Select for Cities

Grant Agreement № 688196

City Enabler Hackathon in Helsinki
14 March 2019
# Table of contents

1. What is City Enabler ........................................................................................................3
2. City Enabler Architecture .............................................................................................4
3. Data Collection layer ......................................................................................................5
   3.1. Data source Discovery Engine (DSDE) .....................................................................5
   3.2. Data Workspace .........................................................................................................5
   3.3. IoT Device Manager (DEMA) ..................................................................................8
4. Data Integration and Harmonization Layer .....................................................................8
   4.1. Orion Context Broker ..............................................................................................8
   4.2. Data Mashup Editor (DME) .....................................................................................9
   4.3. Mockup Editor ...........................................................................................................9
   4.4. Perseo Complex Event Processor ............................................................................10
5. Data Visualization layer ................................................................................................10
   5.1. Dashboard Manager .................................................................................................11
   5.2. Front End ..................................................................................................................12
6. Open Innovation layer ....................................................................................................12
7. Security Layer ................................................................................................................12
8. Logging Layer ................................................................................................................12
9. Performance monitoring layer ........................................................................................13
10. API documentation ........................................................................................................13
11. YouTube Channels ........................................................................................................13
1. What is City Enabler

As a consequence of the digital transformation processes Digital Cities have increasingly started to publicly release city-related data on the web in a format that is not processable.

Data is the drivers of tomorrow’s smart cities. Information needs to be easily accessible and usable, creating a path to sustainable value creation.

City Enabler is the Digital transformation ecosystem platform, powered by FIWARE, supporting the entire value chain of data, from the discovery to the analysis, harmonization and visualization. The City Enabler provides cities with a single point to find, access and gather all this data. It looks for potential unknown data-sources available on the web, real-time data coming from sensors in the city, open data portals, legacy systems, data not apparently processable and makes all these information systems speak with one another, evaluating the quality of metadata and avoiding duplication. All the data sources are transformed into meaningful information accessible through user interface and standard APIs.

Thanks to graphical and map based innovative tools, City managers, city providers and developers can design and create reusable interactive dashboards analysing and visualizing data at right-time, and harmonise real time information to adjust, for instance, public transport supply.

Dashboards can be organised in domains like environment, mobility or manufacturing and be used by businesses and developers to create new interoperable services and apps that can be portable and replicable in different cities.

Hence, the City Enabler embeds the following functionalities:

- **Discover data sources from the web**
- **Data info collection and quality assessment**: Detection of data sources from the web and support in the creation of the “Single Point of Knowledge (SPOK)”
- **Data Integration and Harmonization**: Possibility to graphically integrate data coming from different data sources and wrap any kind of data structure in NGSI LD FIWARE Data Models
- **Analyze and render data through user friendly dashboards**: Assisted creation apps and dashboards relying on data present in the SPOK, that are reusable among different customers
2. City Enabler Architecture

Figure 1 shows the high-level architecture of the City Enabler platform. It’s a layered architecture built up by 7 layers. The following sections describe each layer in more details.
3. Data Collection layer

This layer accounts for the data collection phase, through the platform components within this layer the platform can be fed with raw data coming from several heterogeneous and scattered data sources.

The tools that are directly in charge of the collection of the data are the Data source Discovery Engine, Data Workspace and IoT Device Manager.

3.1. Data source Discovery Engine (DSDE)

Component not required to be used for this Challenge.

3.2. Data Workspace

An introductory video is available: https://www.youtube.com/watch?v=NoEvz2GHCQg

Further information is available in the file DE-Handbook-Data-Workspace.pdf

The Data Workspace is a java-based tool that allows to collect metadata about different kinds of data sources. It plays the role of unique layer to access data from registered heterogeneous data sources. It natively supports CKAN, DKAN, SOCRATA and
OpenDataSoft open data portals but whichever kind of data source can be federated thanks to the so-called **microproxy approach**. Such an approach allows the federation of legacy systems and custom, non-standard, open data portals. Moreover, the with the Data Workspace the user can federate potential data sources found by the Data source Discovery Engine.

