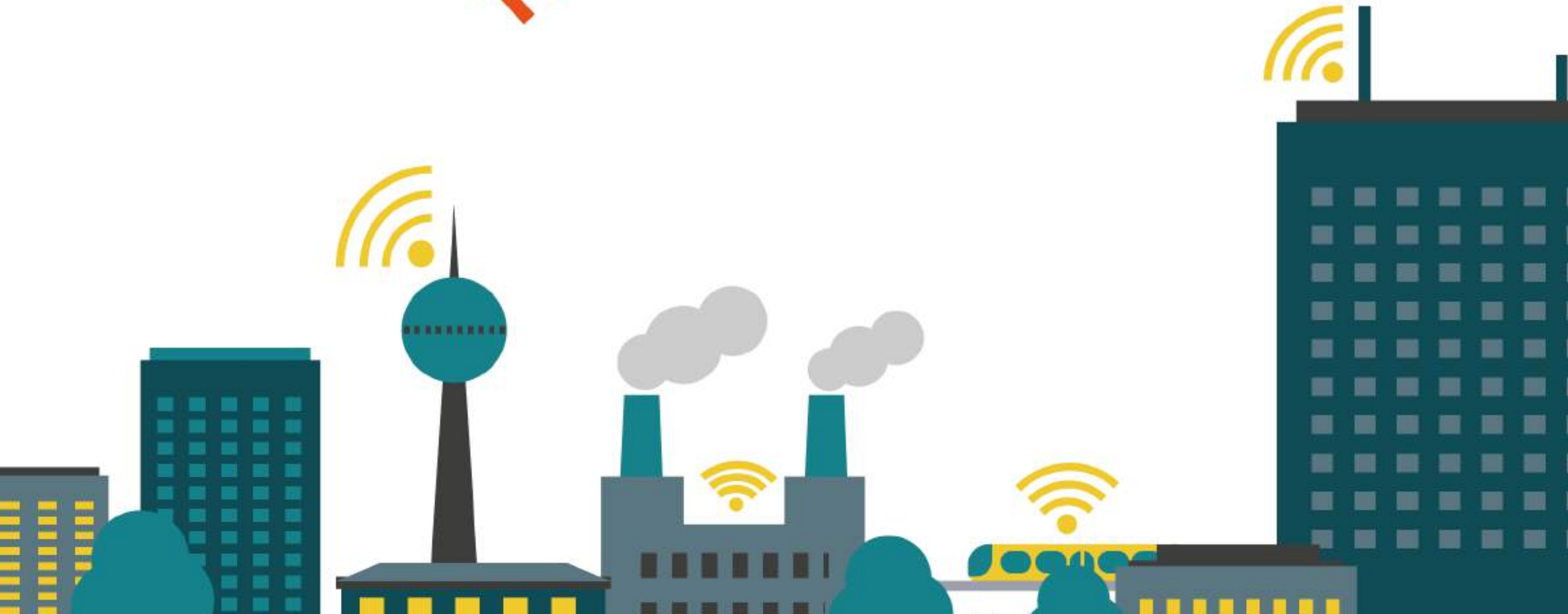
ORCHESTRA *CITIES*

BY MARTEL INNOVATE

AN OPEN PLATFORM FOR CITIES TO DEVELOP AND SHARE SMART CITY SERVICES



ORCHESTRACITIES.COM

PRODUCT SHEET

FEATURES	Security Management Device Management Data Management Dashboard Management Data Integration Management
IOT STANDARDS	COAP, MQTT, AMQP, LORAWAN
SECURITY STANDARDS	OAUTH 2.0, OIDC, SAML, KERBEROS, LDAP, X.509
SUPPORTED CLOUDS	AWS, GCE, AZURE, OpenStack, VWMARE vCloud

TABLE OF CONTENTS

1
CONCEPT

“Technology is nothing. What's important is that you have a faith in people, that they're basically good and smart, and if you give them tools, they'll do wonderful things with them.”

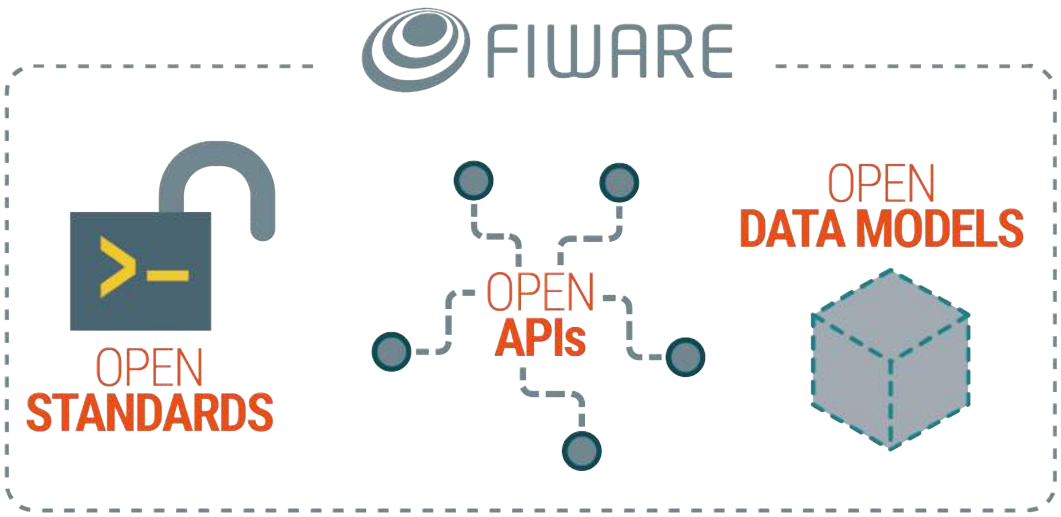
Steve Jobs

WHY ORCHESTRA CITIES?

The Smart City solutions market is mostly driven by large players that rely on proprietary technologies. Recently, a wave of solutions adopting an Open approach has grown to impact the Smart City market.

The forerunner technology leading this wave is FIWARE, the most mature Open Source framework available today dealing with requirements for Smart Cities. The Orchestra Cities concept takes on FIWARE principles and strives to push them further.

The concept of Openness informs the architecture and features of FIWARE and Orchestra Cities in multiple ways, as shown below.

