5th INTEGRATION TEST CAMP
Review

How was the Integration Test Camp carried out?

It was performed as a virtual event where participants could test the interoperability of their components within an IDSA implementation composed of connectors, DAPS and Broker. Each participant was assigned with a slot of two hours in which the SQS lab was entirely dedicated for them. The slots lasted more than originally excepted to get the new team in touch with the components presented.

SQS team kept the same environment as the 4th Integration Test. We are actively looking for a AppStore and a Clearing House.

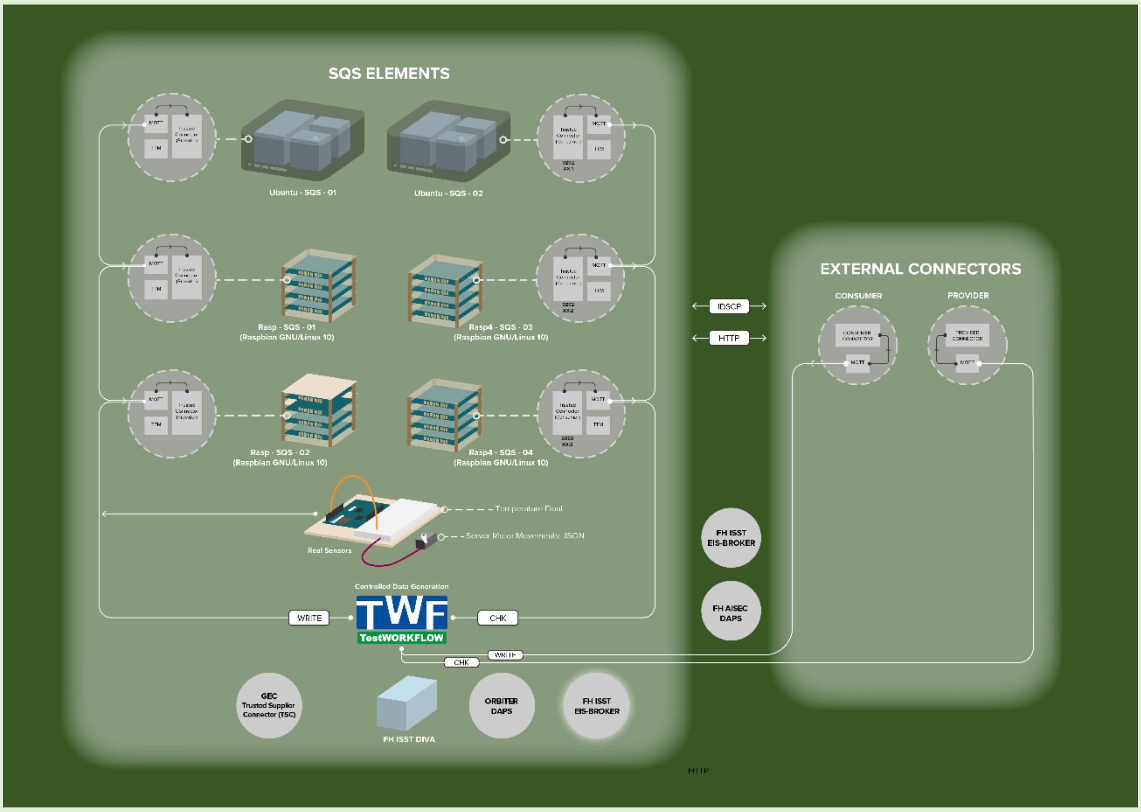


Figure 1 - SQS lab environment (4th edition)