**Example of usage of the Data Workspace**

After the developer logs in in the Data Workspace, she/he registers the data sources through a comprehensive wizard. As an example, the page to register a REST API data source is shown in Figure 3.

Once all the data sources are registered, the user can see the list of federated data sources (see Figure 4).
Furthermore, the user can import in the Data Workspace the data sources found by the DSDE. As an example, the registration as data source of the one platform found by the DSDE is shown in Figure 5.

![Figure 5 - Registration of a data source found by the DSDE](image)

After the data sources are registered, the user can perform search on data sources both through user interface and APIs.

![Figure 6 - Data Workspace home page](image)

Moreover, the quality of all the datasets metadata is evaluated in background following an evaluation algorithm based on the MELODA framework. Such evaluations are
clearly made available both in the GUI and in the REST APIs exposed by the CDW. This way all the stakeholders can have an overview of the quality of the data available in the city, and the city official/expert can have awareness about the usefulness of data provided by the registered data sources. An example is shown in Figure 7.

Figure 7 – Data quality measurement of a dataset

3.3. IoT Device Manager (DEMA)

Component not required to be used for this Challenge.

4. Data Integration and Harmonization Layer

This is the second logical layer in the CE stack. It gets the data from the underlying layer and, thanks to visual and easy-to-use tools, allows to mix up the data to create new knowledge and harmonize the results to make them easy to understand.

4.1. Orion Context Broker

One of the component that belongs to this layer is the Orion Context Broker. It’s part of the FIWARE ecosystem and is the component that brokers the data along the platform. Orion Context Broker is one of the most important GEri of the FIWARE ecosystem. It provides context brokerage within the platform. It interacts with: 1) the IoT Device Manager, for the management of the Devices registered on the platform; 2) the Mashup Editor, for the exchange of the raw and harmonized data; 3) the Perseo Complex Event Processor, for the definition of events and rules; 4) the Dashboard Manager, for the presentation of the real-time data within a dashboard. The Context Broker is a component of the platform where the developers can hook in via API: it
allows the developer to perform subscriptions on the data he is interested in and to obtain real-time updates on every measure modification.

### 4.2. **Data Mashup Editor (DME)**

An introductory video is available: [https://www.youtube.com/watch?v=eS3jm7mFCzo](https://www.youtube.com/watch?v=eS3jm7mFCzo)

Further information is available here: DE-Handbook-Data-Mashup-Editor.pdf

The most important tool of this layer is the **Data Mashup Editor (DME)**, a tool that allows to easily and visually mix the data coming from different services and data sources federated in the Data Workspace. Moreover, it is able to subscribe to specific context of the Orion Context Broker (see next bullet point) in order to obtain real time data to harmonize accordingly with the FIWARE Data model. The resulting harmonized entities can be published back to the Orion Context Broker in order to be available to the rest of the platform. The Mashup editor is one of the components of the platform where the developers can hook in: this tool, indeed, provides a fast way to implement atomic business logic on data and expose it as a black box through REST interface. Finally, the resulting service can be published to the CE marketplace, to be made available for the entire community. Hence, the CDME allows to visually import the data from the CDW or from Orion CB and to mix them in order to create new knowledge. This new information might be serialized in a schema compliant with the **FIWARE datamodels**¹: standard de-facto for the smart cities all around the world. These new data are then published back to the Context Broker ready to be used from all other platform components.

As an example, the DME is used to harmonize the data coming from an air quality sensor by using the **AirQualityObserved** FIWARE Datamodel and the weather data (e.g. from a weather station) by using the **WeatherForecast** FIWARE Datamodel. An example, the mapping between the data structure sent an Air quality sensor with NGSI LD FIWARE Data models is shown in Figure 8.

