EUROPEAN COMMISSION

DIGIT
Connecting Europe Facility

Service Metadata Publisher

Administration Guide

SMP 3.X

Version [1.2]

Status [Final]

Date: 29/06/2018

Document Approver(s):

Approver Name	Role
Adrien FERIAL	

Document Reviewers:

Reviewer Name	Role
Pawel GUTOWSKI	Developer
Akin Olowosile	CEFS

Summary of Changes:

Version	Date	Created by	Short Description of Changes
V0.1	22/02/2016	DHENEIN Christophe	Initial Document Creation
V1.0	15/06/2017	Chaouki BERRAH (CEFS), Christophe DHENEIN (CEFS)	Document restructuring and updating.
V1.1	16/06/2016	Chaouki BERRAH (CEFS), Christophe DHENEIN (CEFS)	3.0.0-RC2 changed to 3.X Screens updated.
V1.2	29/06/2018	CEF Support	Updates + nexus references updated

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 3.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) and user defined words (chosen by the user) as well as comments and preferred values for certain variables. The conventions used in this document, to distinguish between them, are the followings:

- **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..). 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 environnement (JRE), version 7 or 8: 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 Server:
  - WebLogic 12c
  - Tomcat 8

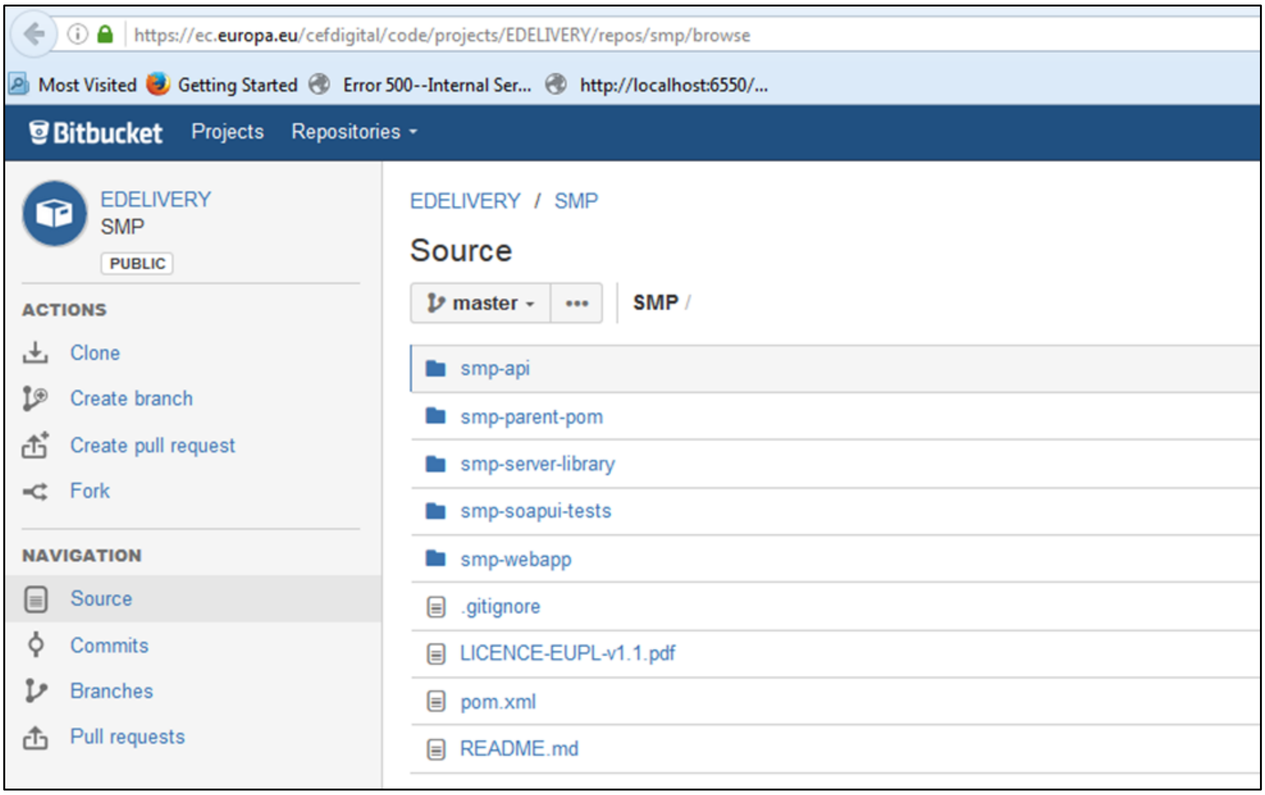
3.1. Binaries repository

All the CEF SMP artefacts can be directly downloaded from the CEF Digital portal.

3.2. Source Code Repository

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

https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse



https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse

EDELIVERY / SMP

Source

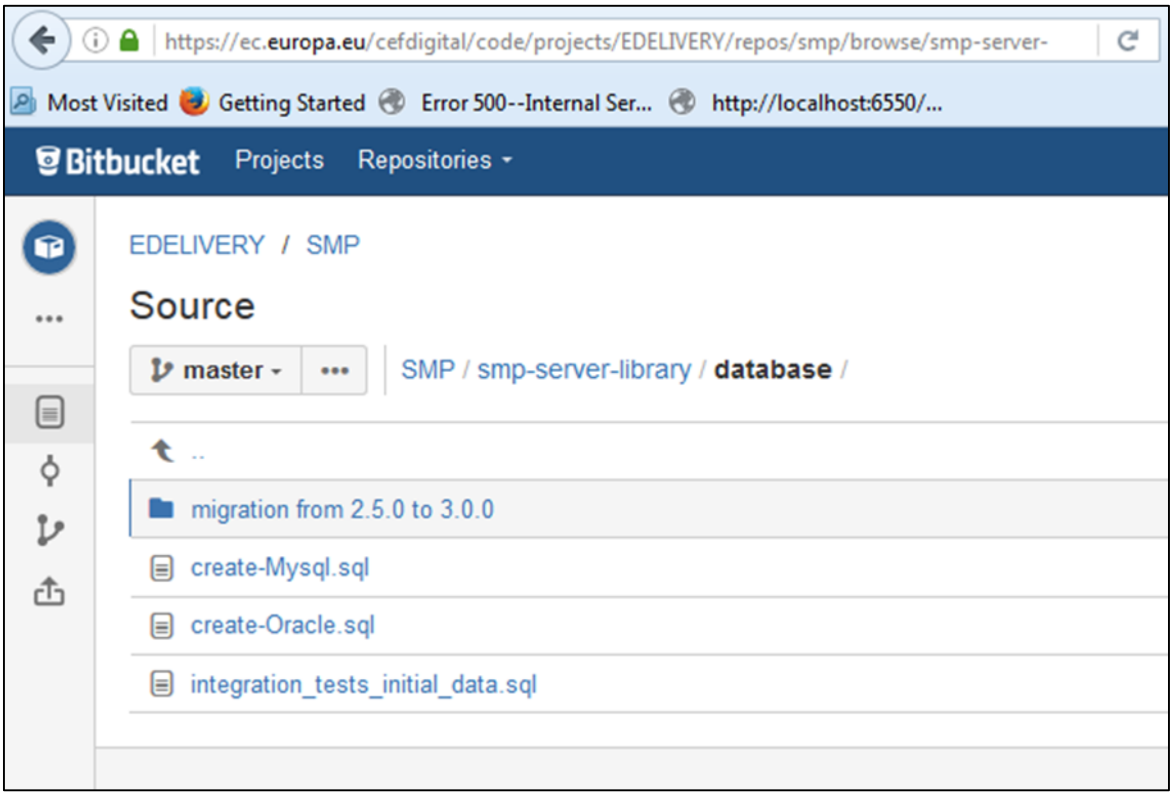
master / SMP /

- smp-api
- smp-parent-pom
- smp-server-library
- smp-soapui-tests
- smp-webapp
- .gitignore
- LICENCE-EUPL-v1.1.pdf
- pom.xml
- README.md

3.3. Database Scripts

The scripts to create (or migrate) the Oracle or MySQL databases can be found at:

https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse/smp-server-library/database



https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse/smp-server-

EDELIVERY / SMP

Source

master / SMP / smp-server-library / database /

- ..
- migration from 2.5.0 to 3.0.0
- create-Mysql.sql
- create-Oracle.sql
- integration_tests_initial_data.sql

4. Deployment

4.1. Deployment overview

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

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

The deployment of the CEF SMP can be summarized in the following mandatory steps:

- Database Configuration
- Application Server Preparation
- 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 and will be used in the remainder of this document.*
>
> *For Tomcat, Its refers to 'CATALINA_HOME'*
>
> *For Oracle WebLogic, it refers to 'DOMAIN_HOME'*

4.2. Database Configuration

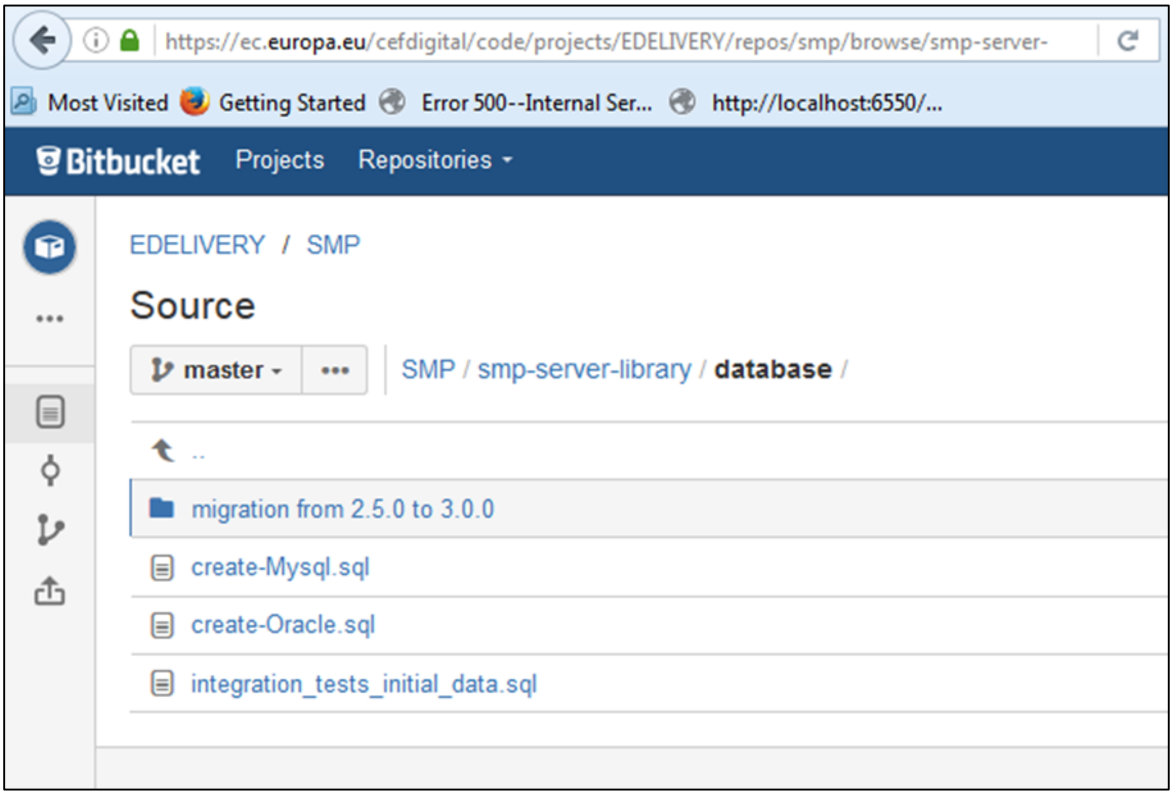
This section describes the steps necessary to create the database, tables and the SMP database user (**dbuser** used for database connection purpose).