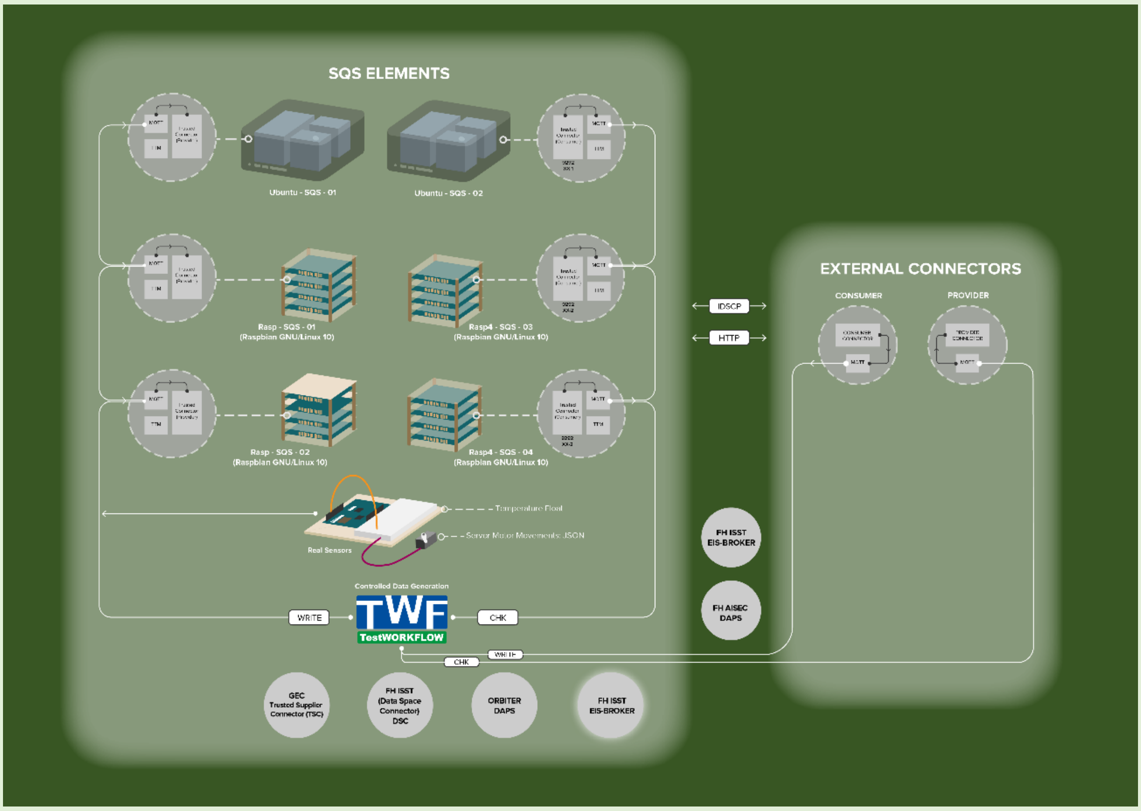


Figure 2 - SQS lab environment (4th edition)

With this architecture, the next scenarios were offered:

1. Connector Under Test (CUT) as Provider and SQS lab connector as Consumer, with IDSCP, IDSCPv2 or HTTP communication protocol
2. Connector Under Test (CUT) as Consumer and SQS lab connector as Provider, with IDSCP, IDSCPv2 or HTTP communication protocol
3. Connector Under Test (CUT) and lab Orbiter DAPS
4. Connector Under Test (CUT) and lab Broker (FH IAIS Broker)

In all the scenarios the DAPS in SQS lab (Orbiter DAPS) and the DAPS from FH were offered. **Including** the 4th one. It was discovered during one of the Integration Camp sessions that a connector with an Orbiter DAT can successfully register to the FH IAIS Broker.

5th INTEGRATION TEST CAMP
Review

Who participated?

The schedule for this Integration Test Camp:

21-Jan	22-Jan	29-Jan
	10:00-12:30 ENGINEERING	
13:00-15:30 FIWARE		15:00-17:00 FIWARE (2)

What were the participants able to do there?

Participants were able to test the interoperability of their pre-commercial components within a real IDSA architecture. They could interact with real IDSA infrastructure components and verify how their components will act in the real world.

For that, SQS lab team proposed the next Test Cases:

- Step 0. Receive Test Camp information

  The participants receive an encrypted document with all the information needed to connect to the Integration Test Camp components.

- Step 1. Environment configuration

  The CUT is registered in the DAPS and configured to communicate with the lab connector.

- Step 2. Data interchange

  After a successful configuration, the CUT is able to send/receive data correctly.

- Step 3. Wrong connection attempt - With an expired certificate

  The consumer connector with an expired certificate, tries to establish a connection with the provider connector, asking for data, and the request is rejected.

- Step 4. Wrong connection attempt – With an expired DAT

  The consumer connector with an expired token, tries to establish a connection with the provider connector, asking for data, and the request is rejected.

- Step 5. Connection failure (only if data flow is continuous)

  While the CUT is sending/receiving data, the lab connector is turned off. 2 minutes later it is turned on, the connection is re-established and data flows correctly.
  *DAT expiration greater than 5 minutes

- Step 6. Connection with more than one provider (optional)

  The CUT as consumer is configured to receive data from more than one provider.

- Step 7. Connection with more than one consumer (optional)

  The CUT as provider is configured to send data to more than one consumer.

