly into the **SMP_USER** table. This initial user's password is generated using the **BCRYPT** utility described previously.

- *If PASSWORD_CHANGED is not set, the user will be asked to change the password at first logon.*

Example of a SYSTEM_ADMIN user creation:

Username : smp_admin

Password (Hashed) : $2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa

Role : SYSTEM_ADMIN

Execute the following database command using the database user/password created in the Database Configuration section of this guide.

MySql example:

```
INSERT into SMP_USER (ID, USERNAME, PASSWORD, ROLE, ACTIVE,
CREATED_ON,LAST_UPDATED_ON,PASSWORD_CHANGED) values ((select max(next_val) from
SMP_USER_SEQ), 'smp_admin',
'$2a$10$olcGeWKGEoRia2DPuFqRNeca0IEdRSmOrljLz57BAjf1jlC9SohrS',
'SYSTEM_ADMIN',1,SYSDATE(), SYSDATE(),SYSDATE() );

update SMP_USER_SEQ set next_val=next_val+1;
```

Oracle example:

```
SQL> INSERT into SMP_USER (ID, USERNAME, PASSWORD, ROLE, ACTIVE,
CREATED_ON,LAST_UPDATED_ON) values (SMP_USER_SEQ.NEXTVAL, 'smp_user',
'$2a$10$olcGeWKGEoRia2DPuFqRNeca0IEdRSmOrljLz57BAjf1jlC9SohrS',
'SYSTEM_ADMIN',1,SYSDATE, SYSDATE );

update SMP_USER_SEQ set next_val=next_val+1;
```

11.3.2. *Admin SMP User Creation*

Username : smp_user1

Password (Hashed) : $2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa

IsAdmin : SERVICE_GROUP_ADMIN

Example: Oracle database command.

```
SQL> insert into smp_user (ID, username, password, ROLE, ACTIVE,
CREATED_ON,LAST_UPDATED_ON) values (SMP_USER_SEQ.NEXTVAL, 'smp_user1',
'$2a$10$6nytsush2bqfboliycxn8euvibcnn.wcjurwotjlmndodati85zma',
'SMP_ADMIN',1,SYSDATE, SYSDATE);
```

11.3.3. *Admin ServiceGroup User Creation*

Username : smp_user2

Password (Hashed) : $2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa

IsAdmin : SERVICE_GROUP_ADMIN

Example: Oracle database command.

```
SQL> insert into smp_user (ID, username, password, ROLE, ACTIVE,
CREATED_ON,LAST_UPDATED_ON) values (SMP_USER_SEQ.NEXTVAL, 'smp_user1',
'$2a$10$6nytsush2bqfboliycxn8euvibcnn.wcjurwotjlmndodati85zma',
'SERVICE_GROUP_ADMIN',1,SYSDATE, SYSDATE);
```

12. LOGGING CONFIGURATION

12.1. Logging properties

The SMP logging properties are defined in the ***./WEB-INF/classes/logback.xml*** file embedded in the SMP **.war** file.

It is possible to modify the configuration of the logs by editing the embedded ***logback.xml*** or by defining new logback file in **smp.config.properties** file as example:

```
log.configuration.file=/opt/apache-tomcat-8.5.30/smp/logback.xml
```

In the example below, a ***logback.xml*** file is shown:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <!-- pattern definition -->
    <property name="encoderPattern" value="%d{ISO8601} [%X{smp_user}] [%X{smp_session_id}] [%X{smp_request_id}] [%thread] %5p %c{1}:%L - %m%n" scope="global"/>
    <property name="consolePattern" value="%d{ISO8601} [%X{smp_user}] [%X{smp_session_id}] [%X{smp_request_id}] [%thread]  %5p %c{1}:%L - %m%n" scope="global"/>

     <appender name="file" class="ch.qos.logback.core.rolling.RollingFileAppender">
        <file>${log.folder:-logs}/edelivery-smp.log</file>
        <filter class="ch.qos.logback.core.filter.EvaluatorFilter">
            <evaluator class="ch.qos.logback.classic.boolex.OnMarkerEvaluator">
                <marker>SECURITY</marker>
                <marker>BUSINESS</marker>
            </evaluator>
            <onMismatch>NEUTRAL</onMismatch>
            <onMatch>DENY</onMatch>
        </filter>
        <rollingPolicy class="ch.qos.logback.core.rolling.SizeAndTimeBasedRollingPolicy">
            <!-- rollover daily -->
            <fileNamePattern>${log.folder:-logs}/edelivery-smp-%d{yyyy-MM-dd}.%i.log</fileNamePattern>
            <!-- each file should be at most 30MB, keep 60 days worth of history, but at most 20GB -->
            <maxFileSize>30MB</maxFileSize>
            <maxHistory>60</maxHistory>
            <totalSizeCap>20GB</totalSizeCap>
        </rollingPolicy>
        <encoder>
            <pattern>${encoderPattern}</pattern>
        </encoder>
    </appender>
    <appender name="stdout" class="ch.qos.logback.core.ConsoleAppender">
        <Target>System.out</Target>
        <encoder>
            <pattern>${consolePattern}</pattern>
        </encoder>
    </appender>

