

Open APIs
for Open
Minds

- FIWARE Summit, Malaga, 2016

IoT Broker

2016/12/13

Flavio Cirillo, Martin Bauer, Stefan Gessler, Gurkan Solmaz,
Bin Cheng, Ernoe Kovacs

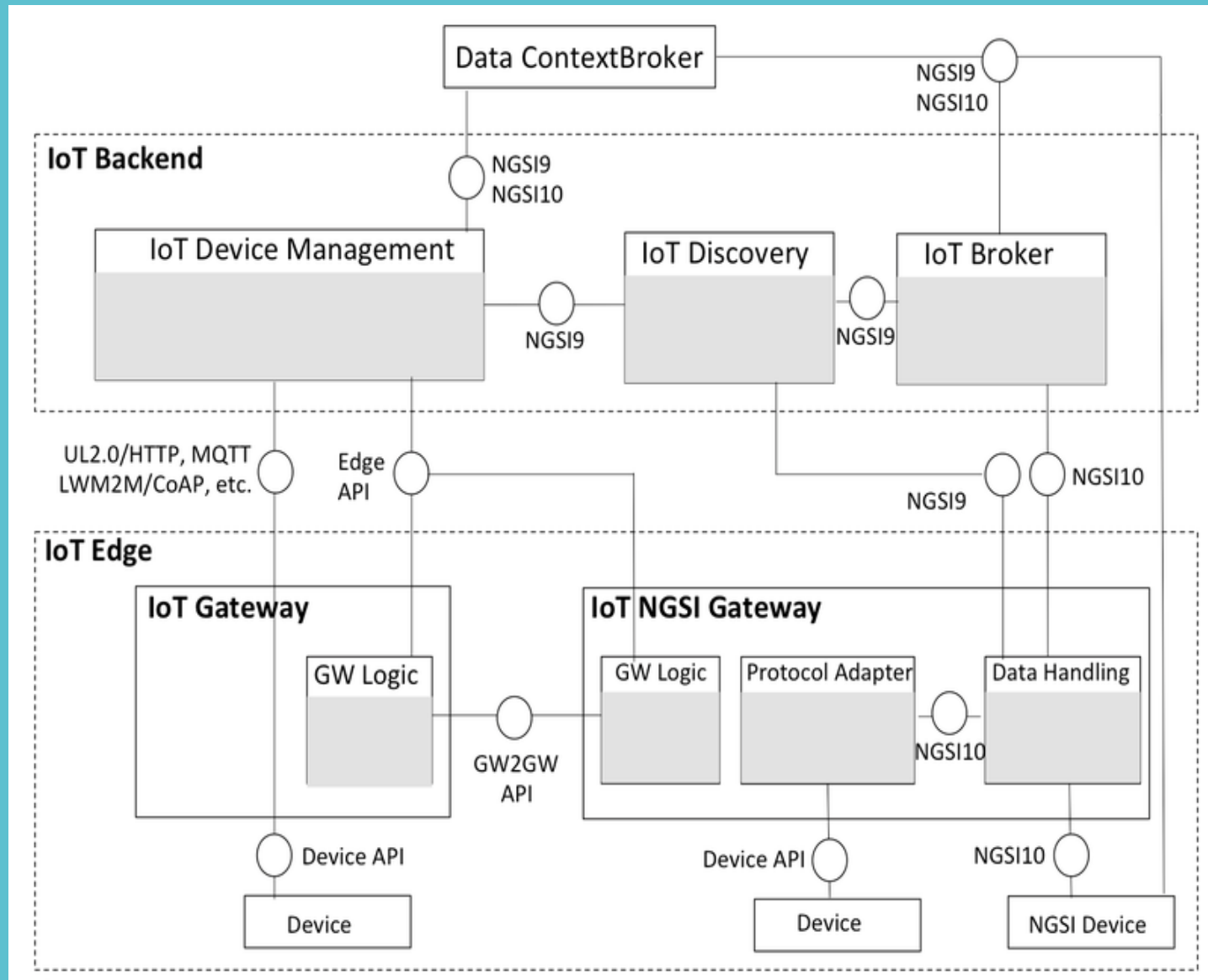
NEC Laboratories Europe Ltd.
Cloud Services and Smart Things
<name>.<surname>@neclab.eu



Table of Contents

- **Introduction**
 - What does IoT Broker do?
- **Advanced Features:**
 - IoT Knowledge Server
 - IoT Broker Federation
 - IoT Broker Scalability
- **Real Scenarios:**
 - Smart City Magnifier
 - FogFlow: Edge Processing
 - Hyper Connected Cloud Service

The FIWARE Enablers for IoT

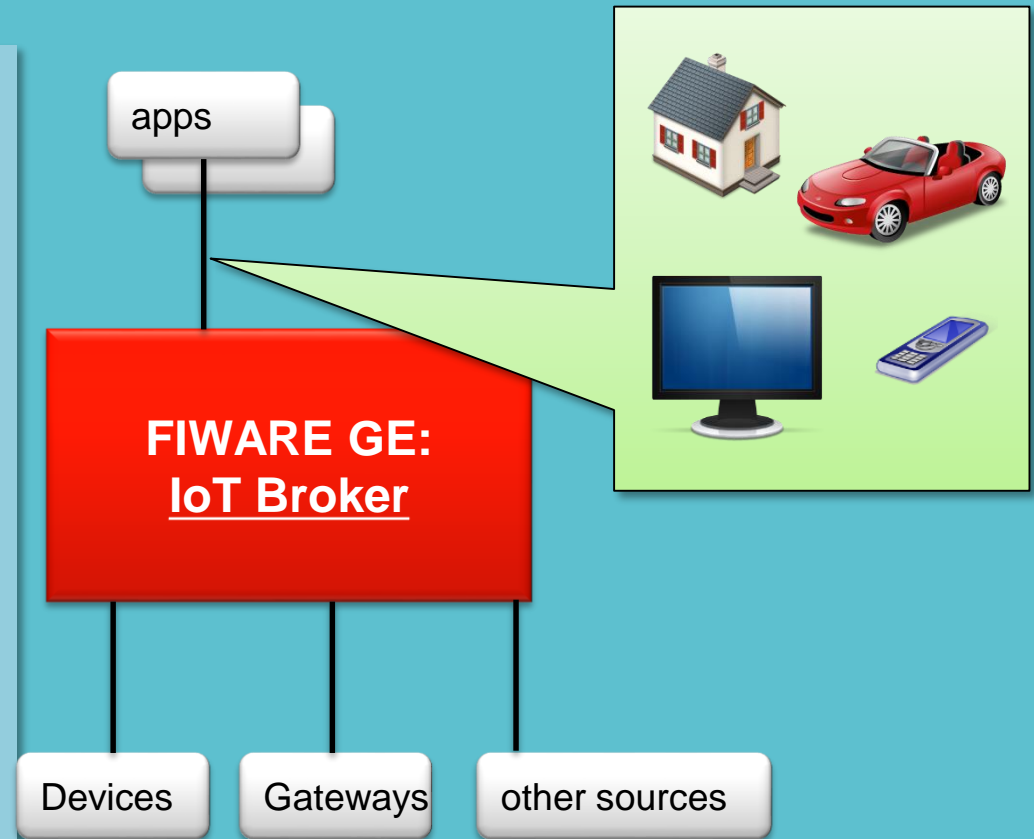


What does the IoT Broker do?

