**CIPA e-Delivery**

*Test Framework*

**Document history**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Comment | Modified pages |
| 1.00 | 30/11/2012 | Creation for CIPA e-Delivery v1.0.0 | All pages |
| 1.10 | 31/05/2013 | Update for CIPA e-Delivery v1.1.0 | All pages |
| 1.20 | 29/11/2013 | Update for CIPA e-Delivery AS2 beta release | All pages |
| 1.30 | 28/05/2014 | Update for CIPA e-Delivery v2.1.0 | All pages |

CONTENTS

[1. Overview 3](#_Toc373502759)

[2. Test Cases 3](#_Toc373502760)

[3. Supporting Tools 3](#_Toc373502761)

[3.1. Instructions SoapUI 4](#_Toc373502762)

[3.2. Instructions Access Point Client 5](#_Toc373502763)

[3.3. Other tools 5](#_Toc373502764)

# Overview

The CIPA test framework consists of test cases and supporting tools to assist the user in running test cases for the supported CIPA components:

* Service Metadata Locator (SML),
* Service Metadata Publisher (SMP),\
* START Access Point (START AP).
* AS2 Access Point (AS2 AP).

# Test Cases

The test cases are detailed in a worksheet per component (SML, SMP, START AP and AS2 AP).

Each worksheet consists of:

* a "Test results" page where a summary of the "passed", "failed" and "not run" test cases is displayed.
* A "Test details" page where the test cases are explained in detail, together with guidelines on how to perform the tests.

The user can use this worksheet as a guide during a test session and complete the blank fields based on the results of his test run.

The resulting completed worksheet can then be discussed with team members or shared with other parties for information or follow-up.

The current set of test cases is extracted several sources:

* The PEPPOL specification documents
* Test cases based on community feedback,
* Test cases based on additional specifications or requirements,
* Regression tests based on bug-fixes.

Additional tests will be added each time that new information is available from any of these sources.

# Supporting Tools

Supporting tools help the user to call the services exposed by each component (SML, SMP, START AP and AS2 AP). To test the scenarios in the different test cases, the user can either change the test data or enable some test options depending on the tool.

## Instructions SoapUI

For the SML, SMP and AS2 AP, a SoapUI project is available to call the SOAP services or REST services using sample data.

SoapUI is an open source test tool for web-service testing. A free version without restrictions is available and this free version is sufficient to run the projects in this package.

Download SoapUI free via <http://www.soapui.org/> and follow the installation instruction on the website.

Inside SoapUI, create a new workspace (File->New Workspace) and load the three projects in this package ([https://joinup.ec.europa.eu/svn/cipaedelivery/trunk/cipa-core/Test/Supporting Tools](https://joinup.ec.europa.eu/svn/cipaedelivery/trunk/cipa-core/Test/Supporting%20Tools)) using the import functionality of SoapUI (File->Import Project).

Update the CIPA dispatcher endpoint for the AS2 AP test if the dispatcher is not running on your local machine (note: to configure the CIPA dispatcher and send a message to a receiver on another machine, consult the AS2 installation guide).

Update the WSDL URLs and service endpoints of the SML test.

To enable two-way SSL in the context of the SML, update the keystore settings via (File->Preferences->SSL Settings). In addition to specifying the Keystore location and password, ensure that the Client Authentication box is checked.

Update the REST Endpoints and Resource Paths of the SMP test.

If a username/password is required, it can be configured in the Request Properties of each service request.

Modify the Request in the left window to your requirements and press the green play/send button to submit the request and to see the response in the right window.

In addition to functional tests using single service calls, SoapUI also supports load testing in different forms. For the AS2, SMP and SML components, load test samples have been added that simulate multiple concurrent users accessing the components. These load test can be run as-is or modified (e.g. threading strategy or duration) to fit specific needs.

## Instructions Access Point Client

For the START AP, an Access Point Client is available to call the AP services using sample data.

Unzip ApClient.zip to your disk and follow the instructions in the readme.txt file to configure the client and send messages to the AP.

Since the Access Point Client is identical to the sender component (uses the send API) of the START Access Point, the Access Point Client can also serve the role as source Access Point in several test cases.

As an alternative to the Access Point Client, it is possible to use the AS2 AP soapUI project to send a START message. If the receiver (in StandardBusinessDocument\StandardBusinessDocumentHeader\Receiver\Identifier) is configured in the SMP to use the START protocol, the CIPA dispatcher will dynamically determine that the START protocol should be used for the message transmission.

## Other tools

Some test cases require manual checks. To help the user in executing these test cases, other tools not included in this package can be used.

Verifications to be done in the message exchange data can be facilitated using a network protocol analyser (e.g. Wireshark is a free and open source tool whereby a user can trace and analyse his network connection). To facilitate the network trace and for testing purposes, temporarily configure the services to use HTTP instead of HTTPS.

Verifications on web-service interoperability (WS-I) can be facilitated using the functionality that is included in SoapUI and/or the tools available from <http://www.ws-i.org/deliverables/matrix.aspx>