![Harmonization of Air quality VMM sensor data in FIWARE Data models](image)

**Figure 8** - Harmonization of Air quality VMM sensor data in FIWARE Data models

### 4.3. **Mockup Editor**

¹ [https://www.fiware.org/developers/data-models/](https://www.fiware.org/developers/data-models/)
An introductory video is available: https://www.youtube.com/watch?v=eS3jm7mFCzo

The twin tool of the Mashup Editor is the **Mockup Editor**: a tool within the Data Integration and Harmonization Layer that allows to visually create applications, both mobile and web, relying on data produced with the Mashup Editor. It strictly interacts with the Mashup Editor that provides the under-the-hood business logic. The Mockup editor is another component of the platform where the developers can hook in: this tool, indeed, provides a fast way to create applications ready to be publicly released.

The mockup editor includes a comprehensive palette with several elements such as forms, tables, maps, charts, and others. The developer can then link every single element of the user interface with mashups (i.e. composition of different data coming from different data sources) and other data sources federated in the Data Workspace. The created mockup projects can be then exported as web applications and/or android apps.

An example of mockup creation is shown in Figure 9.

![Mockup Editor](image)

**Figure 9 - Creation of the user interface of an app exploiting the data exposed by the City Enabler**

### 4.4. **Perseo Complex Event Processor**

Component not required to be used for this Challenge.

### 5. **Data Visualization layer**

This layer is composed by two components: **the Dashboard Manager** and **the Front End**.
5.1. Dashboard Manager

An introductory video is available: https://www.youtube.com/watch?v=H01HTUdzgLs&t=241s

Further information is available here: DE-Handbook-Dashboard-Manager.pdf

The Dashboard Manager is the tool that allows the fast and easy creation of reusable dashboards starting from the data previously collected, integrated and harmonized. On top of the data, the Dashboard Manager allows to:

- define KPIs through the definition of mathematical formulas;
- integrate Machine Learning software to provide valuable insights on data;
- show the data on thematic dashboard that might help the City Managers in decision making.

This tool enables the developer to build advanced dashboard that rely on the data collected in the platform following a self-service approach. Once the data are collected in the Data Workspace they can be exploited in the Dashboard Manager to create dashboards with several types of contents: 1) KPIs that can be defined exploiting a specific editor that helps the developer to define the mathematical formula to transform the raw data into a meaningful indicator; 2) Maps that can show relevant geo localized data. These data can be clustered, updated in real time or exploited to perform what-if analysis; 3) Charts with the historical evolution of data or the current value updated real time; 4) HTML content properly populated with data coming from the configured data sources.

For each of the content above, the Dashboard Manager provides specific widget that make the dashboard building process easy and scalable, thanks to developer friendly wizards.

Example of usage of the Dashboard Manager

In order to build one environmental dashboard, the user will use the data coming from the following data sources:

- BigBelly (smart trash can) real time data, with the current filling Level of the BigBelly waste bins;
- Open data providing with the near real time data about the filling Level of other waste bins in the city.
- Paper bins position data
- Glass bins position data

All these data have previously been harmonized through the DME into the WasteContainer and WasteContainerIsle datamodels. The final dashboard will result in a big map showing all the containers in the city, distinguished by color: gray for paper, blue for glass, green for BigBelly containers and red for containers information provided in Open Data. The result is shown in Figure 10.
By clicking on the single container, the user can view the current filling level, the cargo weight, the type and many other information. This will make easier the decision-making process about the garbage collection planning. All the dashboards, working with data harmonized in NGSI can be imported in the Front End.

5.2. **Front End**

Component not required to be used for this Challenge.

6. **Open Innovation layer**

Layer not required to be used for this Challenge.

7. **Security Layer**

Layer not required to be used for this Challenge.

8. **Logging Layer**

Layer not required to be used for this Challenge.
9. Performance monitoring layer

Layer not required to be used for this Challenge.

10. API documentation

Specific documentation about the APIs available for the different components of the CE are available at: http://digitalenabler.eng.it/SwaggerUI/

11. YouTube Channels

Several videos can be found in the following YouTube channels:
- https://www.youtube.com/channel/UCnAvfnEJC5sH6o1Dto4njw
- https://www.youtube.com/channel/UCqGVlcM7cS0FVG_qlZ3xuRg/feature