Thing Abstraction: enable applications to interact with things instead of sensors

Thing-Level Interaction:
Organizing information flows:

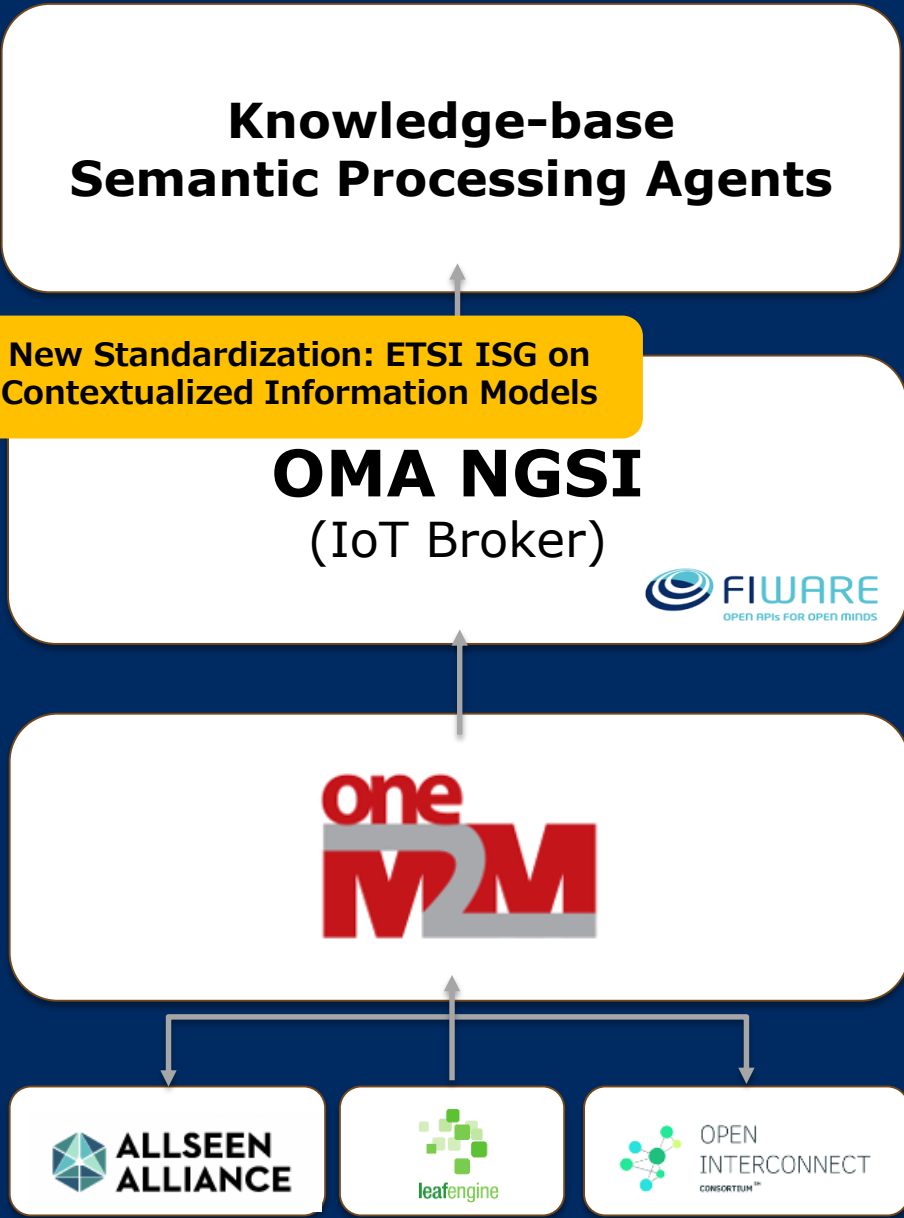
- executing **information queries** on behalf of applications
- **discover** the resources providing the requested information
- **collecting & aggregating** the received information – query and subscribe/notify abstractions



IoT Broker

- decouples applications from underlying IoT device installations
- paradigm adopted: **Subscribe/Notify**
 - Context data fetched directly from repository
 - No need of a centralized repository, but can be added “Plug&Play”
 - Optimized communications with underlying device installations
 - Initialized only when requested from the application
 - Bandwidth communication reduced
 - Scalability ensured in a scenario of billion of devices
- Assemble **lower-level device information** (device-centric access) into **higher-level Thing information** (information-centric access)
 - **Naming:** From Devices (e.g. sensorId) to Things (e.g. Trafalgar Square).
 - **Type & Context:** Close the gap between information-centric applications and device-centric IoT installations
 - **Discovery & Resolution:** IoT applications are agnostic of the device installations
 - **Advanced Features:**
 - Association
 - Entity Composition

Emerging IoT Protocol Stack



Data Integration

- across many systems
- Semantic Representation
- Semantic Mediation

IoT Entities

- Contextualized Information
- Content-based Queries
- Pub / Sub

IoT Integration Layer

- IoT Resources: Black Box Container
- REST-based Access

IoT Development System

- SDK
- OS Integration
- IoT Hardware

Advanced Feature:

- **IoT Knowledge Server**
 - Add **semantic information** into NGSI messages
 - Enhance NGSI messages with **semantic reasoning**
- **IoT Broker Federation**
 - Separate **IoT domains**
 - Improve IoT system integration
- **IoT Broker Scalability**
 - Enhance performances in envisioned scenario of millions of devices in each domain

