# Translator FIWARE

## Service Description

Service ID: *"translator-fiware"*

**Abstract**
This document describes an abstract service that provides functions and features to communicate producers and consumers with different protocols.

ARTEMIS Innovation Pilot Project: Arrowhead
THEME [SP1-JTI-ARTEMIS-2012-AIPP4 SP1-JTI-ARTEMIS-2012-AIPP6]
[Production and Energy System Automation Intelligent-Built environment and urban infrastructure for sustainable and friendly cities]

# Contents

# 1 Overview

This document describes an abstract Eclipse Arrowhead service that was designed to allow Arrowhead Producers and Consumers to be consumed by FIWARE [1] Clients and to consume FIWARE Entities (respectively) in a transparent implementation. The Translator FIWARE service thus enables Arrowhead-compliant systems and FIWARE entities to exchange information.

The rest of this document is organized as follows. In the remainder of this section we consider significant prior art, describe how this service is meant to be used and comment on the status of this document. In Section 2, we describe the abstract interface, in terms of functions invoked by messages, provided by this service. Finally, in Section 3, we present the data types used by those functions. For more information about NGSI data models, see [2].

## 1.1 Introduction

This Arrowhead Translator Core system [3] is a vital part for enabling message and data exchange in an Arrowhead local cloud with different protocols. The Translator's FIWARE service provides features to enable communication between Arrowhead Core Services (Service Registry and Orchestration), Producers, Consumers and FIWARE protocols.

## 1.2 How This Service Is Meant to Be Used

This service is designed to communicate with an NGSI Broker, such as Orion Context Broker [4]. The Translator service provides interfaces and functions to allow Arrowhead-compiant systems to exchange data with FIWARE, as well as allowing FIWARE clients to interact with an Arrowhead Local cloud.

## 1.3 Status of this Document

This document represents the current version of the Translator FIWARE service. Eclipse Arrowhead, being part of an academic and R&D community is constantly evolving to provide more features and increased performance and stability. More features can therefore be added to the Translator FIWARE service in the future. This document must then be updated to incorporate any new features or changes.

# 2 Service Interface

This section lists the *functions* that must be provided by a Translator FIWARE service. Each function represents one feature the Translator FIWARE service can *perform* a task, e.g. list, register and delete entities, request data, etc. In particular, each following subsection names an abstract function, an input type and an output type, in that order. The input type is named inside parentheses, while the output type is preceded by a colon.

All abstract data types named in this section are defined in Section 3.

## 2.1 function PluginGetEntityValue () : Object / Text

This function is called to get the information of a FIWARE entity from an Arrowhead Consumer. The response can be represented using three different Media Types; JSON, JSON+SenML or Plain text.

## 2.2 function FiwareGetIt () : FiwareUrlServices

This function does not have any attributes. Instead it offers the initial API affordances in the form of the links in the JSON body, following the NGSIv2 specification. Instead of using fixed URLs, to keep the FIWARE client decoupled from implementation details.

## 2.3 function FiwareListEntities ( id, type, idPattern, typePattern, q, mq, georel, geometry, coords, limit, offset, attrs, metadata, orderBy, options) : FiwareEntity

Following the standard FIWARE NGSIv2 [2], this function retrieves a list of FIWARE Entities that matches different criteria by id, type, pattern matching (either id or type) and/or those which match a query or geographical query. A given entity has to match all the criteria to be retrieved. The query is also transmitted to Orchestration, and its response is parsed to be compliant to the FIWARE Entity format. The response payload is an array containing one object per matching entity, result of the combination of the responses from the standard FIWARE Broker and Orchestration. Each entity follows the JSON entity representation format. Return list of FIWARE Entities and Arrowhead Providers.

## 2.4 function FiwareCreateEntity (FiwareEntity)

Called to create a new FIWARE Entity on the FIWARE Broker, at the same time, the Translator registers the Entity as a Service Producer on the Service Registry.

## 2.5 function FiwareRetrieveEntity (id, type, attrs, metadata, options) : FiwareEntity

The response is a FiwareEntity identified by the id, type or any other attributes provided on the query. The returned FiwareEntity can be generated directly from the Fiware Broker or can be injected as a result of an Service Registry request.

## 2.6 function FiwareRetrieveEntityAttributes (id, type, attrs, metadata, options) : Object

This function is similar to FiwareRetrieveEntity, but omitting the fields id and type of the FiwareEntity.