    <logger name="eu.europa.ec.edelivery.smp" level="INFO" />
    <logger name="org.springframework.security.cas" level="DEBUG" />
    <root level="DEBUG">
        <appender-ref ref="file"/>
        <appender-ref ref="stdout"/>
    </root>
</configuration>
```

More details on how to confugure logback can be found at:

https://logback.qos.ch/documentation.html

13. SOAPUI TESTING

Soap UI can be used to create, update and delete Service Groups and Metadata.

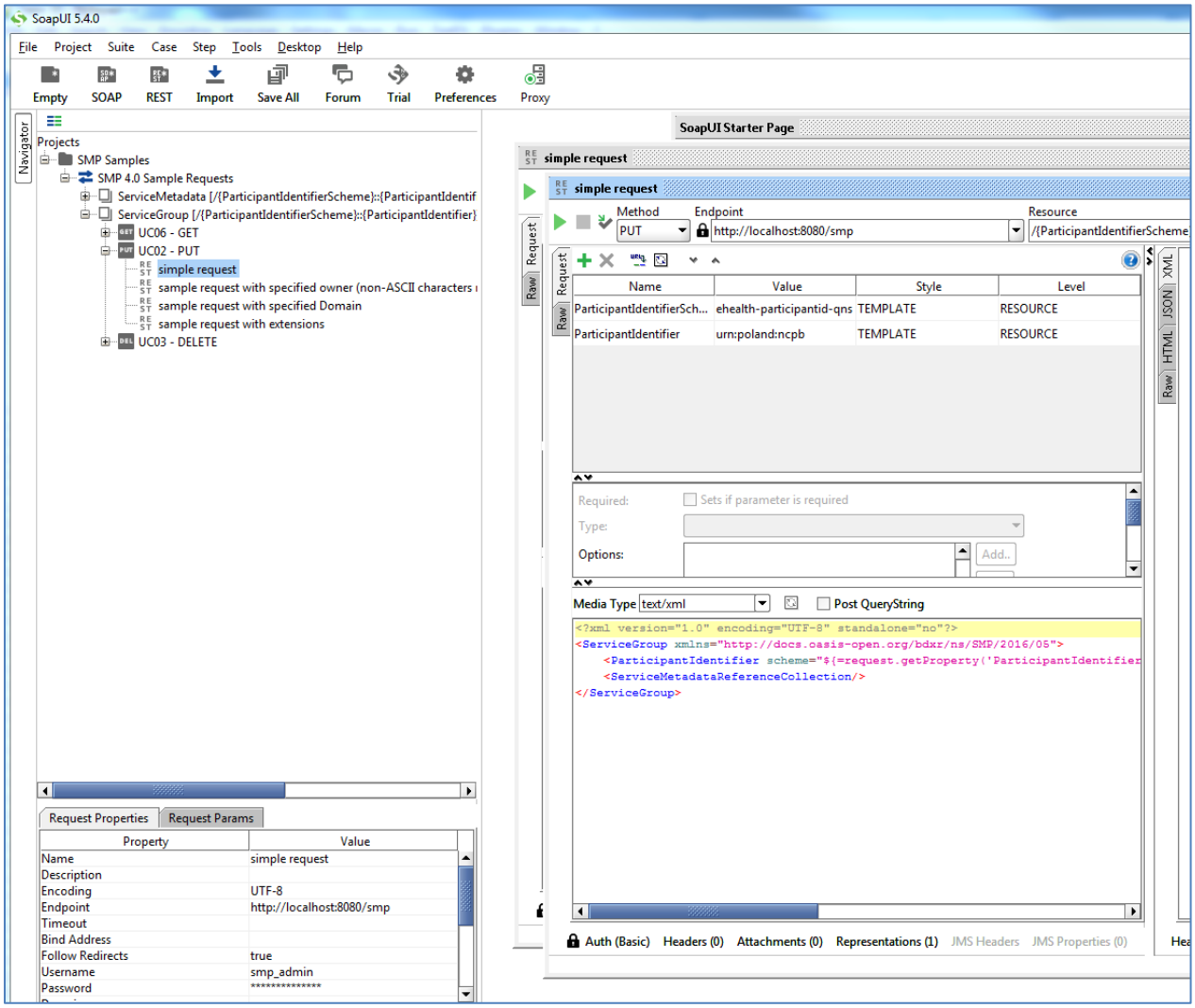
An SMP SoapUI project contains sample requests and is included in the zip file already downloaded.

The procedure to create, update or delete a Service Group is described in the next steps.

13.1. Creation, update and deletion of Service Groups.

13.1.1. *Create a Service Group*

In the left navigation pane of the SoapUI interface, browse to the REST PUT method as shown below:

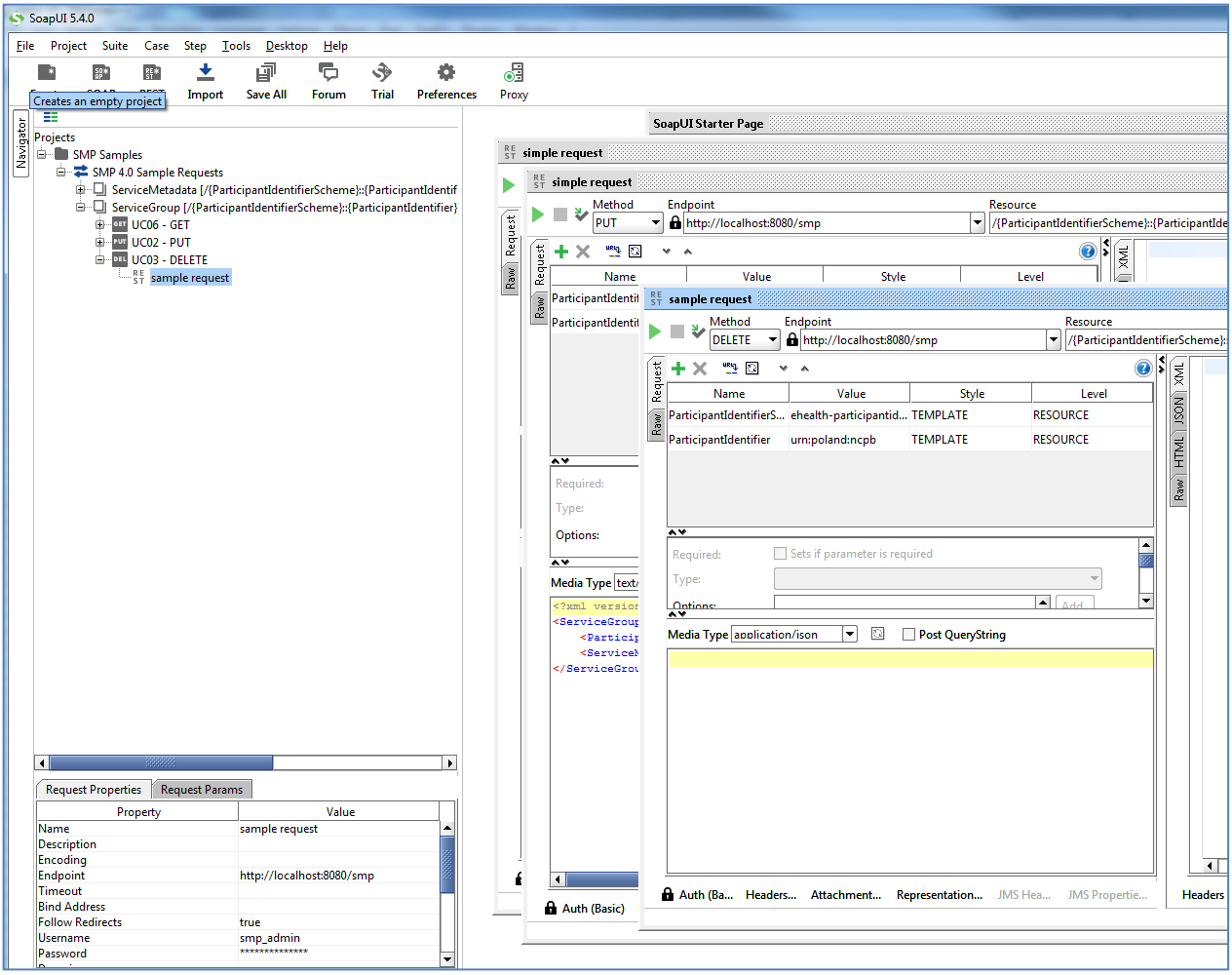


13.1.2. Update a Service Group

The REST method to update the **ServiceGroup** is the same as the one used for creating **ServiceGroup** described in the previous section.

13.1.3. Delete a ServiceGroup

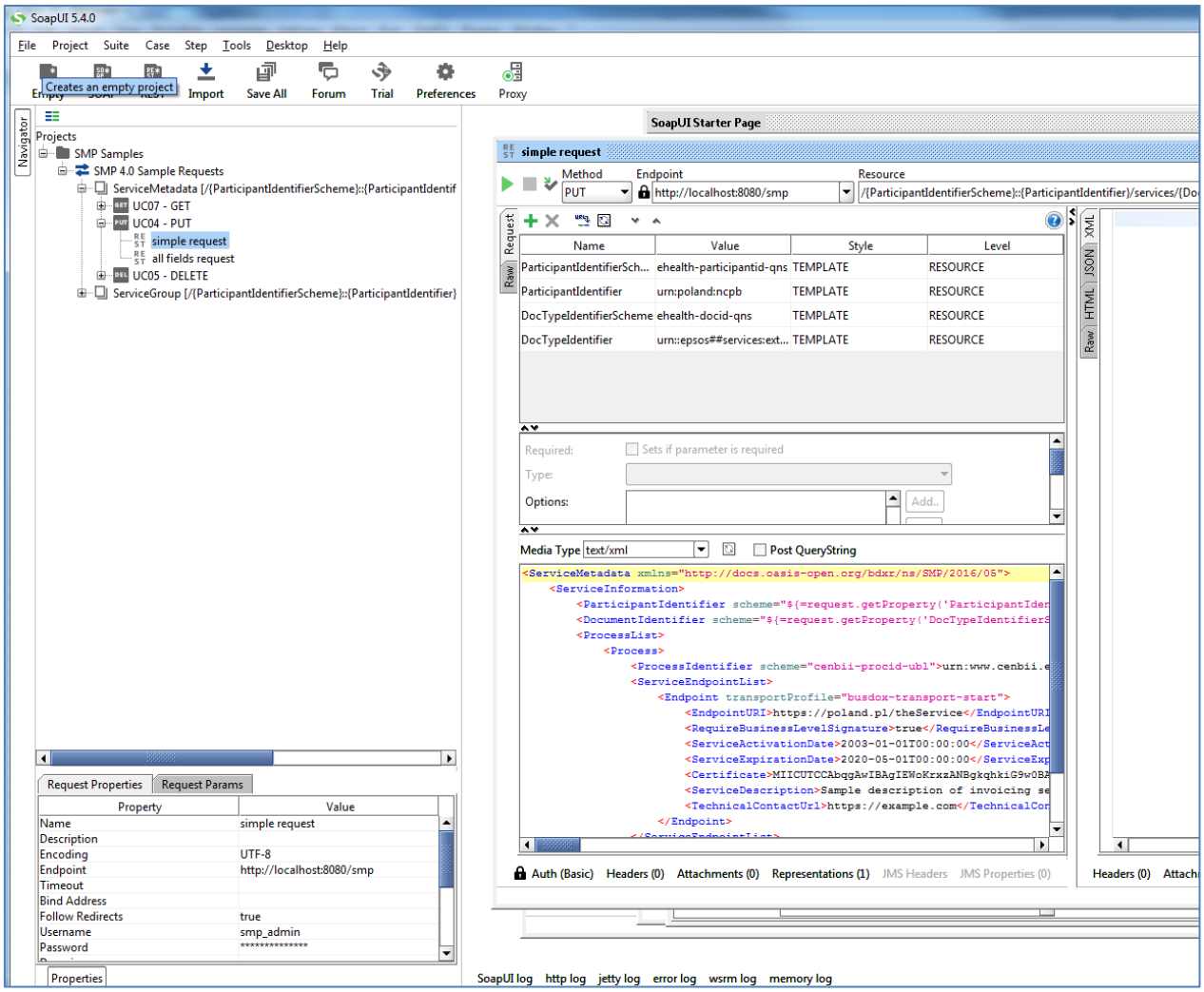
On the SoapUI interface on the left navigation panel, browse to the REST DELETE method as indicated below:



13.2. Creation, update and deletion of Service Metadata.

13.2.1. Create a Service Metadata