IoT Knowledge Server



IoT Knowledge Server: Overview

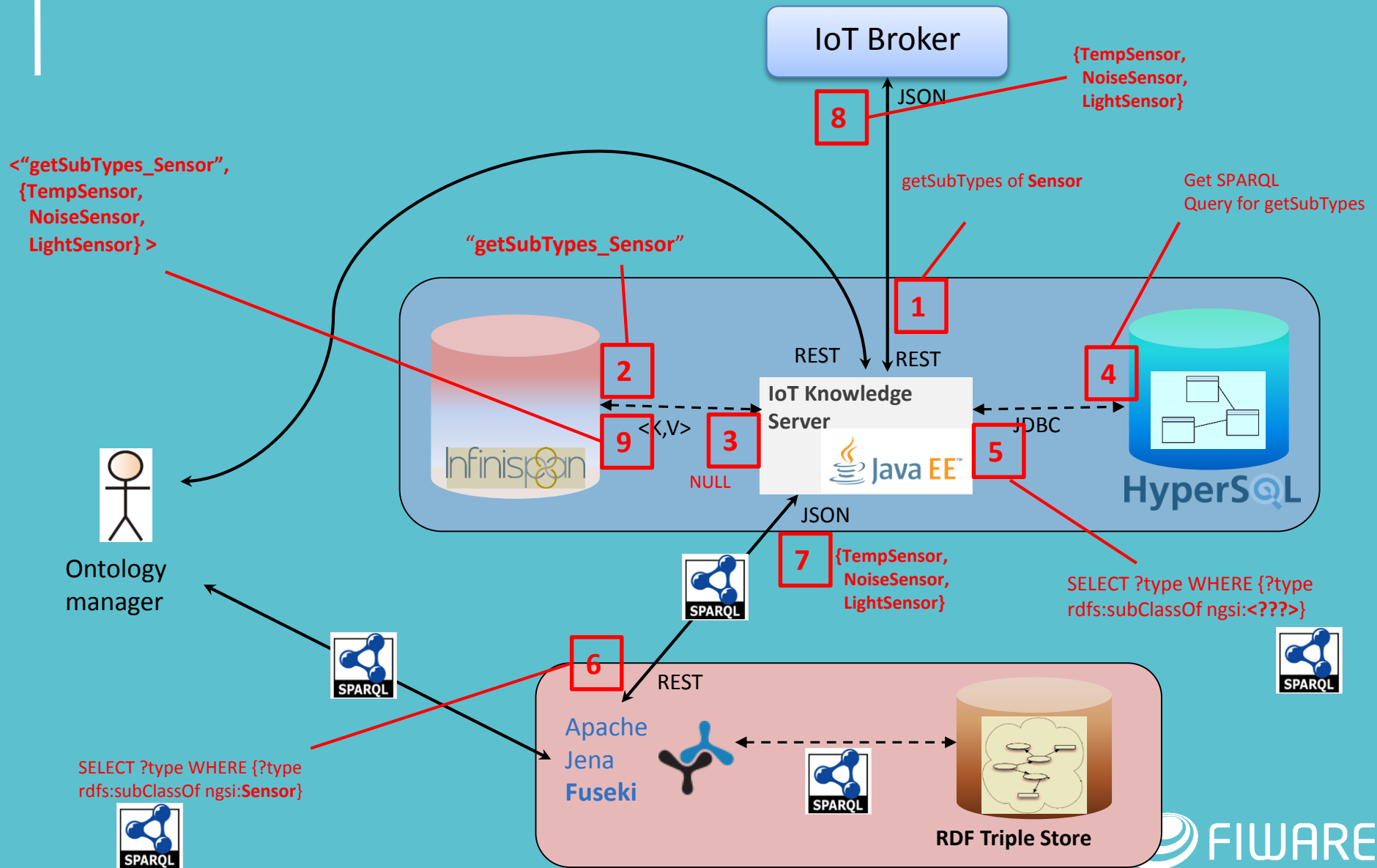
- **IoT Knowledge Server:** A standalone component created for serving semantic information
- Purpose: serving IoT Broker with triple-store datasets of ***semantic ontologies*** (e.g., NGSI/SmartSantander ontology)
- Record and Explore Information Structure contained in the real-world data
 - *“get sub types of an entity type”*
- Interfaces: ***REST API*** and Subscribe/Notify in JSON format
- IoT Knowledge Server is composed of two components (web servers) and two databases along with the servers



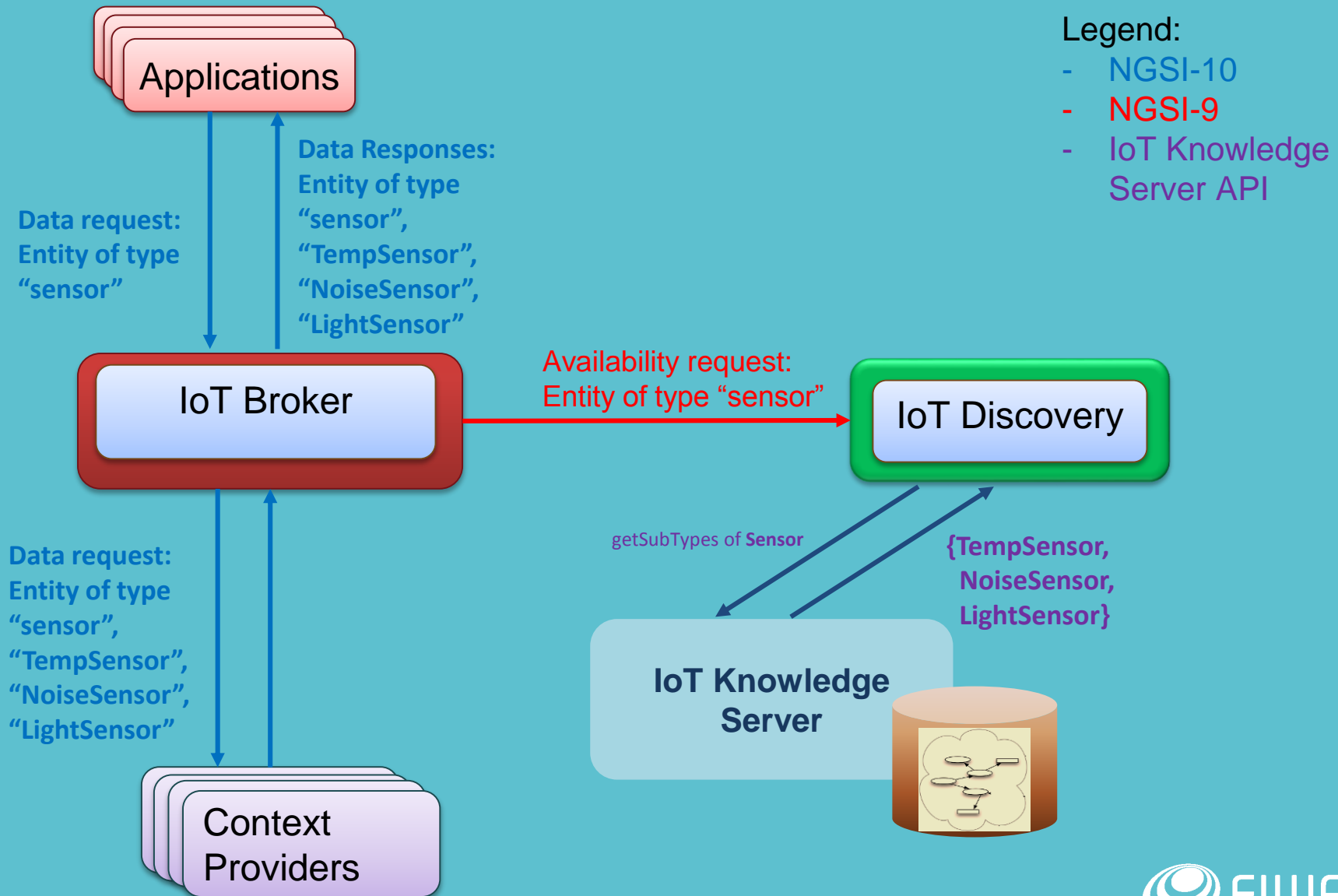
Functionalities

- Pre-Defined Queries
 - HTTP requests for *getSubTypes*, *getSuperTypes*, *getAttributes*, *getAllSubTypes*, *getAllSuperTypes*
- Add new queries
 - New queries with one or zero variables (e.g. Entity Type) can be added to a file and we can start using as a new functionality (other than the 5 above)
- Register new queries
 - Adding new queries by HTTP request on the fly (without restarting the server)
- Forward SPARQL queries
 - To provide single point of contact even for direct SPARQL queries along with the high level ones (*getSubTypes*)
- Subscribe functionality
 - Subscribing to queries and regular (fixed time) updates on change to the subscribers by the IoTKnowledgeServer.
- Caching mechanism
 - Caching mechanisms for fast respond (without asking SPARQL server)
 - Both for Queries and for Subscriptions