## 2.7 function FiwareUpdateAppendEntityAttributes (id, type, attrs, metadata, options, Object)

On this request the FIWARE clients can update or append new attributes to a already existing FiwareEntity at the FIWARE Broker. The entity attributes are updated with the ones in the payload, depending on whether the append operation options are used or not.

- If append is not used: the entity attributes are updated (if they previously exist) or appended (if they do not previously exist) with the ones in the payload.

- If append is used: all the attributes in the payload not previously existing in the entity are appended.

### 2.8 function FiwareRemoveEntity (id, type)

Called to remove an specific FiwareEntity (by id and/or type) from the FIWARE Broker. If the entity is removed successfully, the Translator also unregisters the Service from the Service Registry. As a consecuence of it, the PluginGetEntityValue function will not longer be supported.

### 2.9 function FiwareListEntityTypes (limit, offset, options) : Object

Requested to get the list of all registered Entity Types at the FIWARE Broker. Each Entity Type is defined by type, attrs, and count (number of registered entities of the same type).

### 2.10 function FiwareRetrieveEntityType (type) : Object

Similar to FiwareListEntityTypes, but in this case only one type is requested. This function only returns the attrs, and count.

# 3 Information Model

Here, the main data object model that is used by the Translator service is listed. For additonal information, see FIWARE [1] and NGSI [2].

## 3.1 struct FiwareUrlServices

Below is a summary of the most used data entries in an JSON document.

| Field | Type | Description |
|---|---|---|
| entities_url | String | Entities URL path. |
| types_url | String | Types URL path. |
| subscriptions_url | String | Subscriptions URL path. |
| registrations_url | String | Registrations URL path. |

## 3.2 struct FiwareEntity

Below is a summary of the most used data entries in an JSON document.

| Field | Type | Description |
|---|---|---|
| id | String | Id of the Entity. |
| type | String | Type of the Entity. |
| others | Any | Any other parameters. |

## 3.3 Primitives

The datatypes and structures mentioned throughout this document are assumed to be available to implementations of this service. The concrete interpretations of each of these types and structures must be provided by any IDD document that implements this service.

| Type | Description |
|---|---|
| id | A comma-separated list of elements. Retrieve entities whose ID matches one of the elements in the list.. |
| type | Comma-separated list of elements. Retrieve entities whose type matches one of the elements in the list. |
| idPattern | A correctly formated regular expression. Retrieve entities whose ID matches the regular expression. Incompatible with id. |
| typePattern | A correctly formated regular expression. Retrieve entities whose type matches the regular expression. Incompatible with type. |
| q | A query expression, composed of a list of statements separated by ;. |
| mq | A query expression for attribute metadata, composed of a list of statements separated by ;. |
| georel | Spatial relationship between matching entities and a reference shape. |
| geometry | Geografical area to which the query is restricted. |
| coords | List of latitude-longitude pairs of coordinates separated by ';'. |
| limit | Limits the number of entities to be retrieved. |
| offset | Establishes the offset from where entities are retrieved. |
| attrs | Comma-separated list of attribute names whose data are to be included in the response. The attributes are retrieved in the order specified by this parameter. If this parameter is not included, the attributes are retrieved in arbitrary order. |
| metadata | A list of metadata names to include in the response. |
| orderBy | Criteria for ordering results. |
| options | Options dictionary. |
| URL path | String of the Path of a URL. |
| Any | Any JSON Element (JSON Object or JSON Array). |
| Object | JSON Object. |

# 4 References

[1] “FIWARE Community,” Online, 2019, . [Online]. Available: https://www.fiware.org/

[2] “Next Generation Service Interface - NGSI,” Online, 2019, . [Online]. Available: https://www.opemmobilealliance.org/

[3] J. Delsing, “IoT Automation : Arrowhead Framework,” 2017.

[4] “Open-source FIWARE Context broker,” Online, 2018, . [Online]. Available: https://fiware-orion.readthedocs.io/

# 5 Revision History

## 5.1 Amendments

| No. | Date | Version | Subject of Amendments | Author |
|---|---|---|---|---|
| 1 | 2019-03-27 | 1.0 | Initial | Pablo Puñal Pereira |
| 2 | 2019-05-02 | 1.1 | Models and Interfaces update | Pablo Puñal Pereira |
| 3 | 2020-12-11 | 2.0 | Template Update | Pablo Puñal Pereira |

## 5.2 Quality Assurance

| No. | Date | Version | Approved by |
|---|---|---|---|
| 1 | | | |