In the left navigation pane of the SoapUI interface, browse to the REST PUT method as shown below:

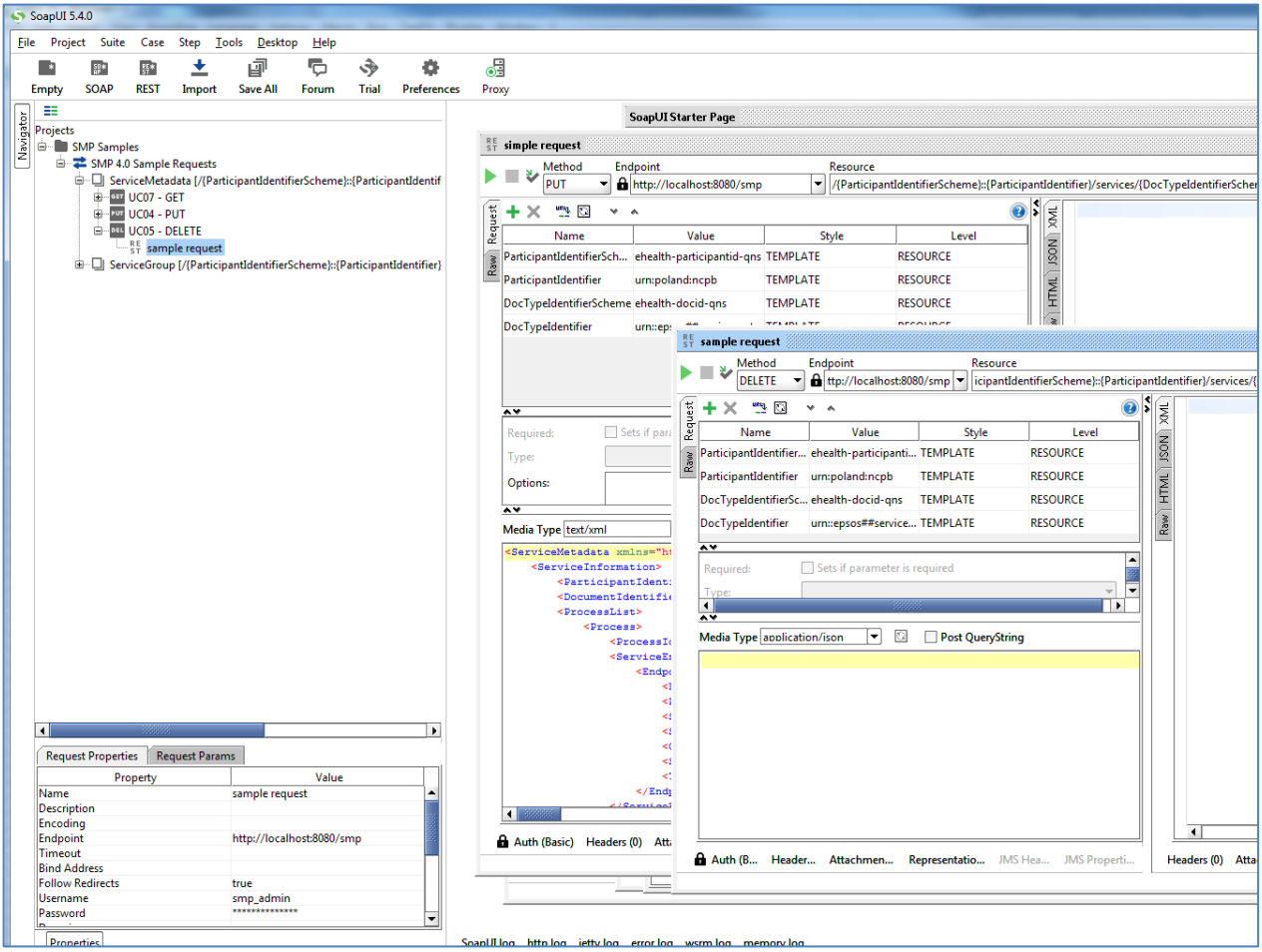


13.2.2. *Update Service Metadata*

The REST method to update **ServiceMetadata** is the same as the one use for creating **ServiceMetadata** as described in the previous section.

13.2.3. *Delete Service Metadata*

In the left navigation pane of the SoapUI interface, browse to the **REST DELETE** method as indicated below:



14. THE SWAGGERUI INTERFACE

14.1. Introduction

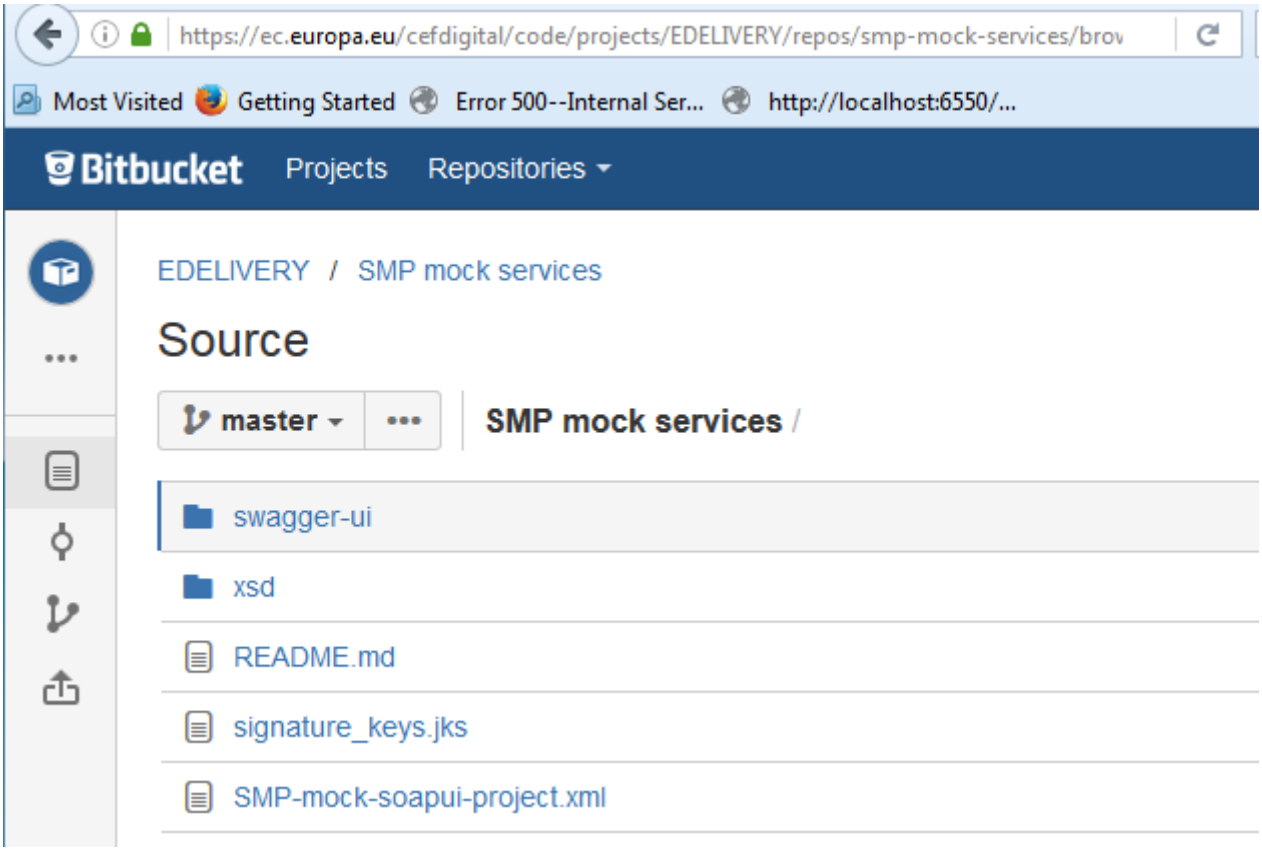
"***Swagger is an API developer tools for the OpenAPI Specification (OAS). It allows anyone (developers or end-users) to interact with the API's resources***"[3].

The SMP Web Client can be tested at: http://130.206.118.4/smp-swagger-ui and, as explained, is a WEB client configured to shoot (PUT, GET or DELETE) at the mocked SMP implementation Metadata.

14.2. Downloading the eDelivery SMP SwaggerUI web application project

The eDelivery SMP SwaggerUI web application project can be freely downloaded from the following location:

https://ec.europa.eu/digital-building-blocks/code/projects/EDELIVERY/repos/smp-mock-services/browse



Create a new **swagger_temp** temporary directory.

Within the previously created **swagger_temp** directory, execute the following command:

[3] Quote from: http://swagger.io/.