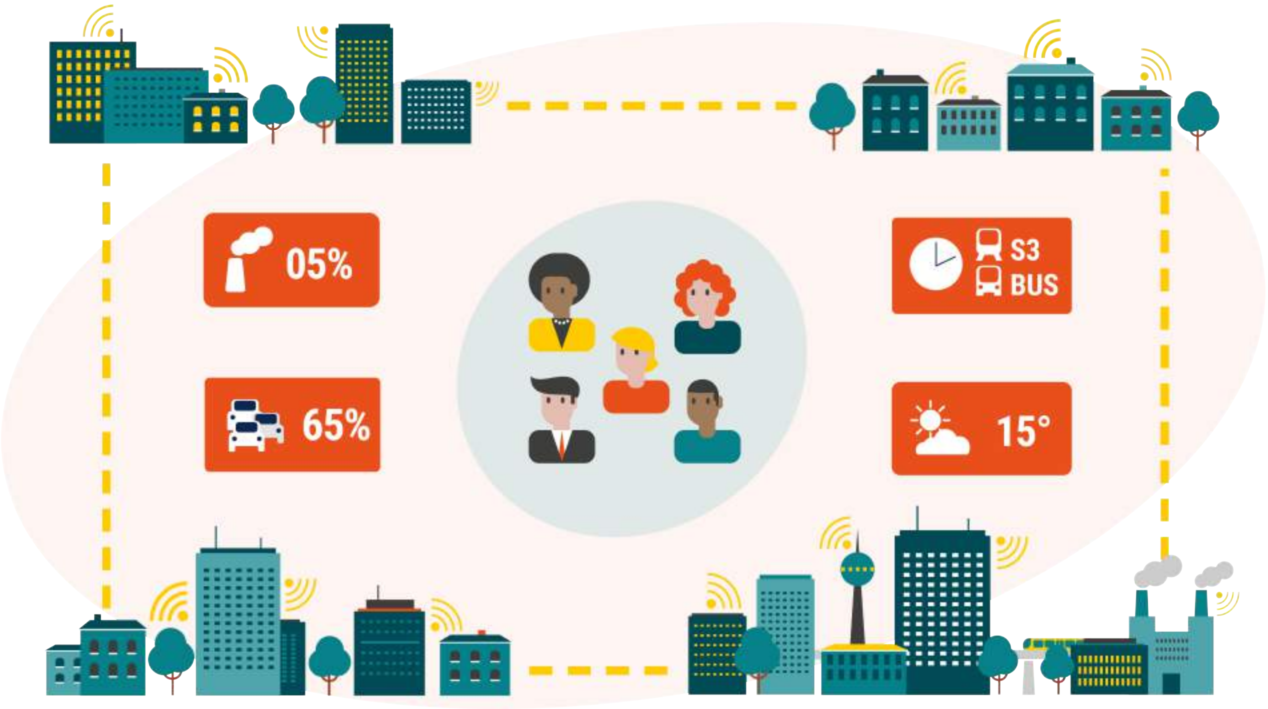


Orchestra Cities embraces the triple openness above and aims to extend them to enable City-to-City collaboration and Citizens-to-City collaboration.

Specifically, what does it mean? Orchestra Cities aims at building a collaborative space for shaping a sustainable and participatory future for our cities, where:

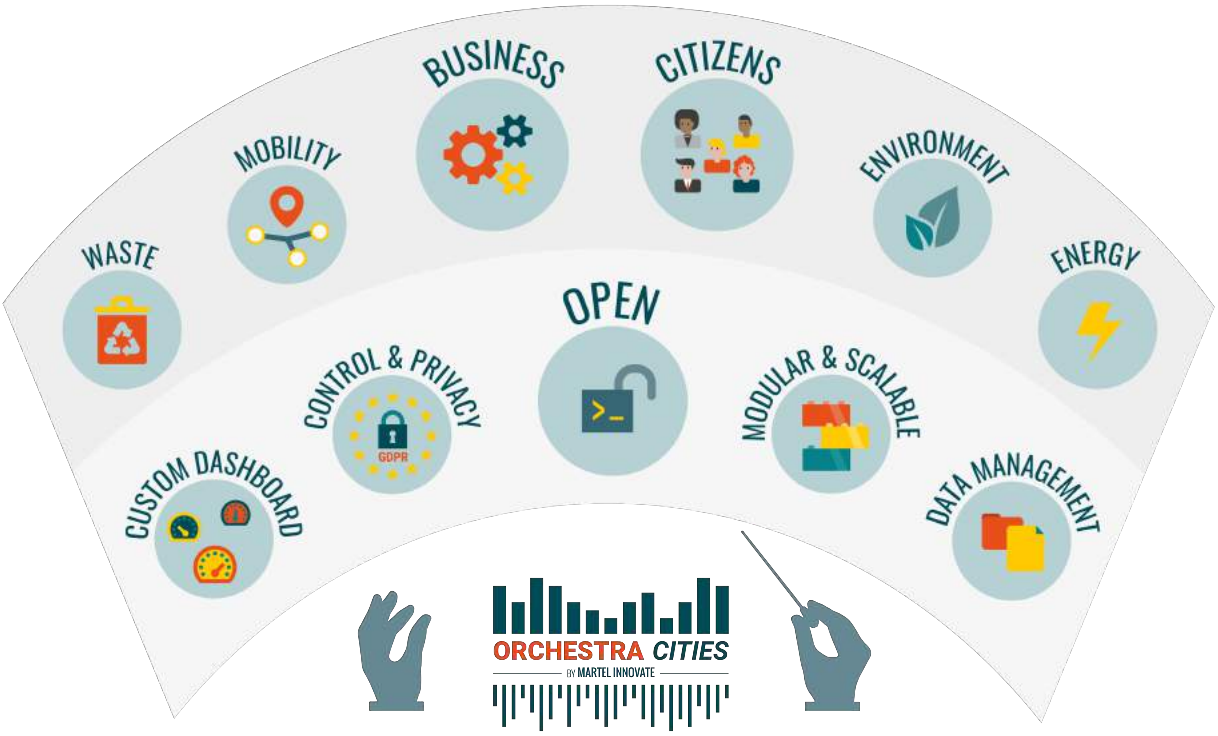
- Citizens can share data from their devices with other citizens or with the city
- Businesses can easily build services on top of APIs that are shared across different cities
- Cities can benefit from data published by other cities to create analysis, comparisons and forecast

Orchestra Cities differs from other platforms by taking the approach that the most efficient and effective way to deliver on the three previous goals is to support multiple cities in a single platform. This approach brings several advantages in terms of costs, scalability and modularity.



KEY BENEFITS

- Support the migration from vertical data silos[1] to a unified data space for a single integrated view over the city
- A collaborative space where different cities can share data and services, while retaining control on their own data
- Modular and flexible approach where each city can acquire just the needed services and resource quotas
- Reduced ownerships costs thanks to the possibility of sharing the platform among different cities
- Leverage Open Standards and Open Source code, thus building on the work of a large European and global community
- Allow citizens and businesses to take part on the city services co-creation process

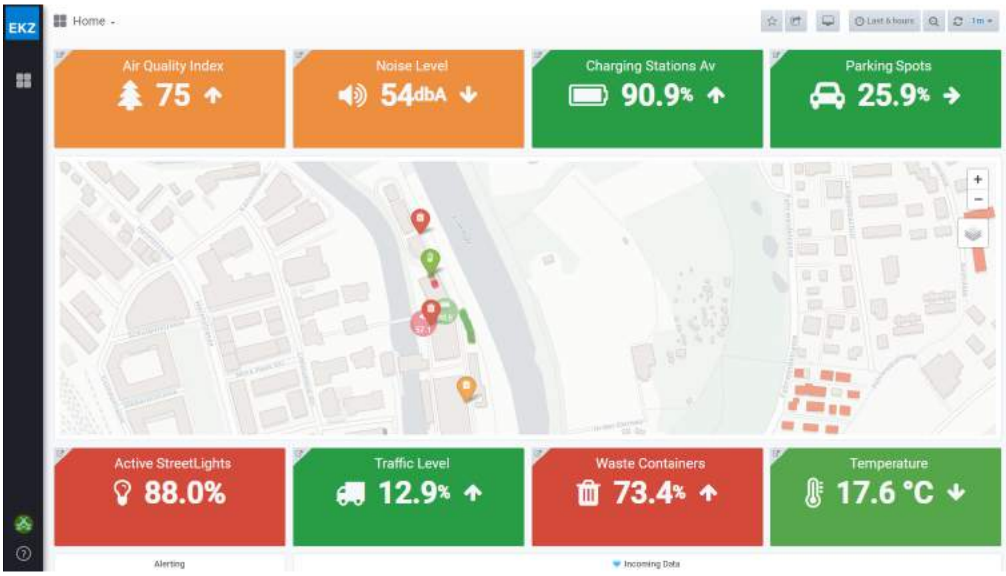


[1] The concept of ***vertical data silos*** refers to data stored in different not interoperable platforms (e.g. waste management, parking management).