- Step 8. Connection failure with more than one provider (only if you have gone throw Step 6 or 7)

  While the CUT is sending/receiving data, one of the lab connectors is turned off. 2 minutes later it is turned on. The connection is re-established and data flows correctly. Then, both lab connectors are turned off. 2 minutes later they are turned on. The connections are re-established and data flows correctly.

- Step 9. Wrong Registration attempt - Without self-information

  CUT, without self-information, tries to register to the Broker and the request is rejected.

- Step 10. Wrong Registration attempt – Without a valid DAT
  CUT, without a valid DAT, tries to register to the Broker and the request is rejected.

- Step 11. Connector Registry to the Broker

  The CUT makes a request to register to the Broker. The Broker accepts the request and registers the CUT.

- Step 12. Connector **Registry to the Broker with Orbiter DAT**
  The CUT makes a request to register to the Broker with an Orbiter DAT. The Broker accepts the request and registers the CUT.

- Step 13. Data interchange with the information received from the Broker

  The CUT makes a request for lab connector information to a Broker and then makes a connection to interchange data with the received information.

- Step 14. Data interchange with the information received from the Broker

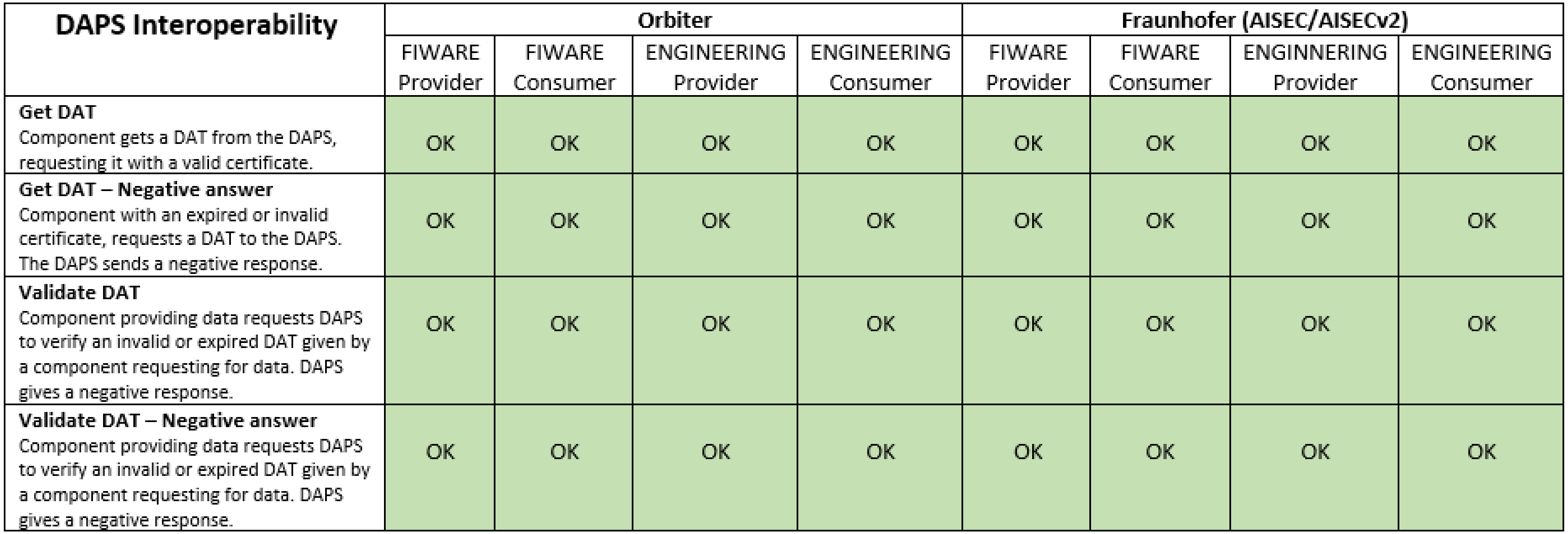
  The CUT makes a request for lab connector information to a Broker and then makes a connection to interchange data with the received information.

- Step 15. Unregister connector from the Broker

  The CUT requests to delete its information from the Broker.

What have been achieved during the Test Camp?

