SNAP4CITY

www.snap4city.org
www.snap4solutions.org

www.km4city.org



Overview: IoT App. / Proc.Logic, Ingestion and Interoperability

February 2024, Course, Synt of Parts 3,5,6

https://www.snap4city.org/944
https://www.snap4city.org/577

0 1 0 0 1 0 0 1 1 0 1 1
0 1 1 1 0 1 0 1 0 0 1 0

DIGITAL TWIN SOLUTIONS TO SETUP SUSTAINABLE DECISON SUPPORT SYSTEMS AND BUSINESS INTELLIGENCE

UNIVERSITÀ DEGLI STUDI FIRENZE
DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

CC BY NC ND
SOME RIGHTS RESERVED

Paolo Nesi, paolo.nesi@unifi.it
https://www.Km4City.org
https://www.disit.org

SNAP4CITY

KM4CITY

SNAP4 Appliances and Dockers Installations



Be smart in a SNAP!

SMARTCITY EXPO WORLD CONGRESS

100% OPEN SOURCE

EU GDPR COMPLIANT

PEN Test Passed

Powered by FIWARE

SNAP4CITY on EUROPEAN OPEN SCIENCE CLOUD MARKETPLACE

Node-RED

Overview: IoT App. / Proc.Logic, Ingestion and Interoperability

February 2024, Course, Synt of Parts 3,5,6
https://www.snap4city.org/944
https://www.snap4city.org/577

LIVING LAB
SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES

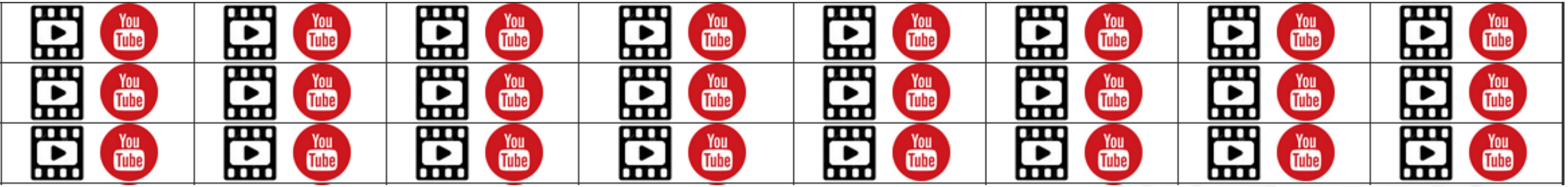
UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

CC BY NC ND
SOME RIGHTS RESERVED

https://www.snap4city.org/944

On Line Training Material (free of charge)

1st part	2nd part	3rd part	4th part	5th part	6th part	7th part	8th
Overview	Dashboards	OT App, IOT Network	Data Analytics	Data Ingestion processes	System and Deploy Install	Smart City API: Web & Mob. App	Design and Develop Smart Solutions
							
							



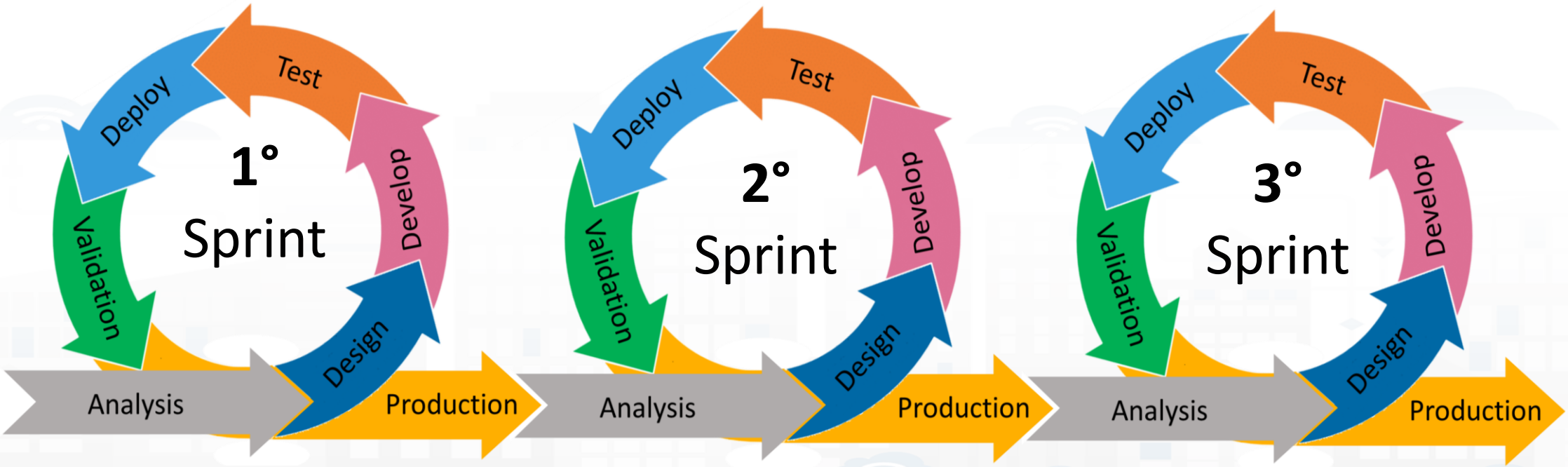
Note on Training Material

- **Snap4City for Dummies https://www.snap4city.org/1046**
- **Course**: https://www.snap4city.org/944
  - Introductionary course to Snap4City technology
  - Italian version CN MOST: https://www.snap4city.org/1068
- **Older version Course** https://www.snap4city.org/577
  - Full training course with much more details on mechanisms and a wider set of cases/solutions of the Snap4City Technology
- **Documentation** includes a deeper round of details
  - Snap4City Platform Overview:
    - https://www.snap4city.org/drupal/sites/default/files/files/Snap4City-PlatformOverview.pdf
  - Development Life Cycle:
    - https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf
  - Client Side Business Logic:
    - https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf
- **On line cases and documentation**:
  - https://www.snap4city.org/108
  - https://www.snap4city.org/78
  - https://www.snap4city.org/426

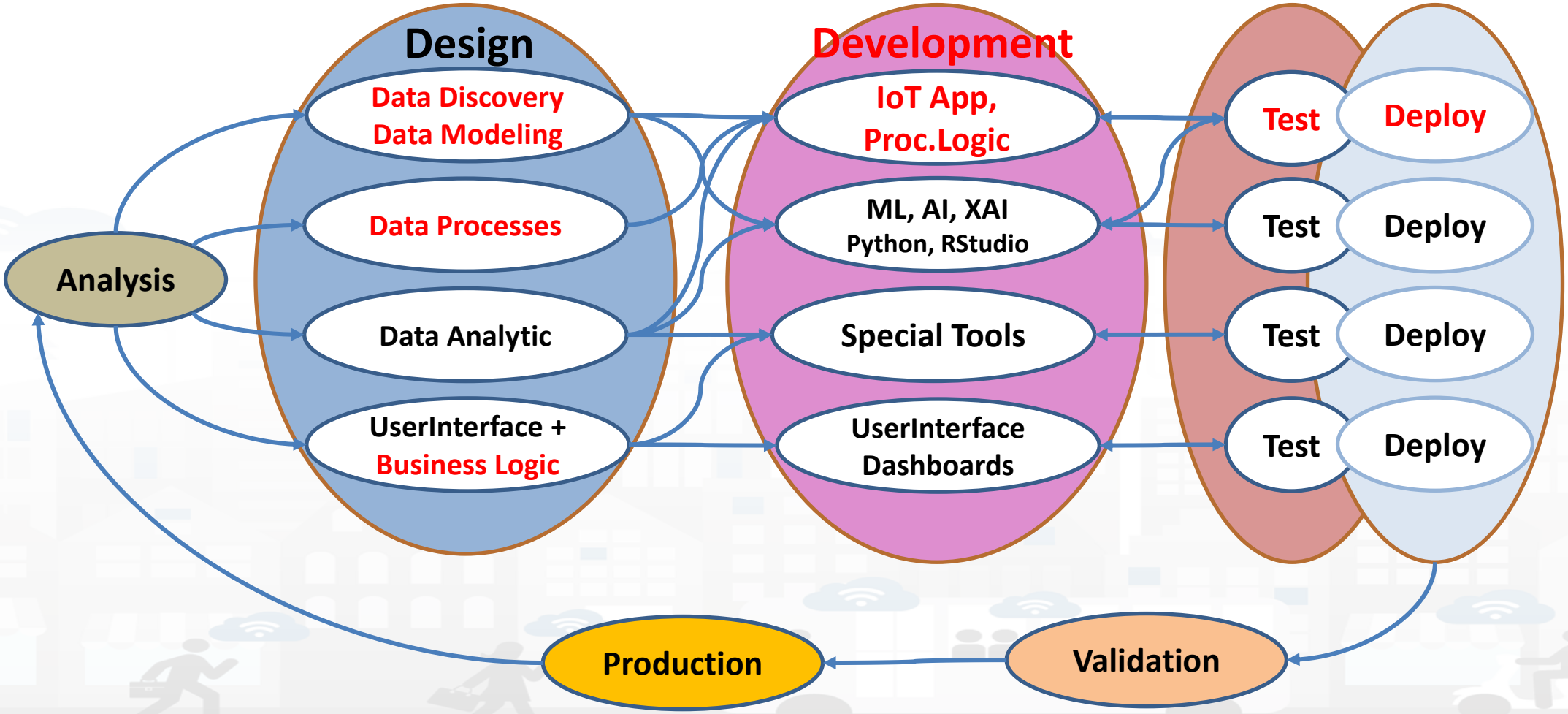
Agenda of this integrated overview part

- Recall of Snap4City Architecture
- Data Ingestion Strategy and Orientation
  - Basic entity elements
  - Knowledge Base: Modelling and ServiceURI as Entity Identifier
  - Models vs Devices/Entities and Registration
- Develop: Data Processes Proc.Logic / IoT App
  - Proc.Logic = Node-RED + Snap4City
  - An Integrated Example for Time Series
  - Verification of Data Ingestion
  - Exploiting Storage data by using: IoT App / Proc. Logic
  - Interoperability with respect to Hardware staff
  - High Performance Ingestion
  - Interoperability of Snap4City Platform
- Some Applicative examples
- Training Material
- Acknowledgements

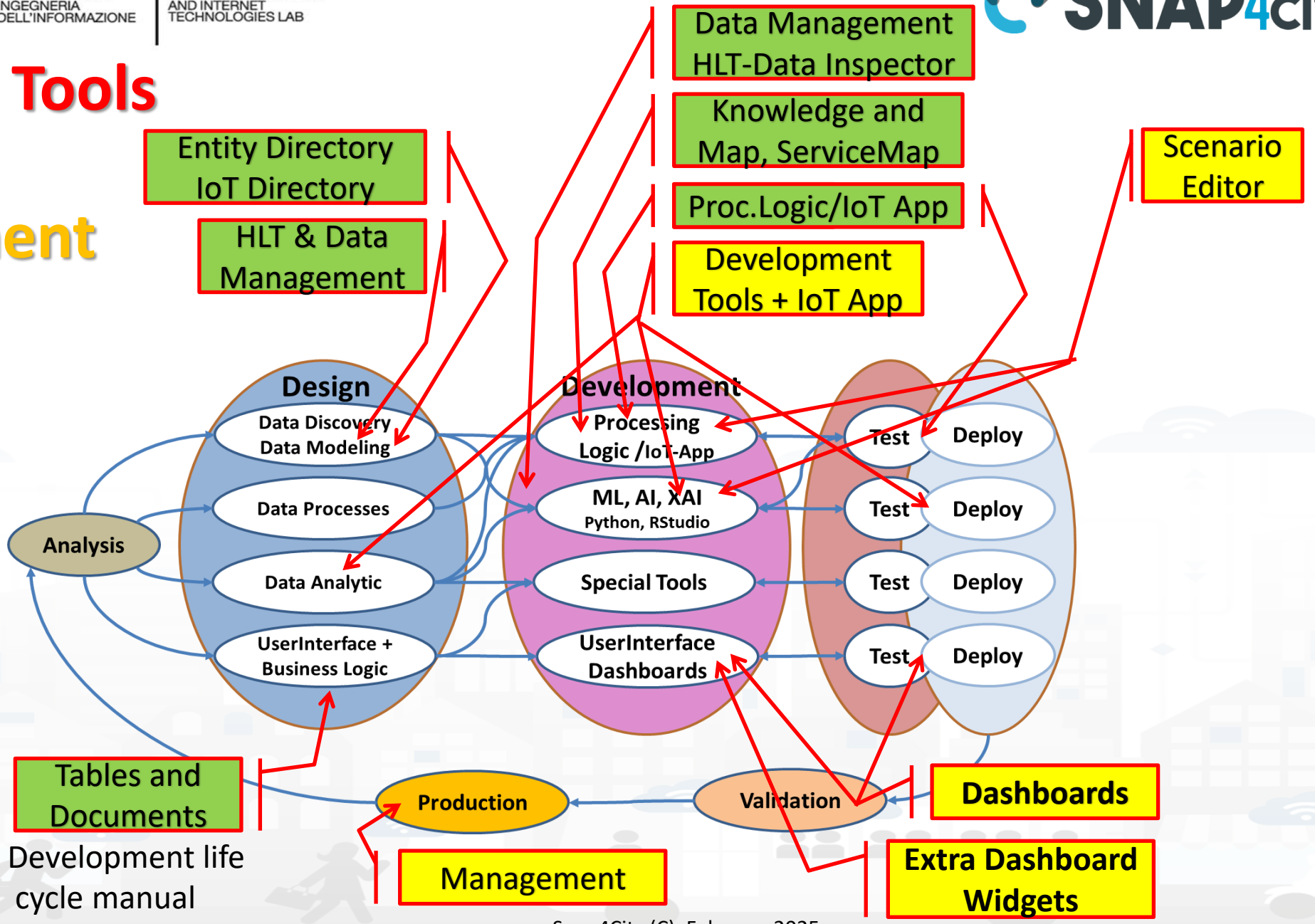
Development Life Cycle Smart Solutions



Development Life Cycle Smart Solutions



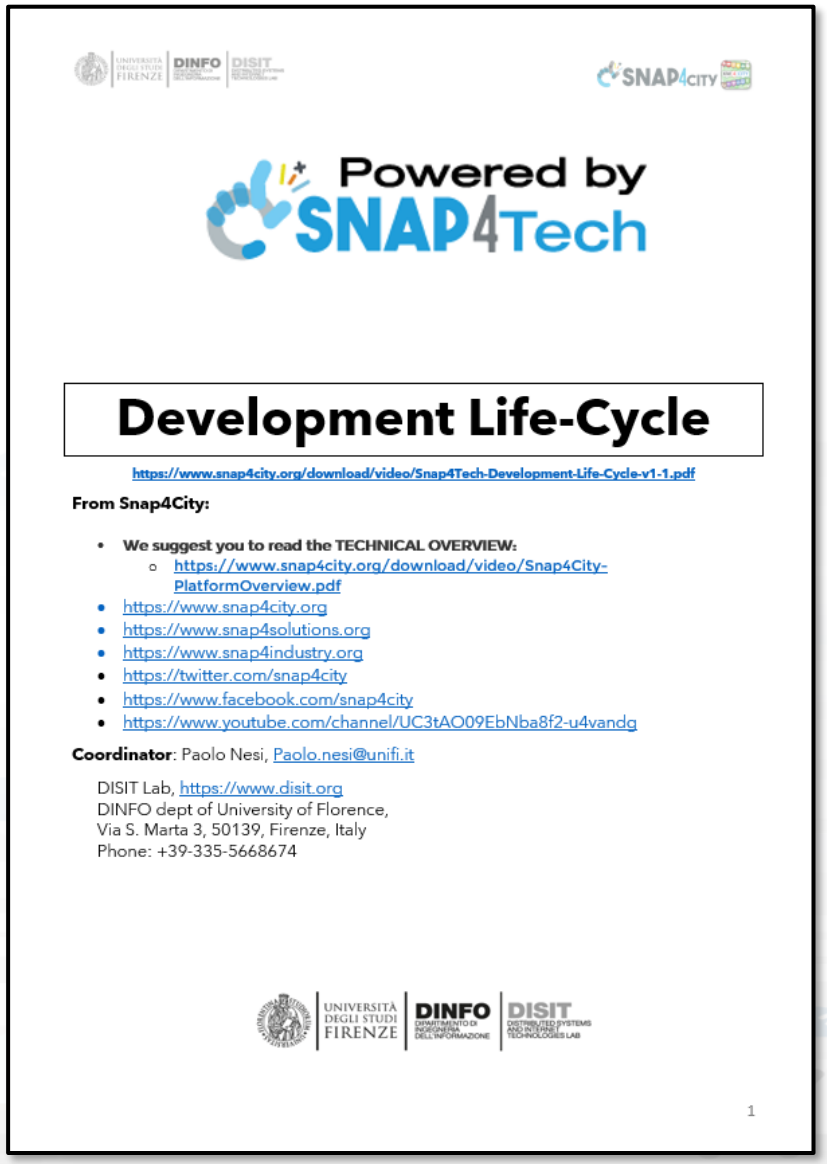
Snap4City Tools vs Development Life Cycle



Development life cycle manual

Development

https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf



Powered by SNAP4Tech

Development Life-Cycle

https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle-v1-1.pdf

From Snap4City:

- **We suggest you to read the TECHNICAL OVERVIEW:**
  - https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- https://www.snap4city.org
- https://www.snap4solutions.org
- https://www.snap4industry.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city
- https://www.youtube.com/channel/UC3tAO09EbNba8f2-u4vandg

Coordinator: Paolo Nesi, Paolo.nesi@unifi.it

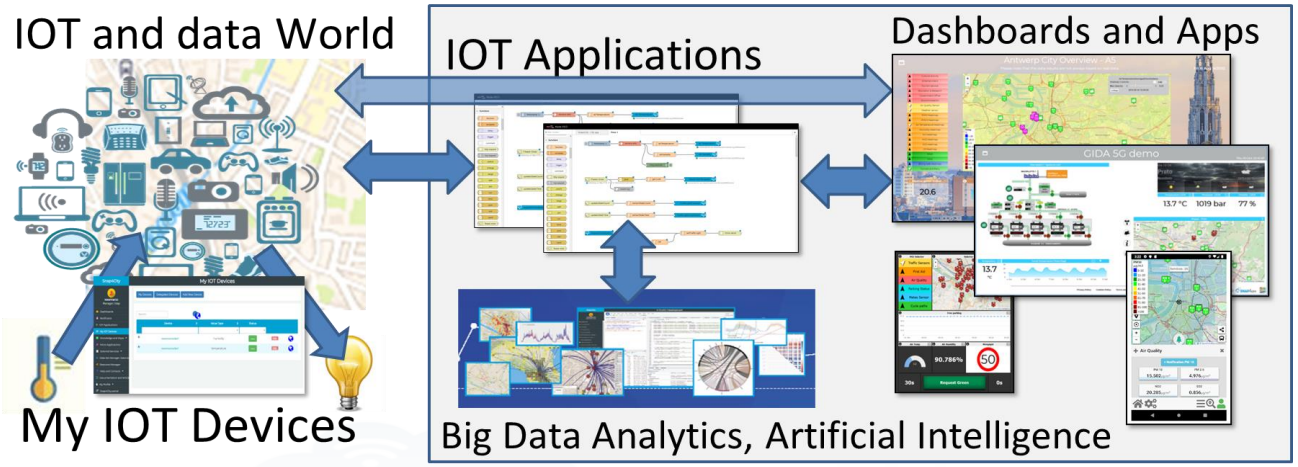
DISIT Lab, https://www.disit.org
DINFO dept of University of Florence,
Via S. Marta 3, 50139, Firenze, Italy
Phone: +39-335-5668674

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

Free Trial

- Register on WWW.snap4city.org
  - Subscribe on **DISIT Organization**
- **You can:**
  - Access on basic Tools
  - Access to a large volume of Data
  - Create Dashboards
  - Create IOT Applications
  - Connect your IOT Devices
  - Exploit Tutorials and Demonstrations



IF you need to go more in deep you can ask us to pass at the next Role becoming full AreaManager with full rights of development, also for Data Analytics, machine learning, etc.

Recall on Snap4City Architecture



Digital Twin Solutions for Sustainability

100% OPEN SOURCE

Powered by FIWARE

FREE TRIAL

PEN Test Passed

EU GDPR COMPLIANT

SNAP4 Appliances and Dockers Installations

EUROPEAN OPEN SCIENCE CLOUD

Node-RED

JS Foundation

E015 digital ecosystem

NVIDIA

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB | SNAP4CITY | KM4CITY

OPERATION AND PLAN - CONTROL ROOMS - DECISION SUPPORT SYSTEMS - WHAT-IF ANALYSIS - OPTIMIZATION - APPLICATIONS

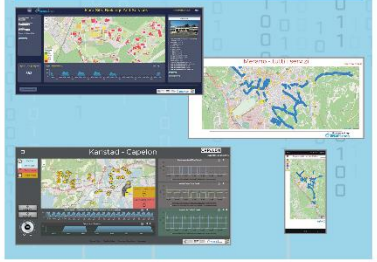
HORIZONTAL AI PLATFORM



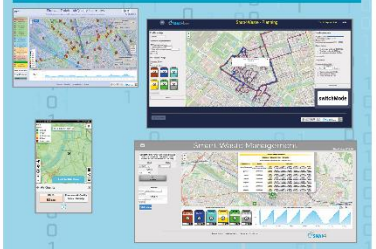
MOBILITY AND TRANSPORT



SMART ENERGY AND SMART BUILDING



ENVIRONMENT AND WASTE MANAGEMENT



CITY USER'S SERVICES AND TOURISM MANAGEMENT



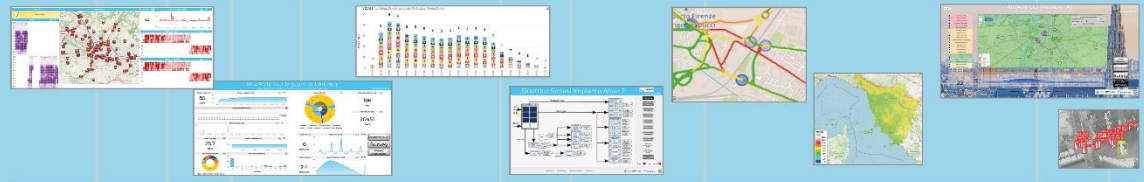
- DEVELOMENT ENVIRONMENT AND METHODOLOGY
- VISUAL PROGRAMMING, ML, AI, HPC
- TRAINING COURSES
- LIVING LABS
- GUI CUSTOM STYLES
- FULL APPLICATIONS, DASHBOARDS AND VIEWS
- MOBILE APPS



VISUAL ANALYTICS - SYNOPTICS - GRAPHICAL WIDGETS - ANALYTICS - BUSINESS INTELLIGENCE - SIMULATIONS



DASHBOARDS, WIDGETS
TEMPLATES



PREDICTION - ANOMALY DETECTION - CLUSTERING - ROUTING - SENTIMENT NLP - TRAFFIC FLOW - PEOPLE FLOWS - SDG
15 MIN CITY INDEX - KPI - HEATMAPS - ORIGIN DESTINATION - ETC...



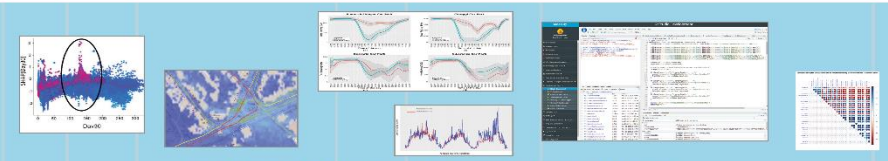
API - MICROSERVICES - GIS - BPM
VIDEO - REPORTS - MAPS - 3D ...



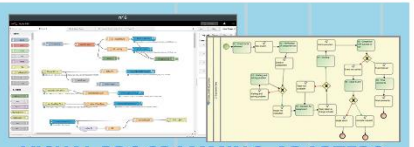




EXPERT SYSTEM, KNOWLEDGE BASE
SEMANTIC REASONING
SMART DATA MODEL
IOT DEVICE MODELS, STORAGE



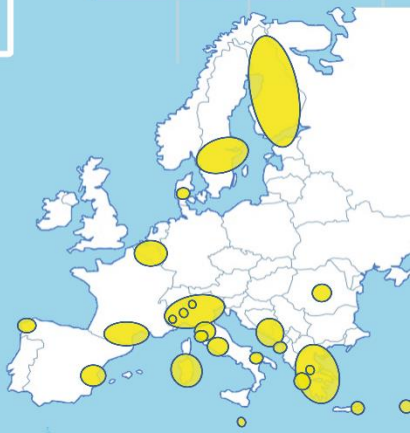
BIG DATA ANALYTICS, ARTIFICIAL INTELLIGENCE
EXPLAINABLE AI, MACHINE LEARNING, GENERATIVE AI
OPERATIVE RESEARCH, STATISTICS



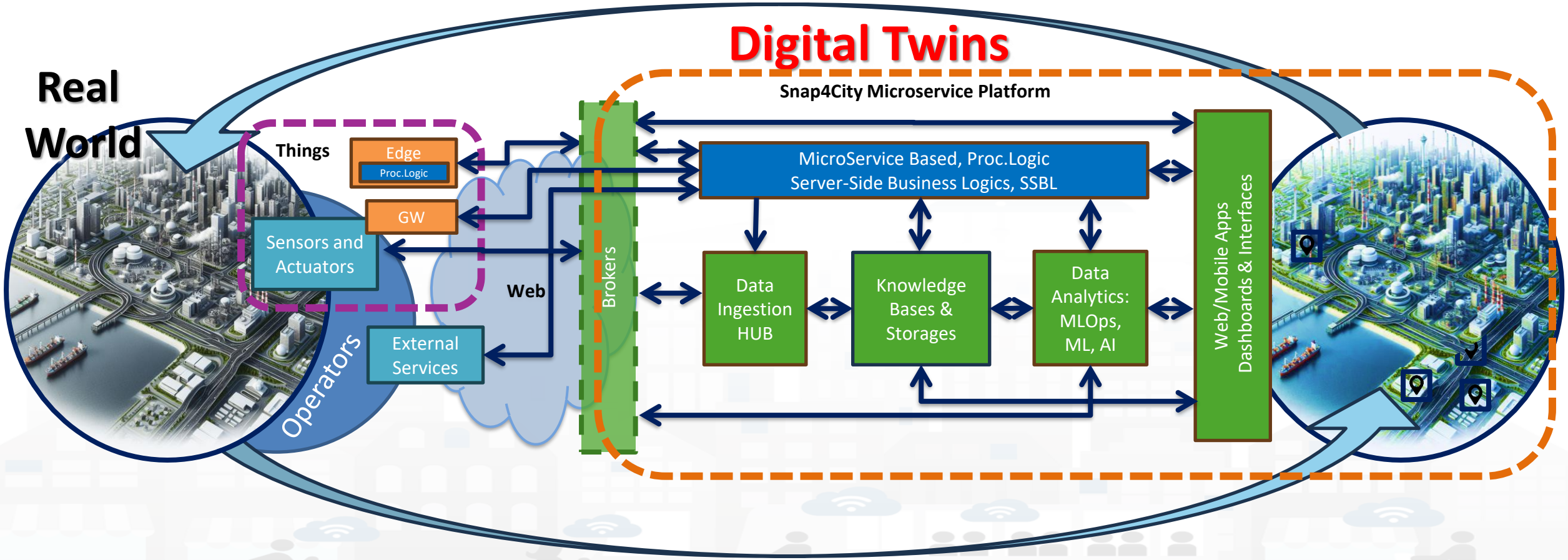
VISUAL PROGRAMMING, ADAPTERS
DATA FLOWS, WORKFLOWS
PARALLEL DISTRIBUTED PROCESSING
DATA DRIVEN

FULL INTEROPERABILITY, ANY: DATA, BROKERS, NETWORKS AND VERTICALS

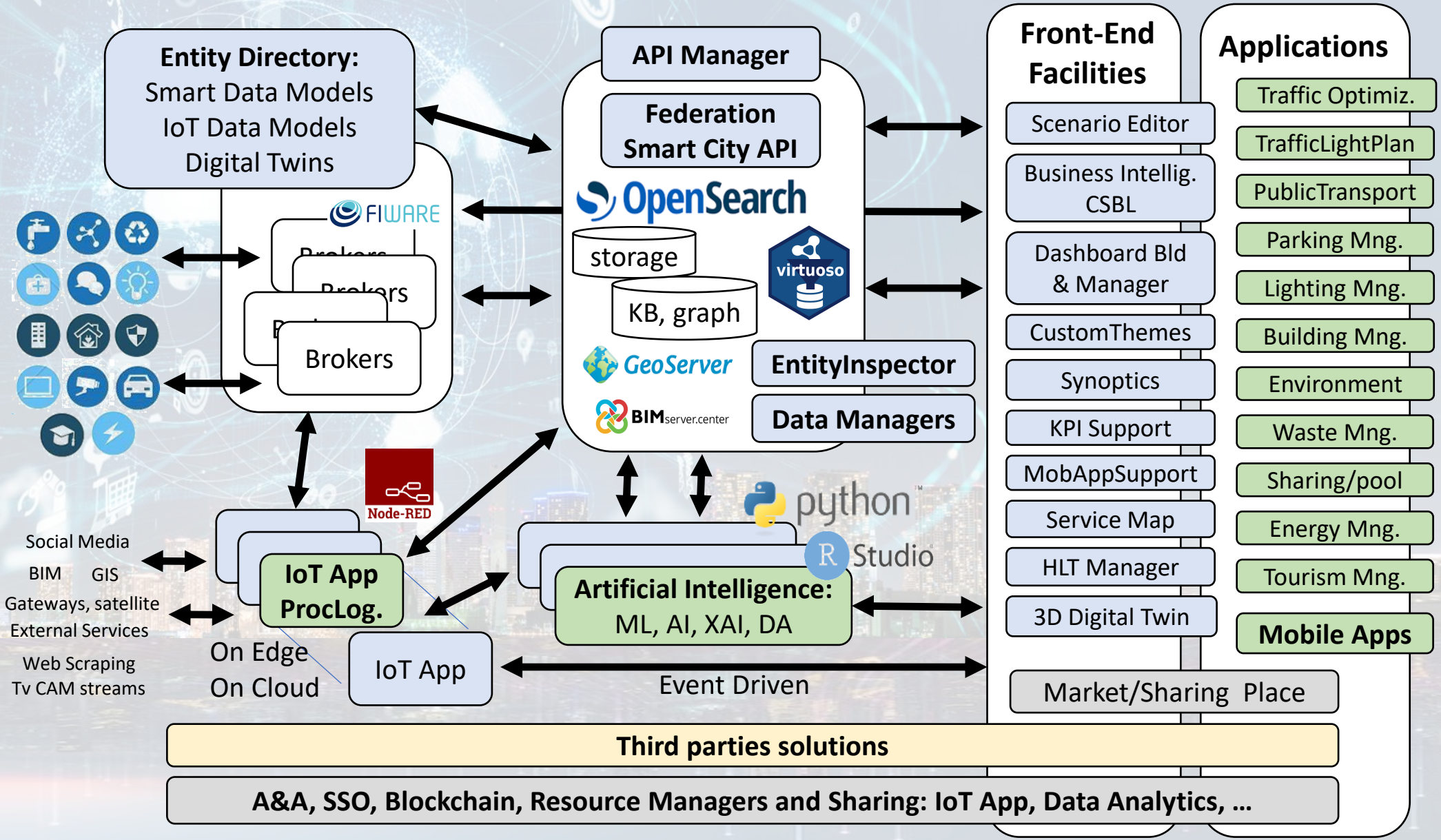




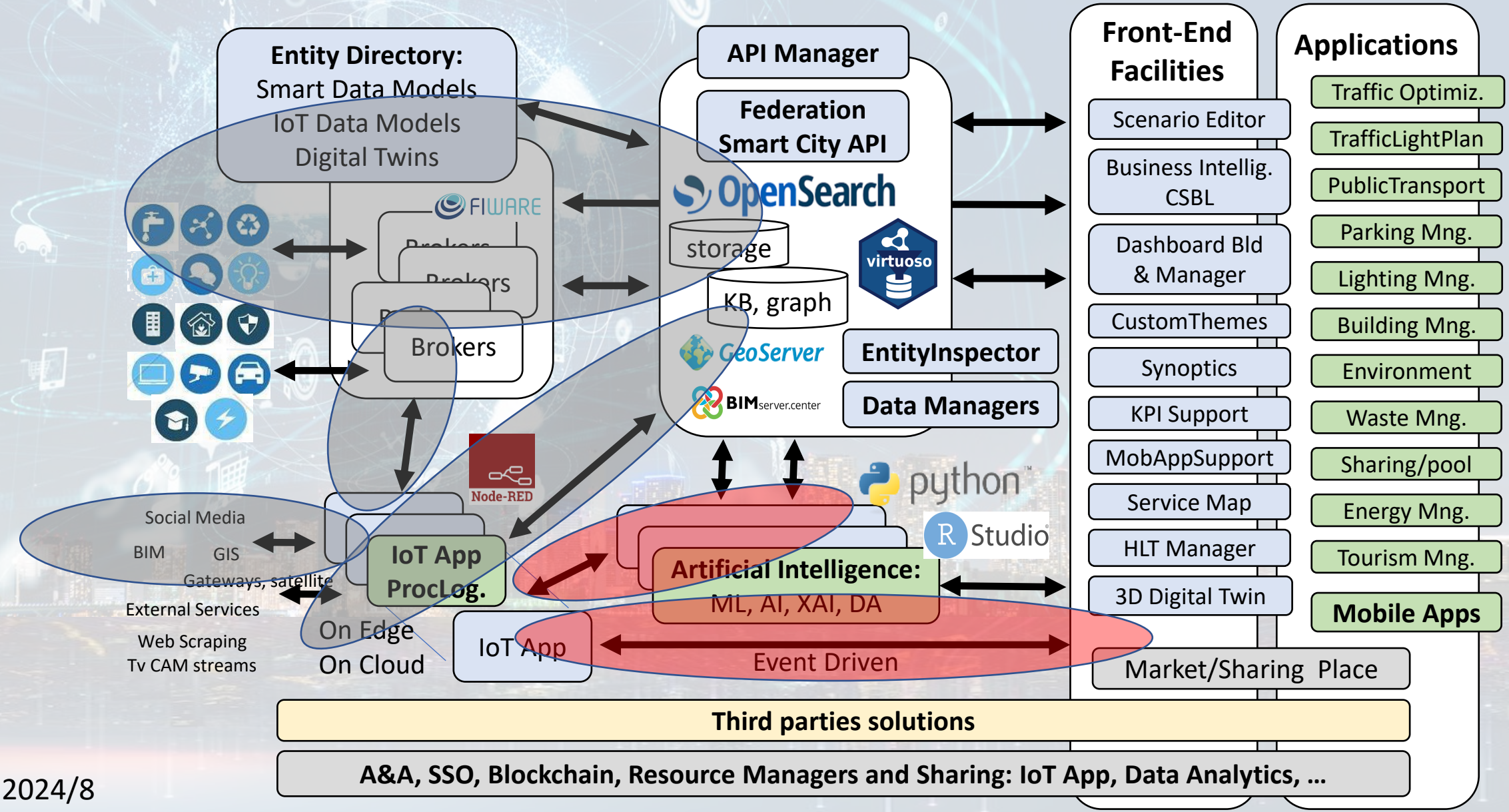
Digital Twin Development Platform



Technical Architecture



Technical Architecture





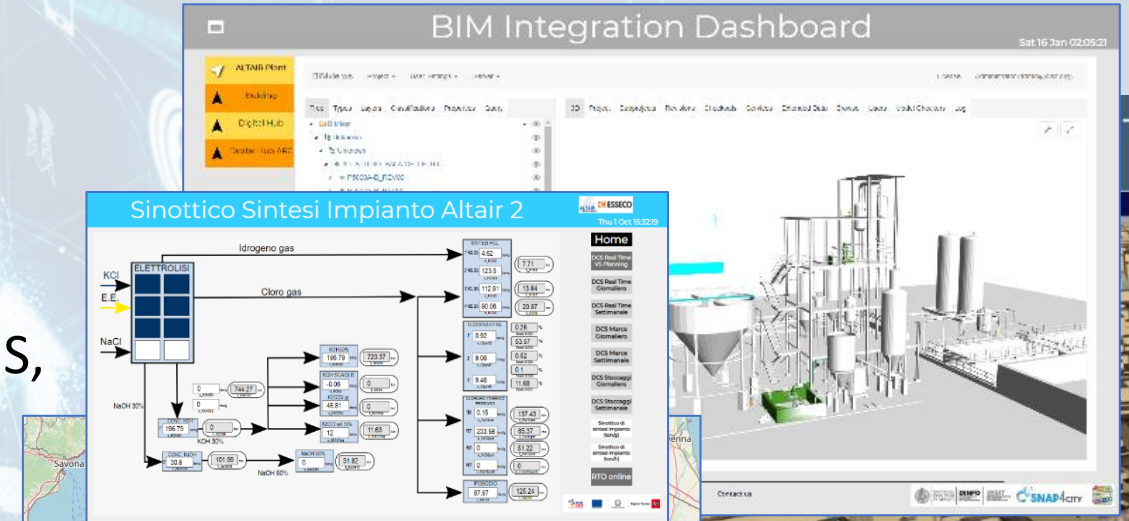
Data Ingestion Strategy and Orientation

Agenda of this integrated overview part

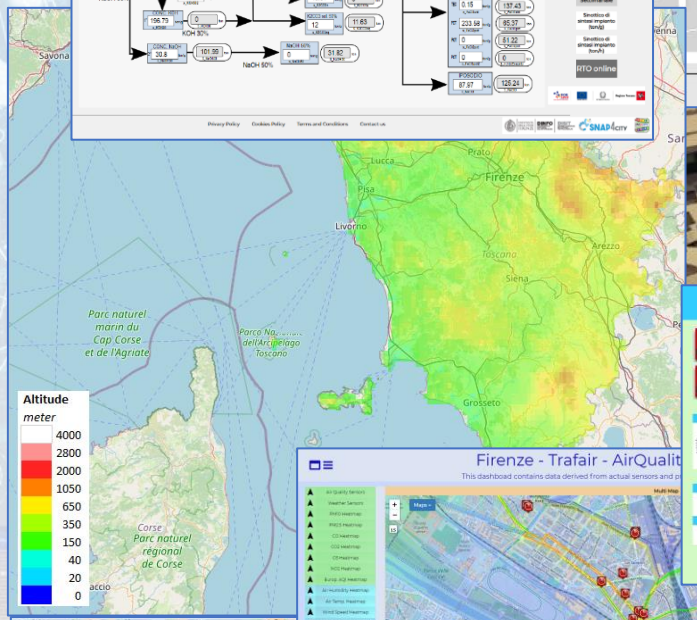
- Recall of Snap4City Architecture
- Data Ingestion Strategy and Orientation
  - Basic entity elements
  - Knowledge Base: Modelling and ServiceURI as Entity Identifier
  - Models vs Devices/Entities and Registration
- Develop: Data Processes Proc.Logic / IoT App
  - Proc.Logic = Node-RED + Snap4City
  - An Integrated Example for Time Series
  - Verification of Data Ingestion
  - Exploiting Storage data by using: IoT App / Proc. Logic
  - Interoperability with respect to Hardware staff
  - High Performance Ingestion
  - Interoperability of Snap4City Platform
- Some Applicative examples
- Training Material
- Acknowledgements

High Level Types

- POI, IOT Devices, shapes,..
  - FIWARE Smart Data Models,
  - IoT Device Models
- GIS, maps, orthomaps, WFS/WMS, GeoTiff, calibrated heatmaps, ..
- Satellite data, ..
- traffic flow, typical trends, ..
- trajectories, events, Workflow, ..
- 3D Models, BIM, Digital Twins, ..
- OD Matrices of several kinds, ..
- Dynamic icons/pins, ..
- Synoptics, animations, ..
- KPI, personal KPI,..
- social media data, TV Stream,
- routing, multimodal, constraints, ..
- decision scenarios, ….
- etc.



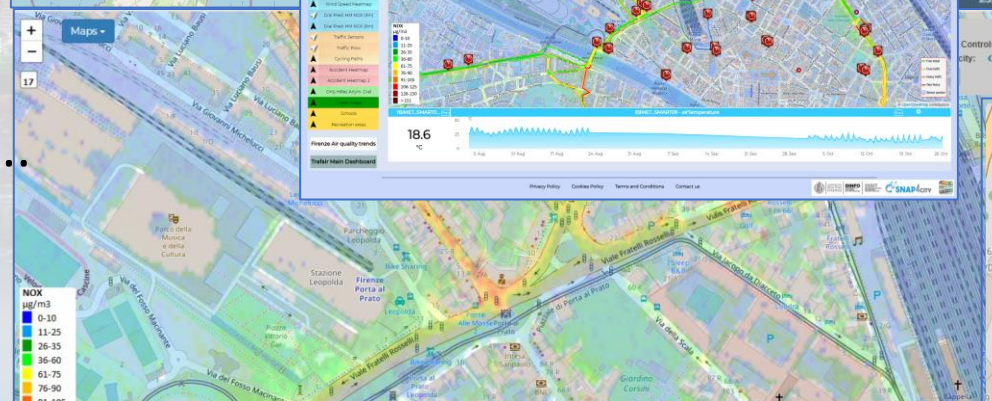


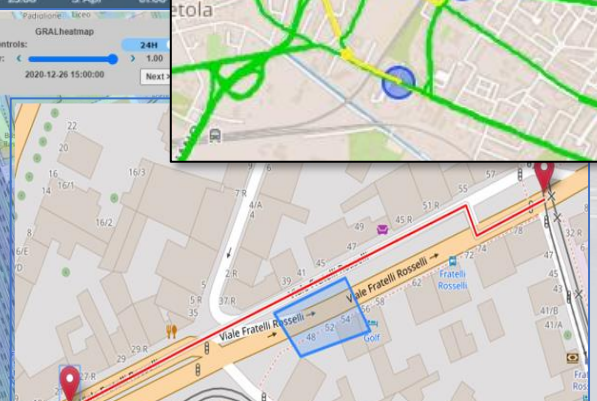












UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

Standards and Interoperability (10/2024)

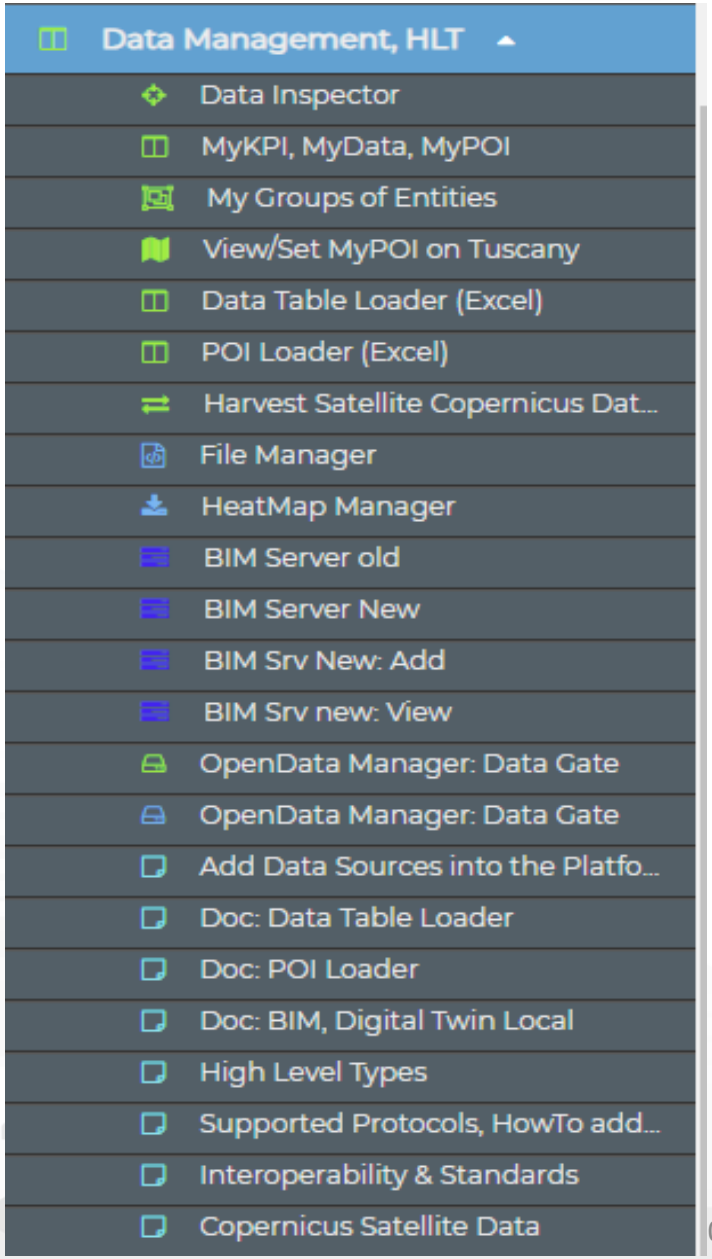
Compliant with:

- **IoT:** NGSI V2/LD, LoRa, LoRaWan, MQTT, AMQP, COAP, OneM2M, TheThingsNetwork, SigFOX, Libelium, IBIMET/IBE, Enocean, Zigbee, DALI, ISEMC, Alexa, Sonoff, HUE Philips, Tplink, BACnet, TALQ, Protocol Buffer, KNX, OBD2, Proximus, ..
- **IoT model:** FIWARE Smart Data Model, Snap4City IoT Device Models
- **General**: HTTP, HTTPS, TLS, Rest Call, SNMP, TCP, UDP, SOAP, WSDL, FTP, FTPS, WebSocket, WebSocket Secure, GML, WFS, WMS, RTSP, ONVIF, AXIS TVCam, CISCO Meraki, OSM, Copernicus, The Weather Channel, Open Weather, OLAP, VMS Milestone, TIM, HERE, ….
- **Formats**: JSON, GeoJSON, XML, CSV, GeoTIFF, OWL, WKT, KML, SHP, db, XLS, XLSX, TXT, HTML, CSS, SVG, IFC, XPDL, OSM, Enfuser FMI, Lidar, glTF, GLB, DTM, GDAL, Satellite, D3 JSON, …
- **Database**: Open Search, MySQL, Mongo, HBASE, SOLR, SPARQL, ODBC, JDBC, Elastic Search, Phoenix, PostGres, MS Azure, ..
- **Industry**: OPC/OPC-UA, OLAP, ModBUS, RS485, RS232,..
- **Mobility**: DATEX, GTFS, Transmodel, ETSI, NeTEx, ..
- **Social**:Twitter, FaceBook, Telegram, ..
- **Events**: SMS, EMAIL, CAP, RSS Feed, ..
- **OS**: Linux, Windows, Android, Raspberry Pi, Local File System, AXIS, ESP32, etc.

https://www.snap4city.org/65

Snap4City vs Formats

- ***Snap4City is capable to ingest*** and work with ***any format***:
  - Data **exchange**: JSON, GeoJSON, XML, HTML, HTML5, DATEX, GTFS, binary, etc.
  - **GIS formats:** WMF, WFS, heatmaps, ….
  - **Table**: CSV, XLSX, XLS, database, …
  - **Road graphs:** OSM, triples, geoJSON, etc.
  - **graphics:** IFC, Shape, WKT, SVG, …
  - **archive** file formats: zip, rar, 7z, tgz, pdf, …
  - **image** formats: png, gif, tiff, geoTiff, ico, jpg, …
  - **ODM**: JSON and other formats
  - **Traffic Flow**: JSON and other formats
  - **Heatmaps**: GeoTIFF, JSON, etc.
  - **video** formats: mp4, avi, mov, RTSP, …
  - **3D elements**: GLB, DWG, IFC, etc.
- Search the format you need to cope on the search box of Snap4City portal!: **Snap4City Supported Protocols, adding new protocols**
  https://www.snap4city.org/65

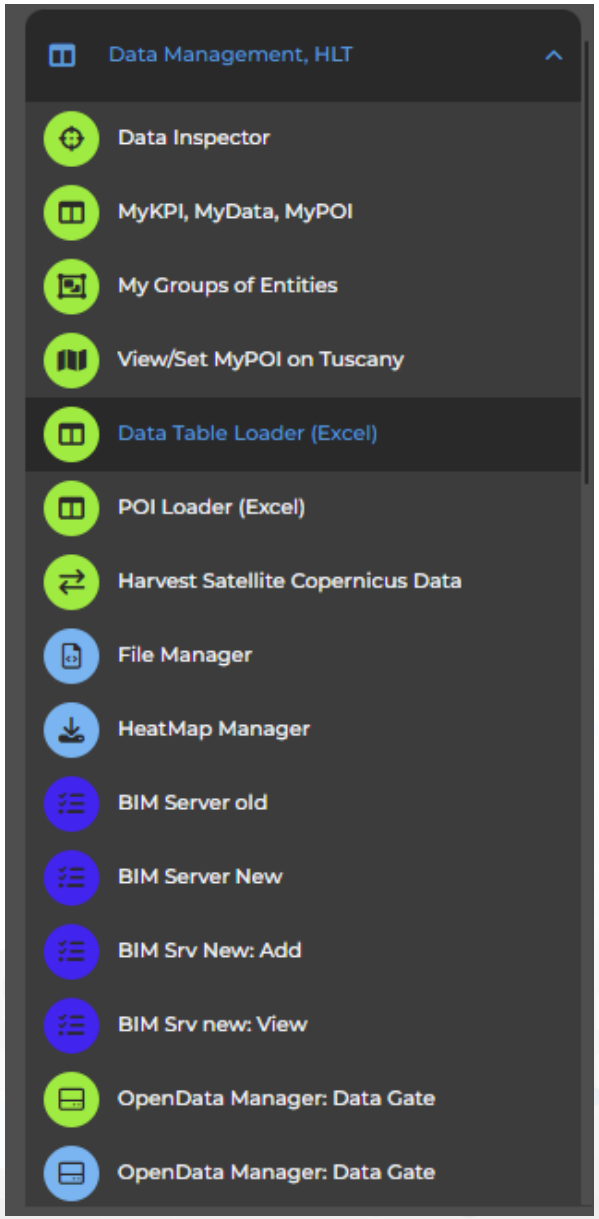


The main High Level Types

- **POI**: codified metadata, static GPS, + info, no time series
- **Entity Instance / IoT Device**: static GPS, Info, variable data, Time Series
  - Sensors and actuators
  - **Entity Mobile / IoT Device Mobile**: if dynamic GPS
- **MyKPI**: dynamic GPS, - info, single variable, Time Series
- ***Heatmaps**: matrices on some area, Time Series → **Entities***
- ***Traffic Flows**: road segments with flow density, Time Series → **Entities***
- ***OD Matrices:** different parameters, Time Series → **Entities***
- ***Vector Fields**: different parameters, Time Series, may be with Heatmaps*
- ***Scenarious:** different parameters, Time Series → **Entities***
- ***Routing, paths, trajectories, etc.…** different parameters,*
- ***3D shapes, elements, BIMs, TV Cam, ….** different parameters, **Entities***
- ***….***

HLT wrt to Time Series, GPS and Geometry

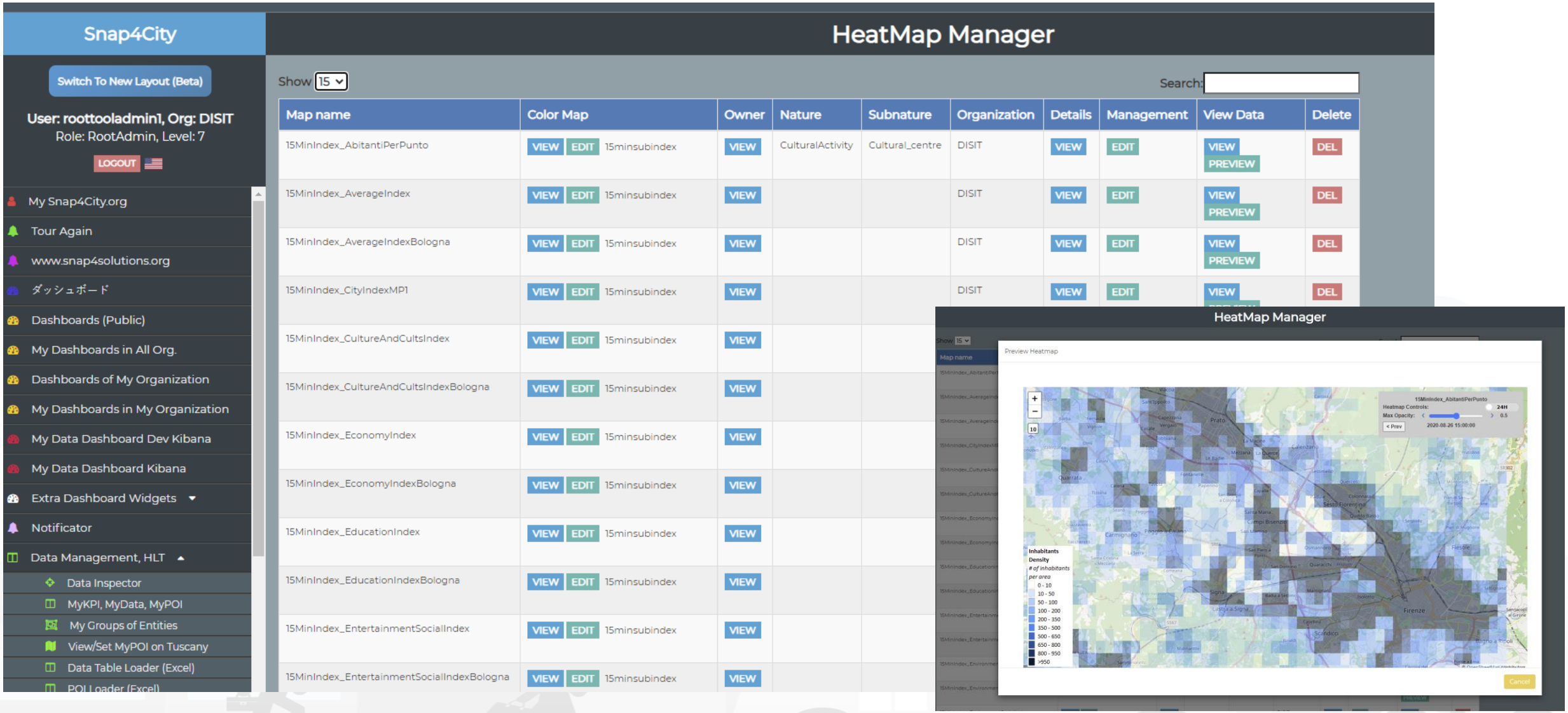
High Level Types	Evolution over time ?	May have GPS/Geom
POI, Point of Interest	**Sporadically, for versioning**	*Yes/Yes*
IoT Devices, KPI multivariable, WoT Entities	A set of values for each time instant of observation (dateObserved)	*Yes/Yes*
GIS data	**Sporadically, for versioning**	*Yes/Yes*
Satellite Data	An image for each time instant of observation (dateObserved)	*Yes/Yes*
Traffic Flow	A Traffic Flow network for each time instant of observation (dateObserved)	*Yes/Yes*
Heatmaps	An Heatmap for each time instant of observation (dateObserved)	*Yes/Yes*
OD Matrices	An ODM for each time instant of observation (dateObserved)	*Yes/Yes*
BIM Models	**Sporadically, for versioning**	*Yes/Yes*
MyKPI (single variable)	A value for each time instant of observation (dateObserved)	*Yes/no*
3D model data	**Sporadically, for versioning or for model kind for example**	*Yes/Yes*
Messages exchanged with Dashboards	*A set of values for each time instant of observation (dateObserved)*	*No/No*
Messages exchanged with Synoptics	*A set of values for each time instant of observation (dateObserved)*	*No/No*



Data Management, HLT

- Data Inspector
- MyKPI…..
- My Groups of Entities
- HeatMap Manager
- BIM Server…….
- Open Data…
- Depending on user's kind:
  - ODM, File, TV CAM, Traffic Flow, …..

HeatMap Manager



TrafficFlow Manager

Snap4City

Switch To New Layout (Beta)

User: roottooladmin1, Org: DISIT
Role: RootAdmin, Level: 7

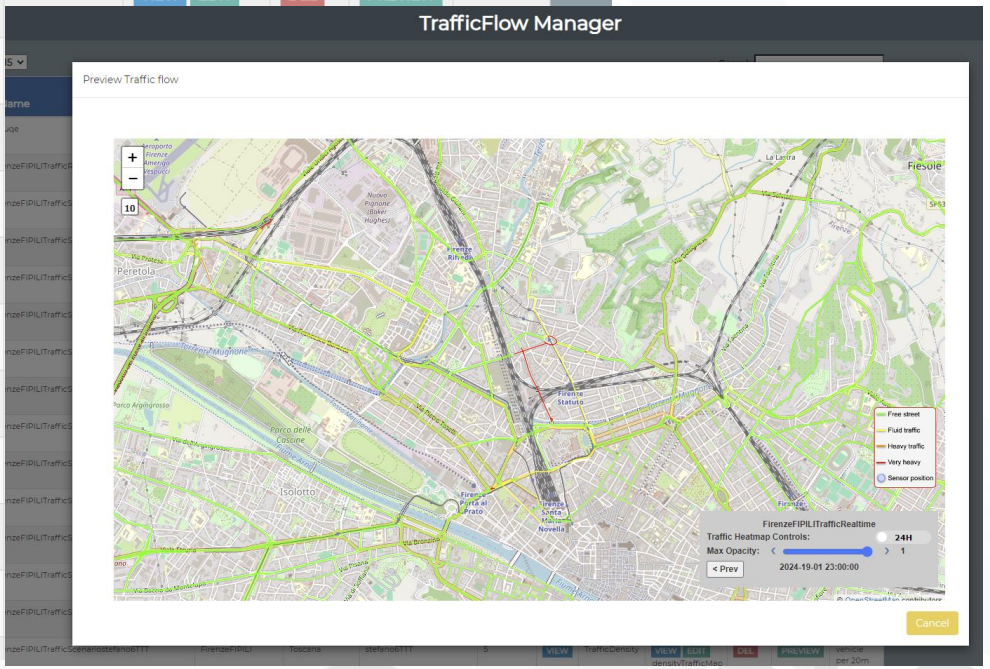
LOGOUT

- Data Management, HLT
  - Data Inspector
  - MyKPI, MyData, MyPOI
  - My Groups of Entities
  - View/Set MyPOI on Tuscany
  - Data Table Loader (Excel)
  - POI Loader (Excel)
  - Harvest Satellite Copernicus Data
  - Data Inspector OpenSearch
  - File Manager
  - HeatMap Manager
  - ColorMap Manager
  - TypicalTimeTrend Manager
  - TrafficFlow Manager
  - TVCam Manager
  - OD Manager
  - BIM Manager
  - BIM Server old
  - BIM Server New
  - BIM Srv New: Add
  - BIM Srv new: View
  - OpenData Manager: Data Gate
  - OpenData Manager: Data Gate
  - OpenData Harvester: Data Gate
  - Add Data Sources into the Platform
  - Doc: Data Table Loader

TrafficFlow Manager

Show 15 Search:

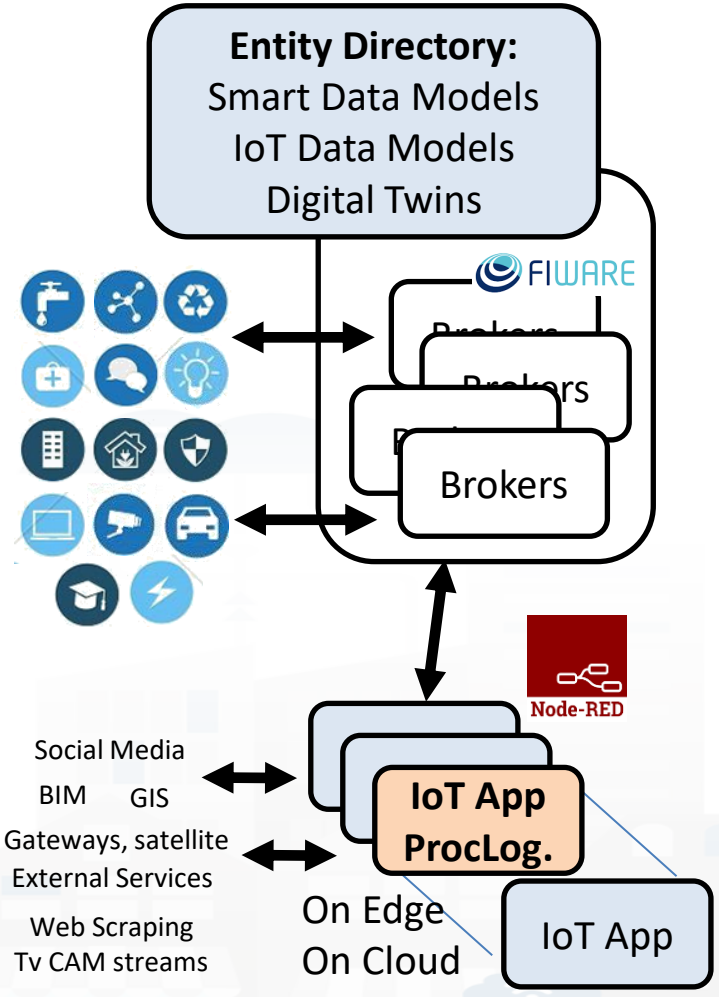
Flux Name	Locality	Organization	Scenario	Instances	View Data	Metric	ColorMap	Delete	Preview	Unit of Measure
cinuqe	nomedelloscenario	Toscana	cinuqe	3	VIEW	TrafficDensity	VIEW EDIT densityTrafficMap	DEL	PREVIEW	vehicle per 20m
FirenzeFIPILITrafficRealtime	FirenzeFIPILI	Toscana	TrafficRealtime	114241	VIEW	TrafficDensity	VIEW EDIT densityTrafficMap	DEL	PREVIEW	vehicle per 20m
FirenzeFIPILITrafficScenarioScenario1before	FirenzeFIPILI	Toscana	Scenario1before	1	VIEW	TrafficDensity	VIEW EDIT densityTrafficMap	DEL	PREVIEW	vehicle per 20m
FirenzeFIPILITrafficScenarioScenarioAFeb2020	FirenzeFIPILI	Toscana	ScenarioAFeb2020	14	VIEW	TrafficDensity	VIEW EDIT	DEL	PREVIEW	vehicle
FirenzeFIPILITrafficScenarioScenarioAFeb2020TTT	FirenzeFIPILI	Toscana	ScenarioAFeb2020TTT	14	VIEW					
FirenzeFIPILITrafficScenarioScenarioANov2019	FirenzeFIPILI	Toscana	ScenarioANov2019	14	VIEW					
FirenzeFIPILITrafficScenarioScenarioANov2019TTT	FirenzeFIPILI	Toscana	ScenarioANov2019TTT	14	VIEW					
FirenzeFIPILITrafficScenariostefano2	FirenzeFIPILI	Toscana	stefano2	4	VIEW					
FirenzeFIPILITrafficScenariostefano2TTT	FirenzeFIPILI	Toscana	stefano2TTT	4	VIEW					
FirenzeFIPILITrafficScenariostefano3	FirenzeFIPILI	Toscana	stefano3	7	VIEW					
FirenzeFIPILITrafficScenariostefano3TTT	FirenzeFIPILI	Toscana	stefano3TTT	7	VIEW					
FirenzeFIPILITrafficScenariostefano5	FirenzeFIPILI	Toscana	stefano5	8	VIEW					
FirenzeFIPILITrafficScenariostefano5TTT	FirenzeFIPILI	Toscana	stefano5TTT	8	VIEW					
FirenzeFIPILITrafficScenariostefano6	FirenzeFIPILI	Toscana	stefano6	5	VIEW					



SubStantially

- **Entities** are used to model Digital Twins and thus:
  - POI, MyPOI, KPI, MyKPI, IoT Devices, etc.
  - **including: metadata info, time series, GPS position, geometries, hyper Links, and Actions which are links to tools/functions/processes to Act on them**
- **Complex Data** such as used to model spec. aspects as Traffic Flow, Heatmaps, ODM, BIM, TV cameras, 3D elements, .. :
  - Are placed on map as an Entity plus **additional information** into a dedicated <data> Manager
  - **Entity includes**: **metadata info, time series, GPS position, geometries, hyper Links, and Actions which are links to tools/functions/processes to Act on them**

Data Ingestions Strategy



1

1) Via ***internal Brokers***: NGSI V1/V2, MQTT (beta), JSON

- the data messages arriving in **PUSH** on the platform,
  - if compliant to a known Model **and** referring to a known Device/Entity
  - They are AUTOMATICALLY: *stored, indexed,* and ready to be used by Wizard, Dashboards, Views and Applications

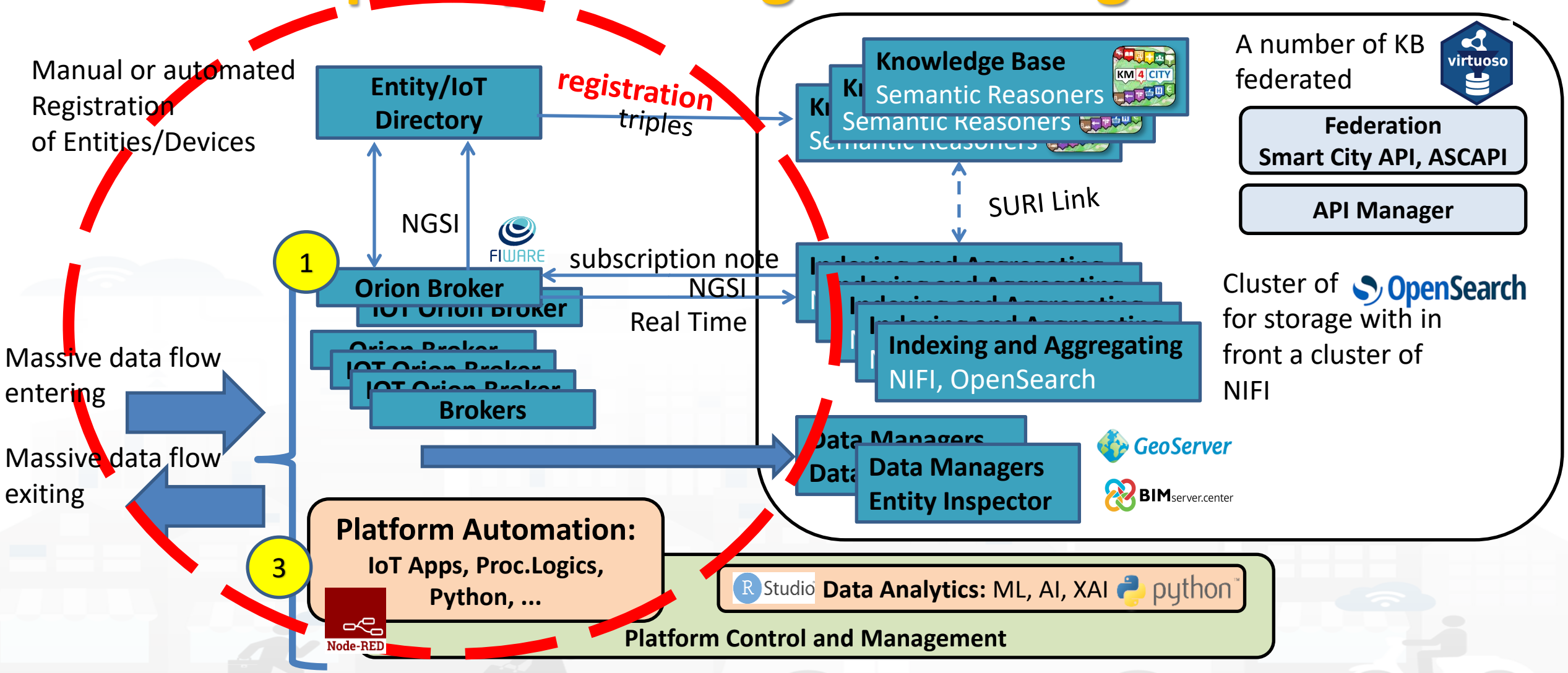
- 2) Via ***external Brokers***: NGSI V2, NGSI LD (beta), JSON
  - the data messages arriving in **PUSH** on the external broker,
    - Have to be **mapped** to a known Model **and** referring to a known Device/Entity
    - To be AUTOMATICALLY: *stored, indexed,* and ready to be used by Wizard, Dashboards, Views and Applications

3

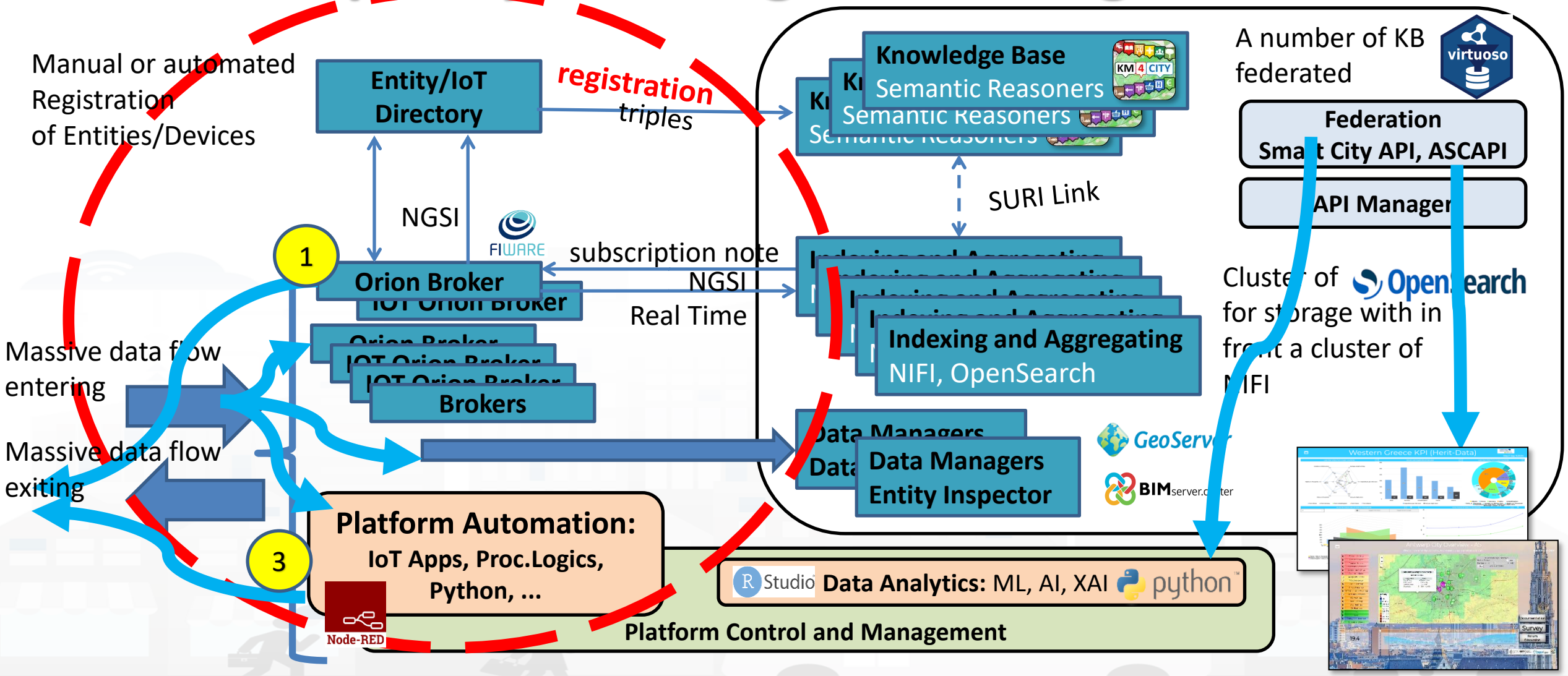
3) Via **any other broker, GateWay, API, server, WebService, database, protocol and/or format of the many available on Snap4City, in PUSH/PULL have to be**

- **mapped** on an Snap4City IoT/Entity Model and Entity/Device Instance
- **ingested in real time or batch**, by a process implemented as
  - an **IoT App/Proc.Logic, on container, on cloud or Edge**
  - a **Python or other language**, **on container, on cloud or Edge**

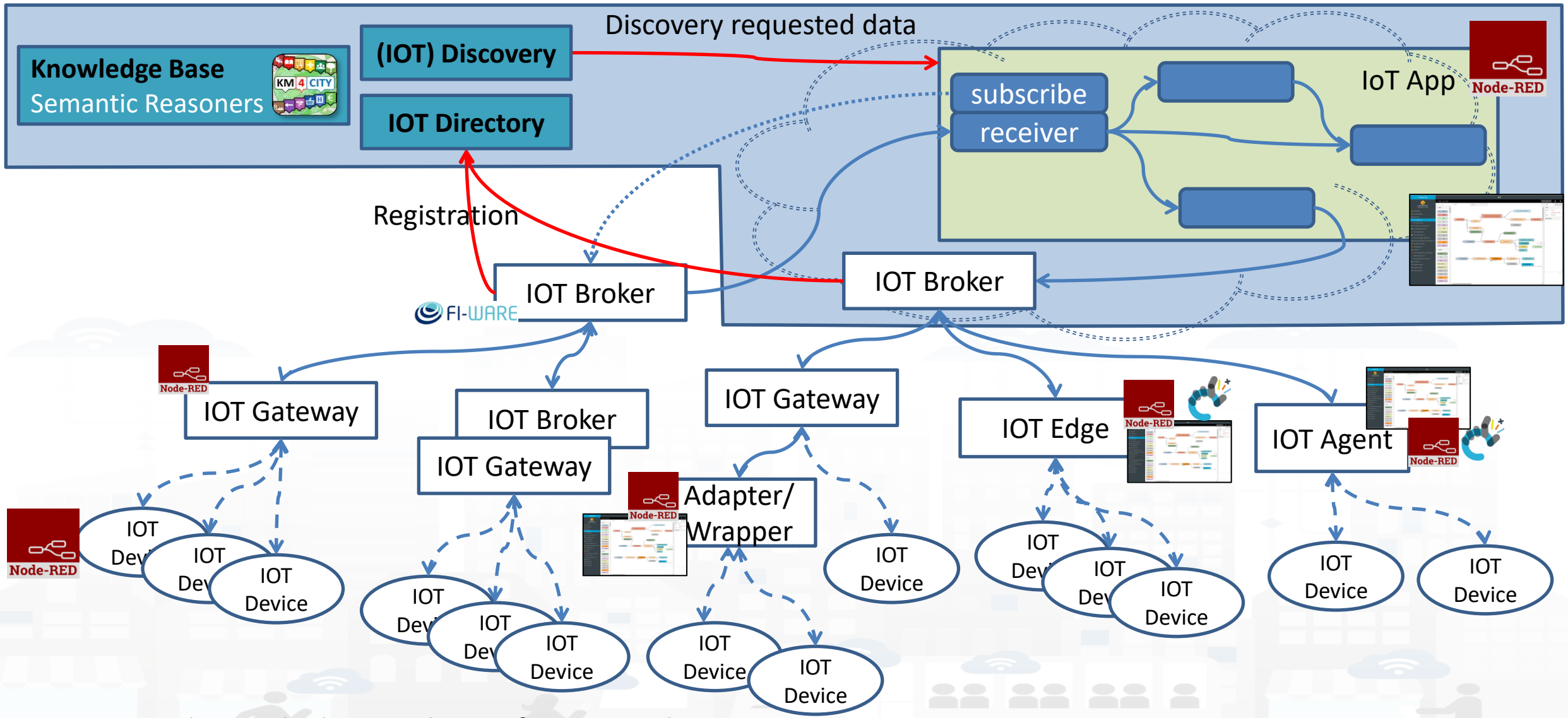
Snap4city Data Ingestion Diagram



Snap4city Data Ingestion Diagram



IoT Network



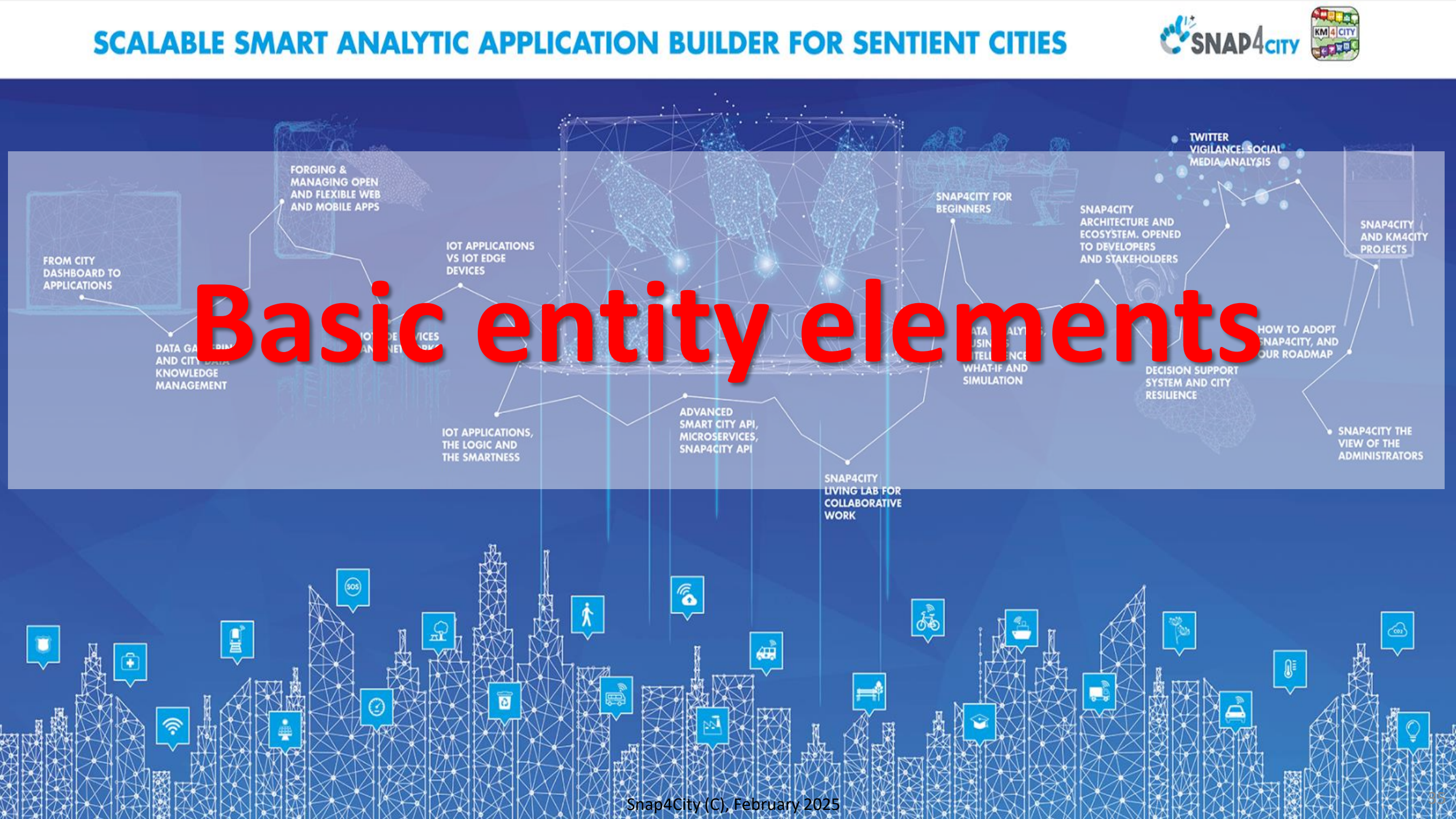
Snap4City make simple the complexity of IoT networks

Basic Concepts of Data Ingestion for 1 3

- **Basic entity elements**
  - Static and Real-Time / Time Series
  - Classification of Entities
  - References among Entities
- **Knowledge Base: Modelling and ServiceURI as Entity Identifier**
  - Entity Indexing and Unified Identifications
- **Models vs Devices/Entities and Registration**
  - Entity Models / IoT Device Models
  - Entity Instanced / IoT Device Instances
  - Messages over time/version
  - Variable as Attributes

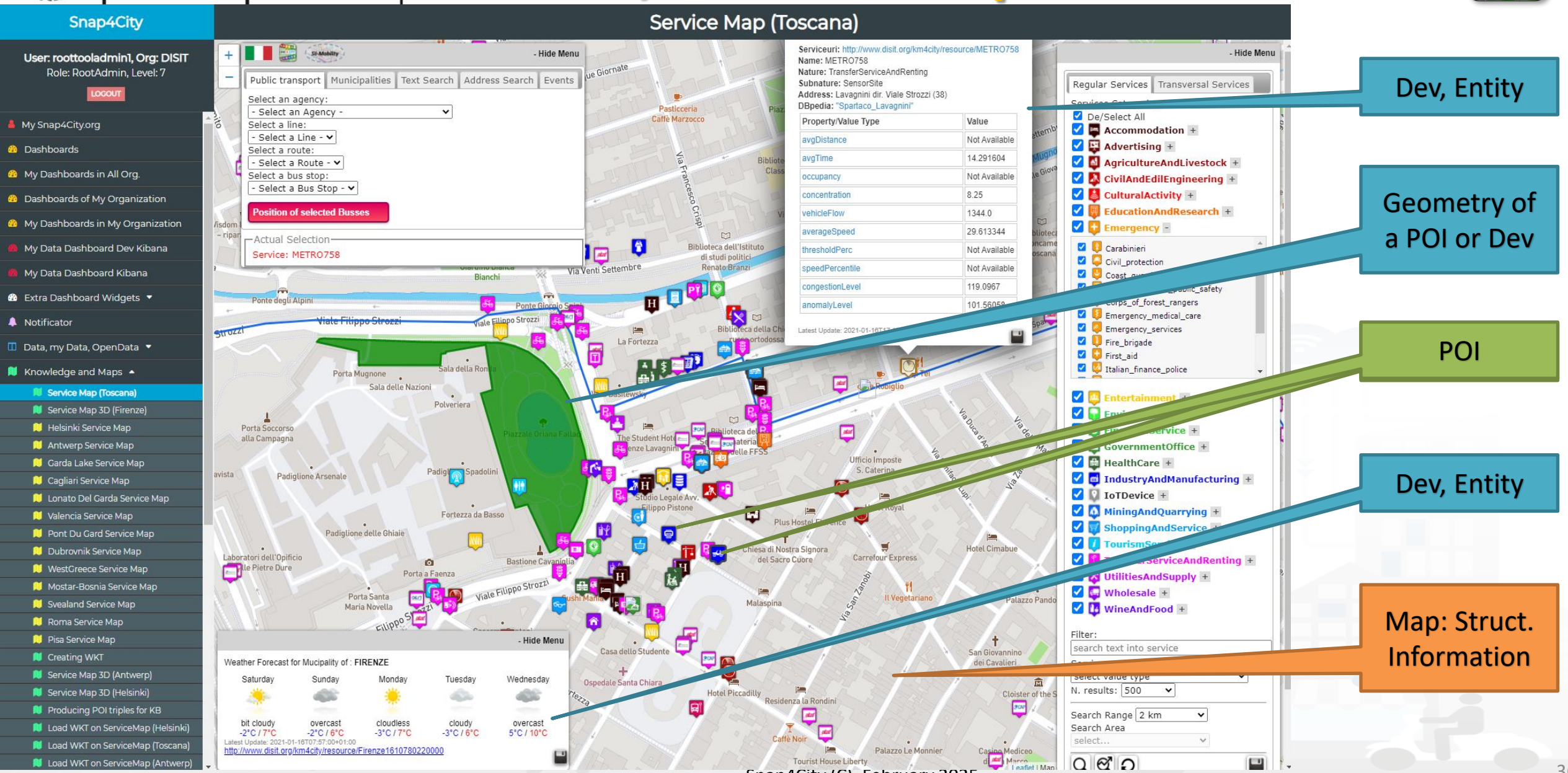
Terminology

Former name	new name, from 2023	What
IoT Device Model	Entity Model	A data model
IoT Device	Entity Instance	A data instance ready to get message for time series
IoT Device Variable, metric	Entity Variable	A variable of an Entity Instance or of an Entity Model
IoT Device Message, device message	Entity Message	A data message
IoT Directory	Entity Directory	The tool for managing models, entities, data models, etc.
IoT Applications, **IoT App**	Processing Logic	= Node-RED + Snap4City Libraries The tool for visual programming, node-red JavaScript, data flow, ingestion logic, data transformation, data loading, interoperability, business logic.
Dashboards	Views and Dashboards	The Snap4City Dashboards are effectively Views of some Web Application, with all the interaction and connection the developer would create among them.
LOG	LOGraph	not all Snap4City platform are provided with the LOGraph, it is optional and can be installed in a second phase