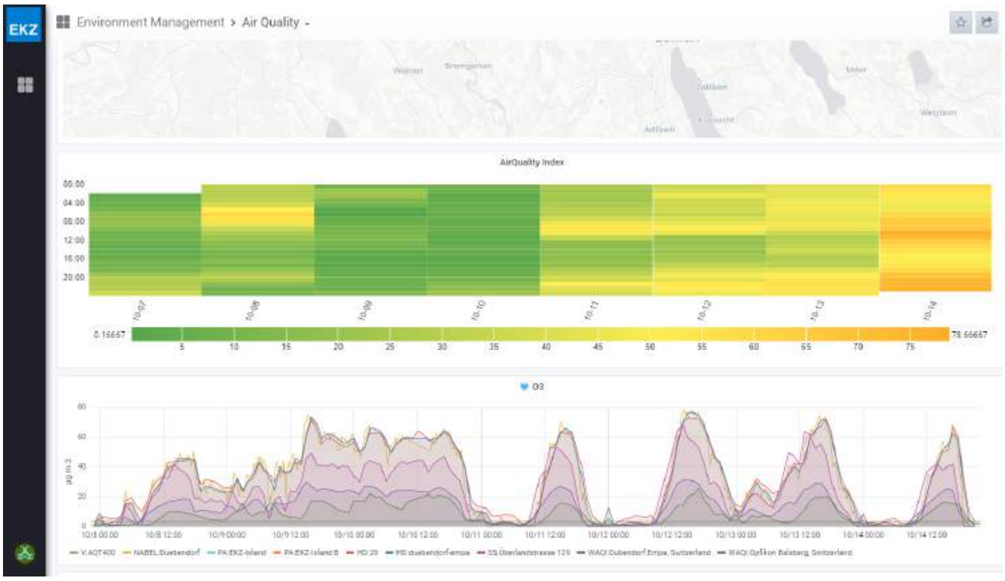
2
STORIES

EKZ

Elektrizitätswerke des Kantons Zürich (EKZ) delivers cost-effective, safe, and environmentally responsible energy to one million people in Switzerland. The power and flexibility of Orchestra Cities allows EKZ to offer cities a tailored, multi-tenant support for multiple use cases.



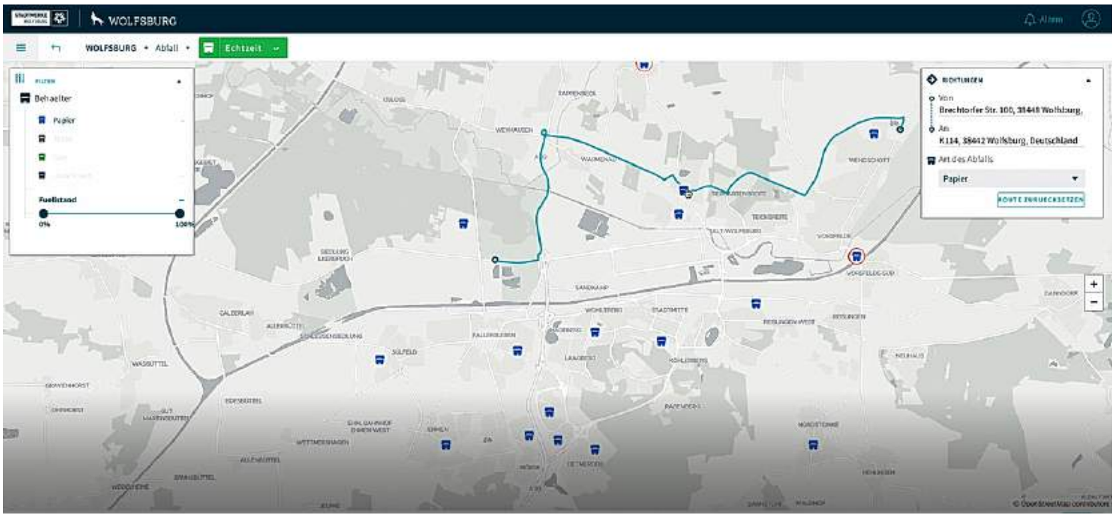
The data collection, integration, geo-tagging and time series capabilities of Quantum Leap enable sophisticated, multi-dimensional visualization of data, such as air quality.



Orchestra Cities scalable back-end reliably delivers real-time data streams to both Web dashboards and end-user mobile applications.

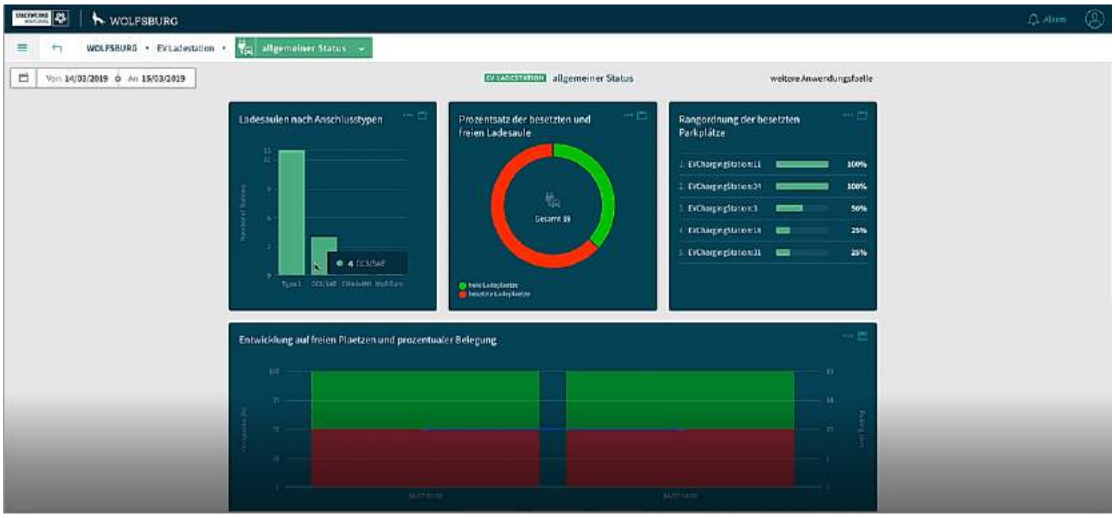
CITY OF WOLFSBURG

The municipality of Wolfsburg, Germany, leverages Orchestra Cities to provide citizens with a smart advisory system for domestic waste management. Orchestra Cities helps users to plan the most effective route for waste disposal, blending container type with real-time fill level information and user destination.



Von
Brechtorfer Str. 100, 38448 Wolfsburg,
An
K114, 38442 Wolfsburg, Deutschland
Art des Abfalls
Papier

A further scenario where Orchestra Cities delivers smart route planning involves electrical vehicles and the monitoring and management of their charging stations, with their occupancy and supported vehicles.