```
git clone https://ec.europa.eu/cefdigital/code/scm/edelivery/smp-mock-services.git
Cloning into 'smp-mock-services'...
remote: Counting objects: 133, done.
remote: Compressing objects: 100% (130/130), done.
remote: Total 133 (delta 50), reused 0 (delta 0)
Receiving objects: 100% (133/133), 823.54 KiB | 0 bytes/s, done.
Resolving deltas: 100% (50/50), Done.
```

The SMP **SwaggerUI** project is downloaded and saved the **smp-mock-services** directory:

```
ls
smp-mock-services
```

14.3. Configuring the SMP SwaggerUI

Navigate to the **swagger-ui** directory located under the **smp-mock-services** directory.

The contents is listed below:

```
ls
css  fonts  images  index.html  lib  smp.json  swagger-ui.js
```

Edit the **smp.json** file and modify it to target your SMP:

Replace:

```
{
  "swagger": "2.0",
  "info": {
    "description": "This WEB client is configured to shoot at the [mocked SMP implementation](http://smp-digit-mock.publisher.ehealth.acc.edelivery.tech.ec.europa.eu/ehealth-actorid-qns%3A%3Aurn%3Apoland%3Ancpb). After a few improvements (both on client and server side) it might be used also for shooting at TEST / PROD environments. You can find out more about Swagger at [http://swagger.io](http://swagger.io)",
    "version": "1.0.0",
    "title": "SMP 3.X WEB client (based on Swagger-UI)"
  },
  "host": "smp-digit-mock.publisher.ehealth.acc.edelivery.tech.ec.europa.eu",
  "basePath": "/",
  "externalDocs": {
    "description": "Find out more about SMP 3.X mock services",
```

With:

```
    "url": "https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp-mock-services"
{
  "swagger": "2.0",
  "info": {
    "description": "This WEB client is configured to shoot at [http://localhost:7003/ smp-4.X](http://localhost:7003/ smp-4.X).  After a few improvements (both on client and server side) it might be used also for shooting at TEST / PROD environments. You can find out more about Swagger at [http://swagger.io](http://swagger.io)",
    "version": "1.0.0",
    "title": "SMP 4.X WEB client (based on Swagger-UI)"
  },
  "host": "localhost:7003",
  "basePath": "/ smp-4.X",
  "externalDocs": {
  },
```

14.4. Generating the Web Application Archive (.war file)

To generate the eDelivery SMP SwaggerUI Web Application archive (**.war** file), jus create a zip file of the content of the swagger-ui directory and rename it as **swagger.war**.

This can be performed using any **zip** utility (**winzip** on Windows or **zip** on Linux).

Example on Linux:

```
zip -r swagger.war swagger-ui/*
```

14.5. Deploy the SMP SwaggerUI war file

14.5.1. *On Tomcat*

Copy the **swagger.war** file to *cef_edelivery_path* **/webapps.**

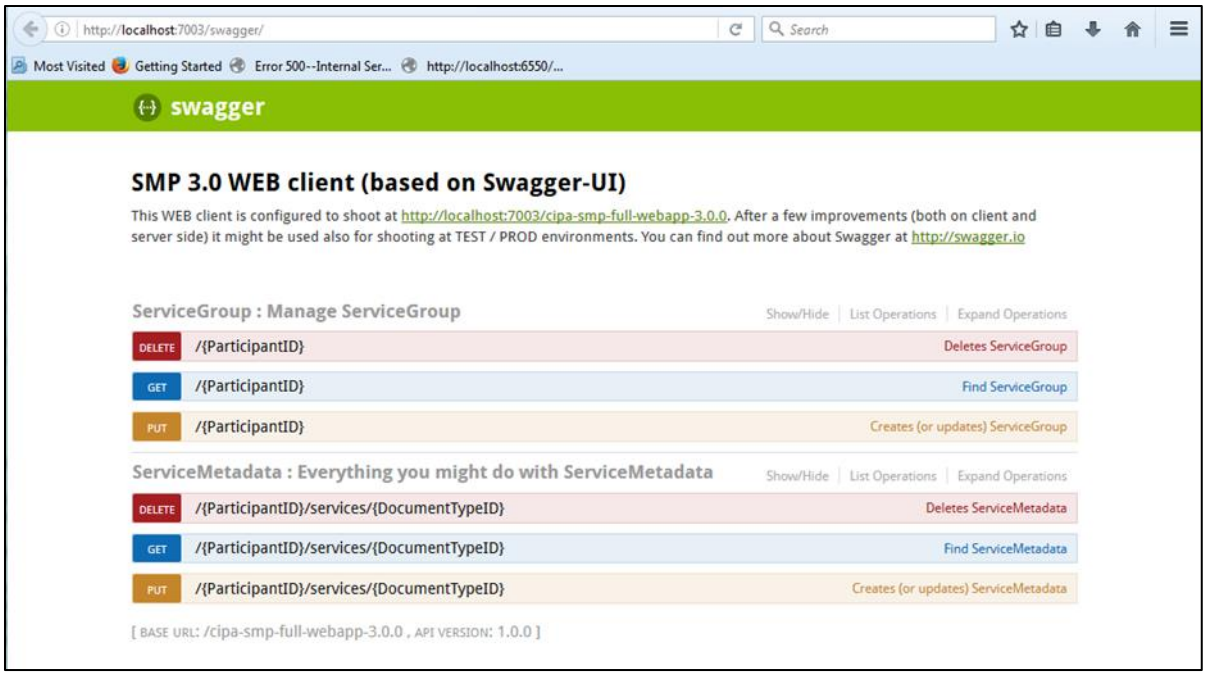
14.5.2. *On WebLogic:*

Deploy the **.war** file within WebLogic:

```
java weblogic.Deployer -adminurl
t3://${WebLogicAdminServerListenAddress}:${WebLogicAdminServerPort} \
-username ${WebLogicAdminUserName} \
-password ${WebLogicAdminUserPassword} \
-deploy -name swagger.war \
-targets ${SMP_ManagedServer} \
```

After starting the application, connect to http://locxalhost:7003/swagger.

A successful deployment should display the following page:



15. SMP COMPILATION

15.1. Compilation prerequisites

15.1.1. *Supported Operating System Platform*

The eDelivery SMP can be built on the following OS platforms:

- Windows Workstation & Server
- Linux platform

15.1.2. *Software Requirements*

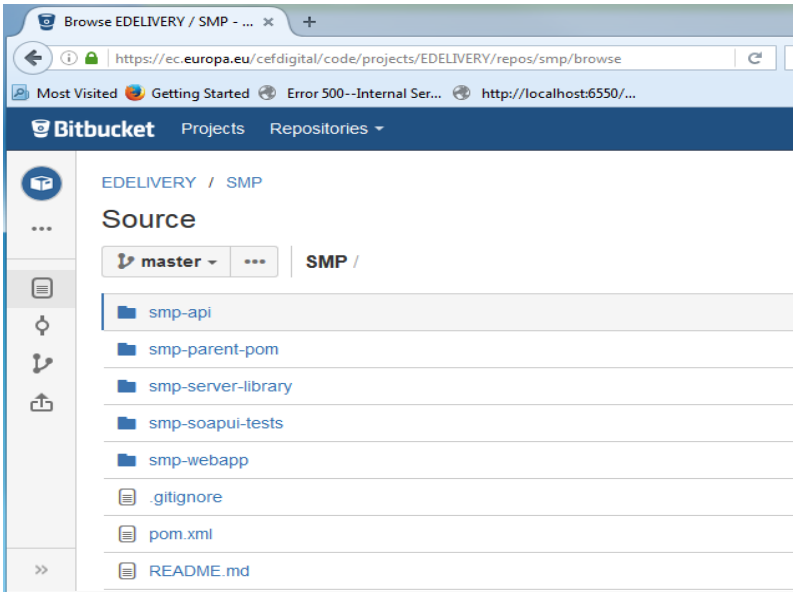
The following software components on the target system:

- Java Development Kit environment (JDK), version 7 or 8: http://www.oracle.com/technetwork/java/javase/downloads/index.html
- Maven 3.0 and above (https://maven.apache.org/download.cgi)
- GIT (optional: Git is only used to download the project sources but these sources can be downloaded from any system having Git installed and then just copied manually on the compilation platform).

15.2. Downloading the source code

The source code of SMP is freely available and can be downloaded from the following location:

https://ec.europa.eu/digital-building-blocks/code/projects/EDELIVERY/repos/smp/browse



15.3. Compilation

Create a new **comp_dir** temporary directory.

Within the previously created **comp_dir** directory, execute the following command:

```
git clone  https://ec.europa.eu/digital-building-blocks/code/scm/edelivery/smp.git
Cloning into 'smp'...
remote: Counting objects: 52788, done.
remote: Compressing objects: 100% (15640/15640), done.
remote: Total 52788 (delta 25293), reused 47993 (delta 23387)
Receiving objects: 100% (52788/52788), 637.14 MiB | 2.06 MiB/s, done.
Resolving deltas: 100% (25293/25293), done.
```

Go to the newly created **smp** directory.

The directory contains the following:

```
ls
pom.xml  README.md  smp-api  smp-parent-pom  smp-server-library  smp-soapui-tests  smp-webapp
```

Start the compilation by executing the following command:

```
mvn clean install -DskipTests
```

A successful compilation will result with the following:

```
mvn clean install -DskipTests
[INFO] Scanning for projects…
/..
../
[INFO] Installing /home/smpcomp/smp/smp/pom.xml to /home/smpcomp/.m2/repository/eu/europa/ec/smp/3.X/smp-3.X.pom
[INFO] ------------------------------------------------------------------------
[INFO] Reactor Summary:
[INFO]
[INFO] smp-angular ..................................... SUCCESS [132.375 s]
[INFO] smp-api ......................................... SUCCESS [ 32.375 s]
[INFO] smp-server-library .............................. SUCCESS [02:01 min]
[INFO] smp-webapp ...................................... SUCCESS [ 23.314 s]
[INFO] SMP Builder POM ................................. SUCCESS [  2.222 s]
[INFO] ------------------------------------------------------------------------
```

```
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 03:00 min
[INFO] Finished at: 2017-06-08T11:35:27+02:00
[INFO] Final Memory: 61M/726M
[INFO] ------------------------------------------------------------------------
```

The resulting will be a Web application Archive (.war file) named **smp.war** located in the **smp-webapp/target/** directory:

```
ls ./smp-webapp/target
smp-4.X  smp.war classes  generated-sources  generated-test-sources  maven-status  test-classes  webapp-classes
```

16. SMP CONFIGURATION FILE AND TABLE

16.1. Multitenancy and Multidomain Support

The SMP is able to support multiple certificates in the same SMP. This is very useful in the Acceptance environment where multiple domains like ISA ITB, eHealth and others are hosted.

The SMP has the capability of keeping a relationship between a particular **Service Group** and its related **domain**.

As a result of this feature, the SMP Administration has the option, if need be, to define extra domains for newly created **Service Groups** meaning that the SMP is able to handle multiple domains environments.

Remark:

*In normal circumstances, when any one SMP is used for only one Domain, the domain used is then considered as the "domain by default" (or "default domain") for configuration purposes. The domain, in this case, does not need to be specified in the **Service Group** definitions or other configurations of the SMP as in previous versions of SMP.*

The SMP configuration is performed in 2 different locations: in the **smp.config.properties** file as well as in the **smp_configuration** table. The following section describes the details of the parameters that are included in the configuration.

16.2. The smp.config.properties file

The eDelivery SMP configuration is performed via the **smp.config.properties** file.

The initial eDelivery SMP configuration is performed via the **smp.config.properties** file. The file contains basic configuration for defining the database connection, logging file configuration and smp folder for deploying the extensions.

This file is delivered by default embedded within the SMP war file.

```
#
# Copyright 2018 European Commission | CEF eDelivery
#
# Licensed under the EUPL, Version 1.2 or - as soon they will be approved
by the European Commission - subsequent versions of the EUPL (the
"Licence");
# You may not use this work except in compliance with the Licence.
#
# You may obtain a copy of the Licence attached in file: LICENCE-EUPL-
v1.2.pdf
#
```

```
# Unless required by applicable law or agreed to in writing, software
distributed under the Licence is distributed on an "AS IS" basis,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the Licence for the specific language governing permissions and
limitations under the Licence.
#


#
********************************************************************************
# Database connection can be achieved using custom datasource
configuration
# or reusing application server datasource.
#
********************************************************************************
## set database hibernate dialect
#hibernate.dialect=org.hibernate.dialect.Oracle10gDialect
hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect


# *********************************
#  Custom defined datasource
# *********************************
# mysql database example
jdbc.driver = com.mysql.jdbc.Driver
jdbc.url = jdbc:mysql://localhost:3306/smp
jdbc.user = smp
jdbc.password=secret123


# Oracle database example
#jdbc.driver = oracle.jdbc.driver.OracleDriver
#jdbc.url=jdbc:oracle:thin:@localhost:1521/xe
#jdbc.user=smp
#jdbc.password=secret123



# *********************************
# Datasource JNDI configuration alternative
# *********************************
# weblogic datasource JNDI example
# datasource.jndi=jdbc/eDeliverySmpDs
```