Basic entity elements

KB, ServiceMap



Dev, Entity

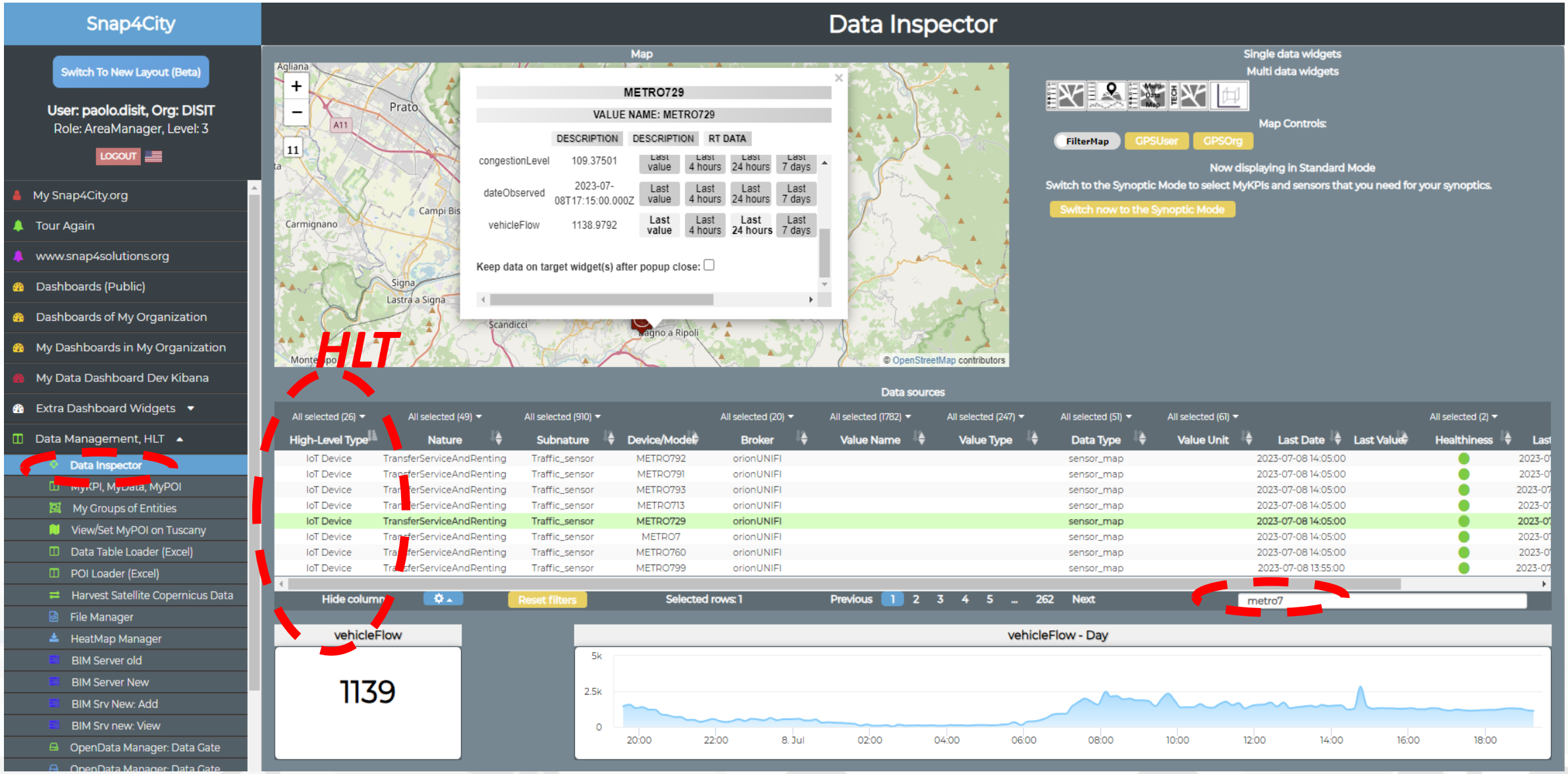
Geometry of a POI or Dev

POI

Dev, Entity

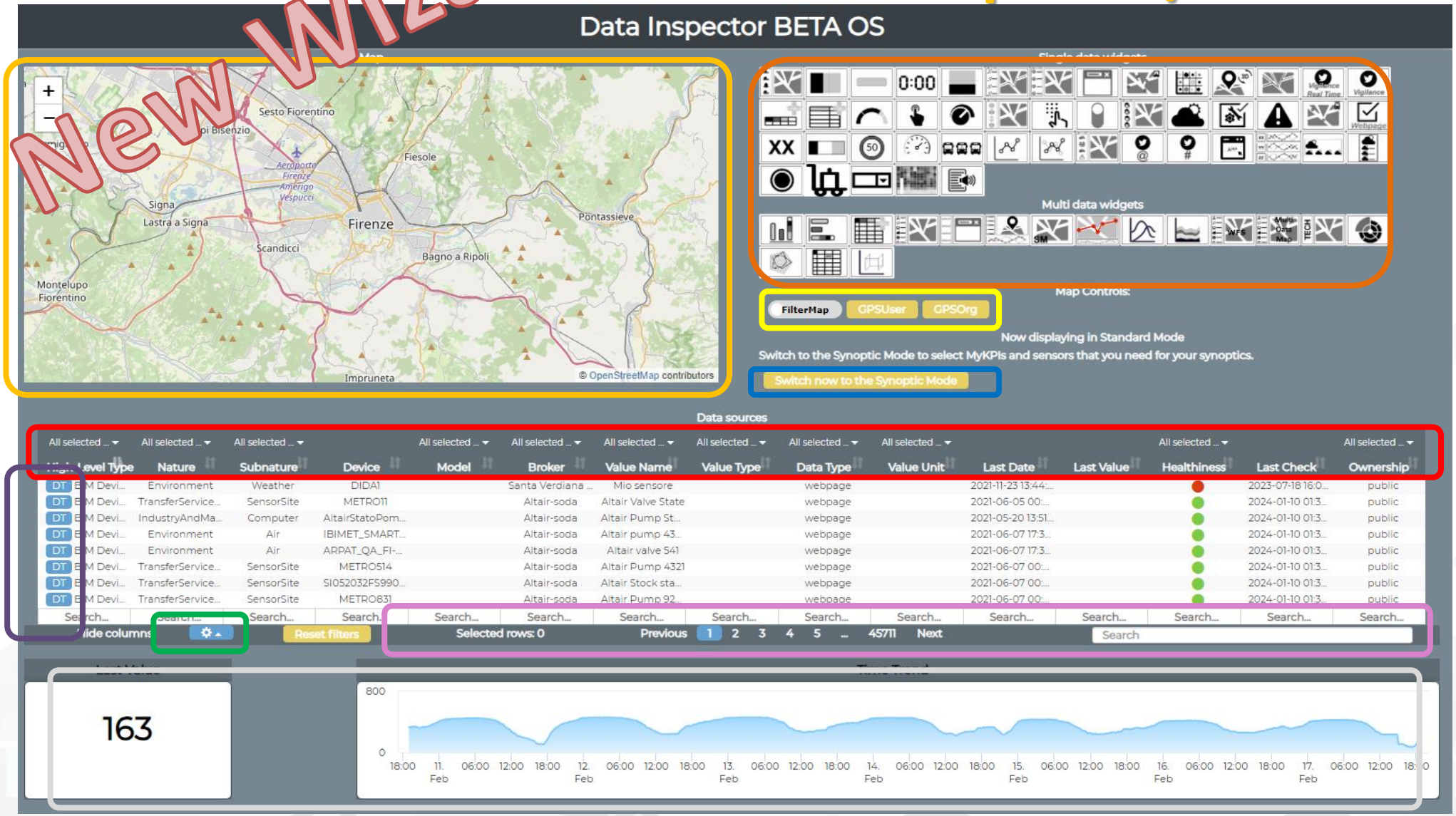
Map: Struct. Information

Data Inspector



New Data Inspector/Wizard

New Wizard



Filtering/Searching for individual fields (even for some fields not displayed as geographic coordinates)

Geographic Filtering

Text Search on all fields

Menu for choosing the fields to display in the table

View on Map(via PREVIEW)

Data and Trend visualization

Opening Digital Twin

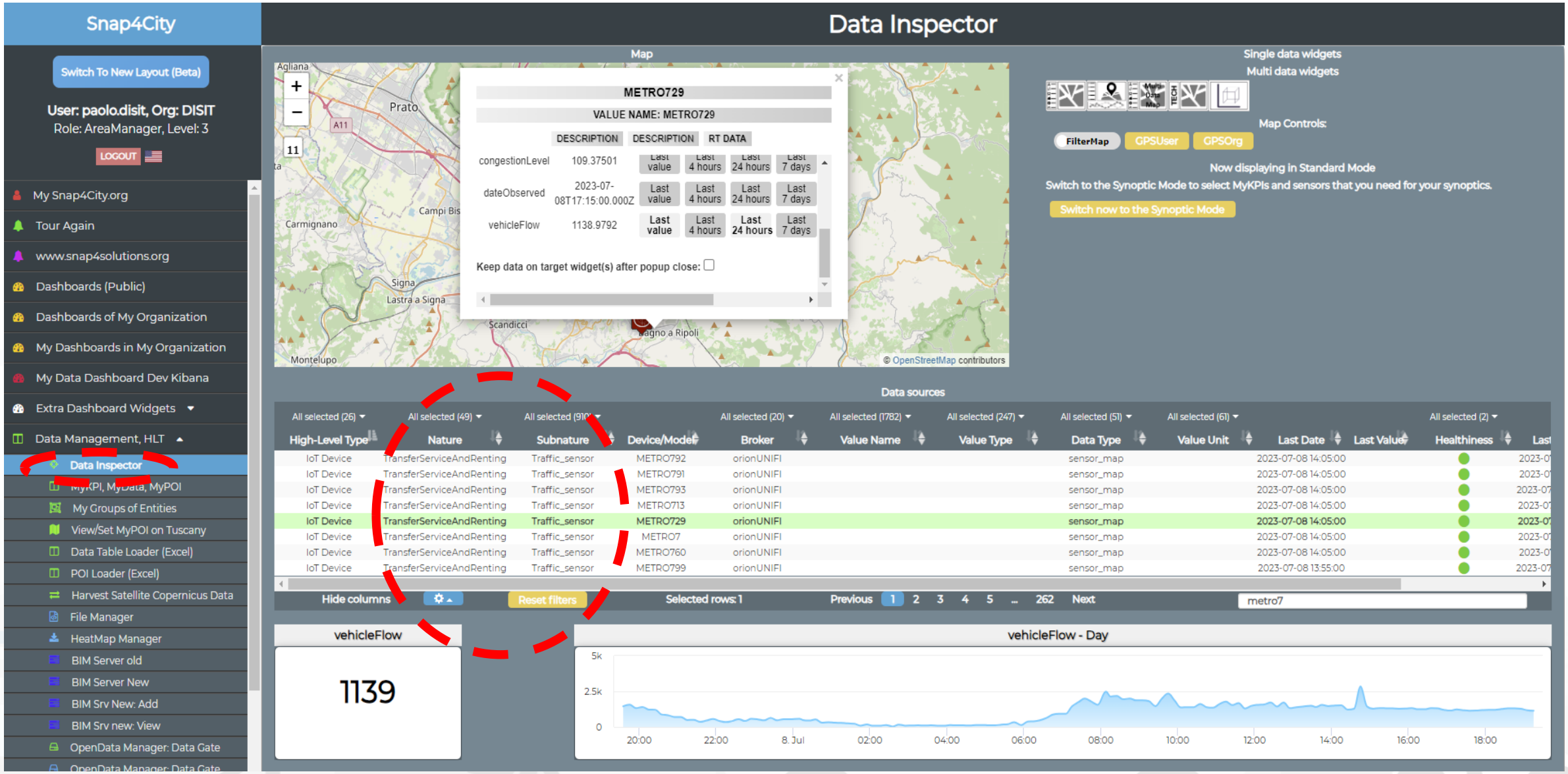
Pass to Synoptic mode

Select the graph representation

Classification by Nature/SubNature
Semantic Classification of Entities



Data Inspector



Any Entity has a Semantic *Classification*

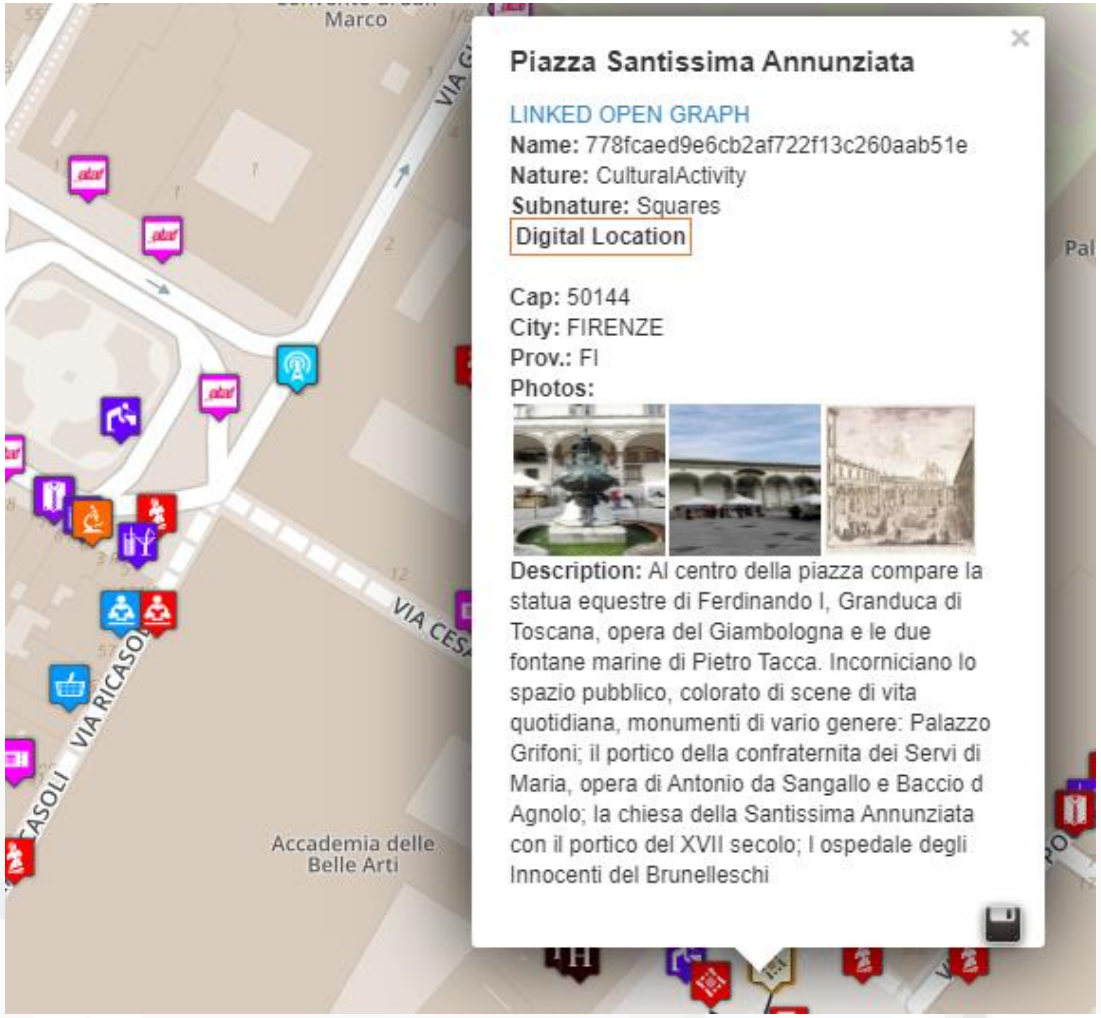
Nature

- Accommodation +
- Advertising +
- AgricultureAndLivestock +
- CivilAndEdilEngineering +
- CulturalActivity +
- EducationAndResearch +
- Emergency +
- Entertainment +
- Environment +
- FinancialService +
- GovernmentOffice +
- HealthCare +
- IndustryAndManufacturing +
- IoTDevice +
- MiningAndQuarrying +
- ShoppingAndService +
- TourismService +
- TransferServiceAndRenting +
- UtilitiesAndSupply +
- Wholesale +
- WineAndFood +

SubNature

EducationAndResearch -

- Educational_support_activities
- Higher_education
- Language_courses
- Performing_arts_schools
- Post_secondary_education
- Pre_primary_education
- Primary_education
- Private_high_school
- Private_infant_school
- Private_junior_high_school



SKOS

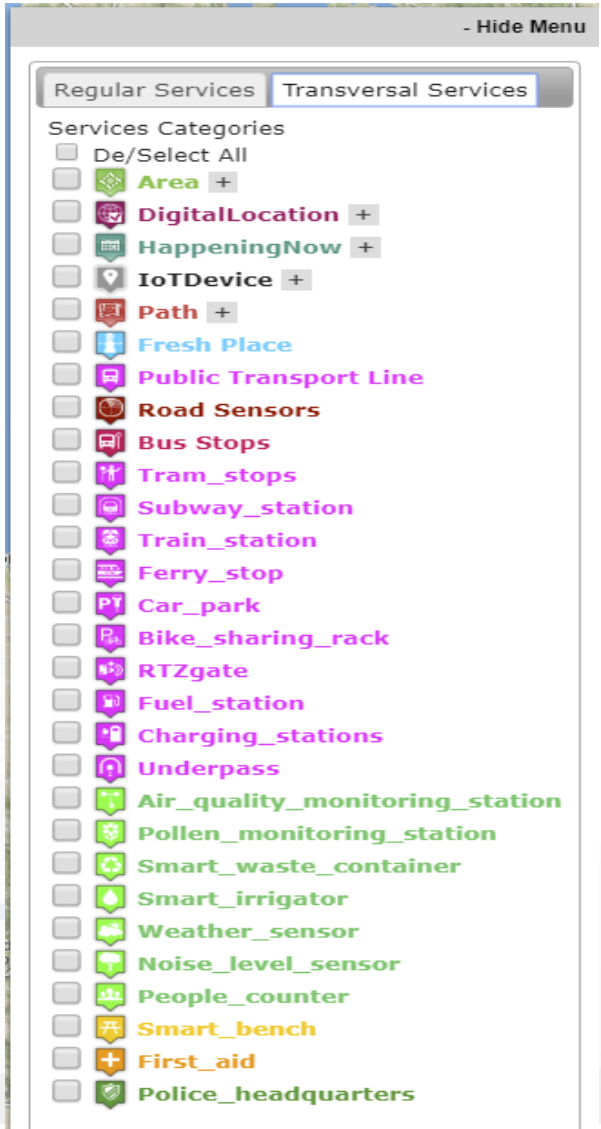
Point of Interests, POI

mainly static data



Access to Point of Interest information, POI

- **POI**: point of interest
- **type**: macro (nature) and subcategories (subnature)
- **Position**: GPS, address, telephone, fax, email, URL, …
- **Description**: textual, multilingual, with images, …
- **Link** to dbPedia, Linked Open Data
- **Links to other services**
  - **Not Real time data if any, please use Entities / devices connected:** sensors data, timeline, events, prices, opening time, rules of access, status of services, status of queue, etc..
- *See transversal services on ServiceMap*
  - *Regular and in test platform*



POI, Point of Interest

- They are
  - classified in terms of nature/subnature
  - relevant services with codified metadata to simplify the massive management of huge amount of POIs
  - mapped on Knowledge Base on **specific GPS location**
  - Do not move over time
  - represented as PIN
- Do not have Time Series for variable over time
- May sporadically change over time





Time Series can be:
IoT Devices, MyKPI, Entities, etc.



What About a Time Series

IOT Device



Entity: IOT Device

Sends a message

Message (
timestamp: 02-04-2020 at 10:30,
Temperature: 29.34,
Humidity: 35
)

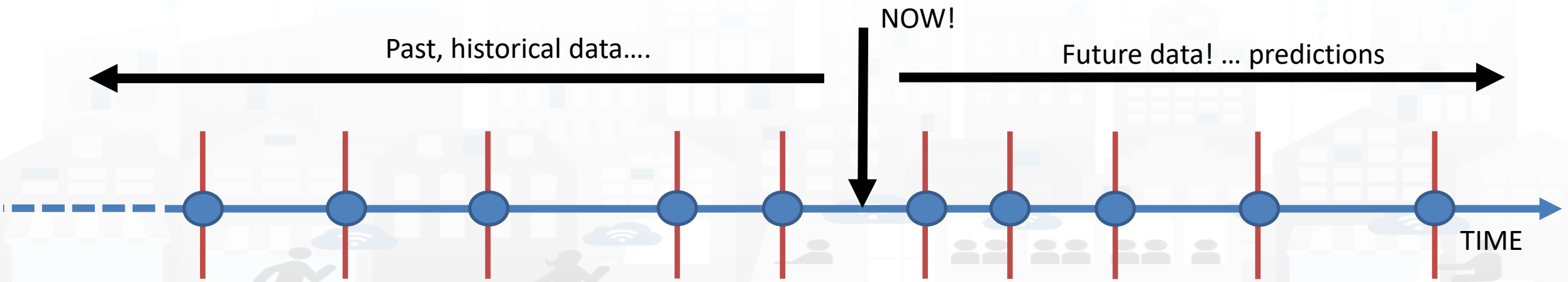
dateObserved	Temp	Humidity
02-04-2020 10:30	34.5	23
02-04-2020 10:40	36.5	24
02-04-2020 10:50	36.0	22.5

- A set of data coming from an IoT Device with multiple sensor become a time series of values for devices.
  - For example: taking a new measure every 10 minutes (Red Lines)
  - Non regular rates can be valid data as well.
- Each new measure in Snap4City is conventionally time located in «dateObserved», which has to be **Unique.**
  - **Only one message per dateObserved is allowed**



Time Series: they are data streams

- As soon as you have a variable changing over time → time series
  - You are ready to get Future data, may be arriving in PUSH
  - Recall and store historical data as well, but they have to be
    - recalled in PULL with some IoT App/Proc.Logic
    - Loaded in PULL with some File or Data Table Loader



A time series

The messages posted on Entity Instances / IoT Devices can produce different effects on time series.

Omitting the message would allow the broker to reuse the last data to fill it, as for V5, which appear

- valid in all messages on graphs
- With holes in tables

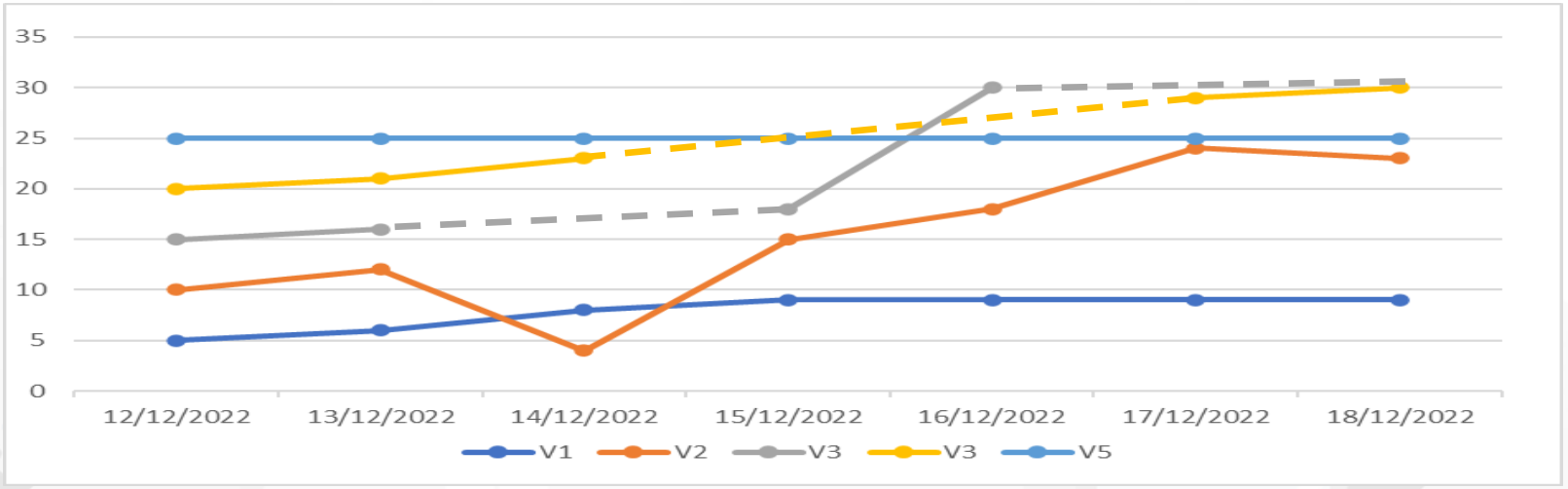
Putting null values (as in V3) would produce a missing data and thus would lead to create:

- interpolate line on graphs: dashed are actually continuous lines in Dashboards
- Empty values in the tables

device42 Entity Messages over time

Color for missed

12-12-2022	13-12-22	14-12-22	15-12-22	16-12-22	17-12-22	18-12-22
V1: 5	5	6	8	9	9	9
V2: 10	10	12	4	15	18	24
V3: 16	15	16	null	18	30	null
V4: 20	20	21	23	null	null	29
V5: 25	25	25	25	25	25	25



IoT Devices, Entities Instances, WoT



IoT Device Models







IoT Device

- Name: CM23
- Model: Lavazza
- Position: …….

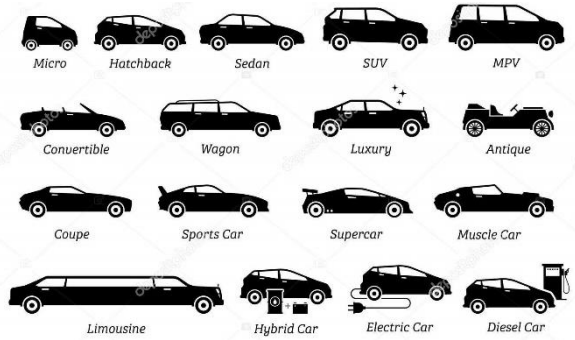
IoT Device Variables

- dateObserved: ……..
- ID: CM23
- Status: ready
- Temperature: 70°
- WaterLevel: 35%
- UsedCapsBox: 30%
- Power: OK
- …..

- Conceptually are Devices with sensors/actuators, IN/IN-OUT
- They are classified in terms of nature/subnature
- For Searching and showing on maps and dashboards
  HLT of Devices/Entities can be:
  - **(IoT Device) Entity Models**, for example: «personal coffee machine»
  - **Entity** name, for example: «mycoffemachine1», «CM23»
  - **Entity Variable**, for example: «Temperature»



Mobile Devices/Entities



Mobile Entity Models



Mobile Entity/device

- Name: BMWJD7356HD
- Model: BMW 318
- Spec:…

Mobile Entity/device Message

- ID: BMWJD7356HD
- **dateObserved: ……..**
- Status: ready
- Temperature: 70%
- Gasoline: 35%
- Velocity: 231,3 Km/h
- **Position: 44.3223, 11.3432**
- …..

- They are a special case of devices/entities
  - they are managed as Devices/Entities in the system
- They are classified in terms of nature/subnature
- For Searching and showing on maps and dashboards, they are different
  HLT of Mobile Devices/entities can be:
  - **Mobile Entity Model**, for example: «sedan»
  - **Mobile Entity Instance** name, for example: «BMW JD7356HD», «Ford KO786KK»
  - **Mobile Entity Variable**, for example: «velocity»



Sensor/Sensor-Actuator





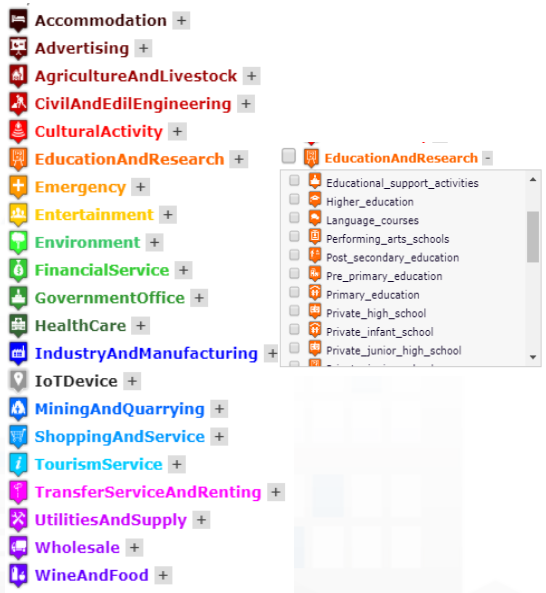


IoT Device

- Name: CM23
- Model: Lavazza
- Position: …….

Sensors

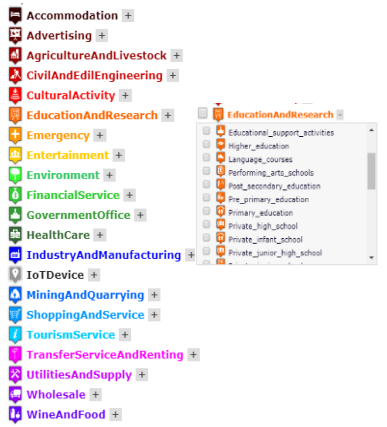
- **dateObserved: ……..**
- ID: CM23
- Status: ready
- Temperature: 60°
- WaterLevel: 35%
- UsedCapsBox: 30%
- Power: OK
- …..



- They are classified in terms of nature/subnature
- For Searching and showing on maps and dashboards **HLT of Sensors/Sensor-Actuator** can be:
  - **Sensor Device** name, for example: «mycoffemachine1», «CM23»
  - **Sensor/sensor-actuator** is a variable of a Sensor Device, for example: «Temperature»
- They do not have a model, while, in KB, have a reference process from which their real time data are collected from the field, from gateways, etc..

KPIs, Key Performance Indicators

- They are classified in terms of nature/subnature
- **Typically associated with**
  - **City or infrastructure, so that the GPS can be city center**
  - Some date: 2019, march 2019, etc.
- **For example:**
  - Number of Arrivals from France in March 2019
  - Average price for **** hotels in 2019, downtown
  - Net income of the region
  - CO2 saved in the April 2020
  - Total number of vehicles sold in 2020
  - Stock option value of Airport
- **Note that in most cases:**
  - They are time series, change over time, by year
  - they can be managed as ***Virtual IoT Devices***



Data from INSETE

Basic Sizes of Incoming Tourism

Basic Sizes of Incoming Tourism of the Region of Western Greece 2019

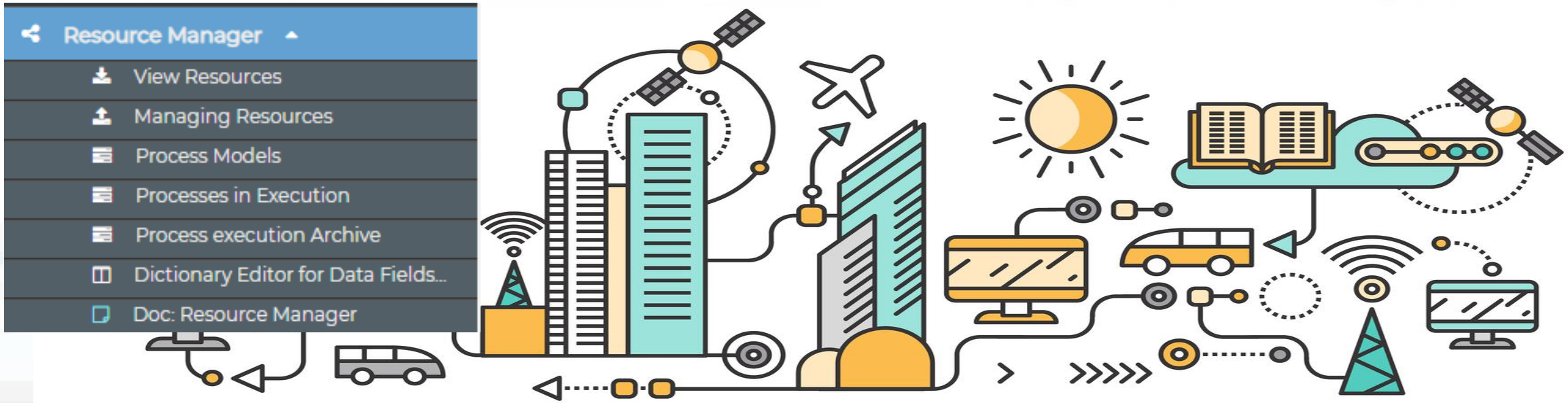
Regions	Countries Origin	Visits (in thousands)	Receipts (in € million)	Nights (in thousands)	Expenditure / Visit (in €)	Cost / Night (in €)	Average Length of Stay
West Hellas	Albania	132.9	26.5	225.8	199.7	117.5	1.7
	United Kingdom	47.7	17.9	345.8	375.2	51.8	7.2
	Germany	70.3	36.4	672.4	517.9	54.1	9.6
	France	55.4	16.5	321.6	298.1	51.4	5.8
	Other	510.7	160.0	2,964.9	313.3	54.0	5.8
	Total	**817.0**	**257.4**	**4,530.4**	**315.0**	**56.8**	**5.5**
	% of the total	**2.2%**	**1.5%**	**1.9%**			

Source: BoG Border Research, INSETE Intelligence Editing

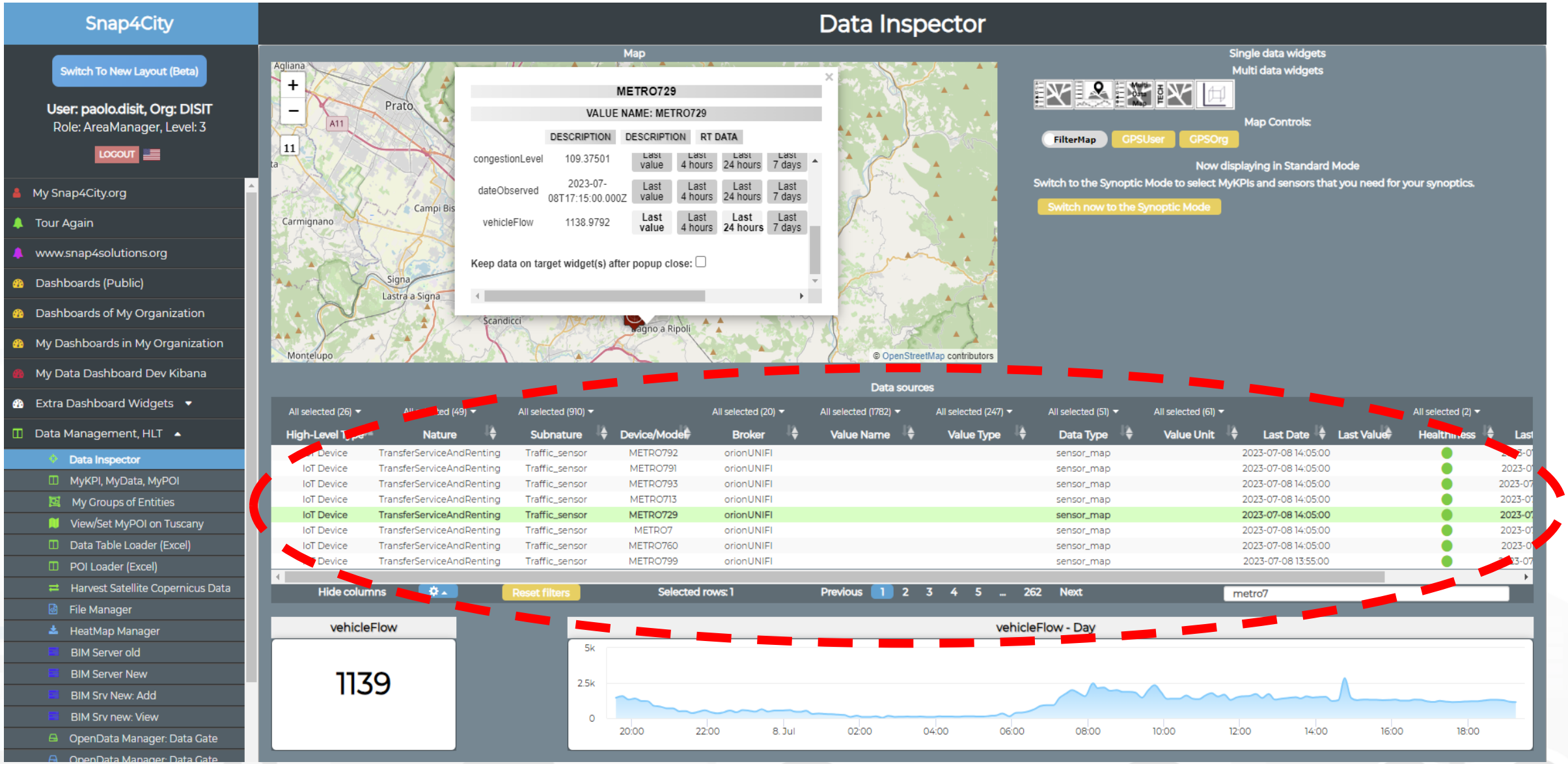
Basic Sizes of Incoming Tourism of the Region of Western Greece 2018

Regions	Countries of Origin	Visits (in thousands)	Receipts (in € million)	Nights (in thousands)	Expenditure / Visit (in €)	Cost / Night (in €)	Average Length of Stay
West Greece	Albania	138.7	29.0	222.9	209.2	130.1	1.6
	United Kingdom	42.6	13.5	180.6	317.6	74.9	4.2
	Germany	71.3	26.0	466.5	365.1	55.8	6.5
	France	44.2	13.5	262.9	304.7	51.2	6.0
	Other	402.5	129.8	2,050.7	322.4	63.3	5.1
	Total	**699.2**	**211.8**	**3,183.5**	**302.9**	**66.5**	**4.6**
	% of the total	**2.0%**	**1.4%**	**1.4%**			

Classification of Variables of Devices, Entities, Sensors, etc. Data Dictionary

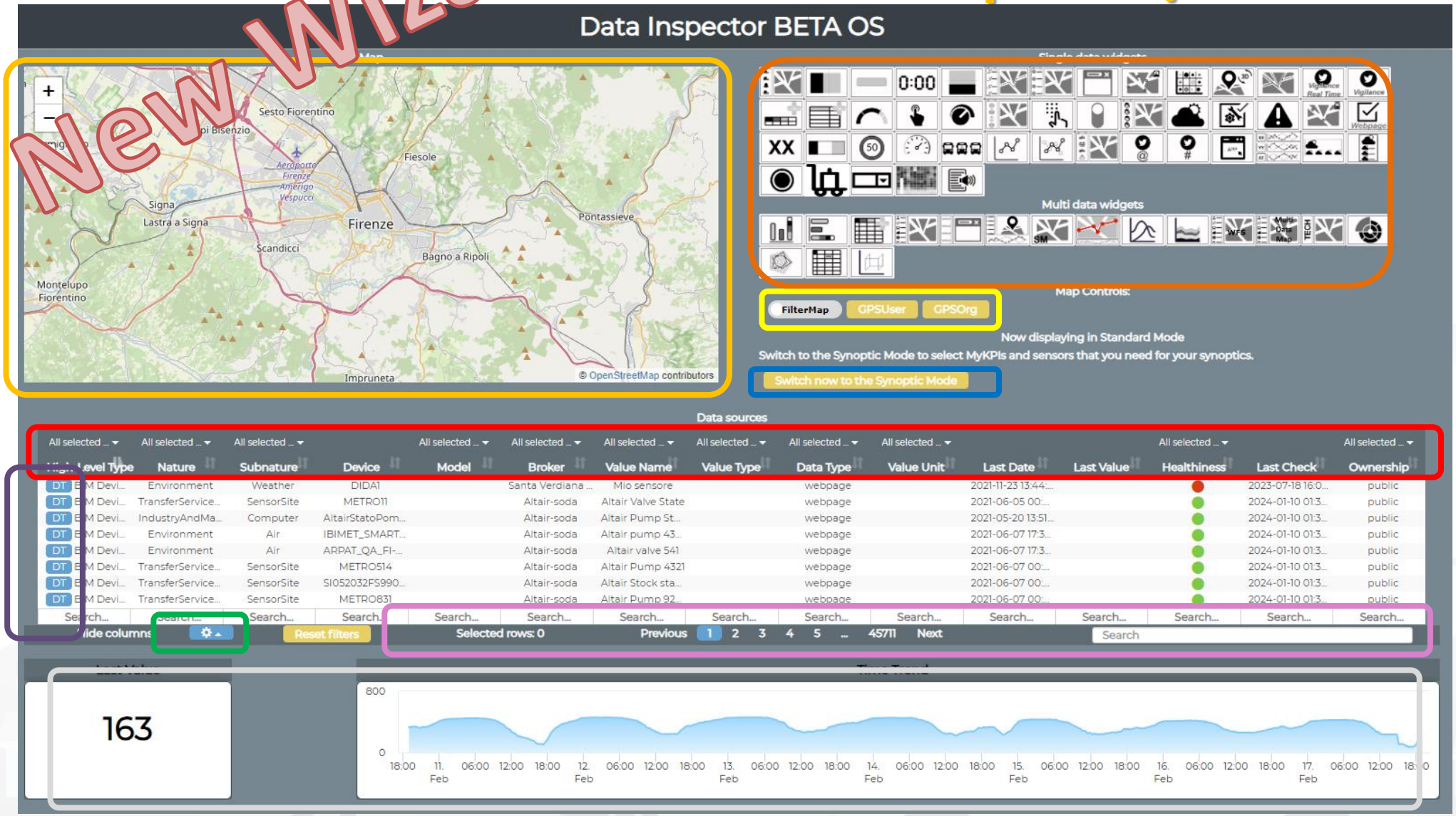


Data Inspector



New Data Inspector/Wizard

New Wizard



Filtering/Searching for individual fields (even for some fields not displayed as geographic coordinates)

Geographic Filtering

Text Search on all fields

Menu for choosing the fields to display in the table

View on Map(via PREVIEW)

Data and Trend visualization

Opening Digital Twin

Pass to Synoptic mode

Select the graph representation

HLT: Unified Classification for Data and Services

IoT Device Variable, Sensor Device	All selected (15)	All selected (48)	All selected (27)		All selected (1499)	All selected (159)	All selected (15)	All selected (63)		All selected (2)		All selected (2)
High-Level Type	Nature	Subnature	Device/Model	Broker	Value Name	Value Type	Data Type	Value Unit	Last Date / Last Value	Healthiness	Last Check	Ownership
IoT Device Variable	IoTDevice	IoTSensor	devicetest1	orionUNIFI	temperature	temperature	float	°C			2021-10-15 10:01:02	private (My Own)
IoT Device Variable	IoTDevice	IoTSensor	devicetest1	orionUNIFI	humidity	humidity	float	#			2021-10-15 10:01:02	private (My Own)
IoT Device Variable	IoTDevice	IoTSensor	MyThermometer_001	orionUNIFI	temperature	temperature	float	°C			2021-10-15 10:01:01	private
IoT Device Variable	IoTDevice	IoTSensor	MyThermometer_001	orionUNIFI	humidity	humidity	float	#			2021-10-15 10:01:01	private
IoT Device Variable	IoTDevice	IoTSensor	adminTest1	orionUNIFI	temperature	temperature	string	°C	2018-05-31 19:16:05		2021-10-15 10:01:00	private (My Own)
IoT Device Variable	IoTDevice	IoTSensor	adminTest1	orionUNIFI	humidity	humidity	string	%	2018-05-31 19:16:05		2021-10-15 10:01:00	private (My Own)
IoT Device Variable	IoTDevice	IoTSensor	newmarcodev1	orionUNIFI	temperature	temperature	float	°C			2021-10-15 10:00:59	private
IoT Device Variable	IoTDevice	IoTSensor	newmarcodev1	orionUNIFI	humidity	humidity	float	%			2021-10-15 10:00:59	private

Hide columns | Reset filters | Selected rows: 1 | Previous 1 2 3 4 5 ... 2466 Next | Search

High Level Types

Nature

SubNature

Semantic Classific.

Dev/Model name

Broker name

Technical Source

Value Name

Value Type

Data Type

Value Unit

Variables, names

Last Date/Time

Last Value

Real Time

Healthiness

Last Check

Status

Ownership

Organization

For Admin

Main HLT diff.

- **HLT: MyKPI**
- **Nature: Industry and manufacturing**
- **Subnature: Chemical**
- **Value Name**: CloroParaffine
- **Value Type: Density percentage**
- **Value Unit: %**
- **Data Type**: float mykpi
- **Last Date**: 2019-02-25 *«DateTime»*
- **Last Value**: 87.0
- **Healthiness**:
- **Last Check**: 2020-04-03 10:28:12
- **Ownership**: private for xyz…
- **Organization**: Firenze

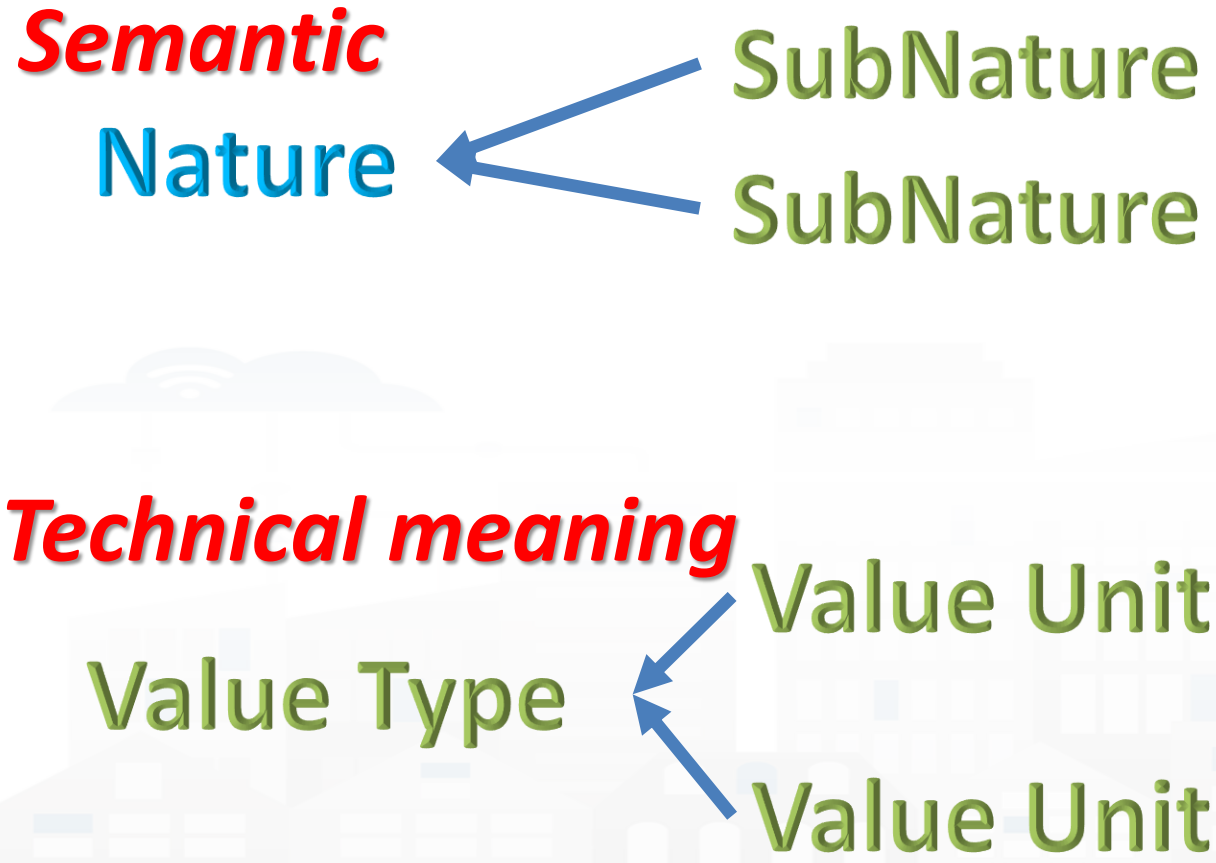
Single Variable for MyKPI

- **HLT: IoT Device, Sensor Device, Data Tab Devic**
- **Nature: Industry and manufacturing**
- **Subnature: Chemical**
- **Value Name**: Irrigator fioriera Gag
- **Value Type: Battery Level**
- **Value Unit: V**
- **Data Type**: float
- **Last Date**: 2020-04-01 12:59:00 *«dateObserved»*
- **Last Value**: 5.18
- **Healthiness**:
- **Last Check**: 2020-04-03 03:28:12
- **Ownership**: public/private
- **Organization**: Firenze

This Section is repeated for each variable

an IoT Device may have multiple Sensors/variables

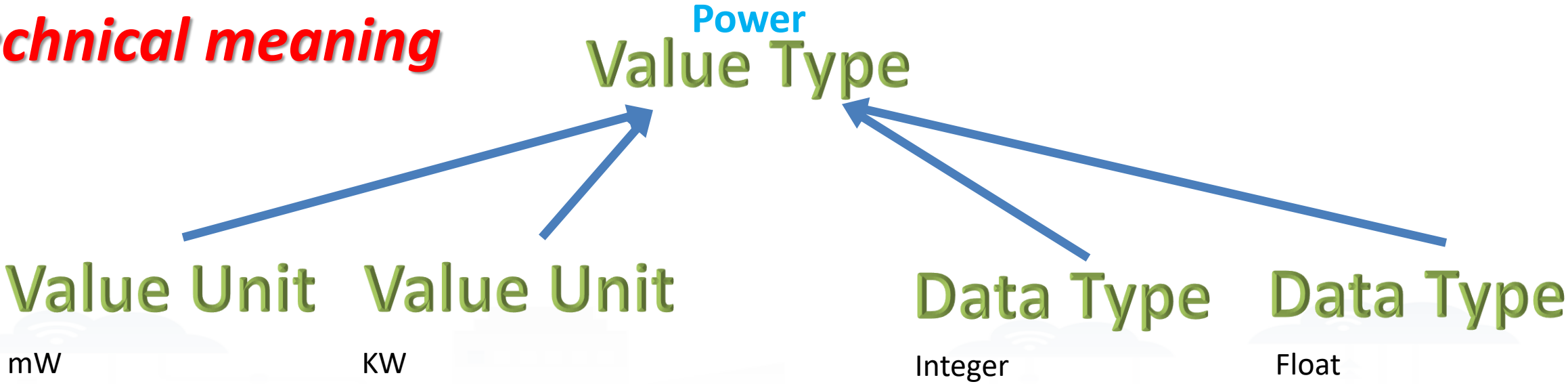
Unified Data and Services Model/Classification



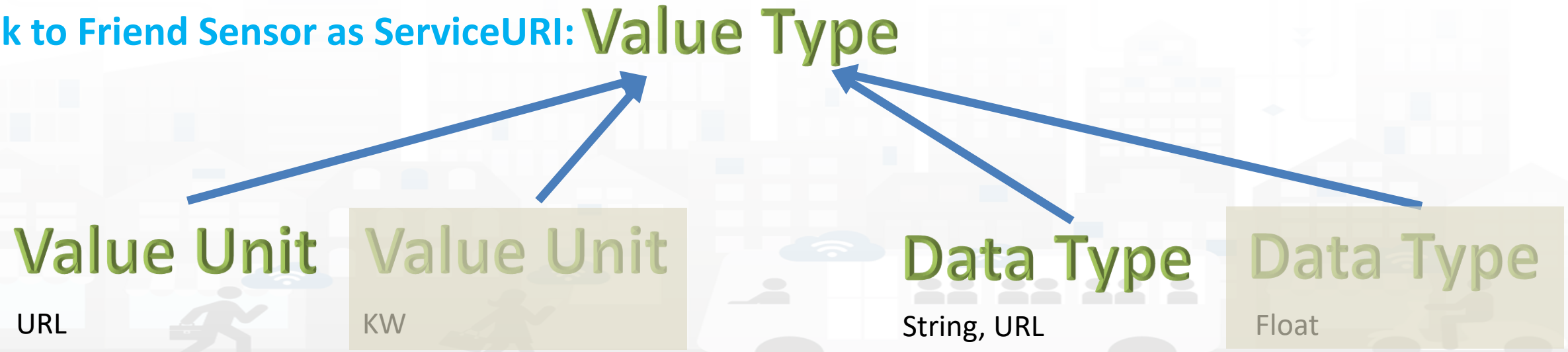
- Exists a Dictionary for the 4 categories
- They are related each other and not all values are possible
- **Right setting** leads to right rendering on graphs / dash
- **automated** combinations and processing
- The Dictionary is used by many tools

For example

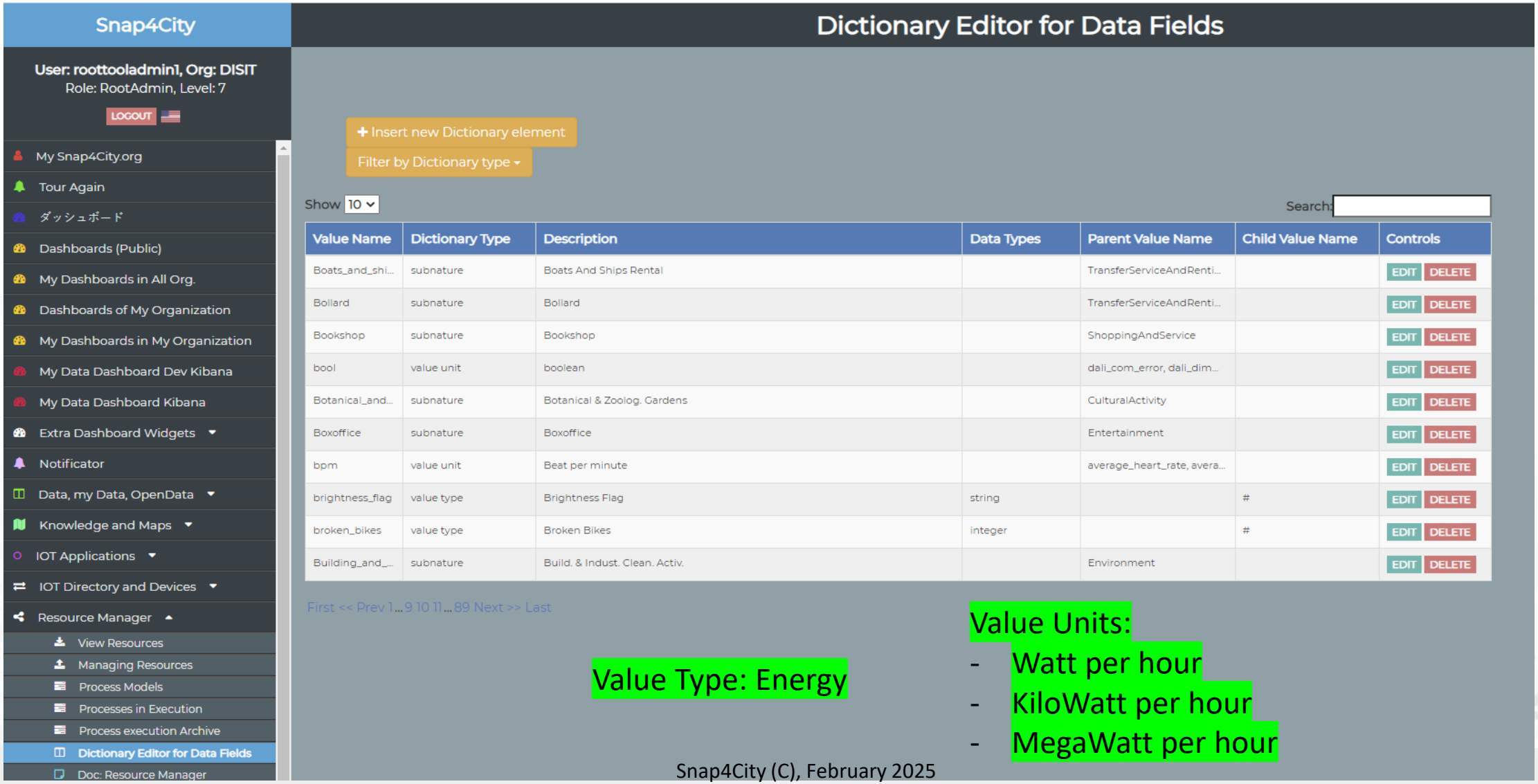
Technical meaning



Link to Friend Sensor as ServiceURI:



Example of Energy and its Value Units



Value Type: Energy

Value Units:
- Watt per hour
- KiloWatt per hour
- MegaWatt per hour

Please note on: Data Type

- Value Types have only a few number of **Data Types** because they represent how the data area treated into the system
- Therefore: main Data Types are:
  - **Float**: numbers with decimals large as you like, etc.
  - **Integer**: numbers, booleans (0/1), etc.
  - **String**: url, links, names, id, descriptions, status code, etc.
  - **Json**: structured data, vector, matrices, etc.
  - …..

IoT Device Model and Devices Data Dictionary: updated on web at 03/2023

UPDATED on Snap4City.org Tool every day

https://www.snap4city.org/818

Home / IoT Device Model and Devices Data Dictionary: updated at 11/2022

IoT Device Model and Devices Data Dictionary: updated at 11/2022

View | Edit | Track | Access control | Convert

Any update and addtion to the dictionary of snap4city.org has to be requested to snap4city@disit.org

If you have your own instance of the platform you can define your own dictionary and request a copy of the snap4city.org dictionary

The dictionary is used into the IoT Device Model definition, in mapping smart data models, and in creating full custom devices.

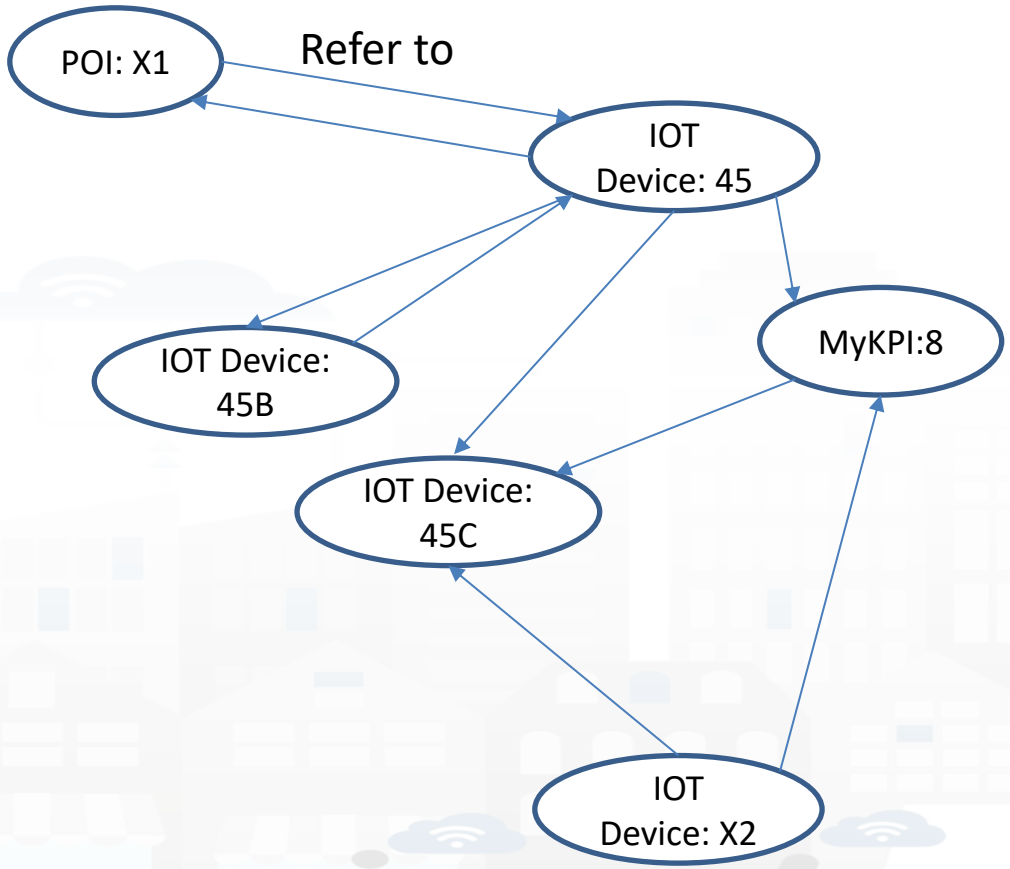
https://www.snap4city.org/drupal/sites/default/files/image_from_word/fil...

value type	Description	possible value Units	Possible Data Types
actuator_canceller	Actuator Canceller		string
actuator_deleted	Actuator Deleted		integer
actuator_deletion_date	Actuator Deletion Date	timestamp	string
air_quality_index	Air quality index		float
altitude	Altitude	m	float,integer
angle	angle	deg	float
annual_C6H6_average	annual_C6H6_average	ppm, mg/m3, µg/m³	float
annual_C6H6_exceedance_count	annual_C6H6_exceedance_count	#	integer,float
annual_CO_average	annual_CO_average	ppm, mg/m3, µg/m³	float
annual_CO_exceedance_count	annual_CO_exceedance_count	#	integer,float
annual_NO2_average	annual_NO2_average	ppm, mg/m3, µg/m³	float
annual_NO2_exceedance_count	annual_NO2_exceedance_count	#	integer,float
annual_O3_average	annual_O3_average	ppm, mg/m3, µg/m³	float

References/Links to Entities Instances / IoT Devices



Relationships among Devices/Entities, POI and MyKPI



- **Devices and POI** may refer to:
  - IoT Devices/Entities, POI, MyKPI, Heatmaps, etc.
  - The Links may change over time
- **MultiDataMap** can be used for navigation:
  - Among: IoT Devices, POI, MyKPI
  - Automated focus
  - Accessing Time Trends

IoT device with References to other and MyKPI


```
{
"id":"ThermalBOX1",
"type":"thermalbox",
"dateObserved":{"type":"string","value":"2022-02-24T17:15:34.609Z"},
"latitude":{"type":"float","value":"43.76965"},
"longitude":{"type":"float","value":"11.25570"},
"SHTdevice":{"type":"string",
      "value":"http:\/\/www.disit.org\/km4city\/resource\/iot\/orionFirenze2\/Firenze\/SHT20lab_new"},
"cam51count":{"type":"string","value":"datamanager\/api\/v1\/poidata\/17058000"},
"cam52count":{"type":"string","value":"datamanager\/api\/v1\/poidata\/17058001"},
…
}
```


SURI

Value Type: Identifier

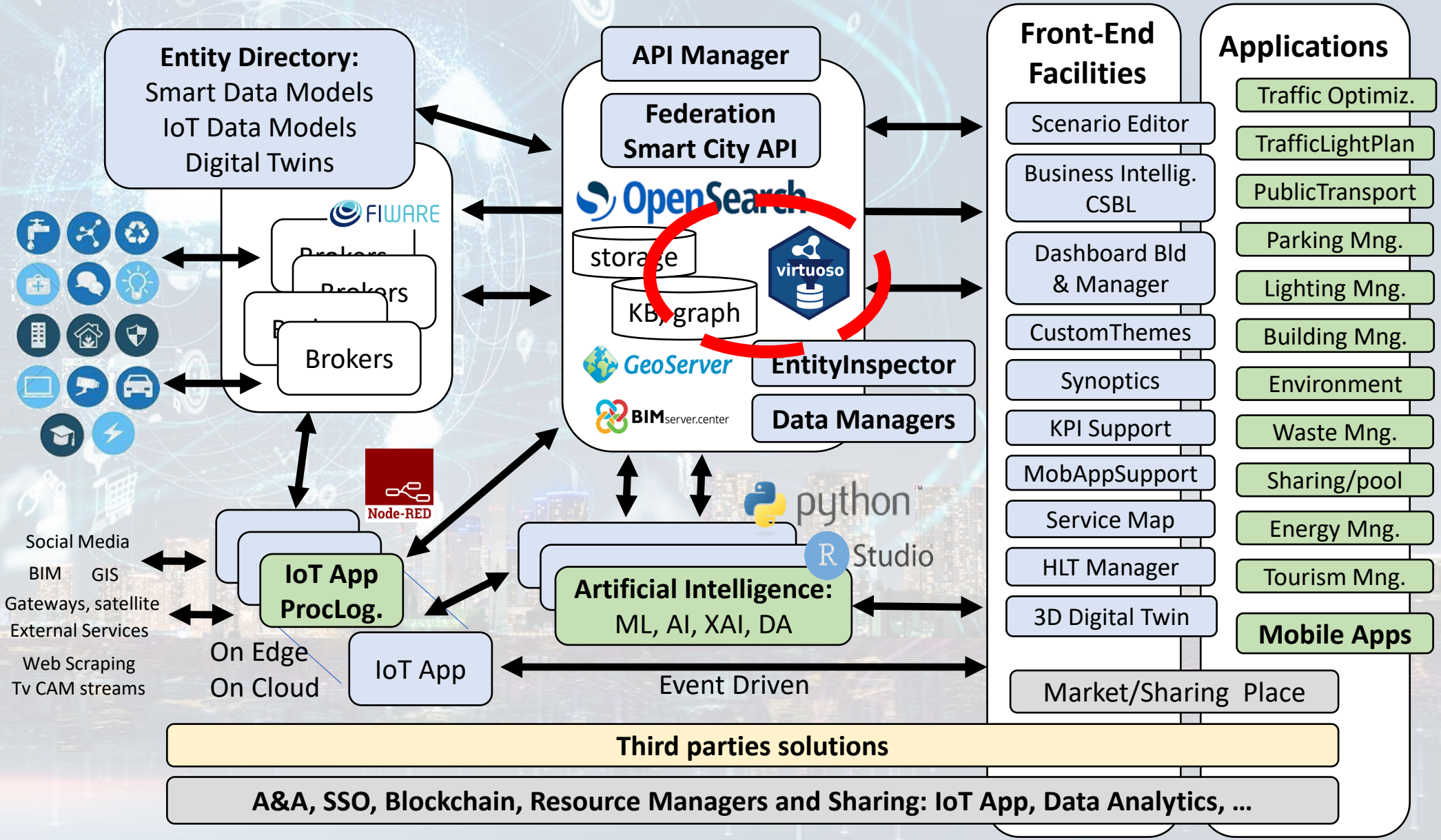
Value Unit: ServiceURI *//any query: such as those of the Selector*

Data Type: String



Knowledge Base: Modelling and ServiceURI as Entity Identifier

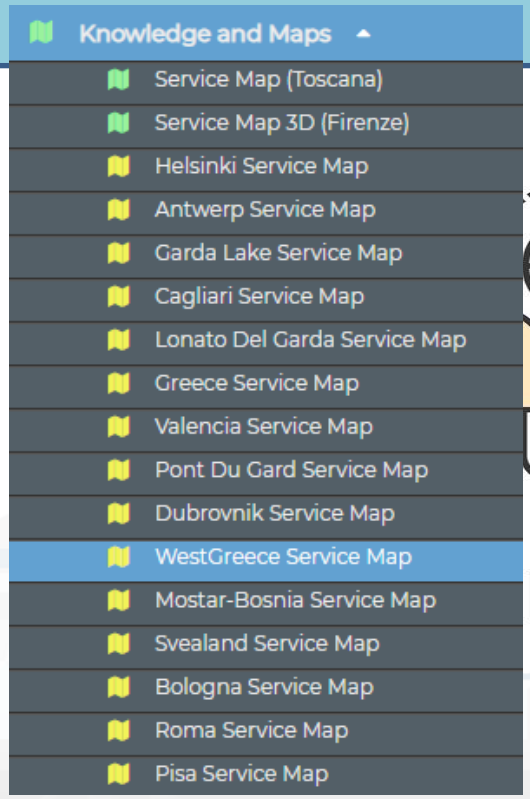
Technical Architecture



Knowledge Base city structure

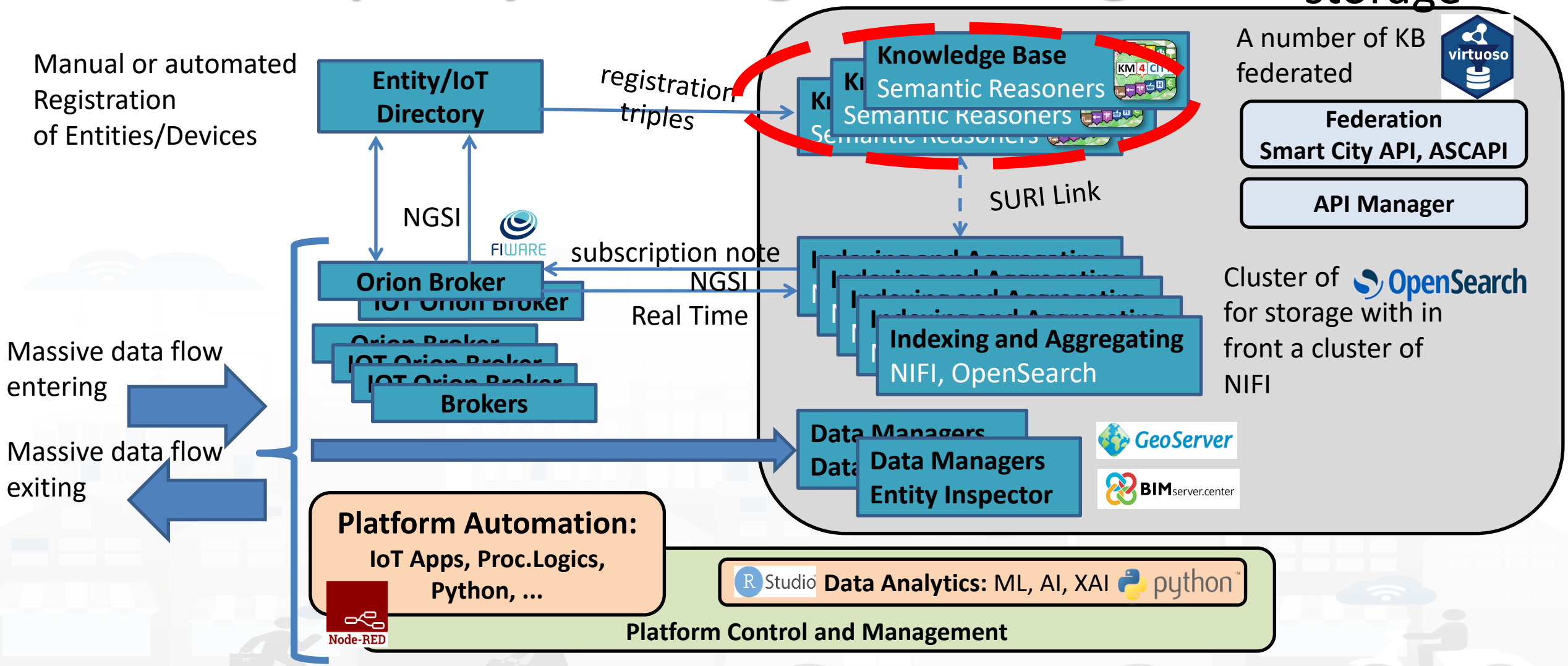
- Needs of the KB city Structure:
  - For many trivial applications of Smart City the KB does not need to be initialized with some road graph, for example taken from OSM
  - For example, if you need only to position devices on map to some GPS coordinates you do now need to set up the KB
- The KB Set up is needed only when is needed to have:
  - Geoserver functionality
  - Routing based on KB
  - Some spatial reasoning queries
  - Etc.