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 one or more of the following resources, in the CEF SMP GIT source code repository (see section §3.2 for the download location):

https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse/smp-server-library/database


https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse/smp-server-
Most Visited
Getting Started
Error 500--Internal Ser...
http://localhost:6550/...
Bitbucket
Projects
Repositories
EDELIVERY / SMP
Source
master
SMP / smp-server-library / database /
..
migration from 2.5.0 to 3.0.0
create-Mysql.sql
create-Oracle.sql
integration_tests_initial_data.sql

4.2.1. ***MySQL configuration***

1. Download and copy the create-Mysql.sql 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 < create-Mysql.sql
```

4.2.2. ***Oracle Database configuration***

1. Download and copy the **create-Oracle.sql 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)

@create-Oracle.sql (run the scripts with the @ sign from the location of the
scripts)

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

4.2.3. Application Server Preparation

This section describes the changes that are applied to the host application server and consist of:

- For Tomcat & WebLogic: Creating and adding a directory in the java CLASSPATH where the CEF SMP configuration files are located
- For Weblogic only: Disabling the application Server default Authentication

4.2.4. Oracle WebLogic

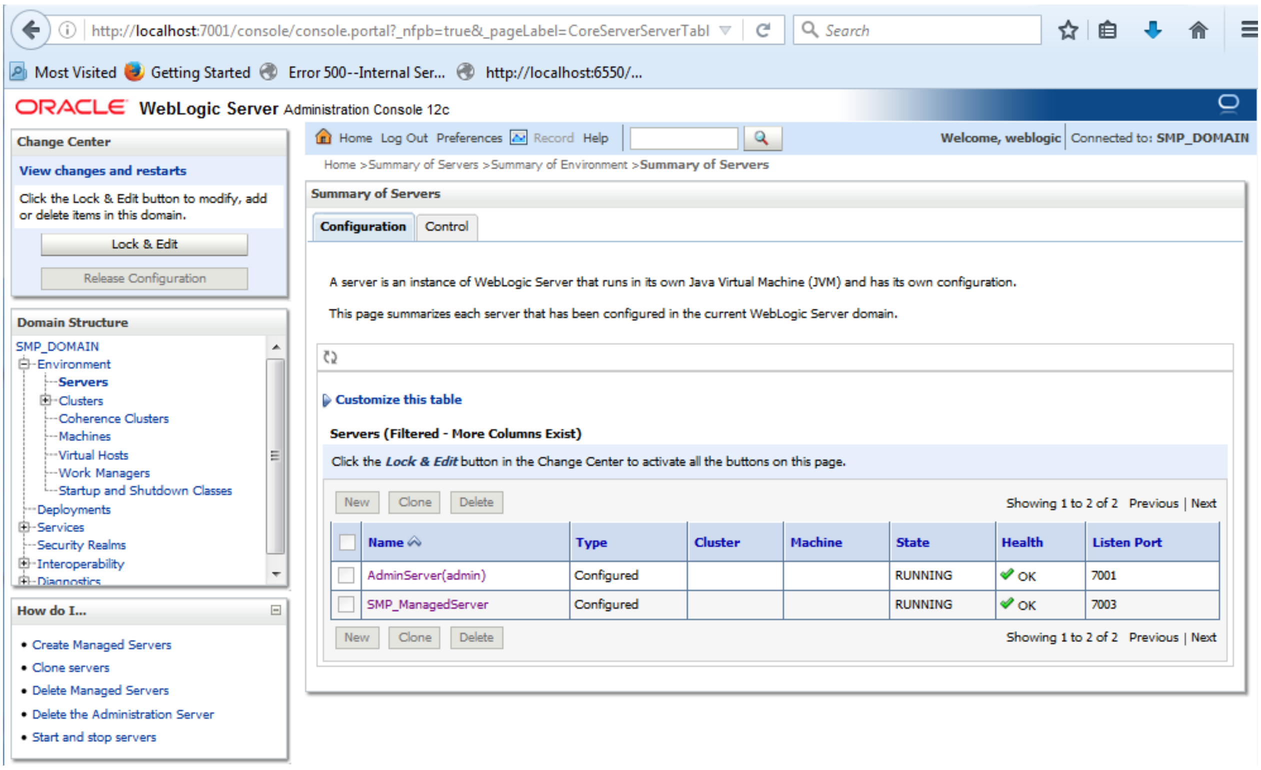
This section does not include the installation of a WebLogic application server. It is assumed that the WebLogic Server is installed and a WebLogic domain is created with an administration server and a managed server on which the CEF SMP will be deployed.

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:


http://localhost:7001/console/console.portal?_nfpb=true&_pageLabel=CoreServerServerTabl
Search
Most Visited
Getting Started
Error 500--Internal Ser...
http://localhost:6550/...
ORACLE WebLogic Server Administration Console 12c
Home Log Out Preferences Record Help
Welcome, weblogic Connected to: SMP_DOMAIN
Home >Summary of Servers >Summary of Environment >Summary of Servers
Change Center
View changes and restarts
Lock & Edit
Release Configuration
Domain Structure
SMP_DOMAIN
Environment
Servers
Clusters
Coherence Clusters
Machines
Virtual Hosts
Work Managers
Startup and Shutdown Classes
Deployments
Services
Security Realms
Interoperability
Diagnostics
How do I...
Create Managed Servers
Clone servers
Delete Managed Servers
Delete the Administration Server
Start and stop servers
Summary of Servers
Configuration
Control
Customize this table
New
Clone
Delete
Showing 1 to 2 of 2 Previous | Next

Click the Lock & Edit button to modify, add or delete items in this domain.

A server is an instance of WebLogic Server that runs in its own Java Virtual Machine (JVM) and has its own configuration.

This page summarizes each server that has been configured in the current WebLogic Server domain.

Servers (Filtered - More Columns Exist)

Click the *Lock & Edit* button in the Change Center to activate all the buttons on this page.

□	Name	Type	Cluster	Machine	State	Health	Listen Port
□	AdminServer(admin)	Configured			RUNNING	✓ OK	7001
□	SMP_ManagedServer	Configured			RUNNING	✓ OK	7003

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 and Configuring the Extra CLASSPATH for WebLogic, as described in the following 2 sections:

4.2.4.1. Disabling the Authentication on the WebLogic

The CEF SMP has its own authentication mechanism which makes the WebLogic authentication redundant. Therefore the WebLogic Authentication has to be disabled to stop it interfering with the SMP authentication:

Edit the config.xml file by adding the following tag before the closing the `</security-configuration>` tag:

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

As shown below:

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

4.2.4.2. Configuring the Extra CLASSPATH for WebLogic

In this Oracle WebLogic example, a directory called '**smp**' will be created in the root path of the WebLogic installation (DOMAIN_HOME) and the CLASSPATH modified to include this new directory.

Create an smp directory in the DOMAIN_HOME directory

Within the cef_edelivery_path/smp directory, create the following sub-directories:

- conf
- keystores
- temp
- logs

Edit the WebLogic DOMAIN_HOME/bin/setDomainEnv.sh

Add the 'EXPORT CLASSPATH=${CLASSPATH}:${DOMAIN_HOME}/smp/conf' 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/conf
        export CLASSPATH
```

```
/..
```

TO windows example

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

4.2.5. ***Configuring Tomcat***

In order to deploy the SMP on Tomcat, the following step needs to be completed:

4.2.5.1. 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

Within the cef_edelivery_path/smp directory, create two new sub-directories:

- conf
- keystores

For Linux:

Edit the $CATALINA_HOME/bin/setenv.sh file

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

For Windows

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

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

4.2.5.2. JDBC Driver

The JDBC driver need 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

4.3. SMP Initial Configuration

4.3.1. *SMP ID Configuration*

Configure the SMP ID as explained in section §5.25.4 of this document

4.3.2. *Database configuration*

Configure the database connectivity properties as explained in section §5.35.4 of this document

4.3.3. *Keystore configuration*

Configure the Keystore as explained in section §5.4 of this document

4.4. SMP .WAR file Deployment

The CEF SMP is deployed following the steps listed below:

4.4.1. *Tomcat*

Copy the cef_edelivery/smp/temp/cipa-smp-full-webapp-3.X.war file to the Tomcat '**webapps**' directory (cef_edelivery/webapps)

4.4.2. *Oracle WebLogic*

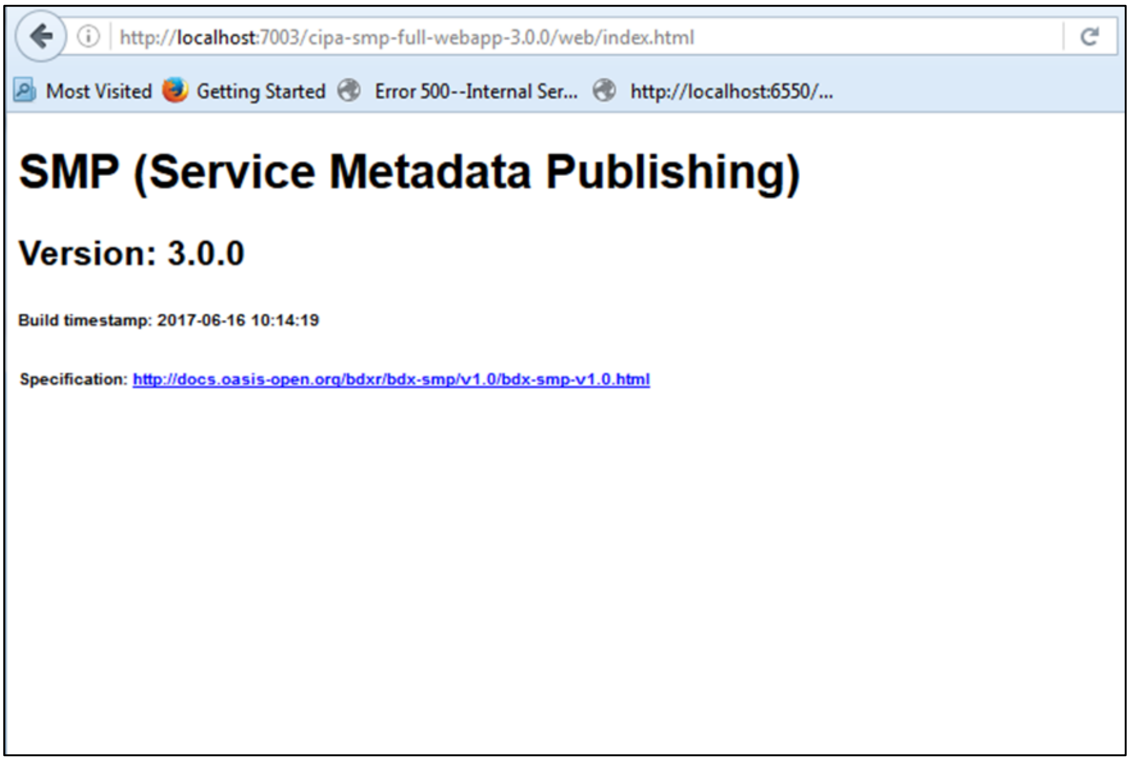
Deploy the 'war' file within WebLogic using the Oracle Weblogic deployer or using the Weblogic Administration Console.