Von 14/03/2019 An 15/03/2019

Rangordnung der besetzten Parkplätze

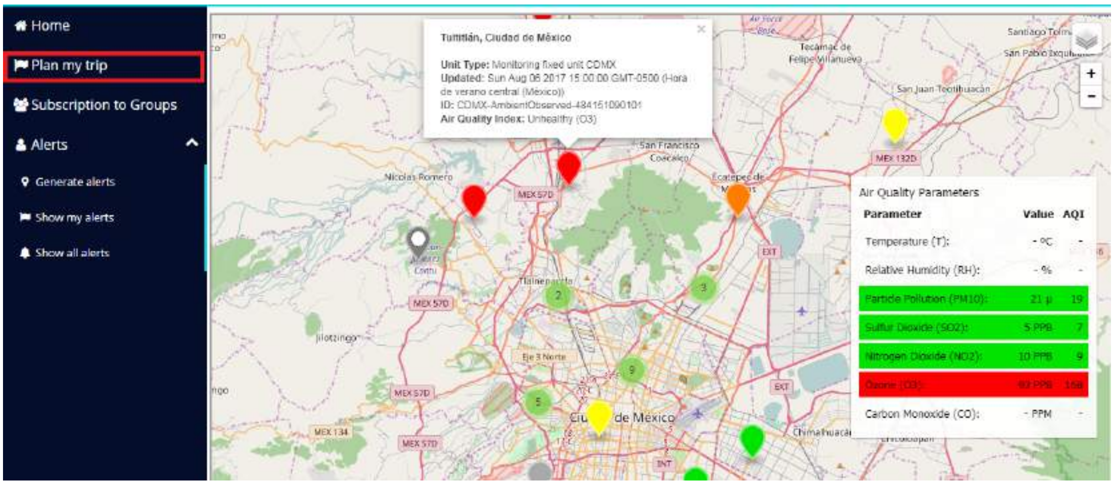
1.	EVChargingStation:11	100%
2.	EVChargingStation:24	100%
3.	EVChargingStation:3	50%
4.	EVChargingStation:18	25%
5.	EVChargingStation:31	25%

WOLFSBURG
Alarm
WOLFSBURG • EVLadestation •
allgemeiner Status
EV LADESTATION allgemeiner Status
weitere Anwendungsfaelle
Ladesaulen nach Anschlusstypen
4 CCS/SAE
Type1
CCS/SAE
Prozentsatz der besetzten und freien Ladesaule
Gesamt 19
freie Ladeplaetze
besetzte Ladeplaetze
Entwicklung auf freien Plaetzen und prozentualer Belegung

Real-time dashboards offer a higher degree of interactive UX through the Urbo framework and scenario-specific customisations.

GREEN ROUTE APPLICATION

The smart mobility application Green Route directly connects the Orchestra Cities platform with Mexico City travellers that care for their health just as much as their time.



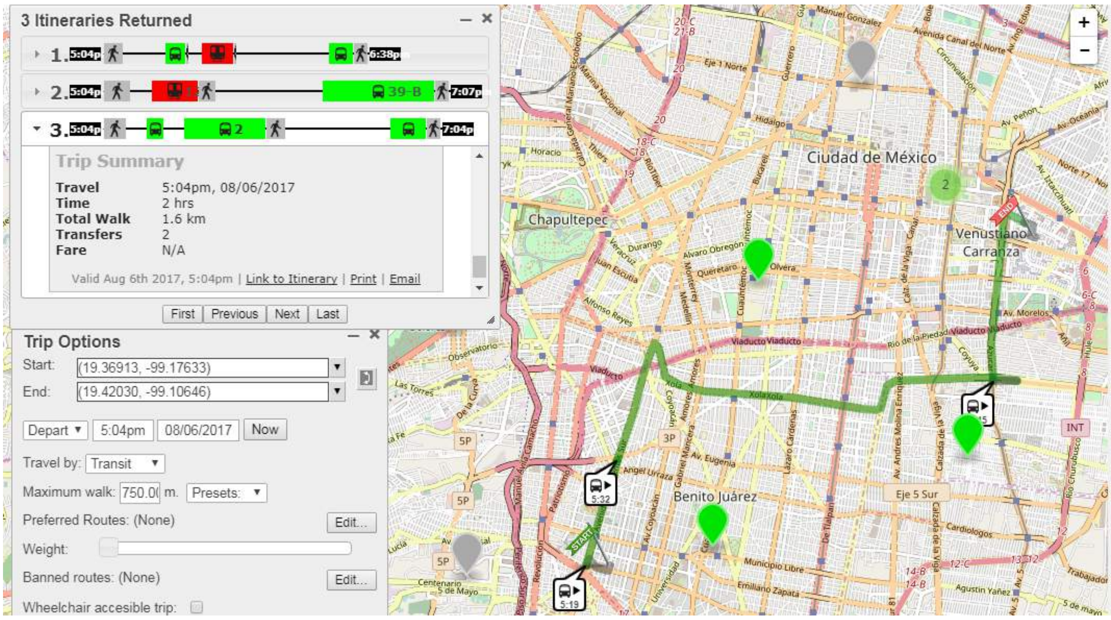
Tultitlán, Ciudad de México

Unit Type: Monitoring fixed unit CDMX
Updated: Sun Aug 06 2017 15:00:00 GMT-0500 (Hora de verano central (México))
ID: CDMX-AmbientObserved-484151090101
Air Quality Index: Unhealthy (O3)

Air Quality Parameters

Parameter	**Value**	**AQI**
Temperature (T):	- ºC	-
Relative Humidity (RH):	- %	-
Particle Pollution (PM10):	21 µ	19
Sulfur Dioxide (SO2):	5 PPB	7
Nitrogen Dioxide (NO2):	10 PPB	9
Ozone (O3):	93 PPB	168
Carbon Monoxide (CO):	- PPM	-

Orchestra Cities integrates an air quality sensor network of 47 base stations in Mexico City with traffic and public transportation data.
This provides citizens with the best route that combines public transportation, walking and city pollution avoidance. Through their mobile app, users know the best route in real time and receive relevant alerts from the communities they decide to join.



3 Itineraries Returned

Trip Summary

Travel	5:04pm, 08/06/2017
Time	2 hrs
Total Walk	1.6 km
Transfers	2
Fare	N/A

Valid Aug 6th 2017, 5:04pm | Link to Itinerary | Print | Email

First | Previous | Next | Last

Trip Options

Start: (19.36913, -99.17633)
End: (19.42030, -99.10646)
Depart ▾ 5:04pm 08/06/2017 Now
Travel by: Transit ▾
Maximum walk: 750.0(m. Presets: ▾
Preferred Routes: (None) Edit...
Weight:
Banned routes: (None) Edit...
Wheelchair accesible trip: □

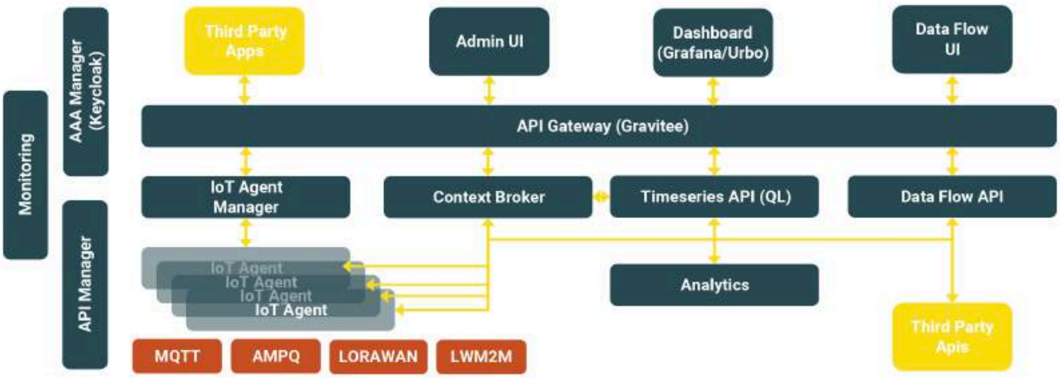
3
PLATFORM

> *"As an artificial world, so the city should be in the best sense: made by art, shaped for human purposes."*
>
> **Kevin Lynch**

The platform, as depicted in the picture above, is composed of different microservices that are orchestrated using state-of-the-art solutions such as Docker[2] and Kubernetes[3].