Usage of the ServiceMap and Knowledge Base Browsing





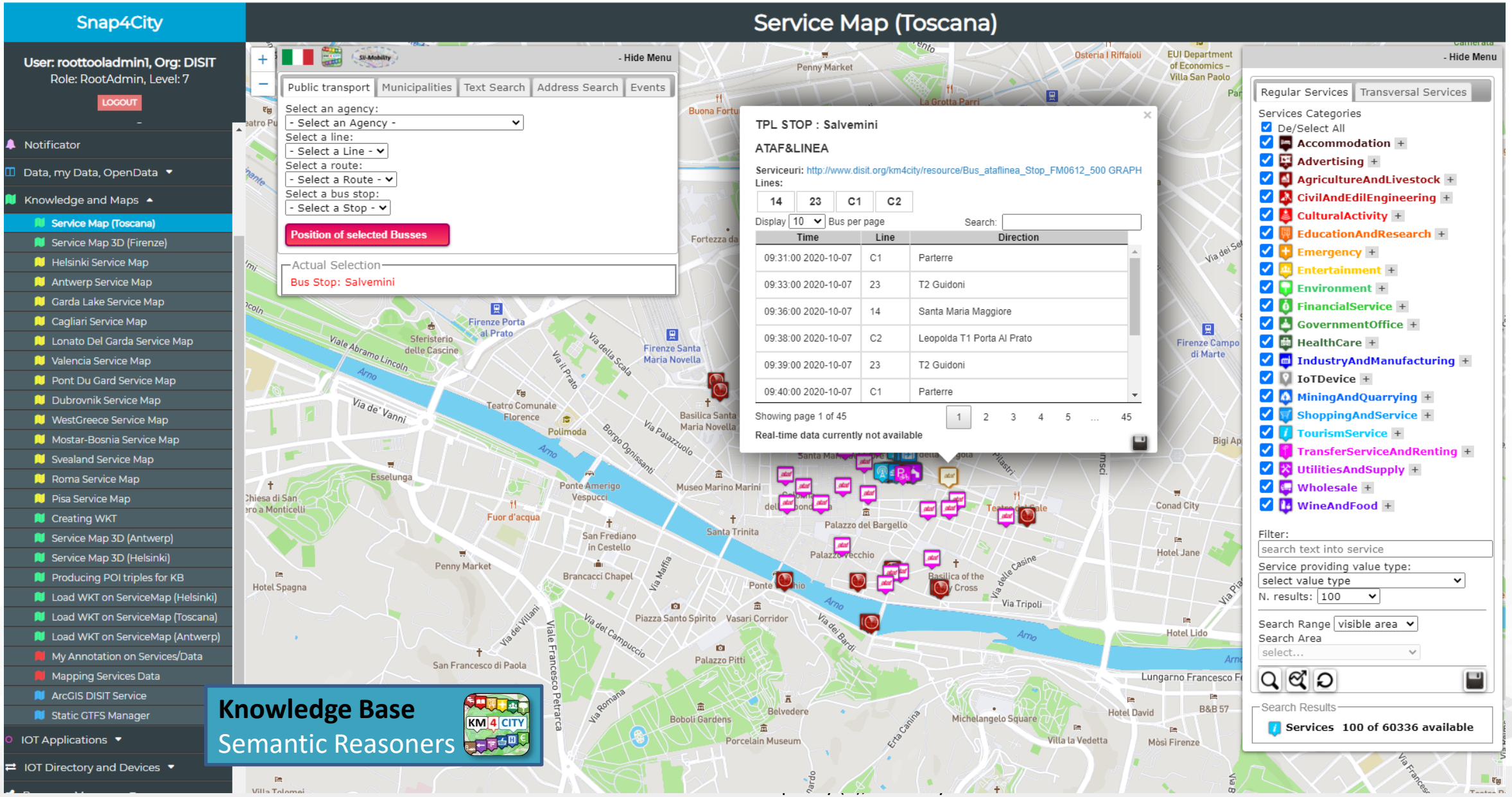
Snap4city Data Ingestion Diagram



UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

ServiceMaps

SNAP4CITY | KM4CITY

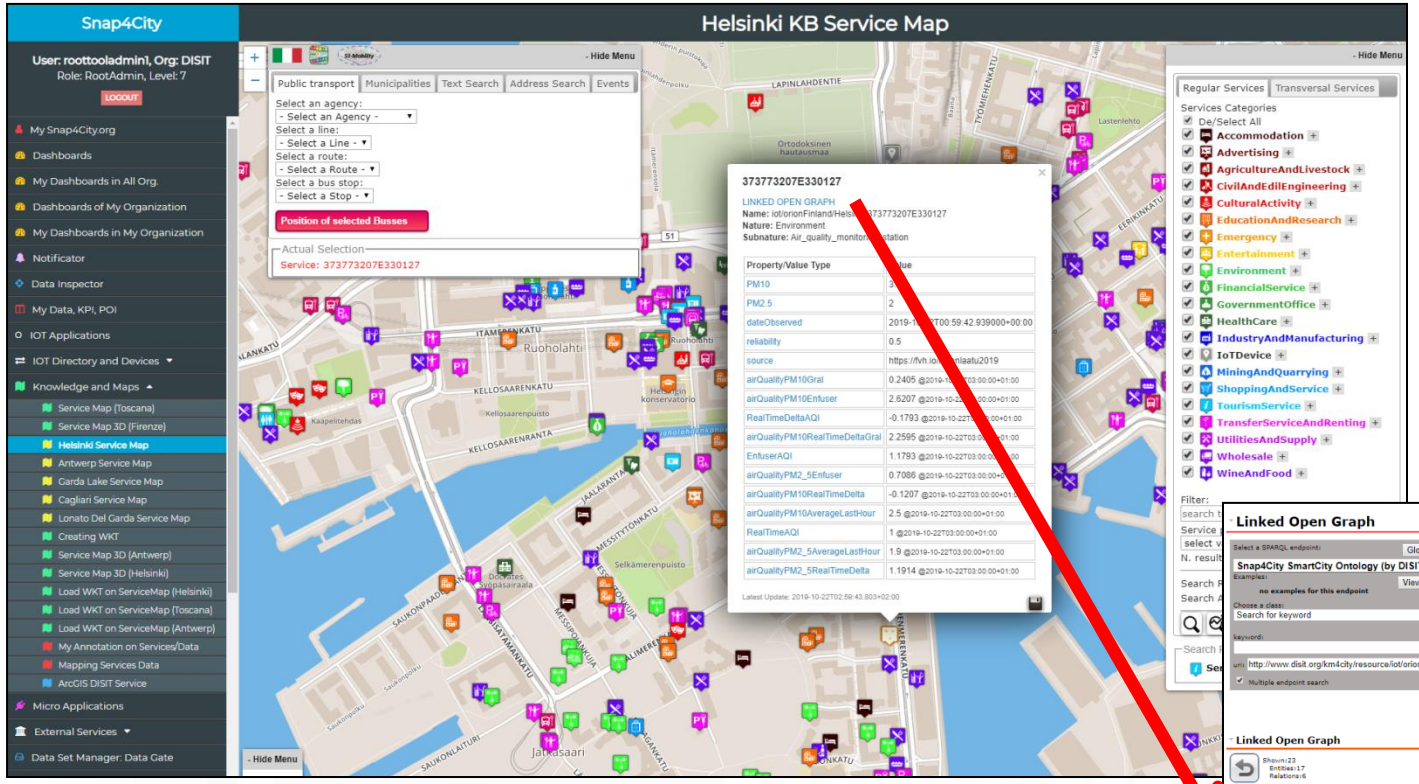


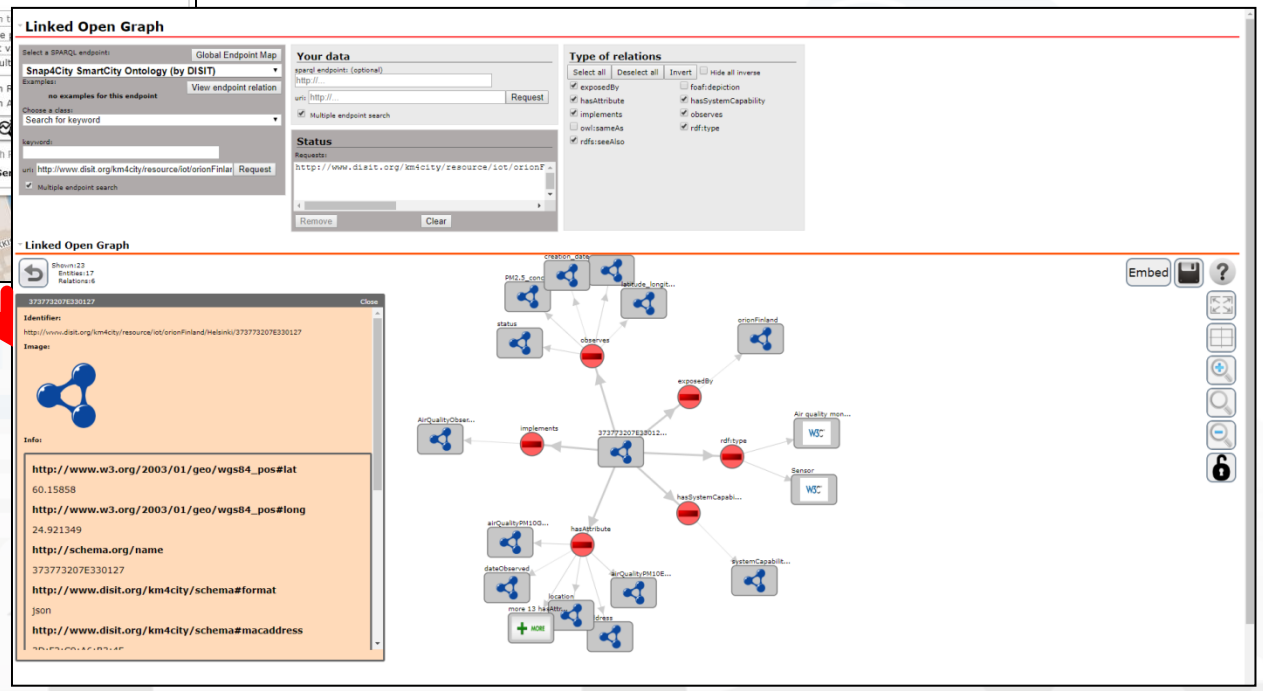
Semantic modeling

- **KB is based on the Km4City ontology,** It allows to:
  - keep connected city entities each other:
    - Semantic Index, reticular
    - **Perform spatial, geo graphic, and temporal reasoning**
  - Discover city entities and their relationships via Proc.Logic / IoT App and Smart City API:
    - Entities / devices, sensors, city elements, roads, services, Brokers, etc. etc.
  - Provide access via Advanced Smart City API, **ASCAPI**
  - **Federate Smart Cities** / Knowledge Bases, the approach allows to scale geographically and create redundancies, improving performances

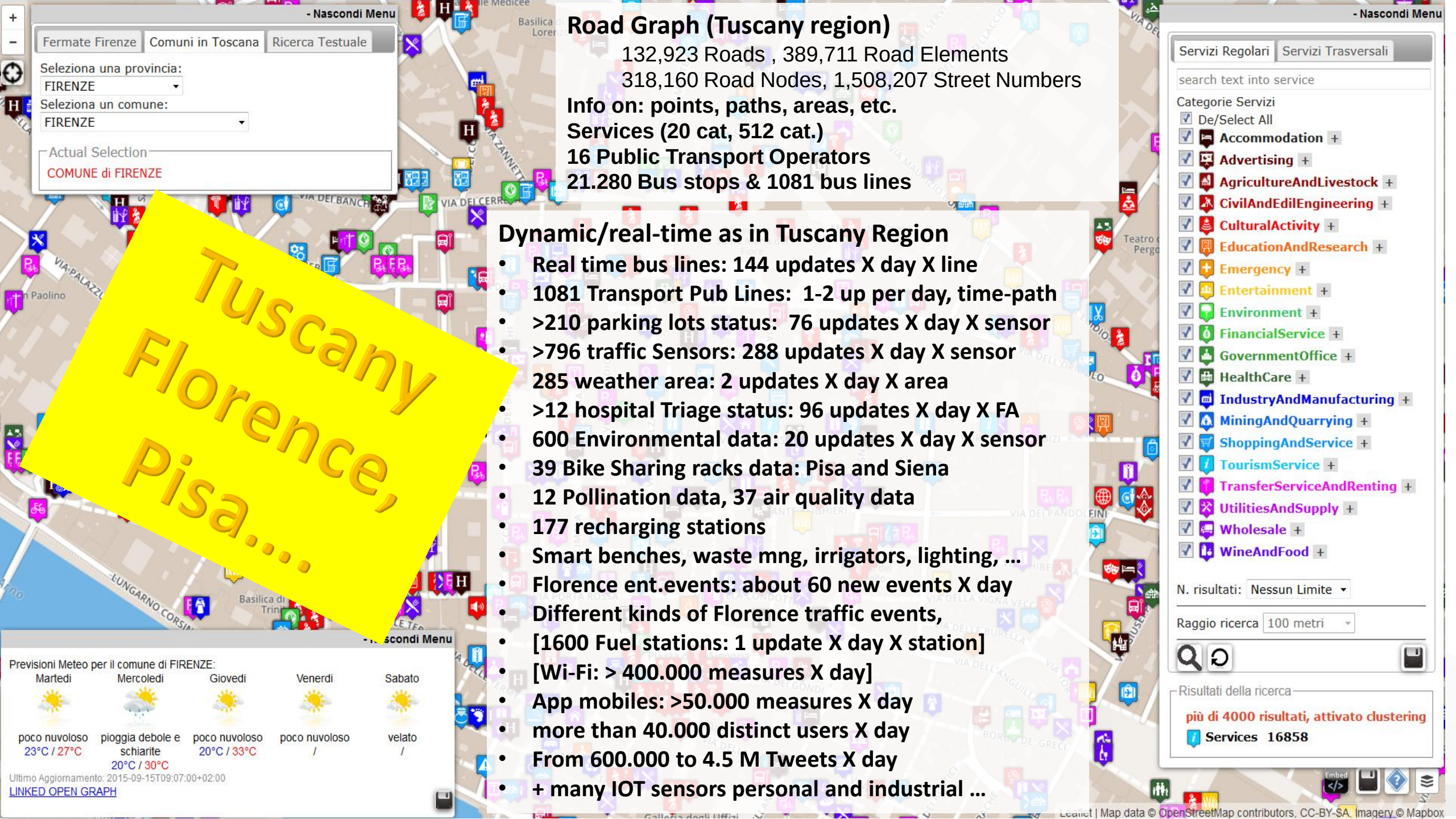
Views of the Knowledge Base

Knowledge Base
Semantic Reasoners





- How pass from ServiceMap to Linked Open Graph, Linked Data view tool


- Nascondi Menu
Fermate Firenze
Comuni in Toscana
Ricerca Testuale
Seleziona una provincia:
FIRENZE
Seleziona un comune:
FIRENZE
Actual Selection
COMUNE di FIRENZE
Tuscany
Florence,
Pisa....
Road Graph (Tuscany region)
132,923 Roads , 389,711 Road Elements
318,160 Road Nodes, 1,508,207 Street Numbers
Info on: points, paths, areas, etc.
Services (20 cat, 512 cat.)
16 Public Transport Operators
21.280 Bus stops & 1081 bus lines
Dynamic/real-time as in Tuscany Region
• Real time bus lines: 144 updates X day X line
• 1081 Transport Pub Lines: 1-2 up per day, time-path
• >210 parking lots status: 76 updates X day X sensor
• >796 traffic Sensors: 288 updates X day X sensor
285 weather area: 2 updates X day X area
• >12 hospital Triage status: 96 updates X day X FA
• 600 Environmental data: 20 updates X day X sensor
• 39 Bike Sharing racks data: Pisa and Siena
• 12 Pollination data, 37 air quality data
• 177 recharging stations
• Smart benches, waste mng, irrigators, lighting, …
• Florence ent.events: about 60 new events X day
• Different kinds of Florence traffic events,
• [1600 Fuel stations: 1 update X day X station]
• [Wi-Fi: > 400.000 measures X day]
• App mobiles: >50.000 measures X day
• more than 40.000 distinct users X day
• From 600.000 to 4.5 M Tweets X day
• + many IOT sensors personal and industrial …
- Nascondi Menu
Previsioni Meteo per il comune di FIRENZE:
Martedi
poco nuvoloso
23°C / 27°C
Mercoledi
pioggia debole e schiarite
20°C / 30°C
Giovedi
poco nuvoloso
20°C / 33°C
Venerdi
poco nuvoloso
/
Sabato
velato
/
Ultimo Aggiornamento: 2015-09-15T09:07:00+02:00
LINKED OPEN GRAPH
- Nascondi Menu
Servizi Regolari
Servizi Trasversali
search text into service
Categorie Servizi
De/Select All
Accommodation
Advertising
AgricultureAndLivestock
CivilAndEdilEngineering
CulturalActivity
EducationAndResearch
Emergency
Entertainment
Environment
FinancialService
GovernmentOffice
HealthCare
IndustryAndManufacturing
MiningAndQuarrying
ShoppingAndService
TourismService
TransferServiceAndRenting
UtilitiesAndSupply
Wholesale
WineAndFood
N. risultati: Nessun Limite
Raggio ricerca 100 metri
Risultati della ricerca
più di 4000 risultati, attivato clustering
Services 16858
Leaflet | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox

Discovery

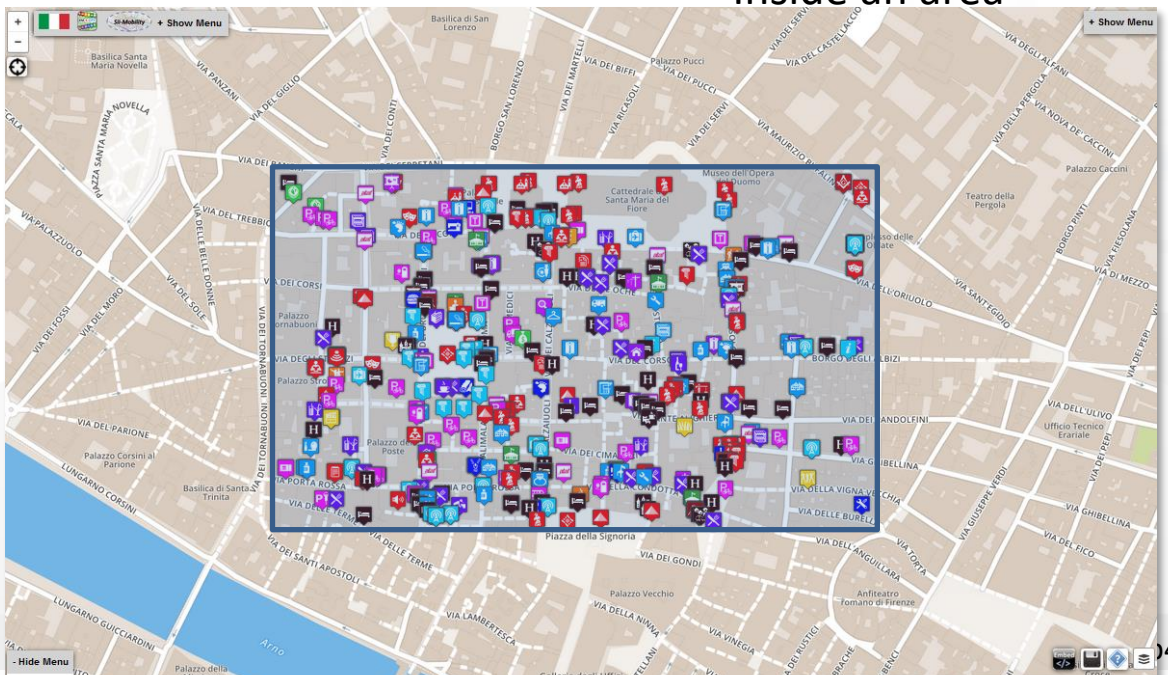
Search by Shape and Distance

Around a point or POI

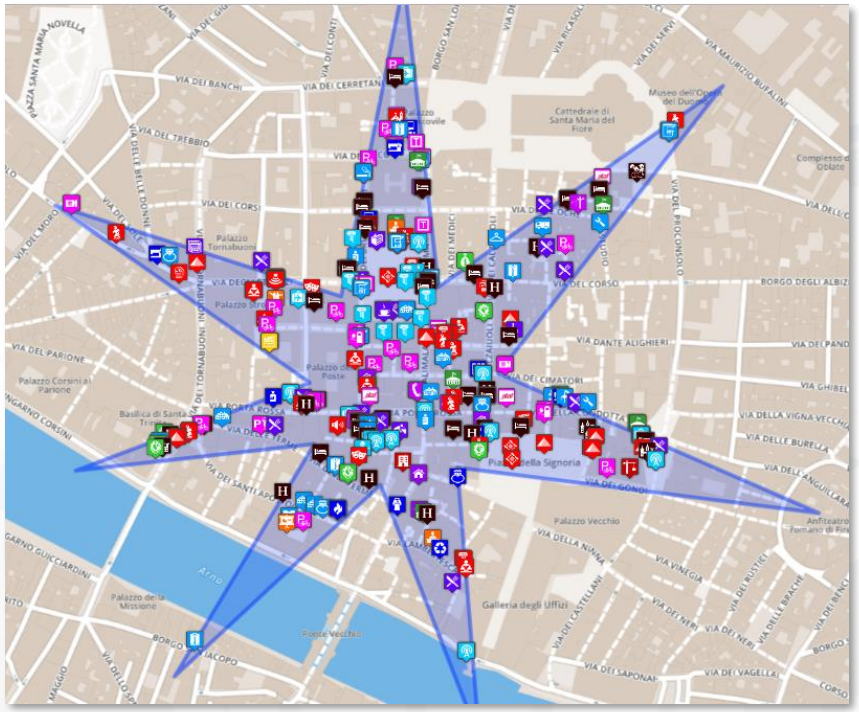


Each request or search in the Km4City model can be referred to a point and a ray, to an area, to a polyline

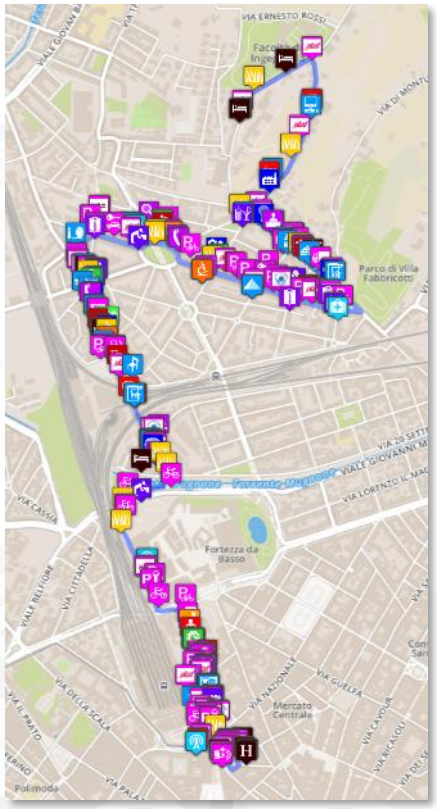
Inside an area



Inside a closed polyline

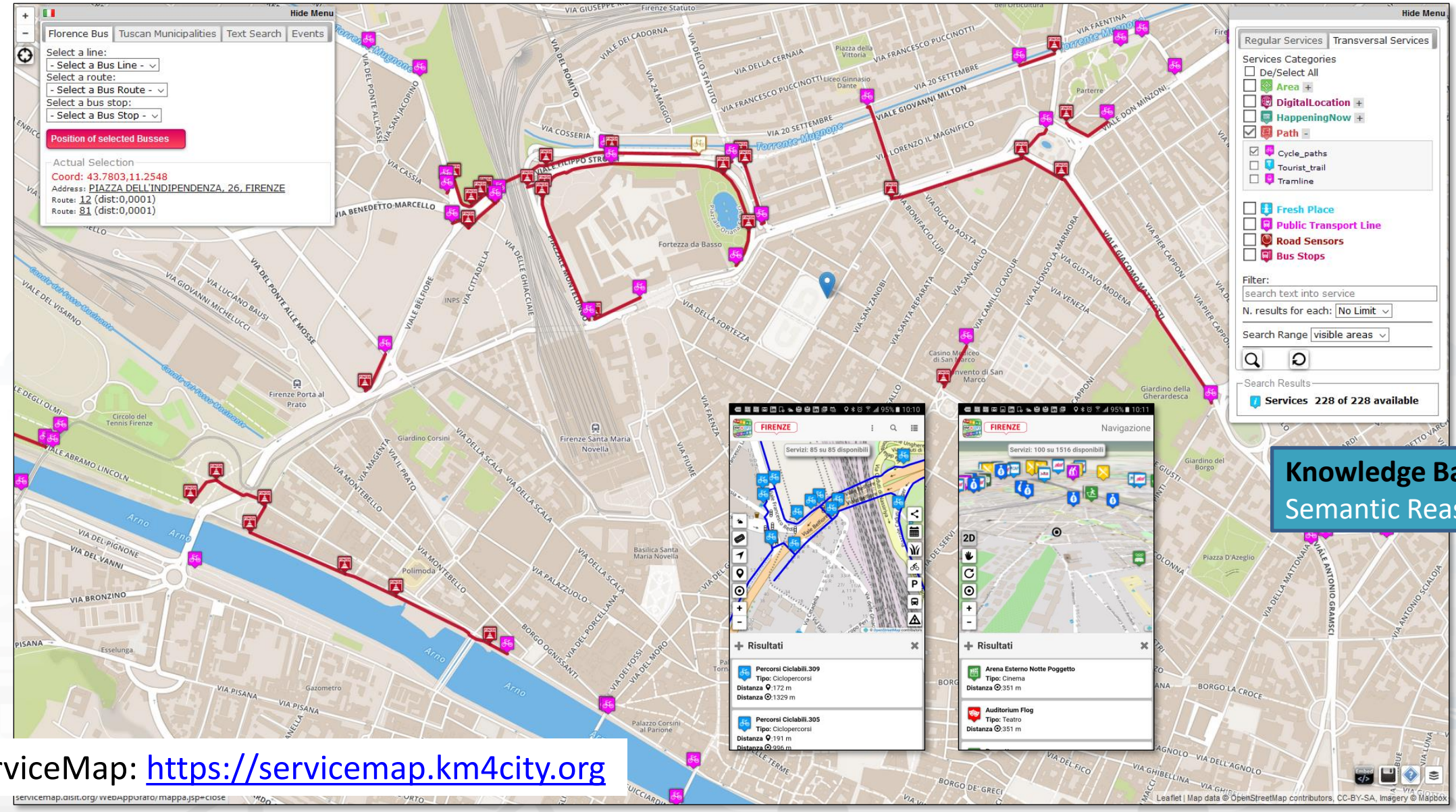


Along a polyline



Knowledge Base
Semantic Reasoners

Cycling Paths



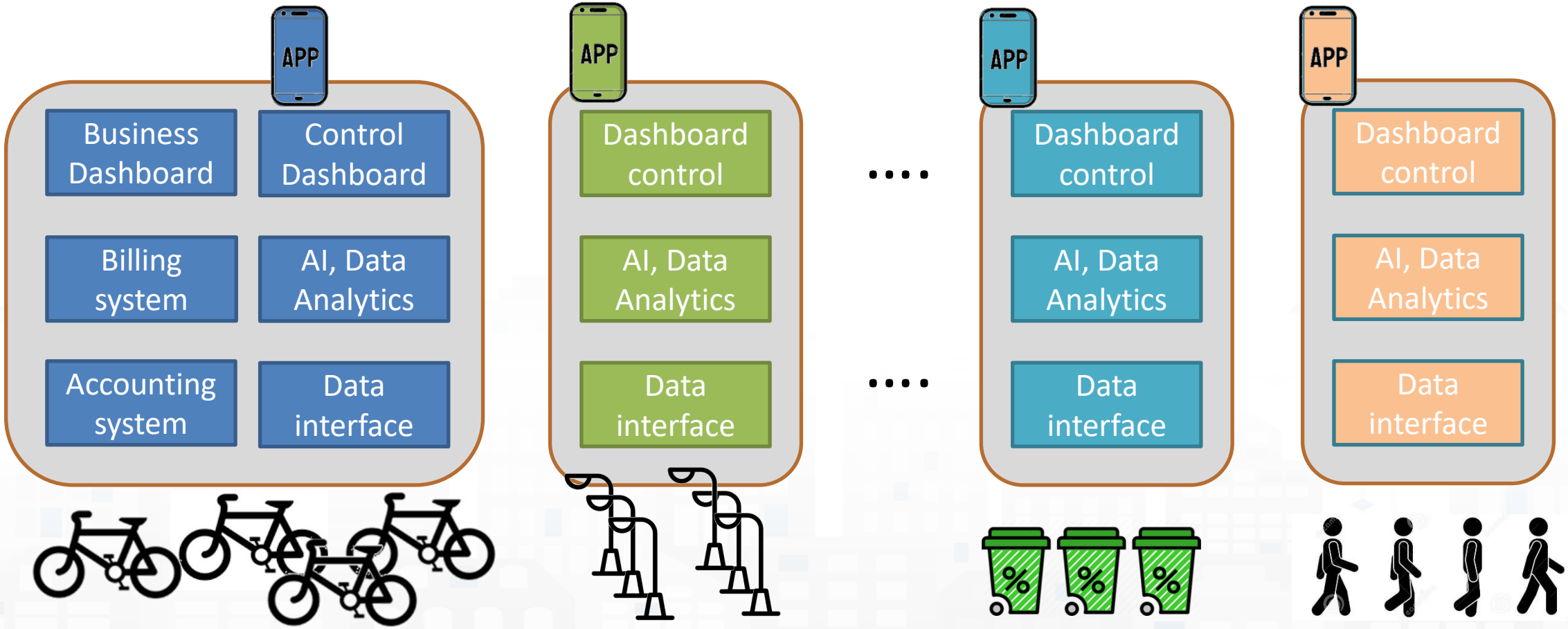
Knowledge Base
Semantic Reasoners

ServiceMap: https://servicemap.km4city.org

Set up of the Knowledge Base performed with an open source tool

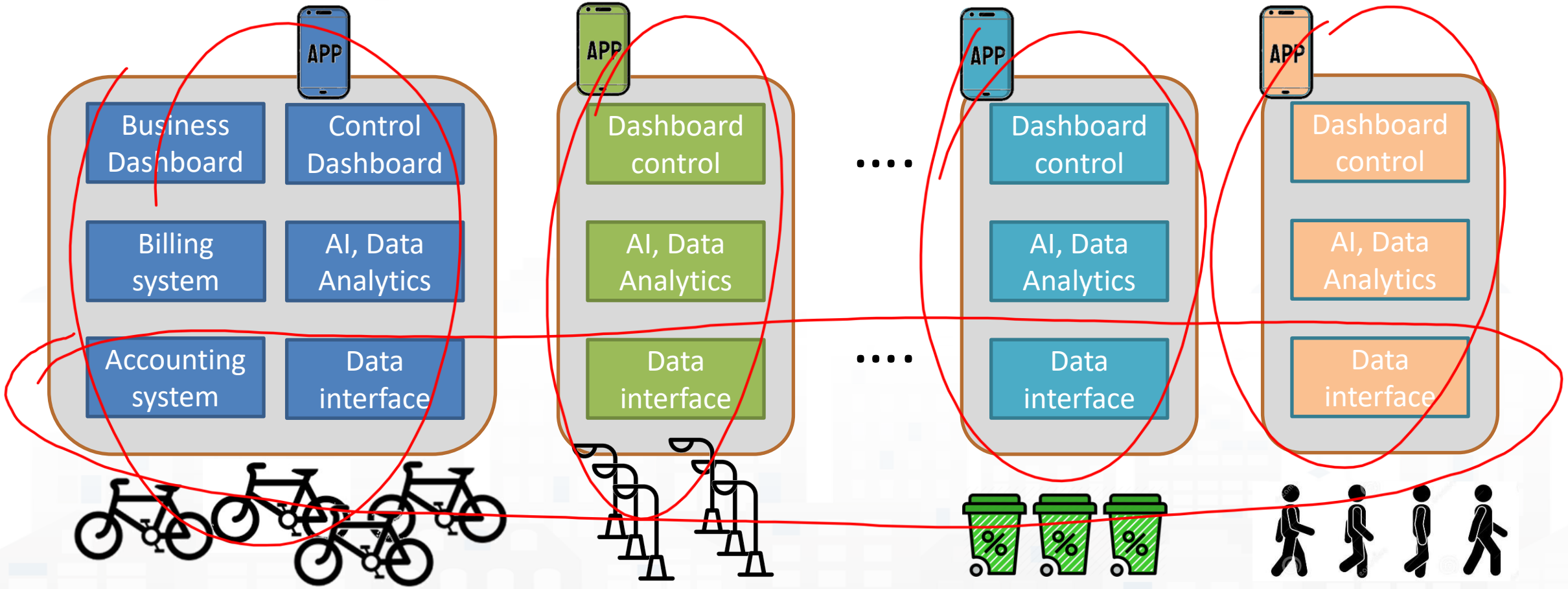


Avoiding to have a collection of verticals



Simplifying the development and integration of verticals

Avoiding to have a collection of verticals

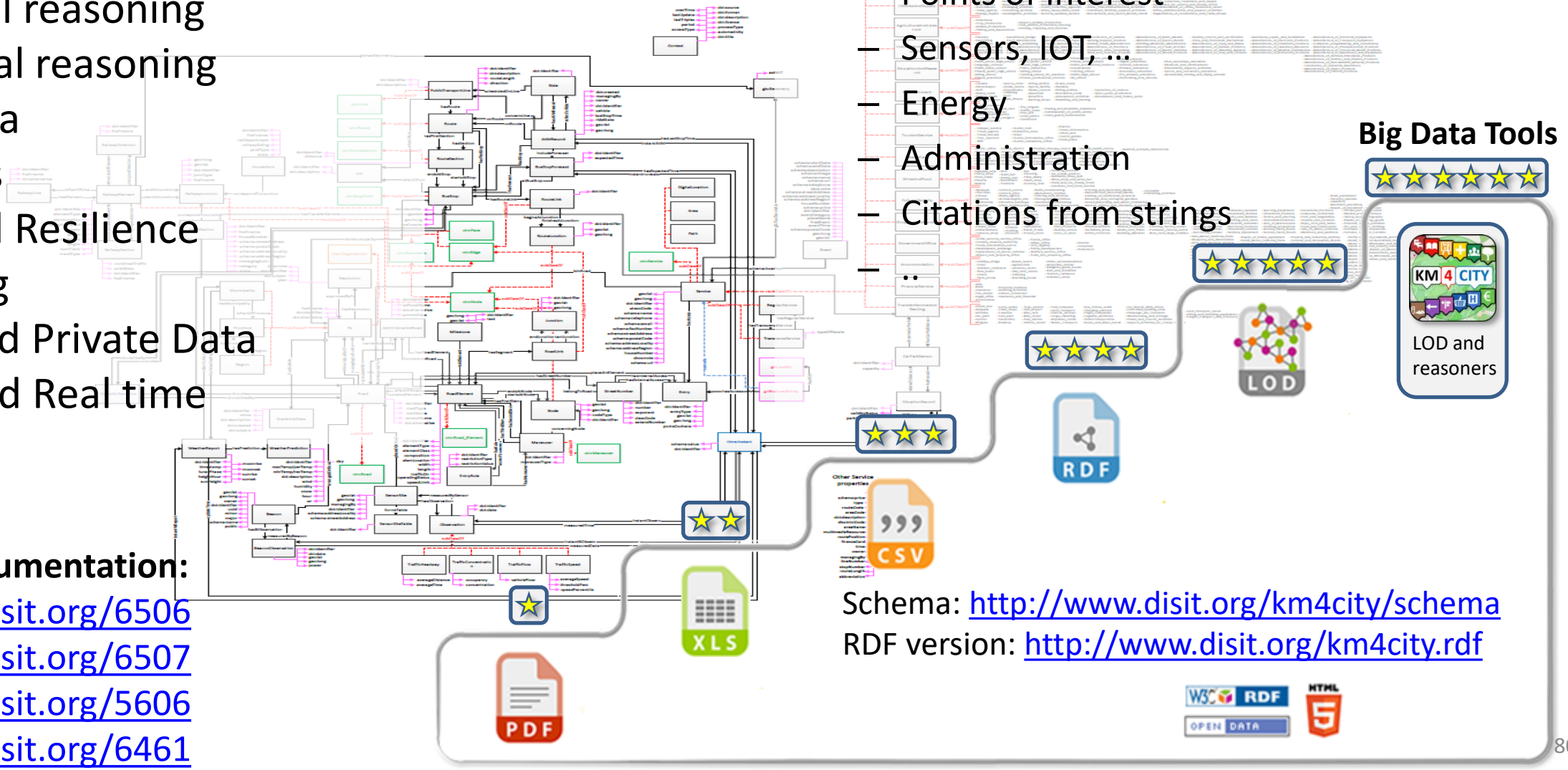


Simplifying the development and integration of verticals

Km4City: Knowledge Base

- **Multiple DOMAINS**
- Geospatial reasoning
- Temporal reasoning
- Relational reasoning
- Metadata
- Statistics
- Risk and Resilience
- Licensing
- Open and Private Data
- Static and Real time
- IOT/IOE

- Street-Guide
- Mobility and transport
- Points of interest
- Sensors, IOT, …
- Energy
- Administration
- Citations from strings
- ..

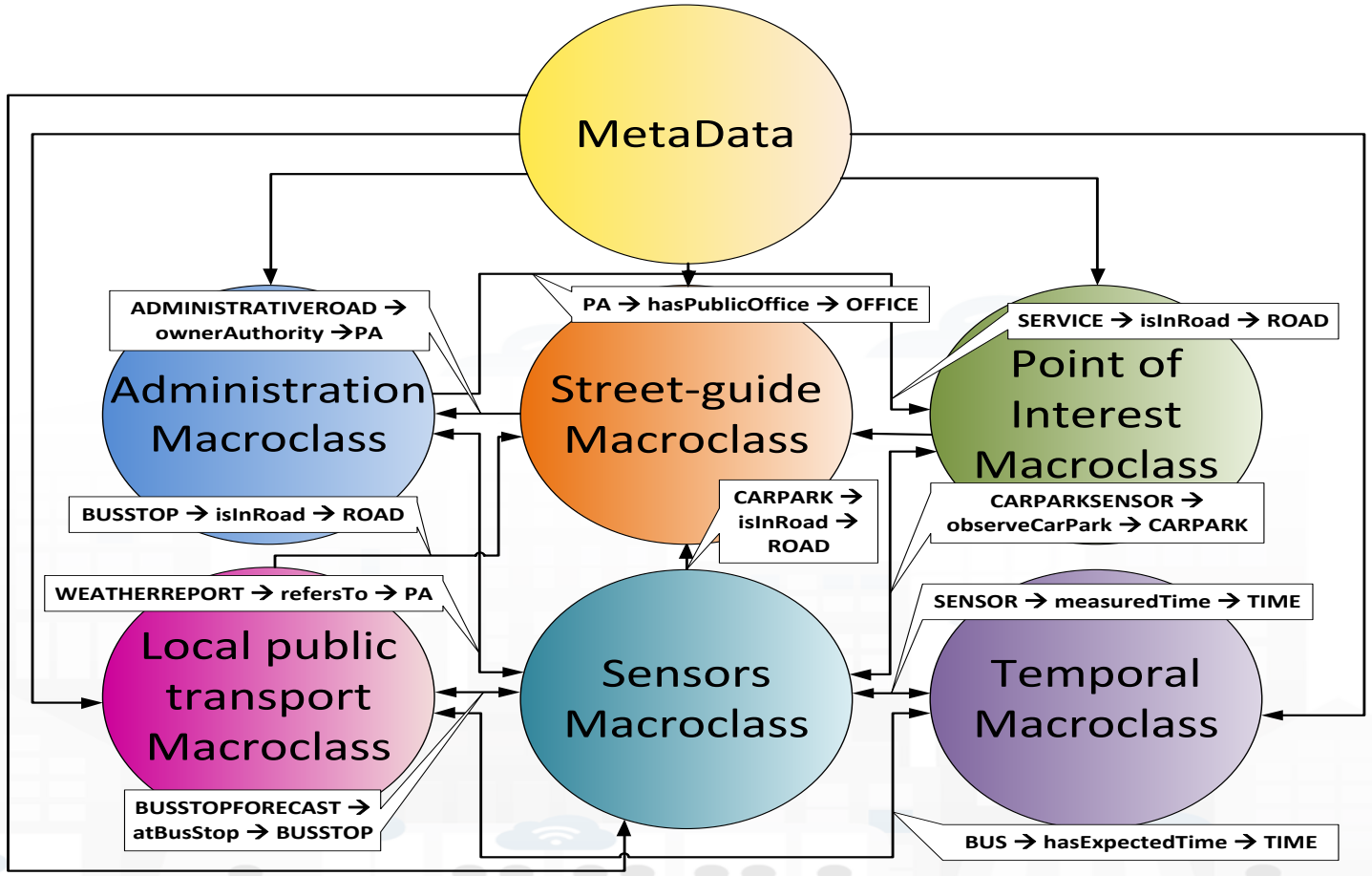


Ontology Documentation:
http://www.disit.org/6506
http://www.disit.org/6507
http://www.disit.org/5606
http://www.disit.org/6461

Schema: http://www.disit.org/km4city/schema
RDF version: http://www.disit.org/km4city.rdf

Smart-city Ontology: 1.6.8

- covers different aspects:
  - Administration
  - Street-guide
  - Points of interest
  - Local public transport
  - Sensors
  - Temporal aspects
  - Metadata on the data
  - Industry 4.0 structures
  - Digital Twin models
  - High Level Types

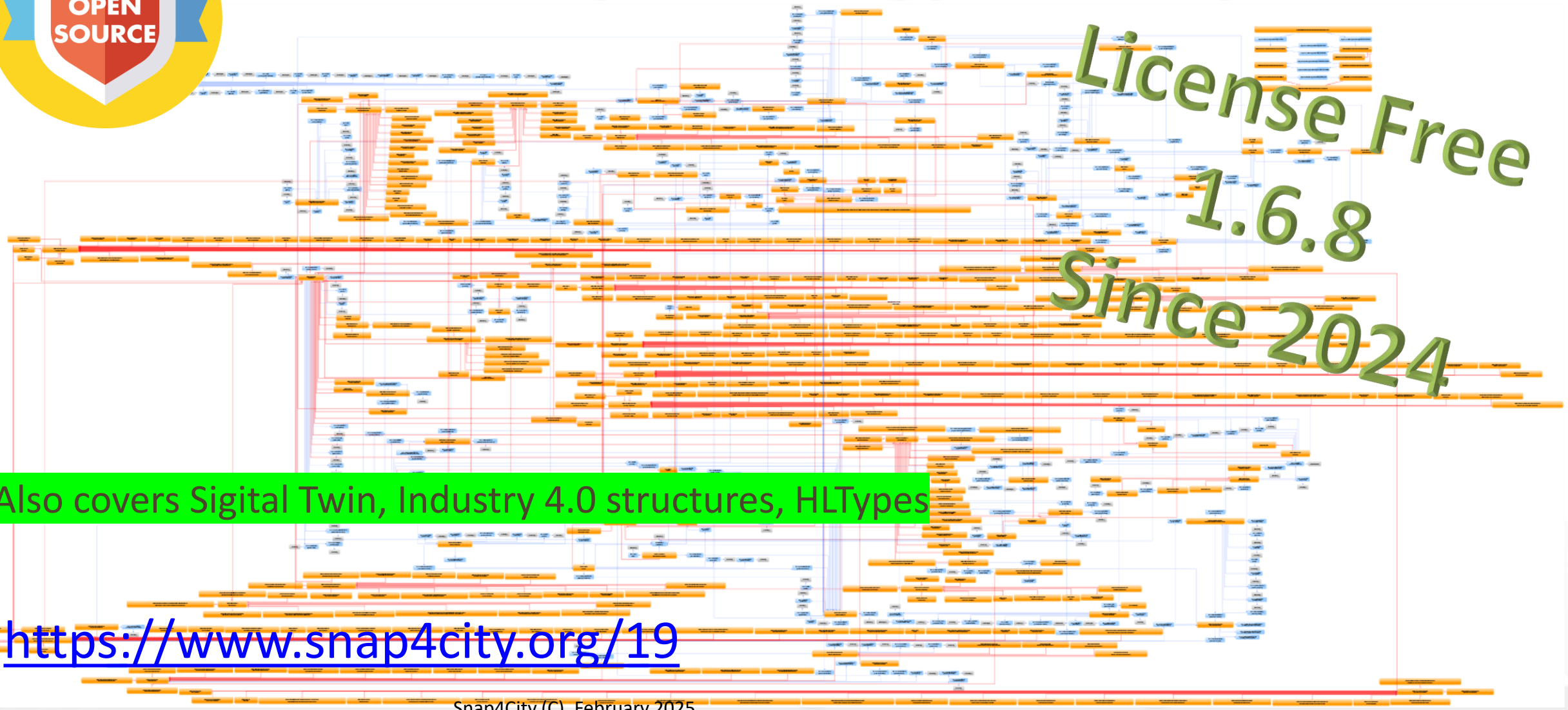


Km4City Ontology elements 1.6.8

- **Km4C**: Km4City 1.6.8
- Using
  - **DCTERMS**: for metadata Dublin Core Metadata Initiative
  - **FOAF**: friends of a friends
  - **Good Relation**: entities relationships
  - **iot-lite**: IOT Vocabuary
  - **OTN**: Ontology of Transportation Networks
  - **OWL-Time**: time reasoning
  - **SAREF** Smart Appliances REFerence extension for building devices available at https://saref.etsi.org/saref4bldg/
  - **Schema.org** for people and organizations
  - **SSN**: Semantic Sensor Network Ontology (see https://www.w3.org/TR/vocab-ssn/
  - **WGS84** Datum of Geo-Objects
  - **GTFS**, General Transit Feed Specification, and **Transmodel**, for public transport infrastructures: lines/rides time schedules, real-time records, paths, etc.;
  - **BOT**: Building Topology Ontology. https://w3c-lbd-cg.github.io/bot/
  - **S4CITY**: SAREF extension for Smart City. https://saref.etsi.org/saref4city/v1.1.2/

Smart-city Ontology km4city

100% OPEN SOURCE



License Free
1.6.8
Since 2024

Also covers Sigital Twin, Industry 4.0 structures, HLTypes

https://www.snap4city.org/19

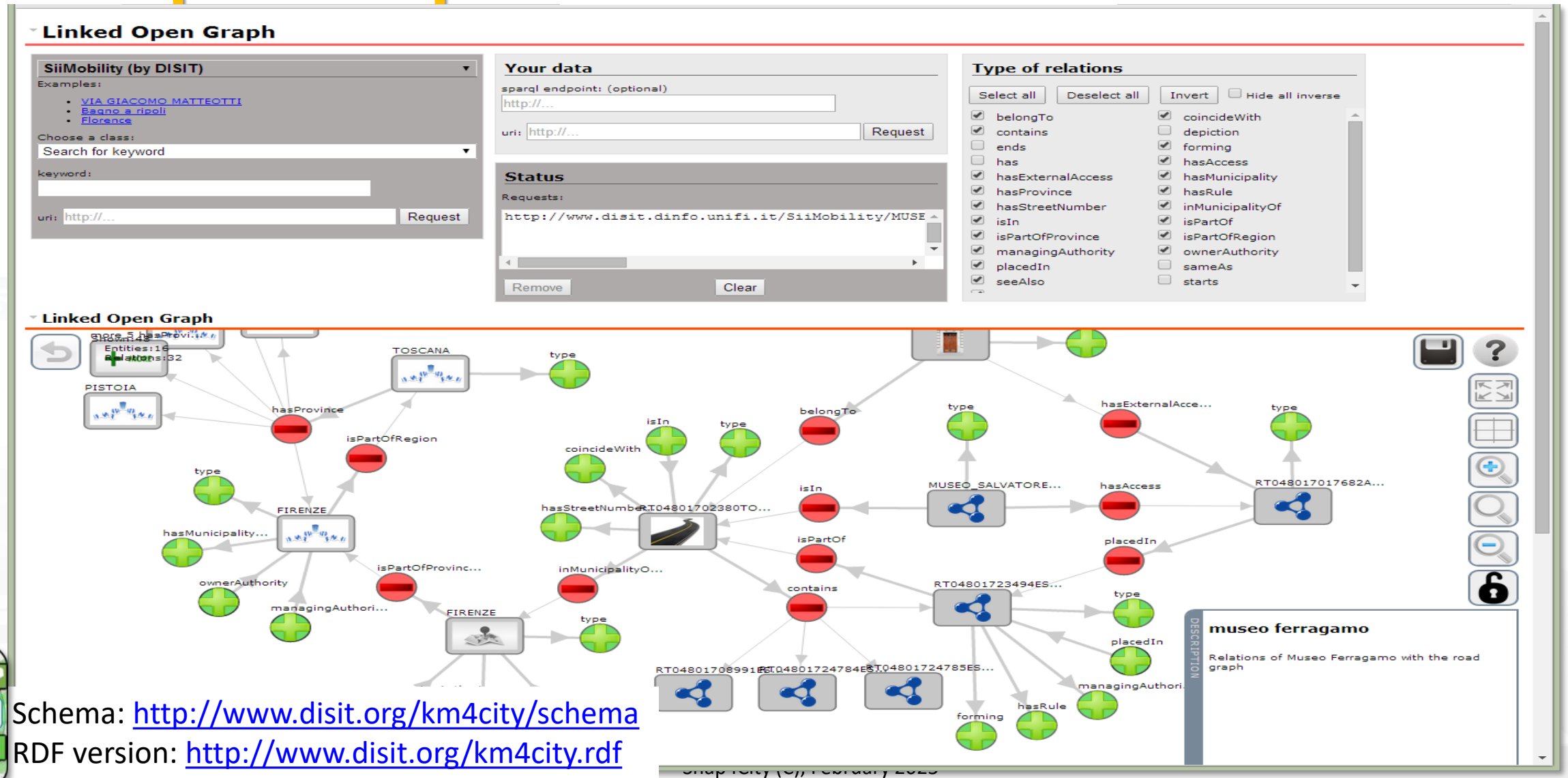
Set up of the Knowledge Base, KB

- The **KB starts with the ontology and empty in terms of instances**, it should to be **initialized** with the Road Graph(s) of interest, but may be not needed in some cases.
- **Road Graphs** can be obtained from:
  - GIS of the municipalities, regional govern, etc.
  - Open Street Map, OSM
  - Etc.
  - See this note on KM vs OSM: https://www.snap4city.org/397
- Snap4City provides a tool to **feed the KB with OSM, or edited OSM**
  - TC5.10- Open Street Map ingestion process
  - From the Open Street Map to the Km4City street graph
    - https://www.snap4city.org/download/video/From%20the%20Open%20Street%20Map%20to%20the%20Km4City%20street%20graph.pdf
  - **OSM2KM4C** tool is included into KBSM, VM and Docker https://www.snap4city.org/471
  - Tool: https://github.com/disit/osm2km4c
- The load of a city of 1 Million of inhabitants can be done in few hours.



Linked Open Graph

LOG: https://log.disit.org



Schema: http://www.disit.org/km4city/schema

RDF version: http://www.disit.org/km4city.rdf

The role of Knowledge base and ServiceMaps



Accessing to Snap4City Services

storage

Knowledge Base
Semantic Reasoners

SURI Link

Indexing and Aggregating
NIFI, OpenSearch

Data Managers
Entity Inspector

A number of KB federated

Federation Smart City API, ASCAPI

SCAPI

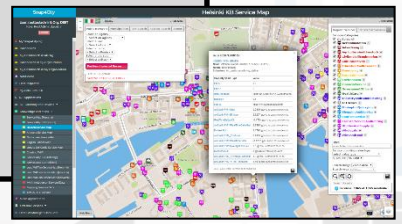
Super SCAPI

API Manager (Optional)

Cluster of OpenSearch for storage with in front a cluster of NIFI

Other federated Smart City

Control via API manager



ServiceMap

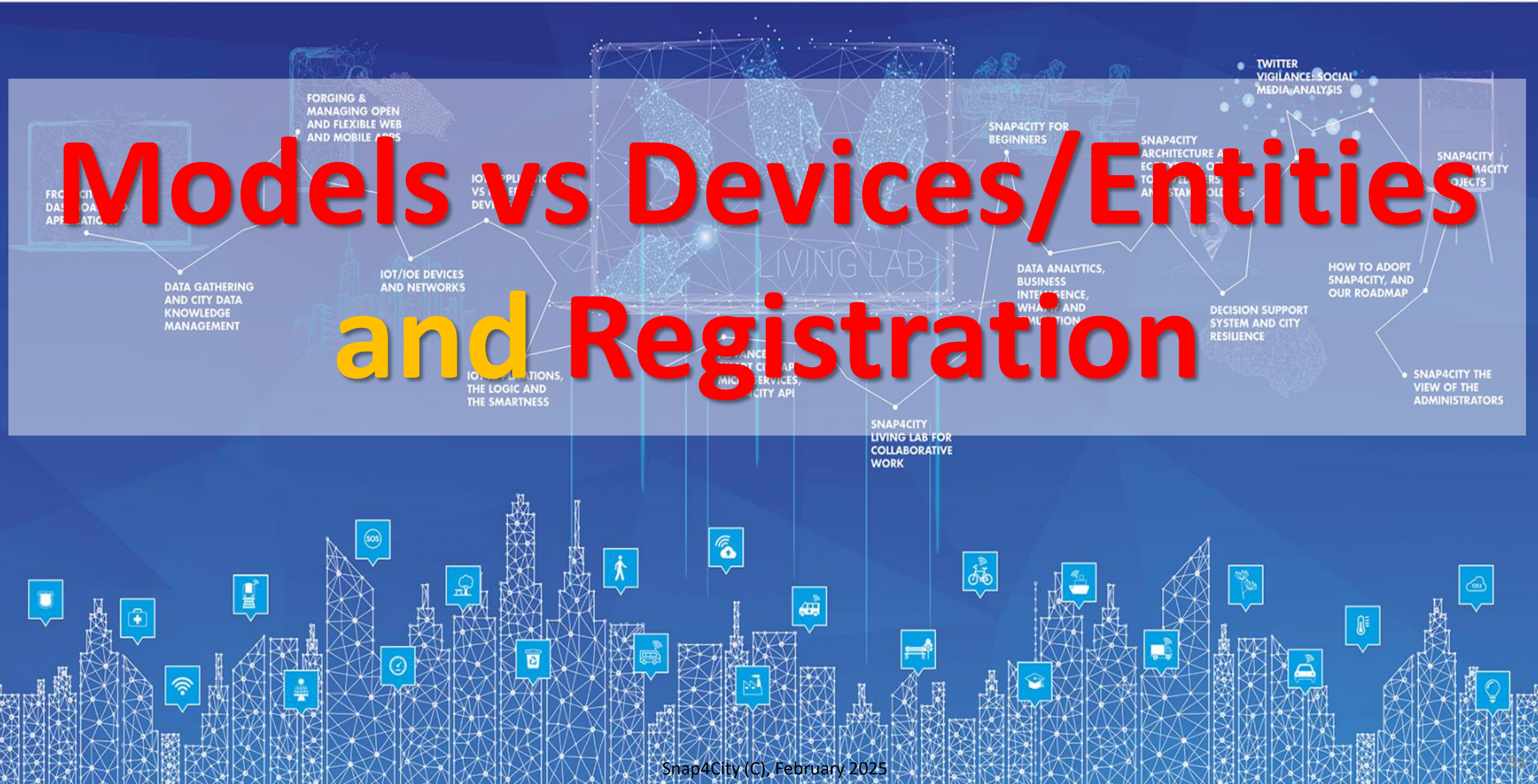


Multi/Super ServiceMap

ServiceMaps

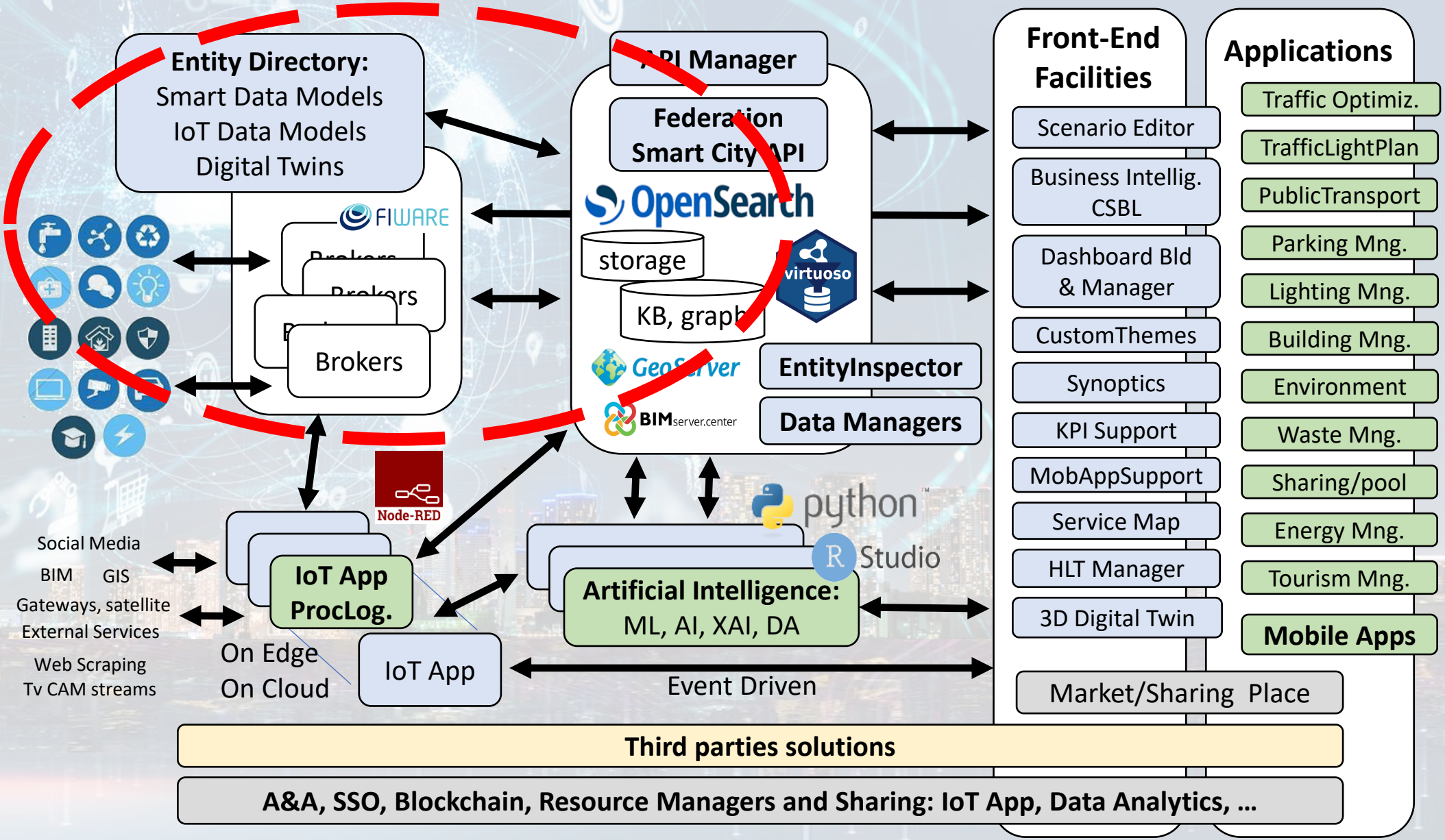
Knowledge Base
Semantic Reasoners

- ServiceMap is the **main Tool** to:
  - monitor the status of the Knowledge Base
  - test queries and produce query and SmartCity API testing calls for developers
    - Any kind of search (semantic, full text, etc.), routing,
  - Access at the specific Graph Data base via LOG.disit.org
- **ServiceMap is showing**:
  - only **public data**. Private data are not shown via ServiceMap but can be accessed via DataInspector, Wizard, Dashboards
  - data regarding **a single Knowledge Base** of the federated network of KBs. Each KB may contain multiple Organizations.
  - technical views for developers
- **Super ServiceMap** shows to you your private data and data which have been delegated in Access to you.
- In most cases we refer as ServiceMap to intend both Super and basic
- In the installations on Cloud, the Super is the Default used by Dashboards, on premise the basic ASCAPI are the default



Models vs Devices/Entities and Registration

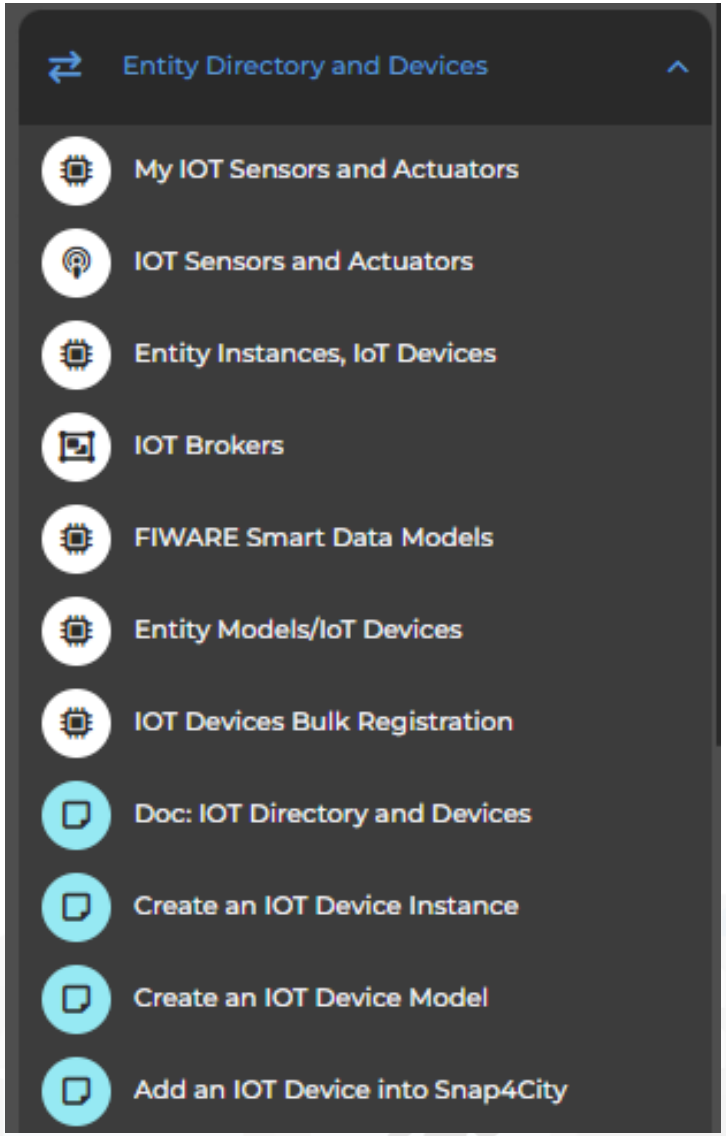
Technical Architecture

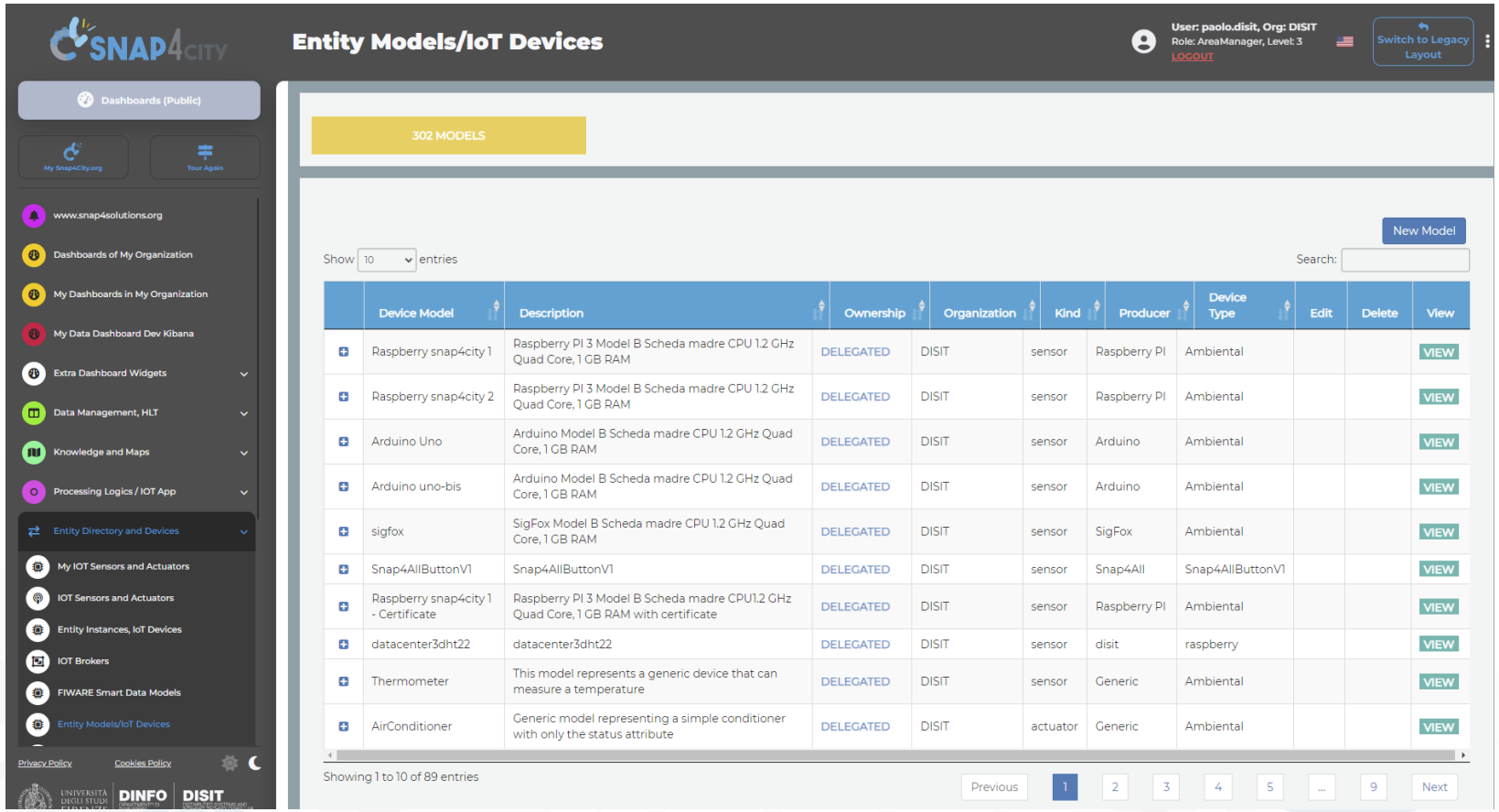


Directory Features vs Users Roles (01/25)

Entities	what	By using IoT/Entity Directory and:	Manager	AreaManager	ToolAdmin/ RootAdmin	Proc.Logic/IOT App microservices
Sensor/Actuator	Browse, use	Several Tools	X	X	X	Yes
	Delegate	API, ..	X	X	X	
	Discovery	KB, API, ..	X	X	X	Yes
Devices/Entities	Browse, use	Several Tools	X	X	X	Yes (use)
	Create, change, delete	API, ..	X	X	X	Yes
	Register in Bulk	API, ..		X	X	Yes
	Delegate, Change Owner	API, ..	X	X	X	Yes
	Discovery	KB, API, ..	X	X	X	Yes
Models (S4C, Fiware)	Browse, Use		X	X	X	(Yes)
	Create, change, delete		X	X	X	(Yes)
	delegate, change ownership		X	X	X	Yes
Brokers	Browse, use		use	Browse, use	X	Yes (use)
	Register/change/Delete				X	
	Deploy Orion Broker				ToolAdmin	
	Delegate				X	
	Periodic Update				X	

Entity / Iot Directory: User Role





Entity Directory for Beginners

- Browse and see models and entities/devices of other users, that publiched them
- Create your Entity / Device Models
  - User the dictionary approach: value type, data type, value unit
  - Manage delegation of the models and ownership
- Create your Entities / Devices from scratch and/or from models
  - Several models are ready to be used…..
- Send a Message to a Device, thus to the broker
- Read a Message from the Broker, see the message forma expected to be sent at the Broker in NGSI format

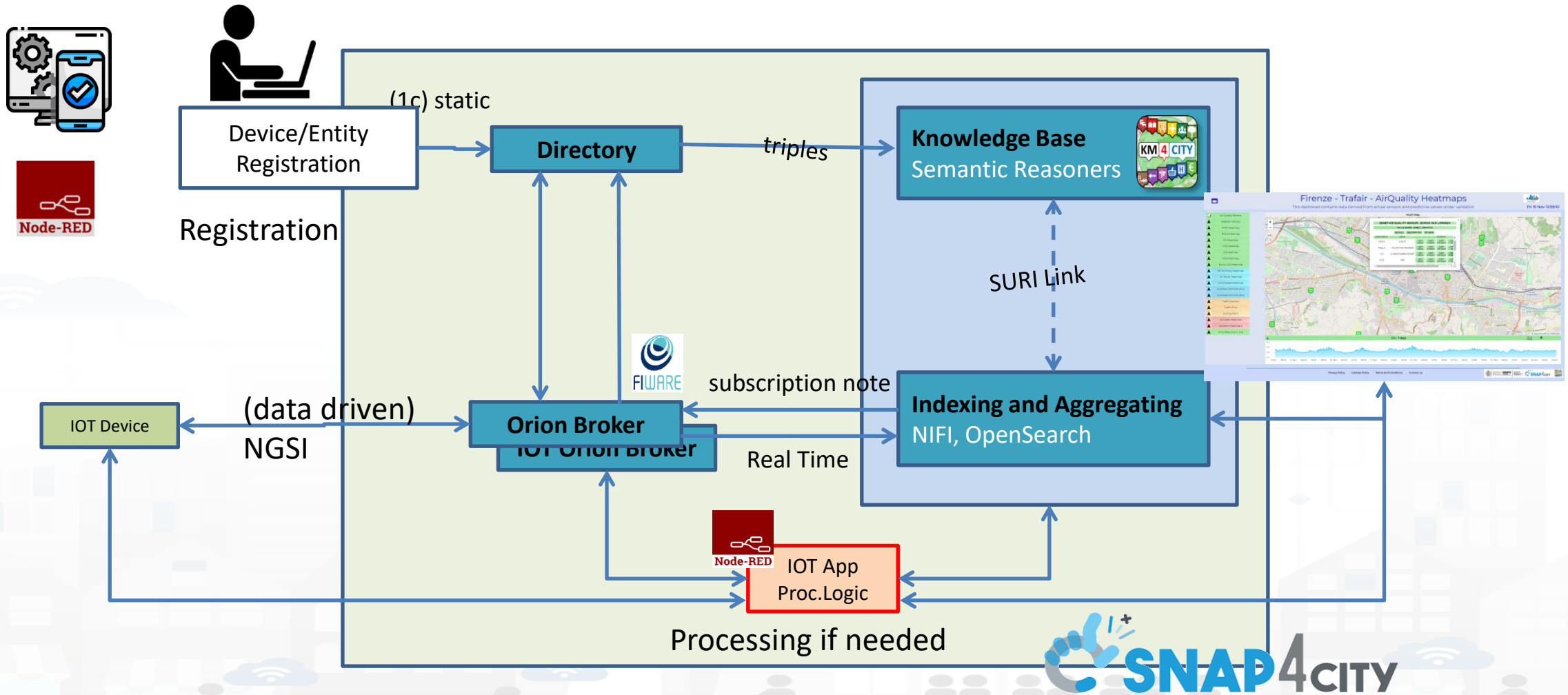
Entity/Device Registration

many possibilities

- IOT Directory and Devices
  - My IOT Sensors and Actuators
  - IOT Sensors and Actuators
  - IOT Devices
  - IOT Devices Management
  - IOT Device Discovery
  - IOT Brokers
  - IOT Device Models
  - IOT Devices Bulk Registration
  - IOT Broker Periodic Update setti...
  - IOT Orion Broker Mapping Rules...
  - Doc: IOT Directory and Devices
  - Create an IOT Device Instance
  - Create an IOT Device Model
  - Add an IOT Device into Snap4Cit...

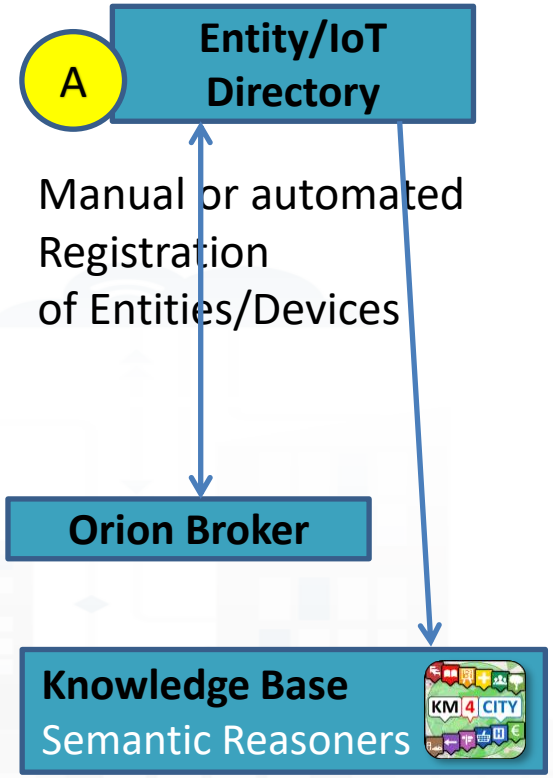


- NGSI devices can be directly connected to Snap4City (data driven)



Benefits of Registration on Directory

- **The registration implies the automated production of the Digital Twin Device into the Knowledge Base**
  - Registration of the Entity on Brokers and KB
  - Activation of the Storage “DataShadow” for historical data access
  - Activation of all the relationships
  - Activation of Discovery mechanisms via Entity Directory, KB and SCAPI, etc.
  - Activation of Dashboard Wizard (after a few minutes), and Data Inspector



Activities for Registration on Directory

- **Manual Registration**
  - From scratch Single Device / Entity Registration → **Entity Directory / IoT Directory**
  - From a template (the templates are called Models)
- **Automated Registration for bulk/massive registration: N** Entities / Devices
  - From IoT App/Proc.Logic on the basis of some **Models** from IoT App
  - From IoT App/Proc.Logic **loading a CSV** (with or without a reference IoT/Entity Model)
  - Programming from scratch or from a **Model**
  - On the basis of some EXCEL file with data by using the **Data Table Loader, which create model, devices and data**
  - Etc.



Manual or automated Registration of Entities/Devices

Entity Instances, IoT Devices

Snap4City

Switch To New Layout (Beta)

User: paolo.disit, Org: DISIT
Role: AreaManager, Level: 3

LOGOUT

- My Snap4City.org
- Tour Again
- www.snap4solutions.org
- Dashboards (Public)
- Dashboards of My Organization
- My Dashboards in My Organization
- My Data Dashboard Dev Kibana
- Extra Dashboard Widgets
- Data Management, HLT
- Knowledge and Maps
- Processing Logics / IOT App
- Entity Directory and Devices
  - My IOT Sensors and Actuators
  - IOT Sensors and Actuators
  - Entity Instances, IoT Devices
  - IOT Brokers
  - FIWARE Smart Data Models
  - Entity Models/IoT Devices
  - IOT Devices Bulk Registration
  - Doc: IOT Directory and Devices
  - Create an IOT Device Instance
  - Create an IOT Device Model

Show delegated dev. | Show public dev. | Show my dev. | Show all dev

Add new device

Show entries

Search:

	Device Identifier	IOT Broker	Device Type	Model	Ownership	Status	Edit	Delete	Location	View
+	1dd79caa95f6771afad4fd38e699c8542022-12-05T18:54:13.000Z	orionUNIFI	File	fileModel	MYOWNPUBLIC	active	EDIT	DELETE		VIEW
+	alert_1610543238306	orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
+	alert_1610548534047	orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
+	alert_1610613189703	orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
+	alert_1610629197473	orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
		orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
		orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
		orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
		orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW
		orionUNIFI	event	AlertGeneric	MYOWNPRIVATE	active	EDIT	DELETE		VIEW

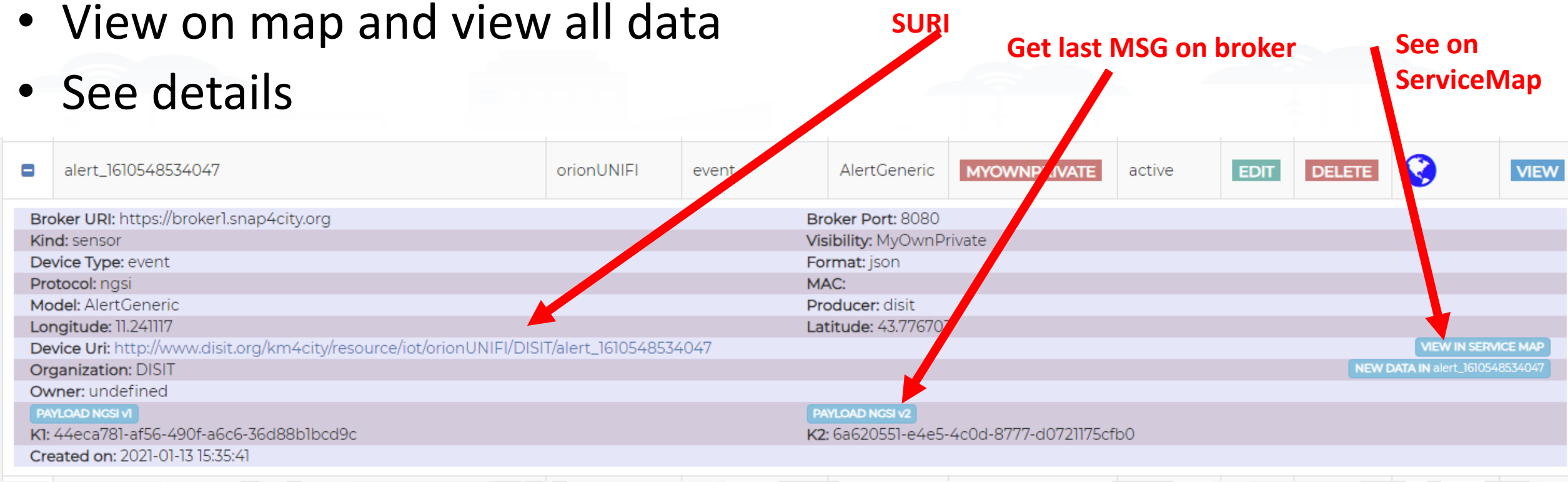
Previous 1 2 3 4 5 ... 12 Next



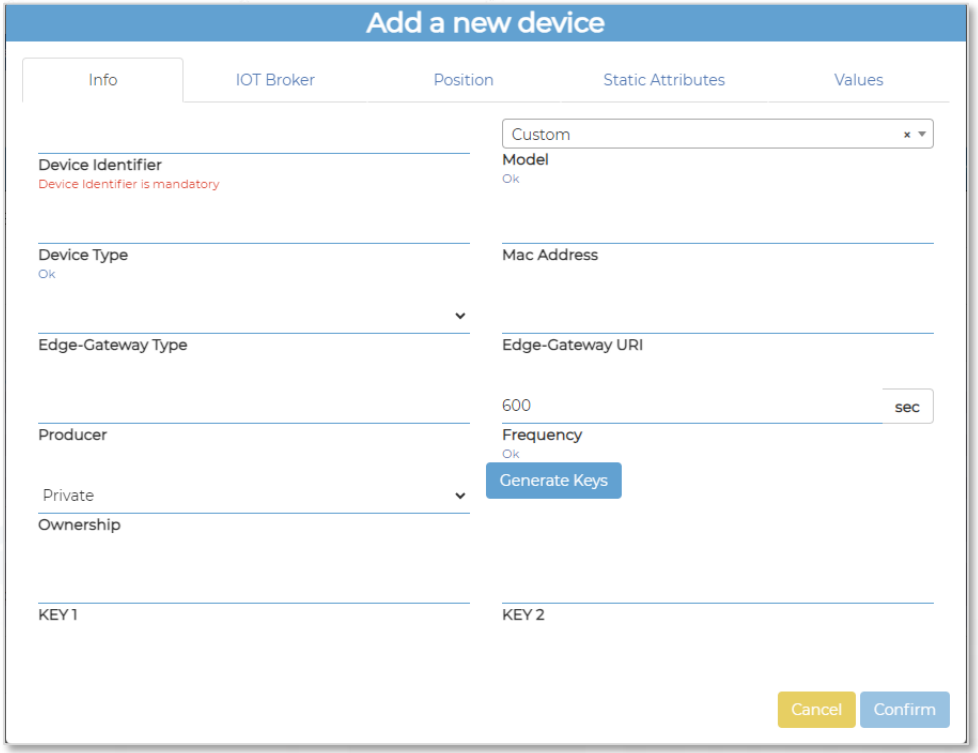
What you can do ?

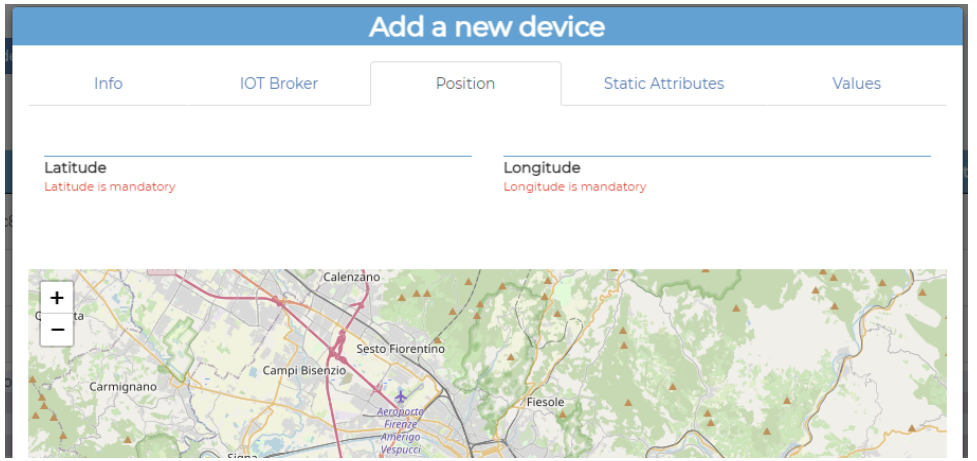
- List and browse your devices and those received in delegation
- Change ownership, control the delegation
- Edit, Change, delete all paramerters
- View on map and view all data
- See details

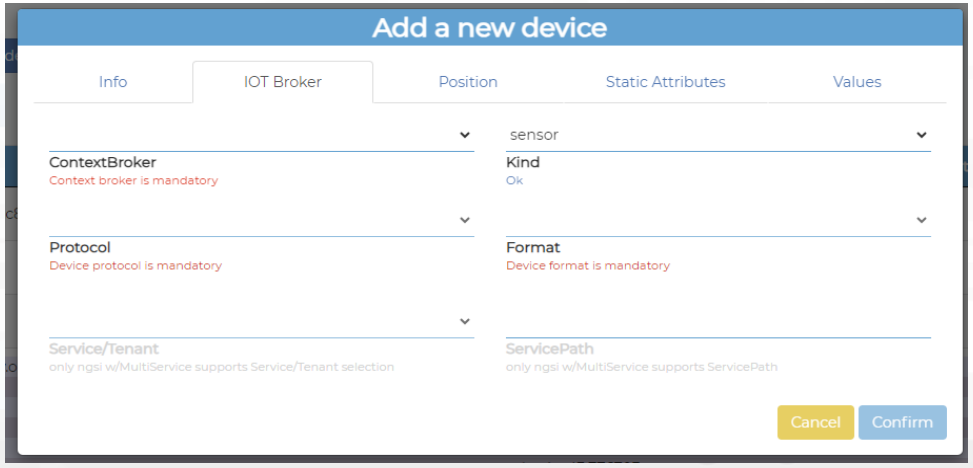
Entity/IoT Directory

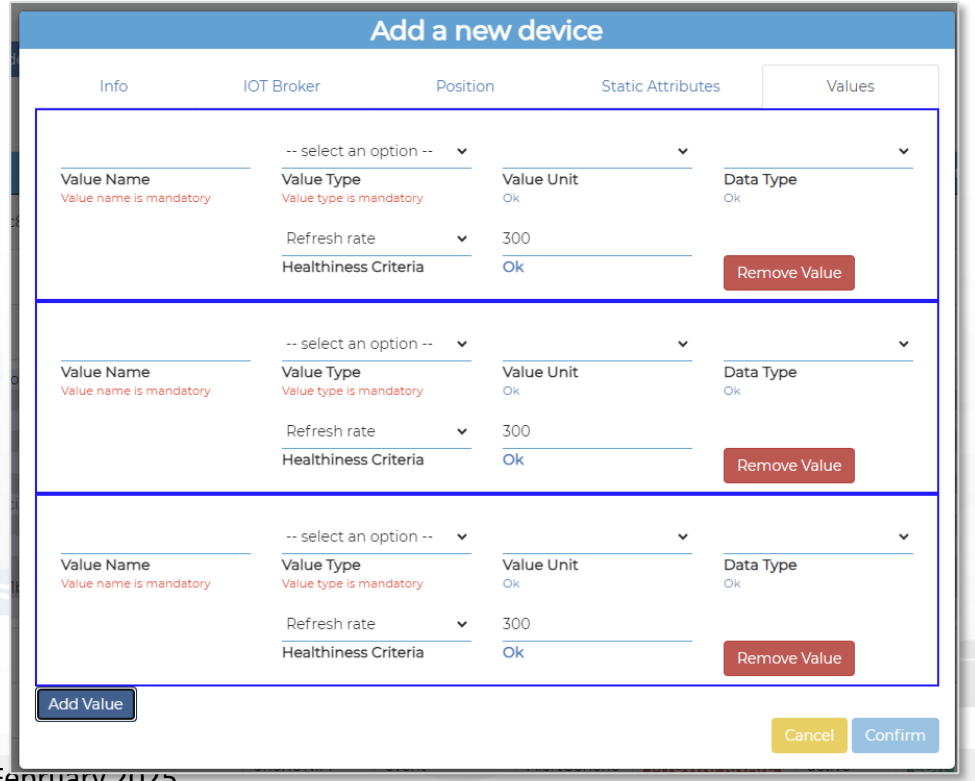


Add New Device





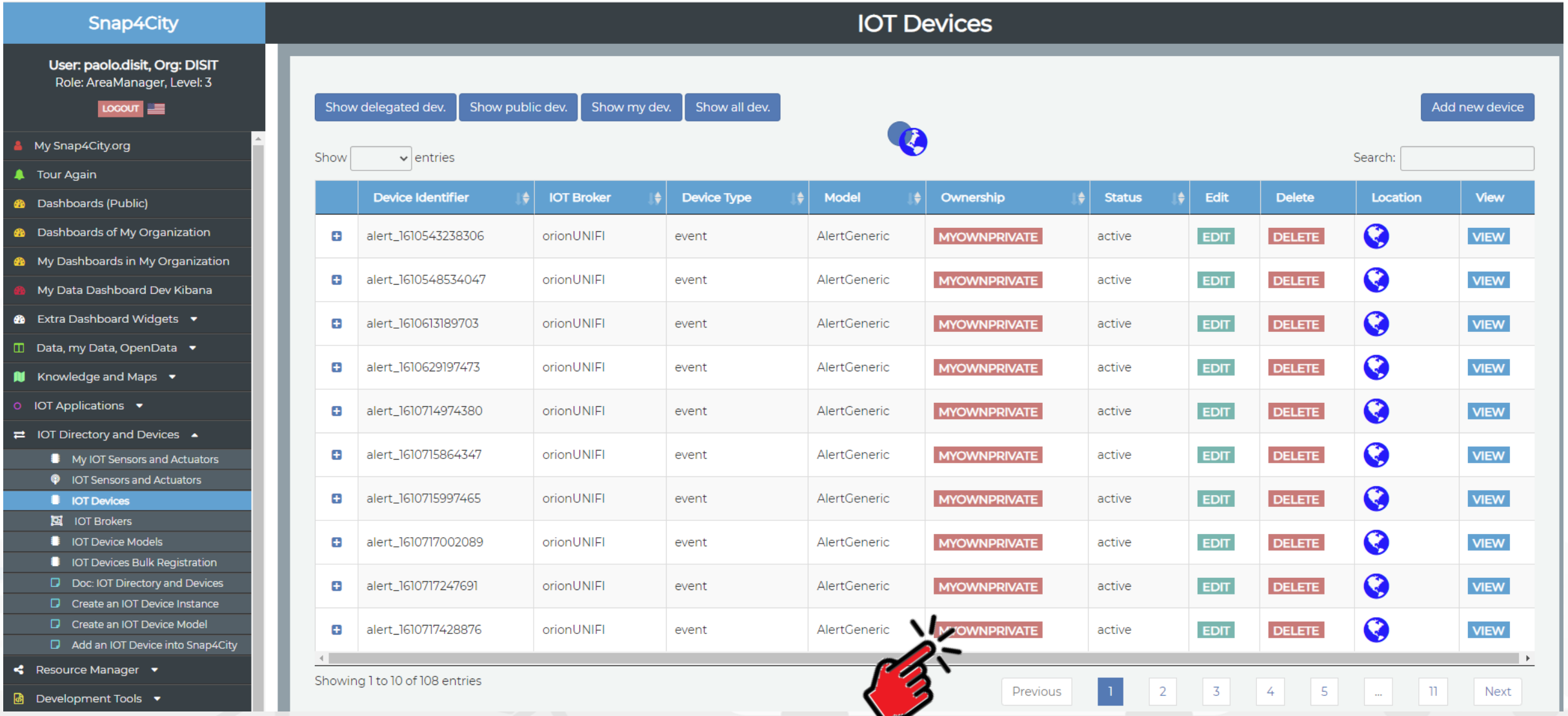




Entity / Device: Attributes

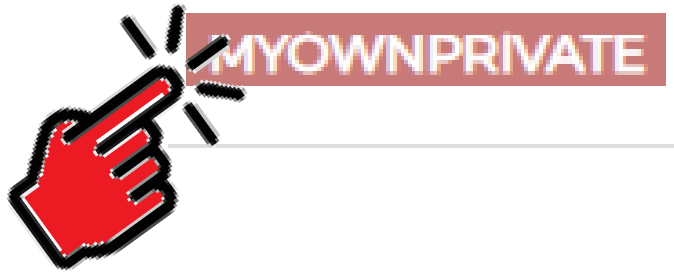
Where	IOT Device	AT 23-12-2019T20:13:12…	AT 23-12-2019T22:13:12…
IOT Broker	**Broker: OrionUNIFI**	--	
IOT Broker	**Protocol: NGSI**	--	
Info	ID: "park45"	**park45**	**park45**
Position	GSP Position: 43.12, 11.34	GSP Position: 44.12, 11.12	GSP Position: 44.14, 11.13
Static attribute	Description: "parking massaia"	--	
Static attribute	Location: "Via Massaia"	--	
Static attribute	Civic Number: 3	--	
Static attribute	MaxCapacity: 456	--	
Values	**dateObserved: Timestamp**	23-12-2019T20:13:12…	23-12-2019T22:13:12…
Values	**FreeSlots: Integer, #**	345	356
Values	**Humidity: float, %**	25,5	25,5
Values	**Temperature: float, celsius**	34	

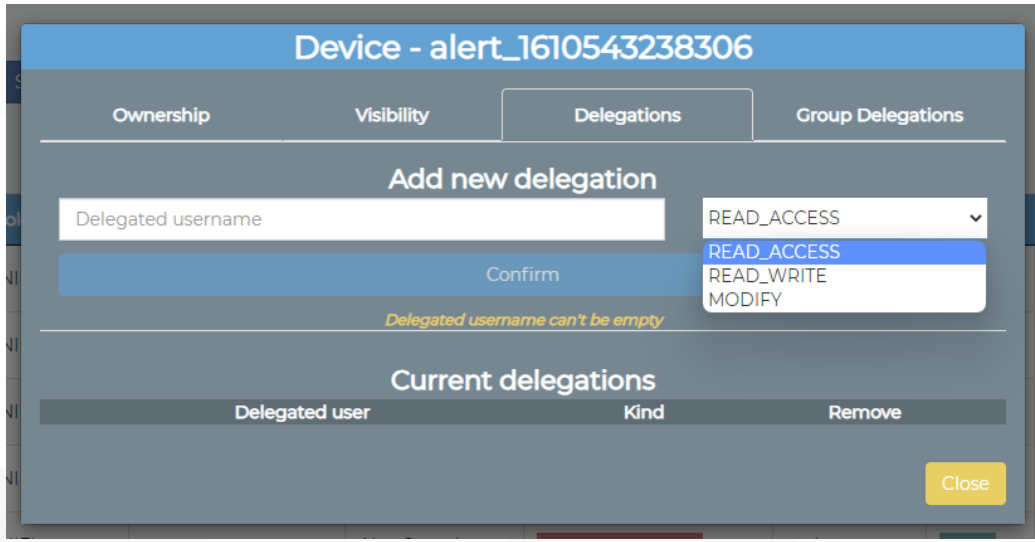
IoT Device Management for All, Developers



Developers can

- Manage their own Devices
  - Edit/Modify, Delete, view, send messages, etc.
  - Pass the ownership of a Device to another user
  - Delegate to other users/groups in
    - READ_ACCESS the IoT Device
    - READ_WRITE the IoT Device
    - MODIFY the IoT Device
    - WRITE_ONLY the IoT Device (is coming)
  - See and change the Delegations
- See Delegated IoT Devices/Entities, …
- See Public IoT Devices/Entities, …





What they mean ?