An example of using the Oracle the '**weblogicc.deployer**' is provided below:

```
java weblogic.Deployer -adminurl
t3://${WebLogicAdminServerListenAddress}:${WebLogicAdminServerPort} \
-username ${WebLogicAdminUserName} \
-password ${WebLogicAdminUserPassword} \
-deploy -name cipa-smp-full-webapp-3.X.war \
-targets ${SMP_ManagedServer} \
-source $TEMP_DIR/cipa-smp-full-webapp-3.X.war
```

Verify the installation by navigating, using your browser, to **http://*localhost*:*7003*/ cipa-smp-full-webapp-3.X**

The following page is only displayed if the deployment is successful.



http://localhost:7003/cipa-smp-full-webapp-3.0.0/web/index.html
Most Visited
Getting Started
Error 500--Internal Ser...
http://localhost:6550/...

SMP (Service Metadata Publishing)

Version: 3.0.0

Build timestamp: 2017-06-16 10:14:19

Specification: http://docs.oasis-open.org/bdxr/bdx-smp/v1.0/bdx-smp-v1.0.html

5. SMP Configuration

For this step, you will have to use the following resources (see section *3.1 Binaries repository* for the download location):

- The SMP Web Application Archive (**smp-full-webapp-3.X.war**)
- An smp config file named ‘smp.config.properties located in the CLASSPATH
- A keystore used for XMLDSIG

By default, the file named ‘config.properties’ is embedded within the SMP .WAR file under the ./WEB-INF/classes directory. In order to offer flexibility and to avoid losing the SMP configuration while deploying a new SMP war file, it is recommended to extract this file from the .WAR archive and store it in a dedicated location located within the CLASSPATH of the application server as explained in the next section.

5.1.1. *Downloading and Extracting the war file*

1. Download the SMP Web Application Archive **smp-full-webapp-3.X.war** from the location specified in section §3.1 and save it in the cef_edelivery/smp/temp directory.
2. Extract the ‘config.properties’ file from the ‘smp-full-webapp-3.X.war’ file and place it into the directory ‘cef_edelivery/smp/conf’ directory created during the Server preparation 4.2.5.1
3. Rename **config.properties** as **smp.config.properties**

5.2. SMP ID Configuration

This is the name of the SMP.

This name can be choosen freely unless the SMP will be registered to an SML.

It is recommended but not mandatory to use only UPPERCASE characters

The configuration of the SMP ID is achieved by the setting the ‘**regServiceRegistrationHook.id**’ in the ‘config.property’ file as follow:

```
../
# SMP ID
regServiceRegistrationHook.id=MY_SMP_ID
/..
```

5.3. Database Configuration

As explained earlier, the CEF SMP uses direct connection to the database.

The CEF SMP databbase back-end configuration is performed within the CEF SMP configuration file (smp.config.properties’ file).

Depending on the selected database back-end, modify the ‘smp.config.properties’ files as indicated below

For Oracle Database:

```
../
## JDBC configuration for DB Oracle
jdbc.driver = oracle.jdbc.OracleDriver
jdbc.url = jdbc:oracle:thin:@dbhost:dbport:smp_database
jdbc.user = smp_user
jdbc.password = smp_user_pwd
target-database = Oracle
jdbc.read-connections.max = 10
/..
```

For MySQL:

```
../
## JDBC configuration for DB MySQL
# JDBC configuration for DB MySQL
jdbc.driver = com.mysql.jdbc.Driver
jdbc.url = jdbc:mysql://dbhost:dbport/smp_database
jdbc.user = smp_user
jdbc.password = smp_user_pwd
target-database = MySQL
jdbc.read-connections.max = 10
/..
```

5.4. SMP Keystores

CEF SMP uses two distinct keystores used for different usage:

- One optional keystore is used to identify the SML in combination with HTTPS
- One **MANDATORY** keystore is used for signing the responses to 'GET' requests (XMLDSIG response signing)

5.4.1. *XMLDSIG response signing Keystore*

This keystore is mandatory and will prevent the CEF SMP from starting if not deployed and/or properly configured.

A sample keystore can be downloaded from the following link:

https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp-mock-services/browse/signature_keys.jks?at=7831890f43098c67f52c4006388f9135325aa318&raw

Details of the sample keystore:

Keytore name	: signature_keys.jks
Keystore Password	: mock
Key Alias	: smp_mock
Key Passsword	: mock

This keystore can be copied to ./smp/keystores/ or any other chosen directory of the SMP server, then configured in the **smp.config.properties** file as shown in the following example:

```
../
## XMLDSIG response signing:
xmldsig.keystore.classpath    = ./smp/keystores/signature_keys.jks
xmldsig.keystore.password     = mock
xmldsig.keystore.key.alias    = smp_mock
xmldsig.keystore.key.password = mock
/..
```

5.4.2. ***SML Keystore***

The SML keystore is optional and need only to be deployed and configured if the SMP will be accessing an SML using HTTPS.

The SML keystore should be deployed in a server location and configured in the **smp.config.properties** using a relative or an absolute location

```
../
# SML URL (incl. the service name)
#regServiceRegistrationHook.regLocatorUrl=https://sml.peppolcentral.org/manageparticipantidentifier
#
regServiceRegistrationHook.regLocatorUrl=https://smk.peppolcentral.org/manageparticipantidentifier
regServiceRegistrationHook.regLocatorUrl=http://localhost:8080/manageparticipantidentifier
regServiceRegistrationHook.keystore.classpath =
regServiceRegistrationHook.keystore.password  =
/..
```

5.5. Configuring the CEF SMP for use with an SML

The SMP can be registered to 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

5.5.1. ***Defining the SMP ID***

The name of the SMP instance can be freely chosen but it is not currently possible to know whether the name chosen is free for registration or not.

It is also RECOMMENDED but not mandatory to use only uppercase characters.

The configuration of the SMP ID is achieved by the configuring the ‘**regServiceRegistrationHook.id’** in the ‘**config.property’** file as follow:

```
../
# SMP ID
regServiceRegistrationHook.id=MY_SMP_ID
/..
```

5.5.2. ***Configuring the Registration Hook***

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

This is achieved by configuring the Registration callback (SML client caller) in the ‘**config.property**’ file.

The “registrationHook.class” property has to be changed

From : **eu.europa.ec.cipa.smp.server.hook.DoNothingRegistrationHook**

To : **eu.europa.ec.cipa.smp.server.hook.RegistrationServiceRegistrationHook**

```
../
## Registration callback (SML client caller)
registrationHook.class=eu.europa.ec.cipa.smp.server.hook.RegistrationServiceRegi
strationHook
#registrationHook.class=eu.europa.ec.cipa.smp.server.hook.DoNothingRegistrationH
ook
/..
```

5.5.3. ***Configuring the SML URL***

The configuration of the SML URL end point is achieved by configuring the ‘**regServiceRegistrationHook.regLocatorUrl**’ property in the ‘**config.property’** file as follows:

```
../
# SML URL (incl. the service name)
```

```
#regServiceRegistrationHook.regLocatorUrl=https://sml.peppolcentral.org/manageparticipantidentifier

#regServiceRegistrationHook.regLocatorUrl=https://smk.peppolcentral.org/manageparticipantidentifier

regServiceRegistrationHook.regLocatorUrl=http://localhost:8080/manageparticipantidentifier

/..
```

5.5.4. SMP authentication to an SML

Once registered in an SML, the SMP needs to authenticate against the SML during normal operation

This can be achieved by using plain text HTTP or HTTPS/TLS

5.5.4.1. Plain Text HTTP

When using plain text HTTP, as mentioned in the '**regServiceRegistrationHook.regLocatorUrl'** covered earlier, the certificate's metadata will be added to each SML request configured via the configured **Client-Cert** HTTP header.

```
 ../

# 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 cert verification made normally by SSL terminator)

regServiceRegistrationHook.clientCert=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

/..
```

5.5.4.2. HTTPS/TLS

When using HTTPS/TLS, as mentioned in the '**regServiceRegistrationHook.regLocatorUrl**' covered earlier, all information related to the keystore containing the SMP certificate must be configured.

Note that in this case, the '**regServiceRegistrationHook.clientCert**', described earlier must be set to **Blank**.

```
../
regServiceRegistrationHook.clientCert=
regServiceRegistrationHook.keystore.classpath = ../keystores/yourKeystore.jks
regServiceRegistrationHook.keystore.password  = youKeystorePassword
/..
```

6. SMP User Management

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

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

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

6.1. User Roles

The CEF 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 store 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 Metadata	Publish detailed information about one specific document service (multiple processes and endpoints). This same service is used to create and update ServiceMetaData.	PUT	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 transport. This service retrieves the SignedServiceMetadata according to the input quadruplet participantIdentifier+participantIndentifierScheme+ documentIdentifier+documentIdentifierScheme. Returns information from the Endpoint table.	GET	SignedServiceMetadata

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/cefdigital/wiki/x/0wvNAg

6.2. BCRYPT password generation

The SMP v3.X uses the BCRYPT algorithm to hash users' passwords. A BCRYPT-hashing tool is bundled into the SMP WAR file. Toget the hashing code, follow the steps below:

Place a copy of the ‘**cipa-smp-full-webapp-3.X.war**’ file into a temporary directory of your choice.

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

```
$ jar -xvf cipa-smp-full-webapp-3.X.war
```

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

```
$ java -cp "WEB-INF/lib/*"
eu.europa.ec.cipa.smp.server.security.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.cipa.smp.server.security.BCryptPasswordHash
password_to_be_hashed

$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.cipa.smp.server.security.BCryptPasswordHash
password_to_be_hashed_1 password_to_be_hashed_2
$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa
$2a$107zNzSeZpxiHeqY2BRKkHE.HknfIe3aiu6XzU.qHHnnPbUHKtfcmDG
```

6.3. SMP Database User Creation

Adding an SMP user is performed by adding a new entry in the SMP database (‘**SMP_USER’** table).

User role is determined by setting the '**isadmin**' field in the SMP_USER table as follow:

User Role	**isadmin value**
Admin SMP	1
Admin Service Group	0
AnonymousUser (Not defined in the SMP User database)	N/A

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

6.3.1. *'Admin SMP' User creation*

Username	: smp_admin
Password (Hashed)	: $2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa
IsAdmin	: 1

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

```
SQL> INSERT into SMP_USER (USERNAME, PASSWORD, ISADMIN) values ('smp_user', '$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa', 1);
```

6.3.2. *'Admin ServiceGroup' User Creation*

Username	: smp_user1
Password (Hashed)	: $2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa
IsAdmin	: 0

Execute the following database command.