```
# tomcat datasource JNDI example
# datasource.jndi=java:comp/env/jdbc/eDeliverySmpDs



# ********************************
# Logging properties
# ********************************
# smp log folder
log.folder=../logs/

# custom log4j configuration file
# log.configuration.file=smp-logback.xml

# ********************************
# Extension folder
# ********************************
# path where SMP extensions are located. The Folder is loaded by the SMP classloader at startup.
libraries.folder=/cef/test/smp/apache-tomcat-8.5.73/smp/ext-lib
```

16.2.1. *SMP configuration properties (smp.config.properties)*

The **WEB-INF/classes/smp.config.properties** file is used to configure initial SMP properties needed for the SMP startup.

The following table describes them briefly:

Parameter	Default Value	Comment
jdbc.driver	com.mysql.jdbc.Driver	Database Configuration: Driver MySQL: com.mysql.jdbc.Driver Oracle Database: oracle.jdbc.OracleDriver
jdbc.url	jdbc:mysql://localhost:3306/smp	Database Configuration: url MySQL : jdbc:mysql://dbhost:dbport/smp_database Oracle Database: jdbc:oracle:thin:@dbhost:dbport:smp_database jdbc:oracle:thin:@dbhost:dbport/smp_service
jdbc.user	smp	Database User/Password Configuration: User
jdbc.password	The_password	Database User/password Configuration: Password
target-database	MySQL	Target Database Backend type/Brand: For MySQL, use: MySQL For Oracle Database, use: Oracle

Parameter	Default Value	Comment
jdbc.read-connections.max	10	Database Configuration: Max Read Connection
datasource.jndi	jdbc/eDeliverySmpDs	If the data source is configured on the application server (recommended), the property defines the JNDI name of the database connection.
log.folder	/var/logs/smp	The provided logback.xml configuration defines logging file as <file>${log.folder:-logs}/edelivery-smp.log</file> With the property we can define the folder for the logging files.
log.configuration.file	/opt/logging/smp-logback.xml	custom logback configuration file (filepath can be absolute or relative to smp configuration.dir.
libraries.folder	/opt/smp/extension-libs	Path where SMP extensions are located. The Folder is loaded by the SMP classloader at startup.

16.2.2. *SMP application configuration (database table SMP_CONFIGURATION)*

eDelivery SMP application configuration values are stored in the database table ***SMP_CONFIGURATION***. If the table is empty (usually at first SMP startup), edelivery SMP populates the table at startup with all properties and default values. When updating properties via the user interface, the property values are taken into account immediately if the server starts in non-cluster mode (property***: smp.cluster.enabled = false***). Otherwise, each node refreshes the properties on all cluster nodes at the same time) according to the property refreshes the cron expression in the property: ***smp.property.refresh.cronJobExpression.***

Parameter	Default Value	Comment	Restart needed	Value type
contextPath.output	true	This property controls pattern of URLs produced by SMP in GET ServiceGroup responses.	true	BOOLEAN
encodedSlashesAllowedInUrl	true	Allow encoded slashes in context path. Set to true if slashes are are part of identifiers.	true	BOOLEAN
smp.http.forwarded.headers.enabled	false	Use (value true) or remove (value false) forwarded headers! There are security considerations for forwarded headers since an application cannot know if the headers were added by a proxy, as intended, or by a malicious client.	false	BOOLEAN
smp.http.httpStrictTransportSecurity.maxAge	31536000	How long (in seconds) HSTS should last in the browser's cache (default one year)	true	INTEGER
smp.http.header.security.policy		Content Security Policy (CSP) default-src 'self'; script-src 'self'; connect-src 'self'; img-src 'self'; style-src 'self' 'unsafe-inline'; frame-ancestors 'self'; form-action 'self';	true	STRING
smp.proxy.host		The http proxy host	false	STRING
smp.noproxy.hosts	localhost|127.0.0.1	list of nor proxy hosts. Ex.: localhost|127.0.0.1	false	STRING
smp.proxy.password		Base64 encrypted password for Proxy.	false	STRING
smp.proxy.port	80	The http proxy port	false	INTEGER

Parameter	Default Value	Comment	Restart needed	Value type
smp.proxy.user		The proxy user	false	STRING
identifiersBehaviour.ParticipantIdentifierScheme.validationRegex	^$\|^(?!^.{26})([a-z0-9]+-[a-z0-9]+-[a-z0-9]+)$|^urn:oasis:names:tc:ebcore:partyid-type:(iso6523|unregistered)(:.+)?$	Participant Identifier Schema of each PUT ServiceGroup request is validated against this schema.	false	REGEXP
identifiersBehaviour.ParticipantIdentifierScheme.validationRegexMessage	Participant scheme must start with:urn:oasis:names:tc:ebcore:partyid-type:(iso6523:|unregistered:) OR must be up to 25 characters long with form [domain]-[identifierArea]-[identifierType] (ex.: 'busdox-actorid-upis') and may only contain the following characters: [a-z0-9].	Error message for UI	false	STRING
identifiersBehaviour.scheme.mandatory	true	Scheme for participant identifier is mandatory	false	BOOLEAN
identifiersBehaviour.ParticipantIdentifierScheme.ebCoreId.concatenate	false	Concatenate ebCore party id in XML responses <ParticipantIdentifier>urn:oasis:names:tc:ebcore:partyid-type:unregistered:test-ebcore-id</ParticipantIdentifier>	false	BOOLEAN
identifiersBehaviour.caseSensitive.ParticipantIdentifierSchemes	sensitive-participant-sc1|sensitive-participant-sc2	Specifies schemes of participant identifiers that must be considered CASE-SENSITIVE.	false	LIST_STRING
identifiersBehaviour.caseSensitive.DocumentIdentifierSchemes	casesensitive-doc-scheme1|casesensitiv	Specifies schemes of document identifiers that must be considered CASE-SENSITIVE.	false	LIST_STRING

Parameter	Default Value	Comment	Restart needed	Value type
	e-doc-scheme2			
bdmsl.integration.enabled	false	BDMSL (SML) integration ON/OFF switch	false	BOOLEAN
bdmsl.participant.multidomain.enabled	false	Set to true if SML support participant on multidomain	true	BOOLEAN
bdmsl.integration.url	http://localhost:8080/edelivery-sml	BDMSL (SML) endpoint	false	URL
bdmsl.integration.tls.disableCNCheck	false	If SML Url is HTTPs - Disable CN check if needed.	false	BOOLEAN
bdmsl.integration.tls.serverSubjectRegex	.*	Regular expression for server TLS certificate subject verification CertEx. .*CN=acc.edelivery.tech.ec.europa.eu.*.	false	REGEXP
bdmsl.integration.logical.address	http://localhost:8080/smp/	Logical SMP endpoint which will be registered on SML when registering new domain	false	URL
bdmsl.integration.physical.address	0.0.0.0	Physical SMP endpoint which will be registered on SML when registering new domain.	false	STRING
smp.keystore.password		Encrypted keystore (and keys) password	false	STRING
smp.keystore.filename	smp-keystore.jks	Keystore filename	false	FILENAME
smp.truststore.password		Encrypted truststore password	false	STRING
smp.truststore.filename		Truststore filename	false	FILENAME
smp.certificate.crl.force	false	If false, then if CRL is not reachable ignore CRL validation	false	BOOLEAN
configuration.dir	smp	Path to the folder containing all the configuration files (keystore and encryption key)	true	PATH
encryption.key.filename	encryptionPrivateKey.private	Key filename to encrypt passwords	true	FILENAME
smp.keystore.password.decrypted		Only for backup purposes when password is automatically created. Store password somewhere save and delete this entry!	false	STRING
smp.truststore.password.decrypted		Only for backup purposes when password is automatically created. Store password somewhere save and delete this entry!	false	STRING

Parameter	Default Value	Comment	Restart needed	Value type
smp.certificate.validation.allowedCertificatePolicyOIDs		List of certificate policy OIDs separated by | where at least one must be in the CertifictePolicy extension	false	STRING