Supposing that User User45 has the Device D34 !

THUS: User45 can delegate Device D34 at Users DD12, DD13 for

- **READ_ACCESS**. This means that User DD12 can read the values/data of Device D34, real time and historical
- **READ_WRITE**. This means that User DD12 can
  - read the values/data of Device D34, real time and historical
  - Send messages to the broker to add them for device DD12
- **WRITE_ONLY.** This means that User DD12, DD13 can send data to the Device D34, while they cannot read back the messages sent
- **MODIFY**. This means that User DD12 can Modify structure of Device D34, for example: changing the name of variables, etc.

Design: from Data Modelling to Data Ingestion



Using the Entity/Device Model notes!!!

- Once performed the Entity/Device Model, a number of Entities/Devices can be produce **using the model as a Template**



  - **NOTE:** the produced Entities/Devices are not going to change if the Entity/Device Model is modified.
  - *Your biscuit is not changing if the template is modified after the printout*

The Data Models can be simply instantiated from

a) **Entity Model / IoT Device Model** which are accessible into the Snap4City environment

b) Creating a **custom Entity Model / IoT Device Model** in standard Snap4City format via **Entity Directory** / **IoT Directory**

c) **FIWARE Smart Data Models**, versioning, and harvesting the standard repository

Connections among Entities

Where	Entity Model (IOT Device Model)	Entity Instance (IOT Device)	Entity Message at 23-12-2019T20:15:00	Entity Message at 23-12-2019T20:30:12
Broker	**Broker: OrionUNIFI**			
Broker	**Protocol: NGSI**			
Info	**ID: string**	ID: "park45"	**park45**	**park45**
Position	**GPS: lat, long**	GSP: 43.12, 11.34	GSP: 44.1256, 11.1234	GSP: 44.1259, 11.1233
Static attribute	**Description: string**	Description: "parking massaia"		
Static attribute	**MyAddInfoSURI: string**	**MyAddInfoSURI**: "http://...../InfoPersonal"		
Values	**dateObserved: Timestamp**		23-12-2019T20:15:00	23-12-2019T20:30:12
Values	**FreeSlots: Integer, #**		**FreeSlots:** 345	**FreeSlots:** 234
Values	**TodayCarSURI: string**		**TodayCarSURI:** "http://...../CarNF126GD"	**TodayCarSURI:** "http://...../CarGF789KK"
Values	**Temperature: float, celsius**		34	34

Model meaning



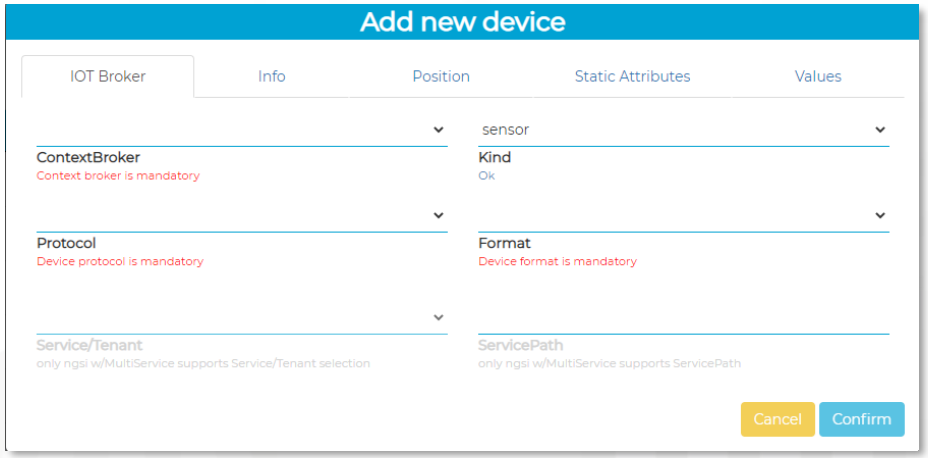
- **ID:** is the unique identifier for reconnecting Temporal Instances with registered Entity / Devices
- **Static Attributes:**
  - Are typically associated with instances of the IOT Device.
    E.g.:, You have a set of parking areas, each of them is located in a specific street, and has its one name, etc.
  - Different kinds of attributes can be set for each SubNature. Their definition has to be prepared into the Knowledge Base ☺ for automated indexing.
- **Values**: they are time varying variables (temporal values/instances)
  - They change over time, the timestamp of the time series is conventionally «dateObserved» in Snap4City
  - In new *SensorMobile* HLT, also GPS can be changing over time as in the MyKPI
- **NOTE for:**
  - **names/IDs**: Spaces or strange characters are not allowed in the. Please use simple alfphanumeric strings, it is a limitation of many solutions including Orion Broker and increase interoperability of your data.
  - **Values of attributes and variables**: can be UTF8, but similarly, they do not accept: () <> “ ‘ ; = into values
  - https://fiware-orion.readthedocs.io/en/master/user/forbidden_characters/index.html

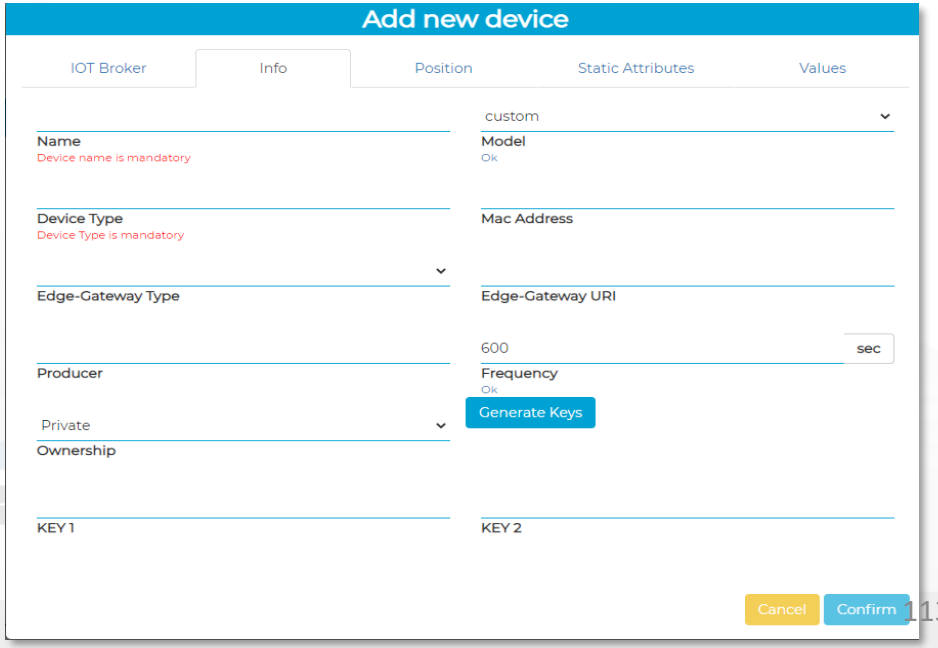
Entity / Device Model (1)



- IOT Broker
  - Name of the Brokers: among those registered
  - Protocol: NGSI, AMQP, MQTT, etc..
  - Format: CSV, JSON, XML.
  - Service/Tenant:…….
  - ServicePath:……

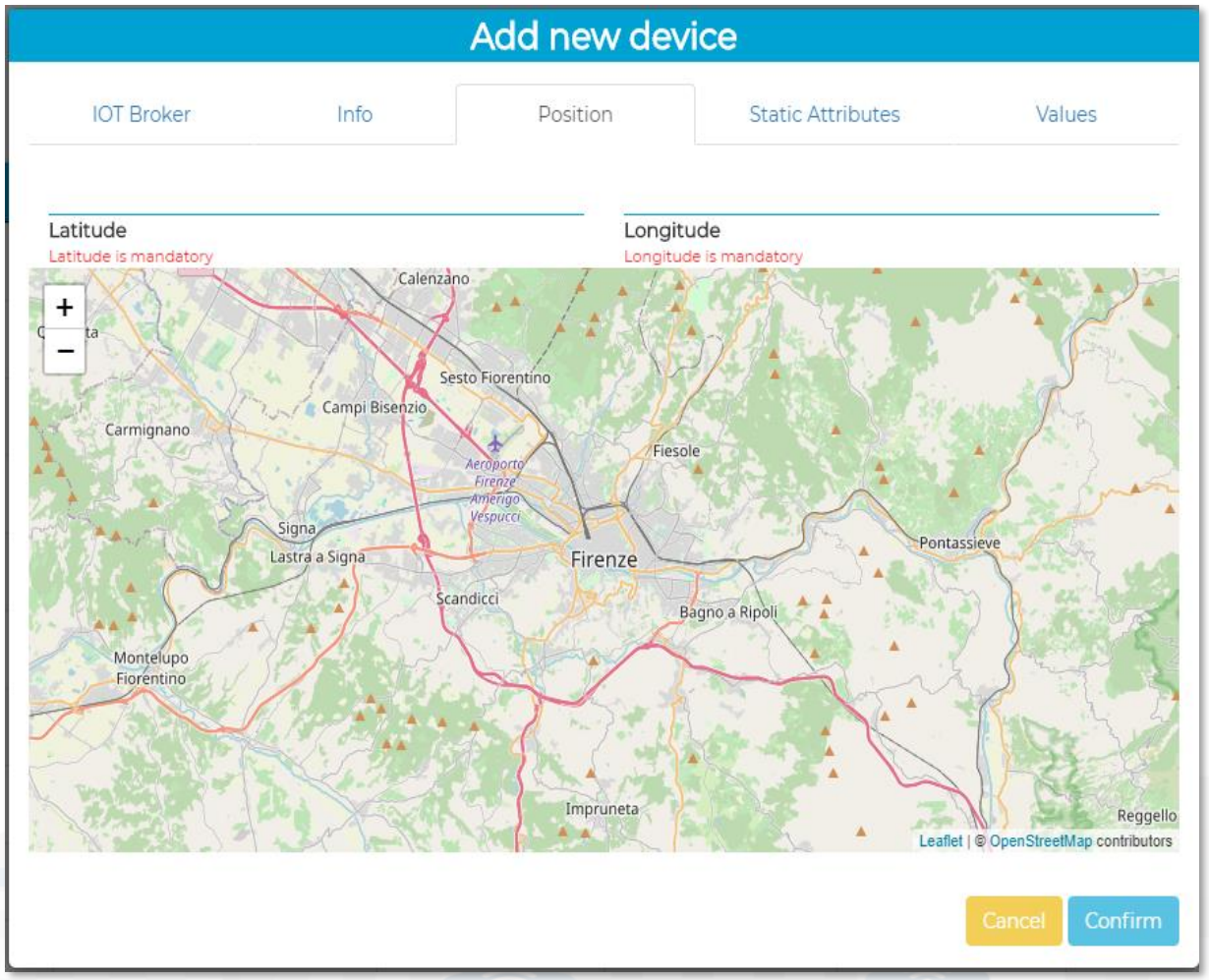
- Info
  - Name (Identifier)
  - Model: Custom or Model ID
  - DeviceType: ..a string..
  - MAC address: …optional…
  - Edge-GW: Raspberry, Android, …
  - Edge-GW: URI
  - Producer
  - Owner
  - Freq: ….. Sec
  - Keys: K1, K2

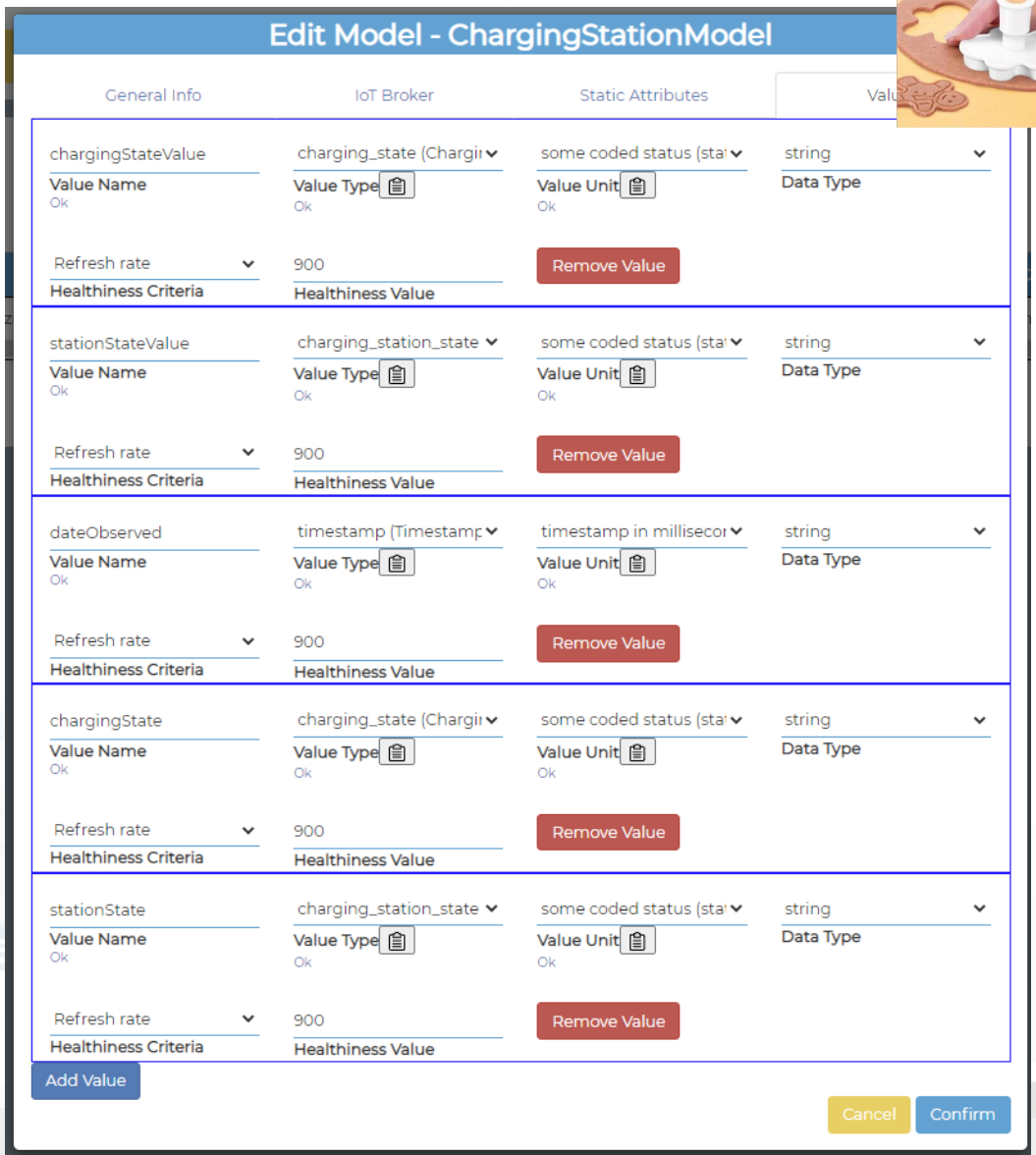




Entity / Device Data Model (2)







SURI Connections

From a

- *Static* Attribute of an Entity Instance to another Entity Instance, as highlighted in green in previous table.
- *Dynamic* Value/Variable of an Entity Message of an Entity Instance to another Entity Instance, as highlighted in green in previous table.

- *the example reports a*
  - *static connection and*
  - *dynamic connection to change the car at a given timestamp, note also change of position and other parameters, if needed*

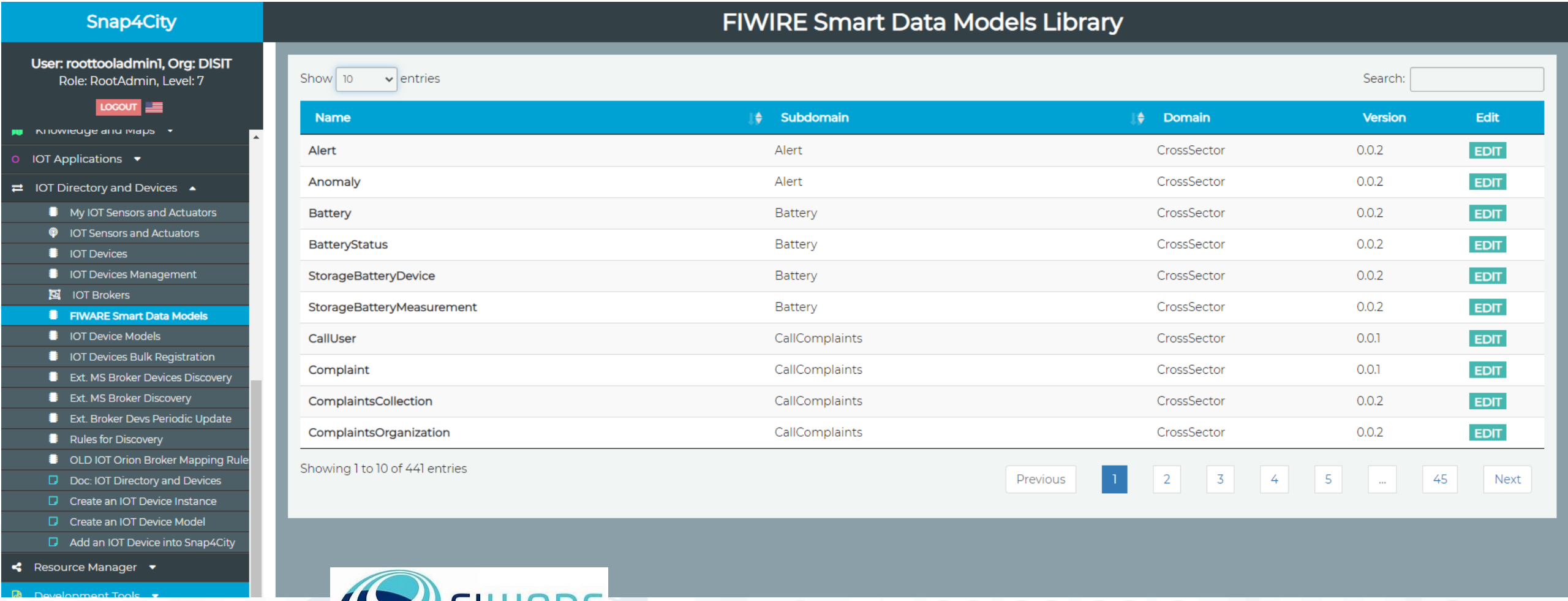
Non sicuro | disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759

App Maps Google Gmail Snap4City Snap4 Calendar Translate Google Scholar Cita... DISIT DISIT old Facebook DataCe

p	o
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/ns/sosa/Sensor
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.disit.org/km4city/schema#Traffic_sensor
http://www.w3.org/ns/ssn/implements	http://www.disit.org/km4city/resource/iot/traffic
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/avgDistance
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/occupancy
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/thresholdPerc
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/speedPercentile
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/dateObserved
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/avgTime
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/concentration
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/vehicleFlow
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/averageSpeed
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/congestionLevel
http://www.disit.org/km4city/schema#hasAttribute	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/anomalyLevel
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/average_vehicle_distance
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/average_vehicle_speed
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/average_vehicle_time
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/vehicle_concentration
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/vehicle_speed_percentile
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/vehicle_threshold_perc
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/vehicle_flow
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/timestamp
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/anomaly_level
http://www.w3.org/ns/sosa/observes	http://www.disit.org/km4city/resource/value_type/traffic_congestion
http://www.w3.org/ns/ssn/hasSystemCapability	http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO759/systemCapability
http://purl.oclc.org/NET/UNIS/fiware/iot-lite#exposedBy	http://www.disit.org/km4city/resource/iot/orionUNIFI
http://www.disit.org/km4city/schema#protocol	"ngsi"
http://www.disit.org/km4city/schema#format	"json"
http://www.w3.org/2003/01/geo/wgs84_pos#long	11.25673
http://schema.org/addressLocality	"FIRENZE"
http://schema.org/name	"METRO759"
http://schema.org/streetAddress	"Lavagnini P.zza Della Liberta' (38)"
http://www.w3.org/2003/01/geo/wgs84_pos#lat	43.78278
http://www.disit.org/km4city/schema#isInRoad	http://www.disit.org/km4city/resource/RT04801703772TO
http://www.w3.org/2003/01/geo/wgs84_pos#geometry	"POINT(11.256730079651 43.782779693604)"^^<http://www.openlinksw.com/schemas/virtrdf#Geometry>
http://www.disit.org/km4city/schema#model	"metrotrafficsensor"
http://www.disit.org/km4city/schema#producer	"metro"
ma#organization	"DISIT"

www.disit.org/km4city/resource/iot/traffic

FIWARE Smart Data Models -- Library



- Snap4City - Powered by FIWARE Solution & Platform:
  - https://www.fiware.org/marketplace/product-details/?category=powered&id=snap4city-snap4city
  - NGSI V1, V2 The IOT Orion Broker
  - IOT Orion Broker can connect JSON, MQTT, Lightweight M2M, LoraWAN, OPC, SigFOX, etc. see FiWare Https://www.fiware.org
- Snap4City - FIWARE Training Services:
  - https://marketplace.fiware.org/pages/solutions/03bccd83a0e1b0398ba7a0bf
- Snap4City - FIWARE Consultancy Services:
  - https://marketplace.fiware.org/pages/solutions/907f5ecc63927f643dd8421b
- **Snap4City is compatible** with all the above protocols
  - via IOT Orion Broker,
  - via IOT Applications.
  - via direct connection on ETL processes on their corresponding IOT brokers, and/or
- **Snap4City is also compatible** with many other protocols, see the table reported in page: https://www.snap4city.org/65

Powered by FIWARE

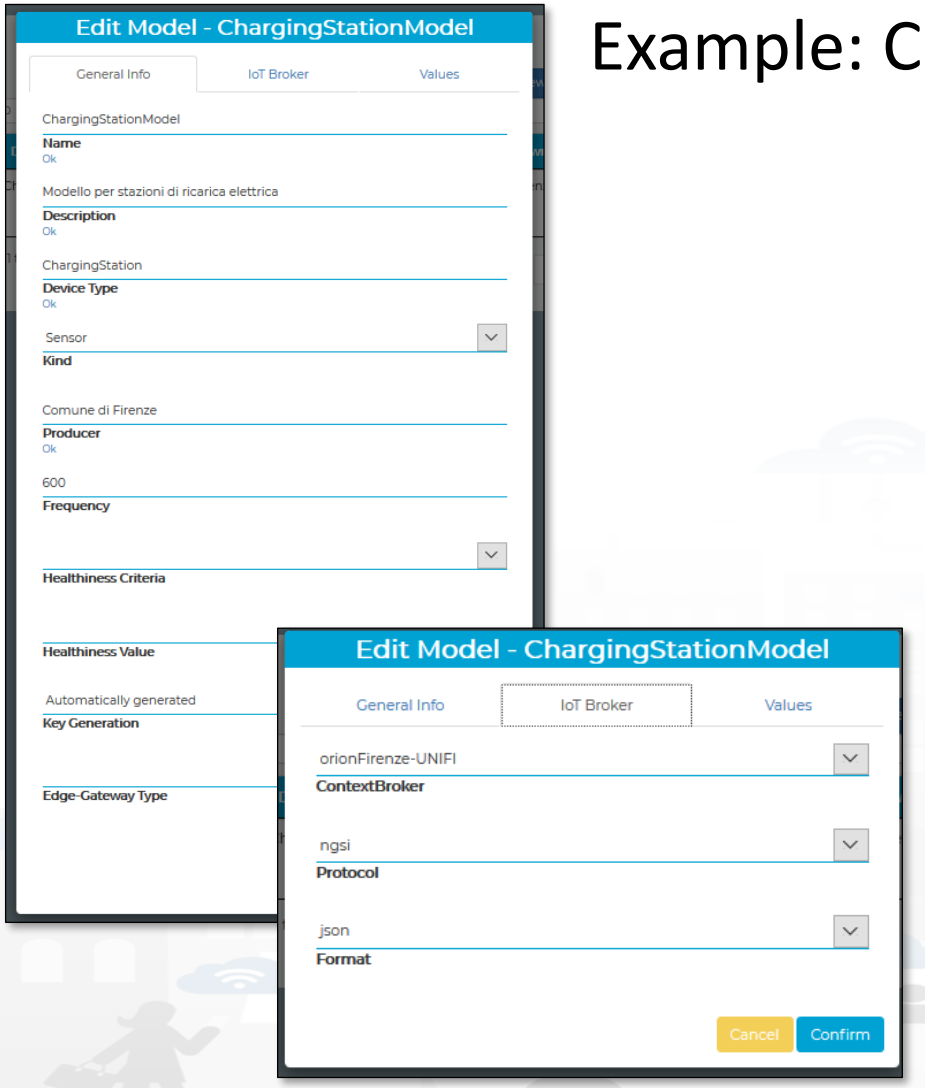
Entity / Device Registration from Model

- IOT Directory and Devices
  - My IOT Sensors and Actuators
  - IOT Sensors and Actuators
  - IOT Devices
  - IOT Devices Management
  - IOT Device Discovery
  - IOT Brokers
  - IOT Device Models
  - IOT Devices Bulk Registration
  - IOT Broker Periodic Update setti...
  - IOT Orion Broker Mapping Rules...
  - Doc: IOT Directory and Devices
  - Create an IOT Device Instance
  - Create an IOT Device Model
  - Add an IOT Device into Snap4Cit...

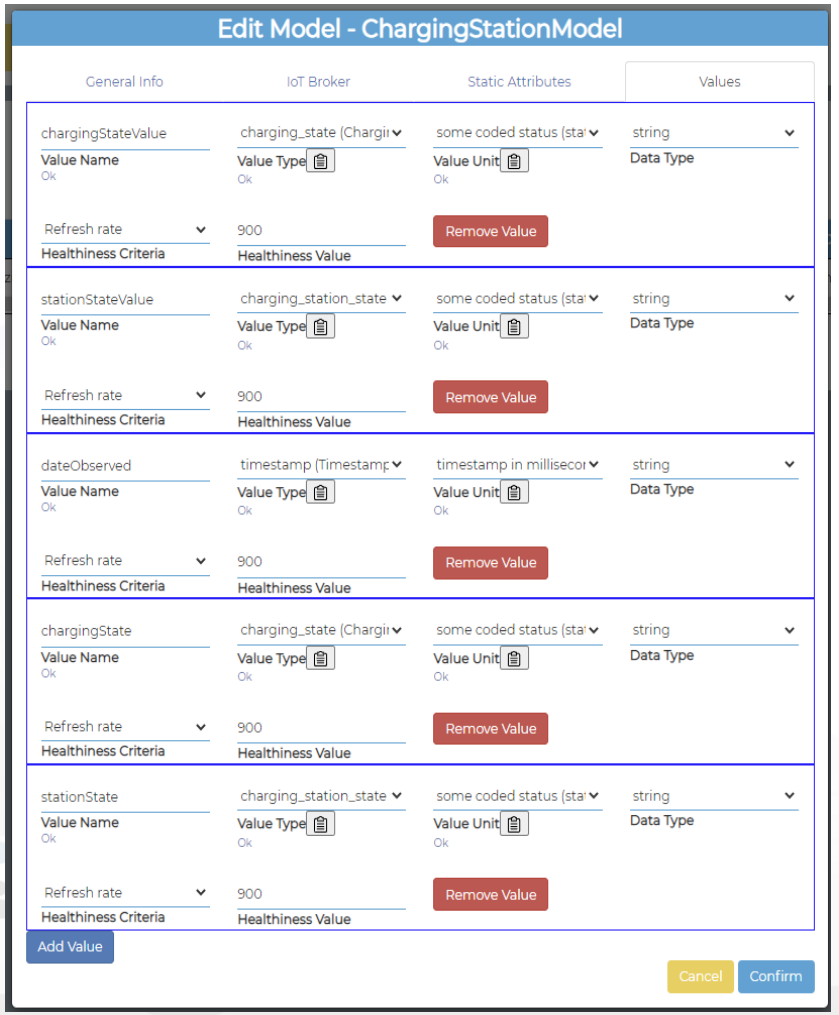


Many IoT Devices? IOT Device Model!!!

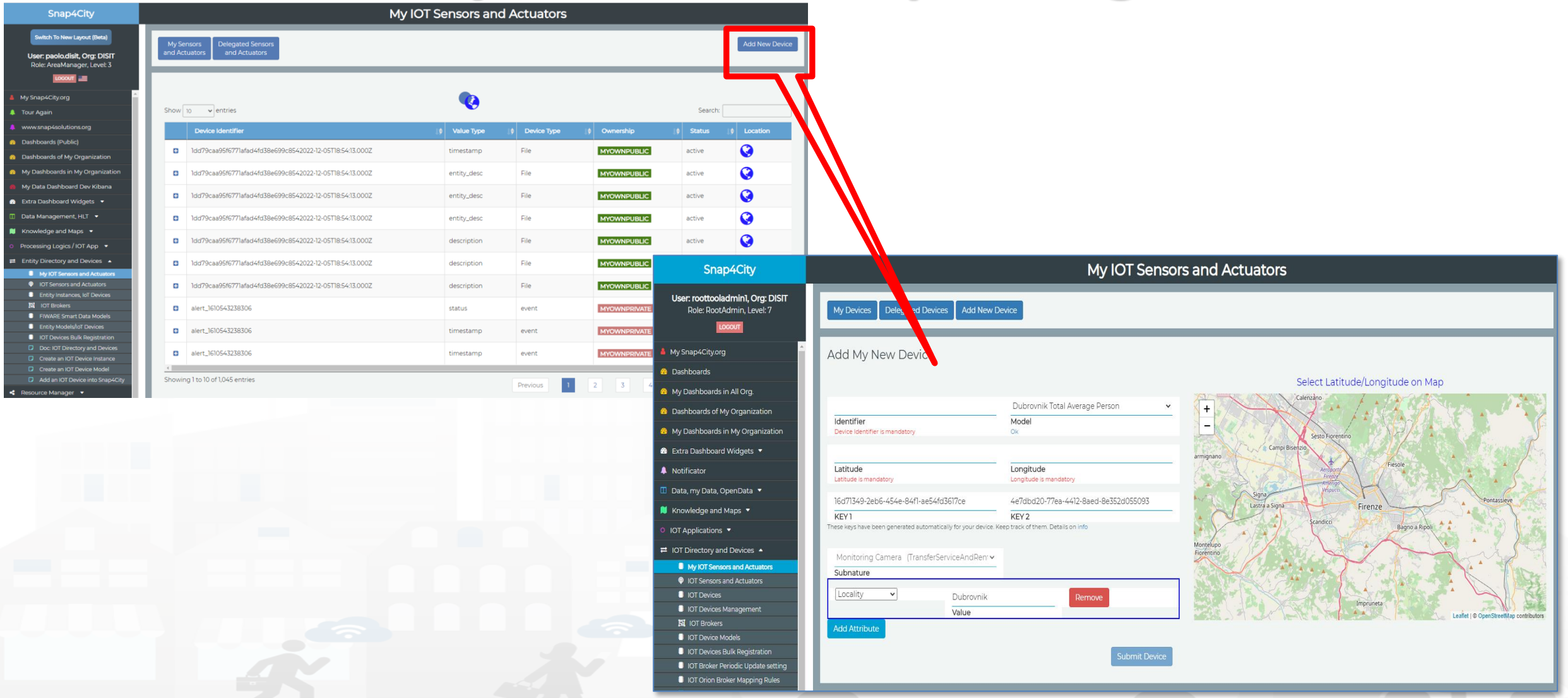
- **Prerequirements**: only for AreaManager users
- If you have a set of sensors with the same features,
  - you can create a model and then a set of instances (IoT Devices) in compliance with the model (not time consuming and avoiding errors)
- IoT Directory and Devices > IoT Device Models > ‘New Model’ button



Example: ChargingStationModel



Add Entity / Devices, exploiting a Model



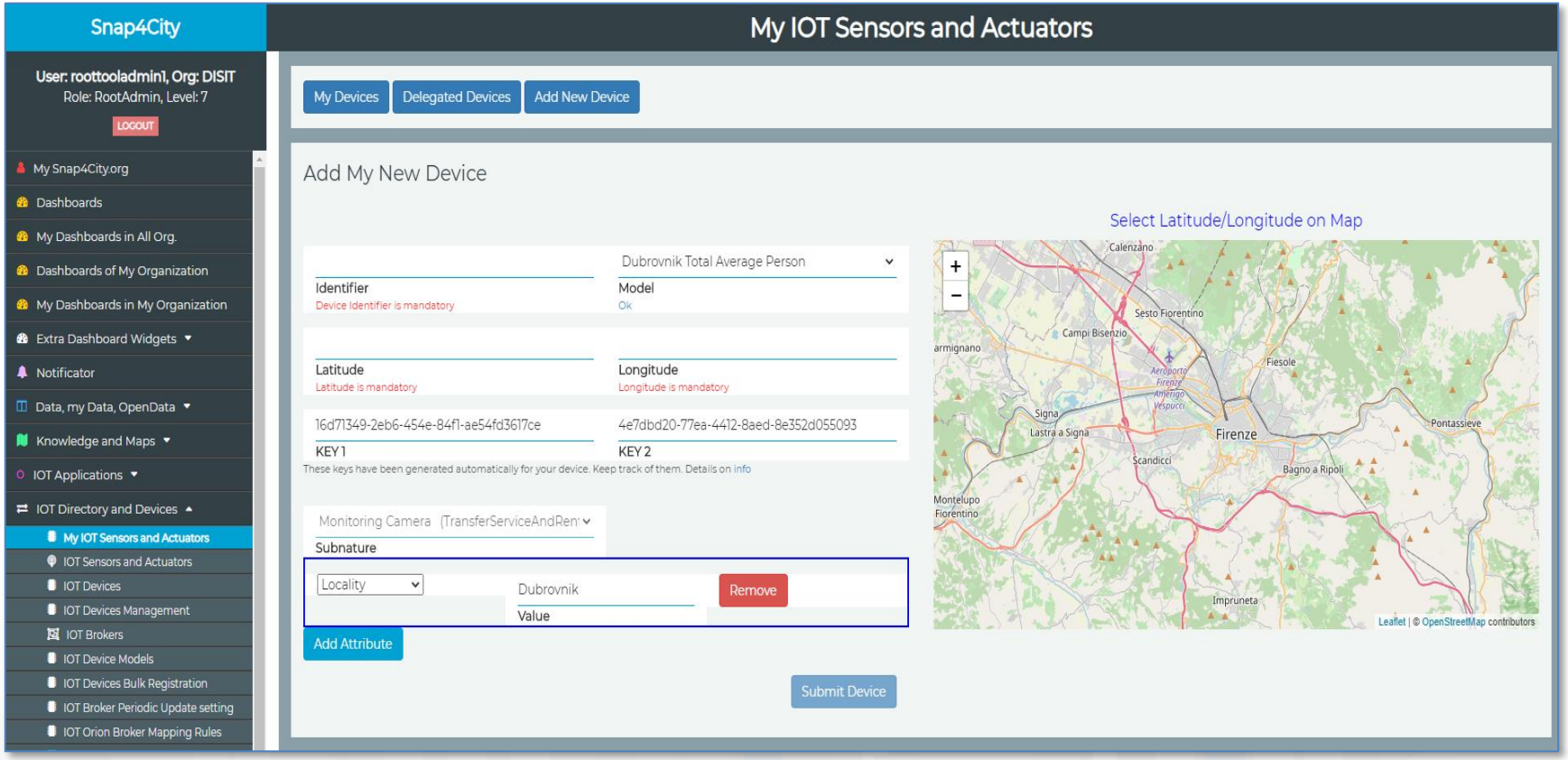
Add Entity / Devices, exploiting a Model

Just Buy an IOT Device and register: SigFOX, MQTT, NGSI (FiWare), …

- Attach them by
  - Models
- A range of protocols, formats, approaches

Create your own devices:

- Arduino,
- Raspberry,
- Android,
- LoraWAN + Arduino,
- etc.

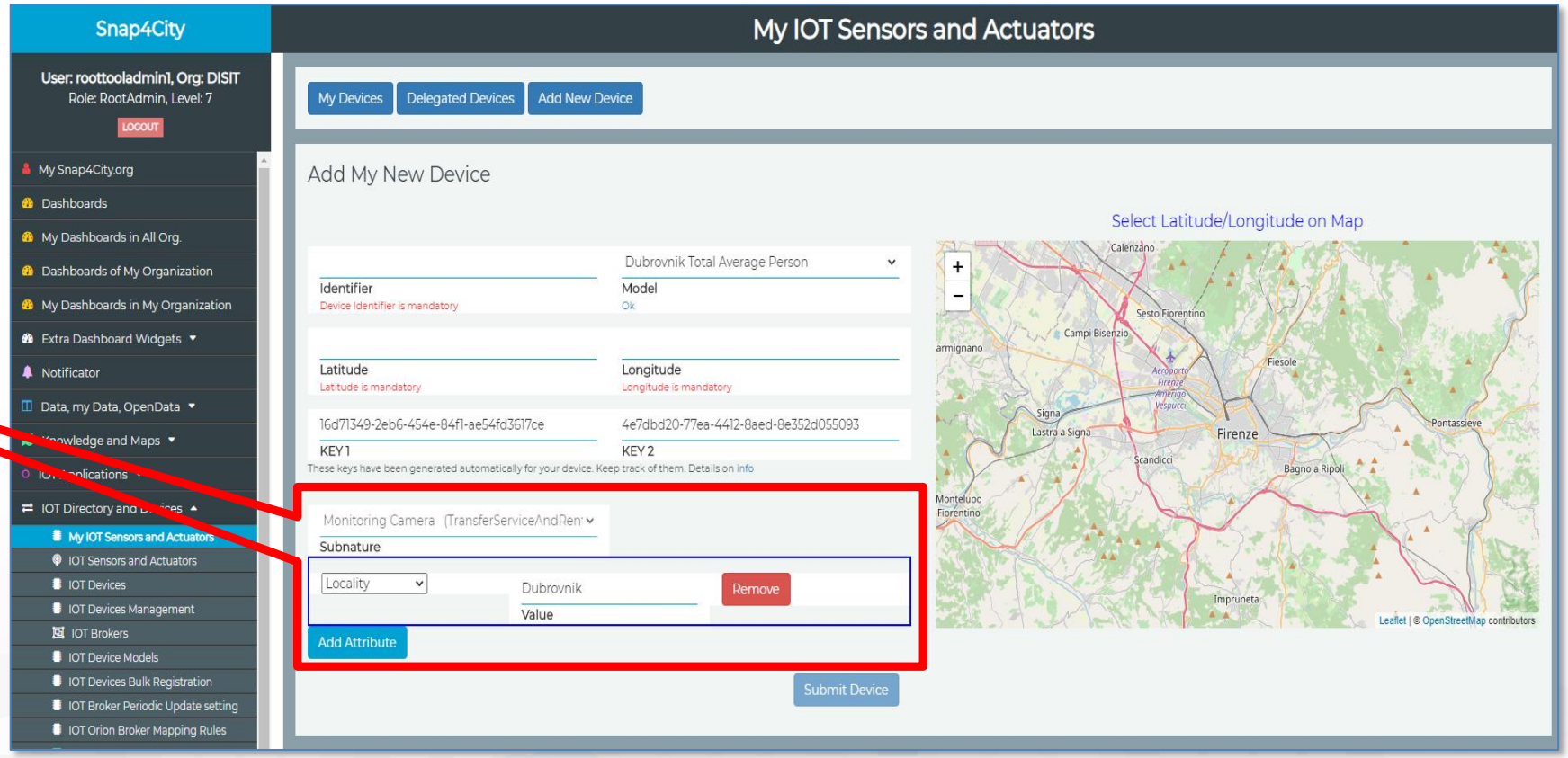


Secure Communication: HTTPS, TLS (K1, K2), Certificates

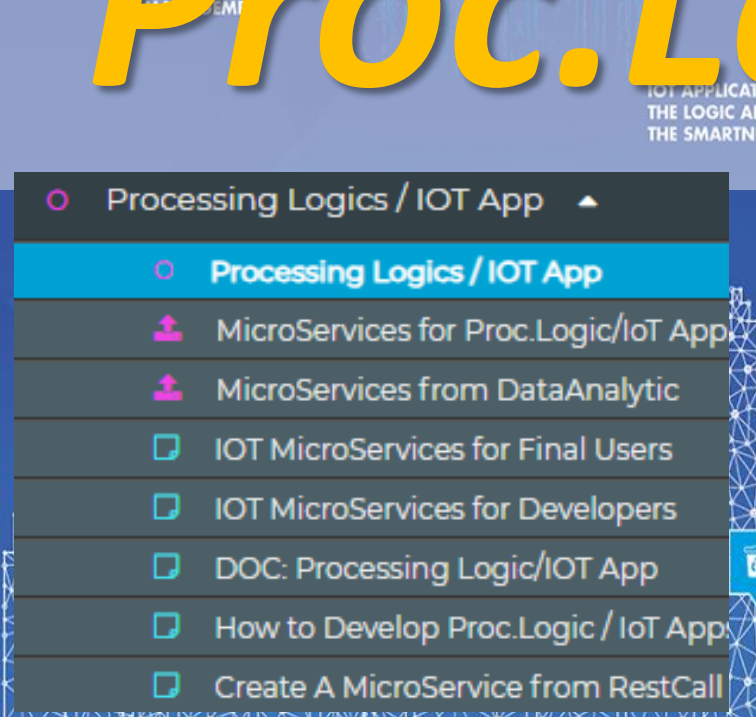
Add Entity / Devices, exploiting a Model

Addition of Static Attributes of the Entity Instance / IoT Device

Only if you enabled from model



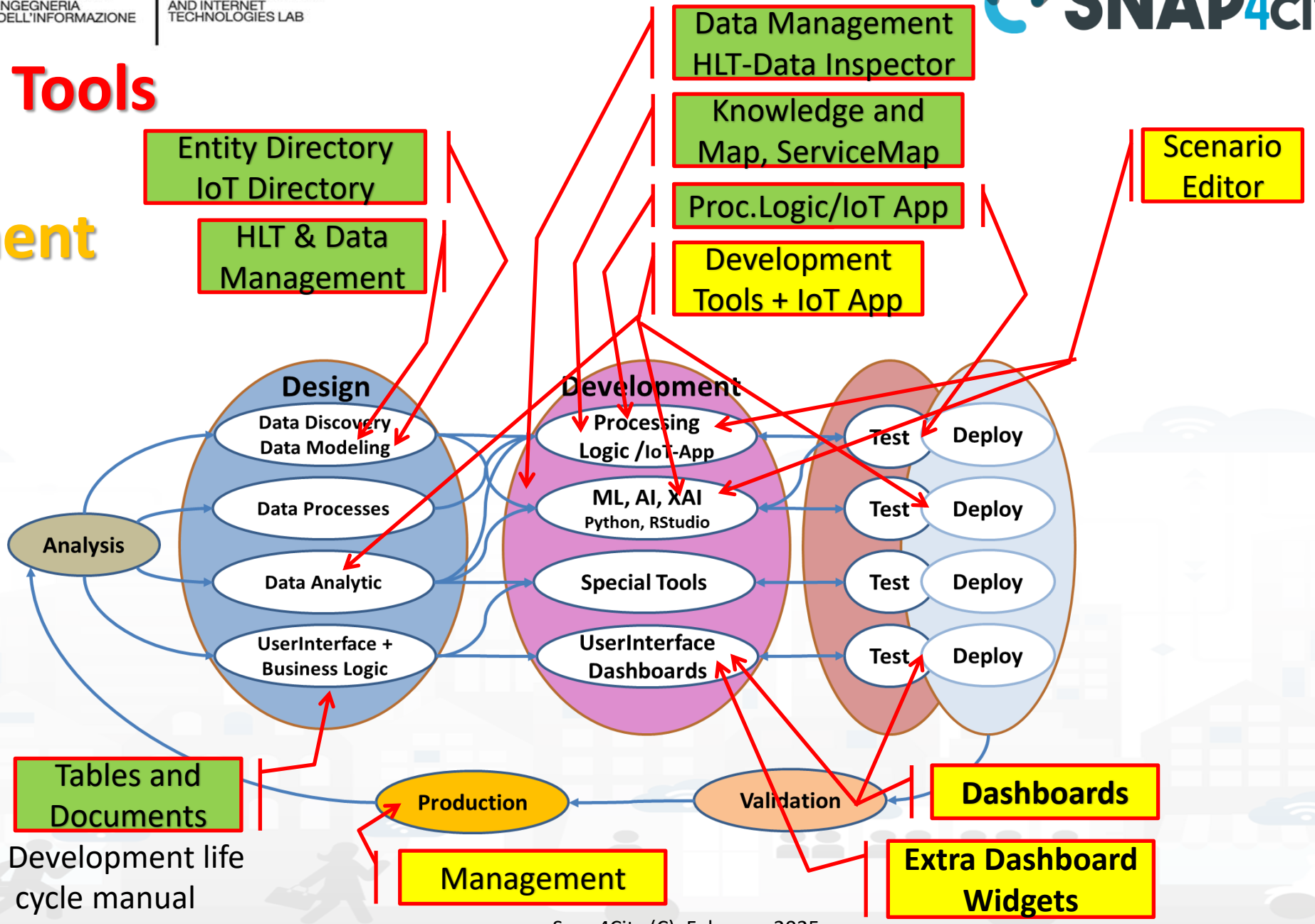
Develop: Data Processes
Proc.Logic / IoT App



Agenda of this integrated overview part

- Recall of Snap4City Architecture
- Data Ingestion Strategy and Orientation
  - Basic entity elements
  - Knowledge Base: Modelling and ServiceURI as Entity Identifier
  - Models vs Devices/Entities and Registration
- Develop: Data Processes Proc.Logic / IoT App
  - Proc.Logic = Node-RED + Snap4City
  - An Integrated Example for Time Series
  - Verification of Data Ingestion
  - Exploiting Storage data by using: IoT App / Proc. Logic
  - Interoperability with respect to Hardware staff
  - High Performance Ingestion
  - Interoperability of Snap4City Platform
- Some Applicative examples
- Training Material
- Acknowledgements

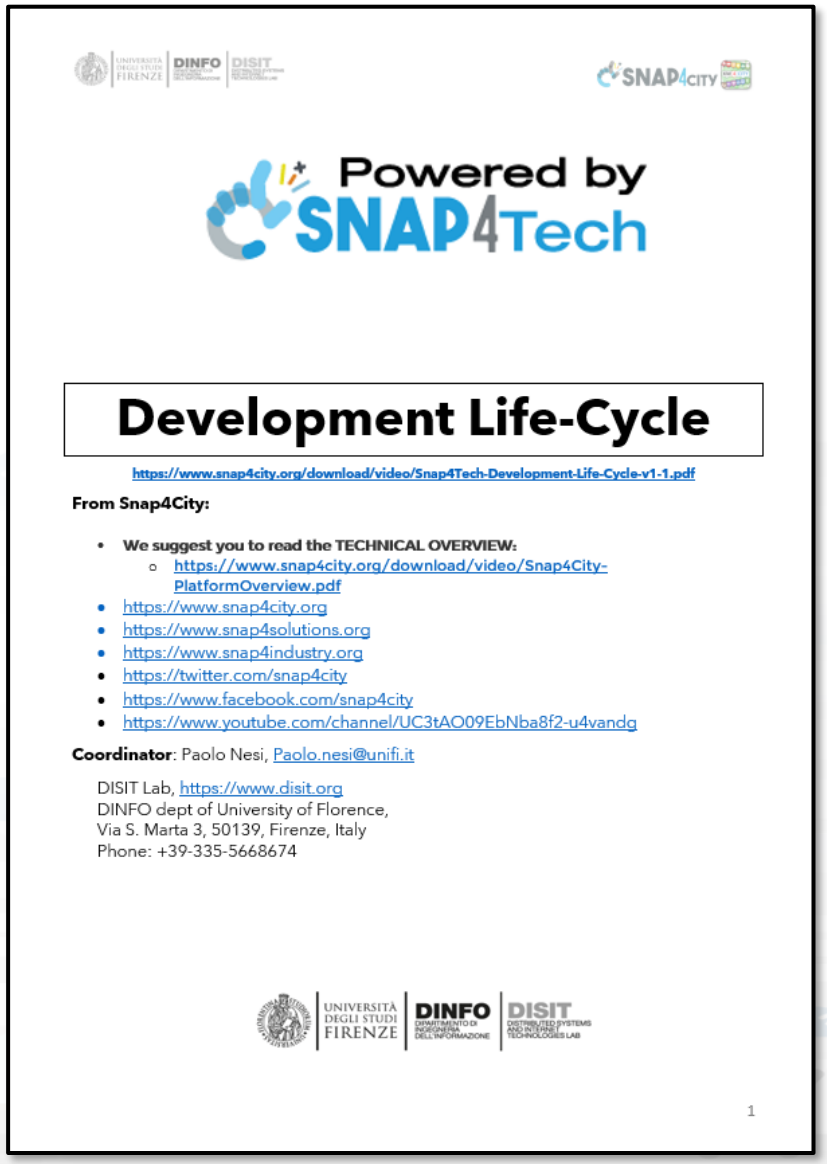
Snap4City Tools vs Development Life Cycle



Development life cycle manual

Development

https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf



Powered by SNAP4Tech

Development Life-Cycle

https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle-v1-1.pdf

From Snap4City:

- **We suggest you to read the TECHNICAL OVERVIEW:**
  - https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- https://www.snap4city.org
- https://www.snap4solutions.org
- https://www.snap4industry.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city
- https://www.youtube.com/channel/UC3tAO09EbNba8f2-u4vandg

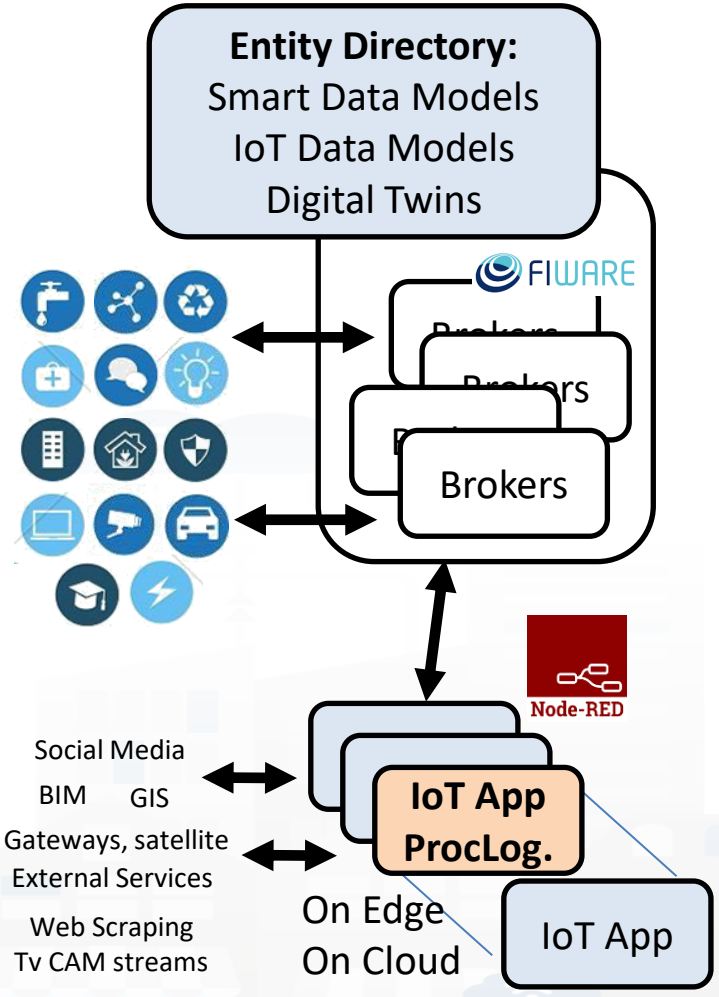
Coordinator: Paolo Nesi, Paolo.nesi@unifi.it

DISIT Lab, https://www.disit.org
DINFO dept of University of Florence,
Via S. Marta 3, 50139, Firenze, Italy
Phone: +39-335-5668674

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

Data Ingestions Strategy



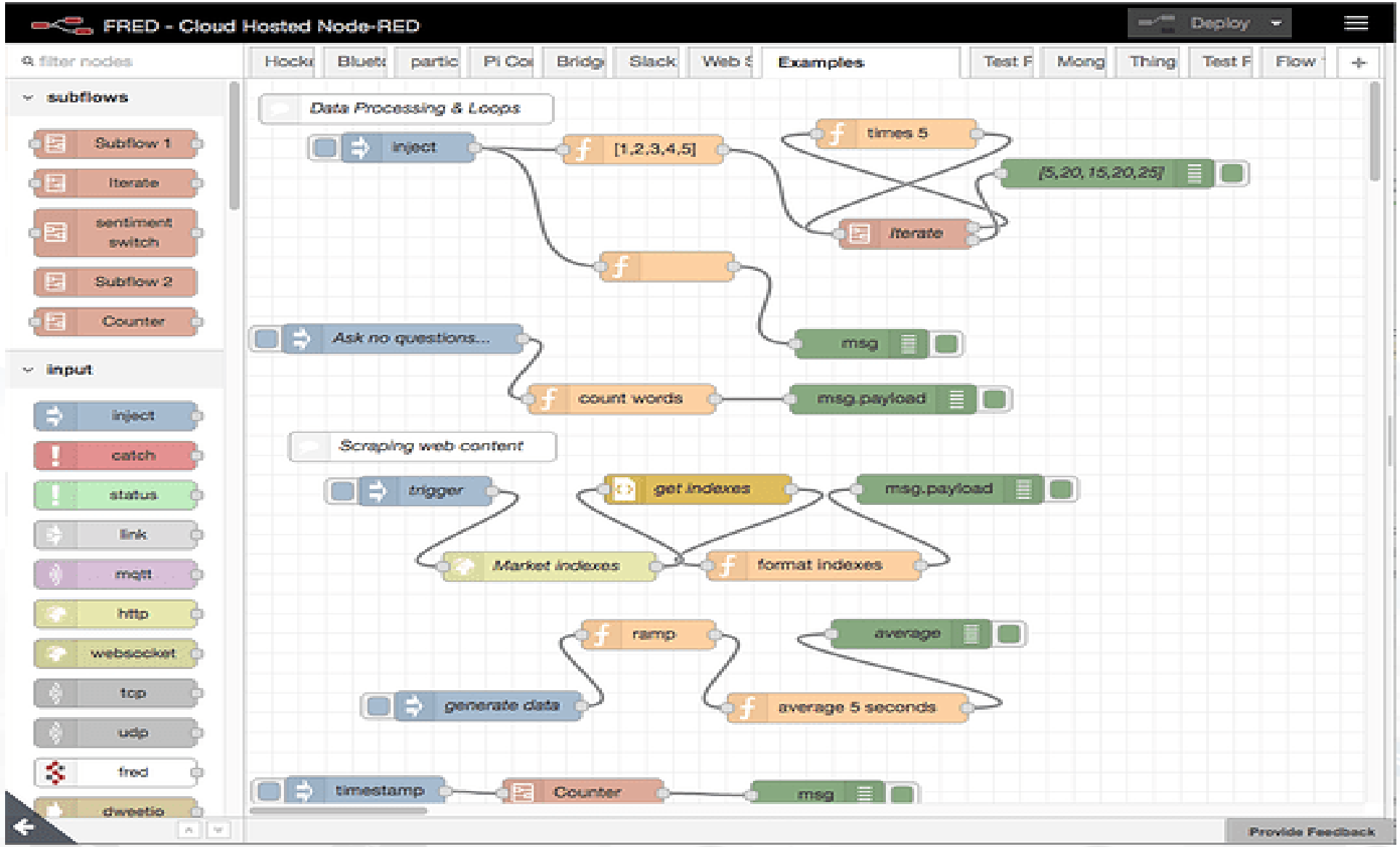
1

1) Via ***internal Brokers***: NGSI V1/V2, MQTT (beta), JSON
- the data messages arriving in **PUSH** on the platform,
  - if compliant to a known Model **and** referring to a known Device/Entity
  - They are AUTOMATICALLY: *stored, indexed,* and ready to be used by Wizard, Dashboards, Views and Applications

- 2) Via ***external Brokers***: NGSI V2, NGSI LD (beta), JSON
  - the data messages arriving in **PUSH** on the external broker,
    - Have to be **mapped** to a known Model **and** referring to a known Device/Entity
    - To be AUTOMATICALLY: *stored, indexed,* and ready to be used by Wizard, Dashboards, Views and Applications

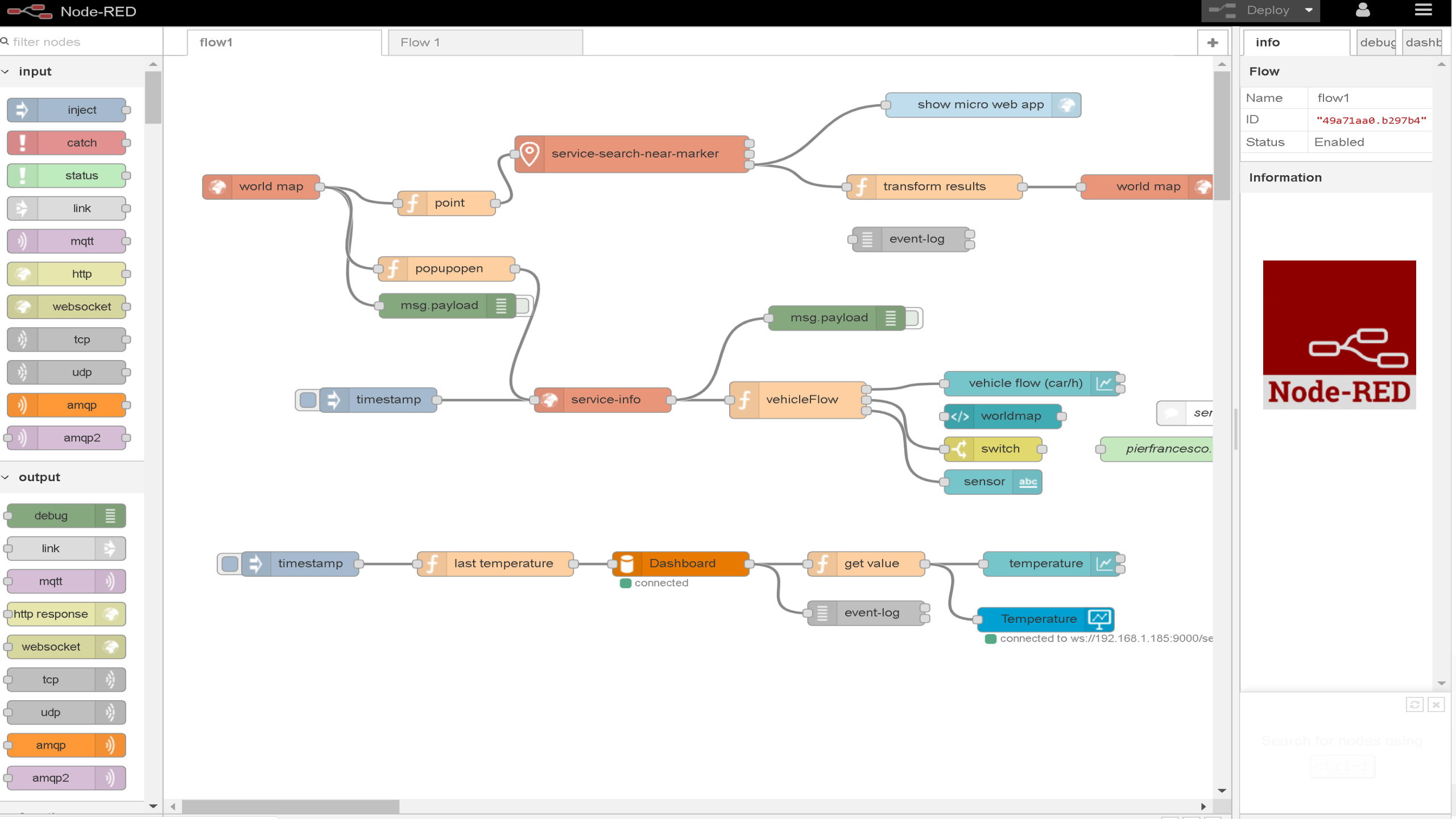
3

3) Via **any other broker, GateWay, API, server, WebService, database, protocol and/or format of the many available on Snap4City, in PUSH/PULL have to be**
- **mapped** on an Snap4City IoT/Entity Model and Entity/Device Instance
- **ingested in real time or batch**, by a process implemented as
  - an **IoT App/Proc.Logic, on container, on cloud or Edge**
  - a **Python or other language**, **on container, on cloud or Edge**


FRED - Cloud Hosted Node-RED
Deploy
filter nodes
Examples
subflows
Subflow 1
Iterate
sentiment switch
Subflow 2
Counter
input
inject
catch
status
link
mqtt
http
websocket
tcp
udp
fred
dweetio
Data Processing & Loops
inject
[1,2,3,4,5]
times 5
[5,20,15,20,25]
iterate
msg
Ask no questions...
count words
msg.payload
Scraping web content
trigger
get indexes
msg.payload
Market indexes
format indexes
ramp
average
generate data
average 5 seconds
timestamp
Counter
msg
Provide Feedback

Node-RED

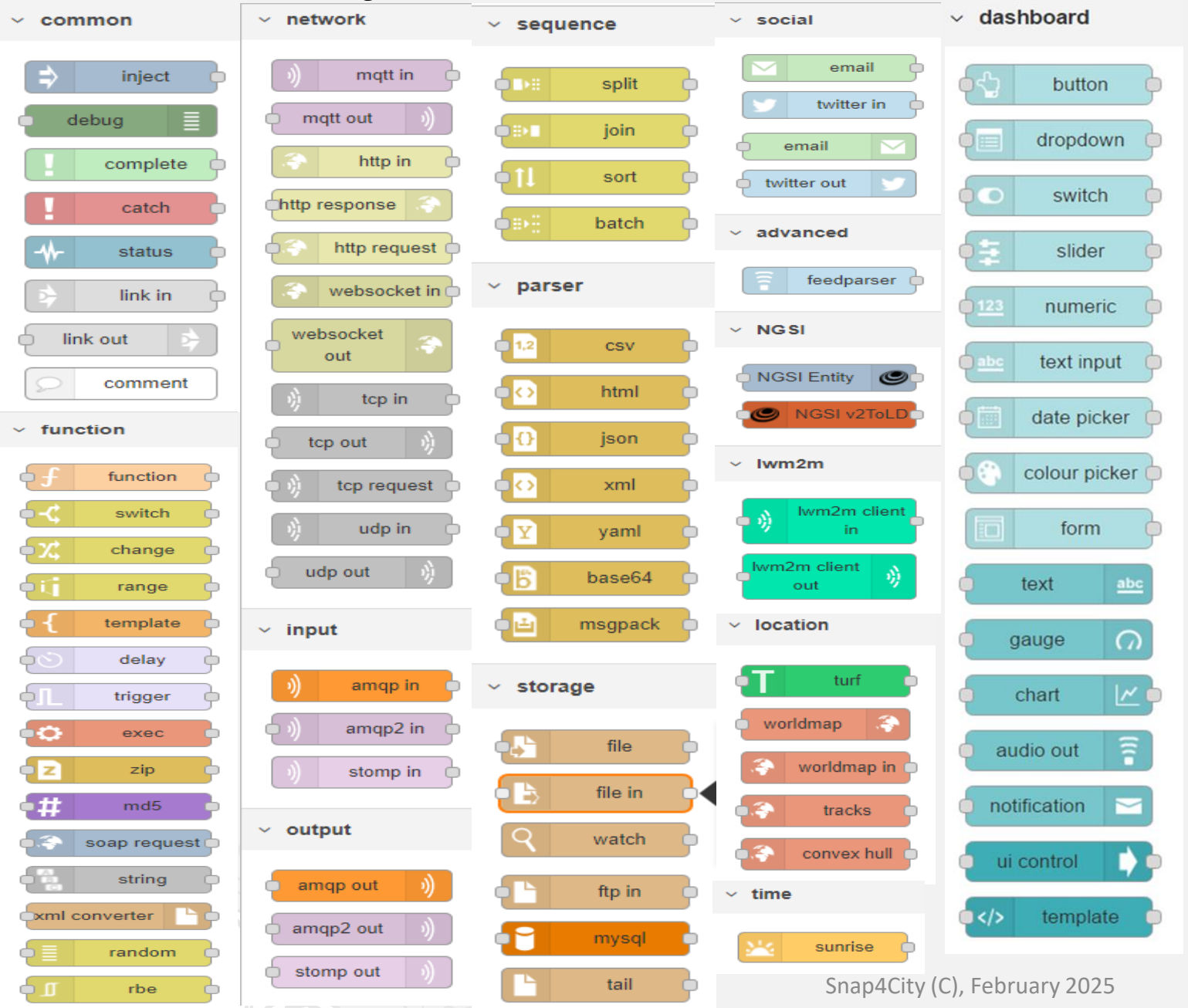



Node-RED
Deploy
filter nodes
flow1
Flow 1
input
inject
catch
status
link
mqtt
http
websocket
tcp
udp
amqp
amqp2
output
debug
link
mqtt
http response
websocket
tcp
udp
amqp
amqp2
show micro web app
service-search-near-marker
world map
point
transform results
world map
event-log
popupopen
msg.payload
msg.payload
timestamp
service-info
vehicleFlow
vehicle flow (car/h)
worldmap
switch
sensor
timestamp
last temperature
Dashboard
connected
get value
temperature
event-log
Temperature
connected to ws://192.168.1.185:9000/se
info
debug
dashb
Flow
Name
flow1
ID
"49a71aa0.b297b4"
Status
Enabled
Information
Node-RED
Search for nodes using

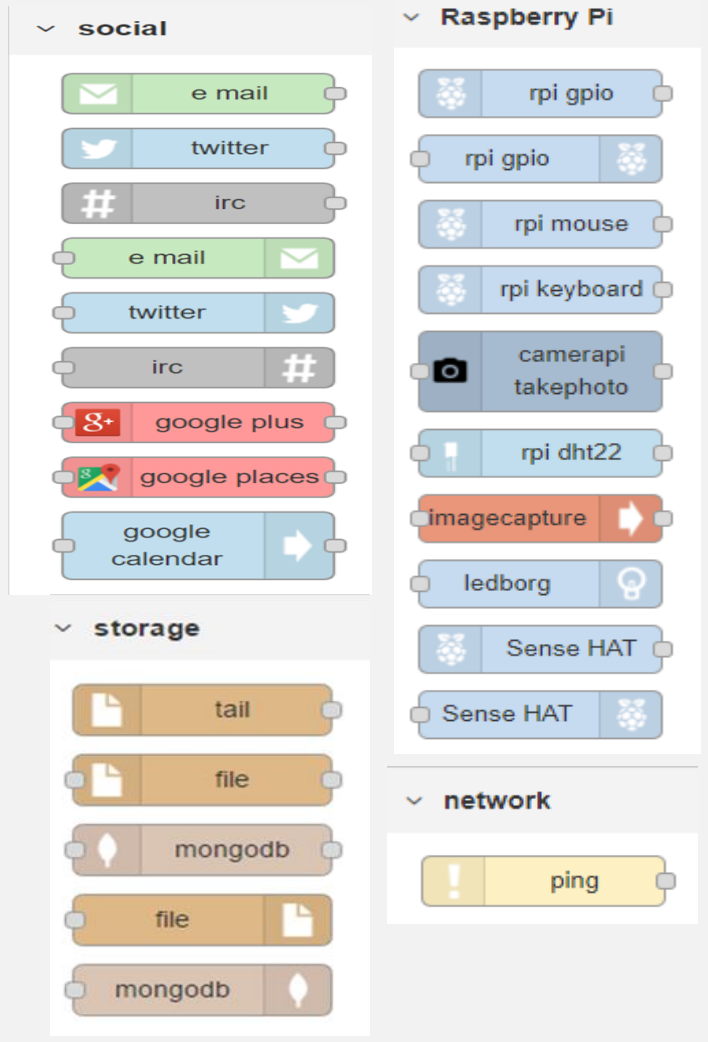
Basic Node.js Blocks on NodeRed on our Advanced IOT Apps

DISIT Lab, Distributed Data Intelligence and
Systems and Internet
Information Engin
http://www.di





+ on IOT Edge Raspberry

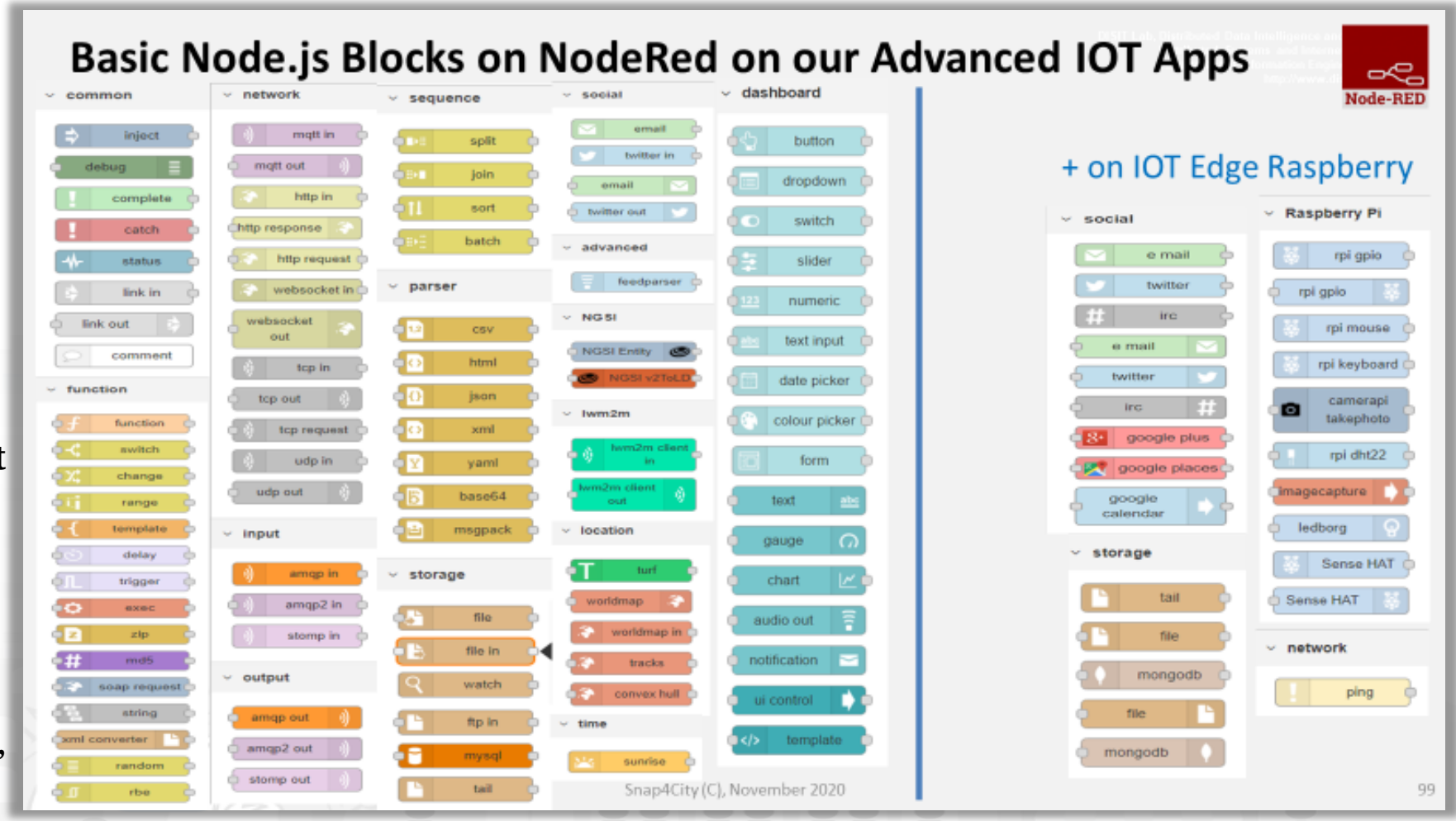


Node-RED Basic Blocks

It is provided with **a minimum set** of functionalities (the building blocks/nodes) while other blocks can be easily added loading them from a **large library** made available by the **JS Foundation**.

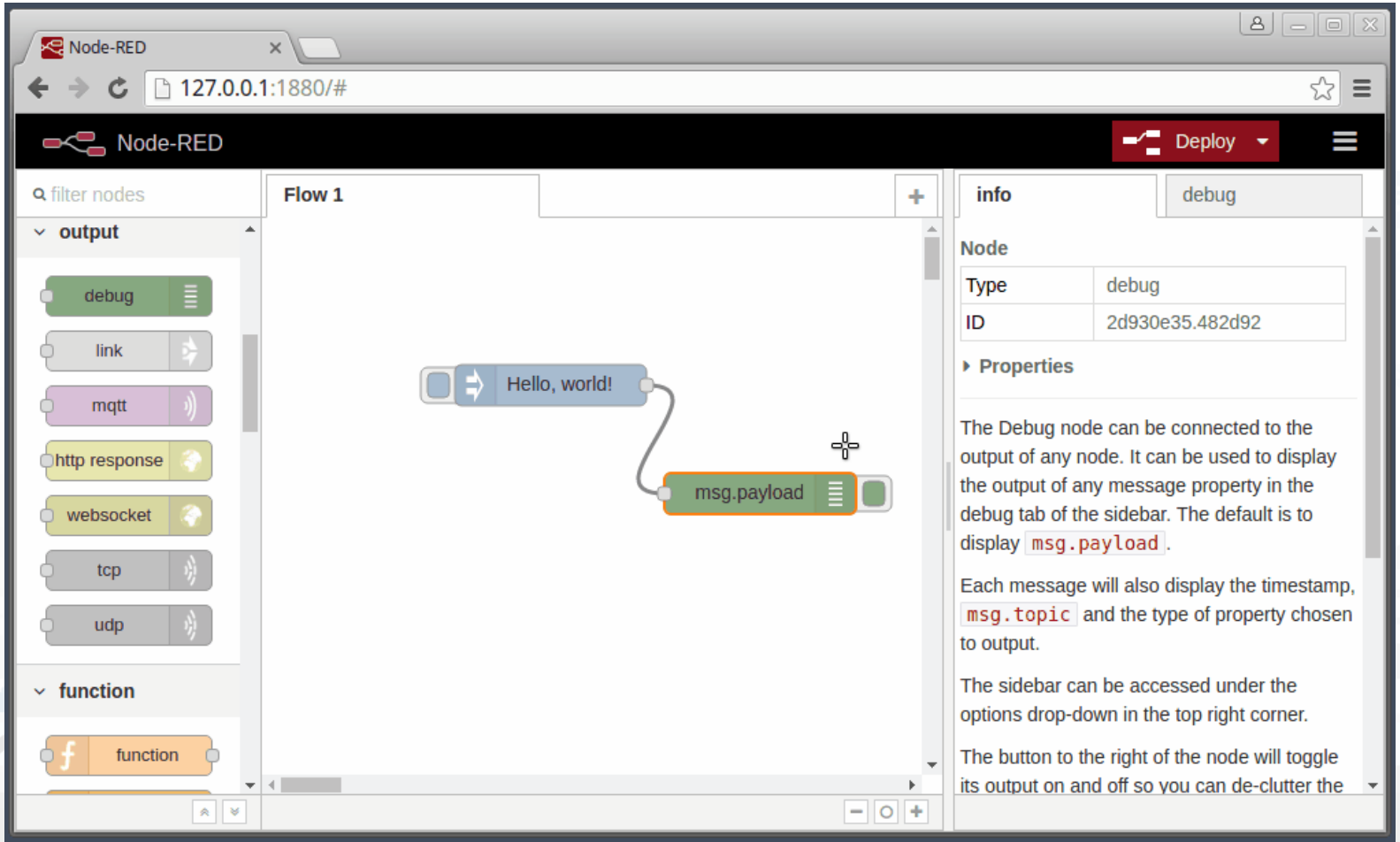
Despite to its diffusion, for the usage in the context of Smart City it was **not powerful** to cope with the **basic requirements** of the domain.

The classical nodes provided in the standard version can be classified as: input, output, function, social, storage, analysis, advanced, and dashboard.