Orchestra Cities functionalities available as of today include:

- Security Management
- Device Management
- Data Management
- Dashboard Management
- Data Integration Management



[2] https://www.docker.com
[3] https://kubernetes.io

SECURITY

The core of the security management is based on OIDC and OAUTH 2.0 standards. The solution supports Identity, Access and Organisation Management.

- The Identity Management support enables it to manage single users (covering authentication aspects)
- The Access Management supports the control of access of users to specific applications or platform services (e.g. dashboard) with a given role (e.g. editor), thus covering authorization and audit aspects
- The Organization Management support maps users to organizations (i.e. cities) so to host a multi-tenant environment within a single platform instance (i.e. different cities, same users) approach

This solution, in combination with the support in the data management layer of data partitioning by tenant, enables the secure and controlled access by each tenant (e.g. a city) to its specific data. Moreover, it empowers different users to have access to different city data spaces with the same account.

The open source solution adopted for the Identity and Access Management is Keycloak[4], the market-leading open source identity and access management solution developed by RedHat.

The adopted open source solution for API Management is gravitee.io[5]. To apply access control to APIs, Gravitee offers a flexible plugin mechanism to implement access control policies.

The process works as follow:

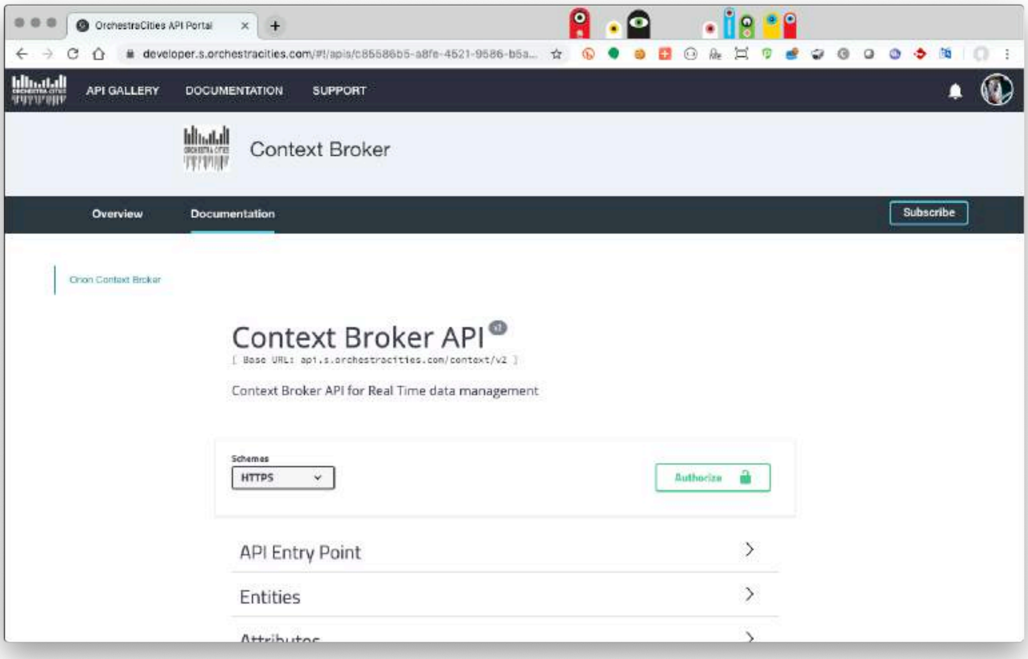
1. When an API request is generated by a client

[4] https://www.keycloak.org

[5] https://gravitee.io

2. The token for the specific Application/Client is checked at Keycloak OAuth 2.0 API

3. If the token is valid (i.e. authenticated and authorized for a given OAuth client), the policy extracts from the token the list of tenants and verifies that the requested tenant space is included in the ones accessible

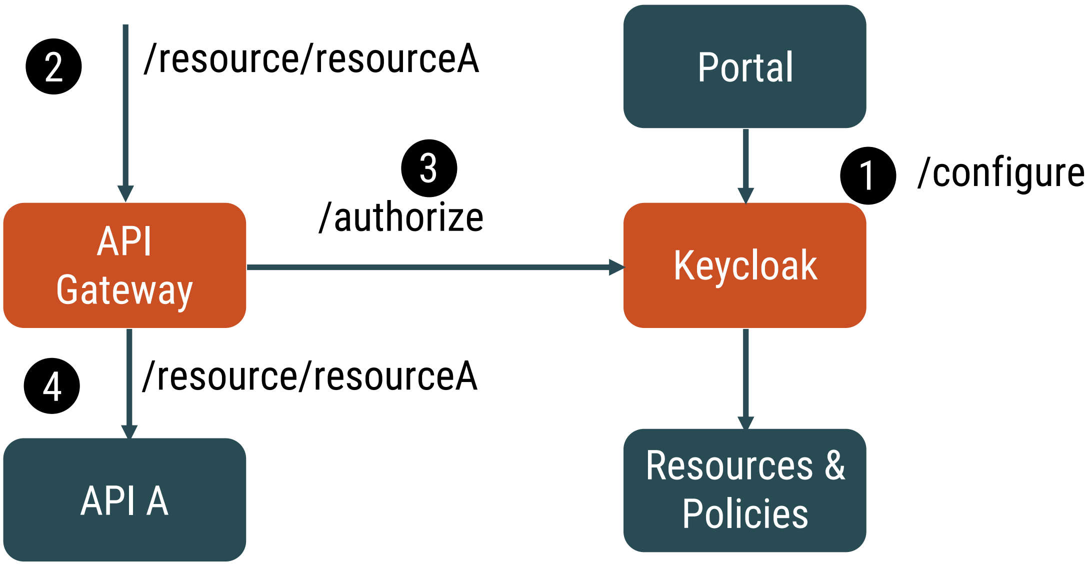


Context Broker API

[Base URL: api.s.orchestracities.com/context/v2]

Context Broker API for Real Time data management

In case of success, the call will be forwarded to the API in the backend (if not, the user will be returned a 401 “Not Authorized” response).



DATA MANAGEMENT

The core of the data management is a “data bus” collecting data from the different sources and forwarding them to the different backend APIs based on the specific scenarios. This “data bus” is provided by Orion Context Broker[6], the reference implementation for a NGSIv2 broker. All data used in the platform transits through it: IoT Devices data, External services data, Platform generated data.

Services can provide data to the Orion Context Broker with the following approaches:

- Data Push: services send data to the Context Broker
- Data Pull: services expose data via a standard API, that the Orion Context Broker queries to retrieve data

Similarly, services can obtain data from the Context Broker with the following two techniques:

- Data Subscription: services subscribe to a given data and get notified when the data is updated
- Data Query: services query the Orion Context Broker to retrieve data

These interaction modes provide a very flexible way to integrate data provider (e.g. sensors) and data processing services (e.g. analytics). In the case of IoT Agent, as discussed in the following section, the recommended solution is the Data Push model (however the Data Pull model is also supported), since this allows constantly up-to-date data in the Orion Context Broker that can be used to generate Data Subscriptions.

The other core component of the Data Management layer is the Timeseries API. The role of this component is to store all the historical data of a given entity (Orion Context Broker stores only the current value in time of a data point). The Timeseries API is provided by Quantum Leap[7], an NGSIv2 compliant time series API. Quantum Leap supports as backend CrateDB and TimescaleDB, which are also supported by Grafana (see next section) to generate dashboards. Quantum Leap supports a variety of queries (including

[6] https://fiware-orion.readthedocs.io/en/master/

[7] https://quantumleap.readthedocs.io

geographical-based), facilitating the access to historical data by services in need of working on batch data sets.