smp.certificate.validation.subjectRegex	.*	Regular expression to validate subject of the certificate	false	REGEXP
smp.property.refresh.cronJobExpression	0 48 */1 * * *	Property refresh cron expression (def 12 minutes to each hour). Property change is refreshed at restart!	true	CRON_EXPRESSION
smp.ui.session.secure	false	Cookie is only sent to the server when a request is made with the https: scheme (except on localhost), and therefore is more resistant to man-in-the-middle attacks.	false	BOOLEAN
smp.ui.session.max-age		Number of seconds until the cookie expires. A zero or negative number will expire the cookie immediately. Empty value will not set parameter	false	INTEGER
smp.ui.session.strict	Lax	Controls whether a cookie is sent with cross-origin requests, providing some protection against cross-site request forgery attacks. Possible values are: Strict, None, Lax. (Cookies with SameSite=None require a secure context/HTTPS)!!)	false	STRING
smp.ui.session.path		A path that must exist in the requested URL, or the browser won't send the Cookie header. Null/Empty value sets the authentication requests context by default. The forward slash (/) character is interpreted as a directory separator, and subdirectories will be matched as well: for Path=/docs, /docs, /docs/Web/, and /docs/Web/HTTP will all match	false	STRING
smp.ui.session.idle_timeout.admin	300	Specifies the time, in seconds, between client requests before the SMP will invalidate session for ADMIN users (System)!	false	INTEGER
smp.ui.session.idle_timeout.user	1800	Specifies the time, in seconds, between client requests before the SMP will invalidate session for users (Service group, SMP Admin)	false	INTEGER
smp.cluster.enabled	false	Define if application is set in cluster. In not cluster environment, properties are updated on set Property.	false	BOOLEAN
smp.passwordPolicy.validationRegex	^(?=.*[0-9])(?=.*[a-z])(?=.*[A-Z])(?=.*[~`!@#$%^&+=\-	Password minimum complexity rules!	false	REGEXP

Parameter	Default Value	Comment	Restart needed	Value type
	_<>.,?:;*/()|\[\]{}'"\\]).{16,32}$			
smp.passwordPolicy.validationMessage	Minimum length: 16 characters;Maximum length: 32 characters;At least one letter in lowercase;At least one letter in uppercase;At least one digit;At least one special character	The error message shown to the user in case the password does not follow the regex put in the domibus.passwordPolicy.pattern property	false	STRING
smp.passwordPolicy.validDays	90	Number of days password is valid	false	INTEGER
smp.passwordPolicy.warning.beforeExpiration	15	How many days before expiration should the UI warn users at login	false	INTEGER
smp.passwordPolicy.expired.forceChange	true	Force change password at UI login if expired	false	BOOLEAN
smp.user.login.fail.delay	1000	Delay response in ms on invalid username or password	false	INTEGER
smp.user.login.maximum.attempt	5	Number of console login attempt before the user is deactivated	false	INTEGER
smp.user.login.suspension.time	3600	Time in seconds for a suspended user to be reactivated. (If 0 the user will not be reactivated)	false	INTEGER
smp.accessToken.validDays	60	Number of days access token is valid is valid	false	INTEGER
smp.accessToken.login.maximum.attempt	10	Number of accessToken login attempt before the accessToken is deactivated	false	INTEGER
smp.accessToken.login.suspension.time	3600	Time in seconds for a suspended accessToken to be reactivated. (If 0 the user will not be reactivated)	false	INTEGER
smp.accessToken.login.fail.delay	1000	Delay in ms on invalid token id or token	false	INTEGER
smp.ui.authentication.types	PASSWORD	Set list of '|' separated authentication types: PASSWORD|SSO.	false	LIST_STRING
smp.automation.authentication.types	TOKEN|CERTIFICATE	Set list of '|' separated application-automation authentication types	false	LIST_STRING

Parameter	Default Value	Comment	Restart needed	Value type
		(Web-Service integration). Currently supported TOKEN, CERTIFICATE: ex. TOKEN|CERTIFICATE		
smp.automation.authentication.external.tls.clientCert.enabled	false	Authentication with external module as: reverse proxy. Authenticated data are sent to application using 'Client-Cert' HTTP header. Do not enable this feature without properly configured reverse-proxy!	false	BOOLEAN
smp.automation.authentication.external.tls.SSLClientCert.enabled	false	Authentication with external module as: reverse proxy. Authenticated certificate is sent to application using 'SSLClientCert' HTTP header. Do not enable this feature without properly configured reverse-proxy!	false	BOOLEAN
smp.sso.cas.ui.label	EU Login	The SSO service provider label.	true	STRING
smp.sso.cas.url	http://localhost:8080/cas/	The SSO CAS URL endpoint	true	URL
smp.sso.cas.urlPath.login	login	The CAS URL path for login. Complete URL is composed from parameters: ${smp.sso.cas.url}/${smp.sso.cas.urlpath.login}.	true	STRING
smp.sso.cas.callback.url	http://localhost:8080/smp/ui/public/rest/security/cas	The URL is the callback URL belonging to the local SMP Security System. If using RP, make sure it target SMP path '/ui/public/rest/security/cas'	true	URL
smp.sso.cas.smp.urlPath	/smp/ui/public/rest/security/cas	SMP relative path which triggers CAS authentication	true	STRING
smp.sso.cas.smp.user.data.urlPath	userdata/myAccount.cgi	Relative path for CAS user data. Complete URL is composed from parameters: ${smp.sso.cas.url}/${smp.sso.cas.smp.user.data.urlpath}.	true	STRING
smp.sso.cas.token.validation.urlPath	laxValidate	The CAS URL path for login. Complete URL is composed from parameters: ${smp.sso.cas.url}/${smp.sso.cas.token.validation.urlpath}.	true	STRING
smp.sso.cas.token.validation.params	acceptStrengths:BASIC,CLIENT_CERT|assuranceLevel:TOP	The CAS token validation key:value properties separated with '|'. Ex: 'acceptStrengths:BASIC,CLIENT_CERT|assuranceLevel:TOP'	true	MAP_STRING
smp.sso.cas.token.validation.groups	DIGIT_SMP|DIGIT_ADMIN	'|' separated CAS groups user must belong to.	true	LIST_STRING

Parameter	Default Value	Comment	Restart needed	Value type
mail.smtp.host		Email server - configuration for submitting the emails.	false	STRING
mail.smtp.port	25	Smtp mail port - configuration for submitting the emails.	false	INTEGER
mail.smtp.protocol	smtp	smtp mail protocol- configuration for submitting the emails.	false	STRING
mail.smtp.username		smtp mail protocol- username for submitting the emails.	false	STRING
mail.smtp.password		smtp mail protocol - encrypted password for submitting the emails.	false	STRING
mail.smtp.properties		key:value properties separated with '|'. Ex: mail.smtp.auth:true|mail.smtp.starttls.enable:true|mail.smtp.quitwait:false.	false	MAP_STRING
smp.alert.user.login_failure.enabled	false	Enable/disable the login failure alert of the authentication module.	false	BOOLEAN
smp.alert.user.login_failure.level	LOW	Alert level for login failure.	false	STRING
smp.alert.user.login_failure.mail.subject	Login failure	Login failure mail subject. Values: {LOW, MEDIUM, HIGH}	false	STRING
smp.alert.user.suspended.enabled	true	Enable/disable the login suspended alert of the authentication module.	false	BOOLEAN
smp.alert.user.suspended.level	HIGH	Alert level for login suspended. Values: {LOW, MEDIUM, HIGH}	false	STRING
smp.alert.user.suspended.mail.subject	Login credentials suspended	Login suspended mail subject.	false	STRING
smp.alert.user.suspended.mail.moment	WHEN_BLOCKED	#When should the account disabled alert be triggered. Values: AT_LOGON: An alert will be triggered each time a user tries to login to a disabled account. WHEN_BLOCKED: An alert will be triggered once when the account got suspended.	false	STRING
smp.alert.password.imminent_expiration.enabled	true	Enable/disable the imminent password expiration alert	false	BOOLEAN