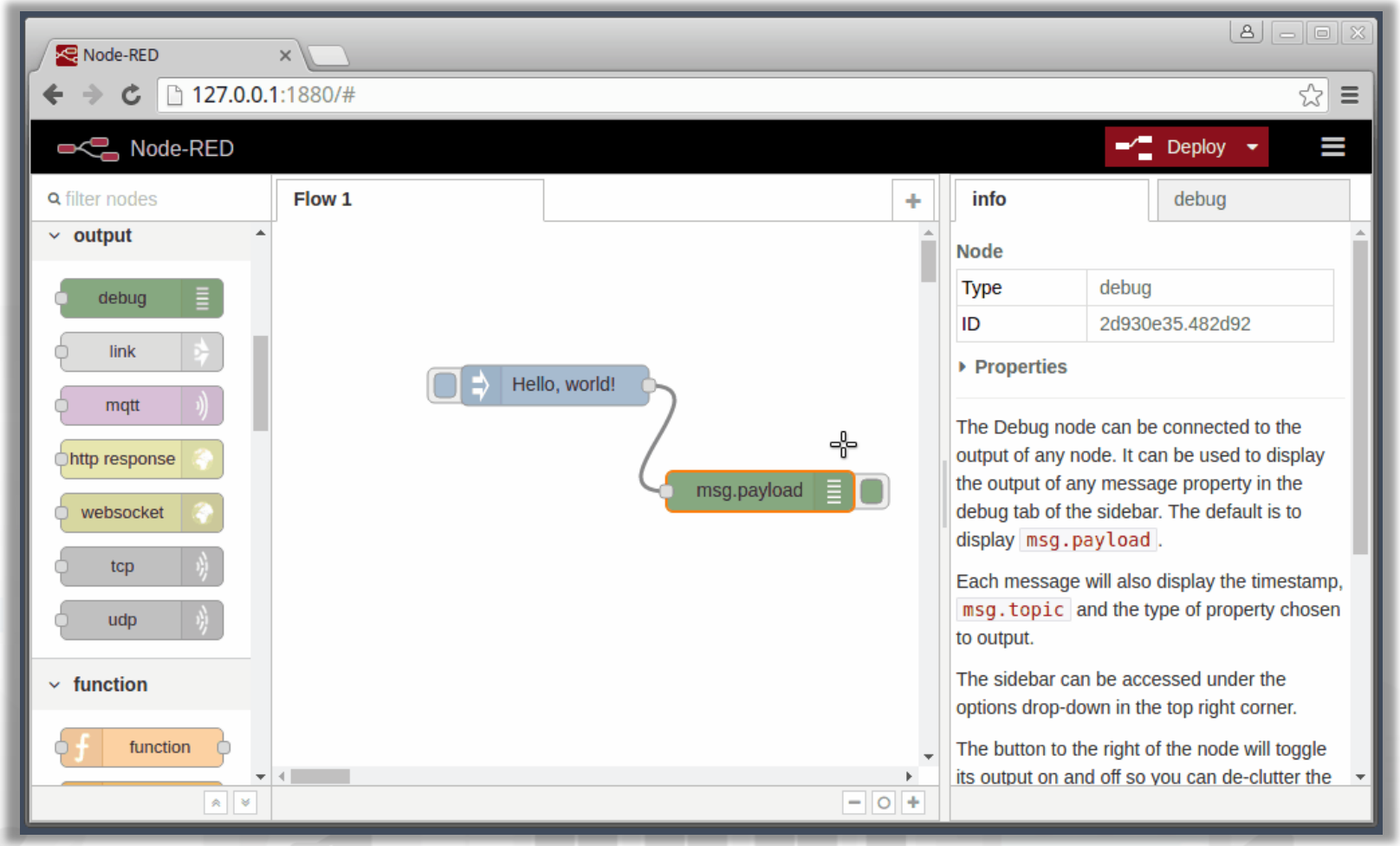
Hello World of Node-RED

- http://developer.opto22.com/nodered/general/getting-started/node-red-hello-world/



Node-RED

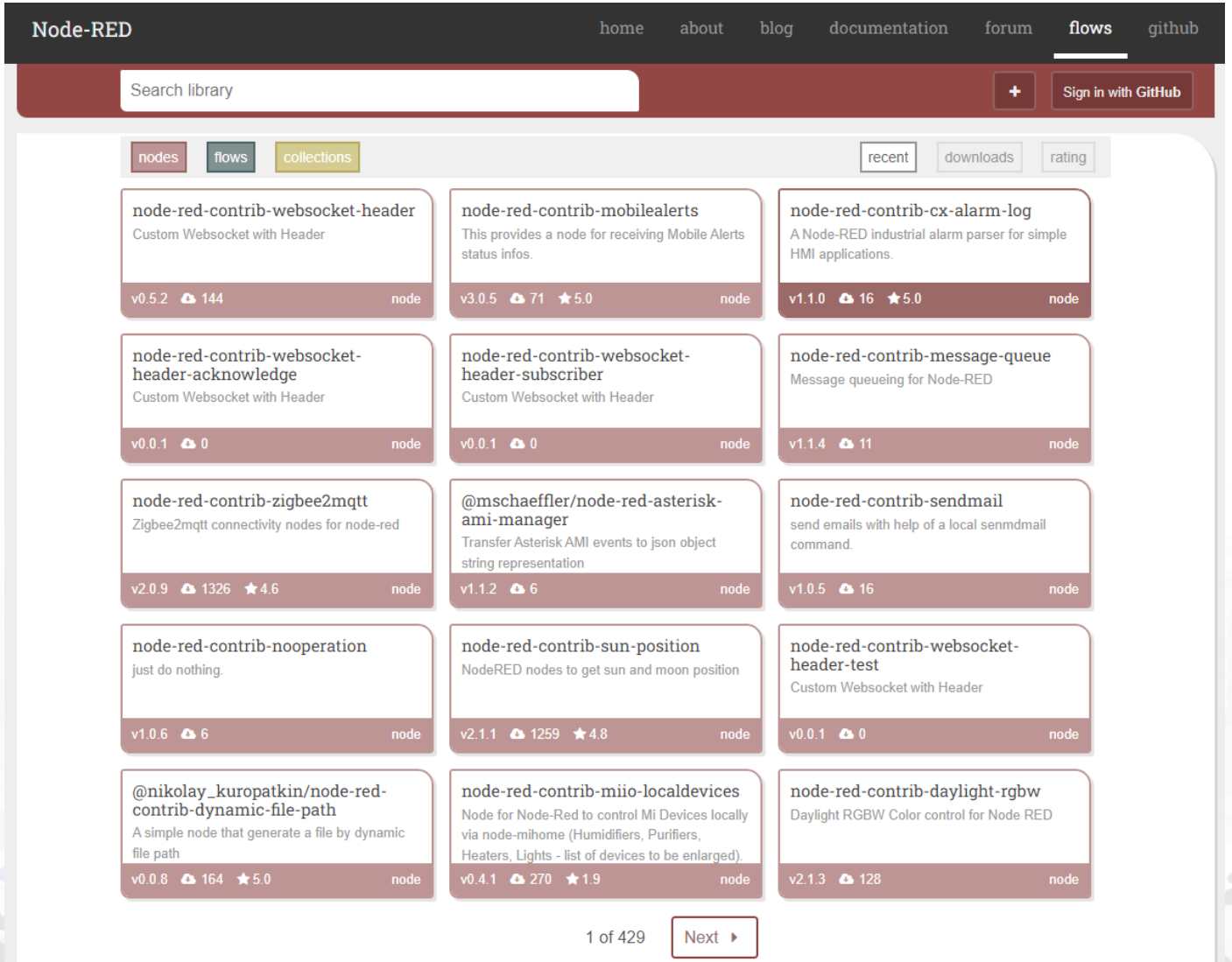
- Node-RED is a **flow-based** development tool for visual programming proposed by **JS Foundation**
- The Node-RED approach is a mix of **visual composition** of **nodes/blocks** to compose the so-called **flows** that are concurrently executed by an engine **Node.js**.
- It is quite diffuse being also directly provided into **official releases** of IOT devices as **Raspberry Pi** family
- Based on **Node.js**
- 100% **open source**



Node-RED Libraries

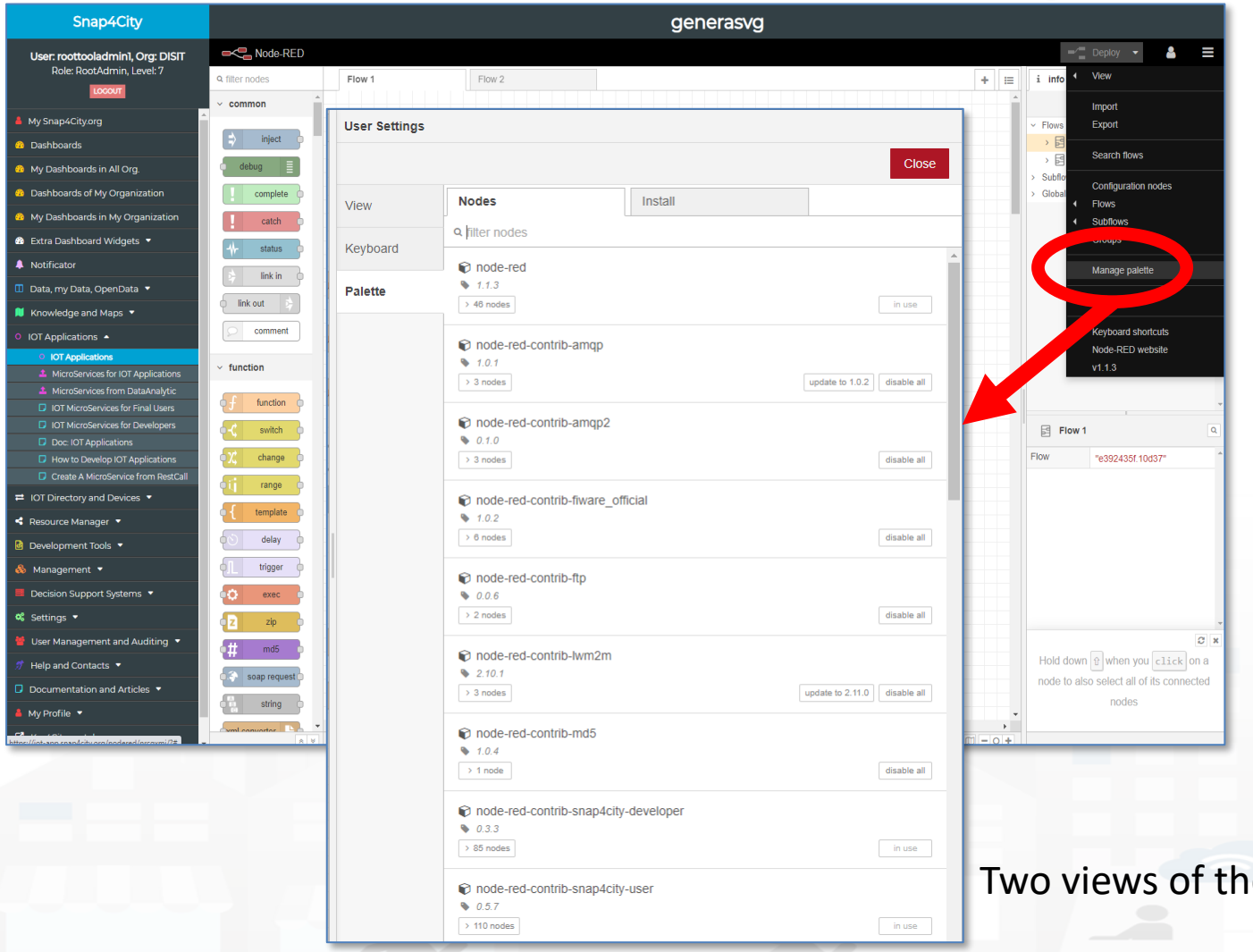


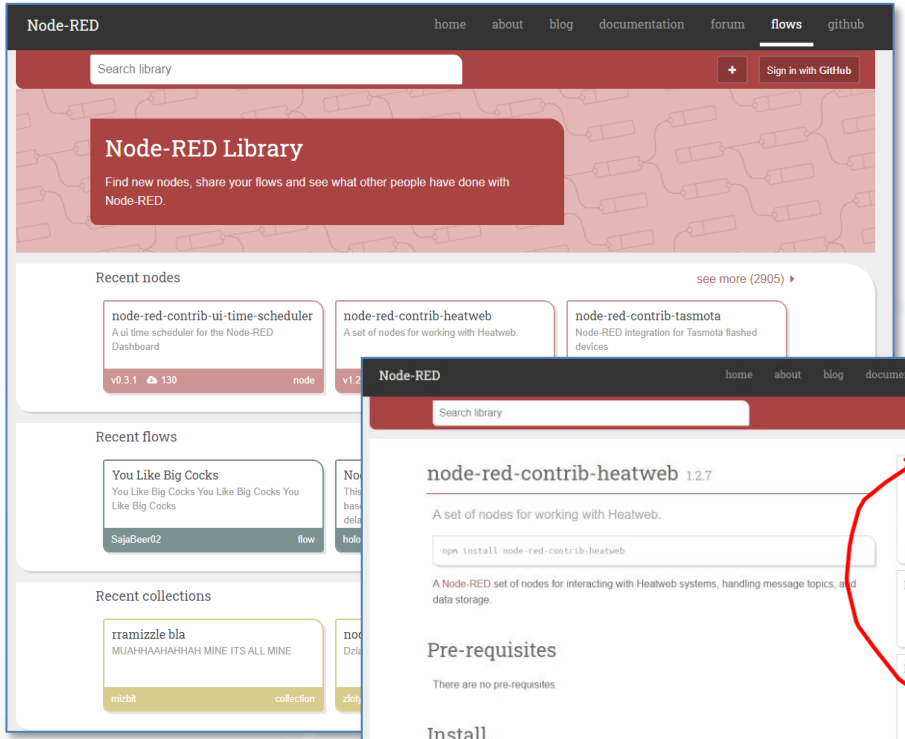
https://flows.nodered.org/search?term=

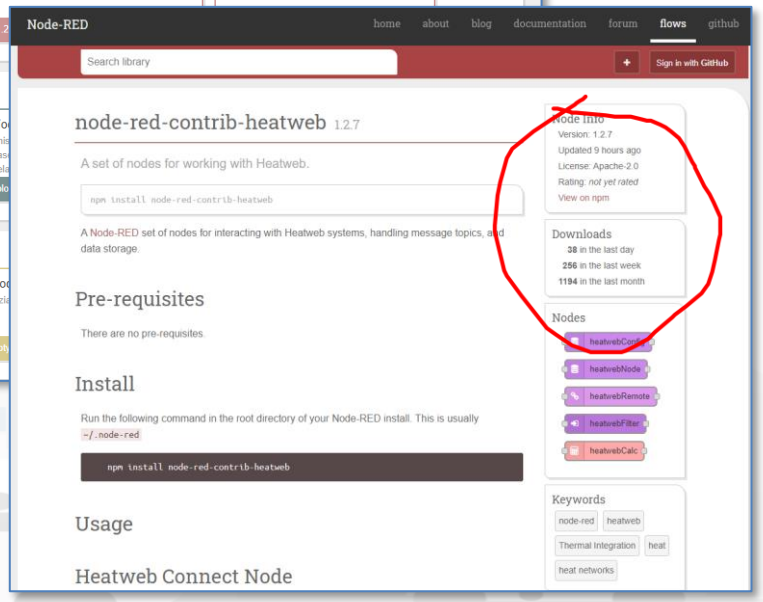


Load Library from Palette

https://flows.nodered.org/







Two views of the same libraries

Node-RED

SnaP4City Libraries



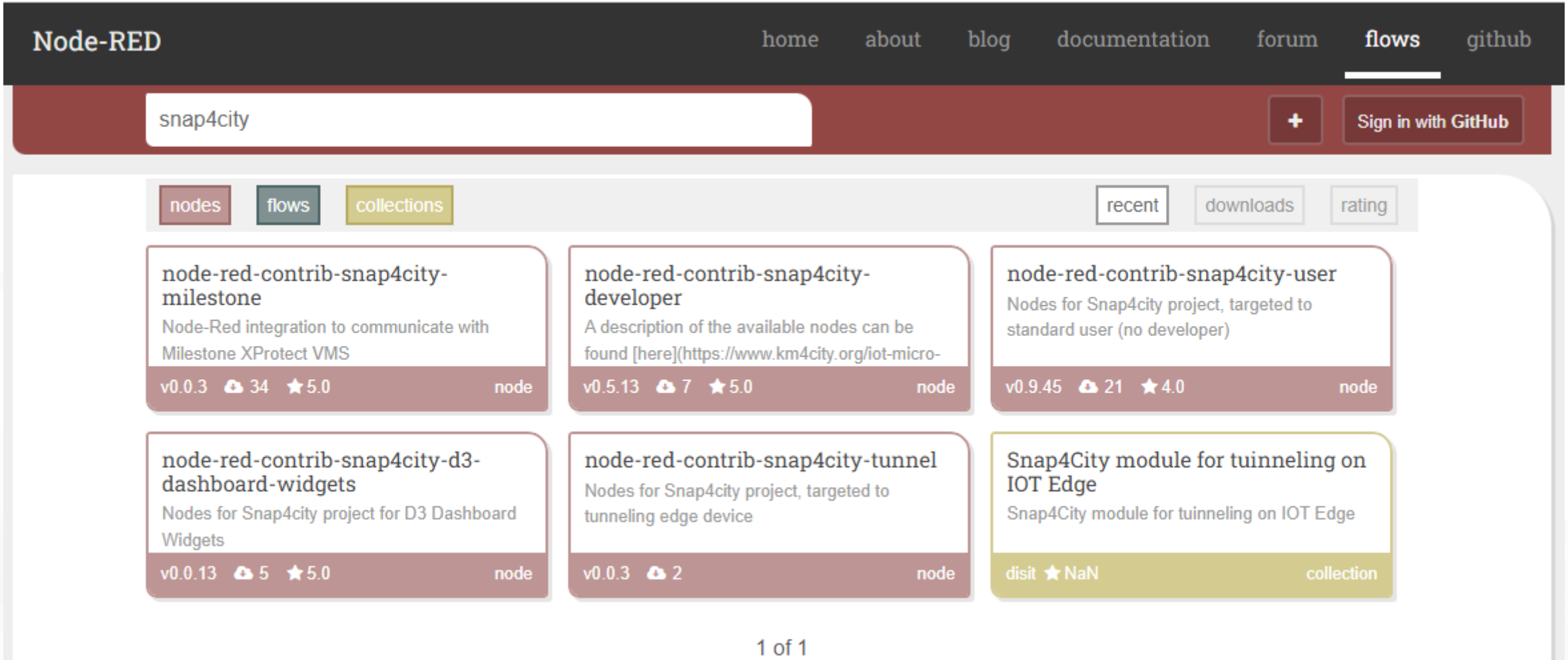
Proc.Logic / IoT App Editor: NODE-RED

- In the Proc.Logic / **IoT App of Snap4City**, it is possible to:
  - Execute flows that process data as: Event Driven, Batch (periodic or not)
  - Create multiple concurrent Flows for each IoT App / Proc. Logic
  - Create subflows as macros to be reused
  - Create Groups of nodes as macro
  - **Save/load**, **share**, of nodes, flows and applications with other users via
    - the Snap4City Resource Manager or
    - with JS Foundation or
    - via email, skype, file sharing in general
  - ..

IoT App / Proc.Logic Editor: NODE-RED

- In the **IoT Apps / Proc.Logic of Snap4City**, it is possible to Extend the Capabilities:
  - Load other Nodes, segments of flow and entire flows from several sources: email, libraries, S4C repository, etc.
  - Load other libraries of MicroServices/Nodes/Blocks from **Manage Palette**
    - A large set of Libraries of Node is available.
    - The loading may have some limitations for security reasons
  - **Get more IOT App / Proc.Logic** above the Limit that may depend on the organization and/or on personal authorizations, ask to Admin
  - ..

Snap4City Libraries on Node-RED

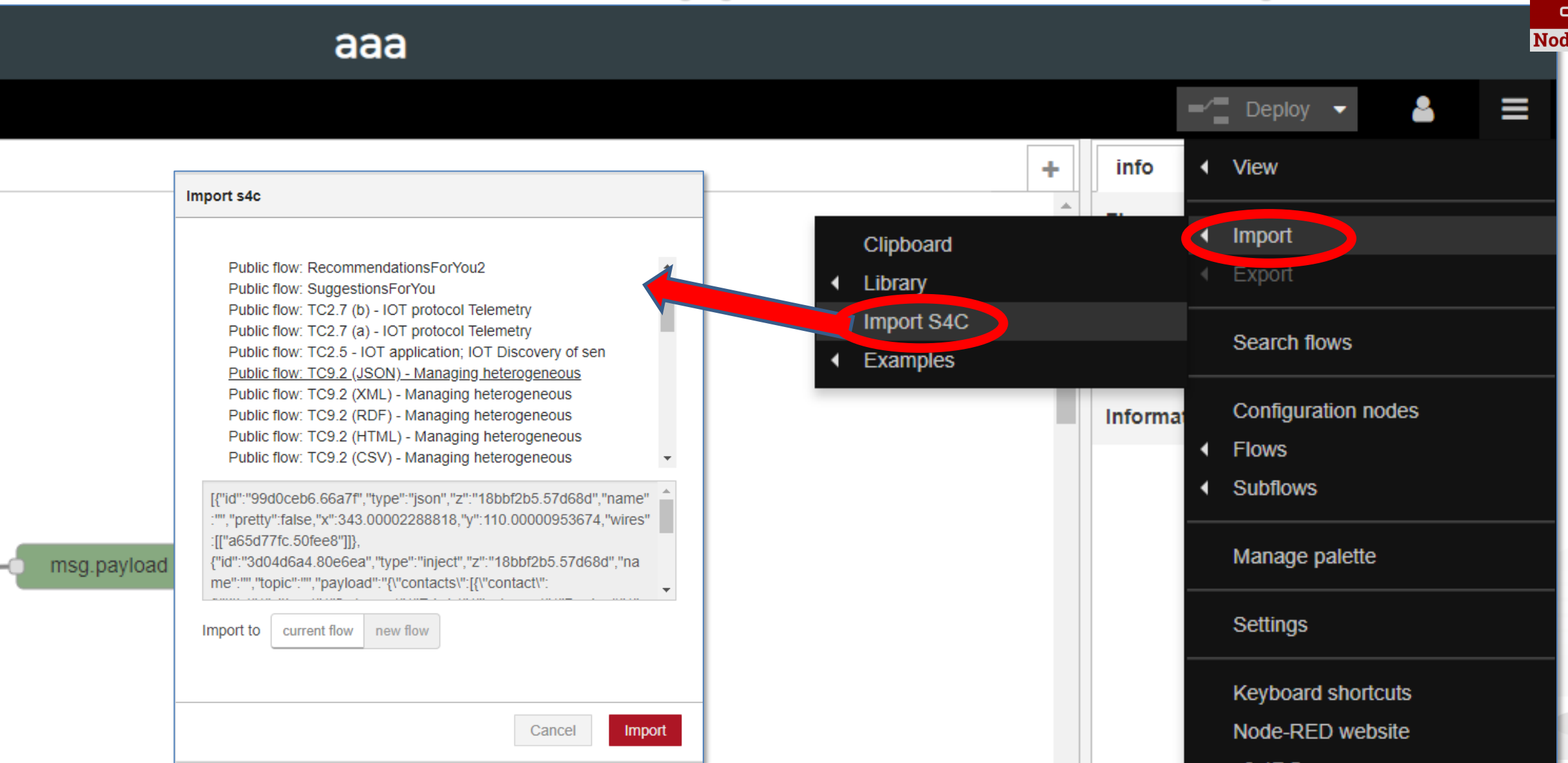


https://flows.nodered.org/search?term=snap4city

Snap4City Libraries on Node-RED

- https://flows.nodered.org/search?term=snap4city
  - https://flows.nodered.org/node/node-red-contrib-snap4city-user
  - https://flows.nodered.org/node/node-red-contrib-snap4city-developer
  - https://flows.nodered.org/node/node-red-contrib-snap4city-d3-dashboard-widgets
  - https://flows.nodered.org/node/node-red-contrib-snap4city-tunnel
  - https://flows.nodered.org/node/node-red-contrib-snap4city-milestone
  - https://flows.nodered.org/node/node-red-contrib-snap4city-clearml

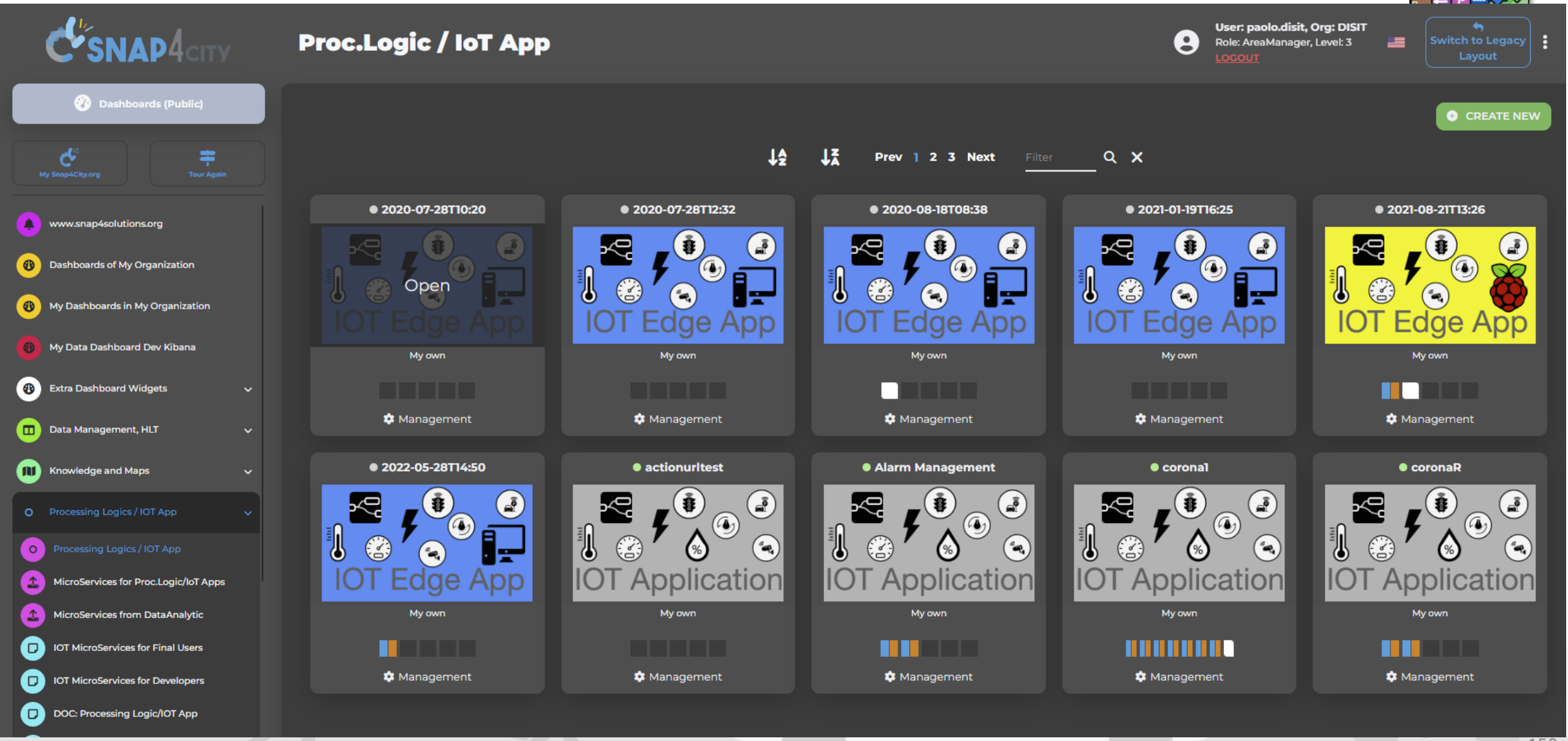
Load an IOT application of example



Node-RED in SnaP4City

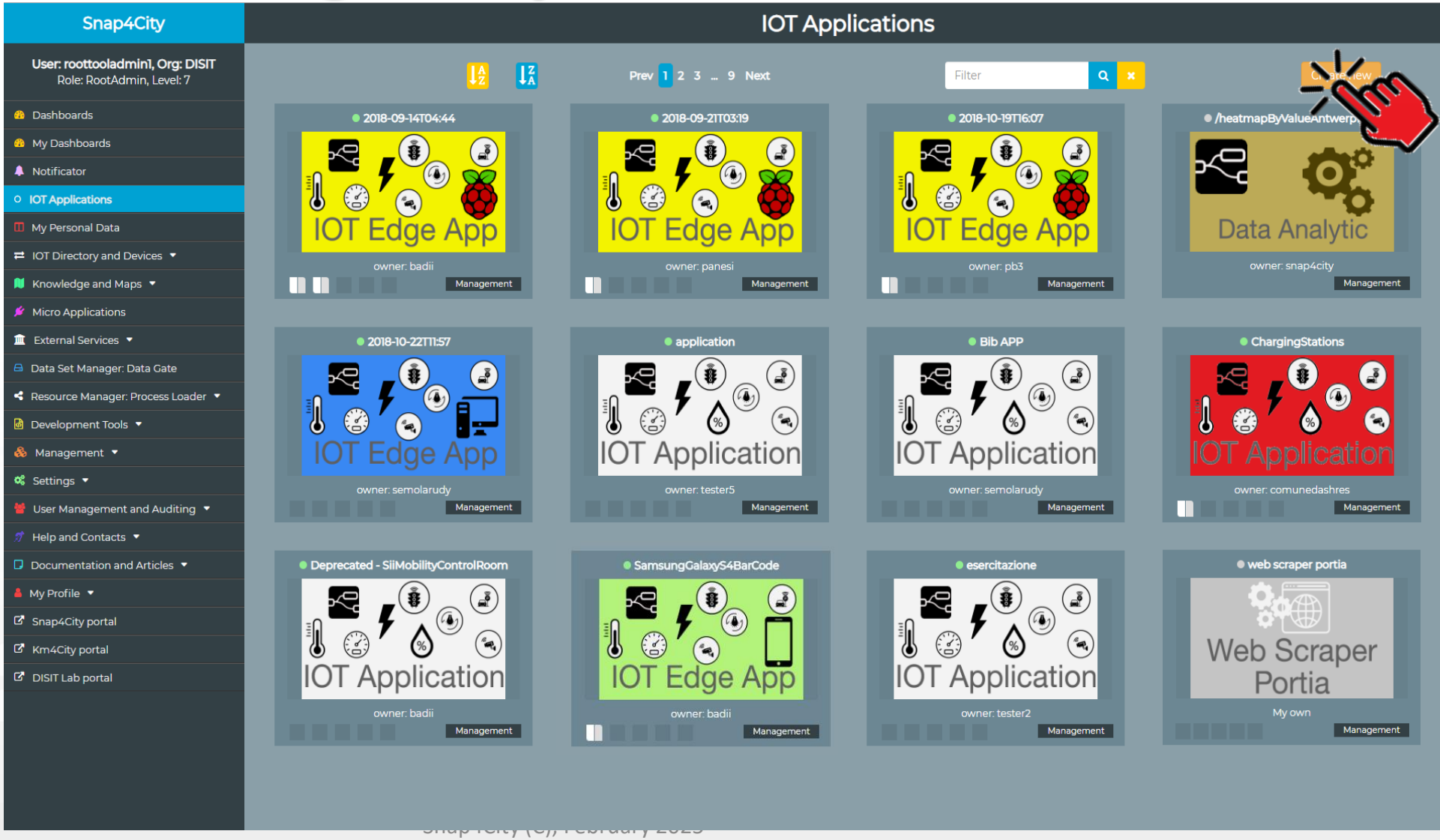
Node-RED




SNAP4CITY
Proc.Logic / IoT App
User: paolo.disit, Org: DISIT
Role: AreaManager, Level: 3
LOGOUT
Switch to Legacy Layout
Dashboards (Public)
My Snap4City.org
Tour Again
www.snap4solutions.org
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
Extra Dashboard Widgets
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Processing Logics / IOT App
MicroServices for Proc.Logic/IoT Apps
MicroServices from DataAnalytic
IOT MicroServices for Final Users
IOT MicroServices for Developers
DOC: Processing Logic/IOT App
CREATE NEW
Prev 1 2 3 Next
Filter
2020-07-28T10:20
Open
IOT Edge App
My own
Management
2020-07-28T12:32
IOT Edge App
My own
Management
2020-08-18T08:38
IOT Edge App
My own
Management
2021-01-19T16:25
IOT Edge App
My own
Management
2021-08-21T13:26
IOT Edge App
My own
Management
2022-05-28T14:50
IOT Edge App
My own
Management
actionurltest
IOT Application
My own
Management
Alarm Management
IOT Application
My own
Management
coronal
IOT Application
My own
Management
coronaR
IOT Application
My own
Management

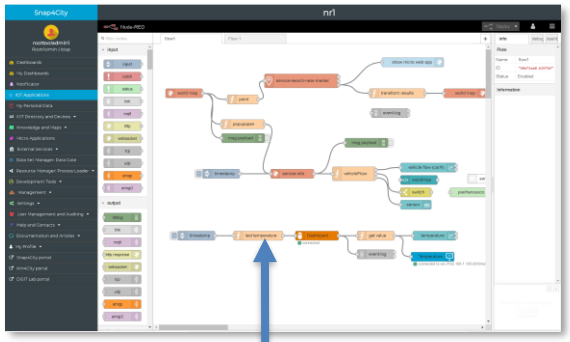
IOT Application Listing, they can be

- Basic (white)
- Advanced (red)
- IOT Edge
  - Raspberry Pi
  - Android
  - Win/Linux
- Data Analytic (Plumber)
- Web Scraper (Portia)

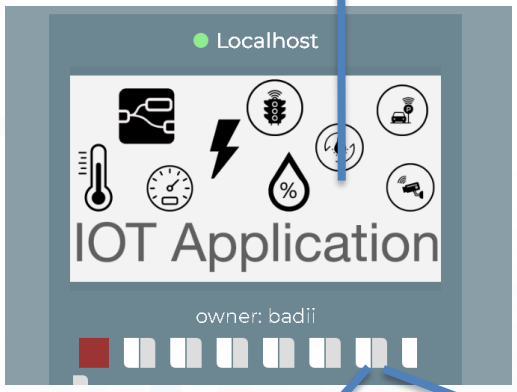


IOT Applications Listing

- Basic / Advanced
- On IOT Edge Raspberry Pi
- On IOT Edge Android
- On IOT Edge Win/Linux



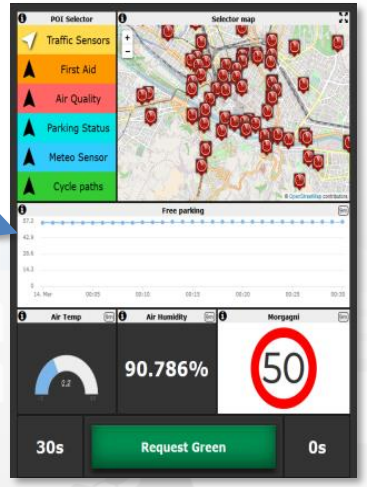
EDIT IOT APP



VIEW

EDIT



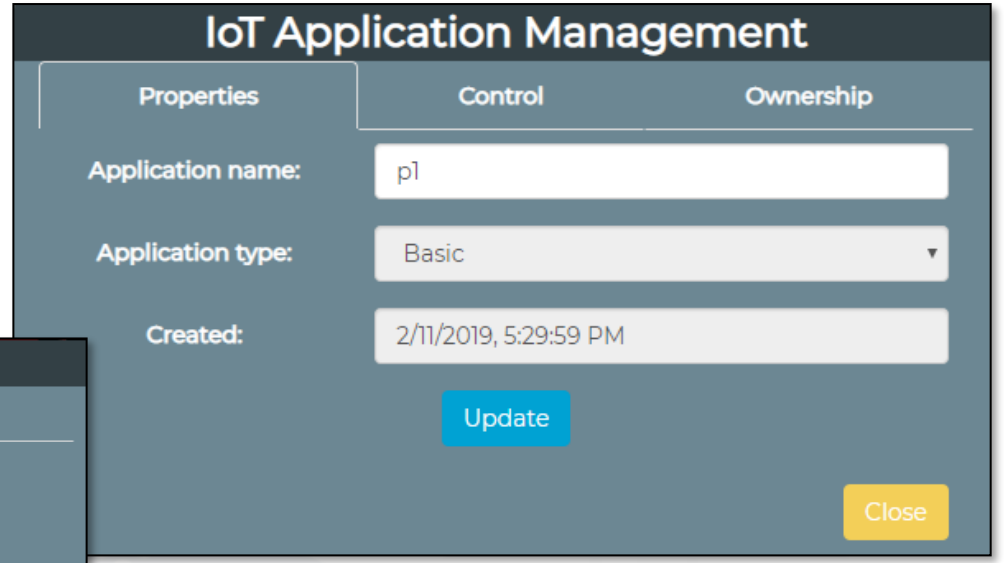


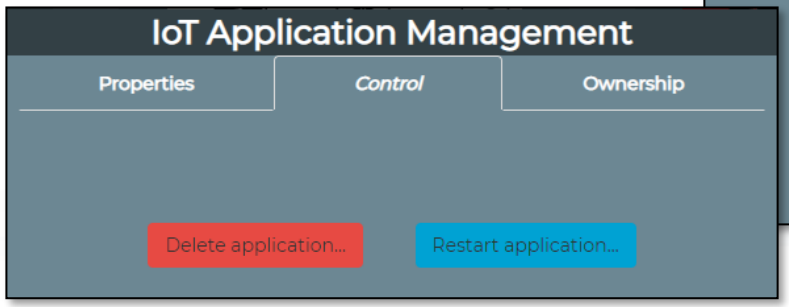


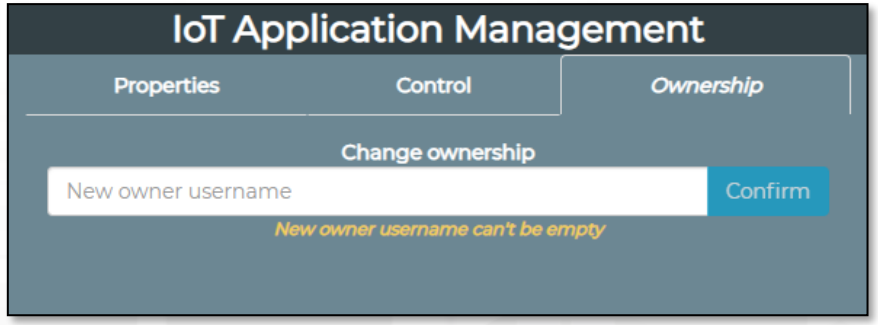


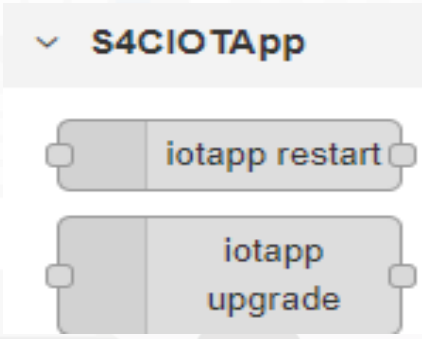
IOT Application Self Control

- **Properties**
  - Name, Type, Creation date
- **Control**
  - Restart Container
  - Delete IOT App
- **Change of ownership**
  - Pass to another Snap4City User
- **From inside the IOT App**
  - Restart
  - Update Snap4City Library





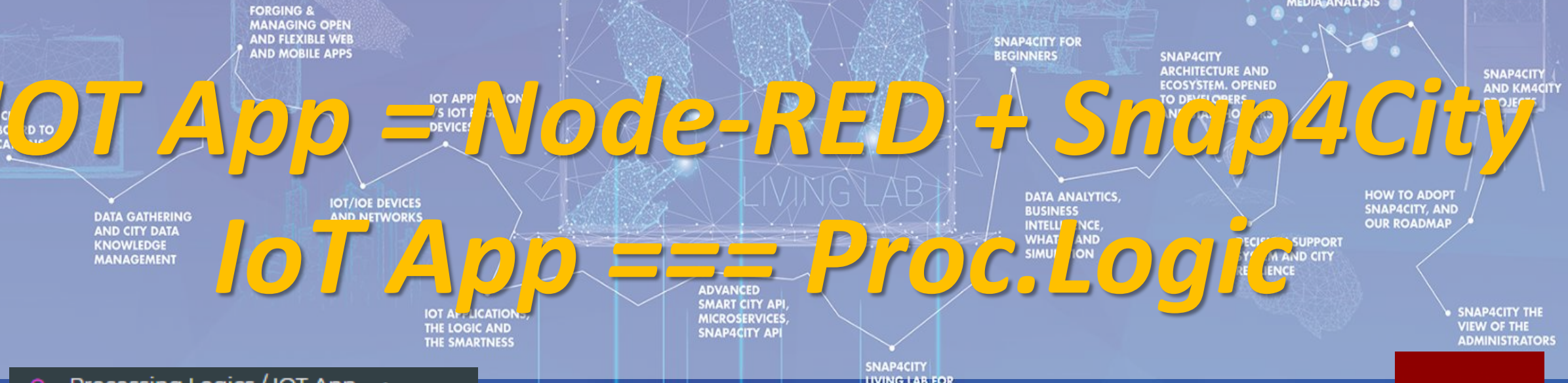


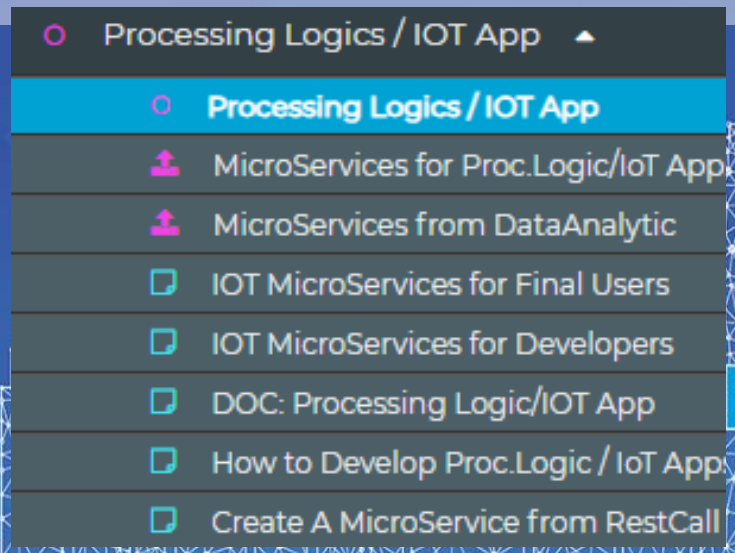


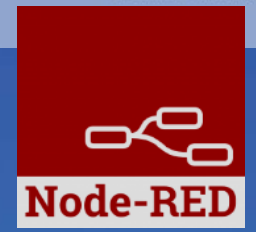
Automating restart and update

IOT App = Node-RED + Snap4City

IoT App === Proc.Logic



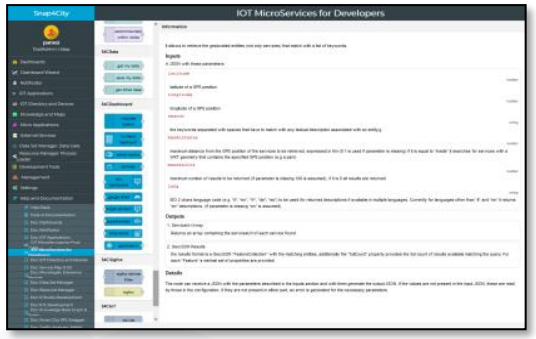




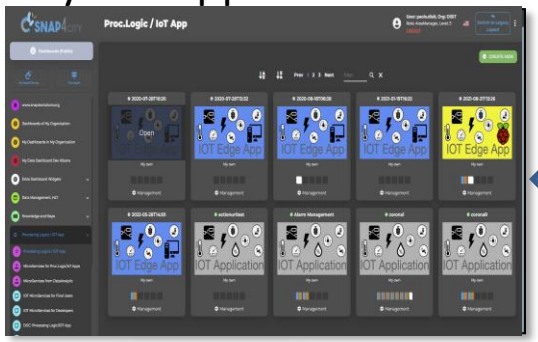
IOT Applications Development

IOT Discovering

MicroServices collections



My IOT Applications

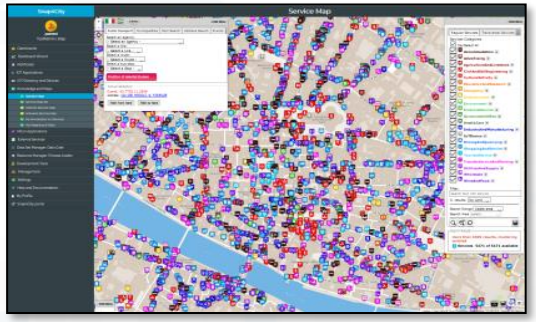


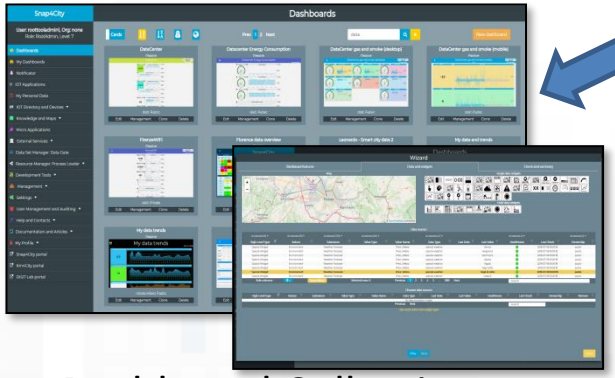
IOT App. Editor



Generating IOT App
With Dashboard







Sharing/saving
reusing IOT App



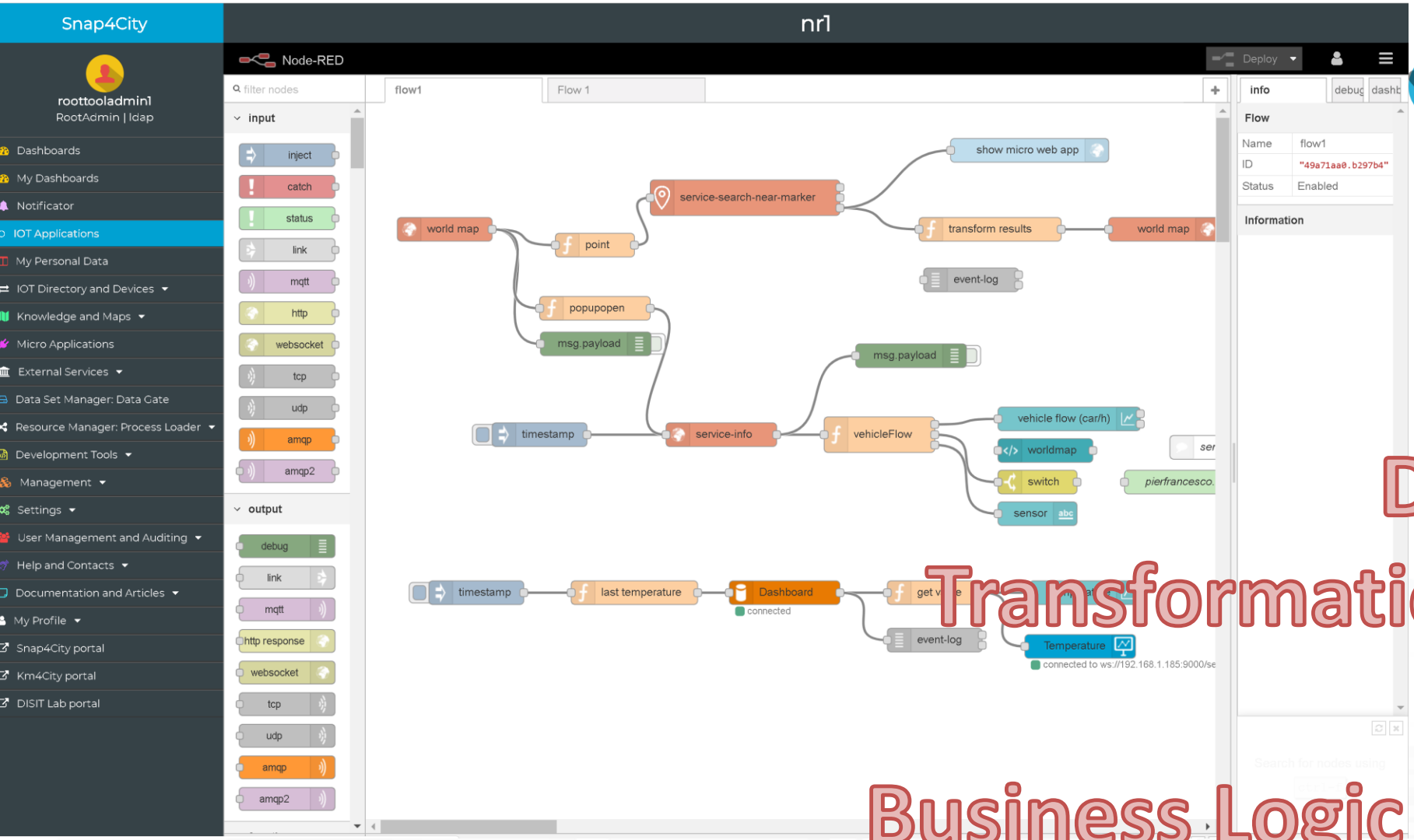
Resource Manager



ServiceMap Discovery



Dashboard Collection,
Editor and Wizard



Editing IOT Applications

Data Adapation

Transformation, Conversion

Integration

Business Logic vs Dashboards

Data Analytics control

Everywhere: Cloud, on IoT Edge Devices

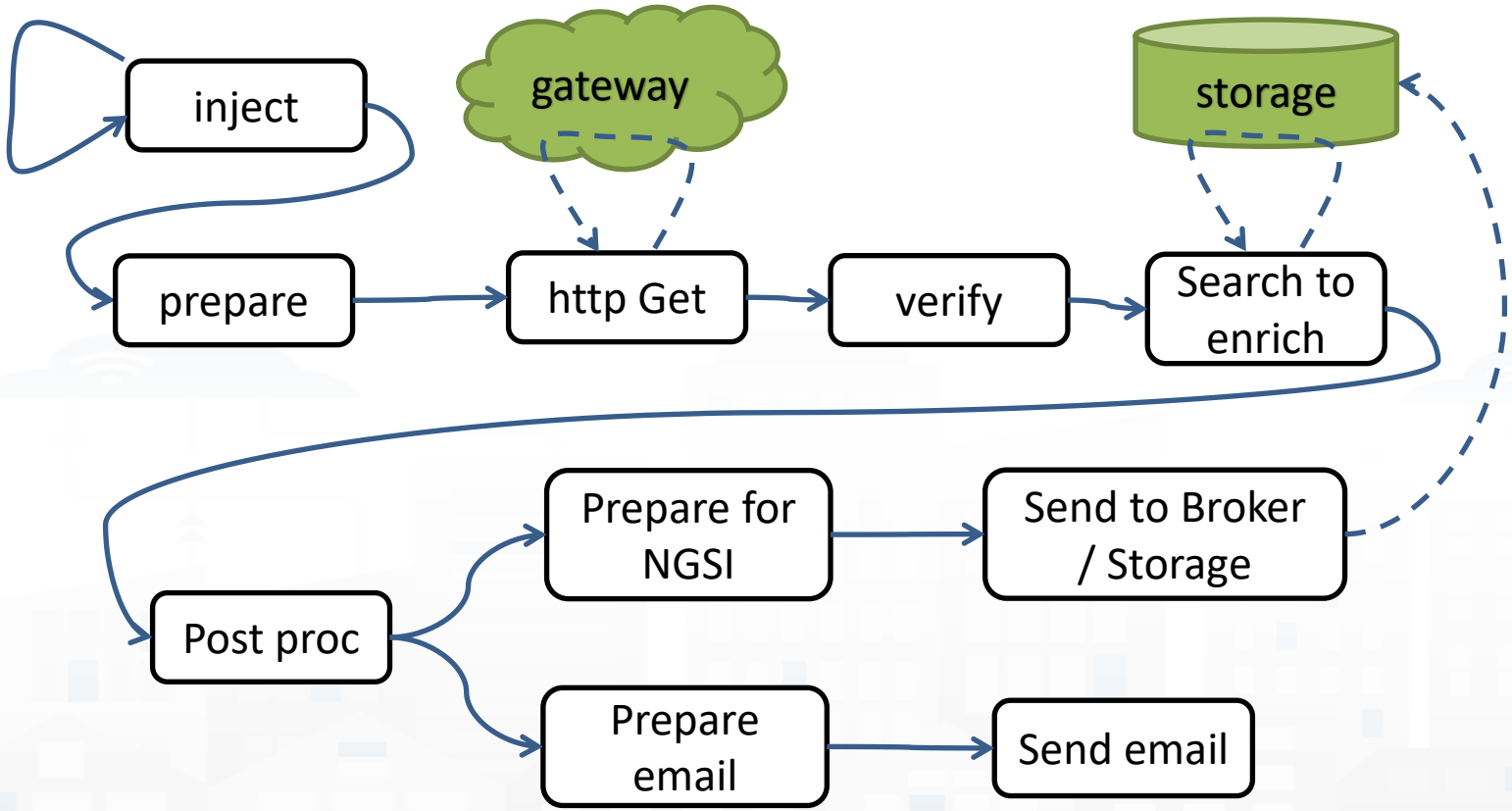
Develop Snap4City IoTApp Processing Logic



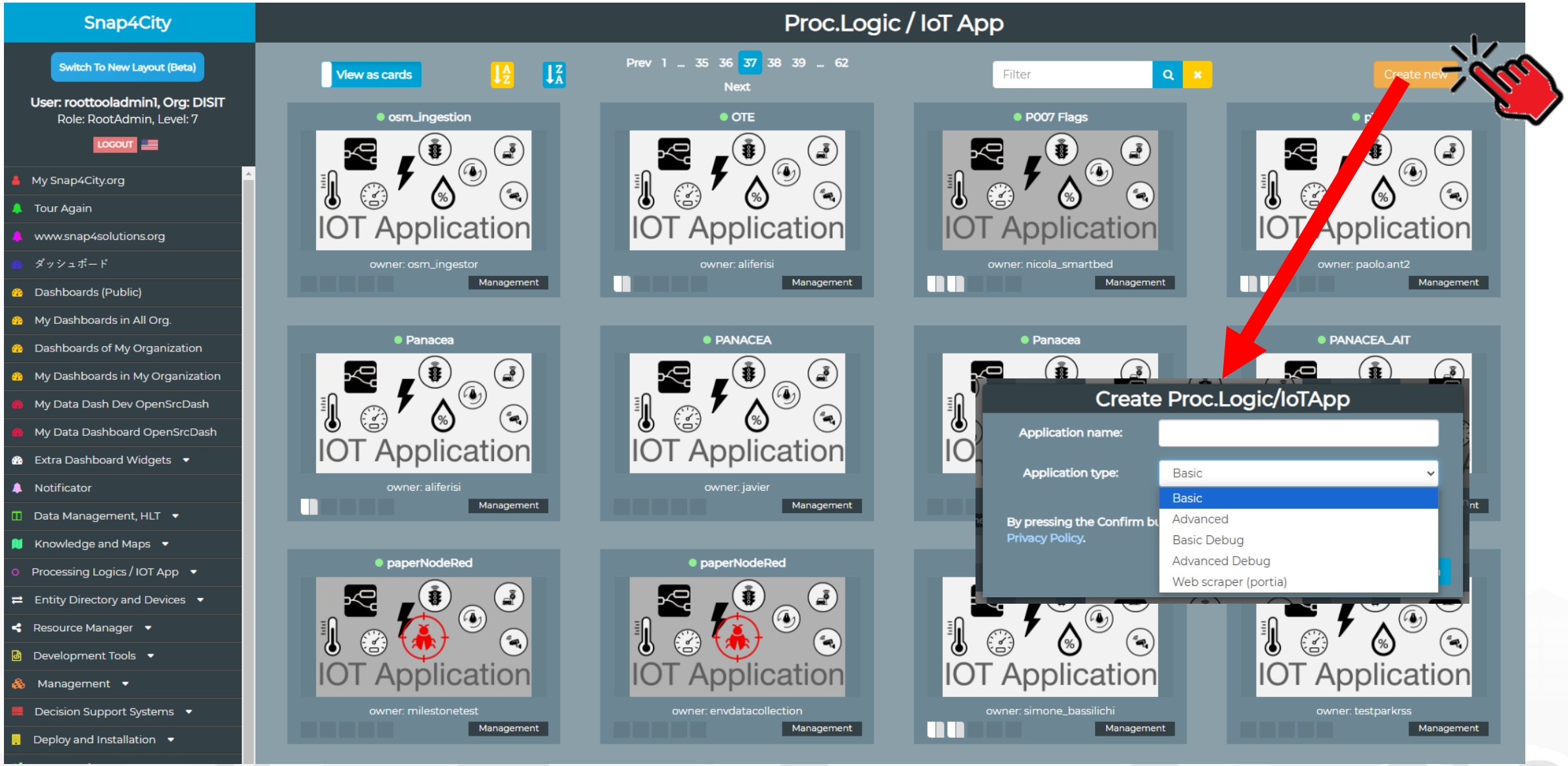
How to Design

1. **Business Logic** is going to be implemented in Proc.Logic (IoT App), with a set of flows.
2. **Decompose your problem** and sequence diagram in single Data/event Flows, from client side and server side.
3. **Identify the single Data/Event Flow**, as those that start from a certain event (periodic or provoked from other messages), and that finish with: sending of data in the storage, change status, send an event, provide a message into a dashboard, send an email, etc.
4. **Design the single Data/Event Flows** with a mixt of possible **activities**.
    1. The design can be performed using data flow diagrams.
    2. It can have sequences, switch, serialization, packing, joining, distribution, communication, transformation, search, etc.
5. When the design of Data/Event Flow mechanism is clear the designers can pass to directly sketch the flow in Node-RED which is a visual programming.
6. **Incrementally improve the Proc.Logic** (IoT App) Node-RED flows by adding nodes needed
7. **Once obtained the Proc.Logic** (IoT App) Node-RED flows in the correct data model you can send data to the ingestion broker, but also perform many other actions on several services.

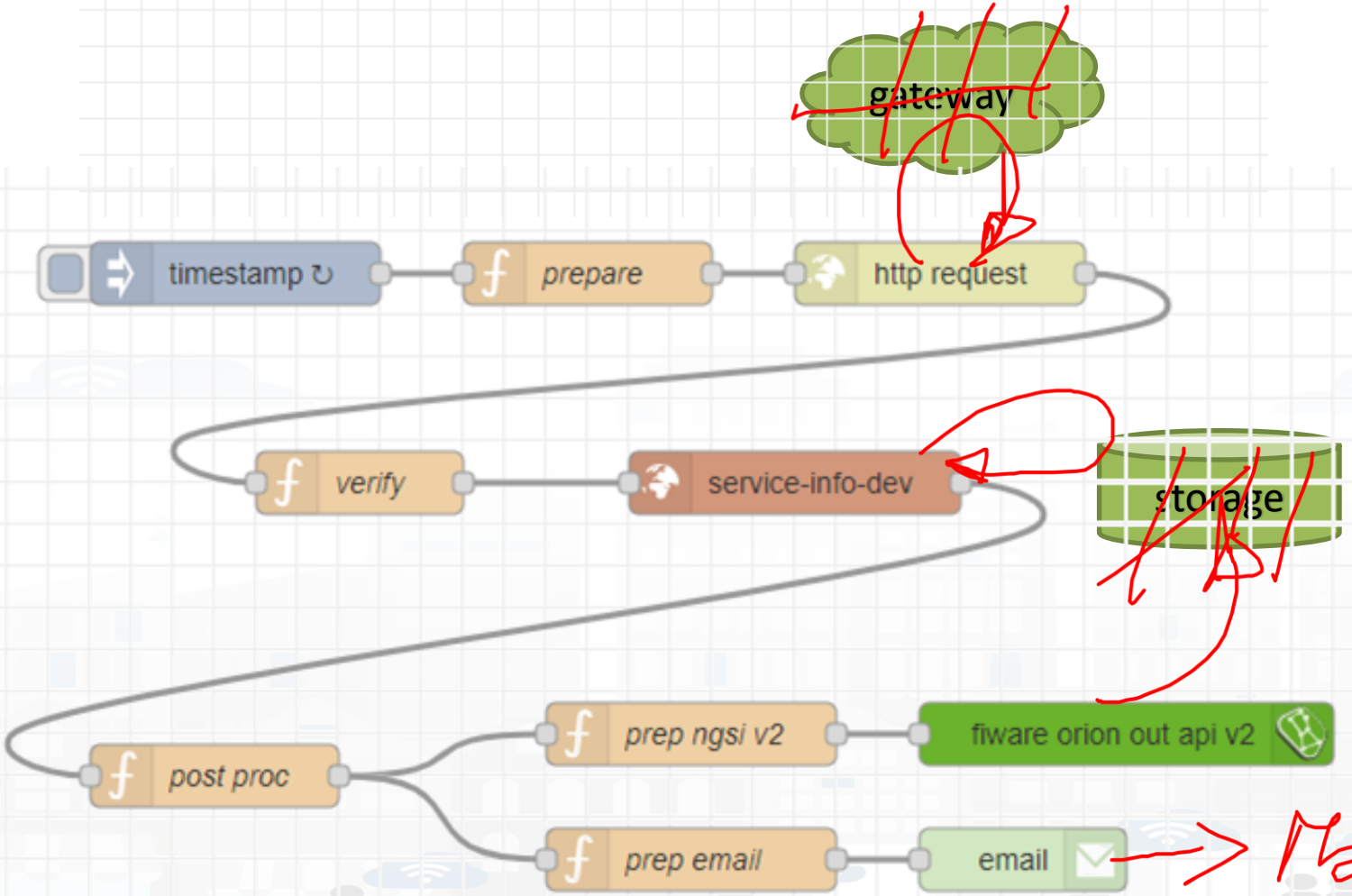
IoT App / Proc.Logic Design, for each Data/Event Flow



a. Periodically activate the flow
b. Call a gateway to get data
c. Verify the correctness of data
d. Enrich the data with other information coming from Cloud data into the storage
e. Transform the data in the correct forma
f. Send the data into the IoT Broker, and thus send the data in the storage on a specific IoT Device
g. Send also a notification via email


Snap4City
Switch To New Layout (Beta)
User: roottooladmin1, Org: DISIT
Role: RootAdmin, Level: 7
LOGOUT
My Snap4City.org
Tour Again
www.snap4solutions.org
ダッシュボード
Dashboards (Public)
My Dashboards in All Org.
Dashboards of My Organization
My Dashboards in My Organization
My Data Dash Dev OpenSrcDash
My Data Dashboard OpenSrcDash
Extra Dashboard Widgets
Notificator
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Entity Directory and Devices
Resource Manager
Development Tools
Management
Decision Support Systems
Deploy and Installation
Proc.Logic / IoT App
View as cards
Prev 1 ... 35 36 37 38 39 ... 62
Next
Filter
Create new
osm_ingestion
owner: osm_ingestor
OTE
owner: aliferisi
P007 Flags
owner: nicola_smartbed
owner: paolo.ant2
Panacea
owner: aliferisi
PANACEA
owner: javier
Panacea
PANACEA_AIT
paperNodeRed
owner: milestonetest
paperNodeRed
owner: envdatacollection
owner: simone_bassilichi
owner: testparkrss
IOT Application
Management
Create Proc.Logic/IoTApp
Application name:
Application type:
Basic
Advanced
Basic Debug
Advanced Debug
Web scraper (portia)
By pressing the Confirm bu
Privacy Policy.

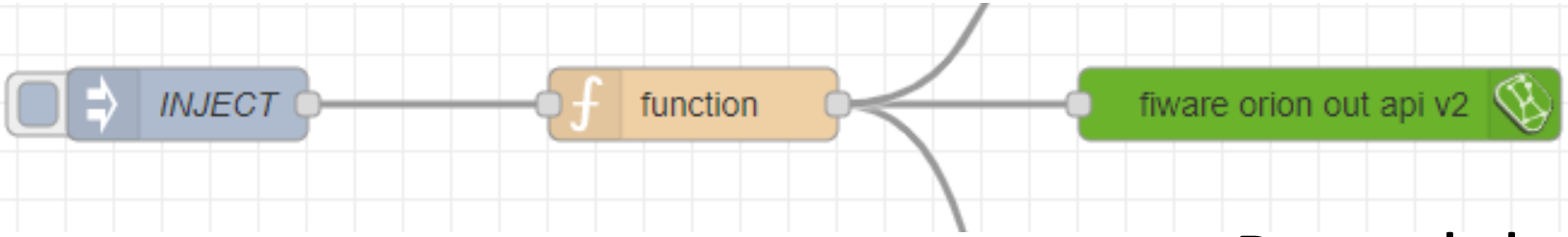
Proc.Logic (IoT App) Design, for each Data/Event Flow



a. Periodically activate the flow
b. Call a gateway to get data
c. Verify the correctness of data
d. Enrich the data with other information coming from Cloud data into the storage
e. Transform the data in the correct forma
f. Send the data into the Broker, and thus send the data in the storage on a specific Entity Instance
g. Send also a notification via email

Implicit services are not drawn

A sample of Data Ingestion

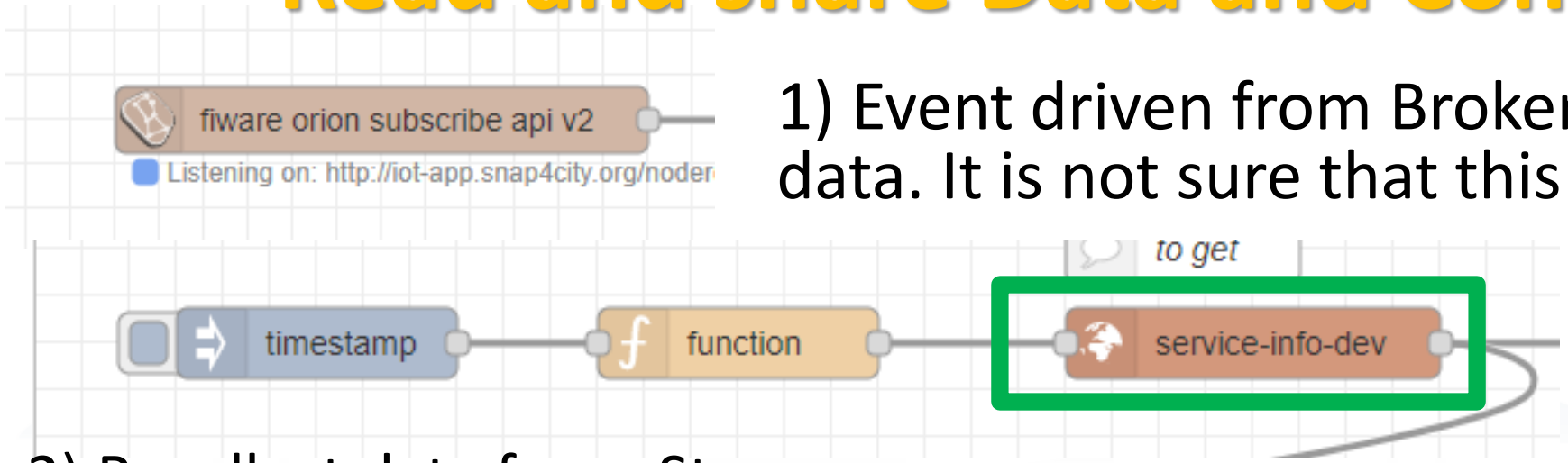


Function, example of NGSI V2 payload:

```
var time_now = new Date().toISOString();
var arandvalue = Math.random()
msg.payload =
        {"id":"mydev",
        "type":"mydevSensor",
        "anID":{"type": "integer", "value": "http://www.disit.org/km4city/resource/iot/...../anuser"},
        "VDDValue":{"type":"float","value":arandvalue},
        "dateObserved":{"type":"string","value":time_now},   // it is a time serie
        "latitude":{"type":"float","value":"28.61810"},      // it may move over time
        "longitude":{"type":"float","value":"11.34300"},     // it may move over time
        "status":{"type":"integer","value":34}
        }
return msg;
```

Posted data on IoT Brokers green nodes are automatically saved into the data Storage

Read and share Data and Context Data



1) Event driven from Broker, read last context data. It is not sure that this change is on Storage

2) Recollect data from Storage
   - This node uses the Smart City API
- Any External Application can get the same data in authenticated authorized manner via Smart City API
- Smart City API is a better approach instead of producing a file outside or providing data via some local API service created from IoT Application (feasible but not protected)

Snap4City MicroServices and programming Patterns



UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

Sept 2024 collection

Two Snap4City Libraries

SNAP4CITY | Node-RED

https://flows.nodered.org/search?term=snap4city

- common
- function
- network
- input
- output
- sequence
- parser
- storage
- social
- advanced
- Advanced FTP
- location
- NGSI
- lwm2m
- S4C SearchDev
- S4CUtility
- S4CMapping
- S4CManagement
- S4CDataAnalytic
- S4CBigData
- S4CIOTApp
- S4COpenMaint
- S4CIoT
- S4CWhatIf
- S4C Search
- S4CData
- S4CKPIData
- S4CDashboard
- S4C Sigfox
- S4CLogDev
- S4CView
- S4C Social
- dashboard
- time

S4C SearchDev

service search
service search near gps position
service search near service
service search within gps area
service search within wkt area
service search within stored wkt area
service search by municipality
service search by queryid
full text search dev
full text search within wkt area
full text search within gps area
full text search near gps position
full text search exp
event search dev
event search exp
event search within wkt area
event search within gps area
event search near gps position
address search near gps position
geometry search near gps position
address poi search by text
address poi search by text exp
address poi search by text near gps position
bus routes search
bus routes search near gps position
bus routes search within gps area
bus routes search within wkt area
tpl routes
tpl stops

S4CUtility

service info dev
distance from coordinates

point within polygon
routing
heatmap picker
coordinates to address
service info
edge-tunnel-to-cloud

S4CMapping

service info mapped
mapping
set mapping
check exist job
check exist trigger
is in standby mode
is shutdown
is started
get currently executing jobs

get job detail
get triggers of job
get job group names
get trigger group names
get paused trigger groups
get job fire times
get system status
trigger job
pause all
pause trigger
pause triggers
resume all
resume job
resume jobs
resume trigger
resume triggers

notificator history events

S4CDataAnalytic

descriptive statistics
trend plot
time series predictions
machine learning predictions
anomaly detection
plumber data analytic
python data analytic

insert
datagate search
datagate create
portia crawler

S4CIOTApp

iotapp restart
iotapp upgrade
ownership

S4C Search

service search near marker
service search within circle
service search within polygon
service search along path
full text search within circle
full text search within polygon
full text search along path
full text search usr
event search near marker
event search within circle
event search within polygon
event search along path
event search usr
address search near marker
geometry search near marker
address poi search by text usr
address poi search by text near marker
address poi search by text within circle
bus routes search near marker
bus routes search within circle
bus routes search within polygon
tpl agencies
tpl lines

tpl routes by agency
tpl routes by line
tpl stops by route
tpl stop timeline
recommendatio within circle
value type search near marker
value type search within circle
value type search within polygon
value type search along path

S4CData

get my data
get my delegator
get my delegated
get my activity

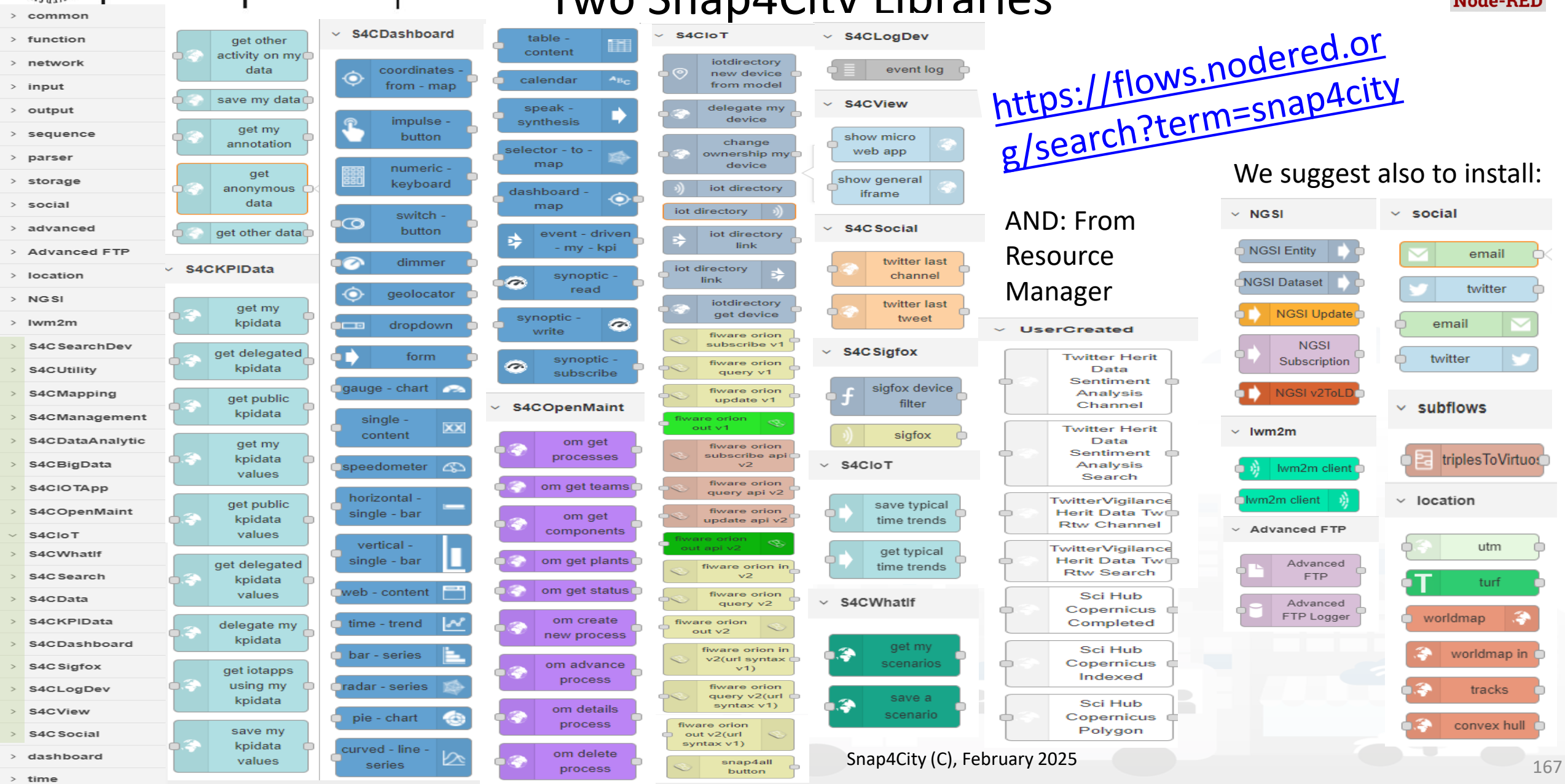
Sept 2024 collection

Two Snap4Citv Libraries

https://flows.nodered.org/search?term=snap4city

We suggest also to install:

AND: From Resource Manager



The Processing Logic (IoT App) microservices

Actually, there are more than 180 nodes/blocks in the Snap4City libraries on Processing Logic (IoT App) which can really facilitate your life and save you time in producing Smart Applications for composition of the following microservices and using those that you can install from internet, thousands of functionalities:

- **Data ingestion**: more than 100 protocols IOT and Industry 4.0, web Scraping, external services, any protocol database, etc.
- **Data access**: save/retrieve data, query search on expert system, georeverse solution, search on expert system Km4City ontology, call to Smart City API, etc.
- **Data Transformation/transcoding:** binary, hexadecimal, XML, JSON, String, any format
- **Integration**: CKAN, Web Scraping, FTP, Copernicus satellite, Twitter Vigilance, Workflow OpenMaint, Digital Twin BIM Server, any external service REST Call, etc.
- **Manipulation of complex data**: heatmaps, scenarios, typical time trend, multi series, calendar, maps, etc.
- **Access to Smart City Entities and exploitation of Smart City Services**: transport, parking, POI, KPI, personal data, scenarios, etc.
- **Data Analytic**: managing Python native, calling and scheduling Python/Rstudio containers as snap4city microservices (predictions, anomaly detection, statistics, etc.)
- **User interaction on Dashboard**: get data and message from the user interface, providing messages to the user (form, buttons, switches, animations, selector, maps, etc.), send data to special graphical widgets: D3, Highcharts, etc.
- **Custom Widgets**: SVG, synoptics, animations, dynamic pins on maps, etc
- **Event management**: Telegram, Twitter, Facebook, SMS, WhatsApp, CAP, etc.
- **Special tools as**: routing, georeverse, Twitter Vigilance and sentiment analysis, etc.
- **Hardware Specific Devices**: Raspberry Pi, Android, Philips, video wall management, etc.
- **Etc**. etc.

Standards and Interoperability (10/2024)

Compliant with:

- **IoT:** NGSI V2/LD, LoRa, LoRaWan, MQTT, AMQP, COAP, OneM2M, TheThingsNetwork, SigFOX, Libelium, IBIMET/IBE, Enocean, Zigbee, DALI, ISEMC, Alexa, Sonoff, HUE Philips, Tplink, BACnet, TALQ, Protocol Buffer, KNX, OBD2, Proximus, ..
- **IoT model:** FIWARE Smart Data Model, Snap4City IoT Device Models
- **General**: HTTP, HTTPS, TLS, Rest Call, SNMP, TCP, UDP, SOAP, WSDL, FTP, FTPS, WebSocket, WebSocket Secure, GML, WFS, WMS, RTSP, ONVIF, AXIS TVCam, CISCO Meraki, OSM, Copernicus, The Weather Channel, Open Weather, OLAP, VMS Milestone, TIM, HERE, ….
- **Formats**: JSON, GeoJSON, XML, CSV, GeoTIFF, OWL, WKT, KML, SHP, db, XLS, XLSX, TXT, HTML, CSS, SVG, IFC, XPDL, OSM, Enfuser FMI, Lidar, glTF, GLB, DTM, GDAL, Satellite, D3 JSON, …
- **Database**: Open Search, MySQL, Mongo, HBASE, SOLR, SPARQL, ODBC, JDBC, Elastic Search, Phoenix, PostGres, MS Azure, ..
- **Industry**: OPC/OPC-UA, OLAP, ModBUS, RS485, RS232,..
- **Mobility**: DATEX, GTFS, Transmodel, ETSI, NeTEx, ..
- **Social**:Twitter, FaceBook, Telegram, ..
- **Events**: SMS, EMAIL, CAP, RSS Feed, ..
- **OS**: Linux, Windows, Android, Raspberry Pi, Local File System, AXIS, ESP32, etc.

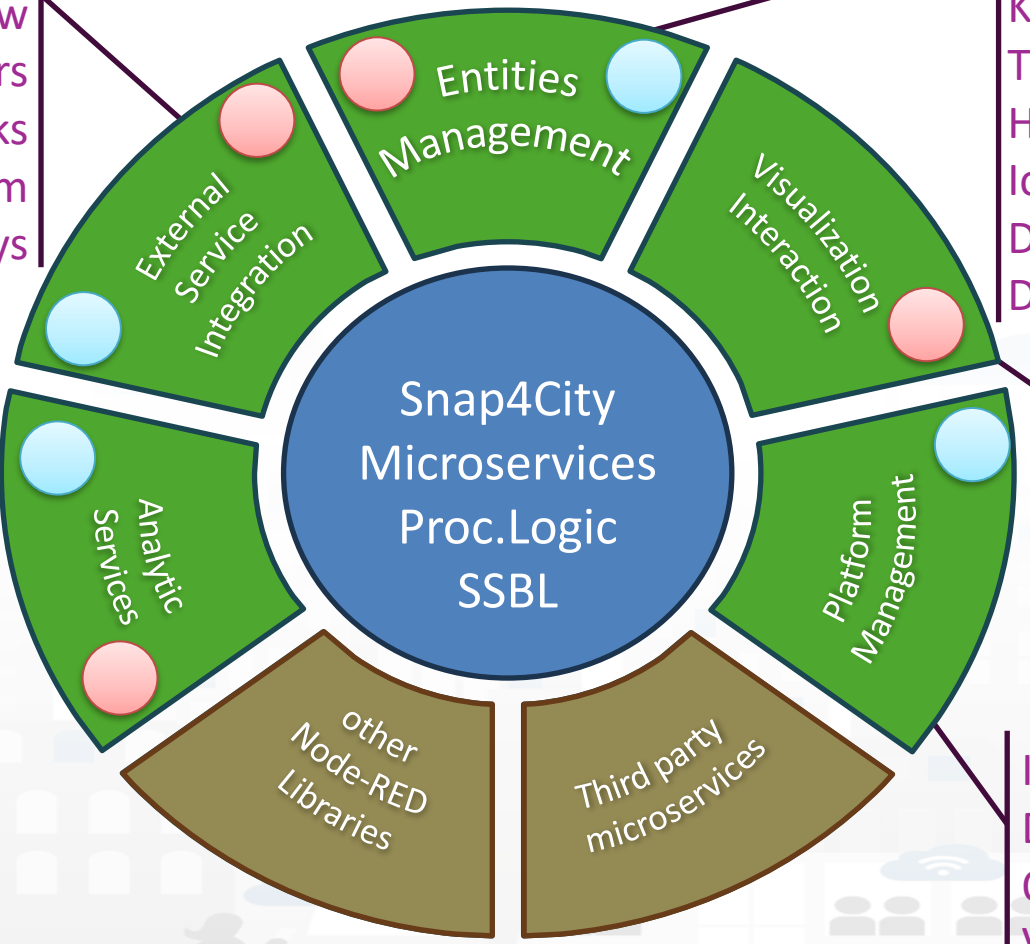
https://www.snap4city.org/65

Snap4City Libraries on Node-RED

- https://flows.nodered.org/search?term=snap4city
  - https://flows.nodered.org/node/node-red-contrib-snap4city-user
  - https://flows.nodered.org/node/node-red-contrib-snap4city-developer
    - https://flows.nodered.org/node/node-red-contrib-snap4city-d3-dashboard-widgets
    - https://flows.nodered.org/node/node-red-contrib-snap4city-tunnel
    - https://flows.nodered.org/node/node-red-contrib-snap4city-milestone
  - https://flows.nodered.org/node/node-red-contrib-snap4city-clearml

MicroServices Areas

> 60.000 downloads



Open Data CKAN
Ticket Management, workflow
BIM Servers
Social Networks
Video Management system
Gateways

Data Load / Search / Retrieval
KPI, POI, GIS Data, Scenarios
Time Series, Public transport
High Level Types: heatmaps, ODM,…
IoT / Entity Discovery
Delegation Management
Data Mapping

Data Analytics
Statistic, Optimization
Simulation
Artificial Intelligence
What-if Analysis Support
Geo Utilities Support
Routing & Traffic Flow
MLOps support
Python support
R Studio Support

Dashboards
Widgets: Graphic Libraries
Interactive Widgets
Maps, 3D representations
Synoptics, External Content
Micro Web App

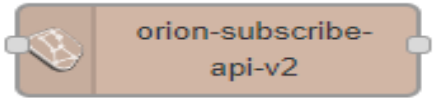
IoTApp Management
Data Logs, A&A, Security
Ownership Management
VPN remote access

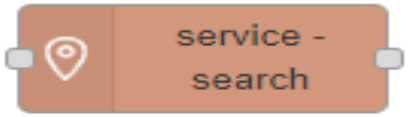
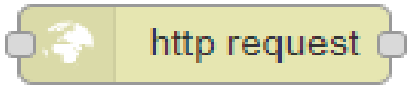
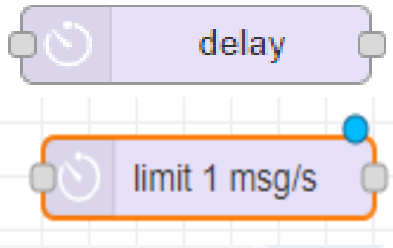
examples

Saving Data on Storage

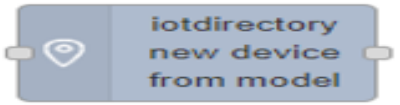
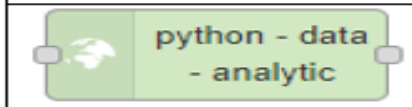
Even Driven

Get Data from Storage

Node shape	Description	Snap4City or standard
	To **generate injection messages** into a flow, scheduled/periodic or on manual demand by click it on left.	standard
	**DATA TRANSFORM** A **JavaScript function**, from a JSON input to one or more JSON outputs, which can be produced by setting it.	standard
	**SAVE to STORAGE via internal BROKER** To send an Entity Message of an Entity Instance into the storage. The Entity Instance has to be registered on Entity Directory (IoT Directory) and you have to be the owner or to be delegated in READ-WRITE to send messages to it. The node represents the broker, so that the same node can be used to send any Entity Message you need. Please manage the error in output.	Snap4city
	**SUBSCRIBE** to an Entity change on BROKER To subscribe the Processing Logic (IoT App) to receive event-driven notifications related to Entity Instances changes. The node is substantially a listener connected to an Orion Broker. You can subscribe to many Entities and then to get all of them from the output of the listener. The new version will go to provide an input port to send at this listener multiple subscriptions. **PLEASE NOTE THAT ALL THAT YOU CAN DO IN MQTT CAN BE DONE IN ORION BROKER NGSI. Moreover, Orion broker is authenticated, in SSO, provides JSON, etc. This node-red block allows you to subscribe to a topic / device and get event driven actions on IoT App directly.** Please manage the error in output.	Snap4city
	**READ from STORAGE** Query call to Smart City API to get any information about a SURI, ServiceURI. There are many other Nodes which can be used to pose Smart City API queries in very simple manner and recover vectors of ServiceURIs. Please manage the error in output.	Snap4city

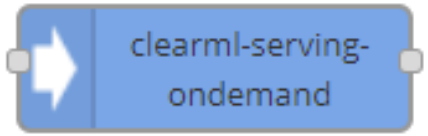
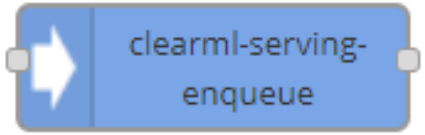
Get Data from Storage		**SEARCH on STORAGE** To perform queries on the storage to obtain a list of ServiceURI. The nodes of this family can allow you to perform searching queries by filtering for distance, area, subnature/category, values of attributes, time period, etc. Please manage the error in output.	Snap4city
Send email		Send email. With other nodes you can send Telegram, SMS, etc.	standard
Gen/access to HTTP, HTML pages		To send a **REST CALL** (get, post, etc.). Please USE THIS NODE ONLY for the access at external API and not to access at the Snap4City API for which a lot of MicroServices are accessible as NODEs/Blocks in the Processing Logic and they are simpler to be used and ready to use. Please manage the error in output.	standard
Monitor messages		A block which is printing on **debug** view the data JSON passed in its input. Please note that the node can be tuned to provide only msg.payload or the full JSON message, change configuration of the node.	standard
Stream Delay, limiting rate		A node to insert a delay to each message arriving, or to limit the rate of messages in output. In some cases, the node creates a buffer of messages regularizing the rate in output if the rate in input is greater in some moments.	standard

examples

	A block which is printing on **debug** view the data JSON passed in its input. Please note that the node can be tuned to provide only msg.payload or the full JSON message, change configuration of the node.	standard
	To **create an Entity Instance** (device instance) from a model prepared on Entity Directory (IoT Directory).	Snap4city
	To **change the ownership** of an Entity Instance (IoT Device).	Snap4city
	To **delegate a certain Entity Instance** (IoT Device) to some other user for which you have to know the Nickname. Delegations can be: Read access, Read write, Modify (to modify the Entity Instance structure)	Snap4city
	To **show something on Snap4City dashboard** with a single content widget (one of the simplest widgets). A large set of dashboard nodes/widgets to send and retrieve data to/from dashboards are provided. This specific Nodes allows to send on dashboard HTML formatted messages with some limitations. Full HTTP widget is also accessible. See in the following section for the Full list of Nodes for	Snap4city
	**MQTT broker listener**, to receive messages from the Broker. Another similar node can be used to send MQTT messages to the MQTT broker. This node allows to perform a subscription to a topic of the MQTT broker.	standard
	**DATA ANALYTICS** Request performed on a Container including a Python data analytics, which is loaded into the node and the container is created at the first Deploy of the Processing Logic. Similar Approach is performed for RStudio Data Analytics.	Snap4city
	**SPLIT:** This block takes in input a buffer, or an array, or an object and split it on a set of messages in output, for each line in the buffer, each element of the array, each element in the object, respectively.	standard
	**JOIN:** This block takes in input a set of messages and join/merge them into a single message (string, buffer, array or object, etc.), on the basis of specific criteria.	standard

DO NOT USE NOW

Integration with ClarML, MLOps, AI, HPC

 	**Exploit MLOps Clusters of CPUs/GPUs and HPC** The request of analytic execution is performed: (i) on demand, which implies the allocation of a process on container allocated on demand on some cluster and perform a single execution; (ii) on a stable analytic process which expose some API to respond in fast manner to request without any overhead of loading container and processes in memory.	Snap4City

The Proc.Logic (IoT App) microservices

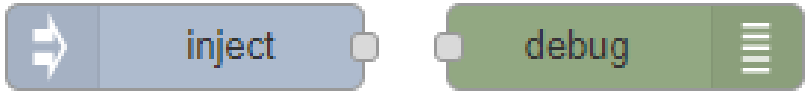
Actually, there are more than 180 nodes/blocks in the Snap4City libraries on Processing Logic (IoT App) which can really facilitate your life and save you time in producing Smart Applications for composition of the following microservices and using those that you can install from internet, thousands of functionalities:

- **Data ingestion**: more than 100 protocols IOT and Industry 4.0, web Scraping, external services, any protocol database, etc.
- **Data access**: save/retrieve data, query search on expert system, georeverse solution, search on expert system Km4City ontology, call to Smart City API, etc.
- **Data Transformation/transcoding:** binary, hexadecimal, XML, JSON, String, any format
- **Integration**: CKAN, Web Scraping, FTP, Copernicus satellite, Twitter Vigilance, Workflow OpenMaint, Digital Twin BIM Server, any external service REST Call, etc.
- **Manipulation of complex data**: heatmaps, scenarios, typical time trend, multi series, calendar, maps, etc.
- **Access to Smart City Entities and exploitation of Smart City Services**: transport, parking, POI, KPI, personal data, scenarios, etc.
- **Data Analytic**: managing Python native, calling and scheduling Python/Rstudio containers as snap4city microservices (predictions, anomaly detection, statistics, etc.)
- **User interaction on Dashboard**: get data and message from the user interface, providing messages to the user (form, buttons, switches, animations, selector, maps, etc.), send data to special graphical widgets: D3, Highcharts, etc.
- **Custom Widgets**: SVG, synoptics, animations, dynamic pins on maps, etc
- **Event management**: Telegram, Twitter, Facebook, SMS, WhatsApp, CAP, etc.
- **Special tools as**: routing, georeverse, Twitter Vigilance and sentiment analysis, etc.
- **Hardware Specific Devices**: Raspberry Pi, Android, Philips, video wall management, etc.
- **Etc**. etc.