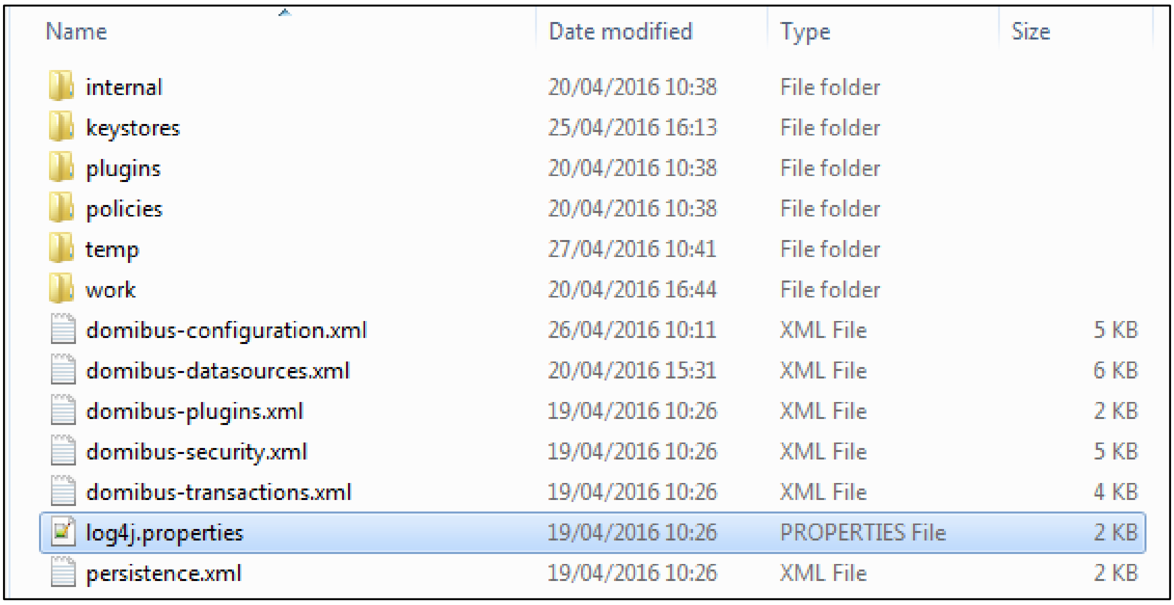
```
SQL> INSERT into SMP_USER (USERNAME, PASSWORD, ISADMIN) values ('smp_user1', '$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa', 0);
```

7. Logging Configuration

7.1. Logging properties

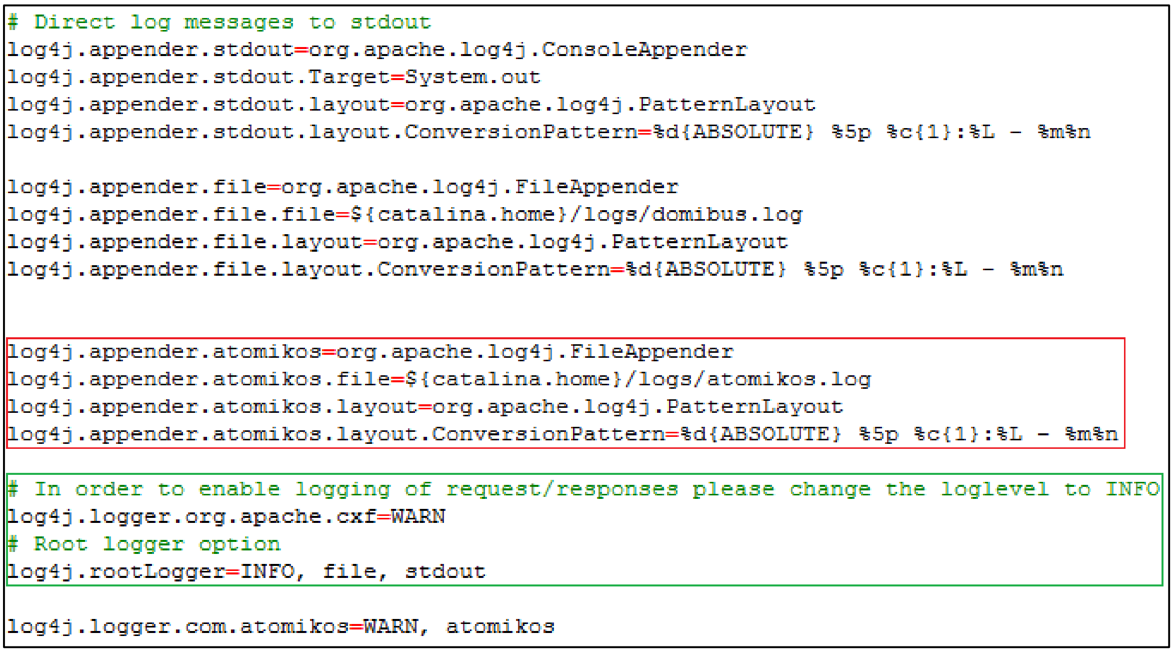
The SMP logging property are defined in the '*./WEBINF*/**log4j.properties' file** embedded in the SMP '.war' file.

It is possible to modify the configuration of the logs by editing the embedded 'log4j properties'.



Name	Date modified	Type	Size
internal	20/04/2016 10:38	File folder	
keystores	25/04/2016 16:13	File folder	
plugins	20/04/2016 10:38	File folder	
policies	20/04/2016 10:38	File folder	
temp	27/04/2016 10:41	File folder	
work	20/04/2016 16:44	File folder	
domibus-configuration.xml	26/04/2016 10:11	XML File	5 KB
domibus-datasources.xml	20/04/2016 15:31	XML File	6 KB
domibus-plugins.xml	19/04/2016 10:26	XML File	2 KB
domibus-security.xml	19/04/2016 10:26	XML File	5 KB
domibus-transactions.xml	19/04/2016 10:26	XML File	4 KB
log4j.properties	19/04/2016 10:26	PROPERTIES File	2 KB
persistence.xml	19/04/2016 10:26	XML File	2 KB

In the example below, A **log4j.properties** file is shown:



```
# Direct log messages to stdout
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.Target=System.out
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L - %m%n

log4j.appender.file=org.apache.log4j.FileAppender
log4j.appender.file.file=${catalina.home}/logs/domibus.log
log4j.appender.file.layout=org.apache.log4j.PatternLayout
log4j.appender.file.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L - %m%n


log4j.appender.atomikos=org.apache.log4j.FileAppender
log4j.appender.atomikos.file=${catalina.home}/logs/atomikos.log
log4j.appender.atomikos.layout=org.apache.log4j.PatternLayout
log4j.appender.atomikos.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L - %m%n

# In order to enable logging of request/responses please change the loglevel to INFO
log4j.logger.org.apache.cxf=WARN
# Root logger option
log4j.rootLogger=INFO, file, stdout

log4j.logger.com.atomikos=WARN, atomikos
```

In red: the parameters can be edited to modify the location of the log file, and the layout.

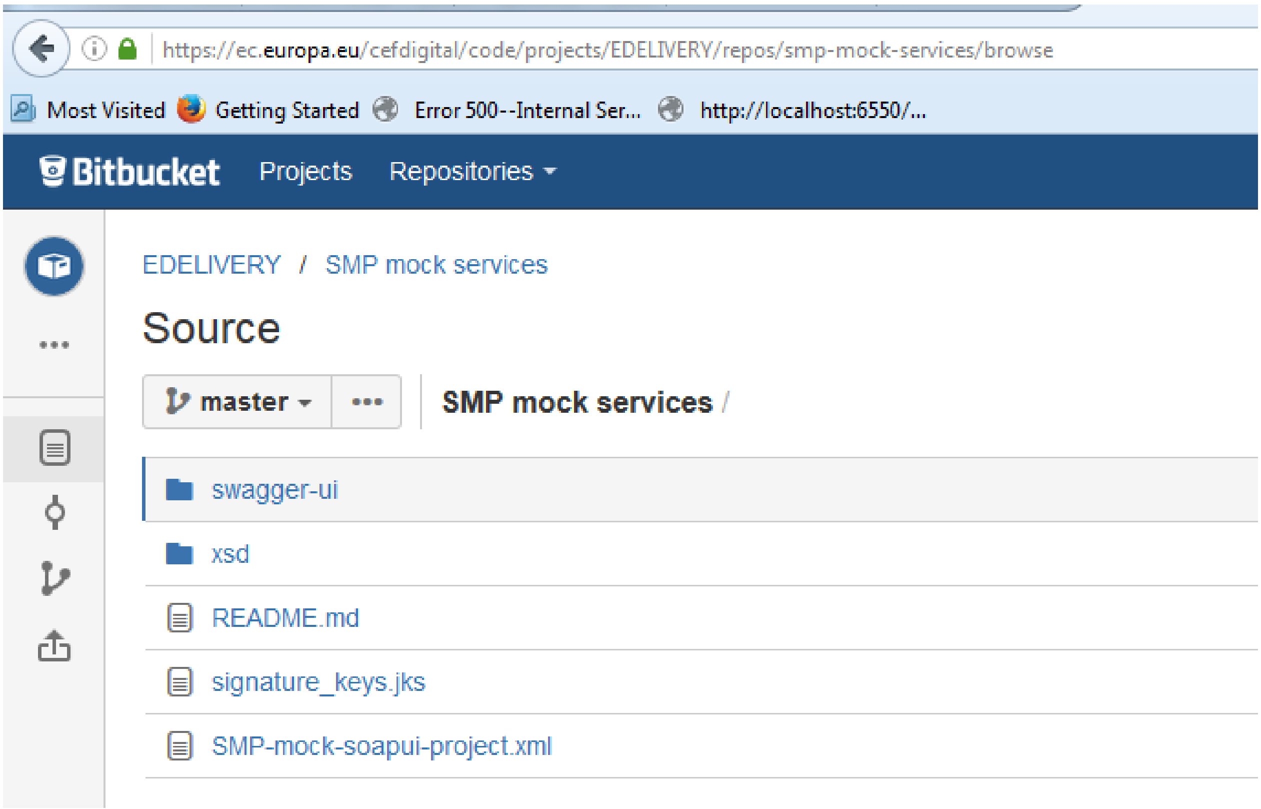
In green: the parameters can be edited to change the level of logging (3 levels definied: INFO, WARN, and ERROR).

8. SOAPUI Testing

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

An SMP MOCK SoapUI project can be freely downloaded to perform these tasks:

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


https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp-mock-services/browse
Most Visited
Getting Started
Error 500--Internal Ser...
http://localhost:6550/...
Bitbucket
Projects
Repositories
EDELIVERY / SMP mock services
Source
master
SMP mock services /
swagger-ui
xsd
README.md
signature_keys.jks
SMP-mock-soapui-project.xml

Download the SMP-mock-soapui-project.xml SOAP UI project and open it using SoapUI.