smp.alert.password.imminent_expiration.delay_days	15	Number of days before expiration as for how long before expiration the system should send alerts.	false	INTEGER
smp.alert.password.imminent_expiration.frequency_days	5	Interval between alerts.	false	INTEGER

Parameter	Default Value	Comment	Restart needed	Value type
smp.alert.password.imminent_expiration.level	LOW	Password imminent expiration alert level. Values: {LOW, MEDIUM, HIGH}	false	STRING
smp.alert.password.imminent_expiration.mail.subject	Password imminent expiration	Password imminent expiration mail subject.	false	STRING
smp.alert.password.expired.enabled	true	Enable/disable the password expiration alert	false	BOOLEAN
smp.alert.password.expired.delay_days	30	Number of days after expiration as for how long the system should send alerts.	false	INTEGER
smp.alert.password.expired.frequency_days	5	Frequency in days between alerts.	false	INTEGER
smp.alert.password.expired.level	LOW	Password expiration alert level. Values: {LOW, MEDIUM, HIGH}	false	STRING
smp.alert.password.expired.mail.subject	Password expired	Password expiration mail subject.	false	STRING
smp.alert.accessToken.imminent_expiration.enabled	true	Enable/disable the imminent accessToken expiration alert	false	BOOLEAN
smp.alert.accessToken.imminent_expiration.delay_days	15	Number of days before expiration as for how long before expiration the system should send alerts.	false	INTEGER
smp.alert.accessToken.imminent_expiration.frequency_days	5	Frequency in days between alerts.	false	INTEGER
smp.alert.accessToken.imminent_expiration.level	LOW	AccessToken imminent expiration alert level. Values: {LOW, MEDIUM, HIGH}	false	STRING
smp.alert.accessToken.imminent_expiration.mail.subject	Access token imminent expiration	accessToken imminent expiration mail subject.	false	STRING
smp.alert.accessToken.expired.enabled	true	Enable/disable the accessToken expiration alert	false	BOOLEAN
smp.alert.accessToken.expired.delay_days	30	Number of days after expiration as for how long the system should send alerts.	false	INTEGER
smp.alert.accessToken.expired.frequency_days	5	Frequency in days between alerts.	false	INTEGER
smp.alert.accessToken.expired.level	LOW	Access Token expiration alert level. Values: {LOW, MEDIUM, HIGH}	false	STRING

Parameter	Default Value	Comment	Restart needed	Value type
smp.alert.accessToken.expired.mail.subject	Access token expired	Password expiration mail subject.	false	STRING
smp.alert.certificate.imminent_expiration.enabled	true	Enable/disable the imminent certificate expiration alert	false	BOOLEAN
smp.alert.certificate.imminent_expiration.delay_days	15	Number of days before expiration as for how long before expiration the system should send alerts.	false	INTEGER
smp.alert.certificate.imminent_expiration.frequency_days	5	Frequency in days between alerts.	false	INTEGER
smp.alert.certificate.imminent_expiration.level	LOW	certificate imminent expiration alert level. Values: {LOW, MEDIUM, HIGH}	false	STRING
smp.alert.certificate.imminent_expiration.mail.subject	Certificate imminent expiration	Certificate imminent expiration mail subject.	false	STRING
smp.alert.certificate.expired.enabled	true	Enable/disable the certificate expiration alert	false	BOOLEAN
smp.alert.certificate.expired.delay_days	30	Number of days after expiration as for how long the system should send alerts.	false	INTEGER
smp.alert.certificate.expired.frequency_days	5	Frequency in days between alerts.	false	INTEGER
smp.alert.certificate.expired.level	LOW	Certificate expiration alert level. Values: {LOW, MEDIUM, HIGH}	false	STRING
smp.alert.certificate.expired.mail.subject	Certificate expired	Password expiration mail subject.	false	STRING
smp.alert.credentials.cronJobExpression	0 52 4 */1 * *	Property cron expression for triggering alert messages!	false	CRON_EXPRESSION
smp.alert.credentials.serverInstance	localhost	If smp.cluster.enabled is set to true then then instance (hostname) to generate report.	false	STRING
smp.alert.credentials.batch.size	200	Max alertes generated in a batch for the type	false	INTEGER
smp.alert.mail.from	test@alert-send-mail.eu	Alert send mail	false	EMAIL

authentication.blueCoat.enabled	false	Property was replaced by property: smp.automation.authentication.external.tls.clientCert.enabled	false	BOOLEAN

16.3. smp_domain table configuration

This table is used to support the multi-tenancy feature of the SMP. Its parameters/fields are:

- **SML_SMP_ID**: This is the SMP ID that must match the SMP ID registered within the SML.
- **SML_CLIENT_CERT_HEADER:** The SMP's certificate - needed only when accessing BDMSL directly through HTTP. The configured "Client-Cert" HTTP header will be added to each BDMSL request (bypassing SSL certificate verification made normally by SSL terminator) .
- **SML_CLIENT_KEY_ALIAS**: This is the Domain scoped alias of the keystore private key used for authentication with the SML. The password is the same as xmldsig.keystore.password defined in the SMP configuration file.
- **SIGNATURE_KEY_ALIAS:** This field points to the **Domain scoped** alias of the Keystore private key certificate, used by the SMP to sign GET Signed Service Metadata responses.
- **SML_SUBDOMAIN:** This is the informative identifier of SML domain code (eHealth, Peppol, etc). Since SML subdomain is part of DNS domain it must be a valid DNS domain part.
- **DOMAIN_CODE:** The unique domain code that is used as HTTP domain parameter when adding participants true REST service API to particular domain. Domain code can be alphanumeric and up to 63 characters long.

Example: Update the default single domain smp_domain table record:

update smp_domain **set SML_SMP_ID**='SMP-MCB-ID14', **SML_CLIENT_KEY_ALIAS**= 'smp_mock';

or

update smp_domain **set SML_SMP_ID**='SMP-MCB-ID14', **SML_CLIENT_CERT_HEADER**= 'serial=000000000000000000009A195D2DD88C&subject=CN=SMP_1000000000,O=DG-DIGIT,C=BE&validFrom=Oct 21 02:00:00 2014 CEST&validTo=Oct 21 01:59:59 2016 CEST&issuer=CN=Issuer Common Name,OU=Issuer Organization Unit,O=Issuer Organization,C=BE' where domainId='default';

17. SMP ADMIN CONSOLE

The SMP Admin console has two purposes:

- Enable anonymous users to search and explore published data in the SMP. Anonymous users can search for participants by participant ID, schema or domain.

- Enable Service Group administrators to manage owned Service groups, SMP administrators to manage Service groups registered on SMP, and System Administrators to manage users and domains.

The administration dashboard is reachable via the following URLs:

http://[host]:[port]/smp[-version]/iu/

If the deployment package (war file) filename changed in order to simply upgrade the old SMP version as for example “smp-4.0.0.war” to “cipa-smp-full-webapp.war”, than application root context might change as well.

Example:

http:// [host]:[port]/cipa-smp-full-webapp/ui/ .

Three types of roles are defined in the SMP admin console:

- **System Administrator**: this is a “super admin” who can manage SMP users and domains

- **SMP Administrator**: the SMP administrator can create/delete Service Groups, manage users and domains for service groups and its extensions and metadata. The SMP administrator has access to all Services Groups registered in the SMP.

- **Service Group Administrator**: this user can administer only the metadata for the Service Groups that he owns. He cannot change ownerships or the domains for the existing Service Group.

When users are logged, their role is displayed in read-only mode (as a label). Only the System Administrator can change the role of another user including an SMP Administrator and a Service group Administrator. They, however, cannot change the role of another System Administrator.

18. Contact Information

eDelivery Support Team

By email: **EC-EDELIVERY-SUPPORT@ec.europa.eu**

Support Service: 8am to 6pm (Normal EC working Days)