Some patterns

Periodic or Sporadic events to data collection, processing

1) Hello world of node-red, the inject may provide a string to the debug.



2) Hello world of node-red at two steps, the inject provides a push while a JSON is created into the function as *msg.payload = {……}* and sent/shown to/by the debug.

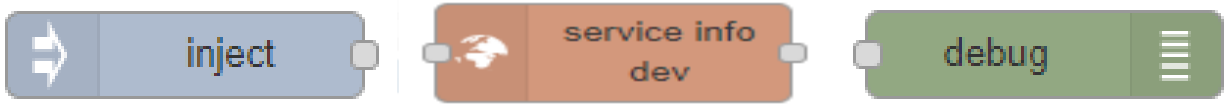


Even Driven

3) Event data reception from an MQTT broker, transformation and send it to the storage pushing data into the Orion Broker V2.

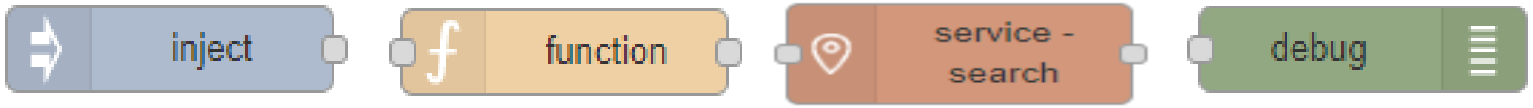


4) request on inject of a SURI to the storage to see data on debug.

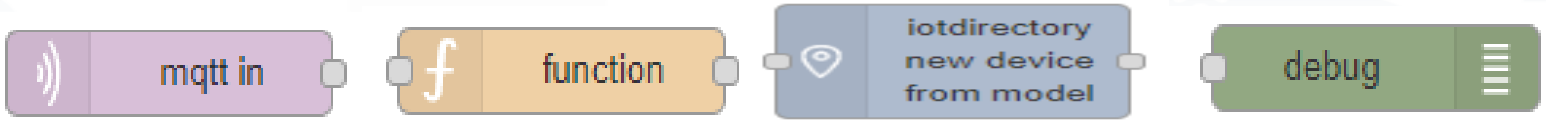


Even Driven

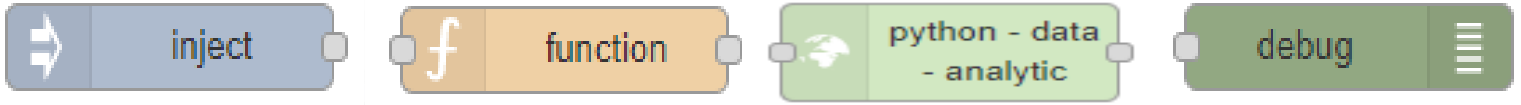
1) Preparation of data request on function, query to the storage and see data result on debug.



2) Event data reception from an MQTT broker, transformation to create an Entity Instance from a known Entity Model, debug to see eventual errors, for example if the device is already present (to avoid production of error, one may verify if the Entity Instance is already present by posing a query on the system):

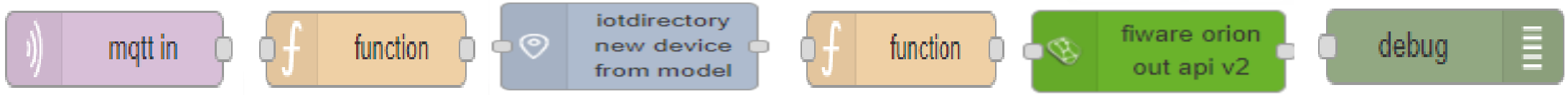


3) Preparation of data parameters on function, request computing Data Analytic, see data result on debug.

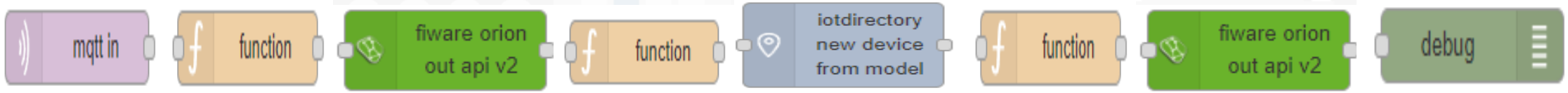


Typical strange patterns that may be not efficient in most cases:

A. data reception from an MQTT broker, their transformation to create an Entity Instance from a known Entity Model, contextually to create and send an Entity Message into newly created Entity Instance, the debug to see eventual errors. This approach is typically strange since at each new message the Entity Directory is queried to see if the Entity is already be created and if not to create it and then pass the data to register the message. In most cases, it is much better to decouple the activity of creating with respect to that of sending message. In fact, this approach would largely reduce the ingestion rate and probably when the Entities are already created would create un-useful workload on Entity Directory (IoT Directory).

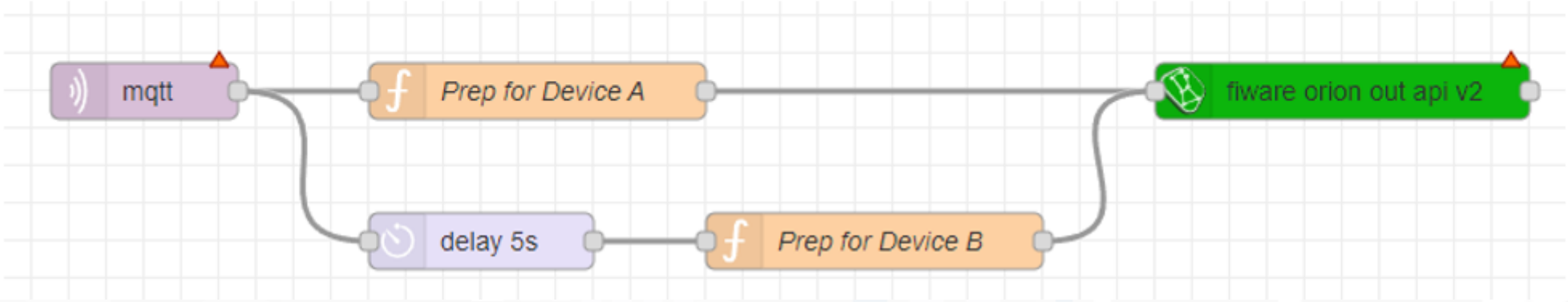


In most cases, it should be done the opposite: try to send the Entity Message, if it fails than create a new Entity Instance by known model, and if successful send again the Entity Message, or just wait for the new message to save it the first.



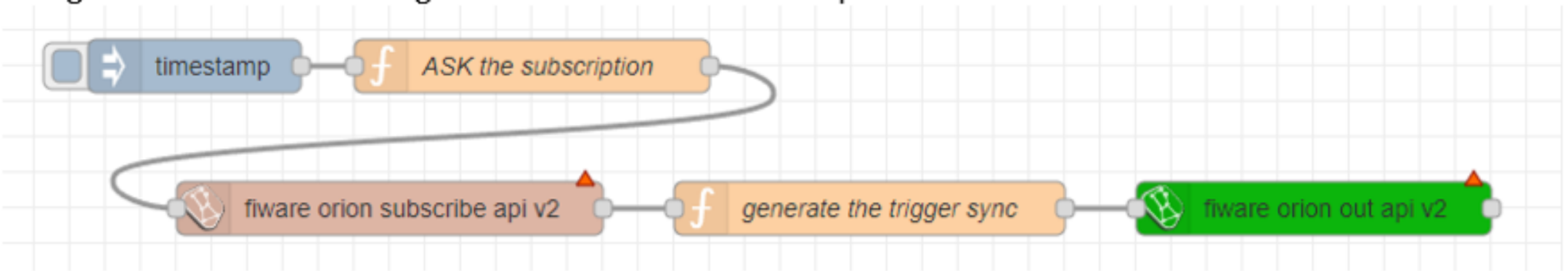
Sync data changes on Entities from an Event

If I would like to synchronize a device data A with another B by trigger event, I can do it in several manners. The first case would be the simplest. A triggering message arrives from MQTT event or from some NGSI ORION, or from some MyKPI, from dashboard event button, or email or anything, it does not matter. I can use two functions to prepare the message for A device and B device as follow:

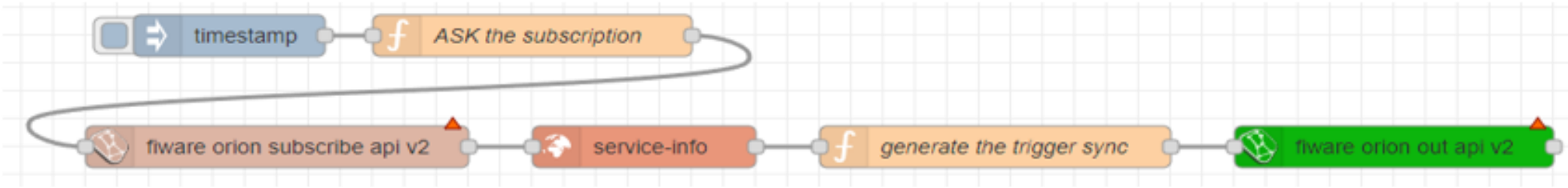


Subscribe on event from Broker, be carefull....

If the event for triggering is from another device/entity changed by some action performed posting a data on Orion Broker V2, you can subscribe with the event on the Orion broker by using a specific Node (do it once otherwise you risk receiving many events). Every time the device / entity receives a message you can take it and generated a new message for a different device and post it on Orion API V2.



If you need to verify if the new data has been changed, you can read the last value of recipient Device/entity to compare and decide to update or not:



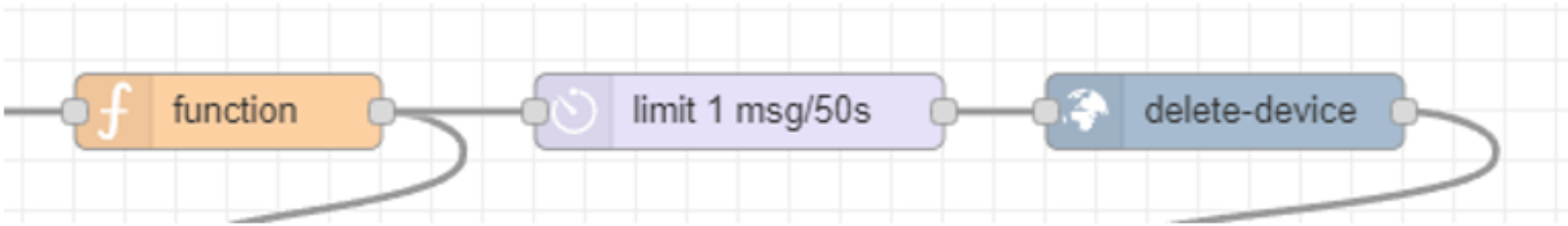
Delete Devices

IV.C.1.e- Delete Devices

Recently a node to delete devices has been added. It has to be used veery carefully since to delete data is always a terrible activity in a big data storage.

The delete of a device is allowed only for the Owner of the device and the root administrator of the platform. The device delete can be also performed from the Entity Directory and now with the Delete-Device node can be performed also from Proc. Logic / IoT App.

The classic pattern is as follows, including preparation, a RATE Limitation avoiding to provide more than one delete message every 50 seconds:



The delete device node needs in input Device ID and Broker ID. All data that you can recover from the Entity Directory.

- **Please note that**
  - the most important blocks nodes to interact with the platforms are reported in these tables to familiarize with the main concepts. ***They are actually families of blocks/nodes*** since many others are present that allow you to perform a very large number of other features.
  - You do not need to take care of Authentication and Authorization, all is performed via SSO, Access Tokens.
  - YOU DO NOT HAVE TO ACCESS AT THE API all is provided in terms of NODEs/BLOCKS into IoT APP. Everything can be parametrized via JSON passed in input to the nodes.
  - nodes input as JSON is primary mode for setting parameters
    - While they can be also configured from their user settings via user interface

Programming Paradigm comparison

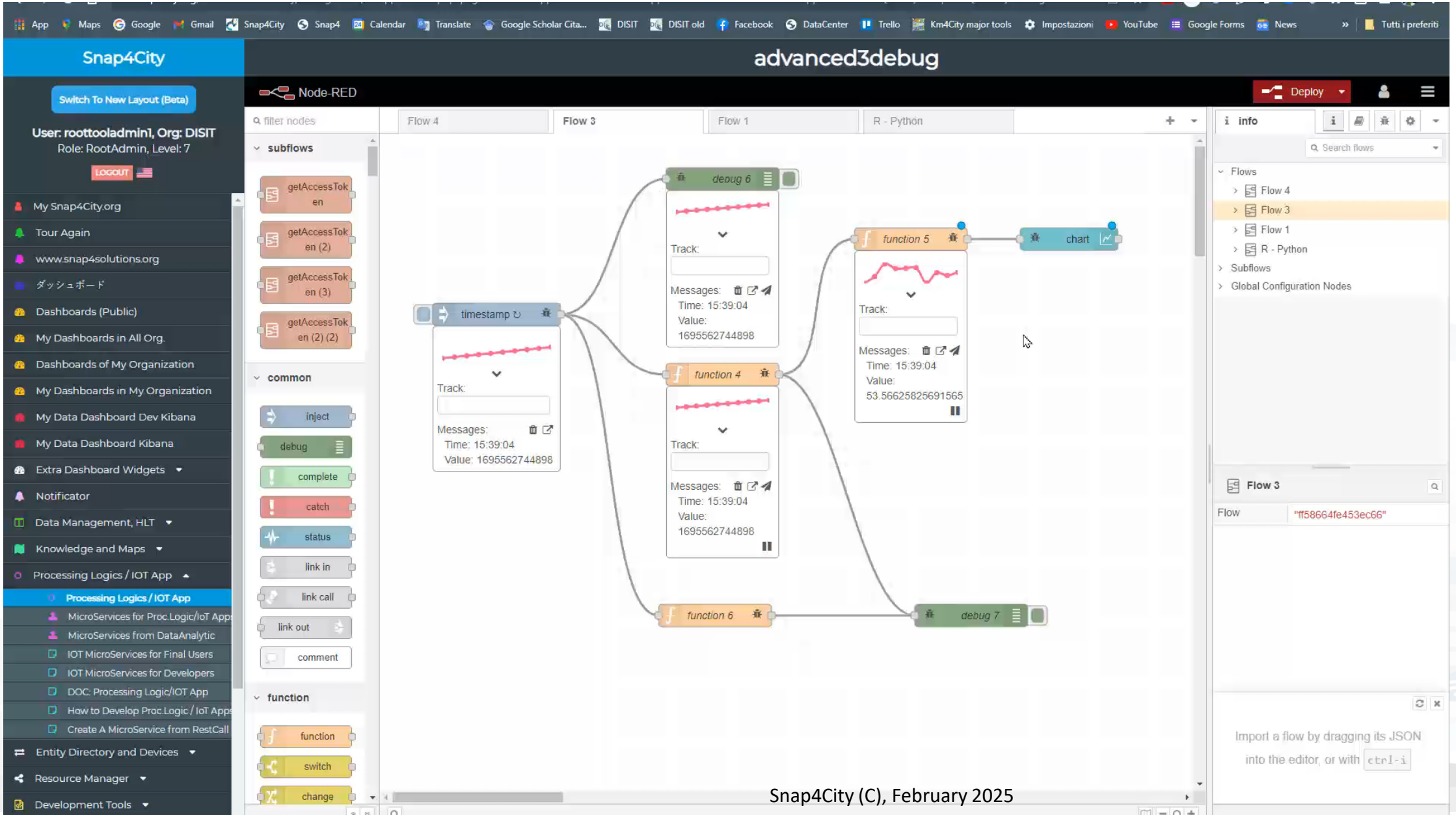
Execution Environment	JavaScript Runs in the browser (e.g., Chrome, Firefox, Edge)	NODE-RED + Snap4City
Programming Model	Procedural, asynchronous, and event-driven	Flow-based programming with visual node wiring, Event-driven, asynchronous, and non-blocking
Use Case	Enhancing web pages, UI interactivity, handling DOM events	IoT, automation, API orchestration, workflow management
Event Handling	Uses addEventListener, Promises, and Async/Await	Event-driven flows triggered by nodes (e.g., HTTP requests, MQTT, WebSockets)
Data Processing	Manipulates DOM, handles user inputs, and fetches API data	Processes messages between nodes (JSON-based) + HTTP, WS ….
Scalability	Limited to browser capabilities; depends on front-end frameworks (React, Angular, Vue)	Highly scalable, can integrate with databases, cloud services, and IoT devices, supports microservices and API-driven architecture
Integration	Primarily interacts with REST APIs, WebSockets, and browser storage	Easily integrates with MQTT, databases (MongoDB, MySQL), REST APIs, WebSockets, and hardware devices
Security Considerations	Runs in a sandboxed environment, limited file system access	Automated inplatform and Edge security handling, as it runs on a server and interacts with external systems. Additional security
Development Approach	Written manually in code editors (VS Code, etc.)	Low-code, flow-based, drag-and-drop UI
Debugging	Uses browser DevTools (console, breakpoints)	Uses built-in debug node and logs, additional Debugging capabilities
Deployment	Deployed in web browsers, hosted via CDNs or bundled	Runs on local, edge and on cloud servers, Docker, Kubernetes, or standalone Node.js environments

Comparison

paradigm	JavaScript traditional	NODE-RED + Snap4City
For on vector iteration	For on vector iteration	Split and single message processing
Variable on process	Local Variables	Status variable on **msg.xxxxx** to be passed ahead, custom made but temporary
Status Variable	Local variables	Local variable of flow and global node-RED
General Status interprocesses	On local database	On MyKPI, On Entities (open search), scalable, GDPR, etc.
User Interface in execution	HTML, not scalable, unsafe	Local Dashboards (to be avoided), Snap4City Dashboard SSBL WS (to be carefully used, Snap4City Dashboards from Storage !!!
storage	Local Databases, remote databases	Preferring Snap4City: (i) NGSI internal broker for internal database, and (ii) MyKPI, (iii) other facilities of Snap4City.., lastly other Databases
EventDriven	Some solutions, WS, MQTT, etc.	**Any NODE**, preferred: NGSI, MQTT, WS, etc.
API exposed	Local API, not scalable, unsafe	Do not create local API, use SCAPI
Forms from HTML	HTML, not scalable, unsafe	CSBL dashboards to be safe, secure and scalable
security	Hand made	OAuth all automated, GDPR, etc.
DA, ML, AI	Hand Made	All automated, ClearML, API, etc.

Snap4City Node-RED Debug extending Cauldron




App Maps Google Gmail Snap4City Snap4 Calendar Translate Google Scholar Cita... DISIT DISIT old Facebook DataCenter Trello Km4City major tools Impostazioni YouTube Google Forms News Tutti i preferiti
Snap4City
advanced3debug
Switch To New Layout (Beta)
User: roottooladmin1, Org: DISIT
Role: RootAdmin, Level: 7
LOGOUT
My Snap4City.org
Tour Again
www.snap4solutions.org
ダッシュボード
Dashboards (Public)
My Dashboards in All Org.
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
My Data Dashboard Kibana
Extra Dashboard Widgets
Notificator
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Processing Logics / IOT App
MicroServices for Proc.Logic/IoT App
MicroServices from DataAnalytic
IOT MicroServices for Final Users
IOT MicroServices for Developers
DOC: Processing Logic/IOT App
How to Develop Proc.Logic / IoT Apps
Create A MicroService from RestCall
Entity Directory and Devices
Resource Manager
Development Tools
Node-RED
Deploy
filter nodes
Flow 4
Flow 3
Flow 1
R - Python
subflows
getAccessTok en
getAccessTok en (2)
getAccessTok en (3)
getAccessTok en (2) (2)
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
change
timestamp
Track:
Messages:
Time: 15:39:04
Value: 1695562744898
debug 6
function 4
function 5
53.56625825691565
chart
function 6
debug 7
info
Search flows
Flows
Flow 4
Flow 3
Flow 1
R - Python
Subflows
Global Configuration Nodes
Flow
"ff58664fe453ec66"
Import a flow by dragging its JSON into the editor, or with ctrl-i
IotAppDebug
IOT Application
owner: testnewimages
Management

Snap4All Mobile App

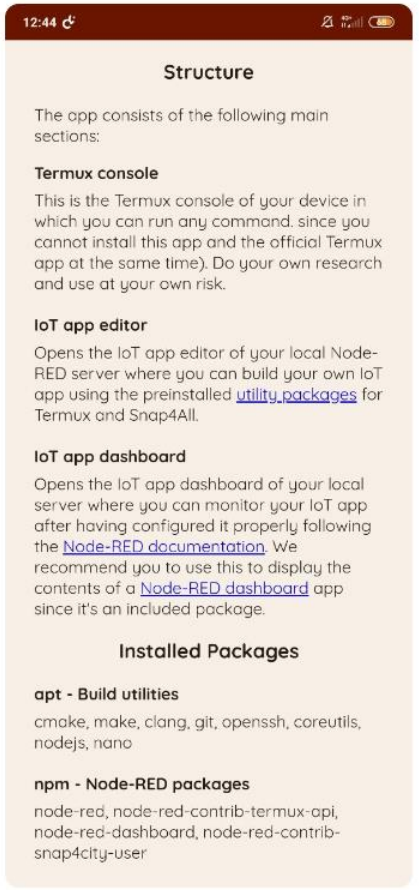
Node-RED on Android



Snap4All mobile app for Android



(a) Home



(b) Info



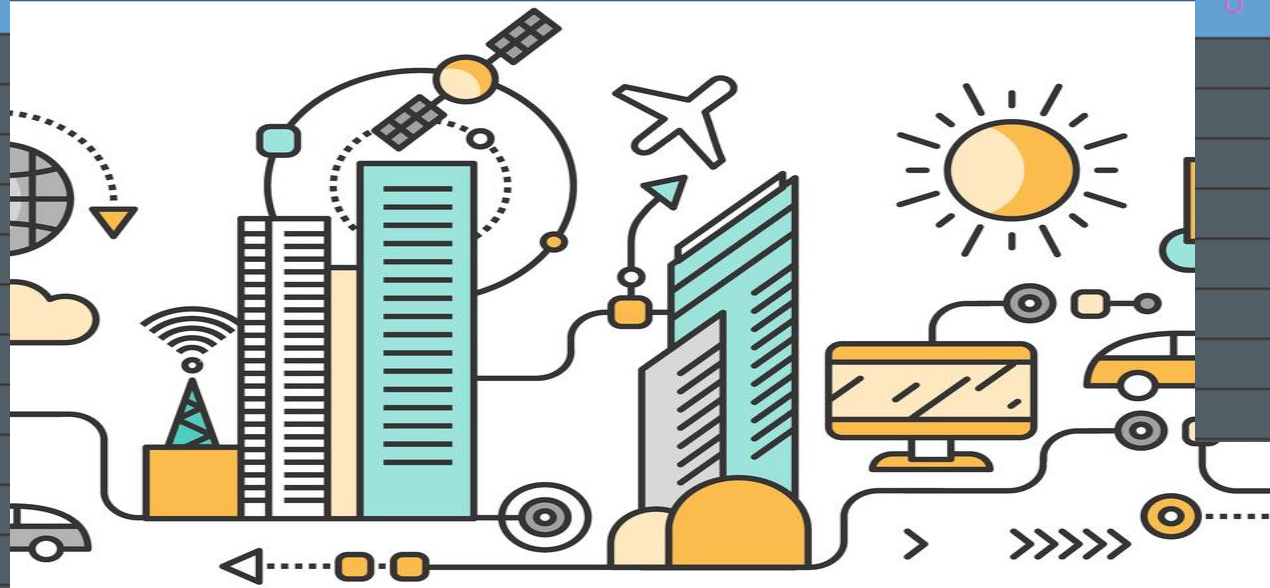
(c) Console



https://www.snap4city.org/download/video/Snap4All-v1.0.5-large.apk

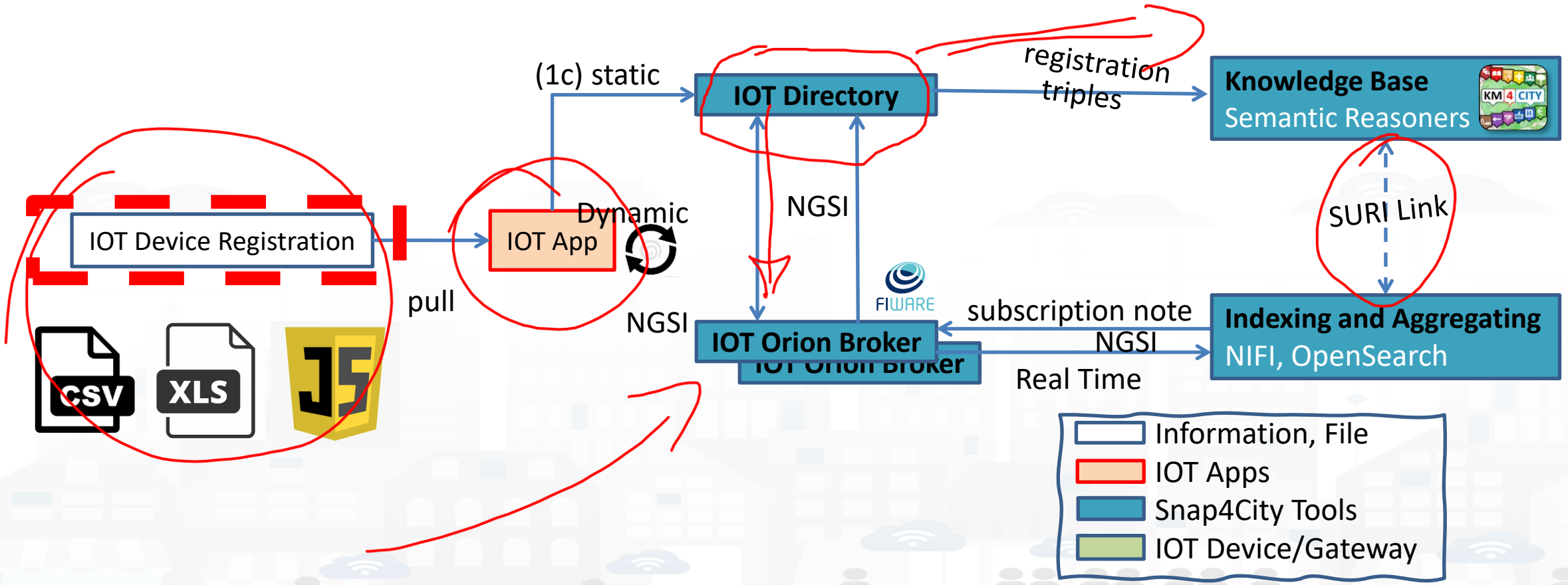
Entity / Device Registration from IOT App/Proc.Logic (automation)

- IOT Directory and Devices
  - My IOT Sensors and Actuators
  - IOT Sensors and Actuators
  - IOT Devices
  - IOT Devices Management
  - IOT Device Discovery
  - IOT Brokers
  - IOT Device Models
  - IOT Devices Bulk Registration
  - IOT Broker Periodic Update setti...
  - IOT Orion Broker Mapping Rules...
  - Doc: IOT Directory and Devices
  - Create an IOT Device Instance
  - Create an IOT Device Model
  - Add an IOT Device into Snap4Cit...



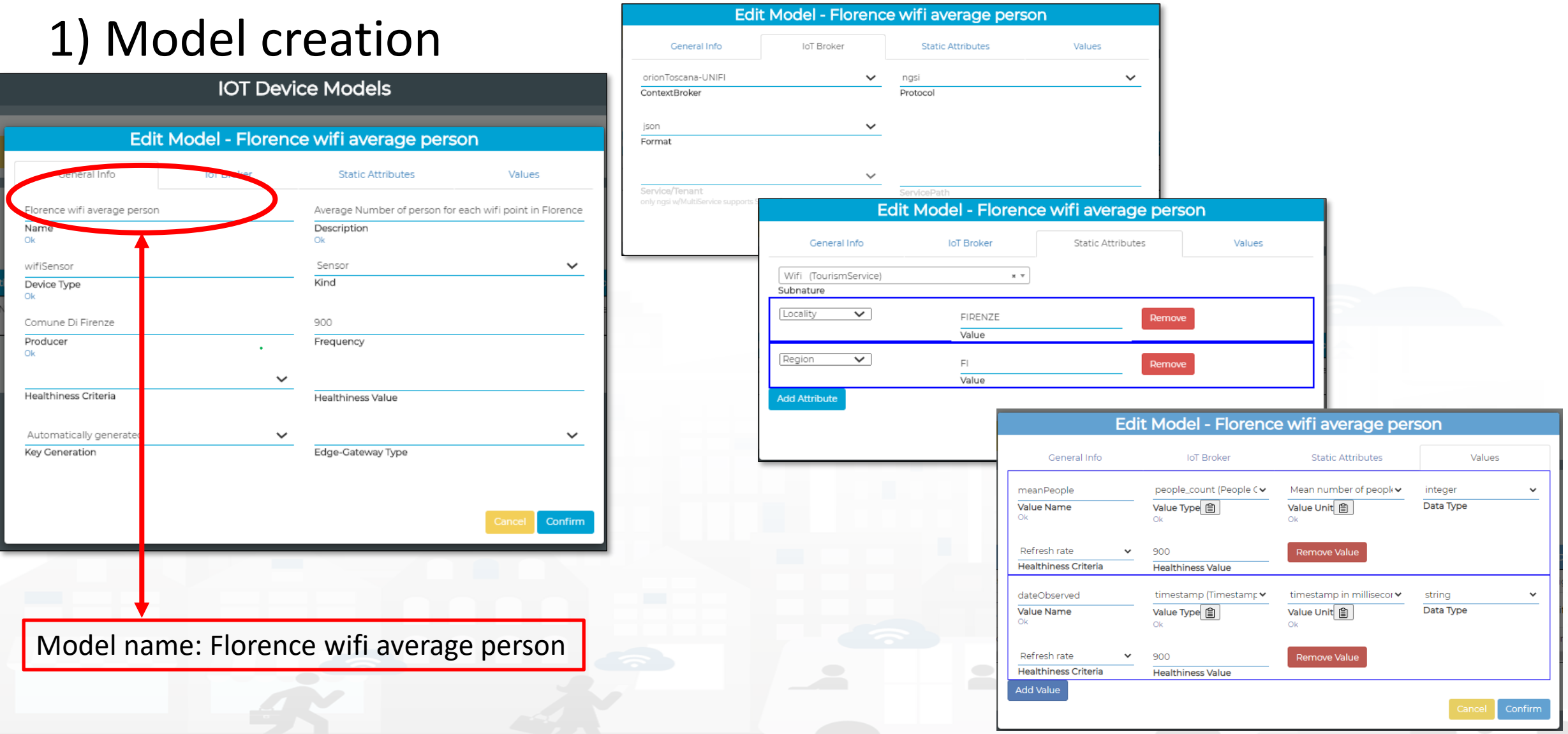
- IOT Applications
  - IOT Applications
  - MicroServices for IOT Applicatio...
  - MicroServices from DataAnalyti...
  - IOT MicroServices for Final Users...
  - IOT MicroServices for Developers...
  - Doc: IOT Applications
  - How to Develop IOT Application...
  - Create A MicroService from Rest...

Snap4city Data Ingestion Flow Diagram



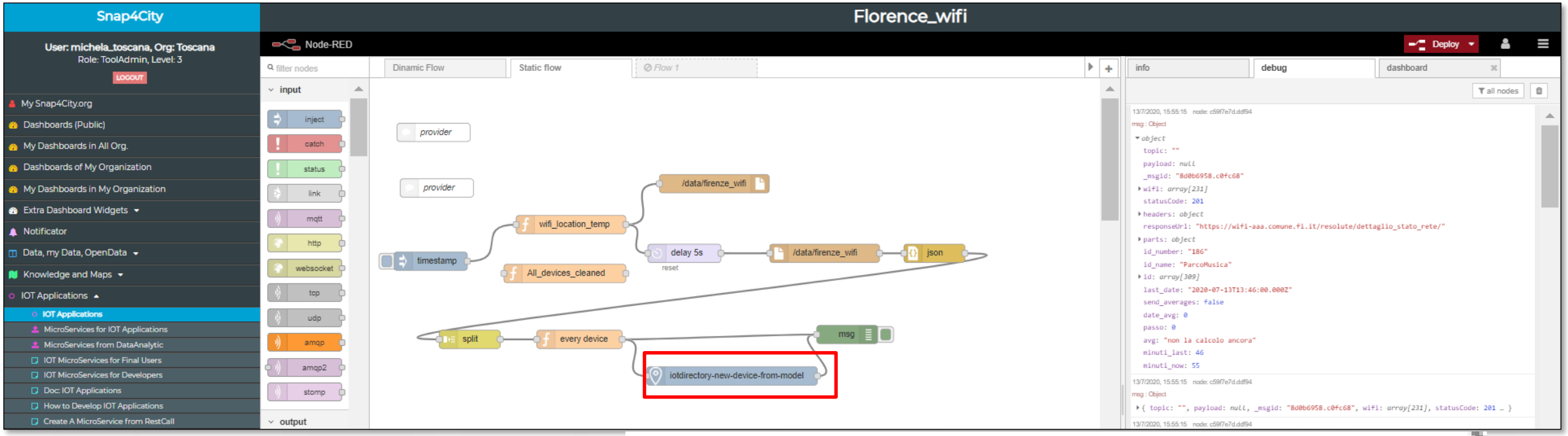
Dataset Creation

1) Model creation

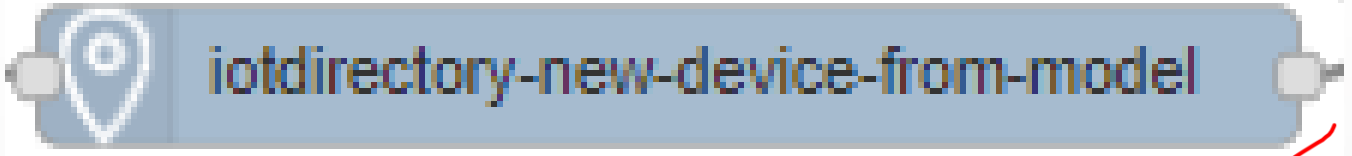


Model name: Florence wifi average person

IoTApp static flow

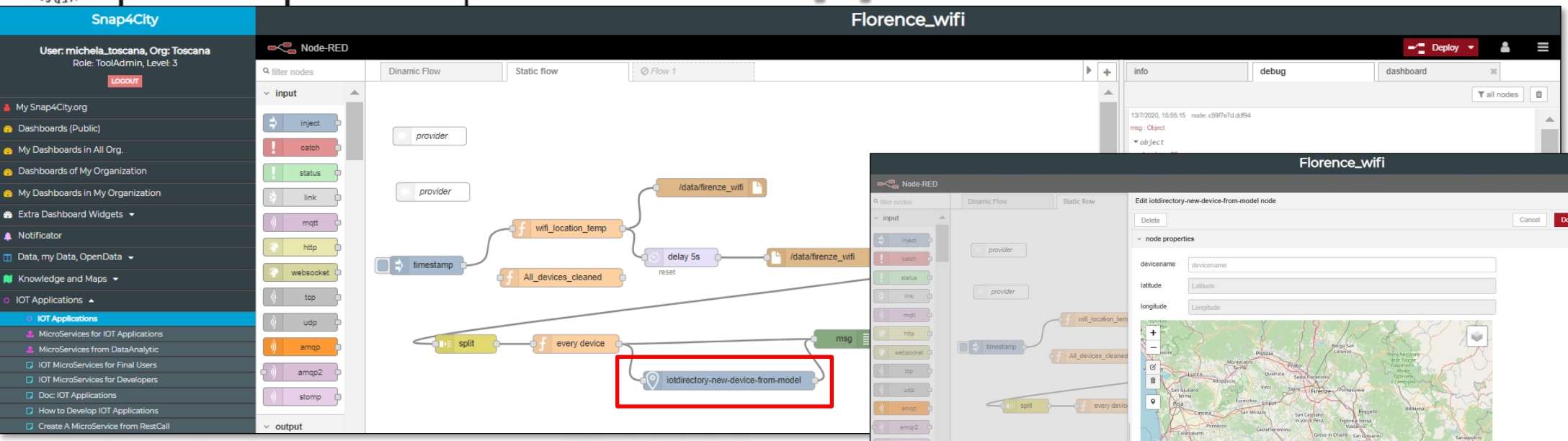


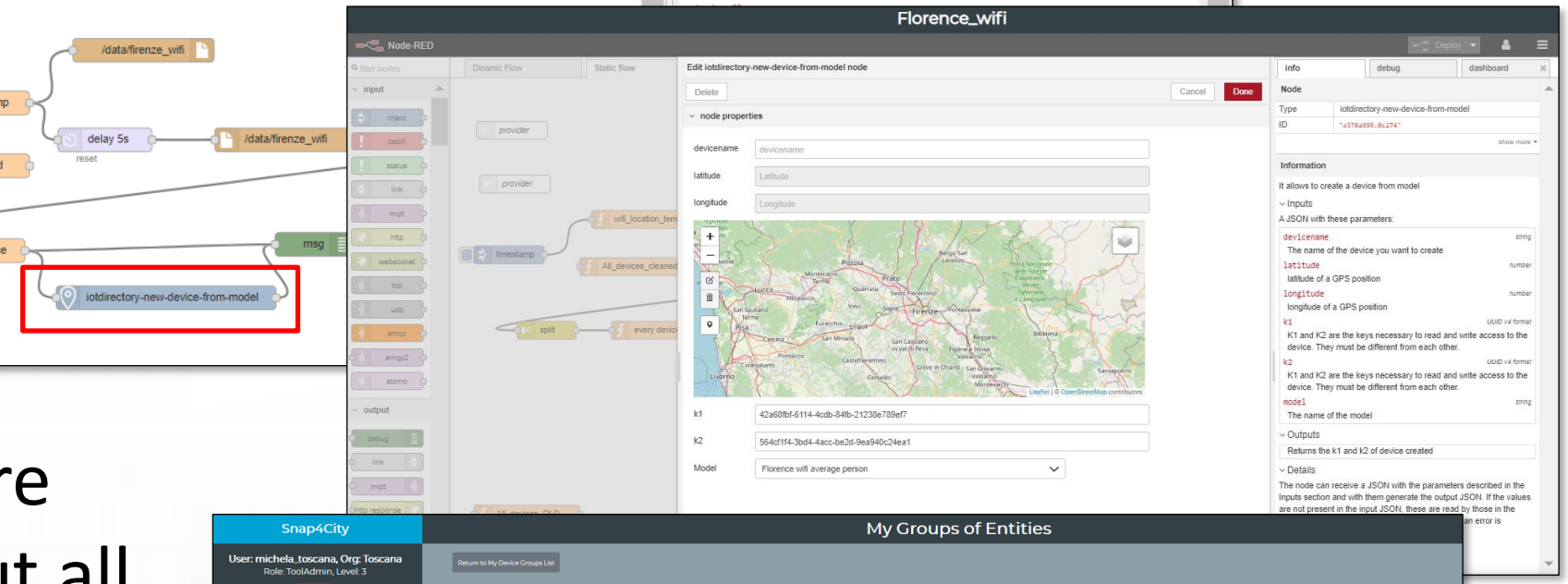
2) IoT Devices Creation from IOT APP

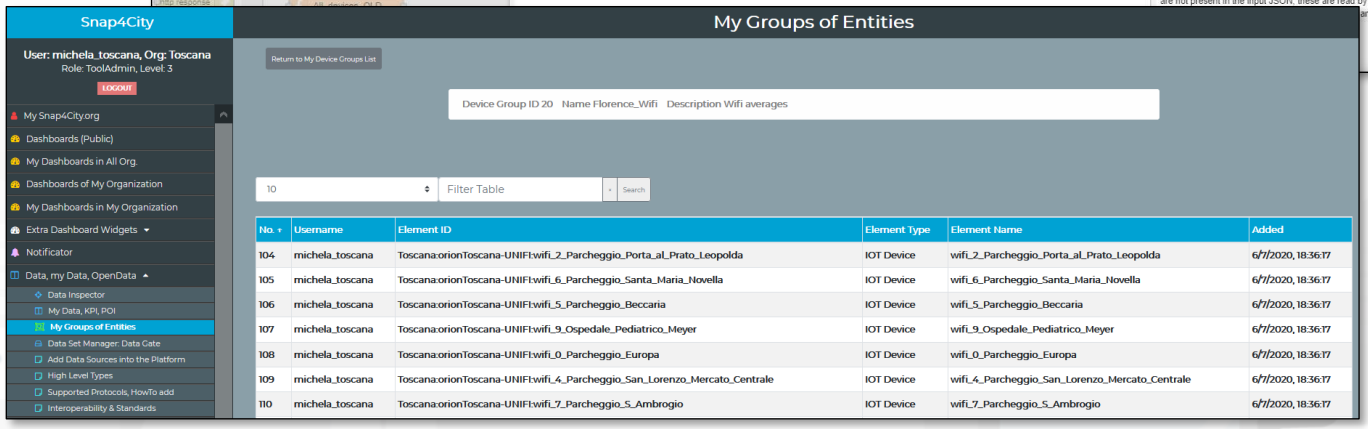


BLOCK: ‘IoTDirectory-new-device-from-model’
Model name: Florence wifi average person

IoTApp static flow



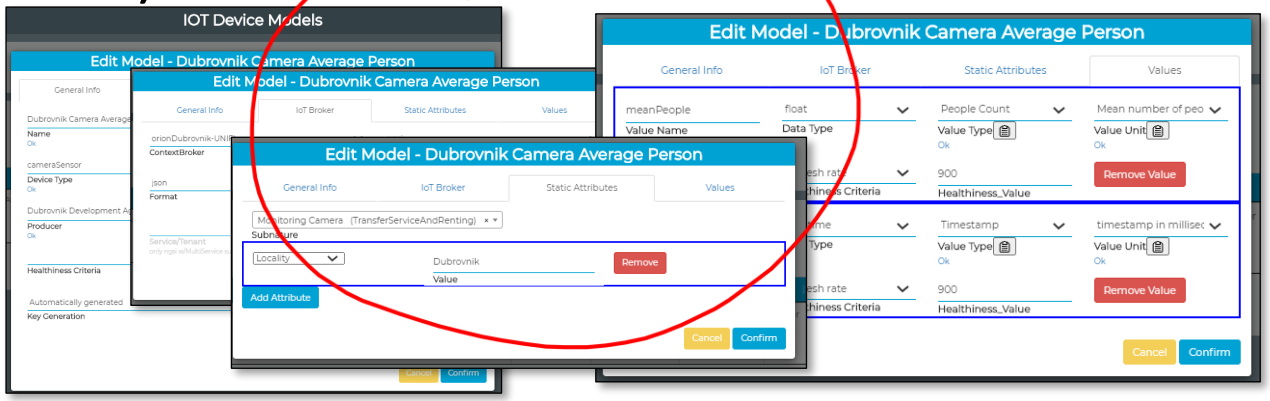




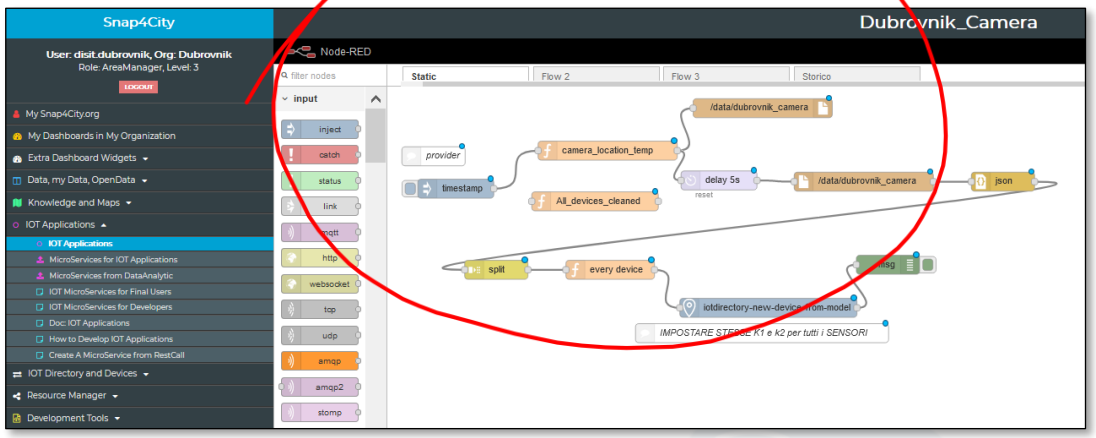
3) Group Creation (more than 200 devices) -> put all the devices in the group and put them as ‘public’ (or they remain private)

How to....

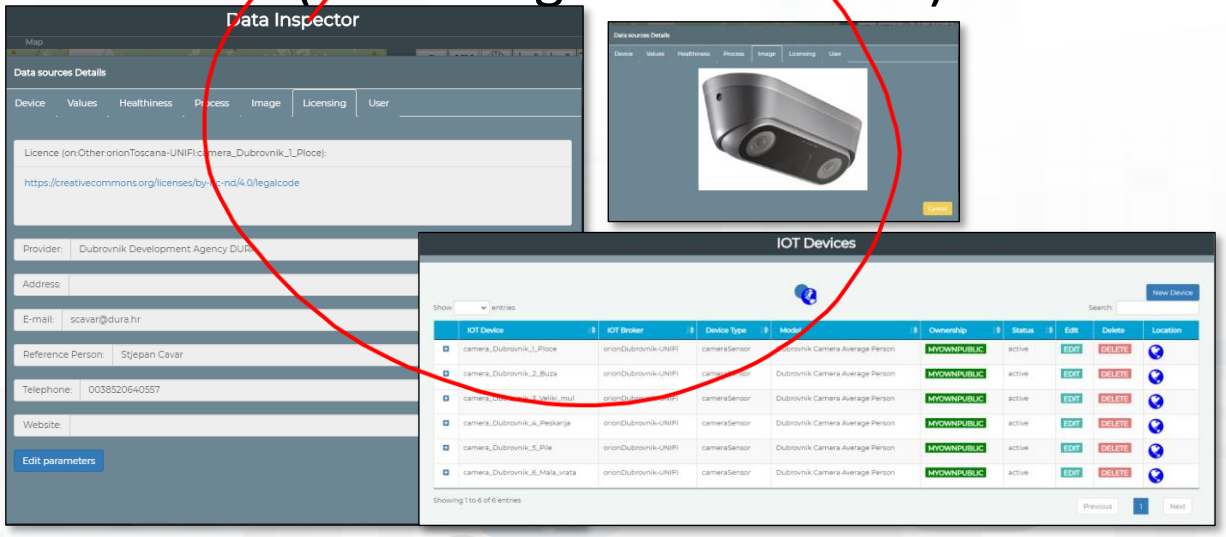
1) IoTModel



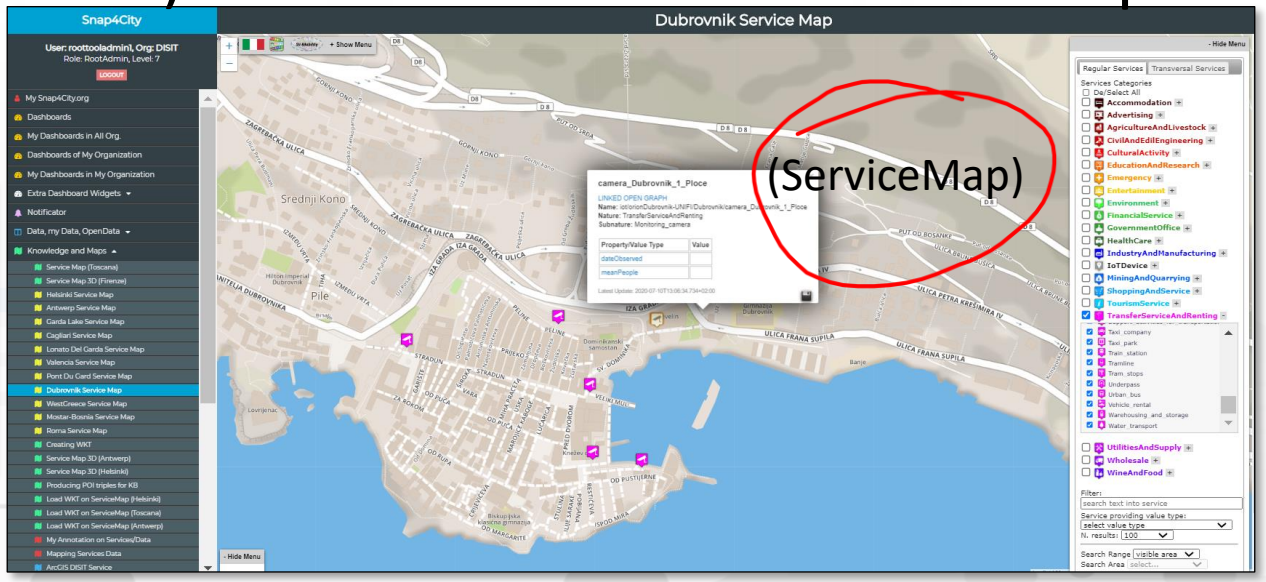
2) Static Flow to create IoTDevices



3) Add the license and Make Public the IoTDevices (according to the license)

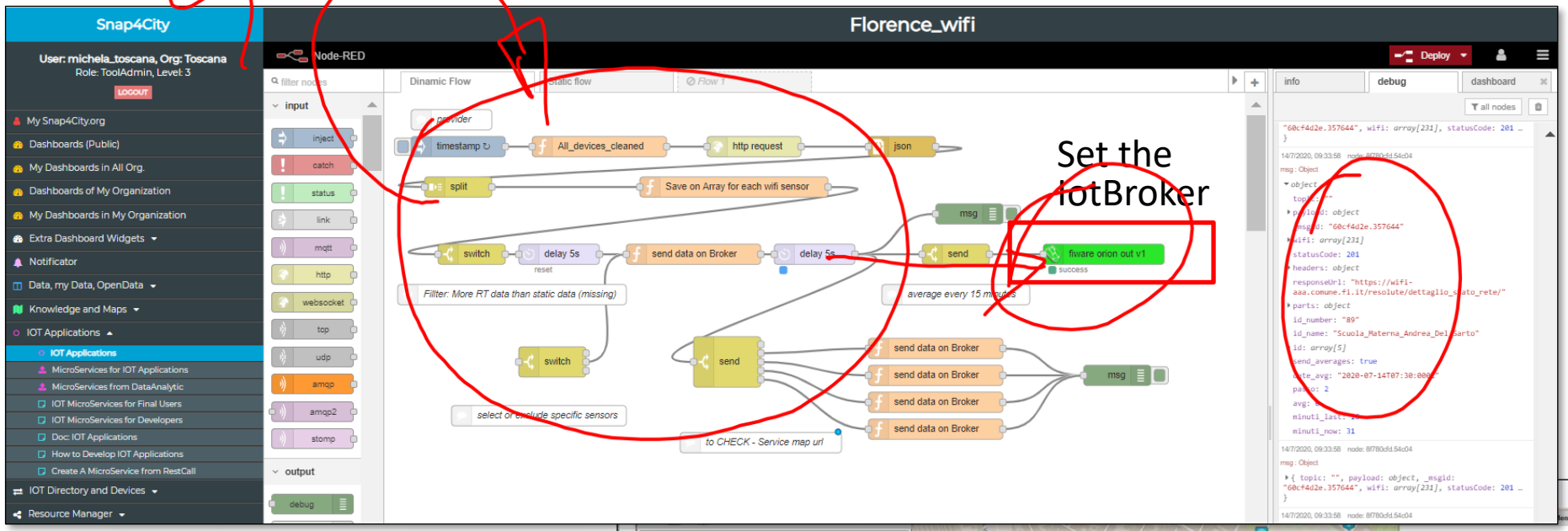


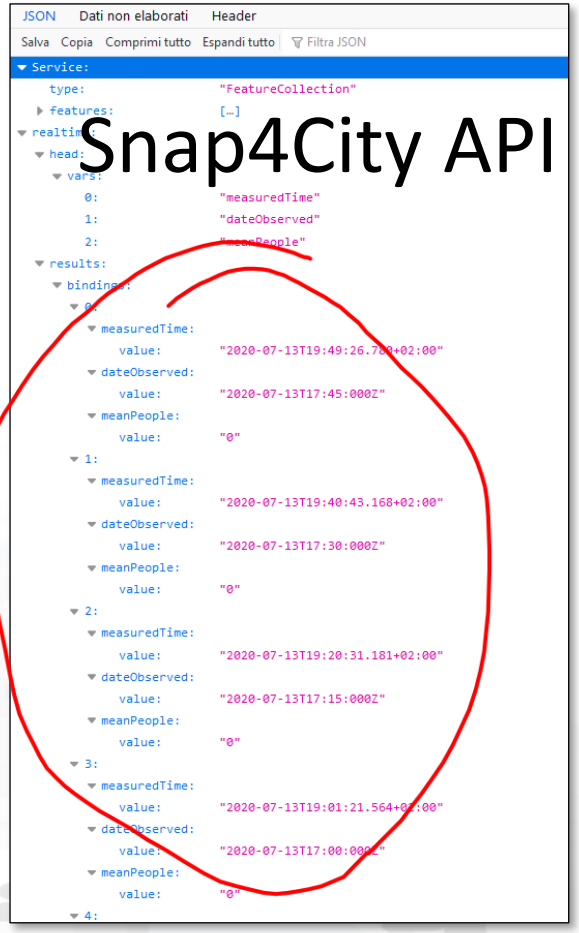
4) Search for the Cameras on Map

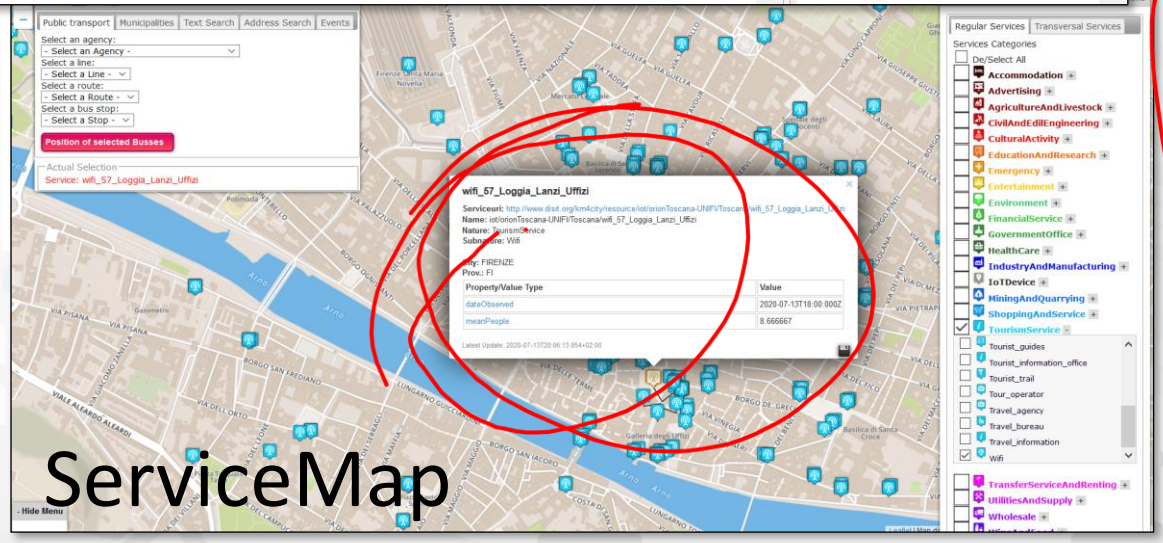


5) **Working on** Dynamic Flow to save Average #people every 15 minutes for each IoTDevice

IoTApp Dynamic Flow

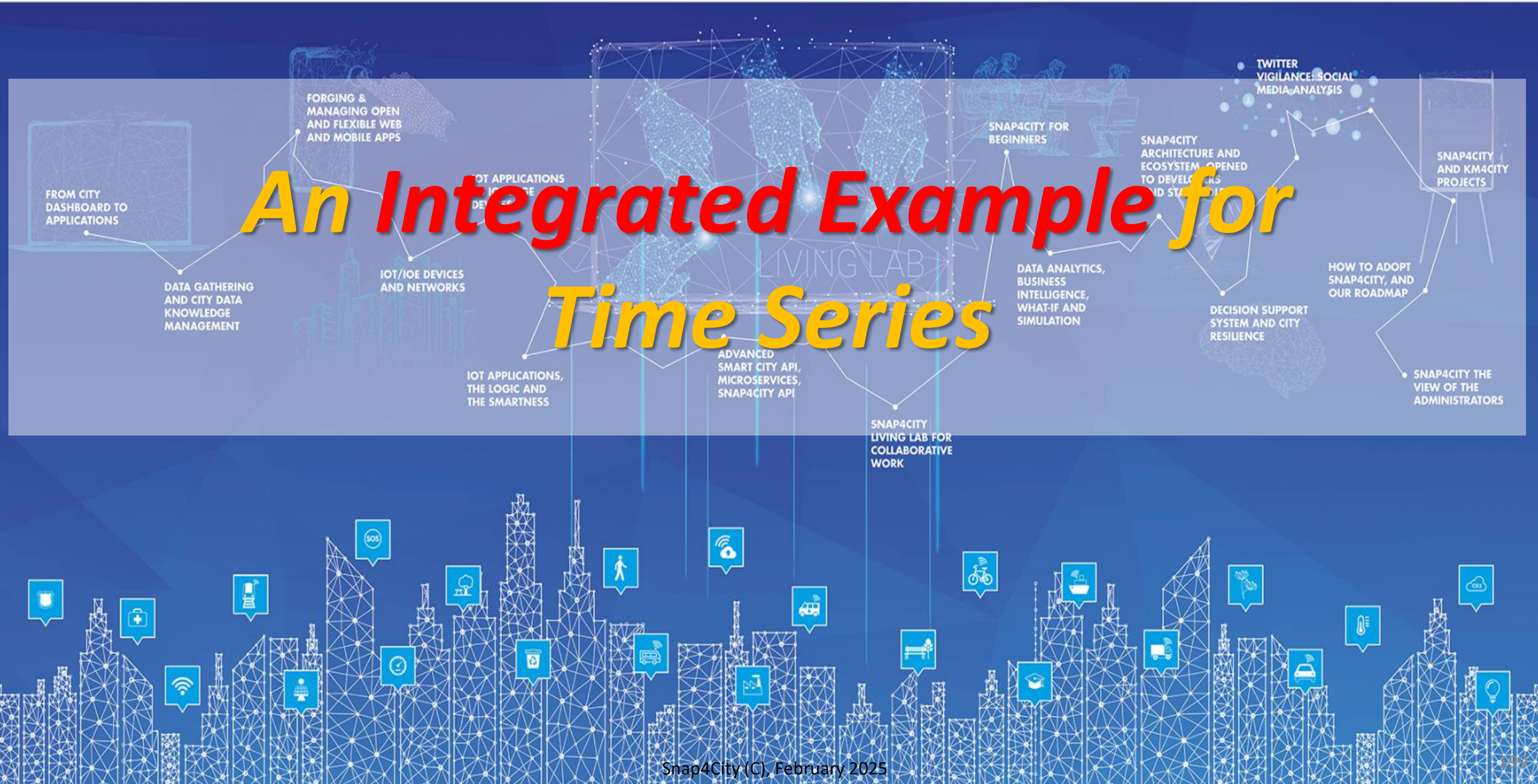






5) Send RT data to the IoTDevices

6) Verify RT Data via Snap4City API or via ServiceMap



An Integrated Example for Time Series

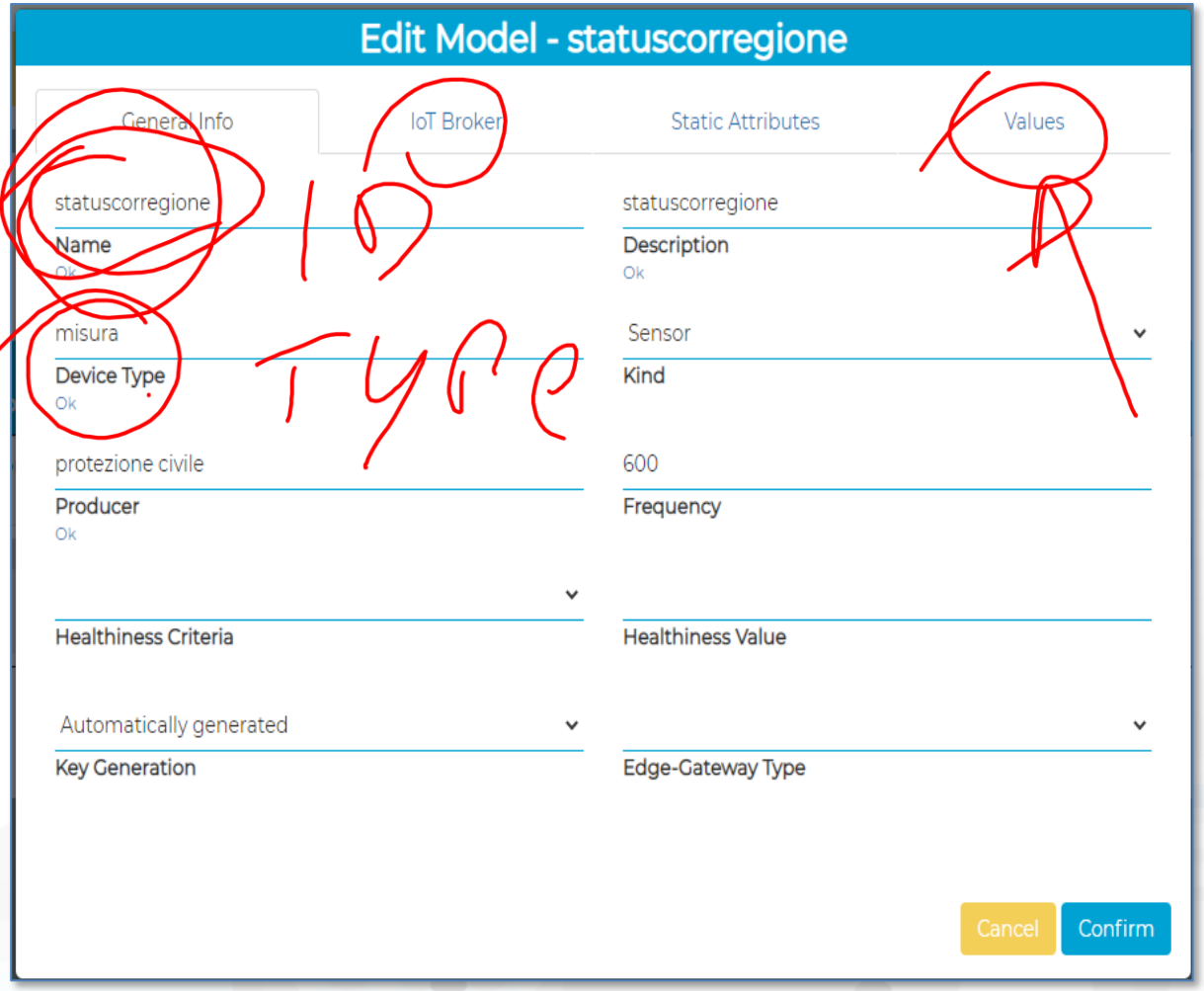
Activities for Registration on Directory

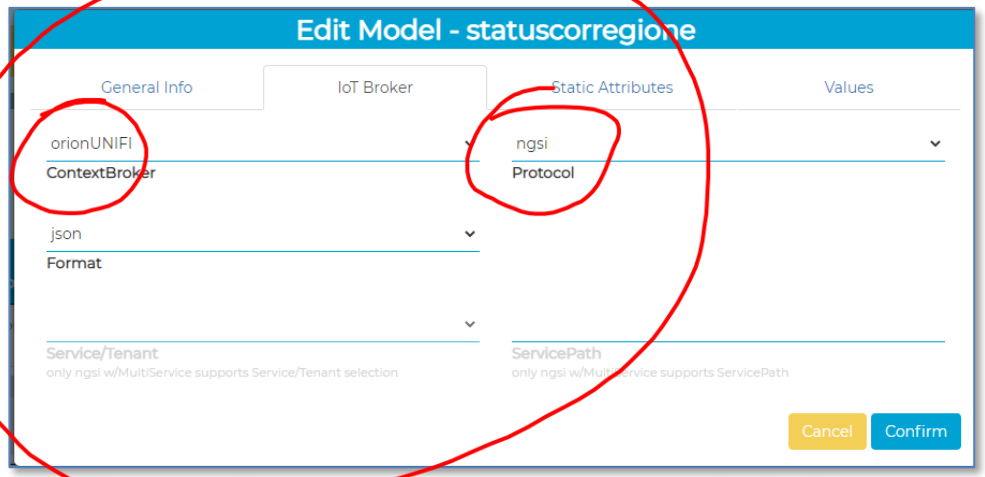
- **Manual Registration**
  - From scratch Single Device / Entity Registration → **Entity Directory / IoT Directory**
  - From a template (the templates are called Models)
- **Automated Registration for bulk/massive registration: N** Entities / Devices
  - From IoT App on the basis of some **Models** from IoT App
  - From IoT App **loading a CSV** (with or without a reference IOT Model)
  - Programming from scratch or from a **Model**
  - On the basis of some EXCEL file with data by using the **Data Table Loader, which create model, devices and data**
  - Etc.

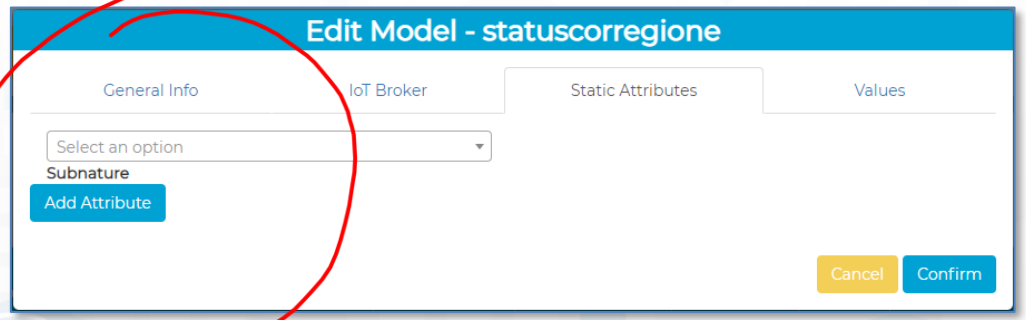


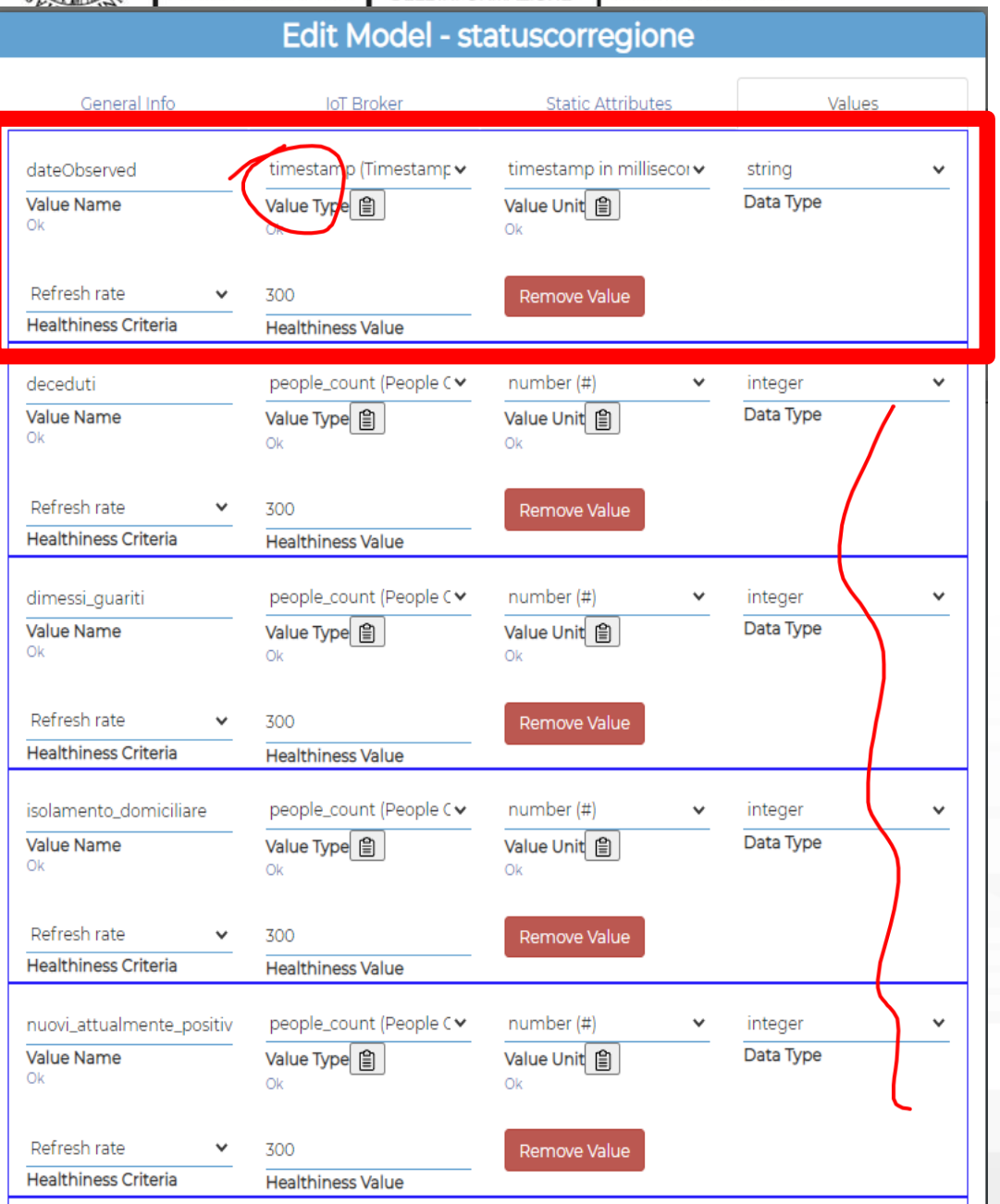
Manual or automated Registration of Entities/Devices

created a Model as:



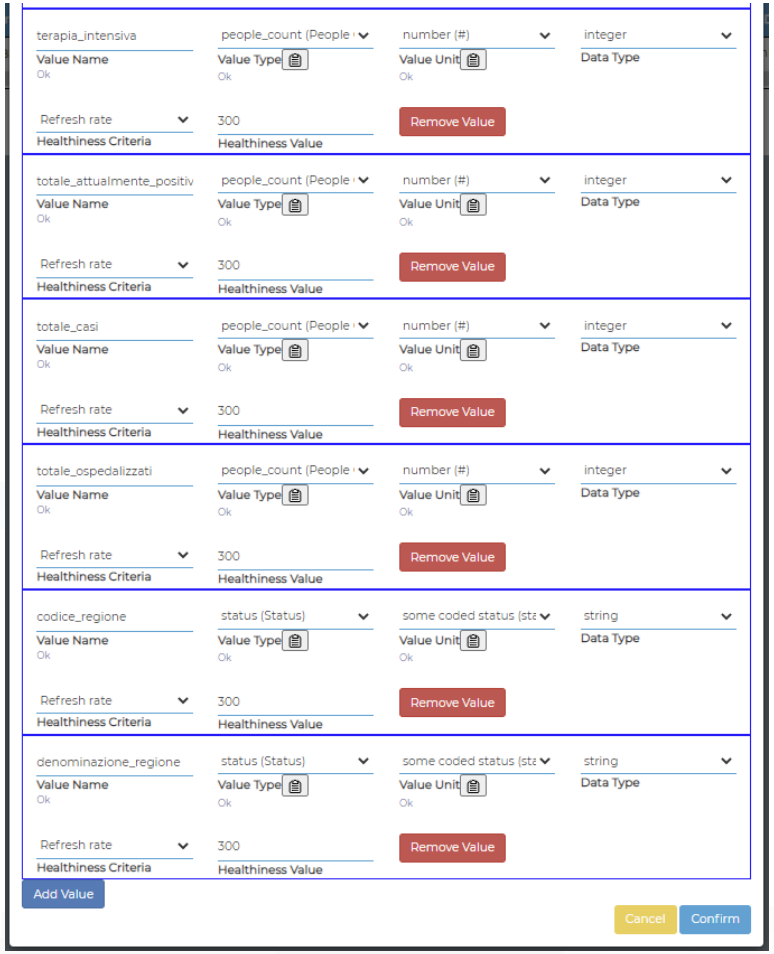






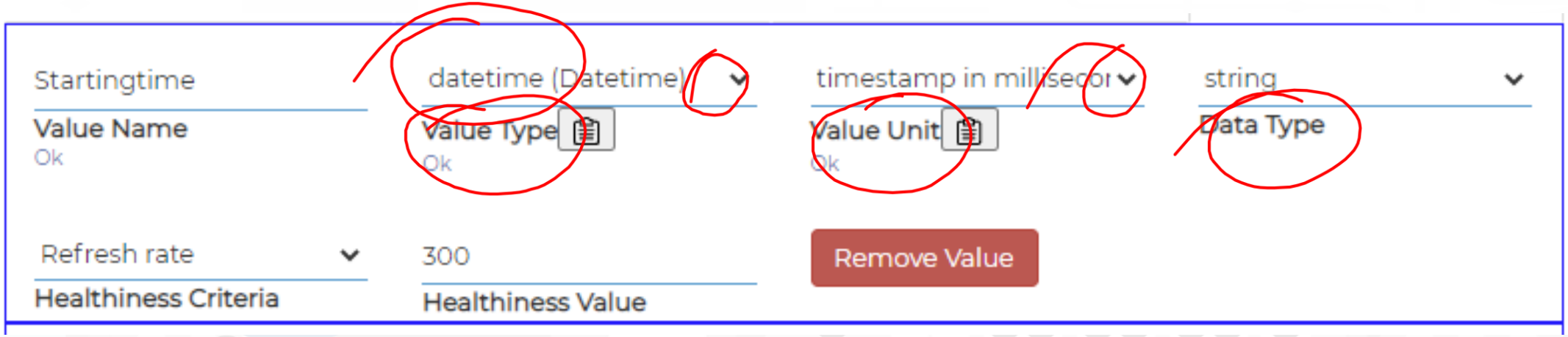
For Time Series

- ValueName: dateObserved
- ~~ValueType:~~ timestamp
- ~~ValueUnit:~~ timestamp in millisecond
- DataType: string
- E.g.: ISO string of the date-time

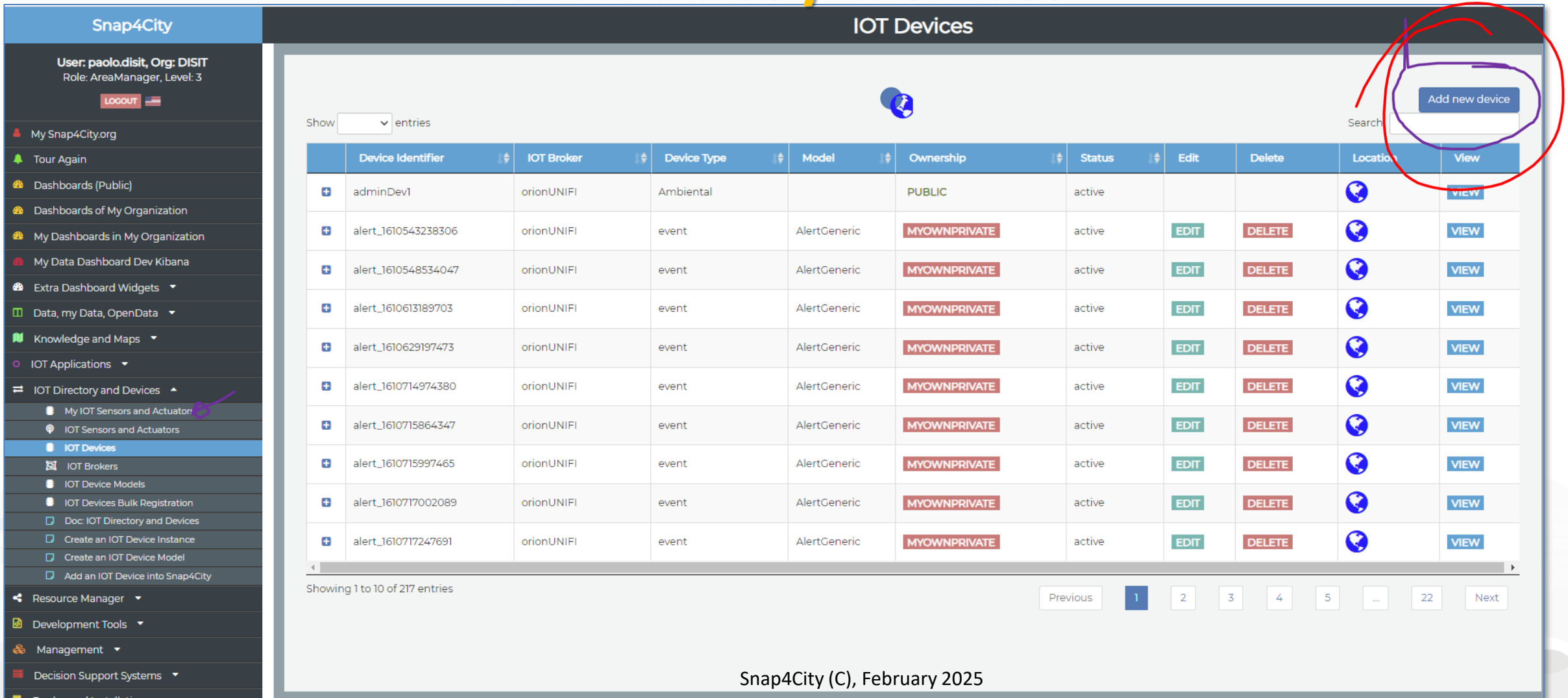


Please note for Time Series of IoT Devices

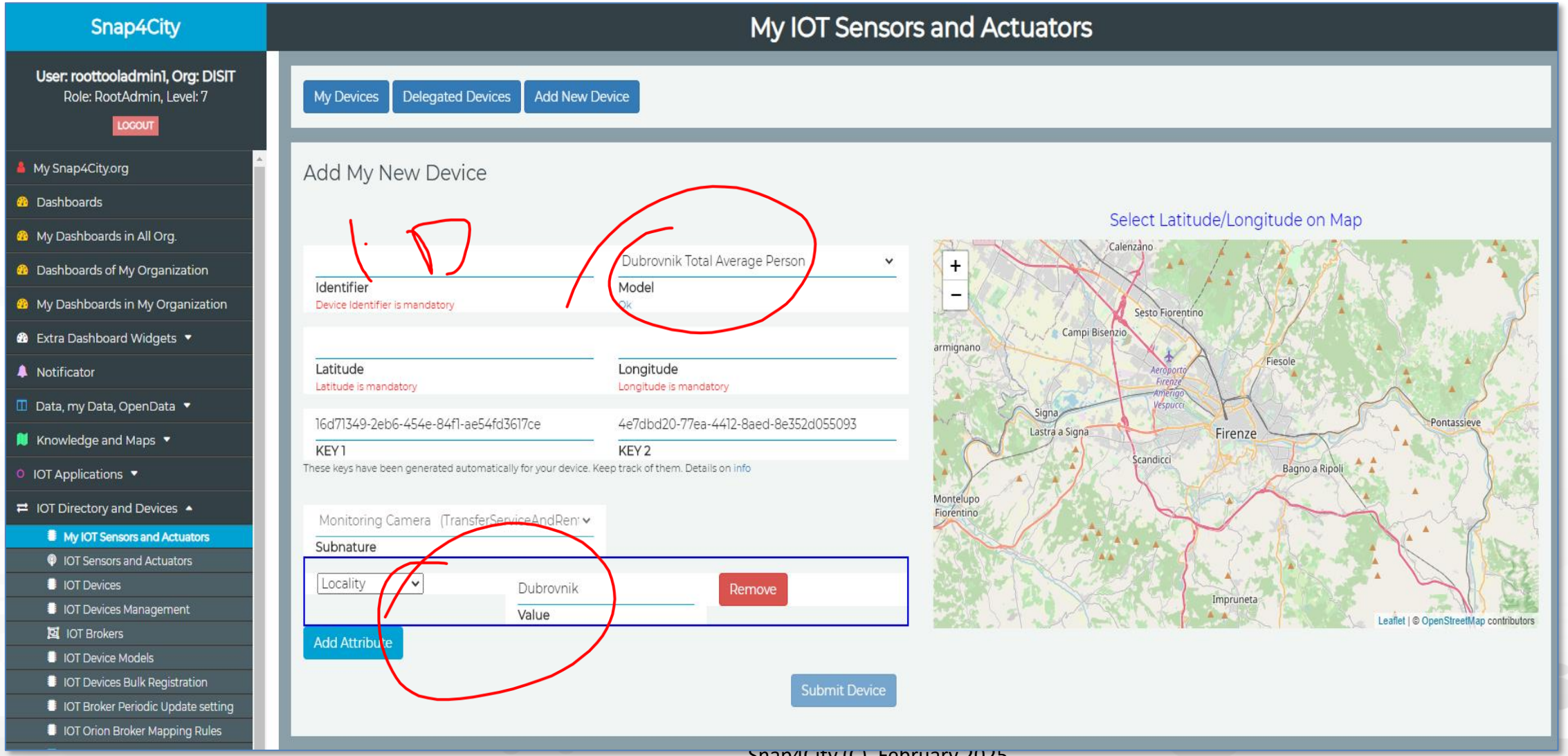
- Snap4City engine recognizes as time basis for the TimeSerie only 1 Varible with
  - ValueType as TimeStamp (in milliseconds)
- **If you need more than one timestamp** in milliseconds use:
  - ValueType as ***DateTime*** (in milliseconds)