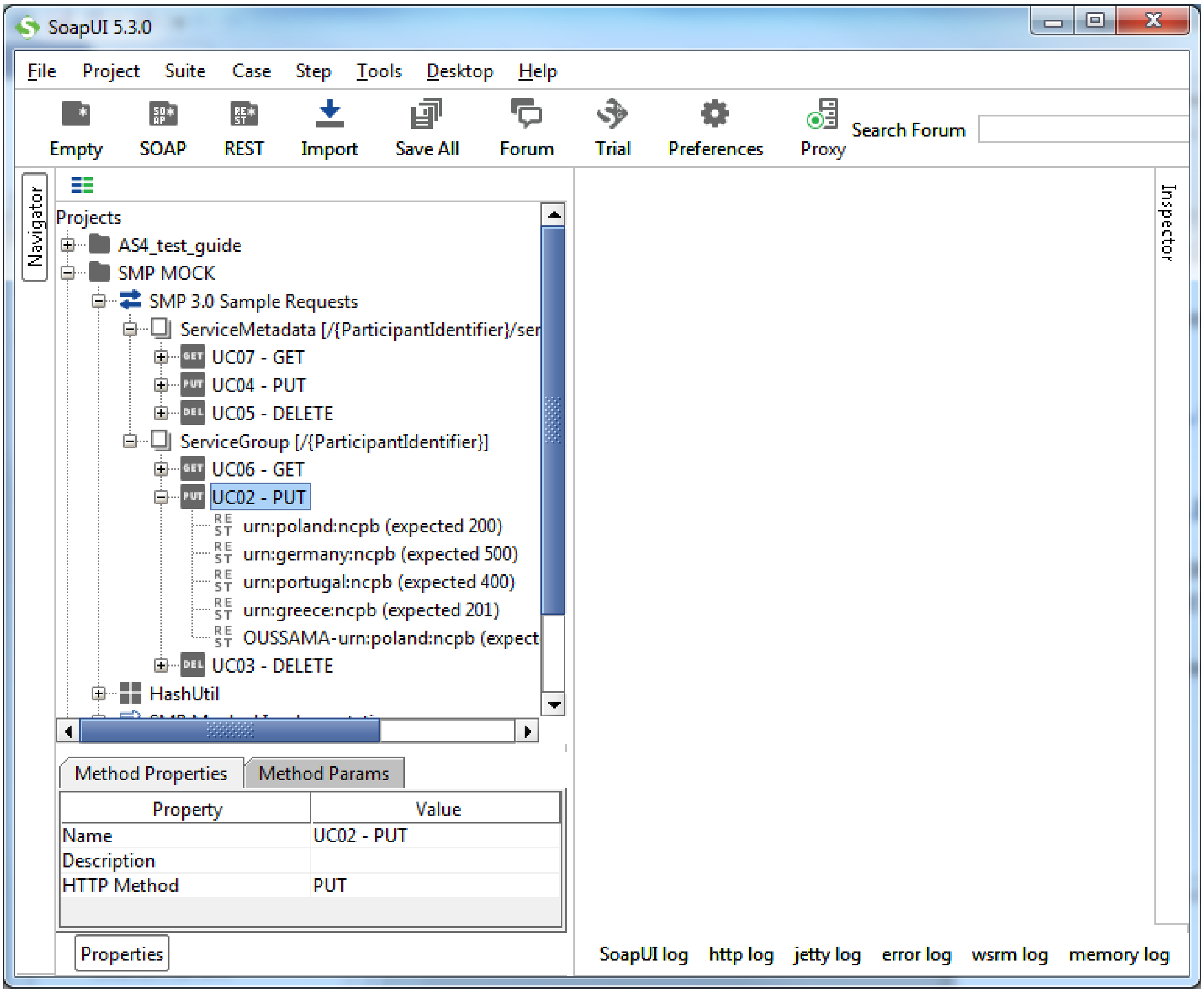
The rest of the procedure will be described in the next steps.

8.1. Creation, update and deletion of Service Groups.

8.1.1. *Create a Service Group*

IMPORTANT NOTE: As per design of the SMP, Creation, update and deletion of Service Group can ONLY BE ACHIEVED USING BASIC AUTHENTICATION

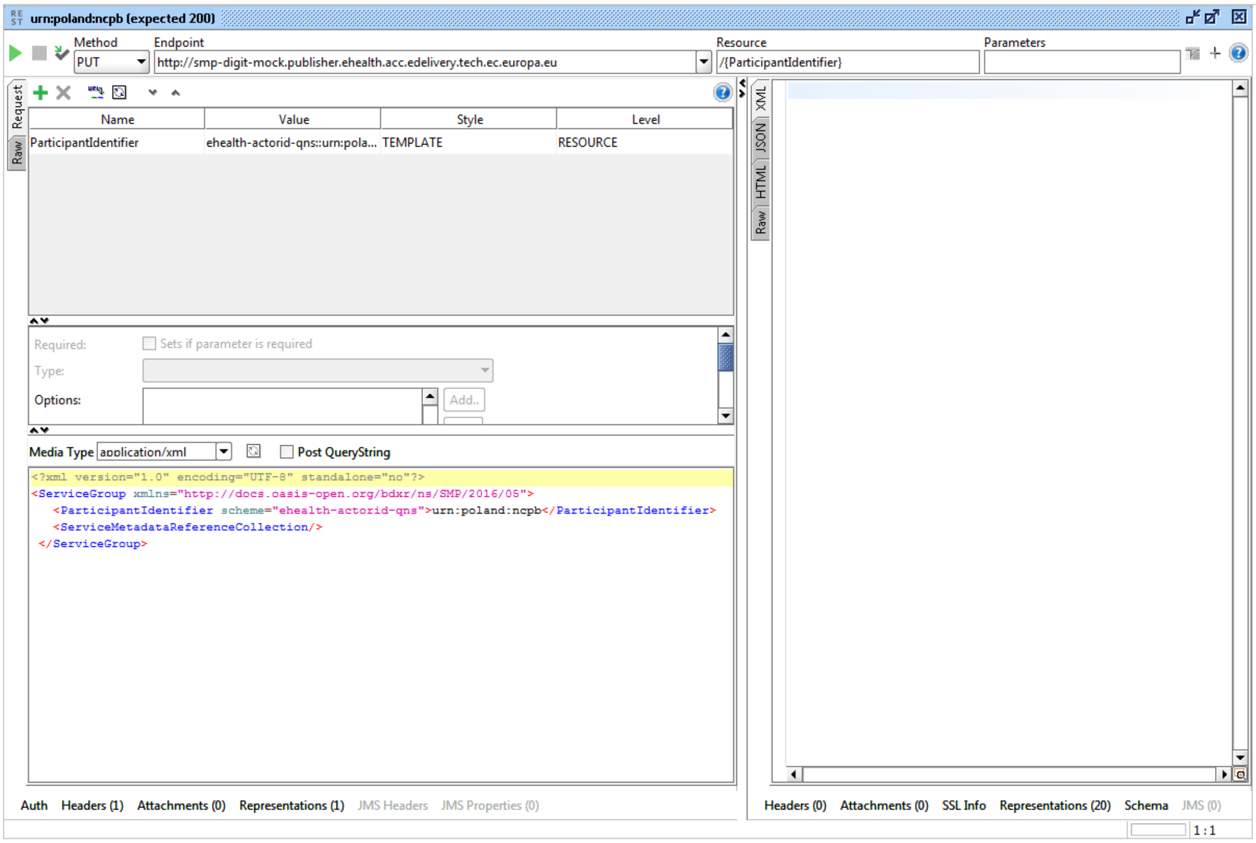
In the left navigation pane of the SoapUI interface, browse to the REST PUT method as shown below:



Property	Value
Name	UC02 - PUT
Description	
HTTP Method	PUT

By default, there are four PUT REST method examples. The next example will use the first one: ‘**urn:Poland:ncpb**’.

Open this REST method:


urn:poland:ncpb (expected 200)

Method: PUT

Endpoint: http://smp-digit-mock.publisher.ehealth.acc.edelivery.tech.ec.europa.eu

Resource: /{ParticipantIdentifier}

Name	Value	Style	Level
ParticipantIdentifier	ehealth-actorid-qns::urn:pola...	TEMPLATE	RESOURCE

Required: □ Sets if parameter is required

Type:

Options:

Media Type: application/xml

□ Post QueryString

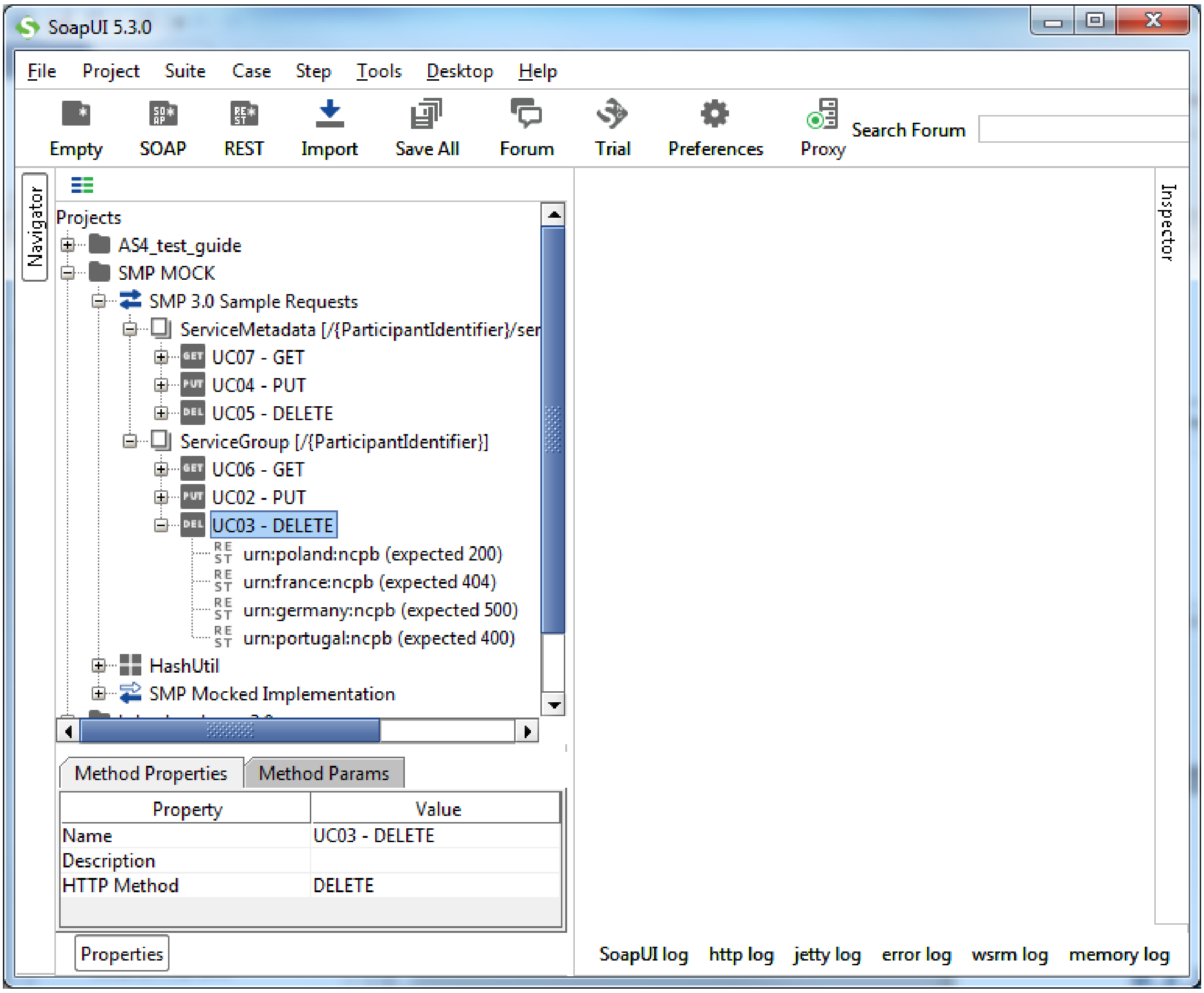
```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ServiceGroup xmlns="http://docs.oasis-open.org/bdxr/ns/SMP/2016/05">
   <ParticipantIdentifier scheme="ehealth-actorid-qns">urn:poland:ncpb</ParticipantIdentifier>
   <ServiceMetadataReferenceCollection/>
 </ServiceGroup>
```

8.1.2. *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.