In the next tables the results of the Test Camp are shown:



DAPS Interoperability	Orbiter				Fraunhofer (AISEC/AISECv2)			
	FIWARE Provider	FIWARE Consumer	ENGINEERING Provider	ENGINEERING Consumer	FIWARE Provider	FIWARE Consumer	ENGINNERING Provider	ENGINEERING Consumer
Get DAT Component gets a DAT from the DAPS, requesting it with a valid certificate.	OK	OK	OK	OK	OK	OK	OK	OK
Get DAT – Negative answer Component with an expired or invalid certificate, requests a DAT to the DAPS. The DAPS sends a negative response.	OK	OK	OK	OK	OK	OK	OK	OK
Validate DAT Component providing data requests DAPS to verify an invalid or expired DAT given by a component requesting for data. DAPS gives a negative response.	OK	OK	OK	OK	OK	OK	OK	OK
Validate DAT – Negative answer Component providing data requests DAPS to verify an invalid or expired DAT given by a component requesting for data. DAPS gives a negative response.	OK	OK	OK	OK	OK	OK	OK	OK

Connector Interoperability	Orbiter				Fraunhofer (AISEC/AISECv2)			
	FIWARE Provider	FIWARE Consumer	ENGINEERING Provider	ENGINEERING Consumer	FIWARE Provider	FIWARE Consumer	ENGINEERING Provider	ENGINEERING Consumer
Data interchange Consumer connector requests data, with a valid DAT, to the provider connector. Provider connector verifies DAT, and the data is interchanged.	OK	OK	OK	OK	OK	OK	OK	OK
Connector Registry – Negative answer Without a self-description or with a wrong self-description.	OK	OK	OK	OK	OK	OK	OK	OK
Data interchange – Negative answer Consumer connector tries to get access to a resource from an already known provider connector, with an invalid DAT. The request in rejected.	OK	OK	OK	OK	OK	OK	OK	OK
Connection failure (only if data flow is continuous) While the provider connector is sending data, the consumer connector is turned off. 2 minutes later it is turned on, the connection is re-established and data flows correctly.	NA	NA	NA	NA	NA	NA	NA	NA
Connection with more than one provider (optional) The consumer receives data from more than one provider.	NA	NA	NA	NA	NA	NA	NA	NA
Connection with more than one consumer (optional) The provider sends data to more than one consumer.	NA	NA	NA	NA	NA	NA	NA	NA
Data interchange – with Broker mediation	NA	NA	NA	NA	NA	NA	NA	NA

5th INTEGRATION TEST CAMP
Review

Broker Interoperability	Fraunhofer (AISEC/AISECv2)			
	FIWARE Provider	FIWARE Consumer	ENGINEERING Provider	ENGINEERING Consumer
Connector Registry The connector gets registered in a Broker.	OK	OK	OK	OK
Connector Registry – Negative answer Without a self-description or with a wrong self-description.	OK	OK	OK	OK
Connector Registry – Negative answer With a wrong DAT.	OK	OK	OK	OK
Get data (request for everything) The connector gets all the information contained in a Broker.	OK	OK	OK	OK
Get data (request for specific connector) The connector gets the information of a specific connector to a Broker.	NA	NA	NA	NA
Get data (request for specific content) The connector gets the information of a specific content to the Broker	NA	NA	NA	NA
Update connector information The connector updates its information to a Broker.	OK	OK	OK	OK
Unregister connector The connector gets unregistered from a Broker.	OK	OK	OK	OK

Broker Interoperability	Orbiter DAPS	
	FIWARE	ENGINEERING
Connector Registry The connector gets registered in a Broker	OK	OK

What difficulties have been encountered?

- The local broker had technical issues. It was unavailable for this Integration Test Camp.
  - ➢ In touch with IAIS Fraunhofer to be able to solve this.
- Issues connecting the CUT to the GEC connector.
  - ➢ Some modifications were required, fixed within the Integration Test Camp.

Discovered

During one of the Integration Test Camp sessions, we were able to discover that a **connector with a valid Orbiter DAT can register to the Broker**. We were testing the CUT and Broker interoperability when we came across this. This was a test case that has not been performed before. The CUT obtained a valid DAT from the Orbiter DAPS and tried registering to the Broker (ConnectorAvailableMessage). The expected outcome was a rejection message as the Broker responds to AISEC DATs. To make sure the Broker was working as intended, we then registered a connector with an AISEC valid DAT. A query request (QueryMessage) revealed the connector registered with the Orbiter DAT in the list.

The Broker developers have been contacted to ensure this is how they intend it to work as there is no information about this being the Broker's behaviour.

Conclusions

SQS wants to thank all the participants that have taken part. It has been a great opportunity to meet each participant and to understand their component better. It allowed us to get a deeper knowledge of what is being developed in IDSA environment. The Integration Test Camp has been a great opportunity to collaborate and face the needs of the components to work with each other.

With the feedback received and the lessons learned from the problems faced in the session, SQS will keep improving the lab environment and the organization for the next Integration Test Camp.