

Overview

FIWARE is a complete framework based on open source components, designed to enable rapid development and deployment of the Internet of Things (IoT) solutions. FIWARE is based on the NGSIv2 (Next Generation Service Interfaces Architecture version 2) from OMA. FIWARE provides a set of license-free APIs that can be used to manage sensor data with context information. Arrowhead is a framework targeted at industrial automation and IoT applications. This task has focused on adding a Service translator for Arrowhead that can exchange information with a FIWARE Broker, enabling Arrowhead's functionality on already existing FIWARE installations, and also expose Arrowhead's valid producers into the FIWARE network.

In traditional automation systems, most often a single communication protocol is used. In contrast, the Arrowhead Framework was designed with interoperability and security in focus. This enables new protocols to be added to the already supported technologies.

By adding support for FIWARE, Arrowhead can now make use of features such context (location, type, etc.) by having a bi-directional translation capability.

Implementation

The focus of this implementation has been to enable compatibility bidirectionally between Arrowhead and FIWARE in the most transparent way, in other words, having zero impact on the already existing installations. For that reason, two services have been created on the Translator Core Service:

Plugin: This service is able to get all the information from the FIWARE network, extract all the devices information and register them into Arrowhead. In parallel, it creates an individual resource that Arrowhead consumers can use to get directly the data of a specific device of FIWARE. It also translates automatically, if

requested, the semantic of the data from JSON to plain text or SenML.

Custom Broker: This service acts as a FIWARE broker, but internally it connects to a standard FIWARE broker and it reads all the messages between the FIWARE clients and the broker. It also has the capability to parse those messages and complete them, or just respond to them by injecting information from the Arrowhead Cloud.

Usage

This implementation requires an Arrowhead Cloud and a FIWARE broker (see Fig.1). The only requirement is that the arrowhead Translator must have the information to contact the broker. Then, everything is done by the Plugin and the Fake Broker as it has been explained in the previous section.

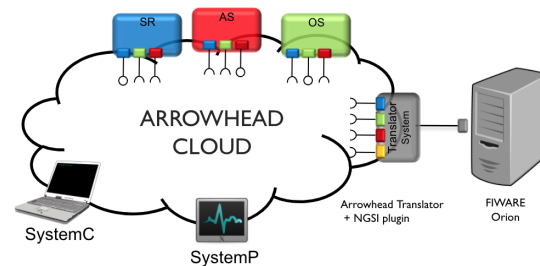


Figure 1: Arrowhead local cloud

Fig. 1. shows an Arrowhead local cloud with an NGSI plugin for the Translator Core system. The plugin enables Arrowhead systems and FIWARE clients to exchange bi-directional information.

There are four possible scenarios for this implementation:

1. **AH consumer - AH producer:** There is no change from normal AH interaction.
2. **AH consumer - FIWARE:** The FIWARE service must be registered on the broker (as normal FIWARE procedure). For this step, we can use a standard broker or the fake broker exposes at Translator Service. After that, the Plugin service will detect the registration and automatically will produce a

service for interfacing that FIWARE service, and it will be registered at Service Registry being accessible for the AH consumer and usable as a standard AH producer.

3. **FIWARE - AH producer:** If the Fake Broker can access the AH producer on the Service Registry, it will add that AH producer as a FIWARE service on the real Broker. After that, FIWARE clients can access the information from the real Broker or the Fake Broker service.
4. **FIWARE - FIWARE:** This communication is completely transparent, both can communicate using only the real broker, only the fake broker or using both simultaneously.

Results

The main results from this work are listed below;

1. A technology investigation of the capabilities of FIWARE
2. System design for integrating FIWARE with Arrowhead
3. A Java-implementation of a FIWARE compliant plugin to the existing Arrowhead Translator service.

Minor results also include; test software with clients (shown in Fig 2), use cases and a hardware demonstrator capable of sensing sensor data and communication both to a FIWARE broker and an Arrowhead local cloud.

Future work

Future work consists of the following items:

1. Adding support for more service types.
2. Including support for more Core services, such as EventHandler, Gateway / Gatekeeper, etc.
3. Investigation how Access control can be integrated with FIWARE



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 767498.

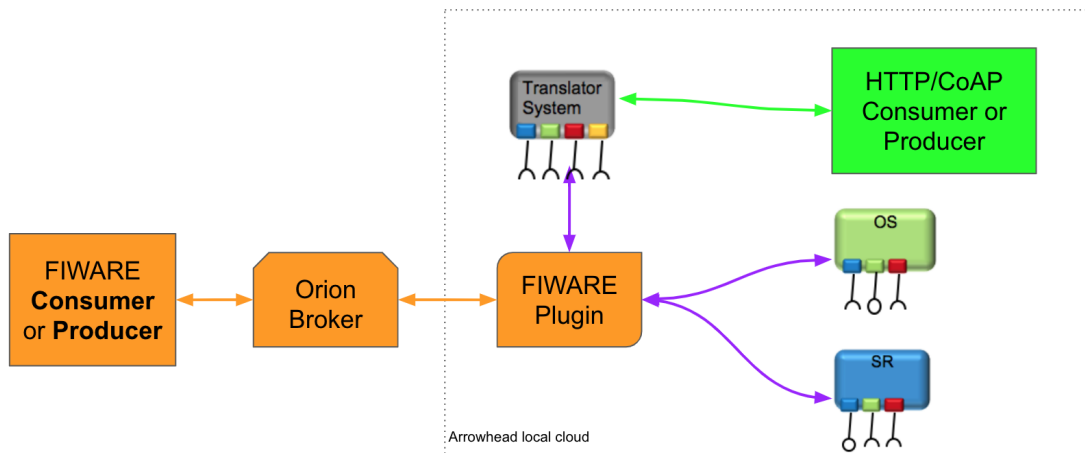


Figure 2: FIWARE/Arrowhead local cloud integration setup