An example query



IoT Knowledge Server: Example



IoT Broker Federation



IoT Broker Federation

- Smart Cities are dominated by **federated information** from different agencies
- An IoT platform is **responsible for a single IoT domain**
 - Separate IoT data in different domains
 - Full power on the produced data to the IoT domain administrator, e.g. for privacy purpose
- **Selective communication** to a specific domain
 - Selected by IoT domain name
 - Selected by entity name
 - Selected by attribute type provided
 - Selected by scope, e.g. geographic scope
 - Mixture of the above.



Federation: hierarchical

IoT Platform Hierarchy

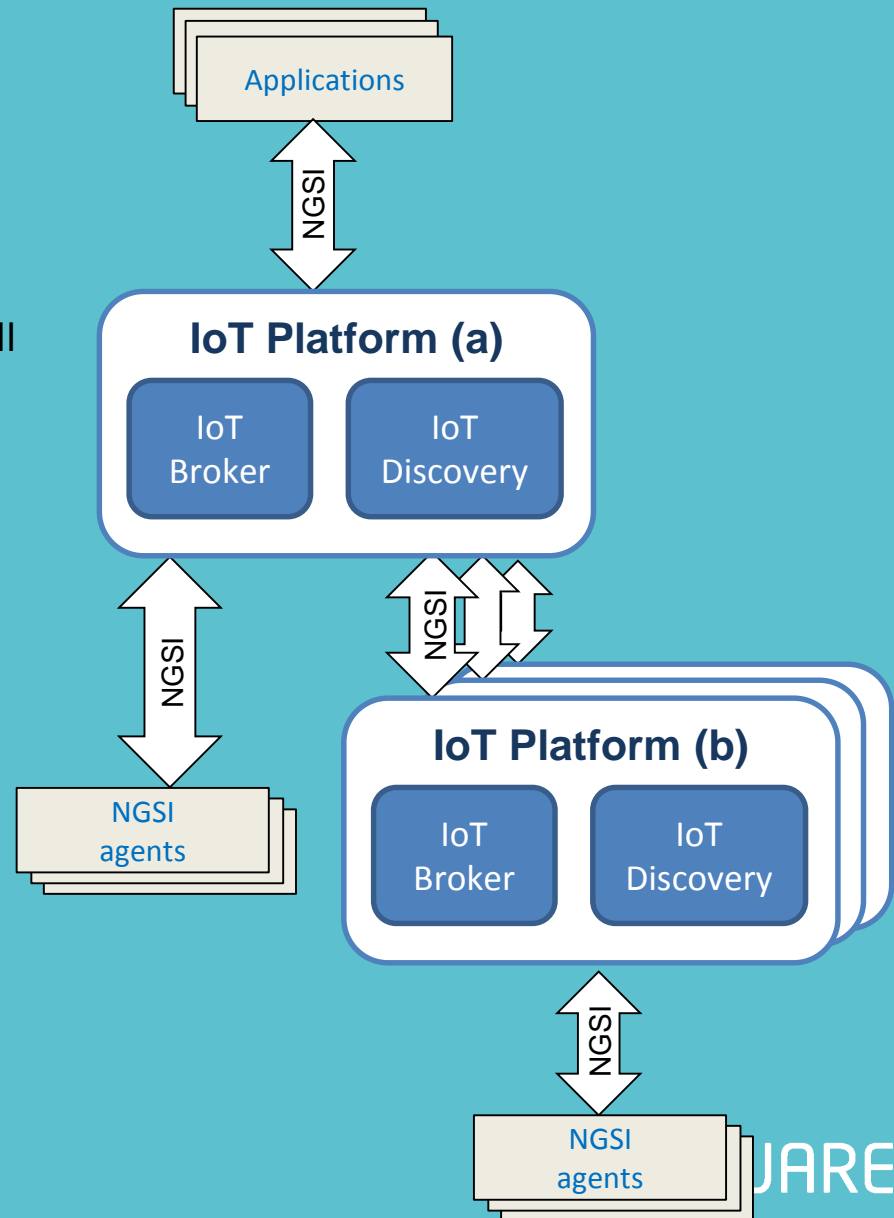
- Two type of platform
 - Subordinated IoT Platform: responsible for its IoT domain; subordinated to Platform
 - Top IoT Platform: responsible of its own domain of NGSI devices; contact point for all subordinated domains
- Two IoT domains manage their data in **separate repositories**
- **Common communication language** based on standard NGSI protocol
- **Mechanism of Subscribe Notify** for accessing the data