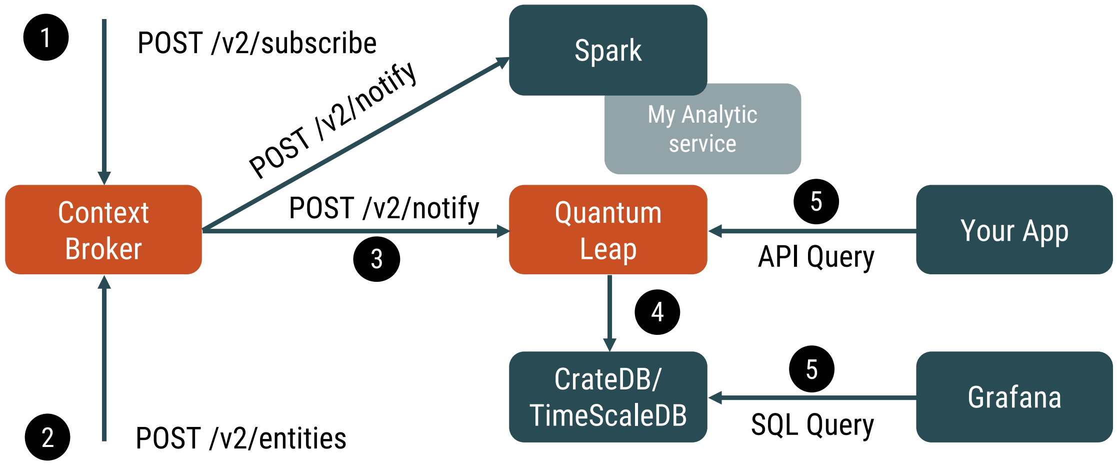
In short, the process will work as follows:

1. A subscription is created for each data model (e.g. Weather), to get notifications forwarded to the Timeseries API service (e.g. Quantum Leap)

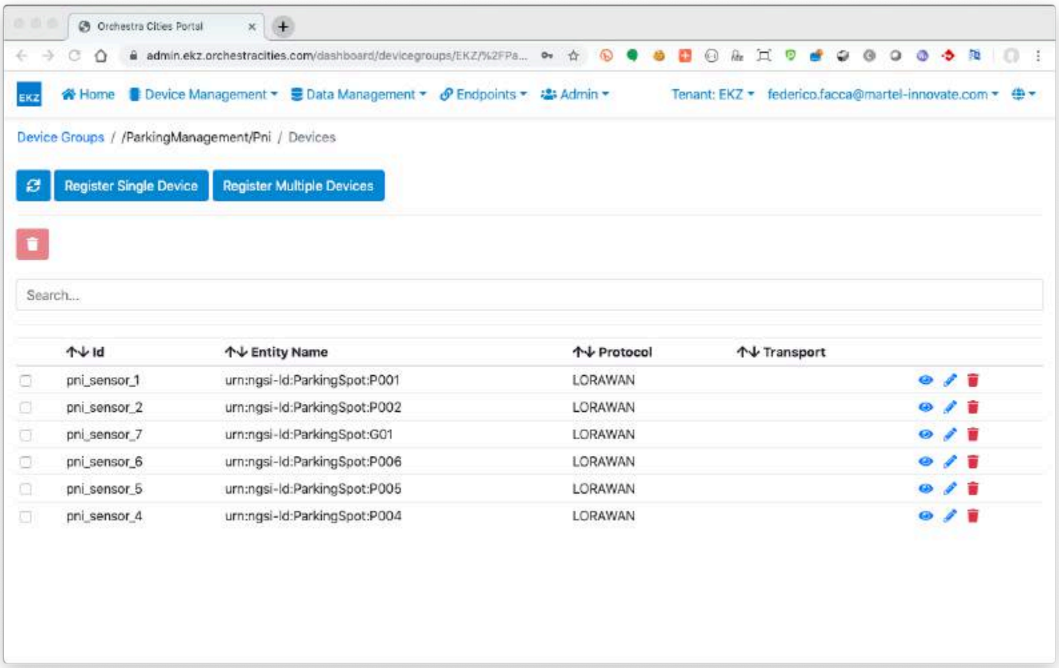
2. Data of entities matching the created subscription gets updated in the Context Broker

3. Notifications are sent to the subscribed services (e.g. Quantum Leap). Each notification can include either the whole data model or a fragment thereof, depending on the created subscription

4. The subscribed service processes it using its logic (in the case of Quantum Leap, it stores the received data in CrateDB or in TimescaleDB)



DEVICE MANAGEMENT


Orchestra Cities Portal
admin.ekz.orchestracities.com/dashboard/devicegroups/EKZ/%2FPa...
EKZ
Home
Device Management
Data Management
Endpoints
Admin
Tenant: EKZ
federico.facca@martel-innovate.com
Register Single Device
Register Multiple Devices
Search...

Device Groups / /ParkingManagement/Pni / Devices

	Id	Entity Name	Protocol	Transport
□	pni_sensor_1	urn:ngsi-ld:ParkingSpot:P001	LORAWAN	
□	pni_sensor_2	urn:ngsi-ld:ParkingSpot:P002	LORAWAN	
□	pni_sensor_7	urn:ngsi-ld:ParkingSpot:G01	LORAWAN	
□	pni_sensor_6	urn:ngsi-ld:ParkingSpot:P006	LORAWAN	
□	pni_sensor_5	urn:ngsi-ld:ParkingSpot:P005	LORAWAN	
□	pni_sensor_4	urn:ngsi-ld:ParkingSpot:P004	LORAWAN	

To manage the IoT devices, Orchestra Cities leverages FIWARE stack and hence the NGSIv2 API and data format. FIWARE offers a wide range of so-called IoT Agents. Each IoT Agent enables different transport and message protocols to be used to connect IoT Devices. Orchestra Cities covers all protocols supported by FIWARE (UL, JSON, LORAWAN), the recommended one being UL, a very lightweight message protocol adopting a compact representation (e.g. attribute1|value1|attribute2|value2 becomes t|10|s|true|l|78.8) that supports MQTT, AMQP or HTTP transports. The IoT Agent role is to map low level messages generated by the device to higher level information used at the so-called application layer and to forward the structured and aggregated information to the data management layer of the platform. FIWARE IoT Agents support the concept of “device groups”, i.e. a set of devices sharing the same information model. This facilitates the registration of a large set of devices that provides the same information, which would otherwise have required the user to configure each device with its individual information (often not required at all).