From Model, you can create instances of Entities / Devices



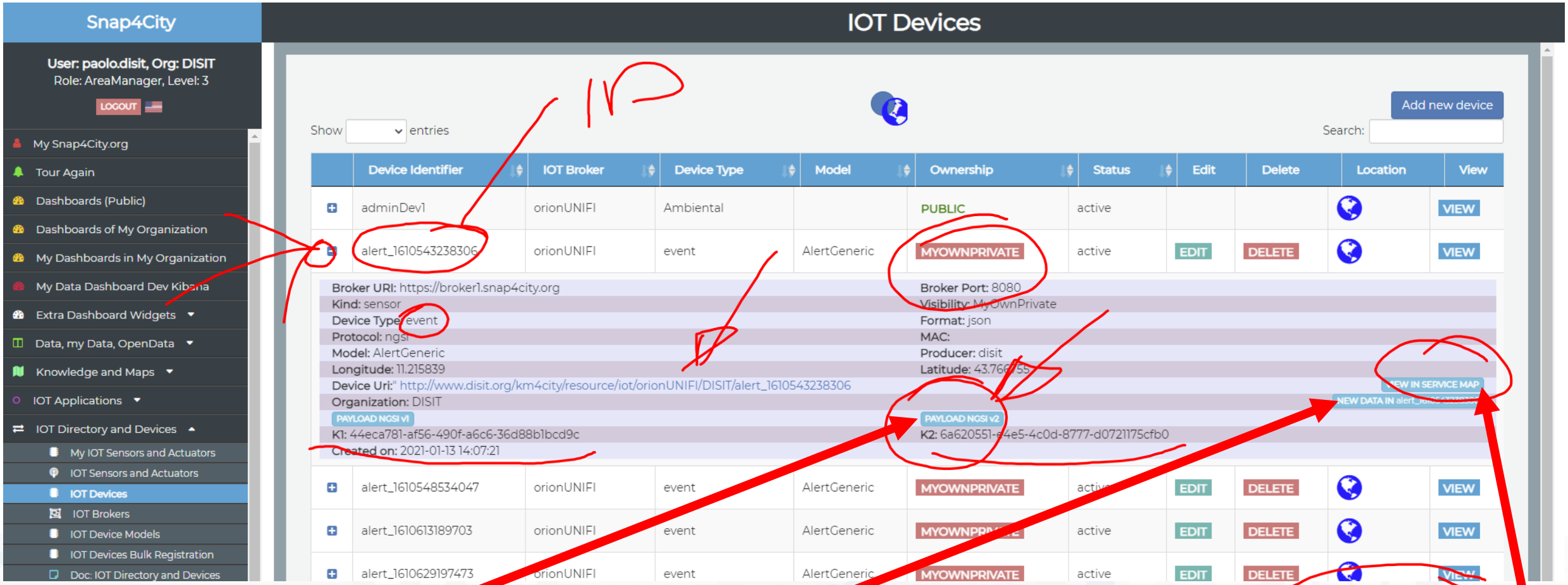
They have been created by «Add new Device»



Device from Model by Providing:

- **NAME (it has to be unique)**
- Select the IoT/Entity Model: «**statuscorregione**»
  - Thus the K1, K2 appears since the model is associated to an Orion Broker that needs to have them, the tool generate them for you but you can impose if you like
  - See in previous slide the ID name of the IOT Broker used
- **Lat** and **Lon**, GPS coordinates you can:
  - pick on the map
  - Write the coordinates manually and see the pin on map

Once Created, I may send a new data to it

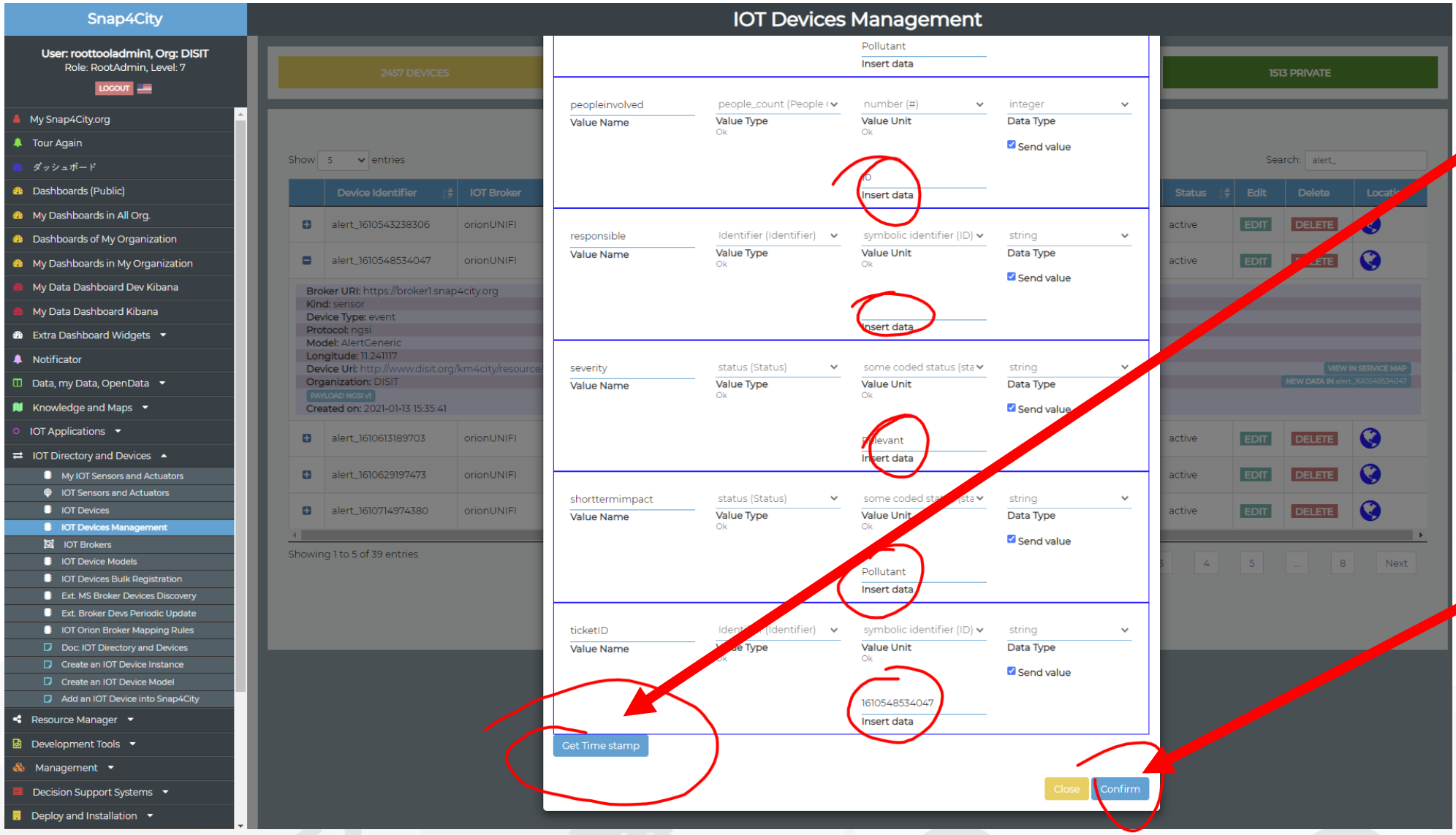


Get/See last message from Broker

Generate a New Message towards the Device, Storage

View IoT Device on map and its last value

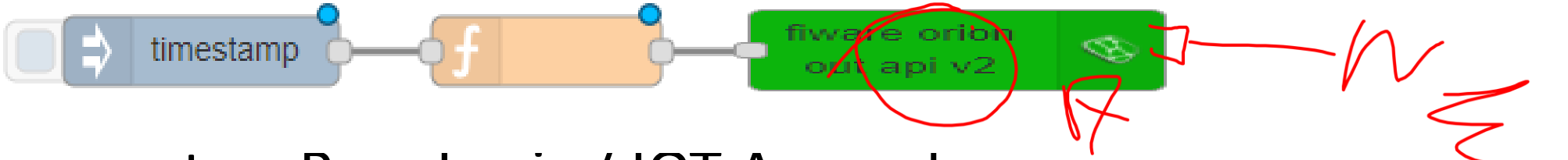
Edit Message



Impose current date time on dateObserved

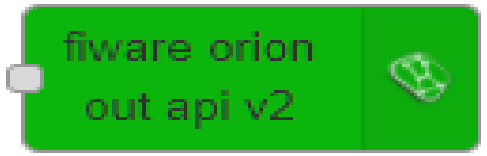
Send the Message to the Device, Storage

Once created the Device you may send data on it

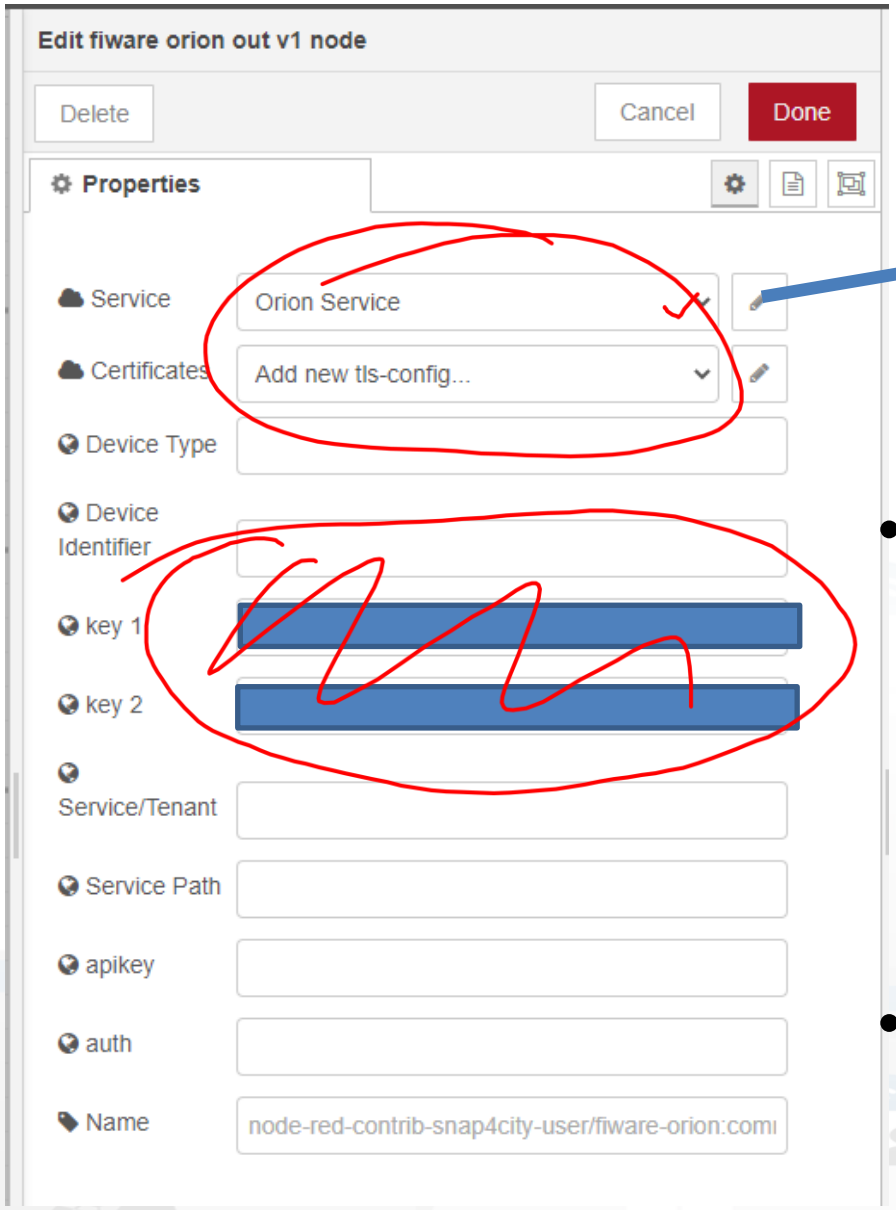


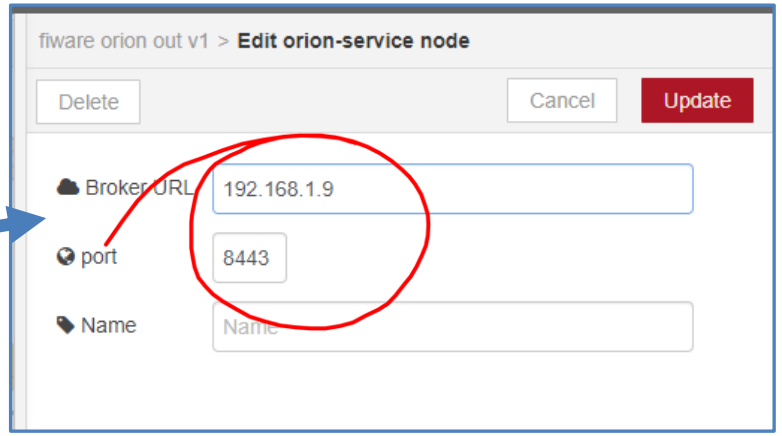
- You may create a Proc.Logic / IOT App, where:
  - Function: is preparing the JSON package
  - Block «FIWARE Orion OUT V2» is sending the data to the Orion Broker. Namely: «OrionUNIFI»
  - Please note that several version of ORION Brokers and protocols exists:
    - So that you have to know which protocols you need to use for your broker

Settings ?



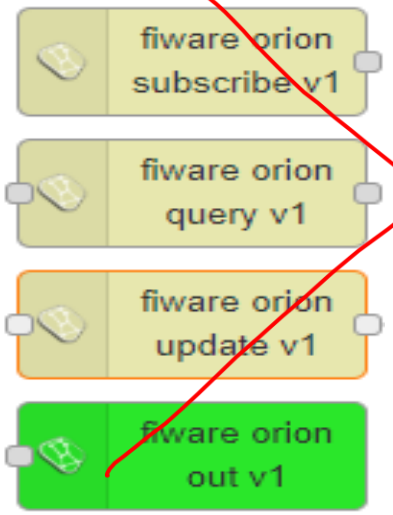
- Certificates are automatically loaded at the first authentication
- Done!!



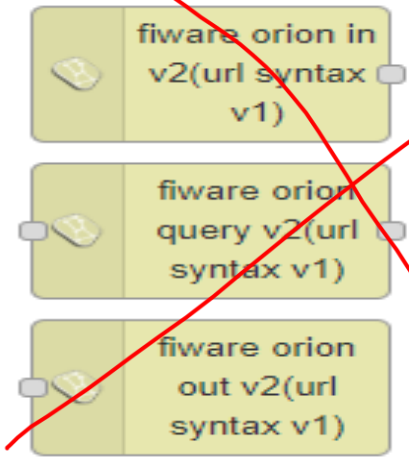


- IP if the Broker is in cloud (internal)
  - List of brokers is automatically provides
  - The K1, K2 is automatically provided if you are authenticated
- Symbolic address of Broker can be taken from Directory

NGSI versions



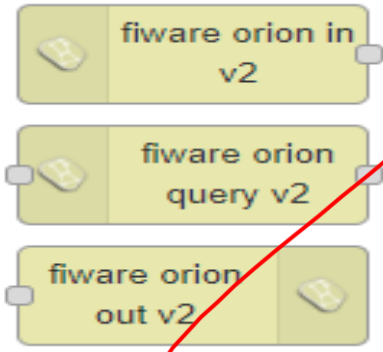
- Orion Broker of V1 with **NGSI syntax of V1** + Secure Filter of Snap4city (deprecated)



- Orion Broker of V2 with NGSI syntax of V1 (deprecated)

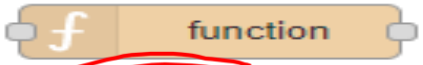
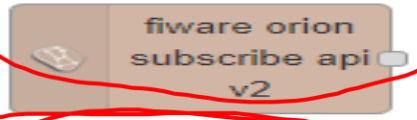
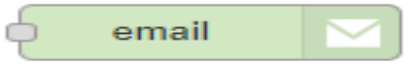
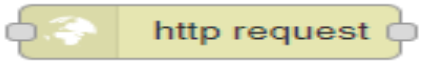
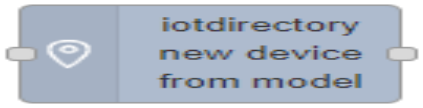


- Orion Broker of V2 with **NGSI syntax of V2** + Secure Filter of Snap4city



- Orion Broker of V2 with NGSI syntax of V2 (deprecated)

examples

	To generate injection messages into a flow, scheduled or on demand by click
	A java script function, from a JSON input to one or more JSON outputs
	To send a message of and Entity Instance into the storage. The Device has to be registered and you have to be the owner or to be delegated in READ-WRITE to send messages to it.
	To subscribe an IoT App to receive event-driven notification about messages and changes on specific devices. You can subscribe to many and then to get all of them from it.
	Query call to Smart City API to get any information about a SURI, ServiceURI. There are many other Nodes which can be used to pose Smart City API queries in very simple manner
	Send email, you can even send Telegram, SMS, etc.
	To send a REST CALL (get, post, etc.). Please USE THIS ONLY for the external API not for the Snap4City API for which a lot of MicroService accessible as NODEs/Blocks in the IoT App are simpler and ready to use.
	A block which is printing on debug view the data JSON passed in input.
	To create a device instance from a model prepared on IoT Directory
	To change the ownership of a device.
	To delegate a certain device to some other user
	To show something on Snap4City dashboard with a simple widget.

- A Json from the IOT App
  - NGSI V1
- **ID**: The Name of the IOT Device: «corveneto»
- **Type** as that defince in the IOT Device when you created
- The **Time stamp**: "dateObserved" to have a time series data
  - "str" is a string with the date and time in standard ISO, such as ,
  - "2020-08-04T04:00:00+02:00",
  - "2020-08-03T00:00:00.000Z"
- And the **vector** of "**attributes**"

```
msg = { payload : {
  "id": "corveneto",
  "type": "misura",
  "attributes":[
 { "name": "dateObserved", "value": str, "type": "timestamp" },
 { "name": "stato",  "value": "active",  "type": "string" },
 { "name": "ricoverati_con_sintomi",  "value": 12,  "type": "integer" },
 { "name": "terapia_intensiva",  "value": 34,  "type": "integer" },
 { "name": "totale_ospedalizzati", "value": 34,  "type": "integer" },
 { "name": "isolamento_domiciliare",  "value": 334,  "type": "integer" },
 { "name": "totale_attualmente_positivi",  "value": 12,  "type": "integer" },
 { "name": "nuovi_attualmente_positivi", "value": 33,  "type": "integer" },
 { "name": "dimessi_guariti",  "value": 22222,  "type": "integer" },
 { "name": "deceduti",  "value": 2,  "type": "integer" },
 { "name": "totale_casi",  "value": 2222,  "type": "integer" },
 { "name": "tamponi",  "value": 222222344,  "type": "integer" }
 ]
}
}
return msg;
```

NGSI V2

{"id":"corveneto",
"type":"misura",
"codice_regione":{"type":"integer","value":""},
"dateObserved":{"type":"time","value":"2021-01-18T17:00:00.000Z"},
"deceduti":{"type":"integer","value":"8025"},
"denominazione_regione":{"type":"string","value":""},
"dimessi_guariti":{"type":"integer","value":"222062"},
"isolamento_domiciliare":{"type":"integer","value":"66514"},
"latitude":{"type":"float","value":"45.43490"},
"longitude":{"type":"float","value":"12.33845"},
"nuovi_attualmente_positivi":{"type":"integer","value":"-1557"},
"ricoverati_con_sintomi":{"type":"integer","value":"2233"},
"stato":{"type":"string","value":"ITA"},
"tamponi":{"type":"integer","value":"3663538"},
"terapia_intensiva":{"type":"integer","value":"336"},
"totale_attualmente_positivi":{"type":"integer","value":"69083"},
"totale_casi":{"type":"integer","value":"299170"},
"totale_ospedalizzati":{"type":"integer","value":"2569"}
}

NGSI V2

The differences are mainly on how the variable are provided:

{
"id":"MyMobileDeviceTest",
"type":"misura",
"dateObserved":{"type":"timestamp","value":"2021-06-11T16:17:23.425Z"},
"status":{"type":"float","value":25}
"mydescription":{"type":"string","value":"**see below the note for the forbidden characters**"}
}

- **NOTE for:**
  - **names/IDs**: Spaces or strange characters are not allowed in the. Please use simple alfphanumeric strings, it is a limitation of many solutions including Orion Broker and increase interoperability of your data.
  - **Values of attributes and variables**: can be UTF8, but similarly, they do not accept: () <> “ ‘ ; = into values
  - https://fiware-orion.readthedocs.io/en/master/user/forbidden_characters/index.html

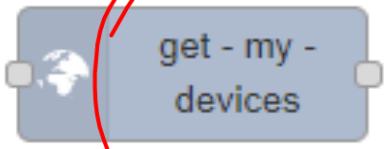
DateObserved

- The **Timestamp**: “dateObserved” to have a time series data
  - “str” is a string with the date and time in standard ISO, such as ,
  - “2020-08-04T04:00:00+02:00”,
  - “2020-08-03T00:00:00.000Z”
- In JavaScript you can obtain by using:
  - Var str = new Date().toISOString();
  - **Str** has to be the ISO date string of today-now (at the current time).

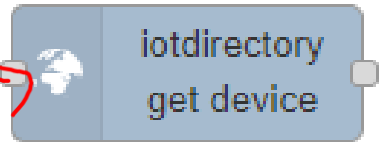
From date to ISOString with fuse aligned time

```
var todaynow = new Date();
dateCET2Z(todaynow).toISOString();
---
function dateCET2Z(date) {
    d = new Date(date).toLocaleString('nl-BE', {timeZone: 'Europe/Brussels'});
    offset = new Date(d).getTime() - new Date(date).getTime();
    return new Date(new Date(date).getTime() - offset);
}
```

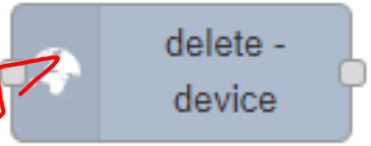
You may use other functions from Directory



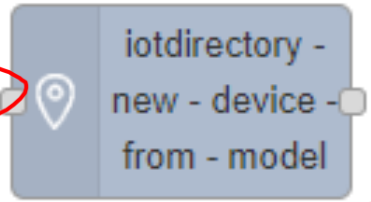
- Discovery in an area
- Query on Directory



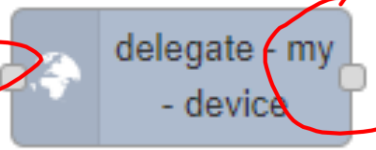
- Get Device Info



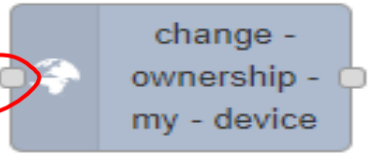
- Delete a Device (max rate 1 per minute)



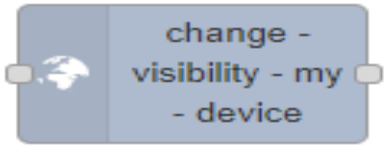
- Registering an Device from model



- Delegate an Device



- Change Ownerhip of an Device



- Change Visibili, Publish

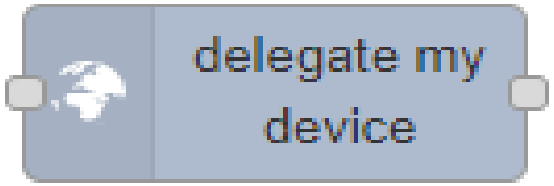
Get IOT Device Info



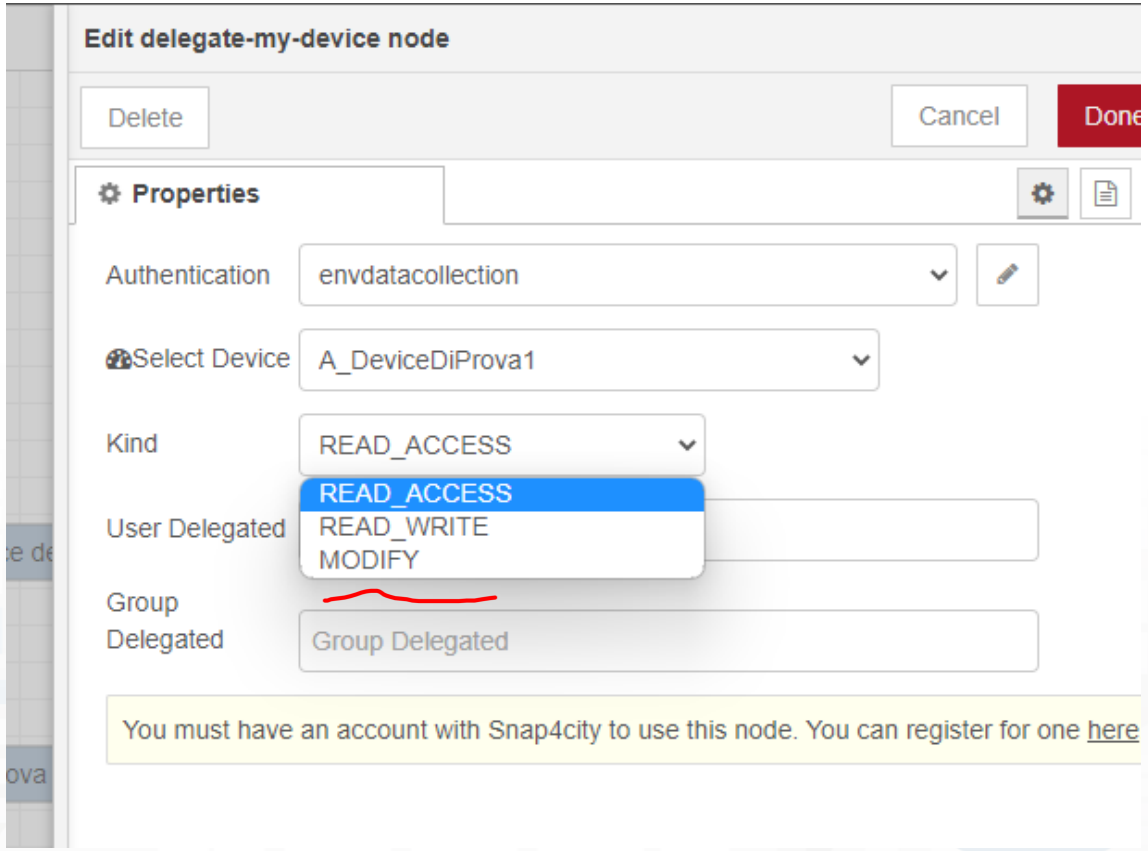
- You can create smart Proc.Logic / IoT App that on the basis of the list of Devices would request all what you need to load data into **YOUR OWN** Devices including:
  - **Service URI**
  - **K1, K2**
  - Authentication

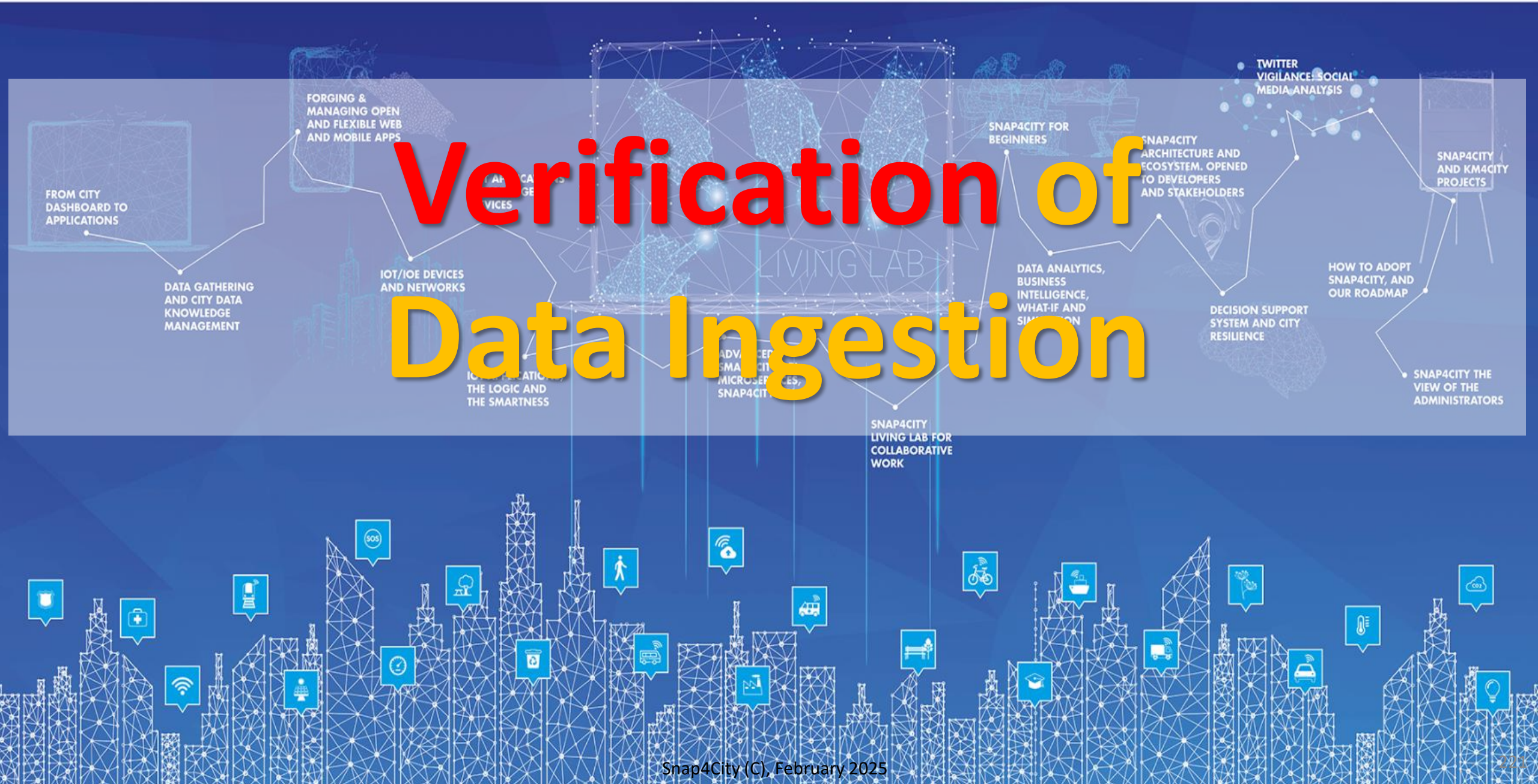
msg.payload : Object
object
status: "ok"
content: object
uri: "http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/corarezzo"
devicetype: "misura"
kind: "sensor"
status1: "active"
macaddress: ""
model: "statuscorregione"
producer: "protezione civile"
longitude: "11.88228"
latitude: "43.46642"
protocol: "ngsi"
format: "json"
visibility: "public"
frequency: "600"
created: "2020-03-21 18:34:32"
privatekey: ""
certificate: ""
organization: "DISIT"
accesslink: "https://broker1.snap4city.org"
accessport: "8080"
sha: "C61E32DBFAE7F14C0810177F2D2300843C41C550"
subnature: null
static_attributes: null
k1: "bf739214-f6b4-45fe-85f7-97cd09fe8e57"
k2: "c19e0b6f-8f98-4130-b135-e7a1dfae9273"
error_msg: ""

Delegate Management from IoT App



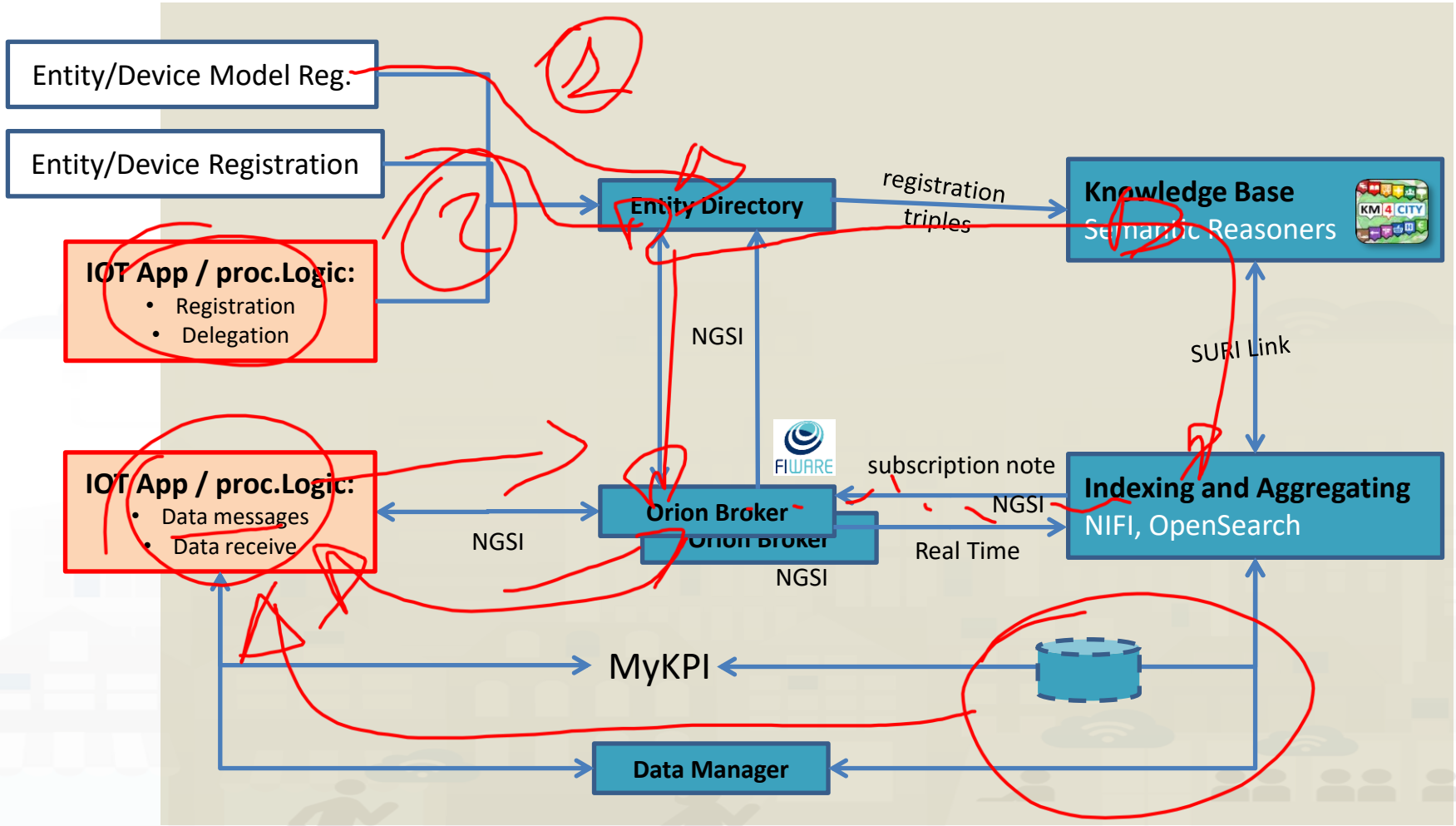
To delegate a certain device to some other user/group of users





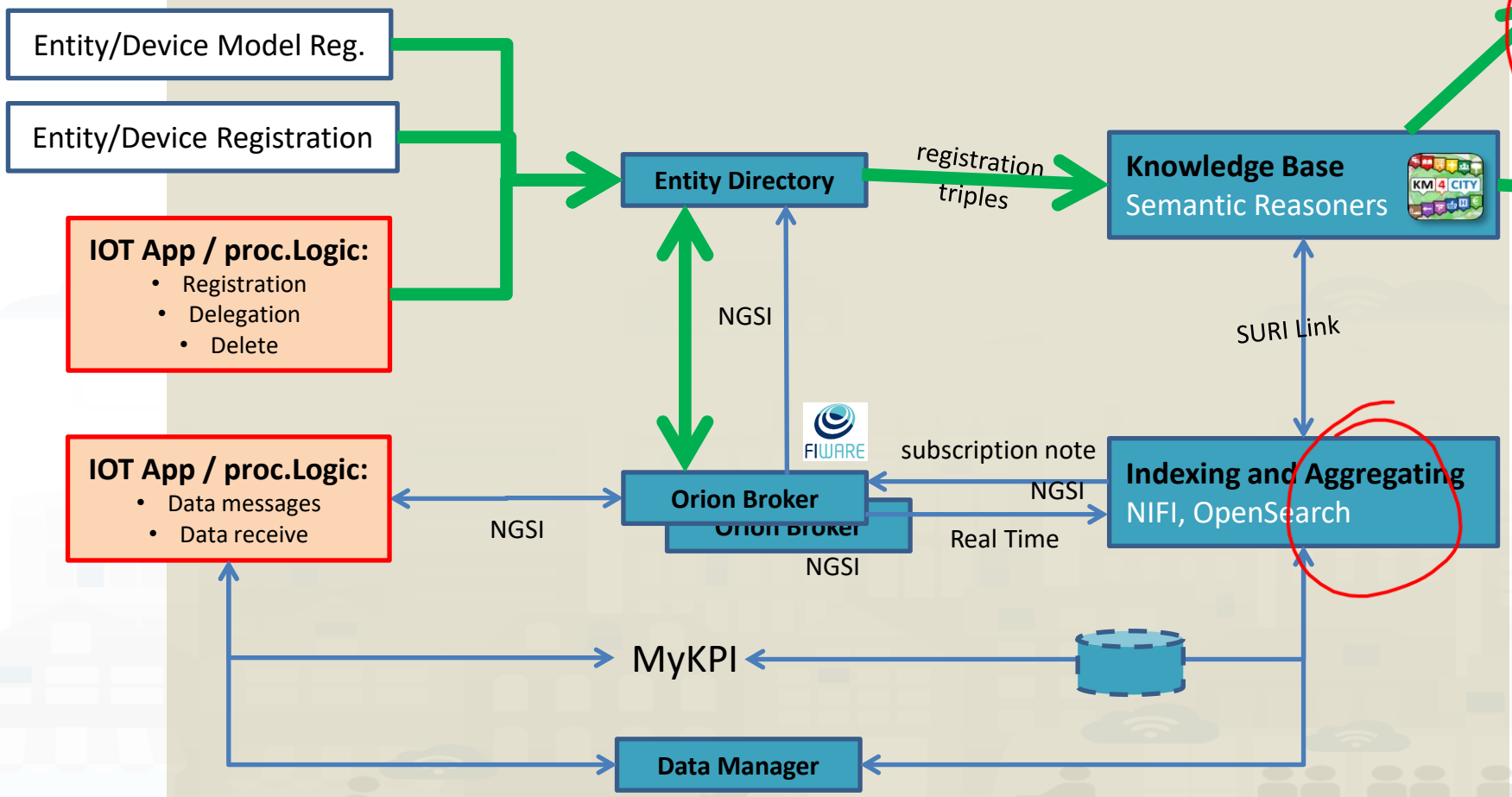
Verification of Data Ingestion

Main Data In/Out flows



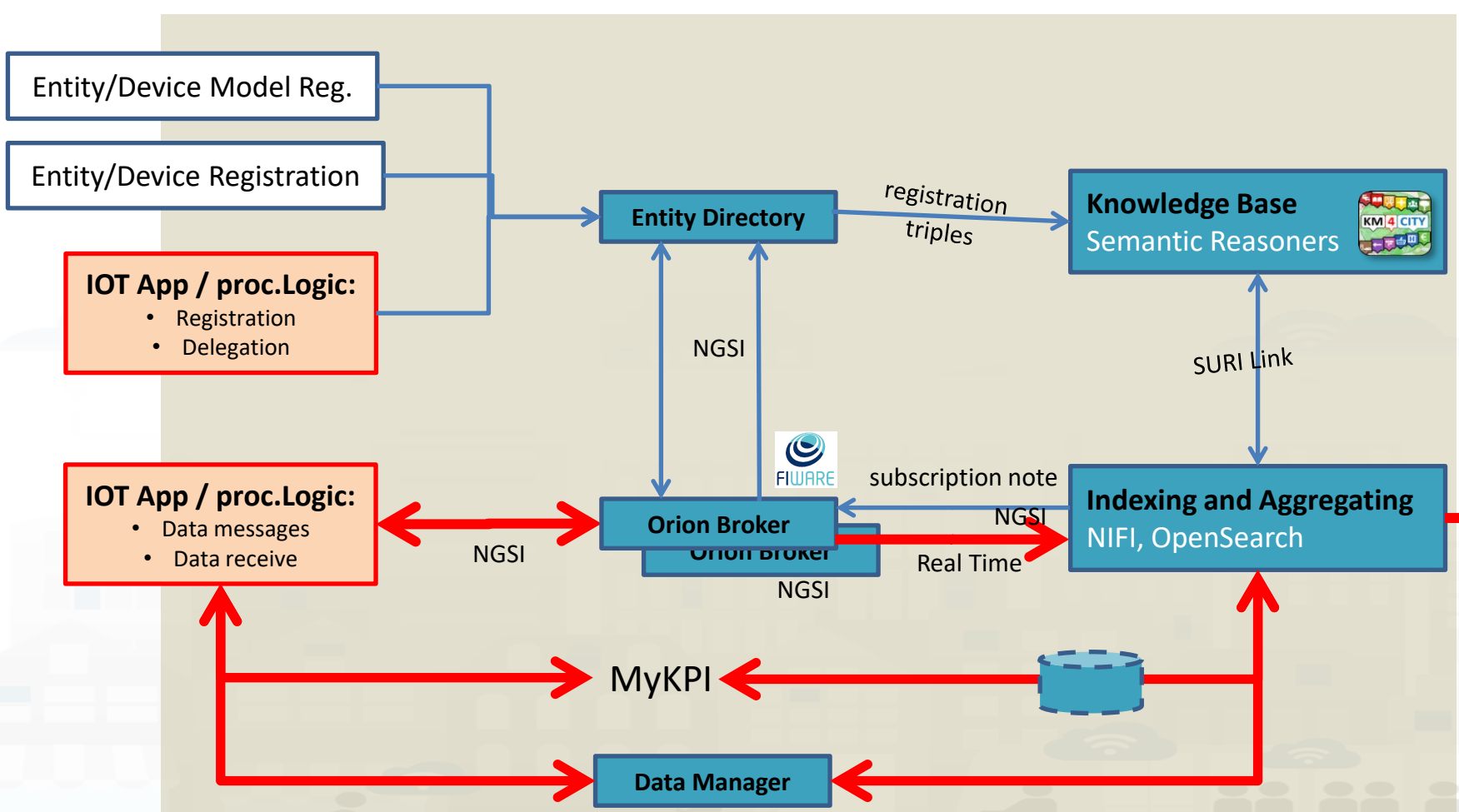


Main Data In/Out flows





Main Data In/Out flows



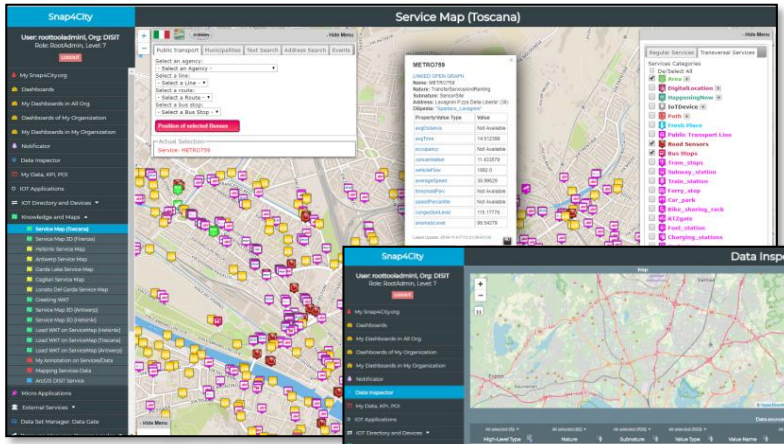


Checking data/Entity ingestion results

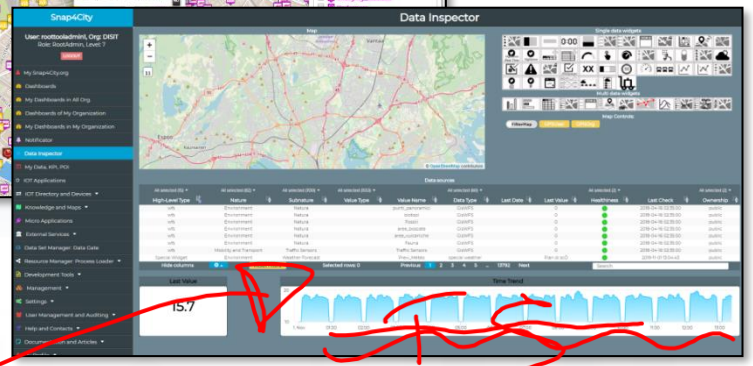
Knowledge base
Semantic reasoners

- All searches
- Metata
- Structure
- Last values of IoT Dev
- GTFS
- Only public IoT Dev

- **ServiceMap**, SCAPI, **SuperSM**
  - LOG / LOD viewer
  - Super Service Map
  - SCAPI: Swagger
  - Last data
- **Data Inspector (last data)**
- IoT/Entity Directory
  - IoT Brokers



ServiceMap or
Super ServiceMap

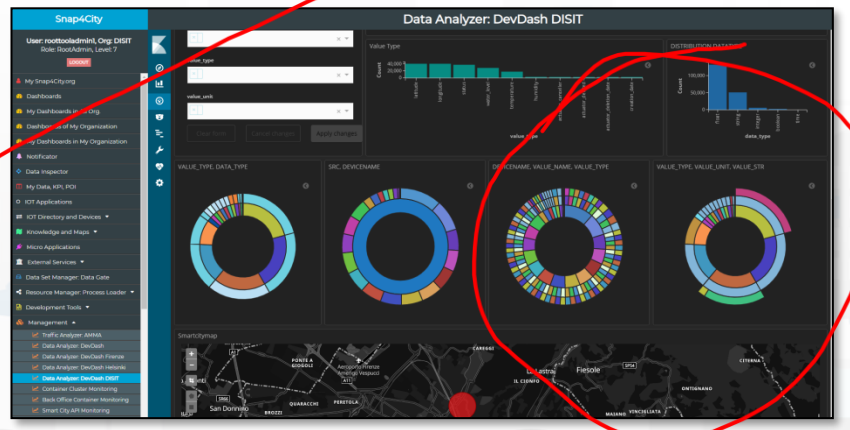


Data Inspector
Digital Twin view

Indexing and aggregating
N'FI, OpenSearch

- Faceted search
- Geo search
- Time Series
- Private and Public

- **ServiceMap**, SCAPI (last data), **SuperServiceMap**
- **My Data Dashboard,** OpenSearchDash
- **Data Inspector (last data)**



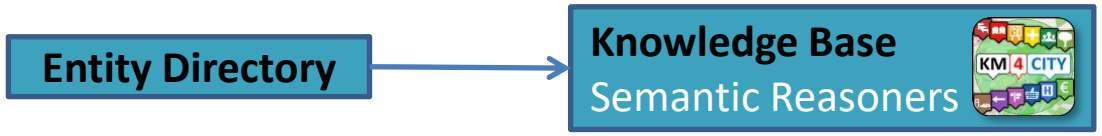
My Data Dashboard

DevDash

Some functionalities are limited to certain roles

Verification of Data Ingestion Process

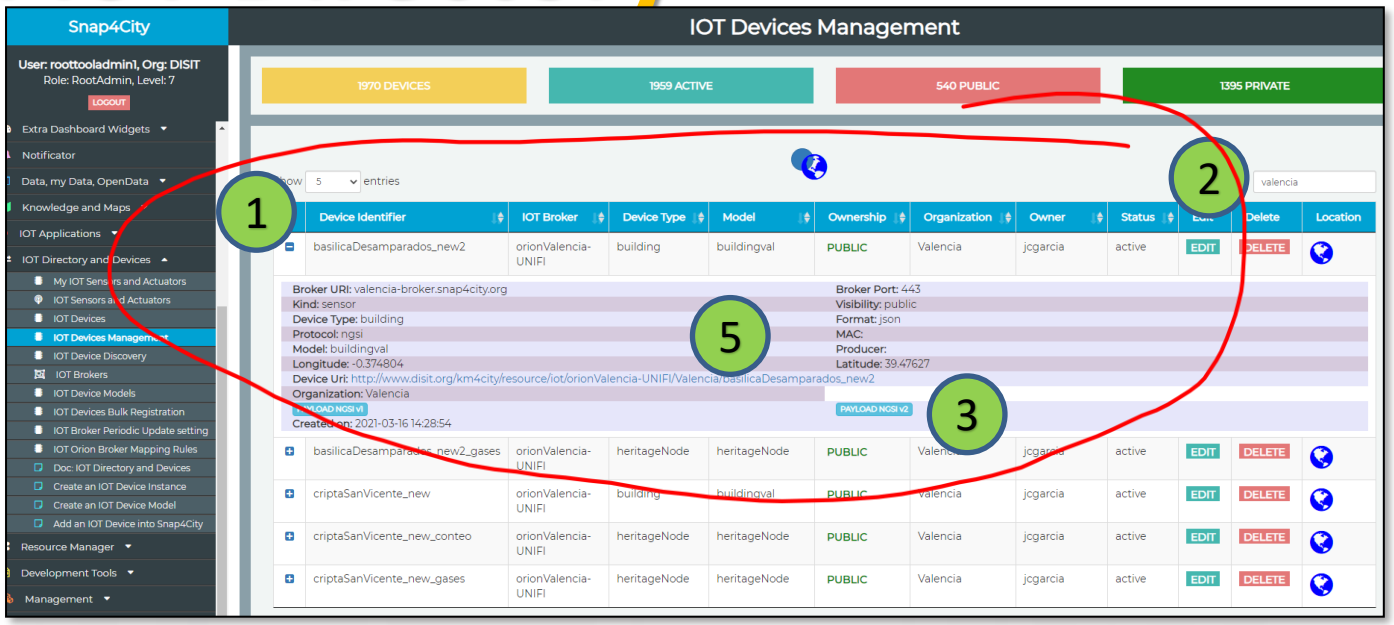
- **Verify that**
  - **Entity/Device Creation** see it on
    - Entity/IoT Directory, Entities/Devices list
    - Service Map if the device is Public
    - Data Inspector if the device is public and/or private
      - You can see the trend to see the time series and last value from the pin on map of the Data Inspector
    - Dashboards…… you can create a specific one using the Dashboard Wizard
  - **Entities/Device Data (time series)** see them on
    - Broker, via …… A dedicated services is coming….
    - Service Map if the device is Public
    - Data Inspector, this means that the data are on Storage
      - You can see the trend to see the time series and last value from the pin on map of the Data Inspector
    - Dashboards…… you can create a specific one using the Dashboard Wizard



Verify on IoT Directory

1. Verify the presence of the IoT Device you created
2. Verify the structure of the device by edit tool
3. See the **NGSI V2 JSON** format to be used on sending data msgs

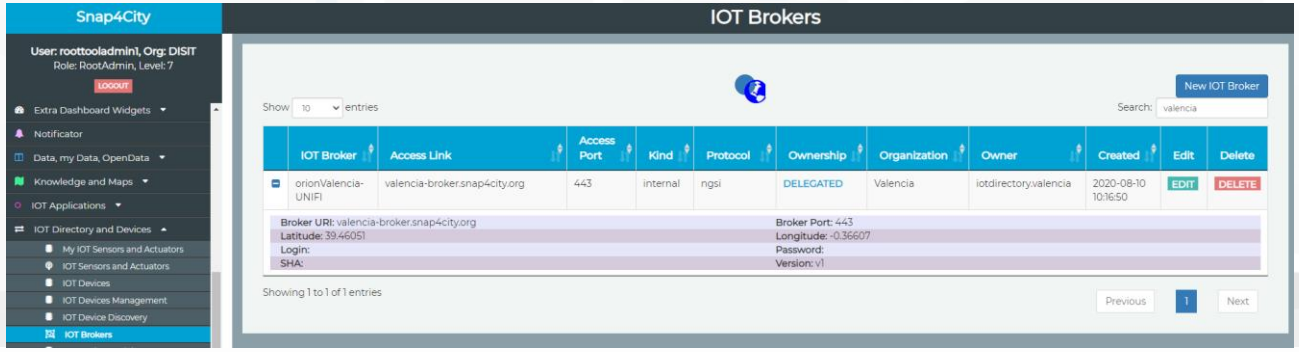
4. Call the IoT Broker to see the last data on it (some user name and Pwd can be needed)
   - Get the Broker Name from the device profile
5. See next slide



Broker service URL such as

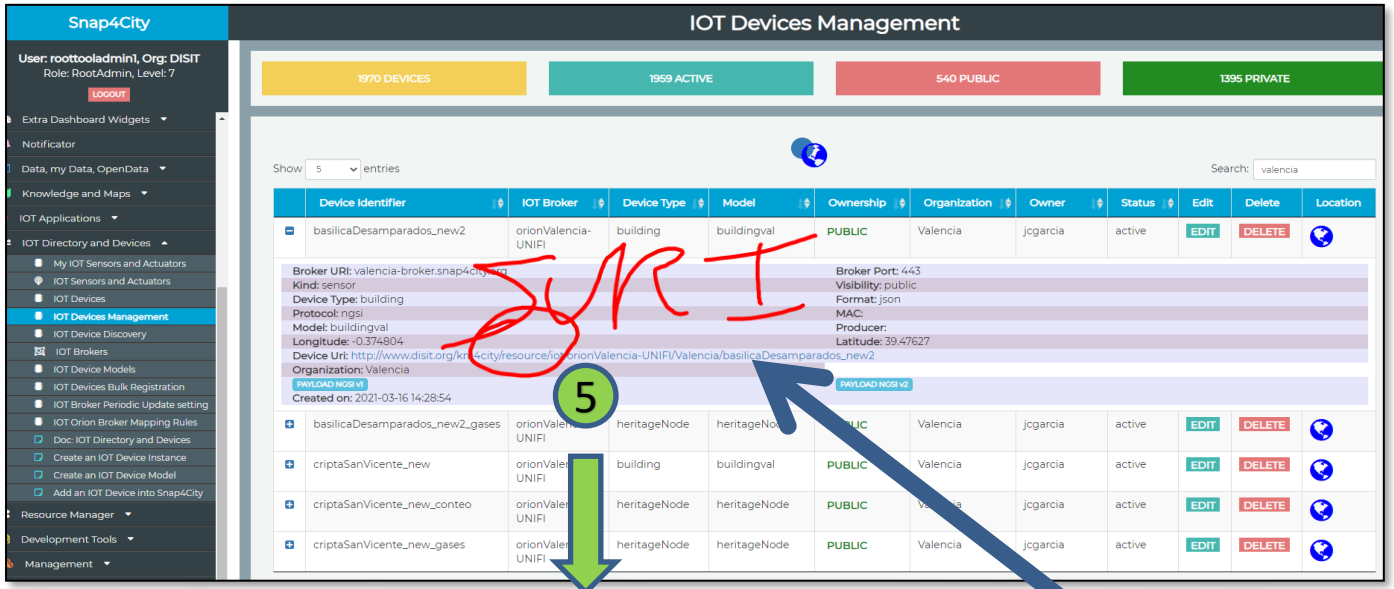
-- https://www.snap4city.org/brokername/v2/entities

-- http:// brokername.snap4city.org/v2/entities/

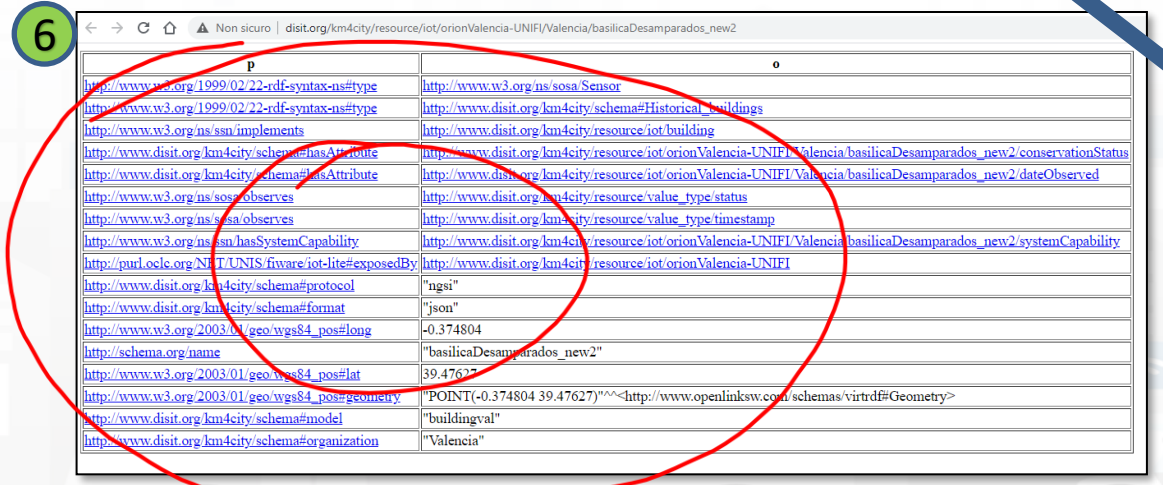


Verify on Knowledge Base

5. Click on ServiceURI (device URI) to Open in a new TAB the data sent on the KB



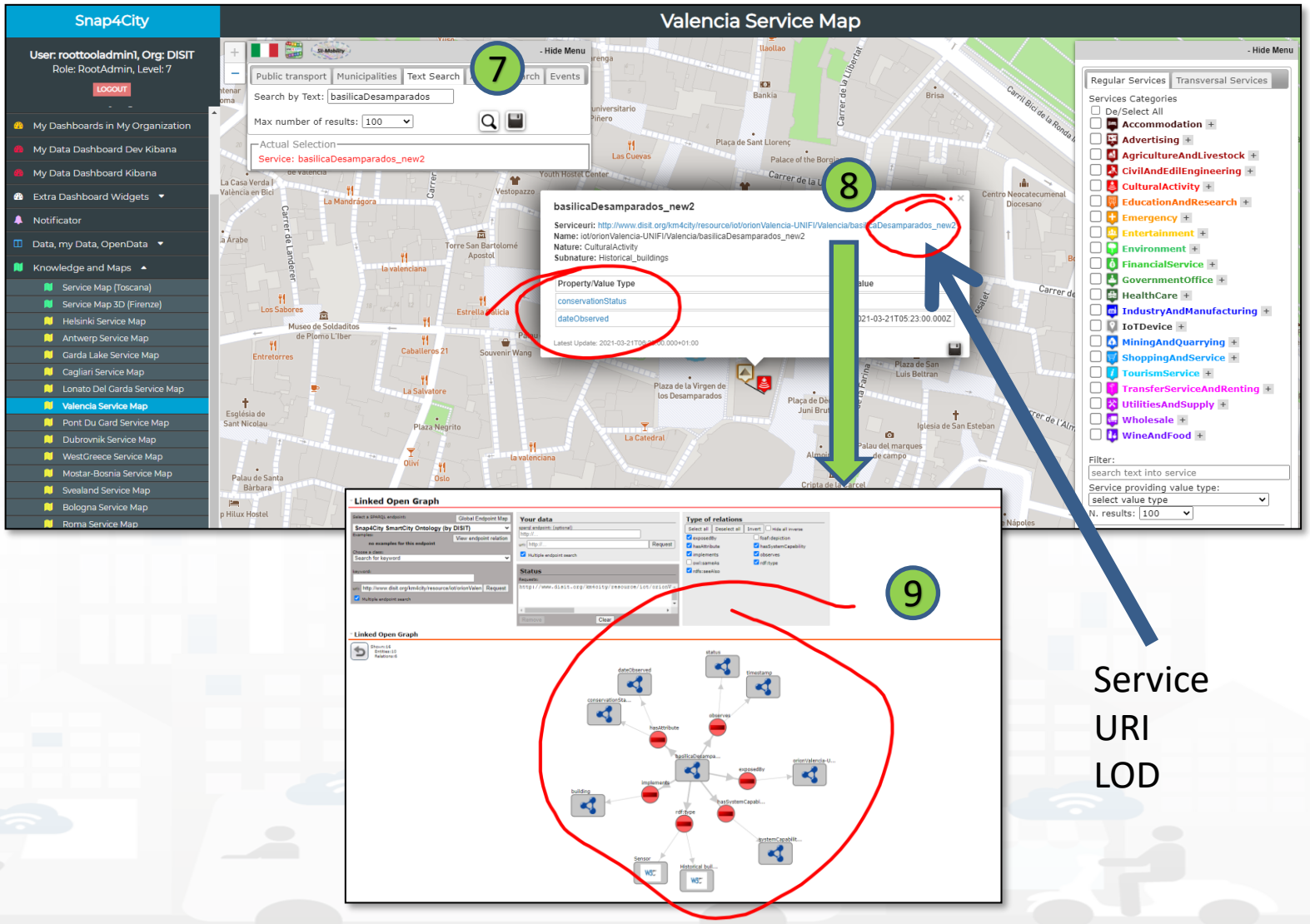
6. If your device is Public and you have sent data → the list of the last data from OpenSearch querying from KB will appear as:



Service URI LOD

Verify data ingestion on KB via ServiceMap

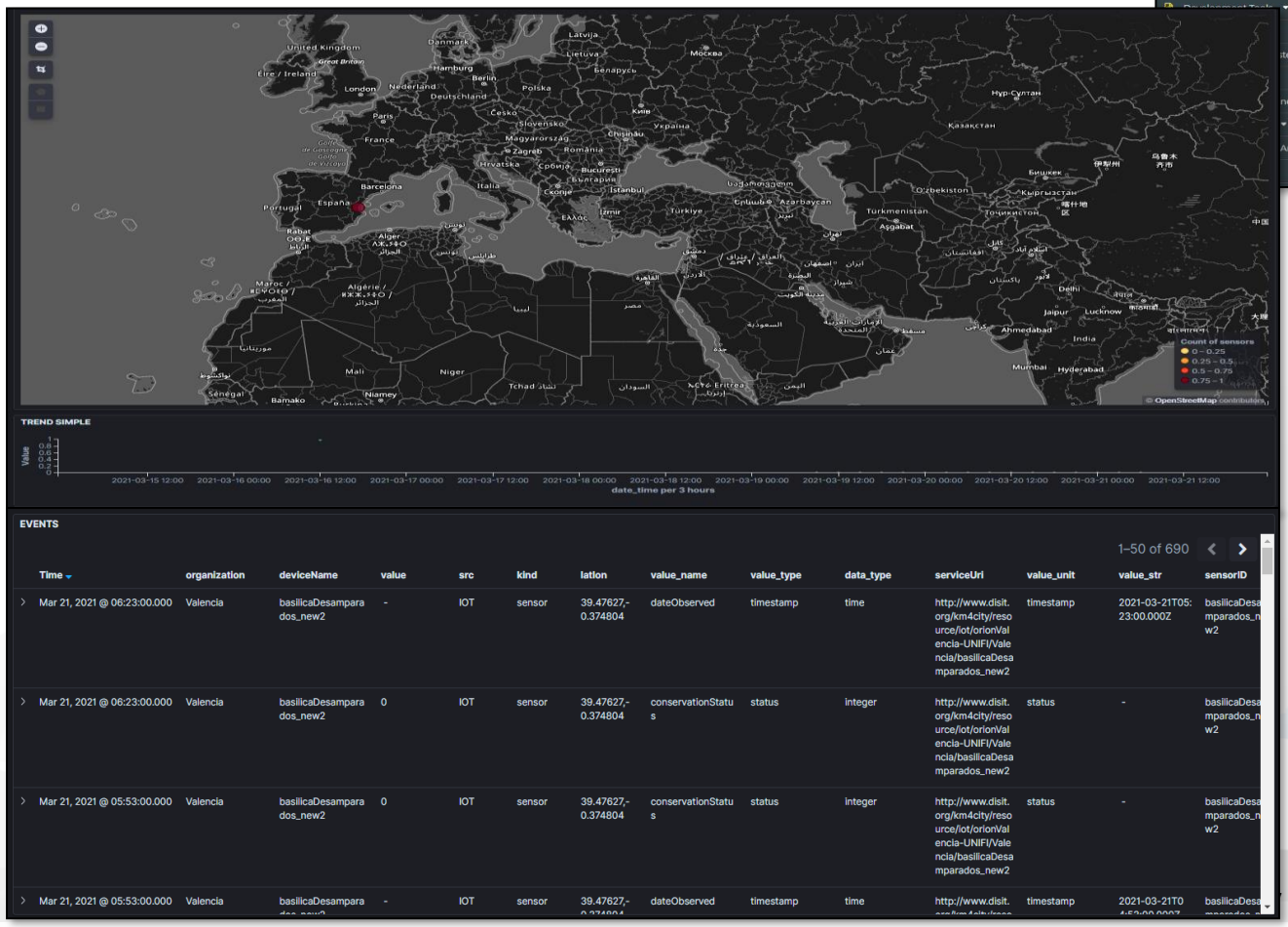
- 7) Verify on ServiceMa by Search on data location or by text name of the device
- 8) click on ServiceURI to jump on 9 LOG.DISIT.ORG to see semantic structure in terms of Linked Open Data, LOD

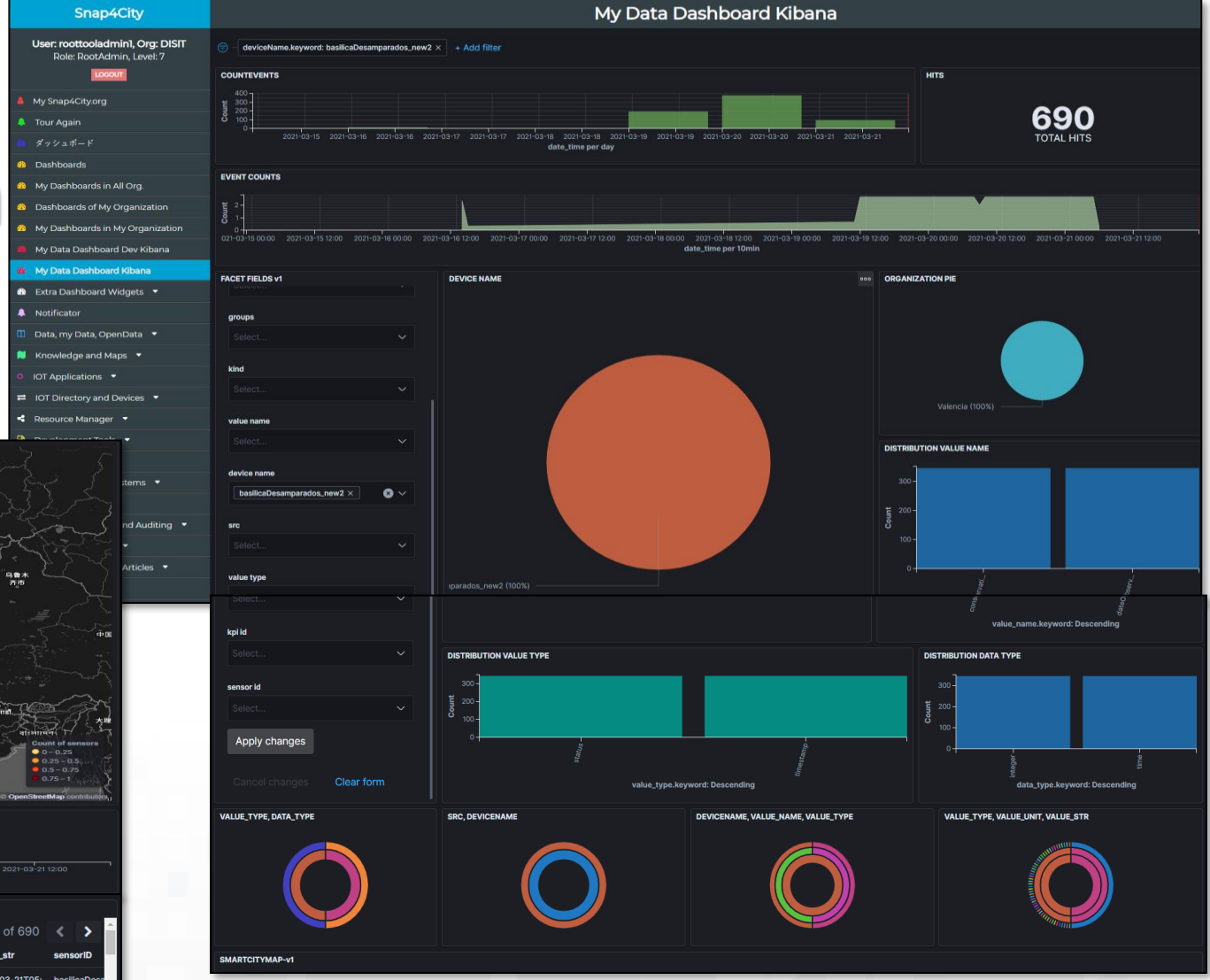


UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

Every data on OpenSearch

- Verify on OpenSearch Dash which monitor Open Search: My data Dash

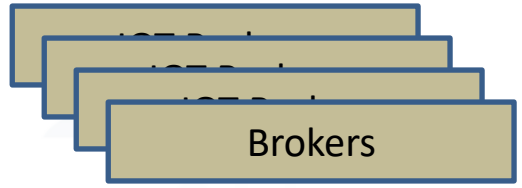




Additional Notes

- The **Orion Brokers** can be feed by means of
  - **IoT App/Proc.Logic** of Snap4City (to implement Agents and/or Adapters)
  - **IoT Agents** and/or **NGSI Adapters** by FiWare for different protocols
  - **Brokers** of any kind, different protocols and producers, also as Gateways, and they can be located on premise and/or on any cloud



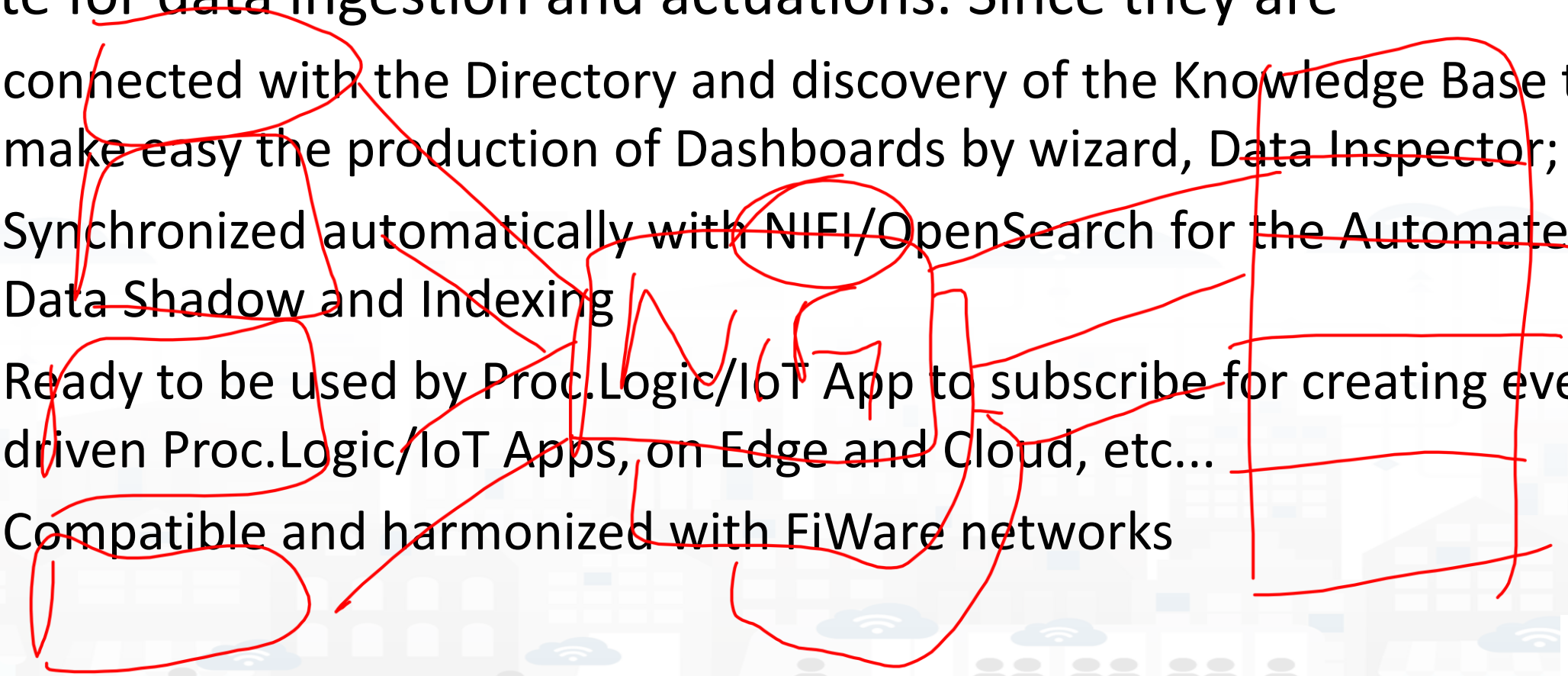


- **Proc.Logic/IoT App, IoT Agents, Adapters** can
  - be on IoT Edge
  - be implemented as IoT App/Proc.Logic of Snap4City
  - be on other clouds and services
  - work on a large range of different protocols and kinds
  - have or not Snap4City libraries installed



Additional Notes

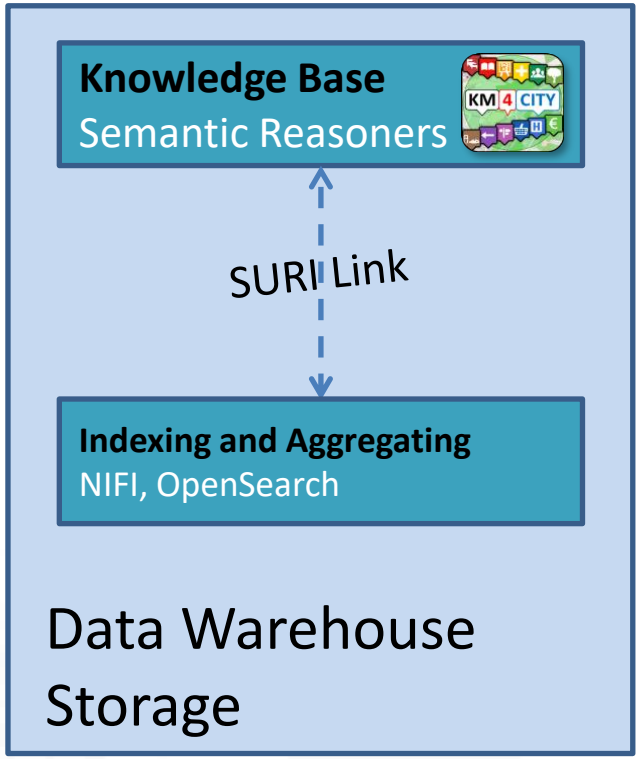
- The **Internal Orion Brokers** at Snap4City Orion Broker are used as a gate for data ingestion and actuations. Since they are
  - connected with the Directory and discovery of the Knowledge Base to make easy the production of Dashboards by wizard, Data Inspector;
  - Synchronized automatically with NIFI/OpenSearch for the Automated Data Shadow and Indexing
  - Ready to be used by Proc.Logic/IoT App to subscribe for creating even driven Proc.Logic/IoT Apps, on Edge and Cloud, etc...
  - Compatible and harmonized with FiWare networks



Direct Ingestion

- **Direct Data Ingestion is also possible:**
  - **From data sources to Data Warehouse Storage of Snap4City, Snap4Industry**

- **Data Warehouse Storage includes:** KB, and I&A, reported on right side can be acted via API REST Call
  - for direct feeding data into store and retrieval,
  - which can be exploited by:
    - IOT App/Proc.Logic
    - applications in Python, R Studio, Java



Notation Terminology

WHERE	Are synonymous at level of service which can be IoT device or entity with data and references to	Are synonymous at level of the single attribute of the entity, device, service, etc.
IoT Directory, Entity Directory	IoT Device, Entity Instance, Device URI	Sensor, Actuator, Attributes, Values (value name)
Knowledge Base, ServiceMap, SmartCity API, ASCAPI	Service, ServiceURI, SURI	Attribute, Metric, SURI with metric
DataInspector, Wizard, Dashboard	Value Name(Model name, Category)	Sensor, Sensor Actuator, ValueType
IoT App., Proc.Logic, Node-RED	ServiceURI, SURI	SURI and its real time results of the objects into the data structure

ServiceURI, SURI of a sensor device:

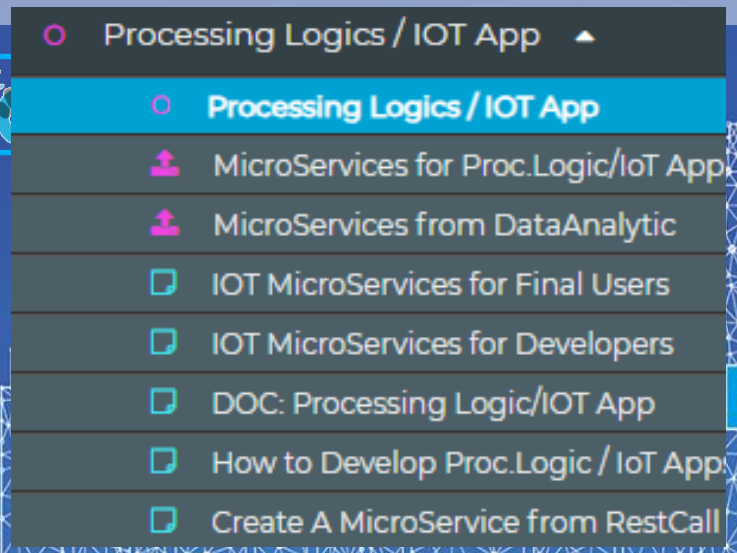
- http://www.disit.org/km4city/resource/METRO759
- http://www.disit.org/km4city/resource/iot/orionCAPELON-UNIFI/CAPELON/Streetlight%3A90FD9FFFFEBD5A7F

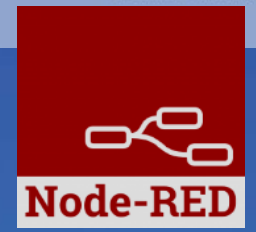
ServiceURI, SURI extended with attribute/variable/value:

- http://www.disit.org/km4city/resource/METRO759&metric=vehicleFlow
- http%3A%2F%2Fwww.disit.org%2Fkm4city%2Fresource%2FMETRO759&metric=vehicleFlow
- In some cases
  - **http://www.disit.org/km4city/resource/METRO759/vehicleFlow**

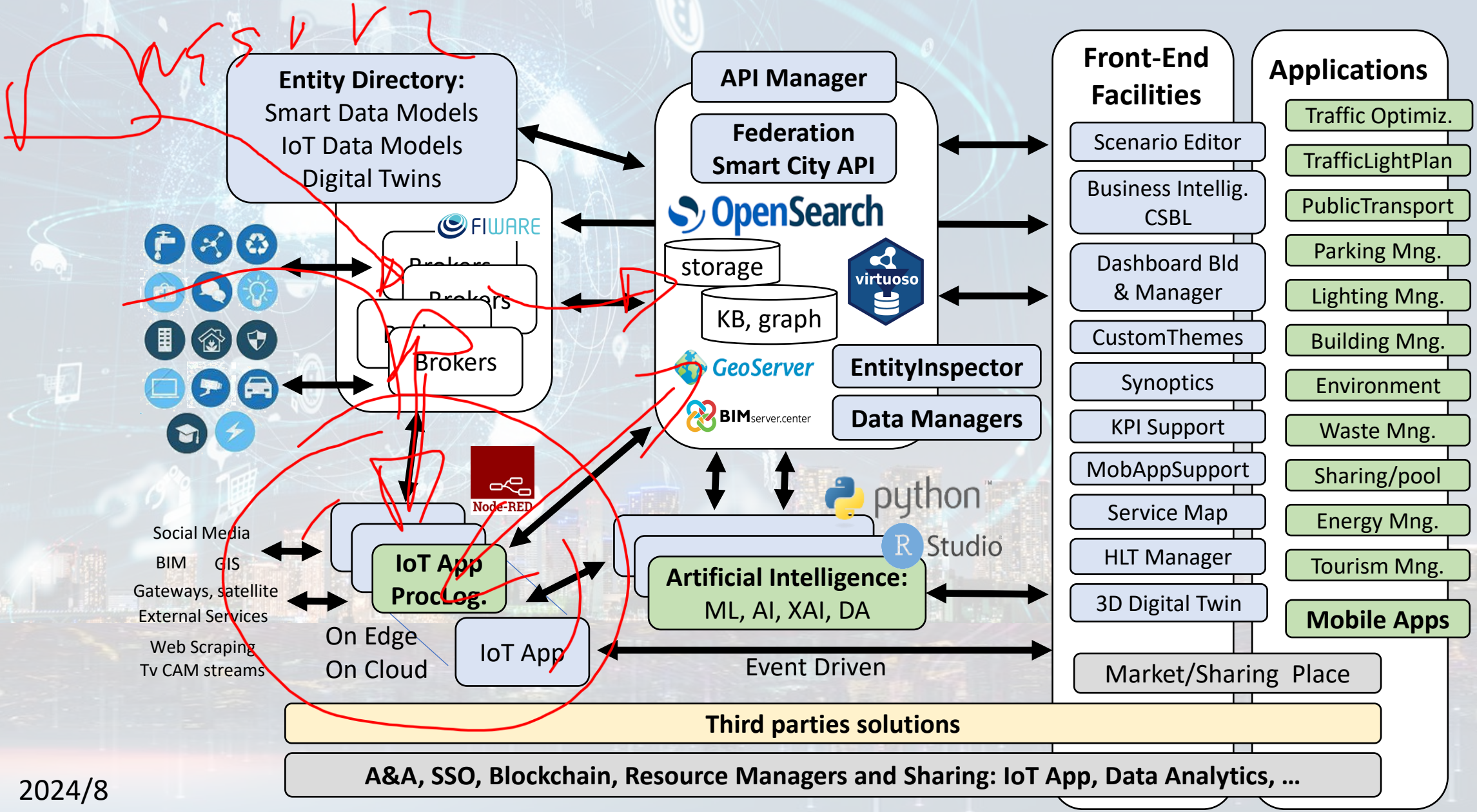
Exploiting Storage data by using: IoT App / Proc. Logic







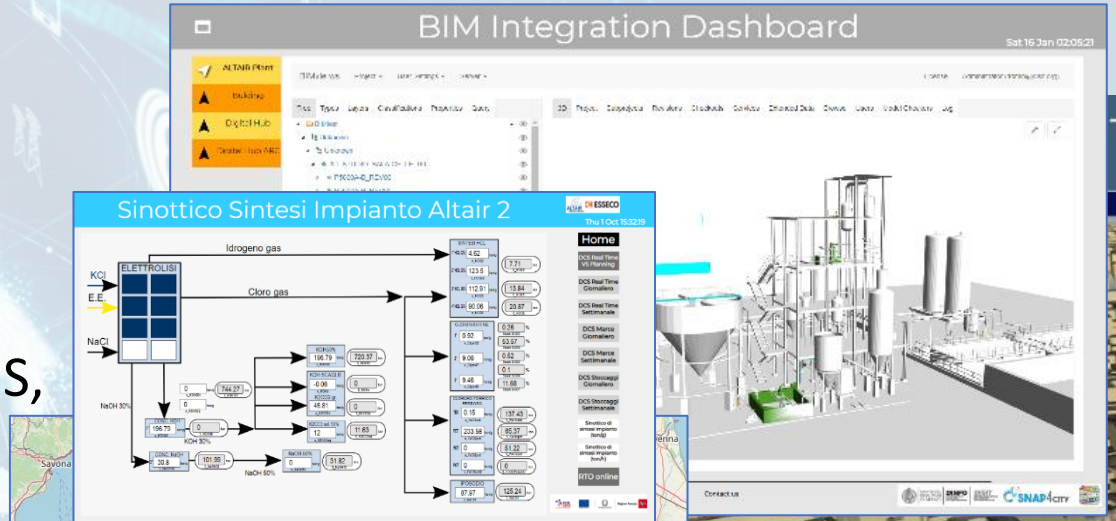
Technical Architecture



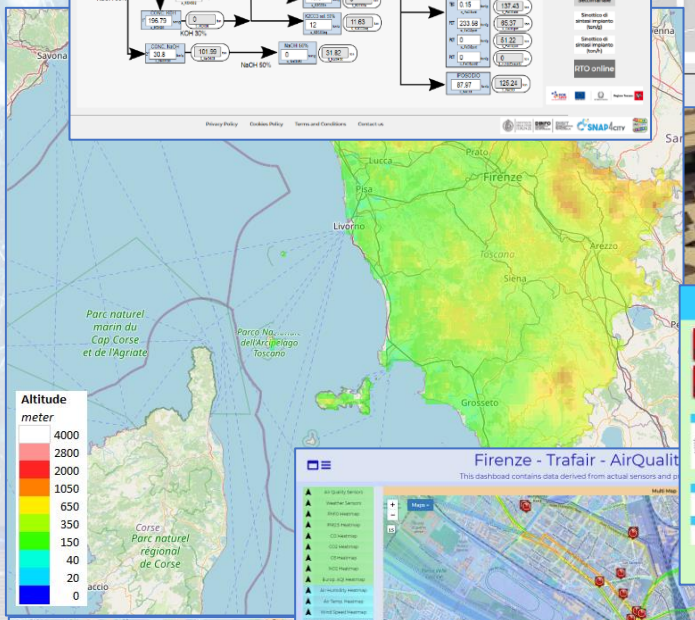
High Level Types

Snap4City (C), February 2025

- POI, IOT Devices, shapes,..
  - FIWARE Smart Data Models,
  - IoT Device Models
- GIS, maps, orthomaps, WFS/WMS, GeoTiff, calibrated heatmaps, ..
- Satellite data, ..
- traffic flow, typical trends, ..
- trajectories, events, Workflow, ..
- 3D Models, BIM, Digital Twins, ..
- OD Matrices of several kinds, ..
- Dynamic icons/pins, ..
- Synoptics, animations, ..
- KPI, personal KPI,..
- social media data, TV Stream,
- routing, multimodal, constraints, ..
- decision scenarios, ….
- etc.