8.1.3. *Delete a 'ServiceGroup'*

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



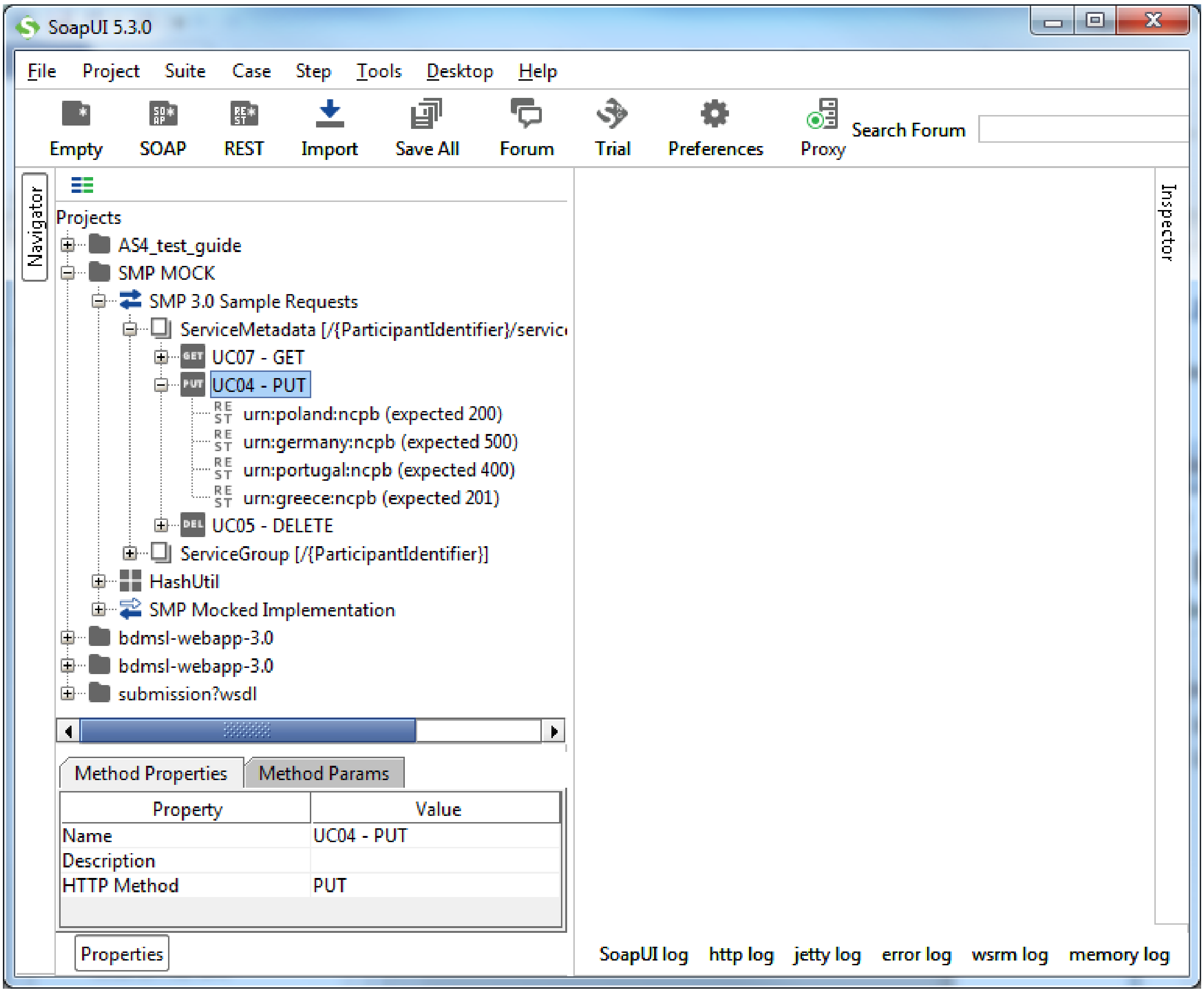
Projects
- AS4_test_guide
- SMP MOCK
  - SMP 3.0 Sample Requests
    - ServiceMetadata [/{ParticipantIdentifier}/ser
      - UC07 - GET
      - UC04 - PUT
      - UC05 - DELETE
    - ServiceGroup [/{ParticipantIdentifier}]
      - UC06 - GET
      - UC02 - PUT
      - UC03 - DELETE
        - urn:poland:ncpb (expected 200)
        - urn:france:ncpb (expected 404)
        - urn:germany:ncpb (expected 500)
        - urn:portugal:ncpb (expected 400)
  - HashUtil
  - SMP Mocked Implementation

Property	Value
Name	UC03 - DELETE
Description	
HTTP Method	DELETE

8.2. Creation, update and deletion of Service Metadata.

8.2.1. *Create a Service Metadata*

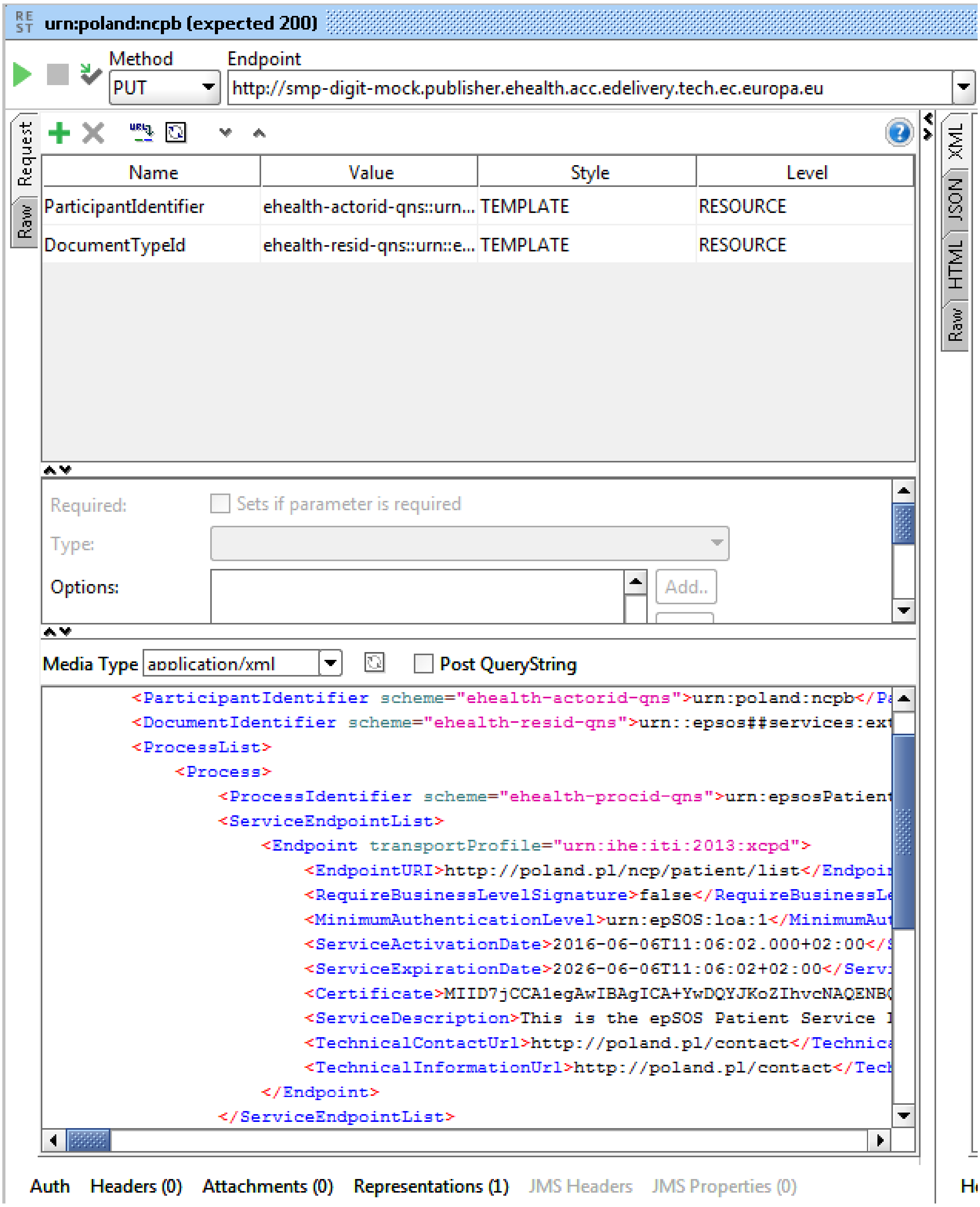
In the left navigation pane of the SoapUI interface, browse to the REST PUT method as shown below:


SoapUI 5.3.0
File Project Suite Case Step Tools Desktop Help
Empty SOAP REST Import Save All Forum Trial Preferences Proxy Search Forum
Navigator
Inspector
Projects
AS4_test_guide
SMP MOCK
SMP 3.0 Sample Requests
ServiceMetadata [/{ParticipantIdentifier}/servic
UC07 - GET
UC04 - PUT
urn:poland:ncpb (expected 200)
urn:germany:ncpb (expected 500)
urn:portugal:ncpb (expected 400)
urn:greece:ncpb (expected 201)
UC05 - DELETE
ServiceGroup [/{ParticipantIdentifier}]
HashUtil
SMP Mocked Implementation
bdmsl-webapp-3.0
bdmsl-webapp-3.0
submission?wsdl
Method Properties Method Params
Properties
SoapUI log http log jetty log error log wsrm log memory log

Property	Value
Name	UC04 - PUT
Description	
HTTP Method	PUT

The next example will use the following PUT Rest method: ‘**urn:Poland:ncpb**’.

Open this REST method:



urn:poland:ncpb (expected 200)

Method: PUT

Endpoint: http://smp-digit-mock.publisher.ehealth.acc.edelivery.tech.ec.europa.eu

Name	Value	Style	Level
ParticipantIdentifier	ehealth-actorid-qns::urn...	TEMPLATE	RESOURCE
DocumentTypeId	ehealth-resid-qns::urn::e...	TEMPLATE	RESOURCE

Required: □ Sets if parameter is required

Type:

Options:

Media Type application/xml □ Post QueryString

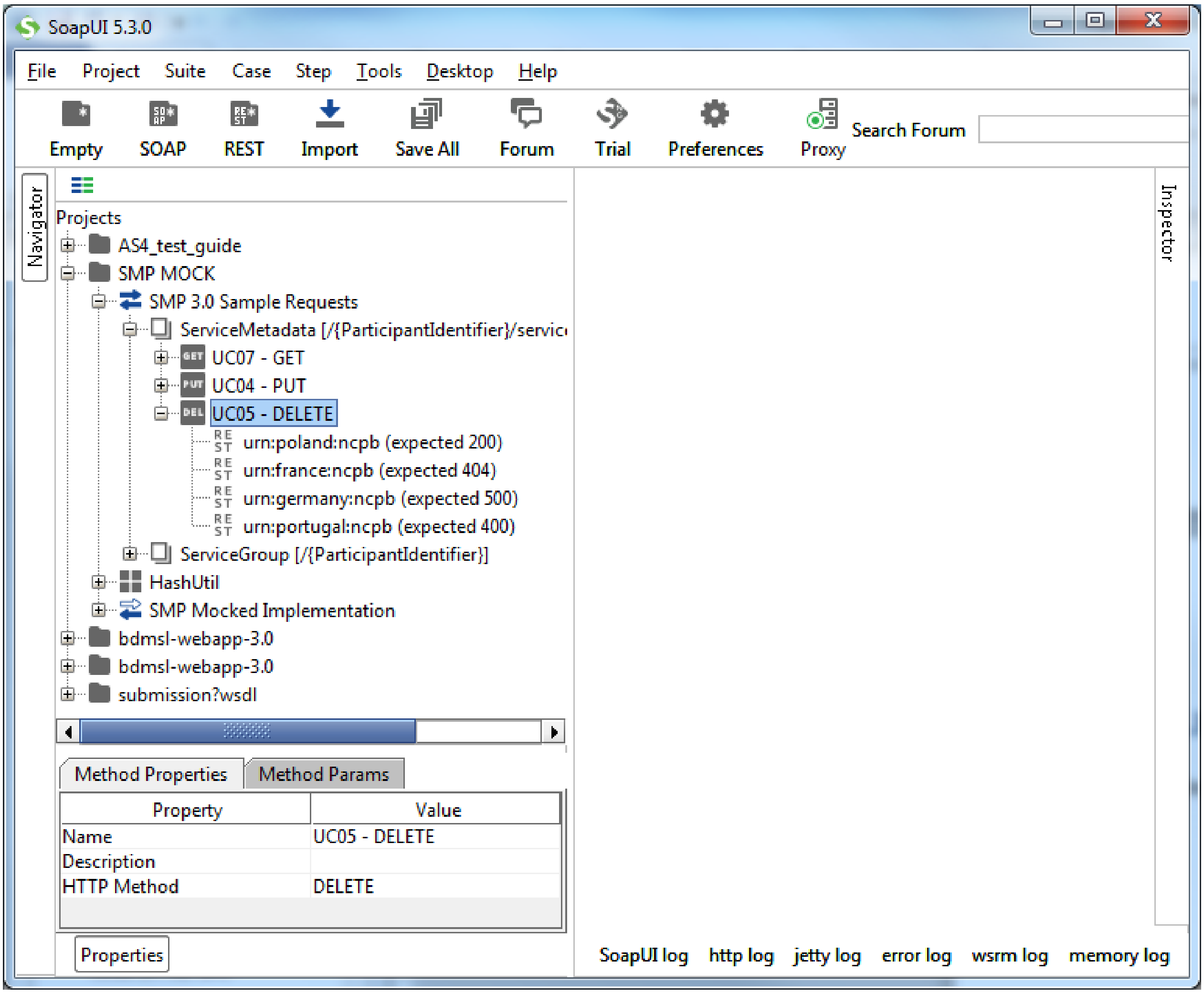
```
        <ParticipantIdentifier scheme="ehealth-actorid-qns">urn:poland:ncpb</Pa
        <DocumentIdentifier scheme="ehealth-resid-qns">urn::epsos##services:ext
        <ProcessList>
            <Process>
                <ProcessIdentifier scheme="ehealth-procid-qns">urn:epsosPatient
                <ServiceEndpointList>
                    <Endpoint transportProfile="urn:ihe:iti:2013:xcpd">
                        <EndpointURI>http://poland.pl/ncp/patient/list</Endpoi
                        <RequireBusinessLevelSignature>false</RequireBusinessLe
                        <MinimumAuthenticationLevel>urn:epSOS:loa:1</MinimumAut
                        <ServiceActivationDate>2016-06-06T11:06:02.000+02:00</S
                        <ServiceExpirationDate>2026-06-06T11:06:02+02:00</Serv
                        <Certificate>MIID7jCCA1egAwIBAgICA+YwDQYJKoZIhvcNAQENB(
                        <ServiceDescription>This is the epSOS Patient Service I
                        <TechnicalContactUrl>http://poland.pl/contact</Technica
                        <TechnicalInformationUrl>http://poland.pl/contact</Tech
                    </Endpoint>
                </ServiceEndpointList>
```

8.2.2. *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

8.2.3. *Delete Service Metadata*

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



Property	Value
Name	UC05 - DELETE
Description	
HTTP Method	DELETE

9. The SwaggerUI Interface

9.1. Introduction

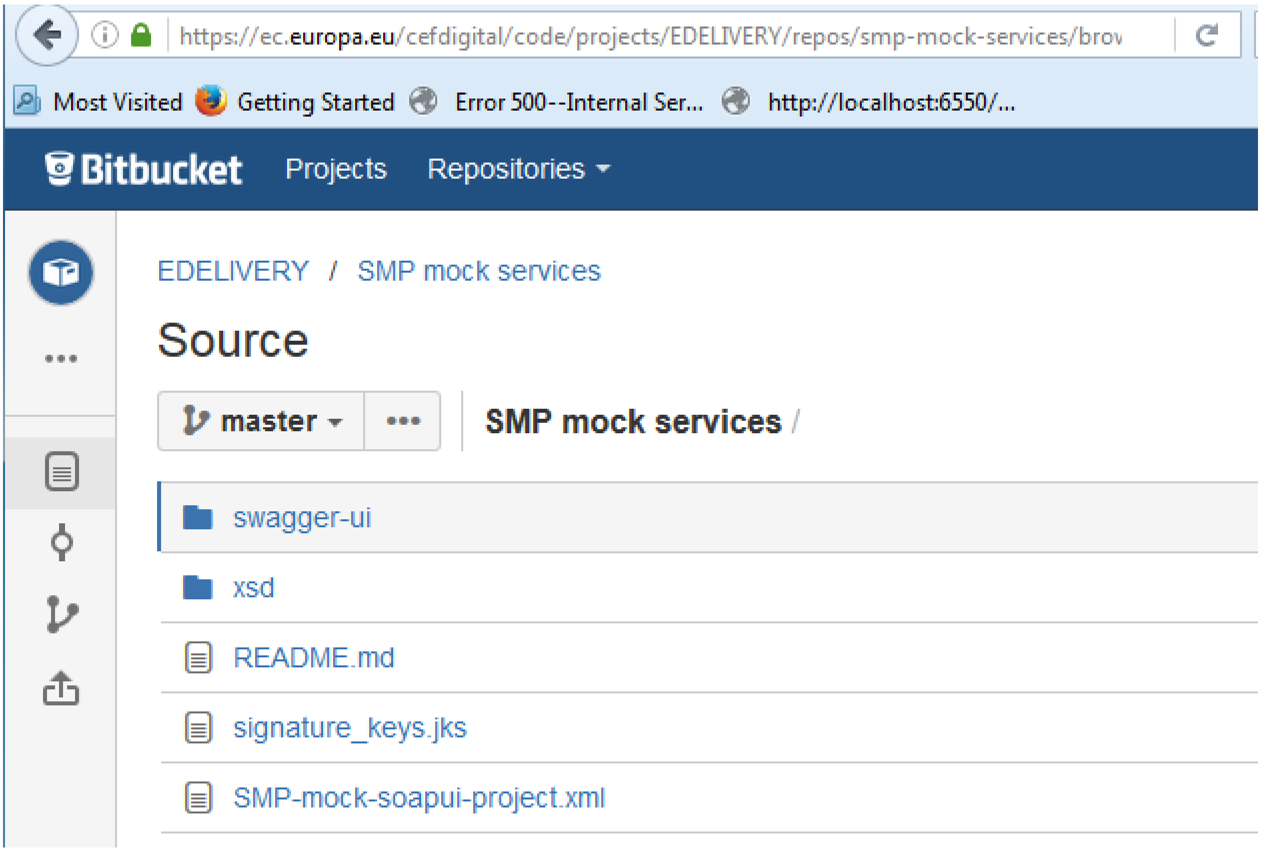
"***Swagger is an API developer tools for the OpenAPI Specification(OAS). It allows anyone (developers or end-users) to interact with the API's resources***". Quote from: http://swagger.io/

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.

9.2. Downloading the CEF SMP SwaggerUI web application project

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

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



https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp-mock-services/brov
Most Visited
Getting Started
Error 500--Internal Ser...
http://localhost:6550/...
Bitbucket
Projects
Repositories
EDELIVERY / SMP mock services
Source
master
SMP mock services /
swagger-ui
xsd
README.md
signature_keys.jks
SMP-mock-soapui-project.xml

Create a new 'swagger_temp' temporary directory

Within the previously created 'swagger_temp' directory, execute the following command

```
# git clone  https://ec.europa.eu/cefdigital/code/scm/edelivery/smp-mock-services.git
```

```
$ 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
```

9.3. 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",
```

By:

```
    "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/cipa-smp-full-webapp-3.X](http://localhost:7003/cipa-smp-full-webapp-3.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 3.X WEB client (based on Swagger-UI)"
  },
  "host": "localhost:7003",
  "basePath": "/cipa-smp-full-webapp-3.X",
  "externalDocs": {
  },
```

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

To generate the CEF 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/*
```

9.5. Deploy the SMP SwaggerUI war file

9.5.1. ***On Tomcat***

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

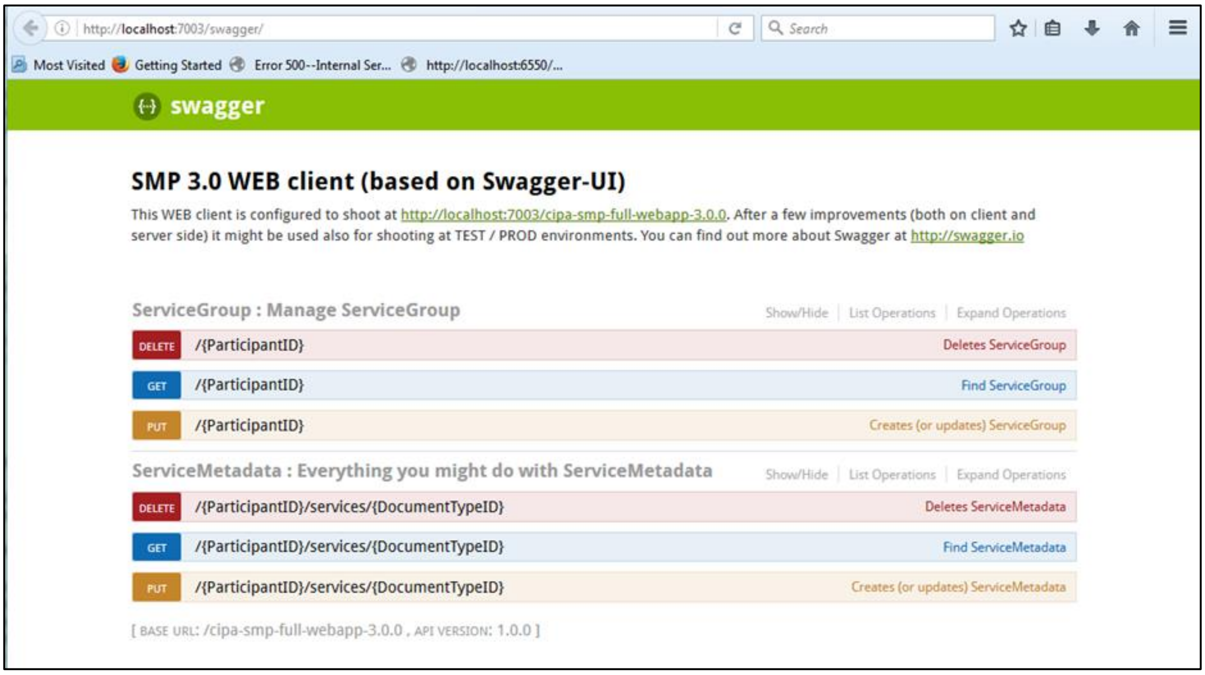
9.5.2. ***On WebLogic:***

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:



http://localhost:7003/swagger/
Search
Most Visited
Getting Started
Error 500--Internal Ser...
http://localhost:6550/...
swagger

SMP 3.0 WEB client (based on Swagger-UI)

This WEB client is configured to shoot at http://localhost:7003/cipa-smp-full-webapp-3.0.0. 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

ServiceGroup : Manage ServiceGroup

Show/Hide
List Operations
Expand Operations

DELETE	/{ParticipantID}	Deletes ServiceGroup
GET	/{ParticipantID}	Find ServiceGroup
PUT	/{ParticipantID}	Creates (or updates) ServiceGroup

ServiceMetadata : Everything you might do with ServiceMetadata

DELETE	/{ParticipantID}/services/{DocumentTypeID}	Deletes ServiceMetadata
GET	/{ParticipantID}/services/{DocumentTypeID}	Find ServiceMetadata
PUT	/{ParticipantID}/services/{DocumentTypeID}	Creates (or updates) ServiceMetadata

[BASE URL: /cipa-smp-full-webapp-3.0.0 , API VERSION: 1.0.0]

10. SMP Compilation

10.1. Compilation prerequisites

10.1.1. ***Supported Operating System Platform***

CEF SMP can be built on the following OS platforms:

- Windows Workstation & Server
- Linux platform

10.1.2. ***Software Requirements***

The following software components on the target system:

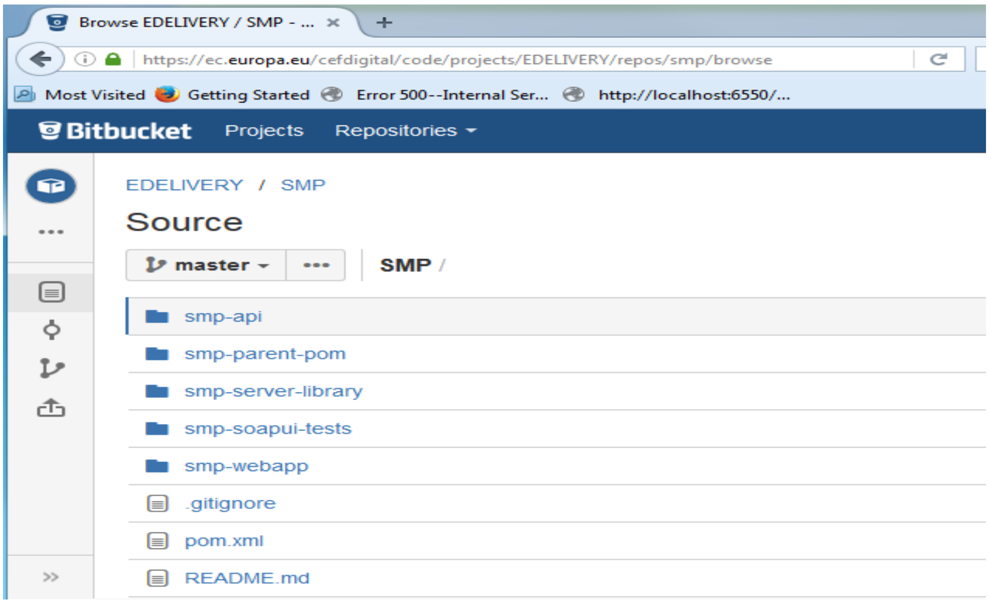
- Java Developement Kit environnement (JDK), version 7 or 8: http://www.oracle.com/technetwork/java/javase/downloads/index.html
- Maven 3.0 and avove (https://maven.apache.org/download.cgi)

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)

10.2. Downloading the source code

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

https://ec.europa.eu/cefdigital/code/scm/edelivery/smp.git



https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse

EDELIVERY / SMP

Source

master / SMP /

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

Browse EDELIVERY / SMP - ...
Most Visited
Getting Started
Error 500--Internal Ser...
http://localhost:6550/...
Bitbucket
Projects
Repositories

10.3. 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/cefdigital/code/scm/edelivery/smp.git***

```
# git clone  https://ec.europa.eu/cefdigital/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-parent-pom ..................................... SUCCESS [  0.120 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 ‘cipa-smp-full-webapp.war’ located in the ‘smp-webapp/target/’ directory

```
ls ./smp-webapp/target
cipa-smp-full-webapp-3.X  cipa-smp-full-webapp.war  classes  generated-sources
generated-test-sources  maven-status  test-classes  webapp-classes
```

11. Annexes

11.1. The SMP 'config.properties' file

The CEF SMP configuration is performed via the 'config.properties' file.

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

```
#
# Version: MPL 1.1/EUPL 1.1
#
# The contents of this file are subject to the Mozilla Public License Version
# 1.1 (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at:
# http://www.mozilla.org/MPL/
#
# Software distributed under the License is distributed on an "AS IS" basis,
# WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
# for the specific language governing rights and limitations under the
# License.
#
# The Original Code is Copyright The PEPPOL project (http://www.peppol.eu)
#
# Alternatively, the contents of this file may be used under the
# terms of the EUPL, Version 1.1 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 at:
# http://joinup.ec.europa.eu/software/page/eupl/licence-eupl
#
# 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.
#
# If you wish to allow use of your version of this file only
# under the terms of the EUPL License and not to allow others to use
# your version of this file under the MPL, indicate your decision by
# deleting the provisions above and replace them with the notice and
# other provisions required by the EUPL License. If you do not delete
# the provisions above, a recipient may use your version of this file
# under either the MPL or the EUPL License.
#

# Blue Coat reverse-proxy authentication, by default disabled (false)
# Be careful with switching it to 'true' - do it only if you know what you are doing.
# Authentication with Blue Coat means that all HTTP requests having 'Client-Cert' header will be authenticated
# as username placed in the header.
# Never expose SMP to the WEB without properly configured reverse-proxy and active blue coat.
authentication.blueCoat.enabled=false

## Only set to false in PRODUCTION mode. This variable is used to clear the context path of the SMP
contextPath.output=true

## Most Java libraries and J2EE containers block encoded slashes in URL - For security reasons.
## Theoretically there are no restrictions on slash "/" characters in document or participant identifiers,
## but by default we block then as well.
## If slash "/" characters must be supported, than switch this property to "true".
## Remember that in such case the relevant change should be also applied on J2EE level
## I.e. for Tomcat it is handled by property: org.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true
encodedSlashesAllowedInUrl=false

## DBMS handler
dataManager.class=eu.europa.ec.cipa.smp.server.data.dbms.DBMSDataManager
eclipselink.ddl-generation.output-mode=sql-script

## All Identifiers by default are CASE-INSENSITIVE.
## Specifies schemes of participant/document identifiers that must be considered CASE-SENSITIVE.
## List values (delimited by pipe character: "|" ) placed here are checked against runtime (request) schemes in the CASE-INSENSITIVE way
identifiersBehaviour.caseSensitive.ParticipantIdentifierSchemes=casesensitive-participant-scheme1|casesensitive-participant-scheme2
identifiersBehaviour.caseSensitive.DocumentIdentifierSchemes=casesensitive-doc-scheme1|casesensitive-doc-scheme2


## Registration callback (SML client caller)
registrationHook.class=eu.europa.ec.cipa.smp.server.hook.RegistrationServiceRegistrationHook
#registrationHook.class=eu.europa.ec.cipa.smp.server.hook.DoNothingRegistrationHook


# SMP ID
regServiceRegistrationHook.id=TEST-SMP-ID1


# 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 cert verification made normally by SSL terminator)
regServiceRegistrationHook.clientCert=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


# SML URL (incl. the service name)
#regServiceRegistrationHook.regLocatorUrl=https://sml.peppolcentral.org/manageparticipantidentifier
# regServiceRegistrationHook.regLocatorUrl=https://smk.peppolcentral.org/manageparticipantidentifier
regServiceRegistrationHook.regLocatorUrl=http://localhost:8080/manageparticipantidentifier
regServiceRegistrationHook.keystore.classpath =
regServiceRegistrationHook.keystore.password  =


## XMLDSIG response signing:
xmldsig.keystore.classpath    = ../keystore/keystore.jks
xmldsig.keystore.password     = peppol
xmldsig.keystore.key.alias    = smp keypair
xmldsig.keystore.key.password = peppol
```

```
## JDBC configuration for DB
jdbc.driver = com.mysql.jdbc.Driver
jdbc.url = jdbc:mysql://localhost:3306/smp
jdbc.user = smp
jdbc.password = the_password
target-database = MySQL
```

11.2. Detailed SMP configuration file (smp.config.properties)

The ‘WEB-INF/classes/config.properties’ file is used to configure various SMP properties, the following table describes them briefly, detailed excplaination are describes after this table.

Parameter	Default Value	Comment
authentication.blueCoat.enable	false	Authentication with Blue Coat means that all HTTP requests having 'Client-Cert' header will be authenticated as username placed in the header. # Never expose an SMP to the WEB without properly configured reverse-proxy and active blue coat.
contextPath.output	true	PRODUCTION mode. This variable is used to clear the context path of the SMP
org.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH	true	Most Java libraries and J2EE containers block encoded slashes in URL - For security reasons. Theoretically there are no restrictions on slash "/" characters in document or participant identifiers, but by default we block then as well. If slash "/" characters must be supported, than switch this property to "true". Remember that in such case the relevant change should be also applied on J2EE level I.e. for Tomcat it is handled by property:
encodedSlashesAllowedInUrl	false	
dataManager.class	eu.europa.ec.cipa.smp.server.data.dbms.DBMSDataManager	DBMS handler
eclipselink.ddl-generation.output-mode	sql-script	DBMS handler
identifiersBehaviour.caseSensitive.ParticipantI	casesensitive-participant-scheme1|casesensitive-participant-scheme2	## All Identifiers by default are CASE-

dentifierSchemes		INSENSITIVE. ## Specifies schemes of participant/document identifiers that must be considered CASE-SENSITIVE. ## List values (delimited by pipe character: "|") placed here are checked against runtime (request) schemes in the CASE-INSENSITIVE way
identifiersBehaviour.caseSensitive.DocumentIdentifierSchemes	casesensitive-doc-scheme1|casesensitive-doc-scheme2	
registrationHook.class	eu.europa.ec.cipa.smp.server.hook.RegistrationServiceRegistrationHook eu.europa.ec.cipa.smp.server.hook.DoNothingRegistrationHook	Registration callback (SML client caller)
regServiceRegistrationHook.id	TEST-SMP-ID1	SMP ID / SMP Name
regServiceRegistrationHook.clientCert	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	#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 cert verification made normally by SSL terminator)
regServiceRegistrationHook.regLocatorUrl	#https://sml.peppolcentral.org/manageparticipantidentifier #https://smk.peppolcentral.org/manageparticipantidentifier http://localhost:8080/manageparticipantidentifier	The URL of the targeted SML (incl. the service name)
regServiceRegistrationHook.keystore.classpath		Thye location of the keystore
regServiceRegistrationHook.keystore.password		The password of the keystore
xmldsig.keystore.classpath	../keystore/keystore.jks	The location of the xmldsig keystore
xmldsig.keystore.password	peppol	The password of the xmldsig keystore
xmldsig.keystore.key.alias	smp keypair	The alias of the xmldsig key
xmldsig.keystore.key.password =	peppol	The password of the xmldsig key
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
jdbc.read-connections.max	10	Database Configuration: Max Read Connection

11.2.1. *registrationHook.class*

The registrationHook.class determines the integration of the SMP with and SML & DNS. Two values are recognised as described below:

eu.europa.ec.cipa.smp.server.hook.DoNothingRegistrationHook'

The SMP will only populate the database and will not forward the GroupParticipantid request to THE SML.

This do not apply to the metadata as they are not distributed to the DNS

eu.europa.ec.cipa.smp.server.hook.RegistrationServiceRegistrationHook

The SMP will store the REST ParticipantId request to the SMP Database and forward these request via 'SOAP' messages to the SML SML. The SML will store these Participand ID request in the SML database and the will update the DNS

This do not apply to the metadata as they are not distributed to the DNS

11.2.2. *regServiceRegistrationHook.id*

This is the name of the SMP.

11.2.3. *regServiceRegistrationHook.clientCert*

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 cert verification made normally by SSL terminator) .

11.2.4. *regServiceRegistrationHook.regLocatorUrl*

Only used if SML/DNS intgegrationn has been selected.

In case the integration tith the SML/DNS is needed, this value has to be set to the address of the SML or the loadBalancer/proxy tagetting these SML instance(s)

11.2.5. *XMLDSIG response signing*

```
xmldsig.keystore.classpath    = ../keystore/keystore.jks
xmldsig.keystore.password     = peppol
xmldsig.keystore.key.alias    = smp keypair
xmldsig.keystore.key.password = peppol
```