To facilitate the registration of IoT Devices within IoT Agents, we provide a UI, thus simplifying the overall operation. The process will work as follow:

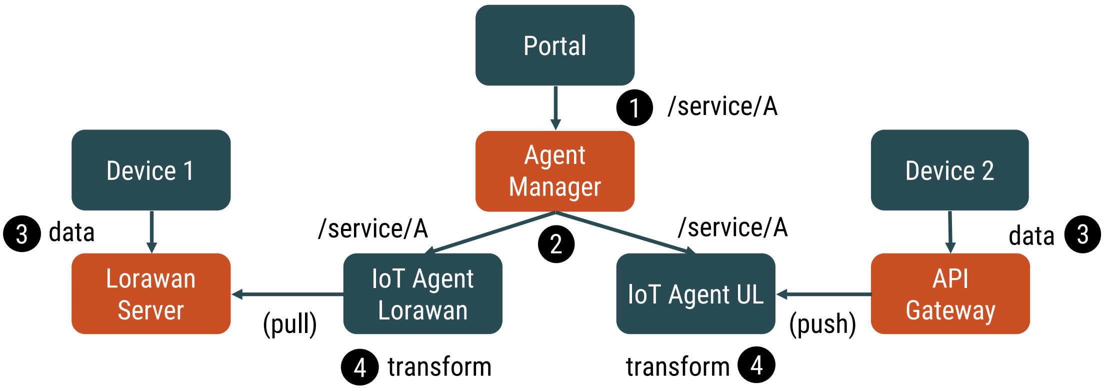
1. Through the portal (or the API) a user can register a device

2. The device configuration (which includes for example the transport, besides the attribute mapping discussed above) is stored in the device registry for later use

3. Once configured, the device can send messages to the agent (in the picture, the HTTP or LORAWAN transport is assumed)

4. When receiving a payload, the Agent checks the configuration of the device to transform the incoming “simple” message into the NGSIv2 payload

5. Finally, the Agent sends the NGSIv2 payload to the Context Broker

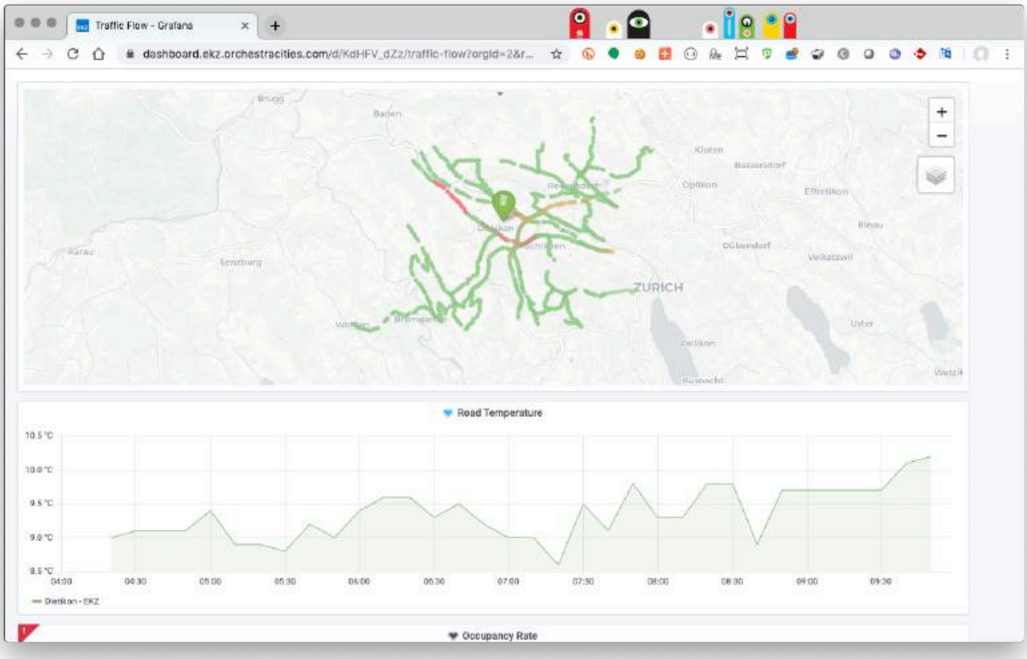


DASHBOARD MANAGEMENT

To allow the creation of custom dashboards, we use an open-source dashboard engine called Grafana, an open platform for beautiful analytics and monitoring. This technology integrates a set of “panels” that provides support for rendering objects such as lines, points, bars and heat graphs, basic maps with info pointers, picture panels and more useful panels to display any kind of data. It also includes a collection of data-source plugins, that allows it to integrate Grafana with different databases and backends such as CrateDB and TimescaleDB (the back-ends supported by Quantum

Leap), JSON, and Google calendar. Grafana aims to provide an easy and intuitive way for public officers to monitor different KPIs of their city. This dashboard can run on multiple end-user devices without installation and provides good responsiveness for the screen sizes of desktop computers, mobile phones and tablets.

Also, it can show online historical data while filtering and sorting data dynamically. The data can be zoomed in to have fine-grained views of values or the same data can be seen in a Tabular format, allowing users to sort data by different column values. The ability to present maps with information points allows cities to visualize in real-time all the data that is being collected by the sensors. Besides that, cities are able to define actions based on the information and events of the dashboards and create alerts based on data thresholds; for example set an alert when a waste bin is on fire. Dashboards can be easily shared, customized and embedded in other tools.

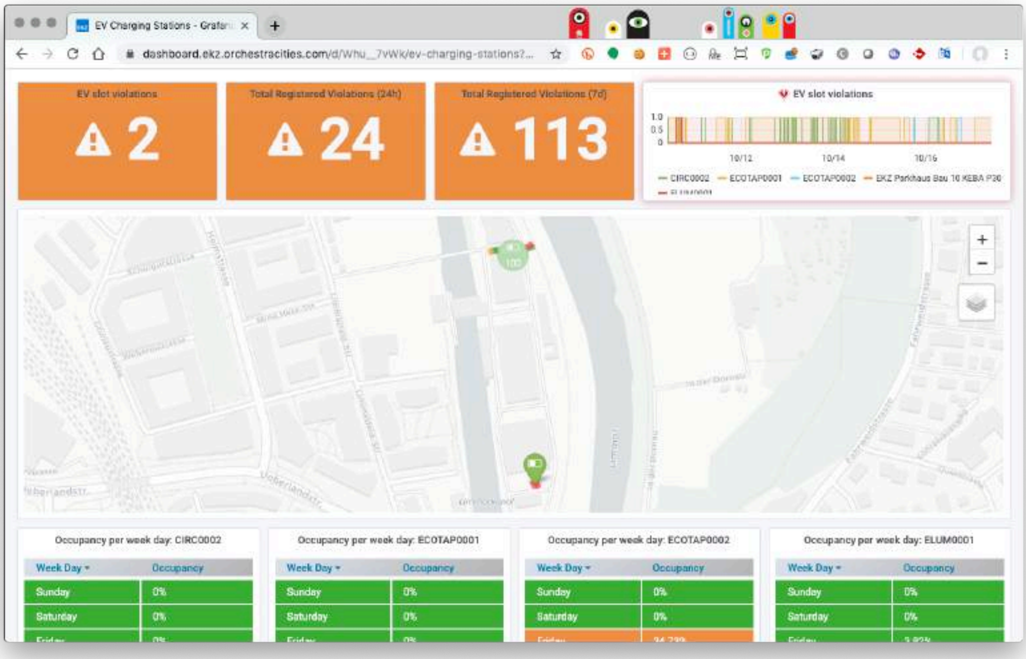


ANALYTICS

While the **dashboard** can provide simple real-time analytics, for more complex tasks we integrated an Apache Spark[8] cluster in the platform. The

[8] https://spark.apache.org

cluster, integrated to the Data Management layer, enables the analysis of a data set (e.g. Parking Spots) whether they are real-time or historical data. For example combining data of parking spots and electric vehicle charging stations, it is possible to detect parking violations (e.g. cars parked in a spot for electric vehicles).


EV Charging Stations - Grafana
dashboard.ekz.orchestracities.com/d/Whu__7vWk/ev-charging-stations?...
EV slot violations
1.0
0.5
0
10/12
10/14
10/16
CIRC0002
ECOTAP0001
ECOTAP0002
EKZ Parkhaus Bau 10 KEBA P30
Uebelandstr.