Feature: broadcasting

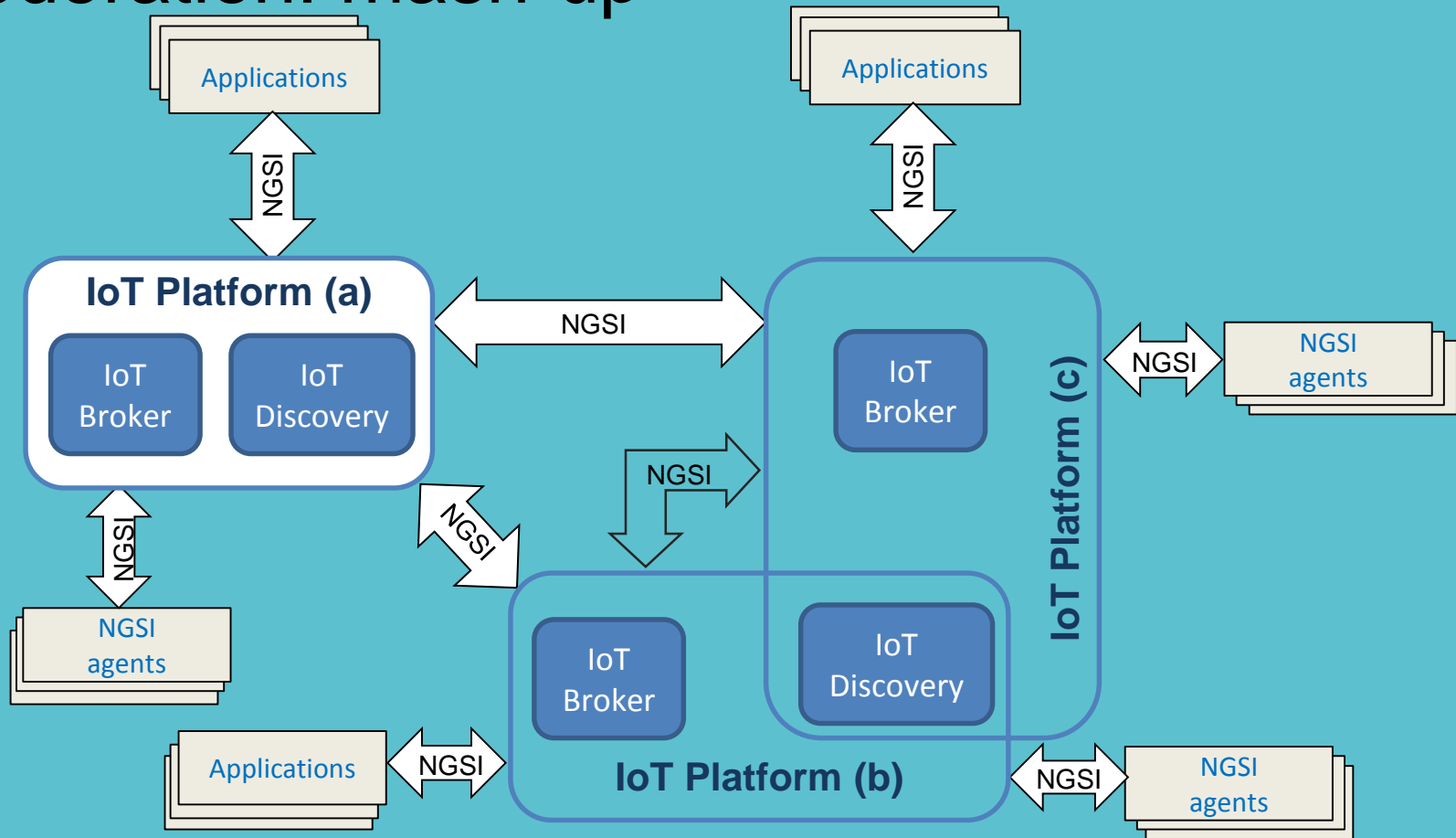
- Top IoT Platform dispatches query/subscription to subordinated IoT Platform

Feature: selective communication

- Possibility to query/subscribe only to a specific subordinated IoT Platform



Federation: mash-up



IoT Platform Mesh

- Each platform is a **peer**
- Each peer is responsible of its own domain
- Applications requesting a peer will get data coming from other peer transparently

Feature: **broadcasting**

- Peer broadcast request to all known peer

Feature: **selective communication**

- Possibility to query/subscribe only to a specific known peer

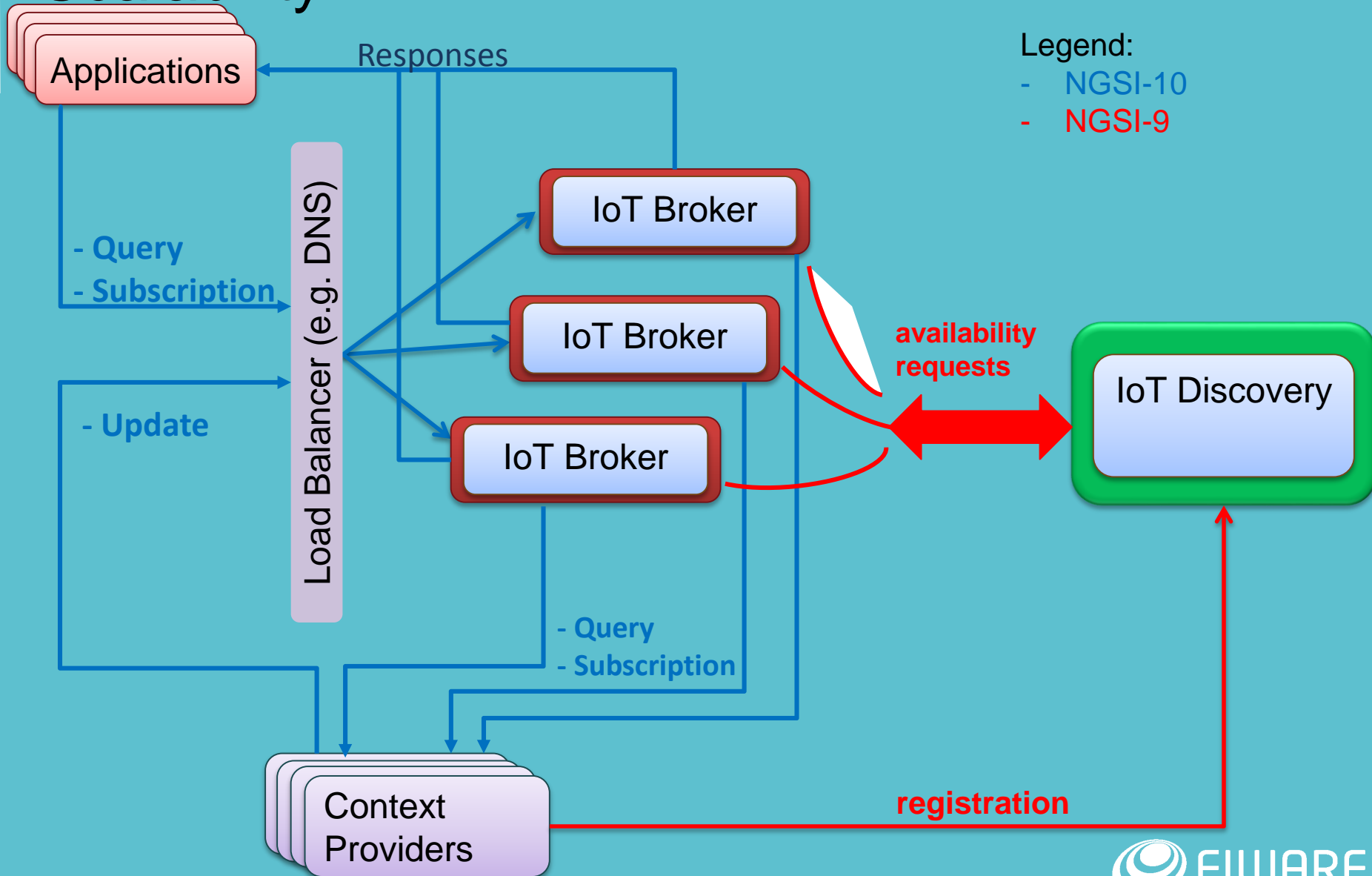
Feature: **loop detection**

- A loop detection feature avoid loop in the topology

IoT Broker Scalability



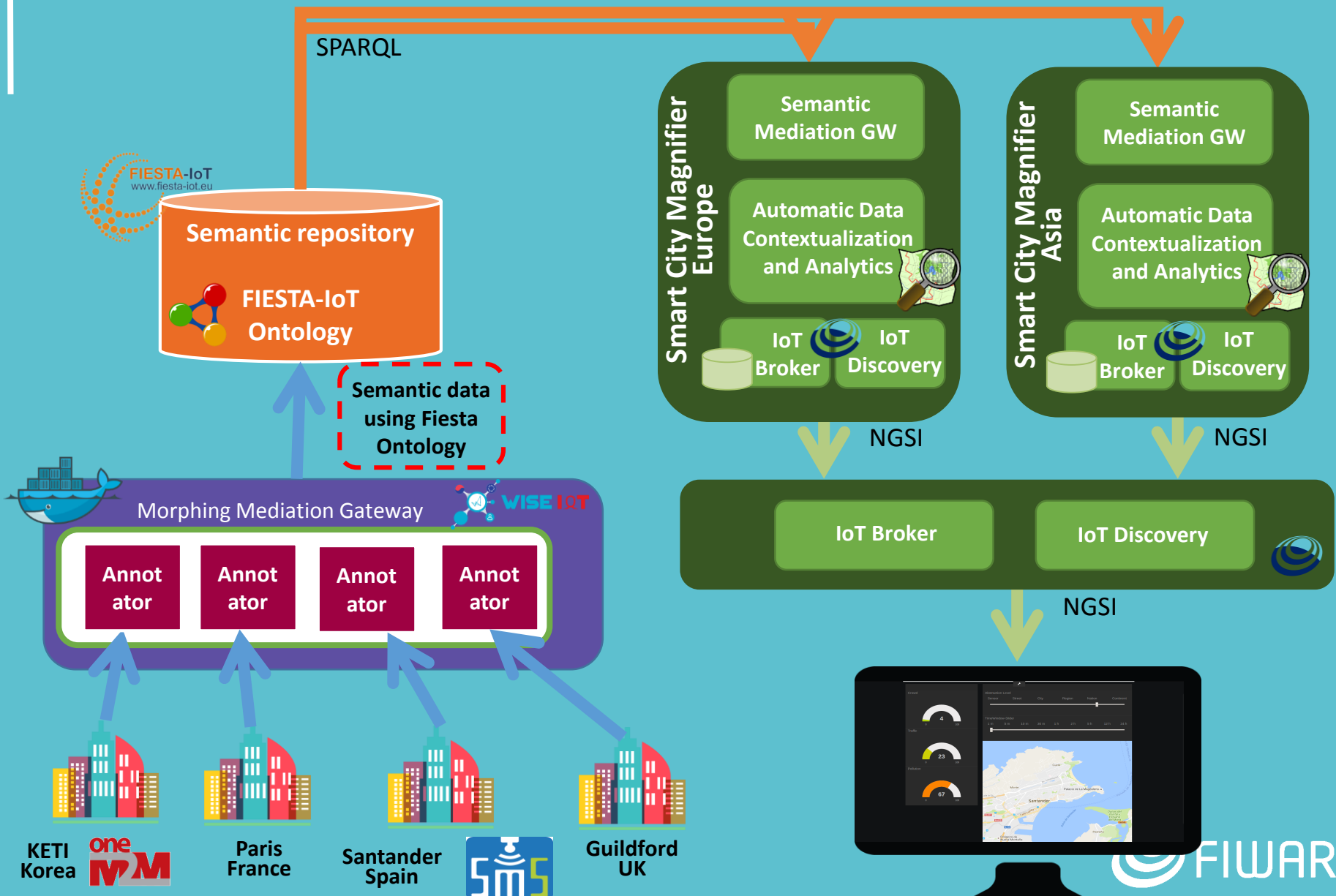
Scalability



Scenarios

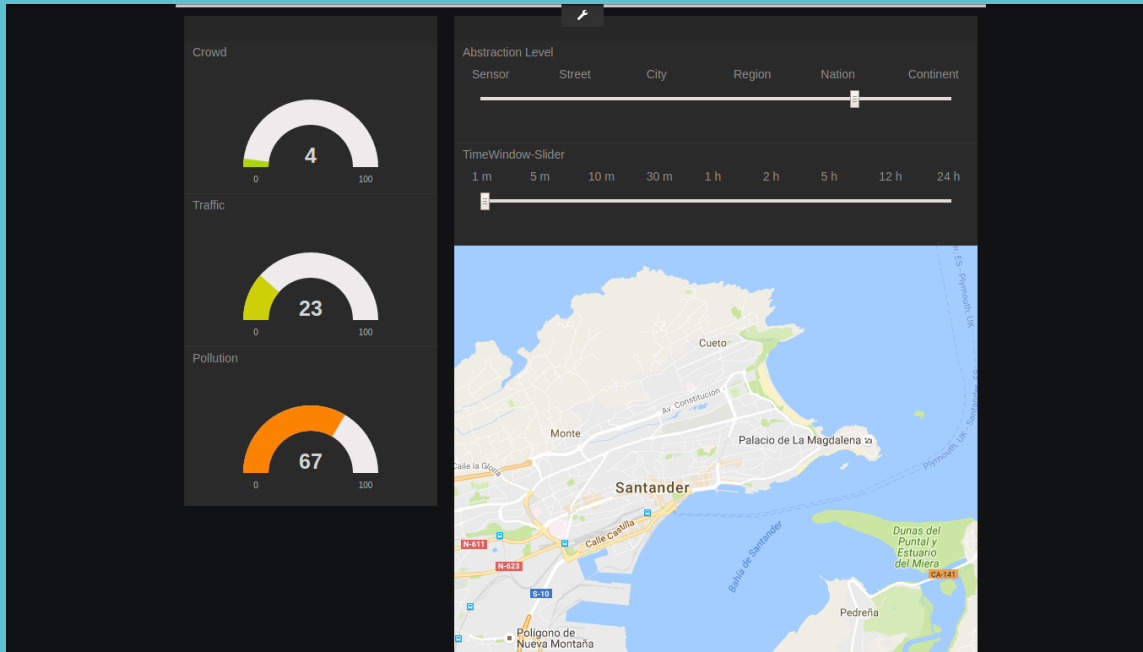


Scenario 1: Smart City Magnifier



Scenario 1: Smart City Magnifier

Contextualized data visualization



1

3 dimensions of zooming:

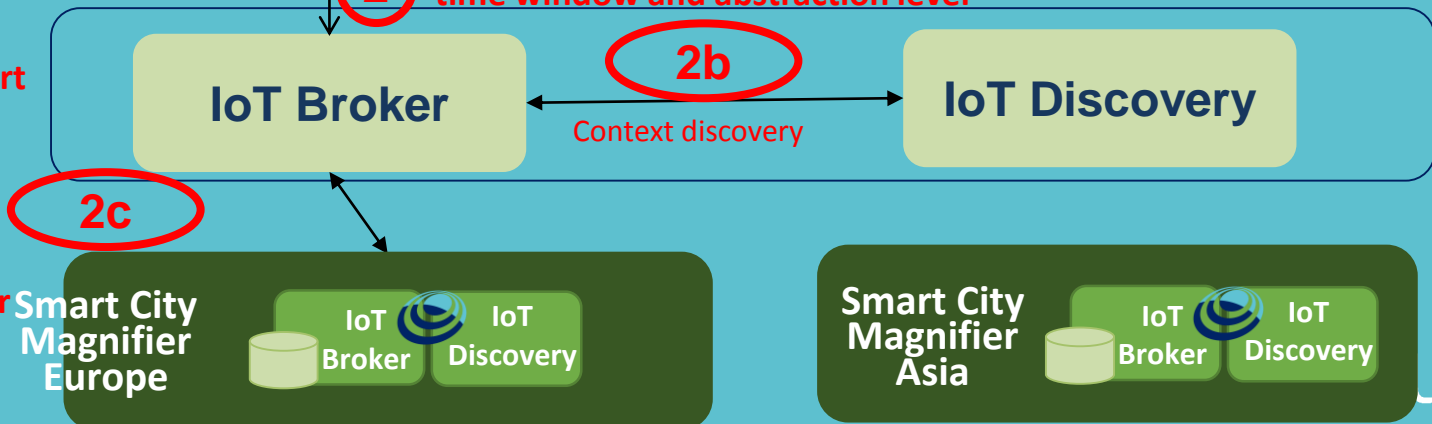
- Abstraction
- Time
- Geographic

3

Data visualization

2

Data query specifying geographic scope, time window and abstraction level



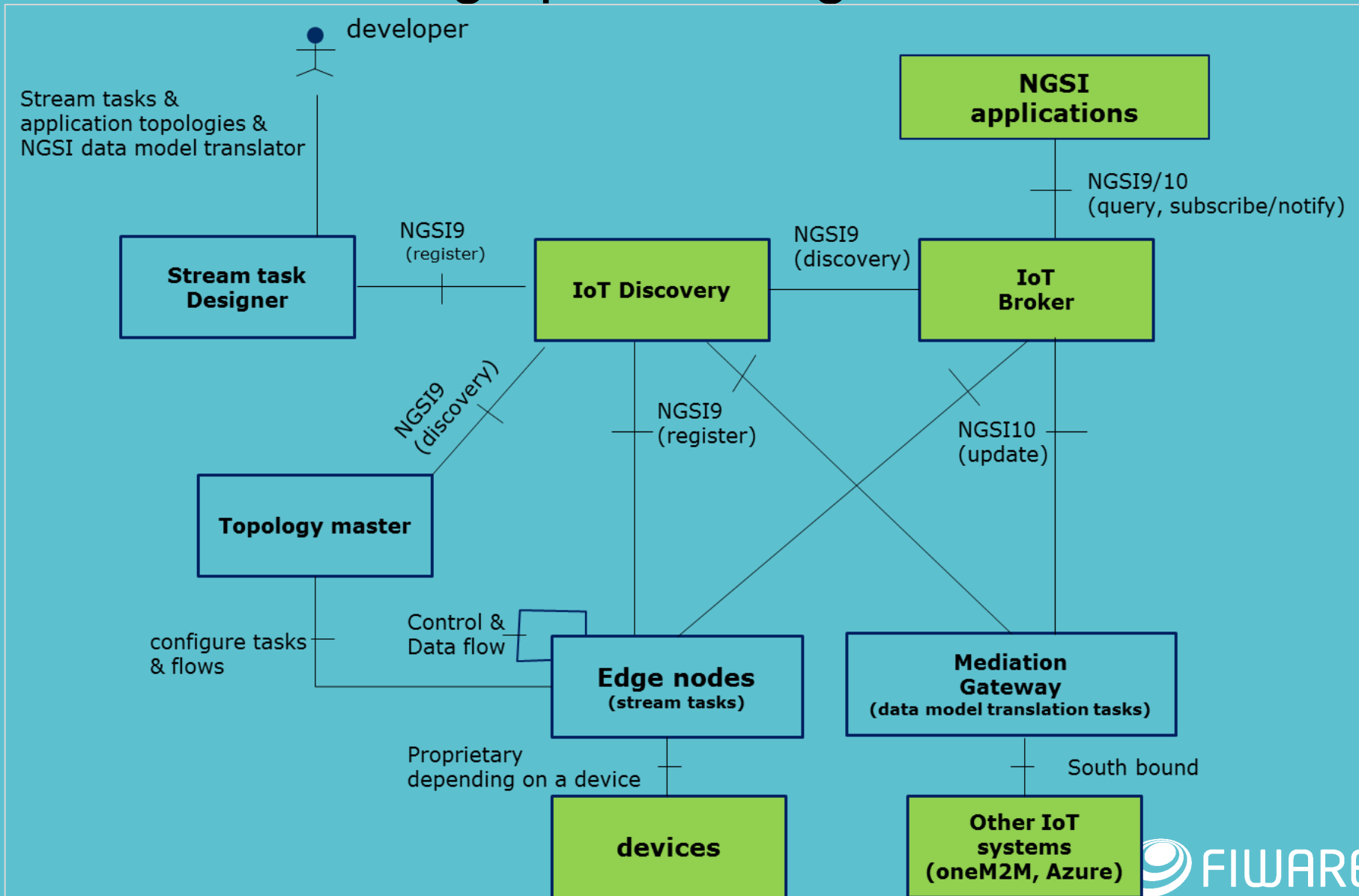
Data request forwarded only to the European instance of the Smart City Magnifier because the data requested is geographically scoped in Santander (Spain)