EV slot violations: 2

Total Registered Violations (24h): 24

Total Registered Violations (7d): 113

Occupancy per week day: CIRC0002

Week Day	Occupancy
Sunday	0%
Saturday	0%
Friday	0%

Occupancy per week day: ECOTAP0001

Week Day	Occupancy
Sunday	0%
Saturday	0%
Friday	0%

Occupancy per week day: ECOTAP0002

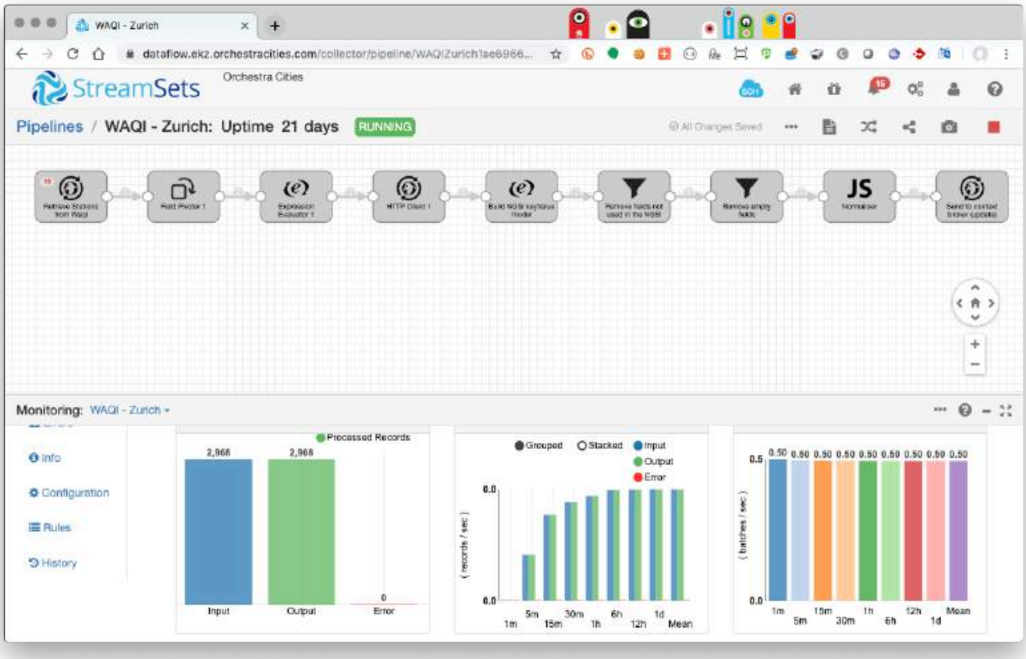
Week Day	Occupancy
Sunday	0%
Saturday	0%
Friday	34.73%

Occupancy per week day: ELUM0001

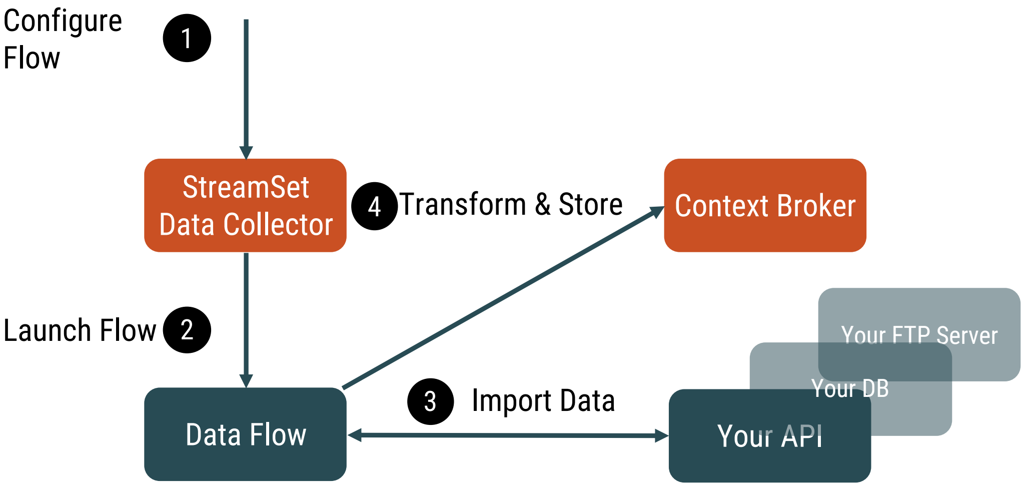
Week Day	Occupancy
Sunday	0%
Saturday	0%
Friday	3.92%

THIRD PARTY DATA INTEGRATION

To integrate external services and data sources, Orchestra Cities currently leverages StreamSets Data Collector. This tool allows the creation of pipelines for data harmonization and injection into the platform. Pipelines can be live tested, thus simplifying their development. Pipeline processors allow the interaction with web services, databases, files and other sources allowing to integrate virtually any external source. Each pipeline can be parameterised and saved to replicable templates so that different cities can instantiate and customise their own data import flow for a given service.



1. Through StreamSet UI (or the API) a developer can define a data import pipeline
2. Once started, the pipeline is deployed and executes the different steps defined
3. These usually include interaction with external APIs, files, or databases
4. Aggregated data are transformed into NGSIv2 and stored in the Context Broker



OPEN STANDARDS

Orchestra Cities relies on different Open Standards that facilitate the integration with existing solutions.

SECURITY	OAUTH 2.0, OIDC, SAML, KERBEROS, LDAP, X.509
IOT	**Protocols:** UL, JSON, LWM2M **Transport:** HTTP, COAP, MQTT, AMQP, LORAWAN
DATA EXCHANGE	**Protocols:** JSON/REST, XML/SOAP, FTP/HTTP, WEBSOCKETS **Data format:** NGSI, JSON, GEOJSON, CSV, EXCEL, TEXT
CLOUD	Docker, Kubernetes

4
ROADMAP

> *“If everyone is moving forward together, then success takes care of itself.”*
>
> **Michael Jordan**

Orchestra Cities is under active development, these are some of the features we plan to work on in 2020:

- **Live City:** add new use cases to our City-Centric App supporting pre-cooked scenarios to help the management of cities based on Smart Data models (https://fiware-datamodels.readthedocs.io/en/latest/index.html). E.g. status of waste bins, parking, …
- **Machine Learning:** extend demonstrators to show how Machine Learning can be applied to create models and data forecasts in Orchestra Cities
- **Linked Data:** explore the integration of NGSI-LD in the platform by testing components supporting NGSI-LD as of today
- **Advanced data access control:** increasing control on data access via more fine grained access policies
- **Supporting edge computing solutions:** a demonstrator of how services can be deployed across the cloud-edge
- **Blockchain:** explore hashing of data to collect proof of data integrity
- Improved **integration with CKAN**

ORCHESTRA CITIES
BY MARTEL INNOVATE

5
OUR PARTNERS

> *"It is the long history of humankind that those who learned to collaborate and improvise most effectively have prevailed.".*
>
> **Charles Darwin**

To develop our solutions and to test them we have worked and continue to work with a number of selected partners.

Atos FIWARE redhat

Telefonica Gravitee Source ubiwhere

therapænts hop ubiquitous SYNELIXIS

Solutions have been tested in POC with:

ANTWERP
WOLFSBURG
HELSINKI
MEXICO CITY
ZURICH

ORCHESTRA CITIES

BY MARTEL INNOVATE

MARTEL innovate

DO YOU WANT TO BE THE NEXT ORCHESTRA CITY?

🌐 ORCHESTRACITIES.COM

@ORCHESTRACITYEU

CONTACT US AT **INFO@ORCHESTRACITIES.COM**