2c

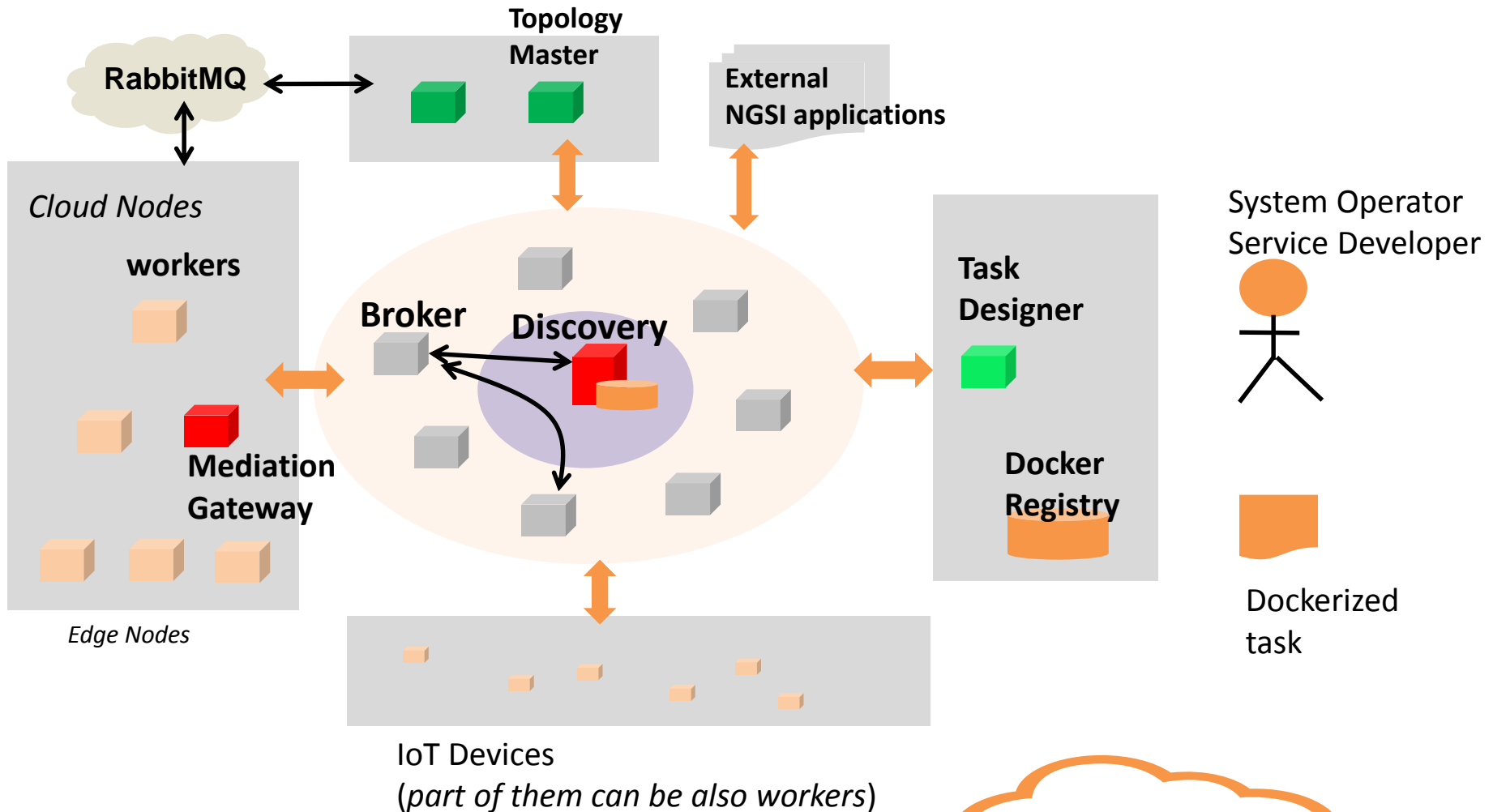
2b

WARE

Scenario 2: Edge processing

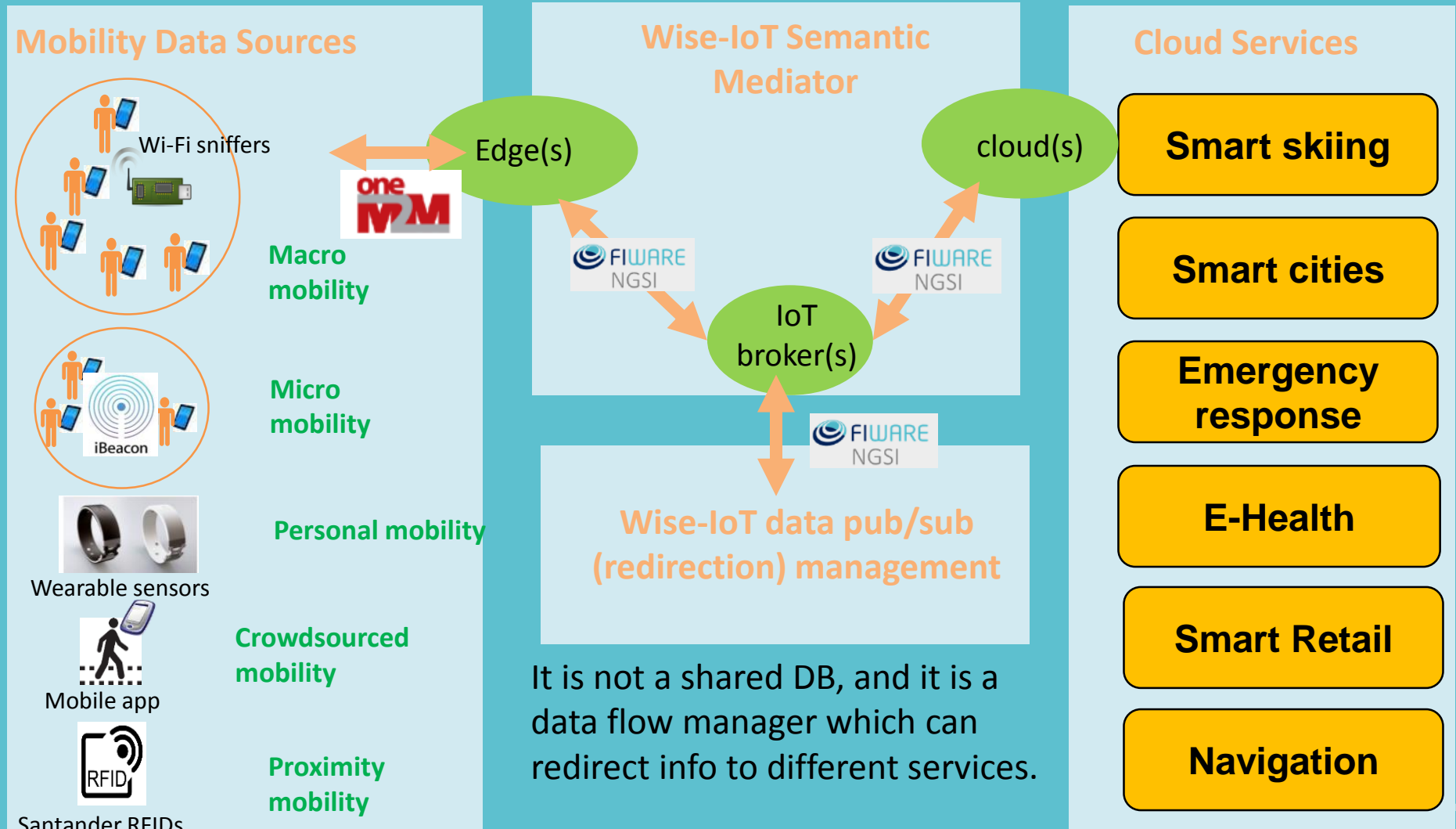


Scenario 2: Architecture of FogFlow



Scenario 3 (Wise-IoT): A Vision Architecture of Hyper-connected Cloud Services

Crosscutting data reuse between hyper-connected cloud



 **Orchestrating** a brighter world

NEC