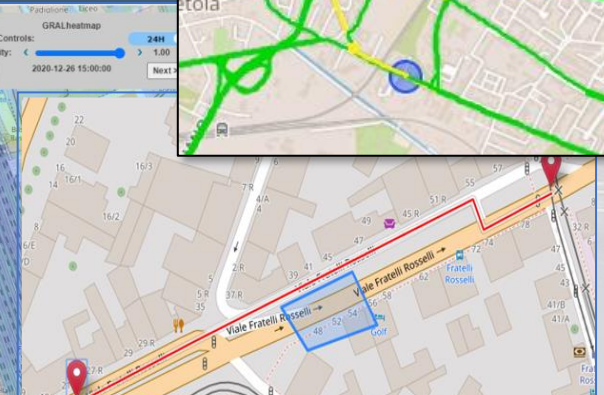








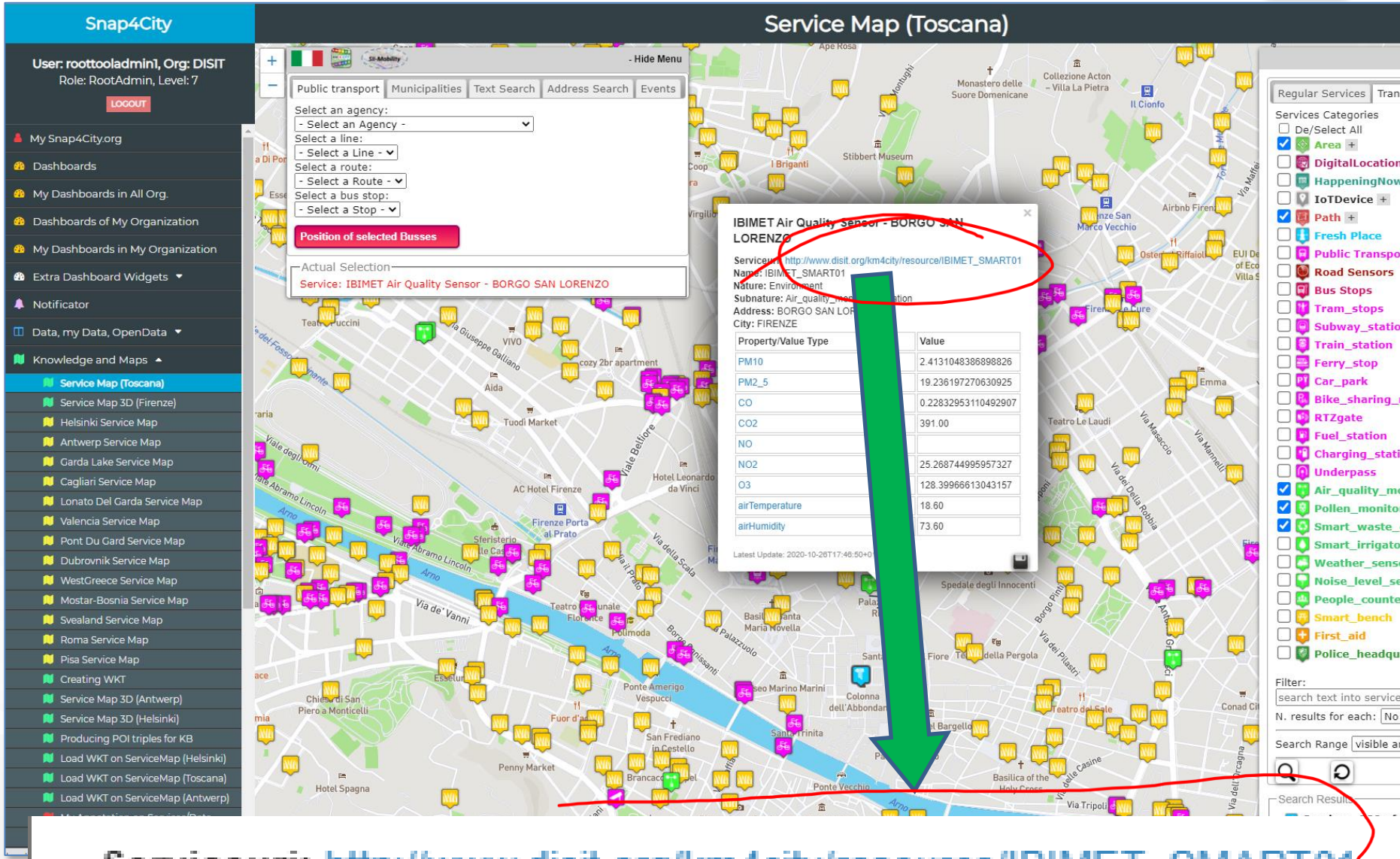






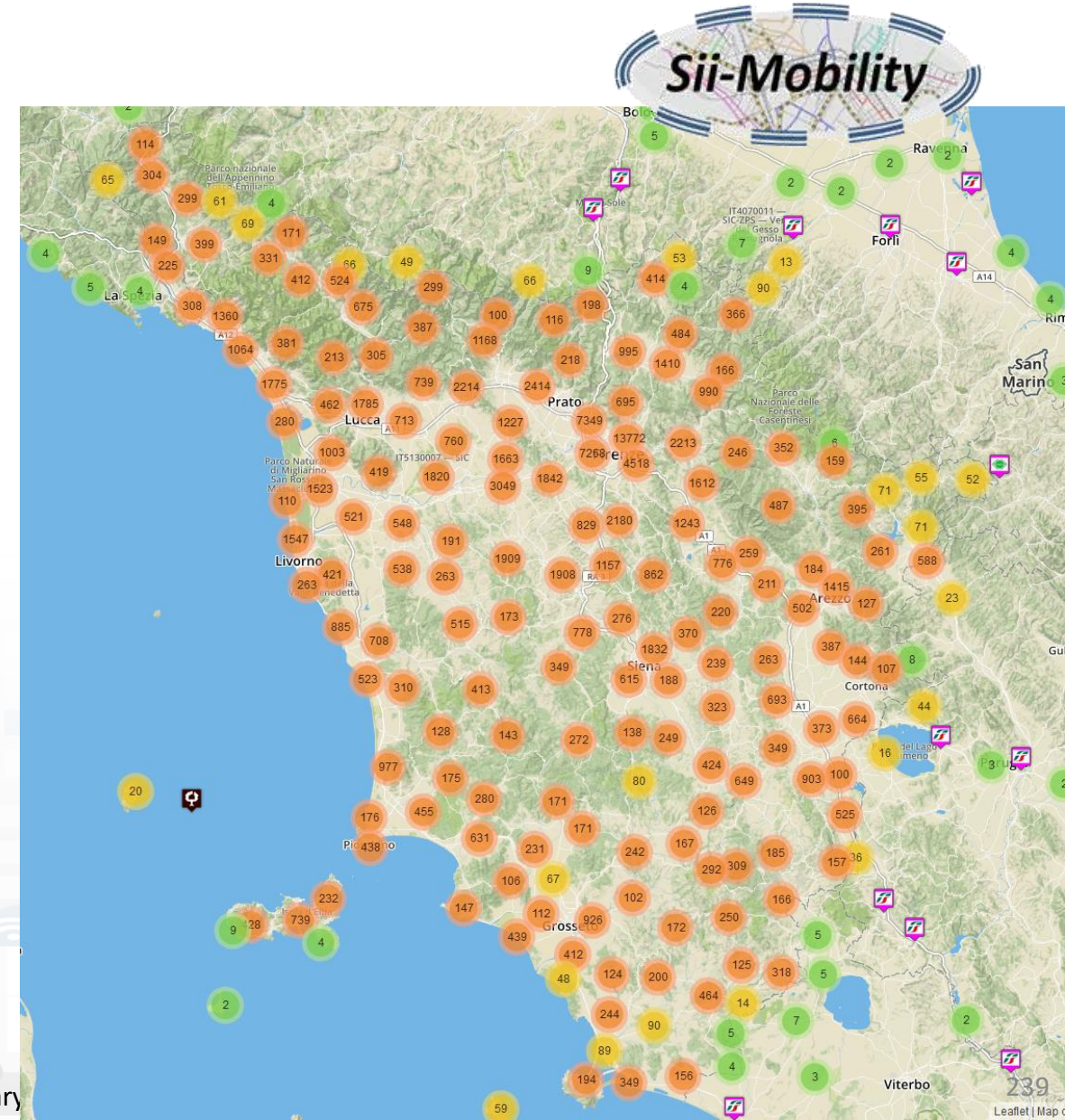
ServiceMap

- **For PUBLIC**:
  - IOT Devices, Sensors,
  - Sensor mobile,
  - Actuators,
  - Virtual Sensors,
  - POI, etc.
- See as
  - ServiceURI



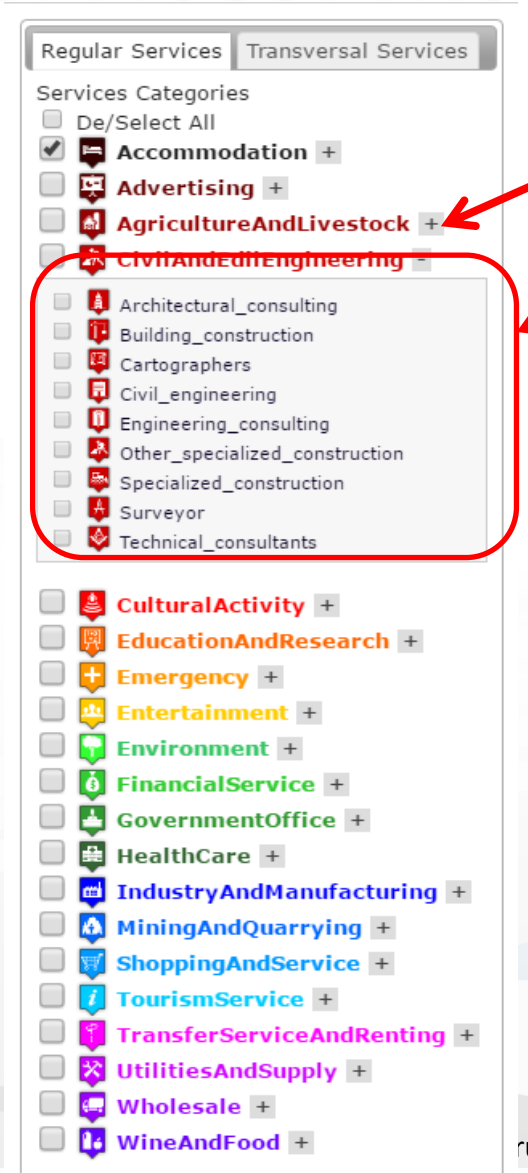
Data Domains

- **Street and geoinformation of the territory and details for routing, navigation, …**
- **GeoResolution, Environmental data**
- **Mobility and Transport**: public and private, public transport, parking status, fuel stations prices, traffic sensors, etc.
- **Culture and Tourism**: POI, churches, museum, schools, university, theatres, events in Florence
- **Environmental**: pollution real time, weather forecast, etc.
  - Environmental data geo resolution
- **Social Media**: twitter data
- **Health**: hospital, pharmacies, status of the first aid triage in major hospitals, …
- **Alarms**: civil protection alerts, hot areas, …



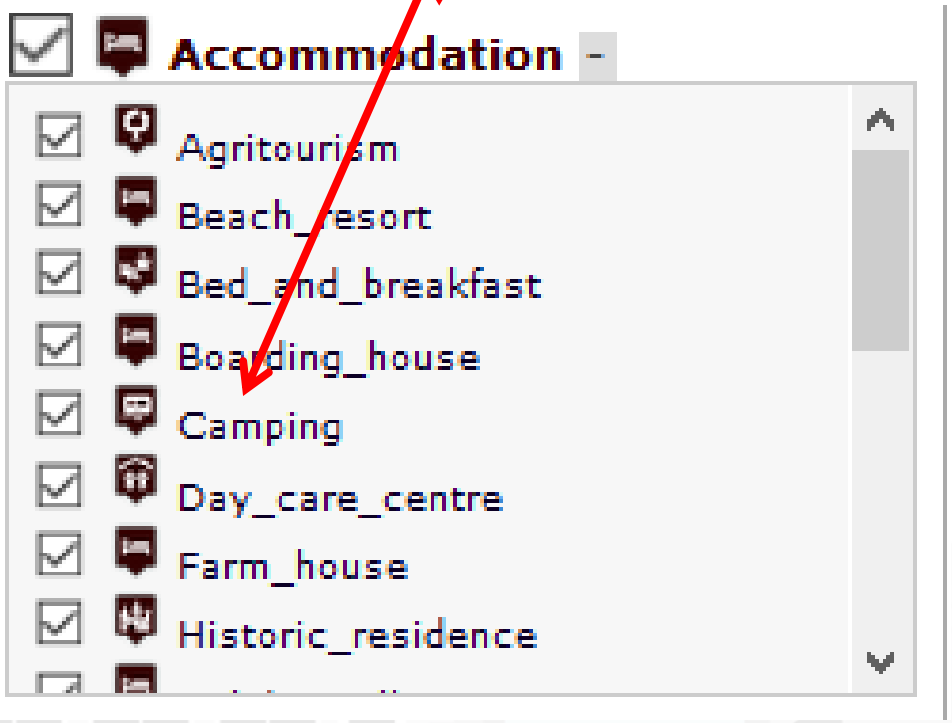
Concepts of Services: Macro and subcathegory

A SKOS area into the Km4CIty Ontology and Knowledge base for modeling POI and any element on map



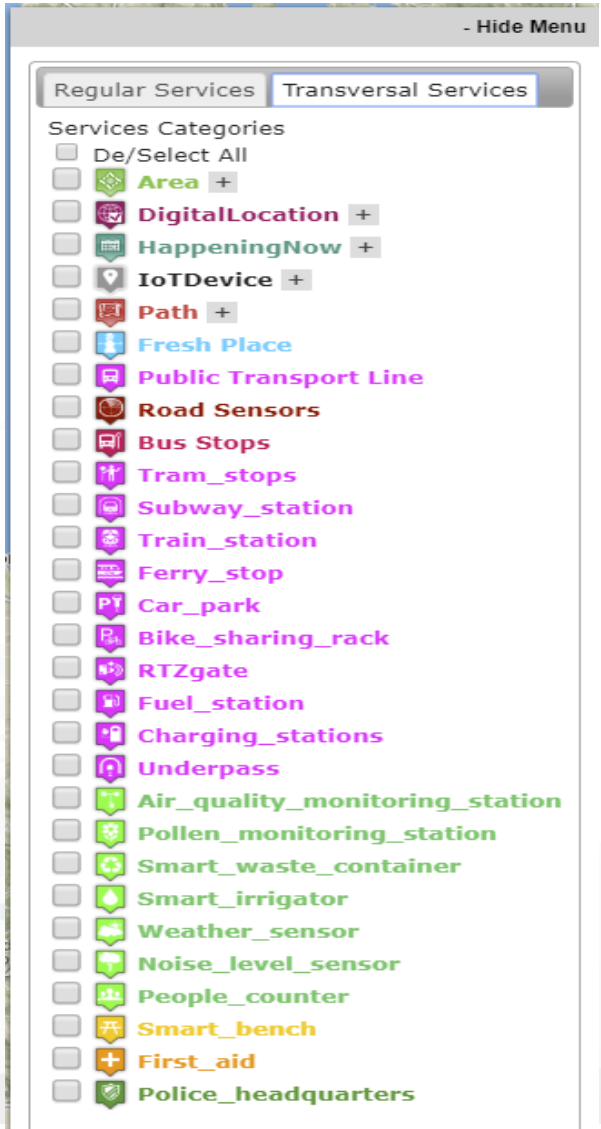
20 Service Macro Classes (The Nature)

Service subClasseses (the SubNature)

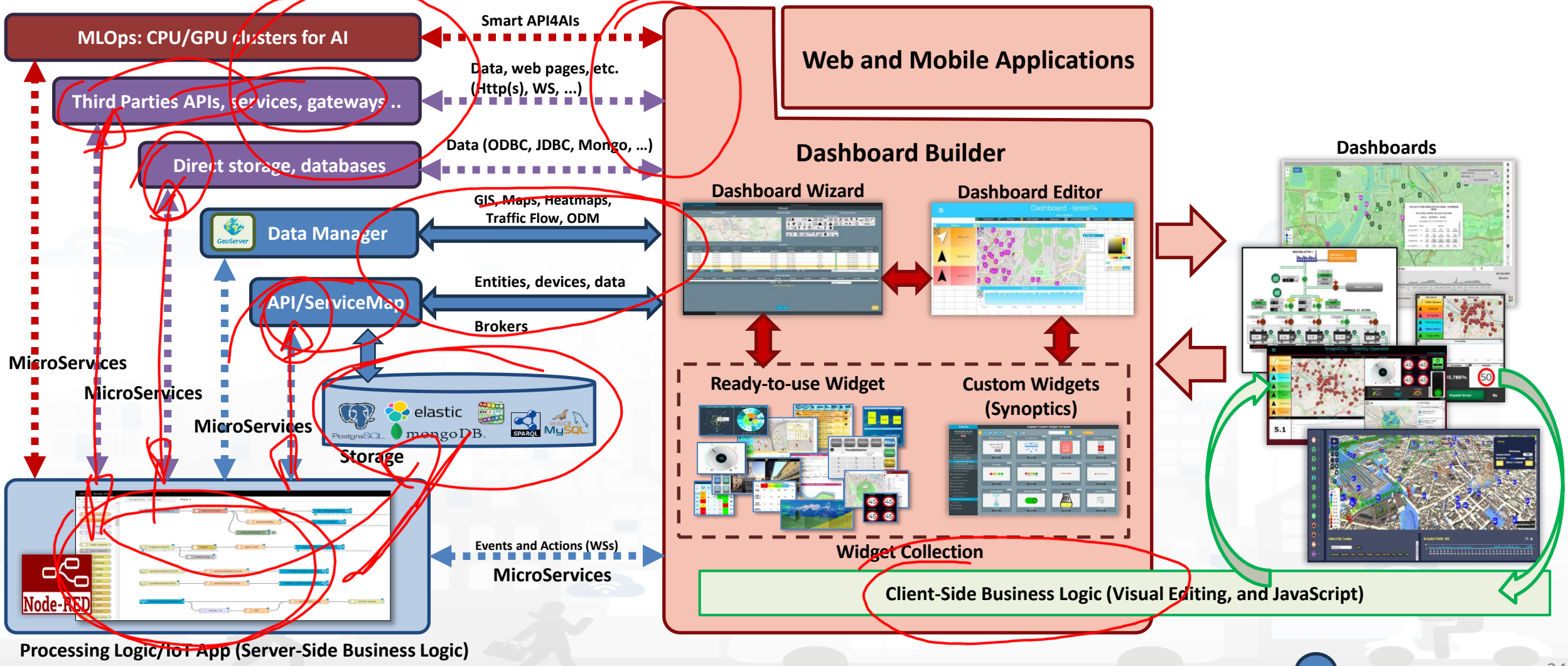


Access to Entities

- IoT, POI, etc.
- **Classify**: macro (nature) and subcategories (subnature)
- **Position**: GPS, address, telephone, fax, email, URL, …
- **Description**: textual, multilingual, with images, …
- **Link** to dbPedia, Linked Open Data
- **Links to other services**
- **ActionURL: links to actions on data (open, edit, show, etc.)**
- **Real time data if any:** sensors data, timeline, events, prices, opening time, rules of access, status of services, status of queue, etc..
- *See transversal services on ServiceMap*
  - *Regular and in test platform*

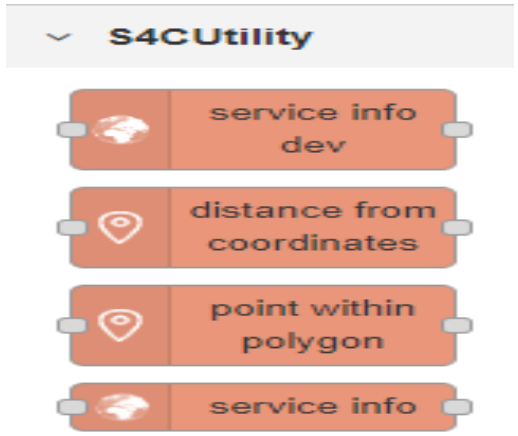


How the Dashboards / Apps Exchange data (2024/8)

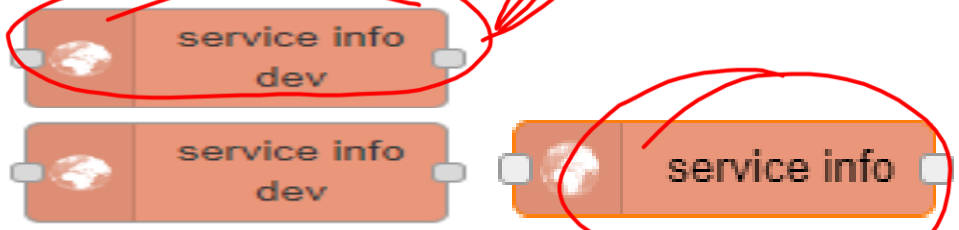


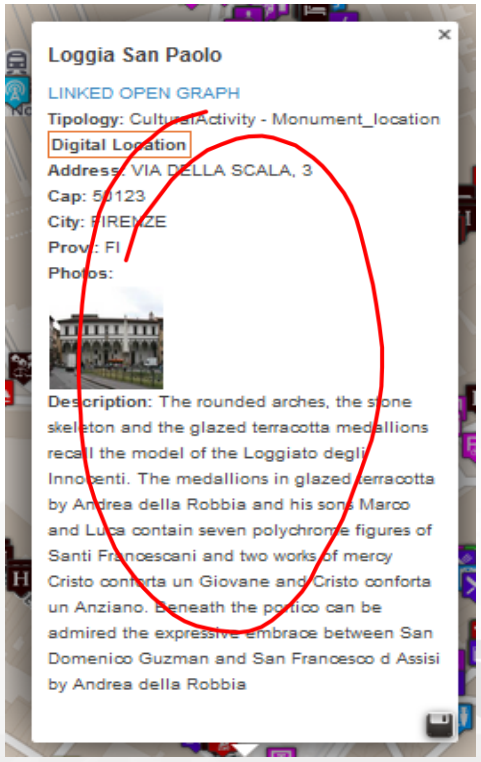
S4CSearch

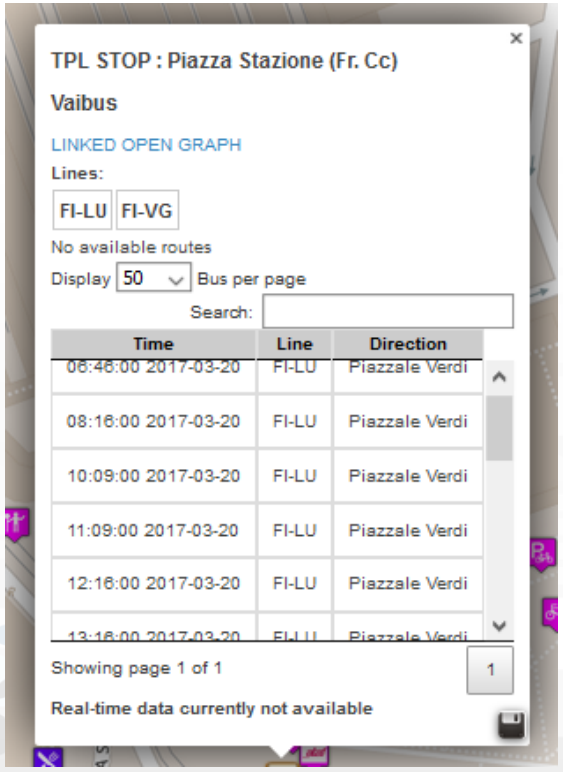
Node-RED

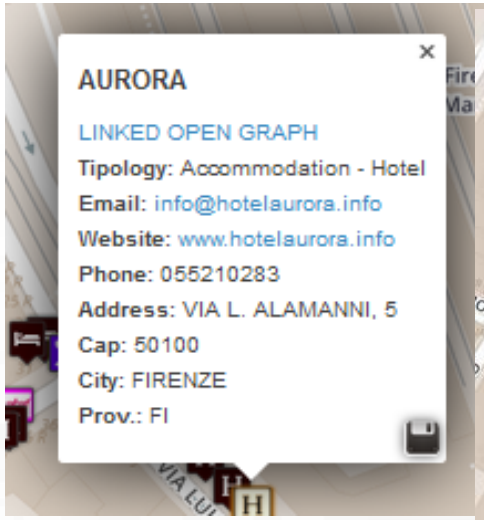


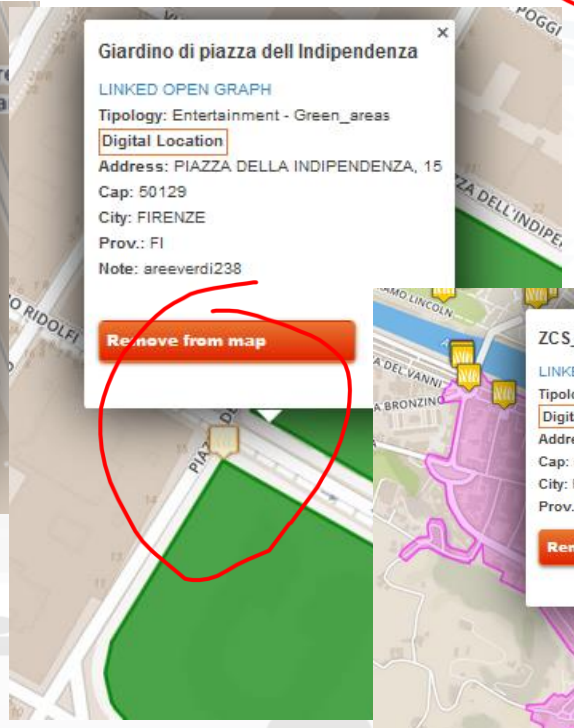
- ANY kind of sensors
- **To Get DATA of a Service / POI /sensor**
  - Historical and real time
  - Real Time

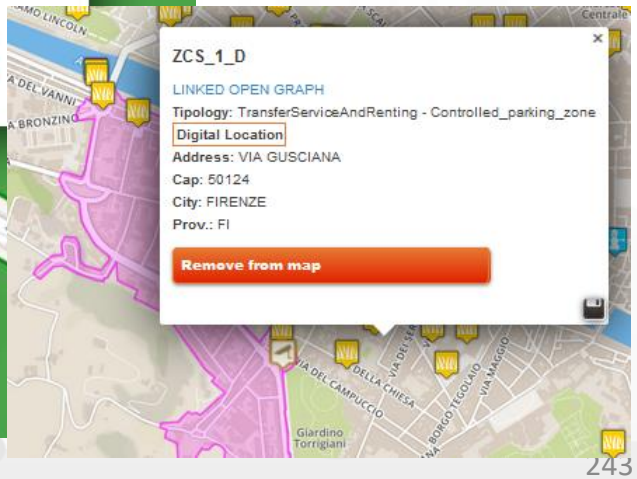












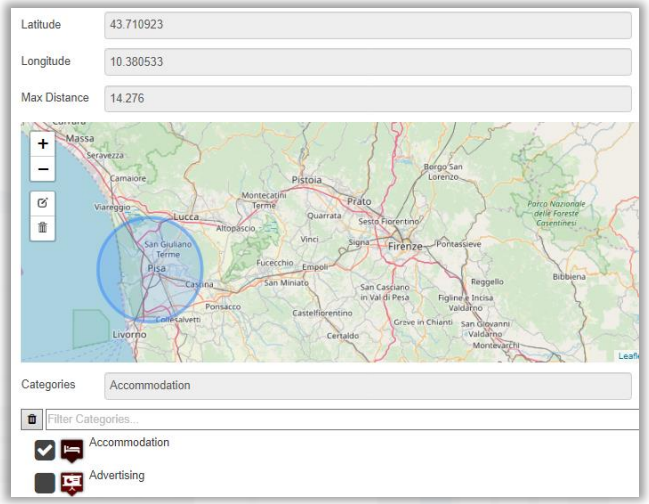
S4CSearch

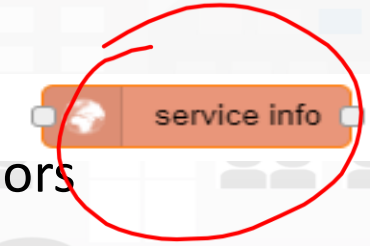
Smart City Entities Search

Simple and Fast

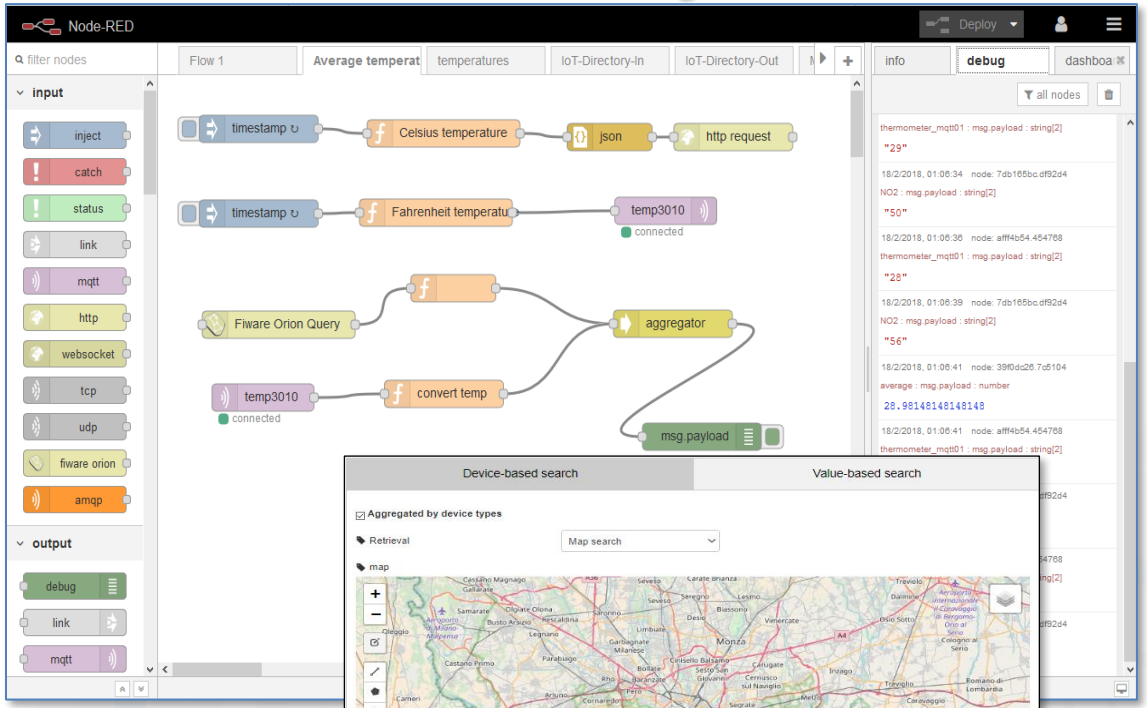


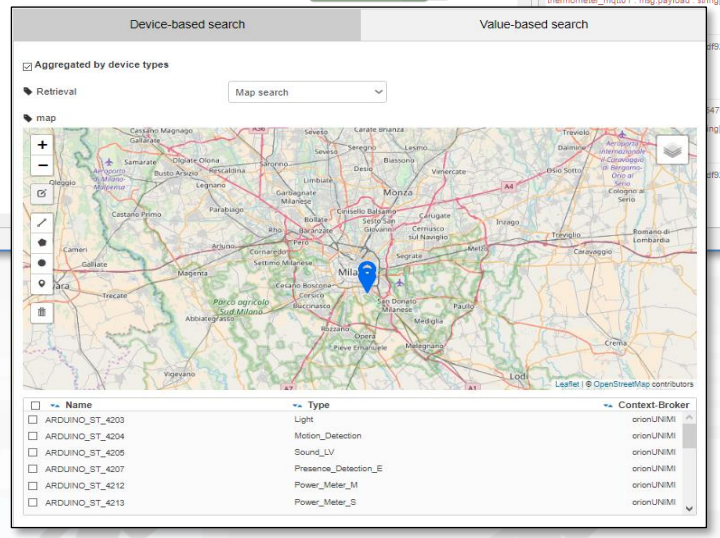
- **For example to search for:**
  - POIs:
    - near a GPS position, from text, along a path, in an area, etc..
  - Public Transport information / data
  - Suggestions
  - Public Transport Means Routes/Paths
  - Events in the area
  - Value Type (kind of data)
  - Etc.
- **To Get DATA of a Service / POI /sensor**
  - Real Time
  - ANY kind of sensors

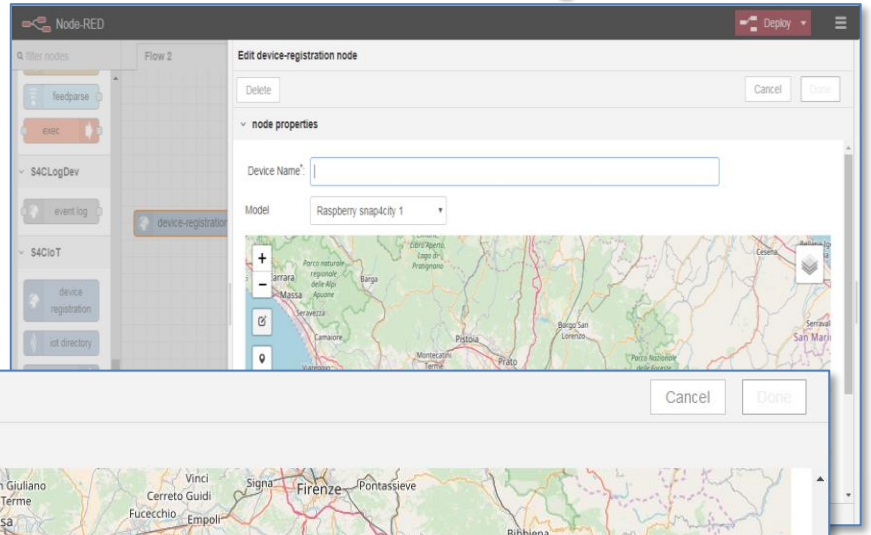


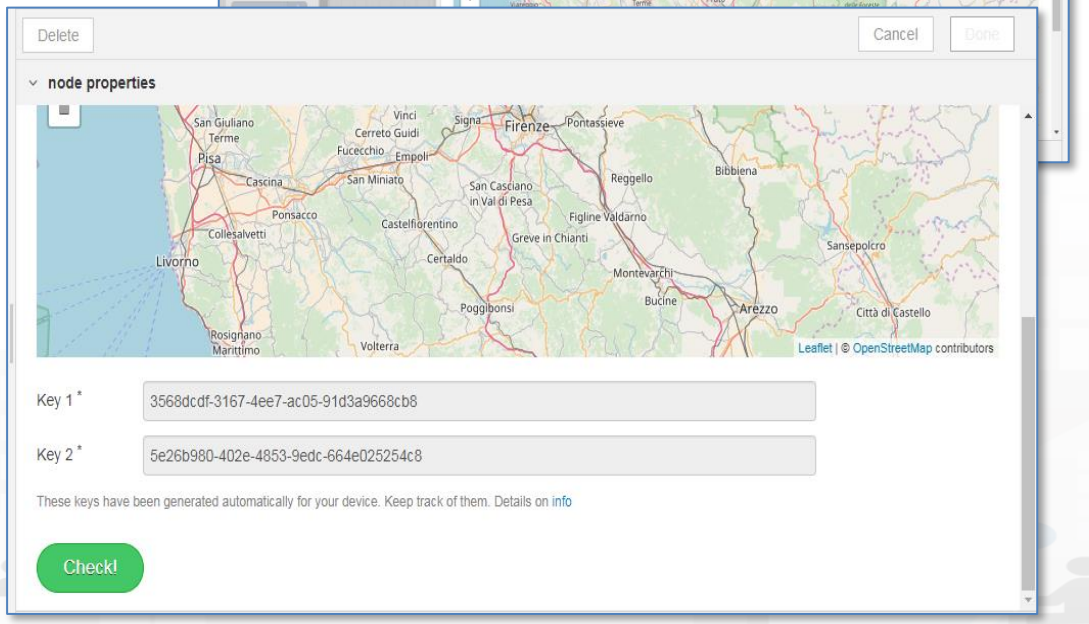


IOT Discovery on IOT Application Development









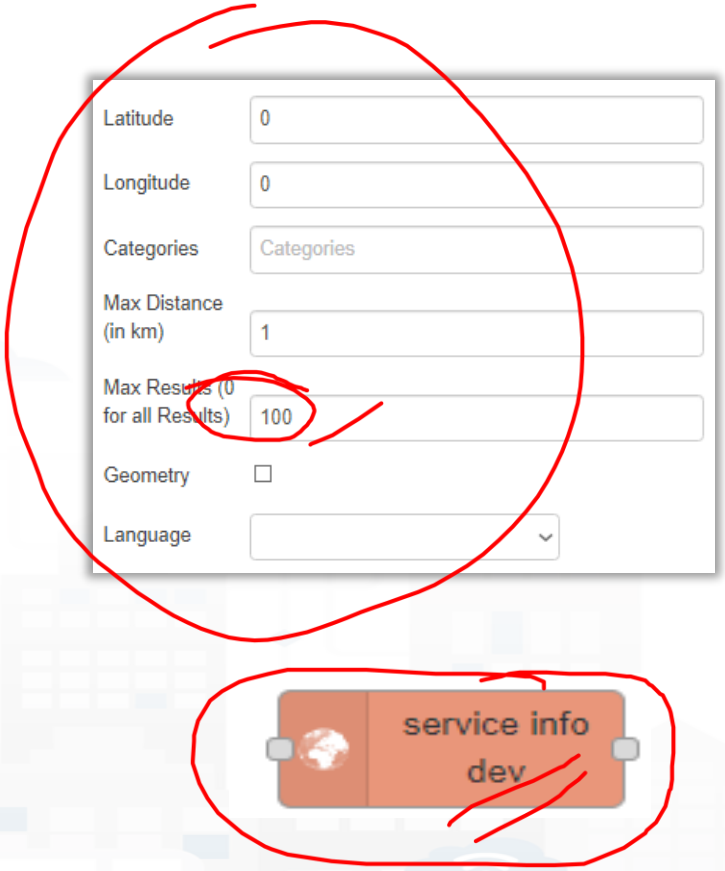
S4CSearch Adv

Smart City Entities Advanced Search

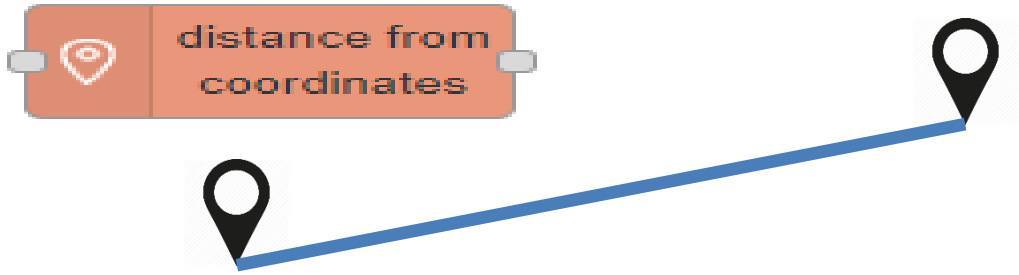
Flexiblity



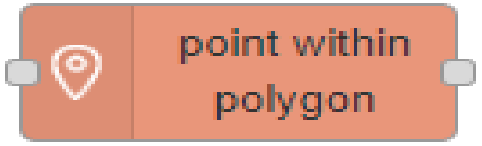
- Similar to basic Search functions but with more flexibility of the function for programming the search
- Adding Dynamic behavior:
  - Getting in input JSON with parameters
- **To Get DATA of a Service / POI /sensor**
  - Historical and real time
  - ANY kind of sensors



- **Distance from GPS point**



- **Point is in Polygon ?**
  - **Polyline as WKT**

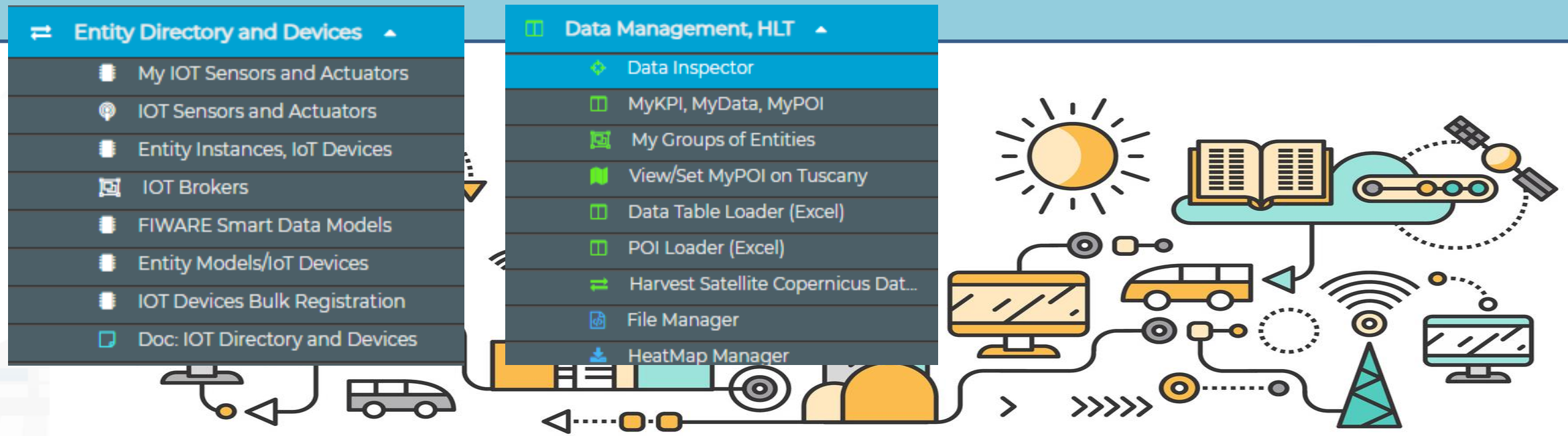




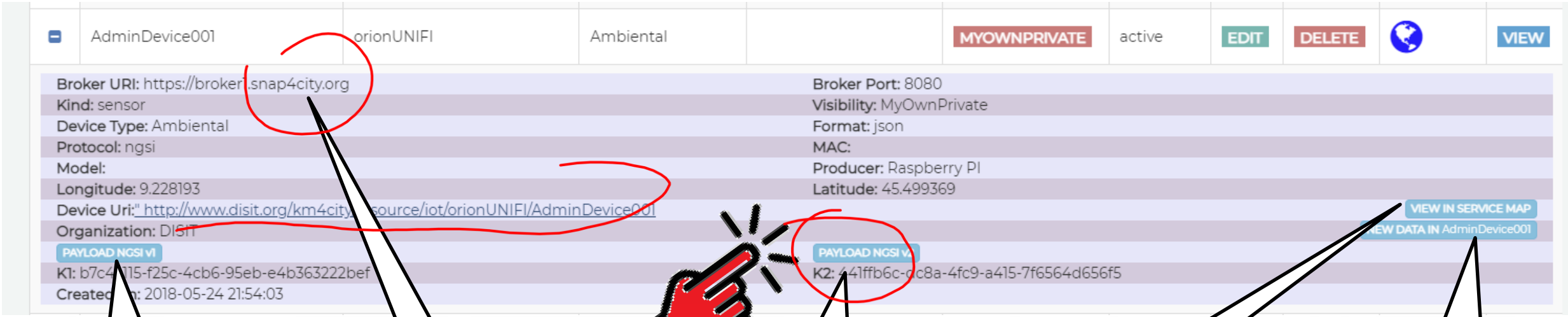
Your exercise

- serching of an IoT device/Entity on Service map… e.g., METRO….
  - Identification of the service URI, SURI
  - Go on Service Map, multi Org, or Super
  - Alternatively: get SURI from the Entity Directory
- Go in Data Inspector OS on the same device
  - See the data in the inspector
- Create an example
  - Inject SURI on Service Info Dev, see the data, and time series if you aer Area Manager
  - Use pattern: Inject → function → service info dev → debug
  - Retrieve a data, retrieve a set of data in the last 24 hours

search vs services, the ServiceURI



Understanding / Testing an Entity/ IoT Device



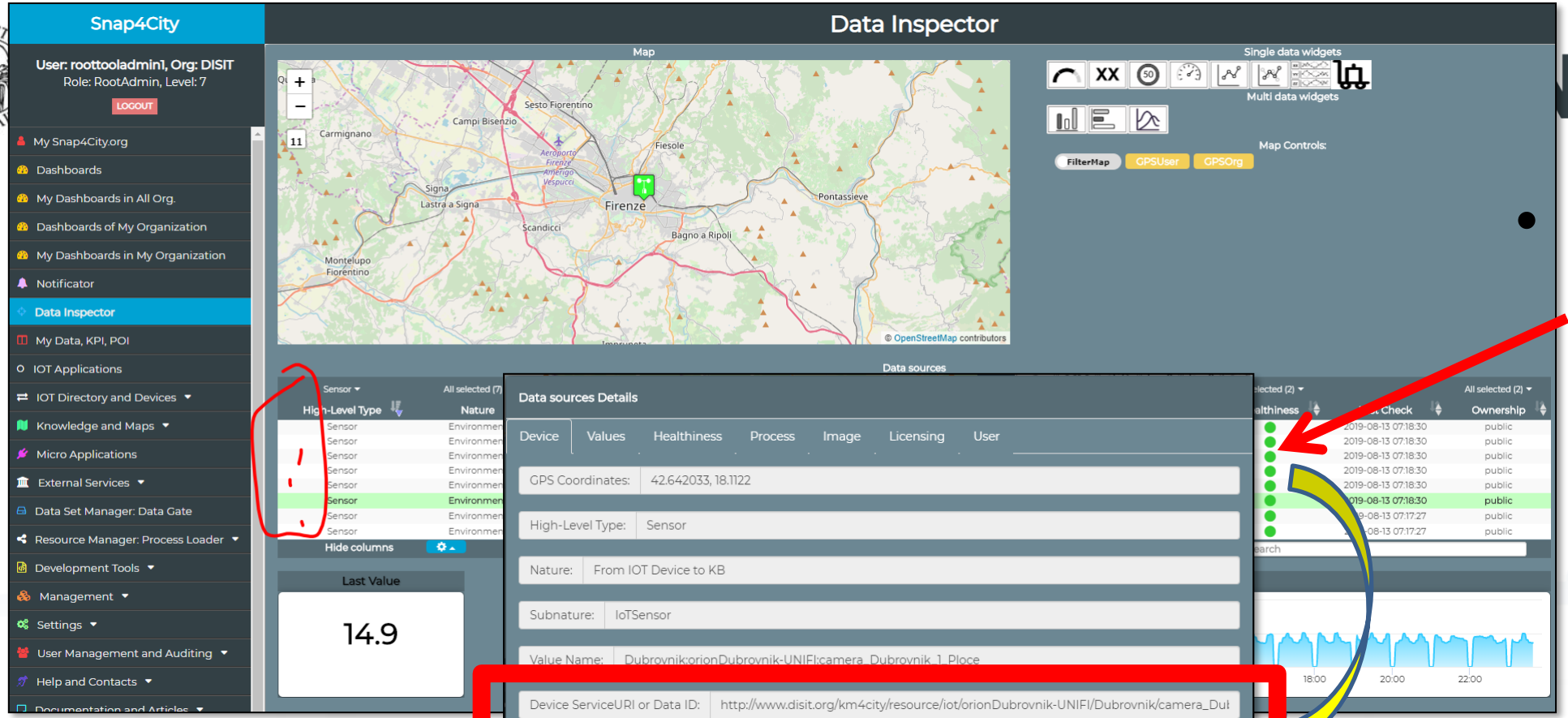
See Payload NGSI V1 in JSON directly from the Broker, Last message of the broker

The Broker

See Payload NGSI V2 in JSON directly from the Broker, Last message of the broker

See IoT Device on ServiceMap

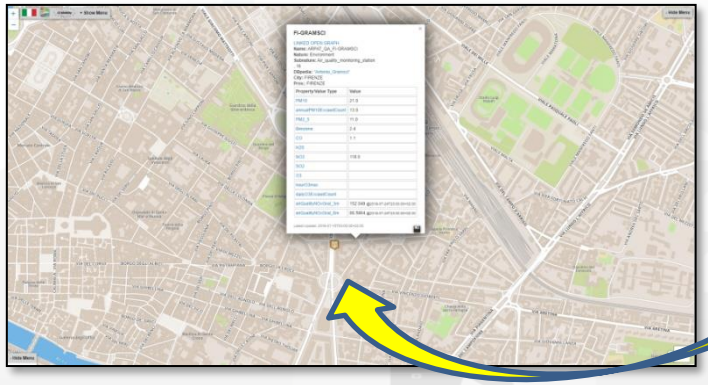
Create a Message to be sent at the IoT broker regarding this device.

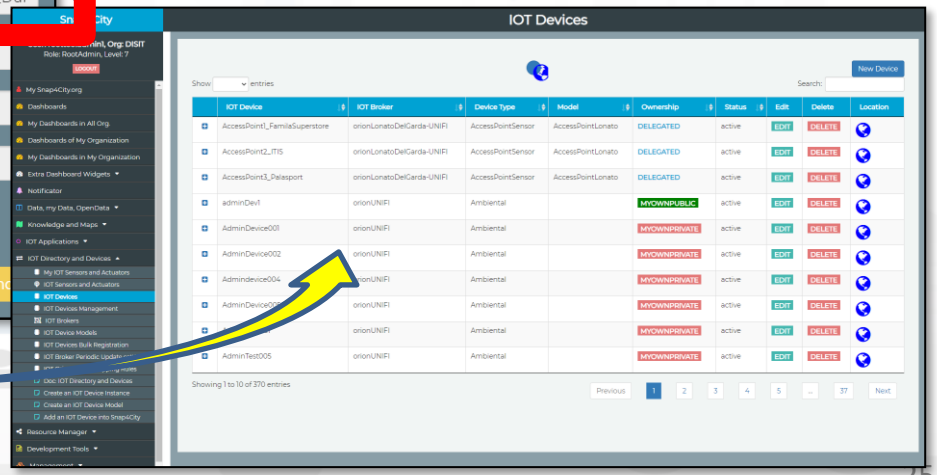


Knowledge Base view

- Click with the mouse on it

Data Inspector Wizard





Some functionalities are limited to certain roles

Notation Terminology

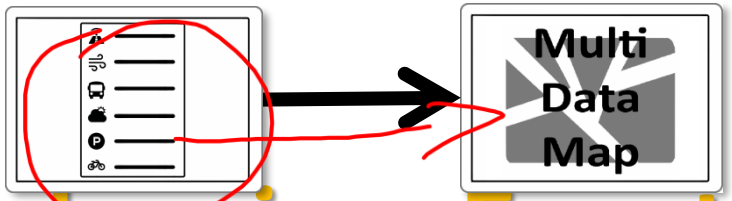
WHERE	Are synonymous at level of service which can be IoT device or entity with data and references to	Are synonymous at level of the single attribute of the entity, device, service, etc.
IoT Directory, Entity Directory	IoT Device, Entity Instance, Device URI	Sensor, Actuator, Attributes, Values (value name)
Knowledge Base, ServiceMap, SmartCity API, ASCAPI	Service, ServiceURI, SURI	Attribute, Metric
DataInspector, Wizard, Dashboard	Value Name	Sensor, Sensor Actuator, ValueType
IoT App., Proc.Logic, Node-RED	ServiceURI, SURI	SURI and its real time results of the objects into the data structure

ServiceURI, SURI of a sensor device:

- http://www.disit.org/km4city/resource/METRO759
- http://www.disit.org/km4city/resource/iot/orionCAPELON-UNIFI/CAPELON/Streetlight%3A90FD9FFFFEBD5A7F

ServiceURI, SURI extended with attribute/variable/value:

- http://www.disit.org/km4city/resource/METRO759&metric=vehicleFlow
- http%3A%2F%2Fwww.disit.org%2Fkm4city%2Fresource%2FMETRO759&metric=vehicleFlow
- In some cases
  - **http://www.disit.org/km4city/resource/METRO759/vehicleFlow**



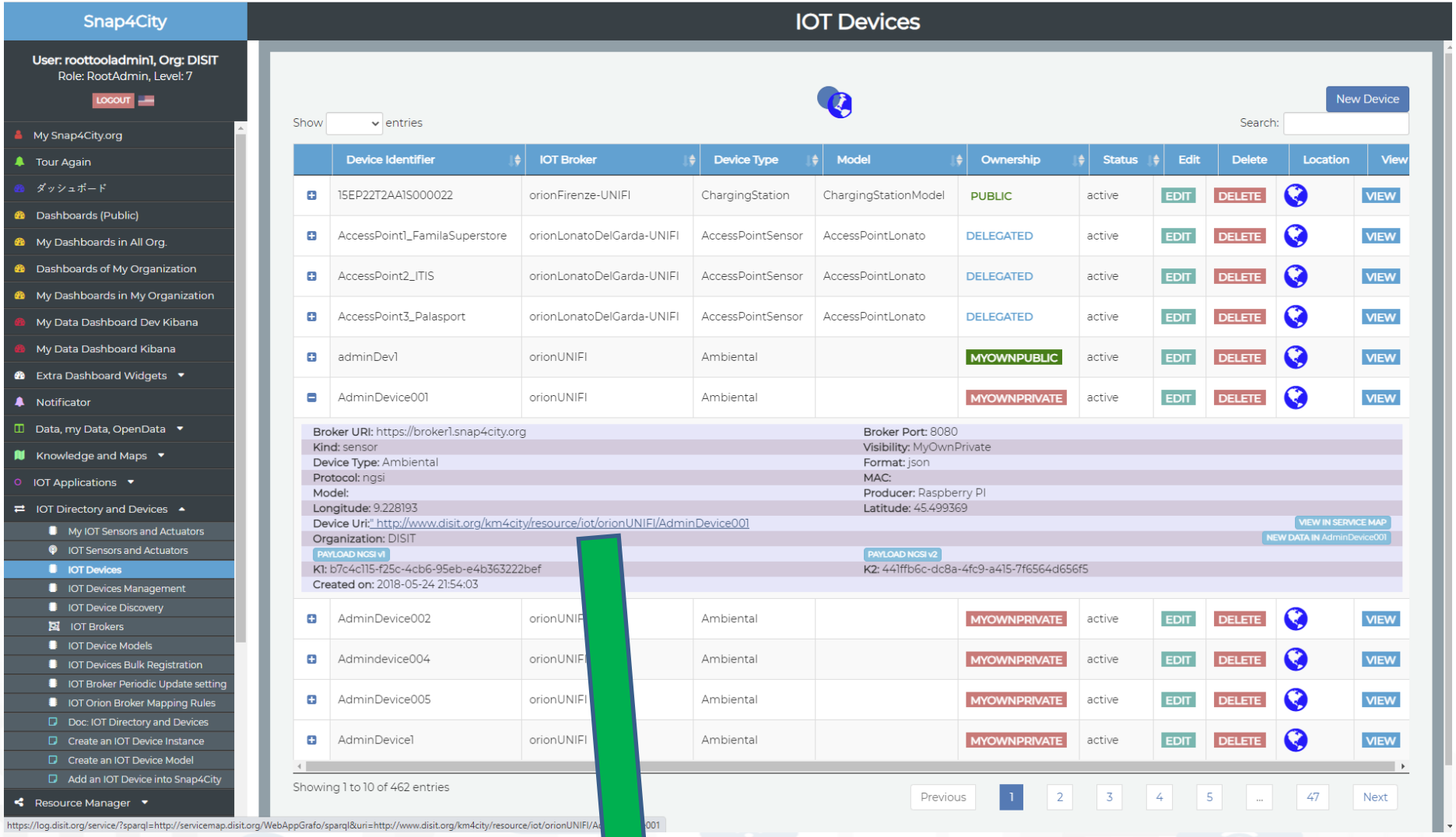
Dashboard Usage and recipe: Event map target

- **Selector to Show on Map a**
  - **category of Map positioned elements**
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.08694333811321;8.791809082031252;44.93758500391093;14.065246582031252&categories=Traffic_sensor&maxResults=0&maxDists=0.1&text=&model=&value_type=&format=json
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?queryId=e5f39066cd68ffe259ed8877bcee222b&format=json
  - **Entity by Model**
    - https://www.disit.org/superservicemap/api/v1?selection=59.36535064975547;13.457822799682619;59.39031474260852;13.566999435424806&model=SmartLightCapelon&format=json
  - **Single Entity**
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/iot/orionFirenze2/Firenze/SHT20lab_new&format=json&fromTime=3-day
  - **Heatmap among many**
    - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=Florence_PM10
  - **Traffic flow**
    - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=FirenzeFIPILITrafficRealtime&trafficflowmanager=true
    - https://firenzetraffic.km4city.org/trafficRTDetails/roads/read.php
  - **Origin Destination Map**
    - https://odmm.snap4city.org/api/get?precision=communes&from_date=&organization=Toscana&inflow=True&longitude=11.255751&latitude=43.769710&od_id=mobile_Toscana_1000&perc=True
- **Events which are also PIN on map**

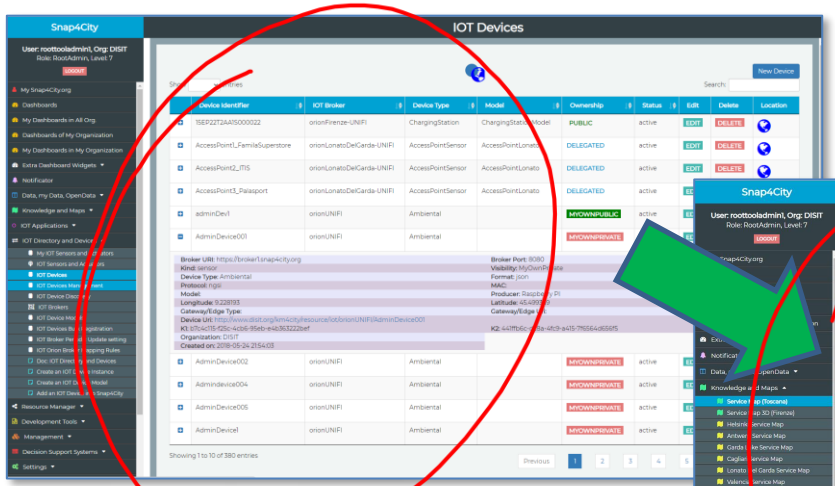
- **Il Service URI as the unique identifier of the Entity**
  - **http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO632**

IoT Directory

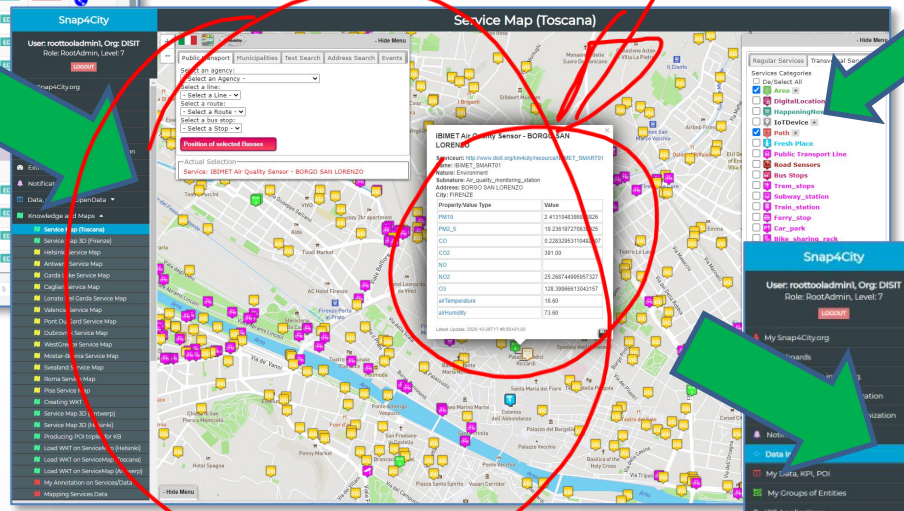
- **For:**
  IOT Devices, Sensors, Sensor mobile, Actuators, Virtual Sensors, etc.
- Accessible as
  - ServiceURI
  - **Device URI**



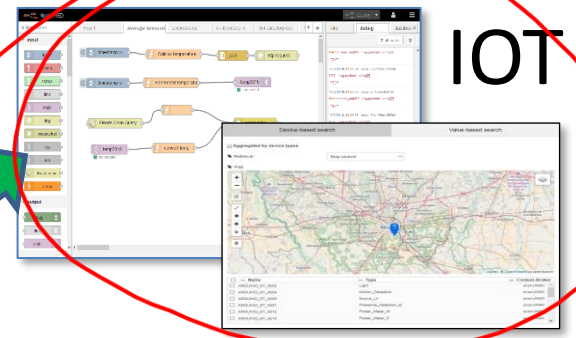
Data Registration Flow at a Glance



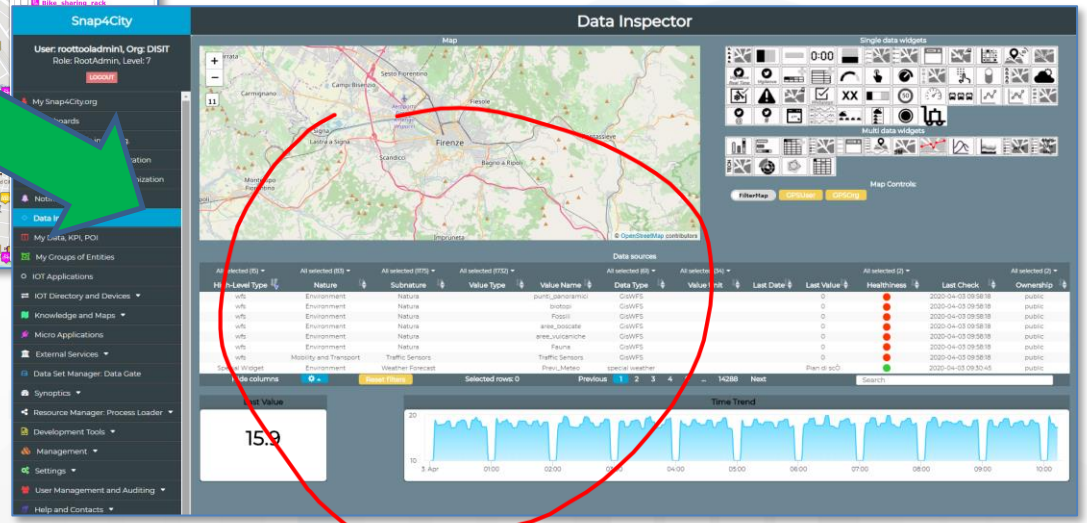
IOT Directory:
Devices…
Sensors..
Actuators…



Knowledge Base,
ServiceMap,
SuperServiceMap
SmartCity API,
ASCAPI



IOT Apps

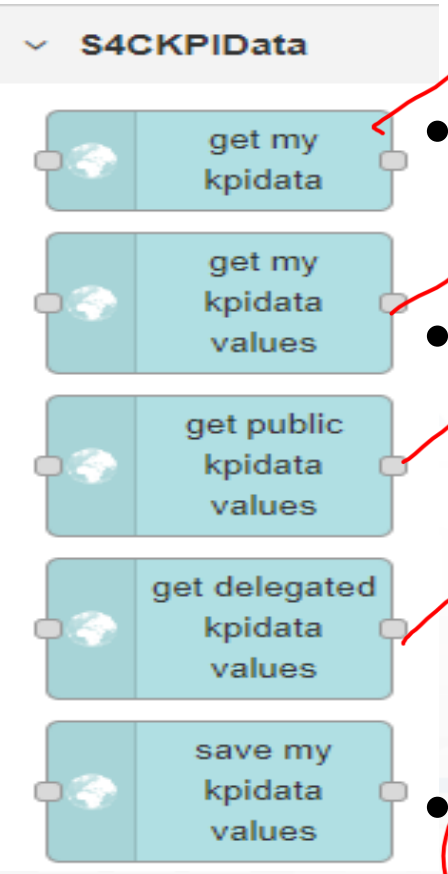


DataInspector
Dashboard Wizard

MyKPI Nodes


Data Management, HLT
Data Inspector
MyKPI, MyData, MyPOI
My Groups of Entities
View/Set MyPOI on Tuscany
Data Table Loader (Excel)

S4C KPI Data



- Save and retrieve MyKPI into the safe personal data storage
- Access to MyKPI and to those that other user have delegated to Me
- **MyKPI are:**
  - Time series of data with GPS coordinates that can chage over time
  - Suitable for: moving sensors, trajectories, data from OBU, data from mobile, sensor data (if needed), etc. etc.
- **MyPOI are:**
  - POI with full metadata description and static coordinates



Interoperability with respect to Hardware staff

Some examples of Hardware Interoperability

- **Any Broker/Gateway** can be connected to Snap4City with any protocol:
  - For example: MQTT, COAP, SNMP, AMQP, OneM2M, LoraWAN, SigFox, etc..
- **Any Device** can be connected.
  - For example: Libelium, Arquino, Modbus, etc.
- **AXIS Cameras** can host
  - Snpa4City plugins and Proc.Logic/IoT Apps
- **Any TV Camera** can be conneced via VMS Milestone

Libelium devices

Smart Environment PRO



libelium

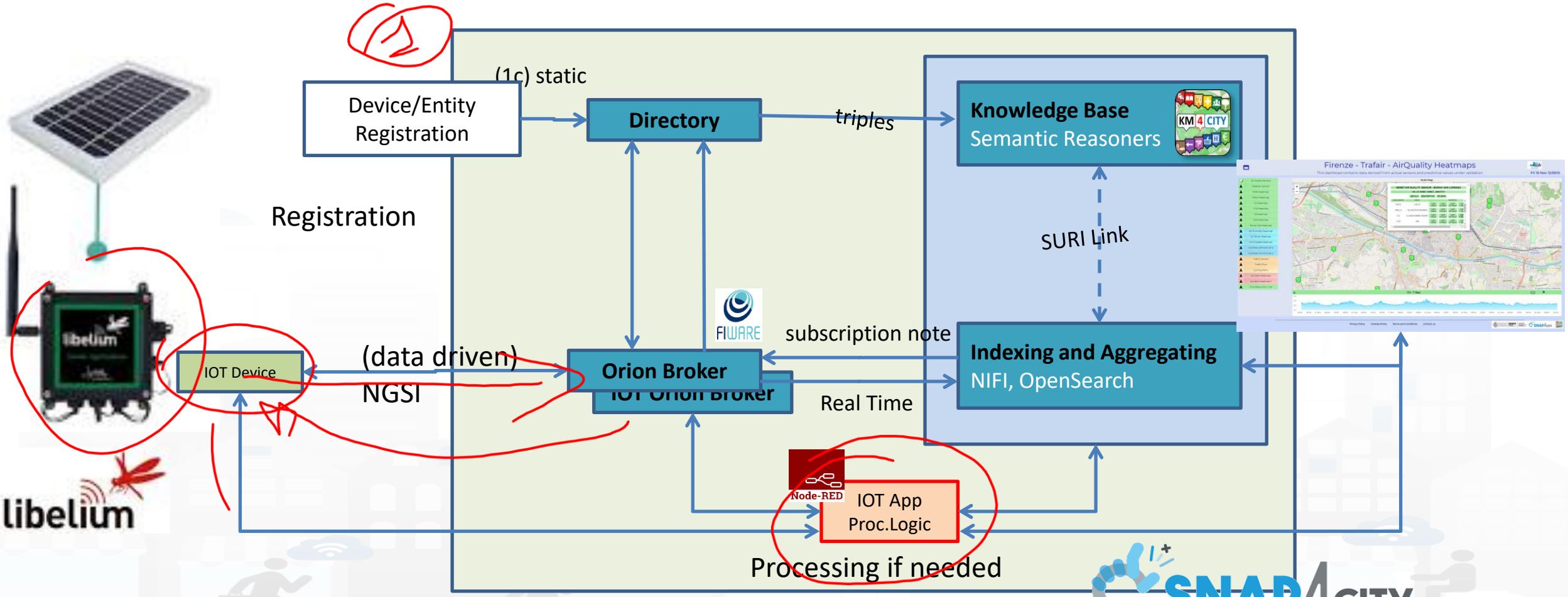




- PM10
- Temp
- Humidity
- Pm2.5
- NO
- NO2
- CO2
- Etc.

https://www.snap4city.org/659 how to set up on Snap4City

- Can be directly connected to Snap4City (data driven)

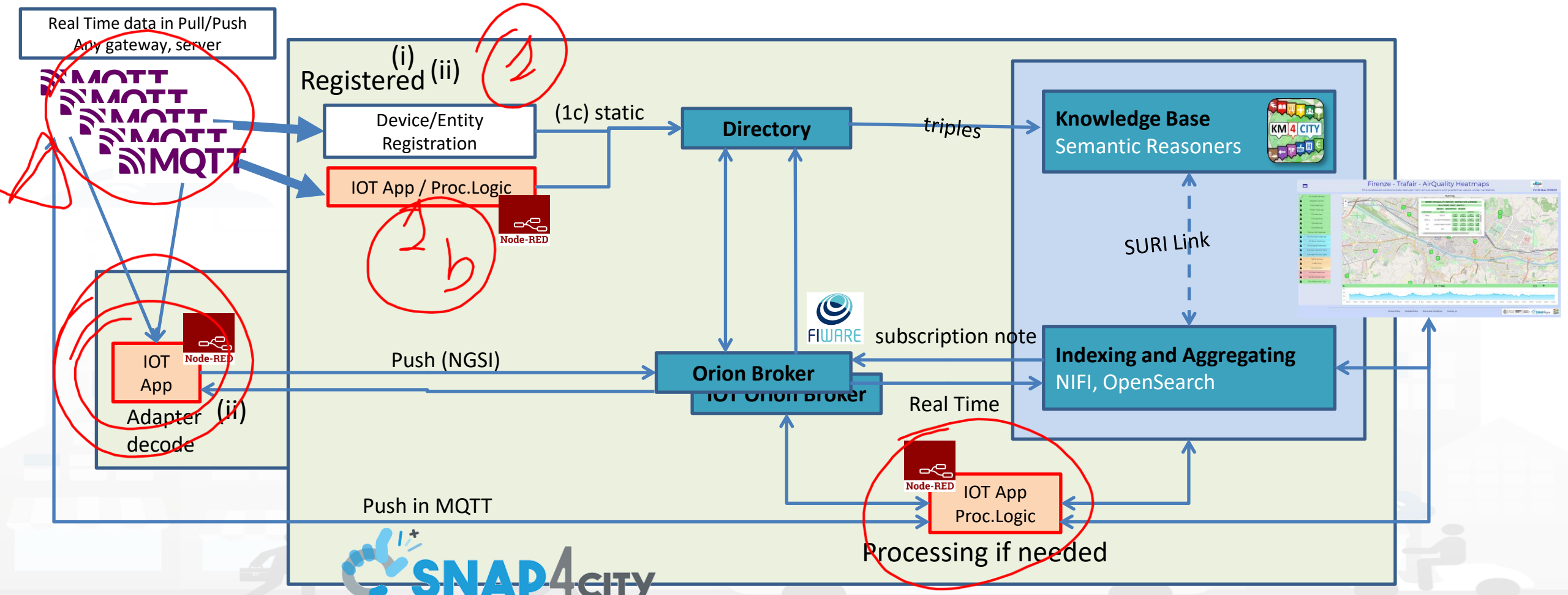


MQTT Integration

mosquitto

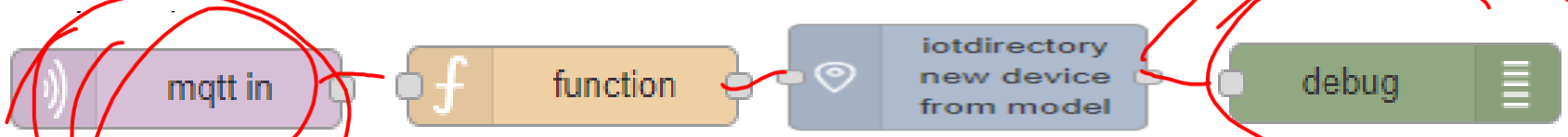


- Can be connected from/to MQTT devices or gateways in push



flows

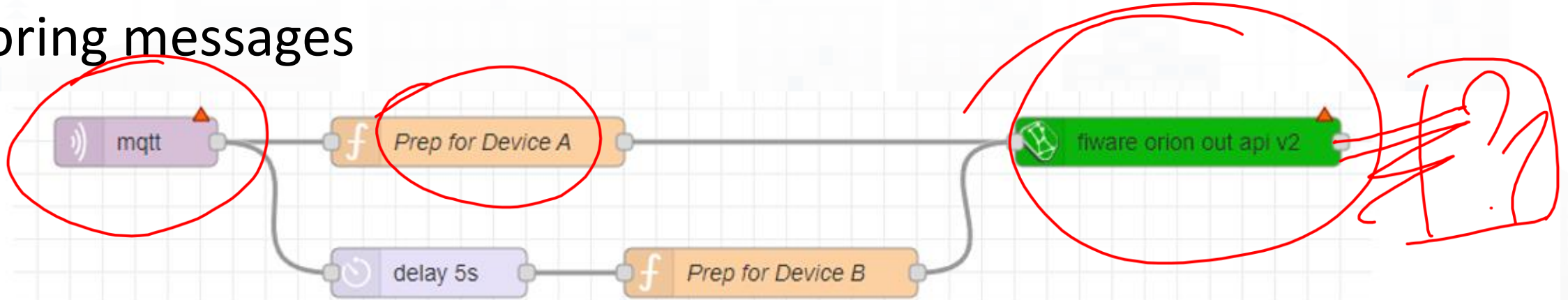
- Registration



- Delegate

change ownership my device	To **change the ownership** of an Entity Instance (IoT Device).	Snap4city
delegate my device	To **delegate a certain Entity Instance** (IoT Device) to some other user for which you have to know the Nickname. Delegations can be: Read access, Read write, Modify (to modify the Entity Instance structure).	Snap4city

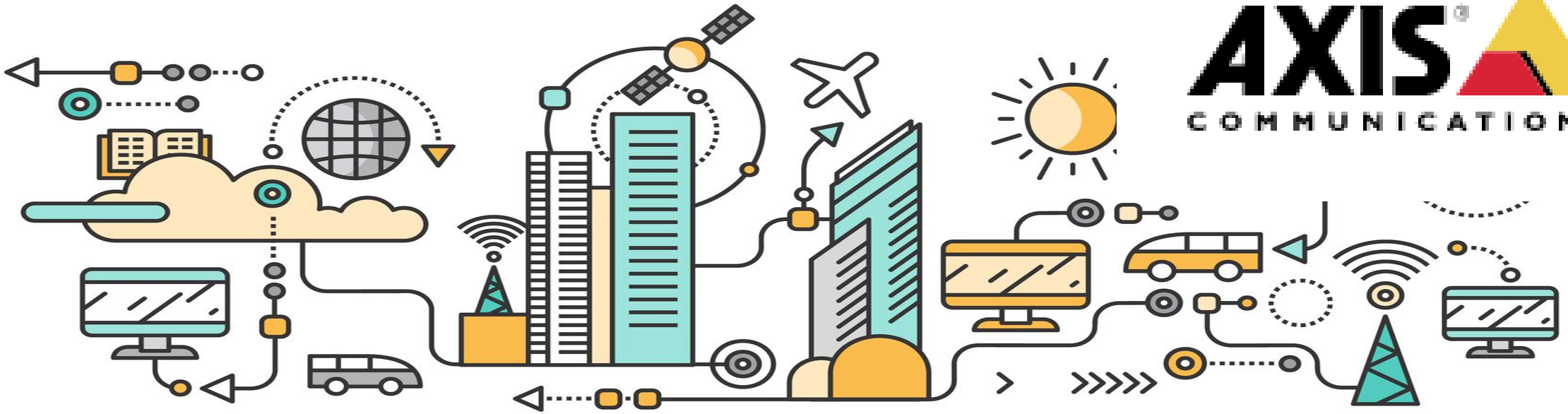
- Storing messages

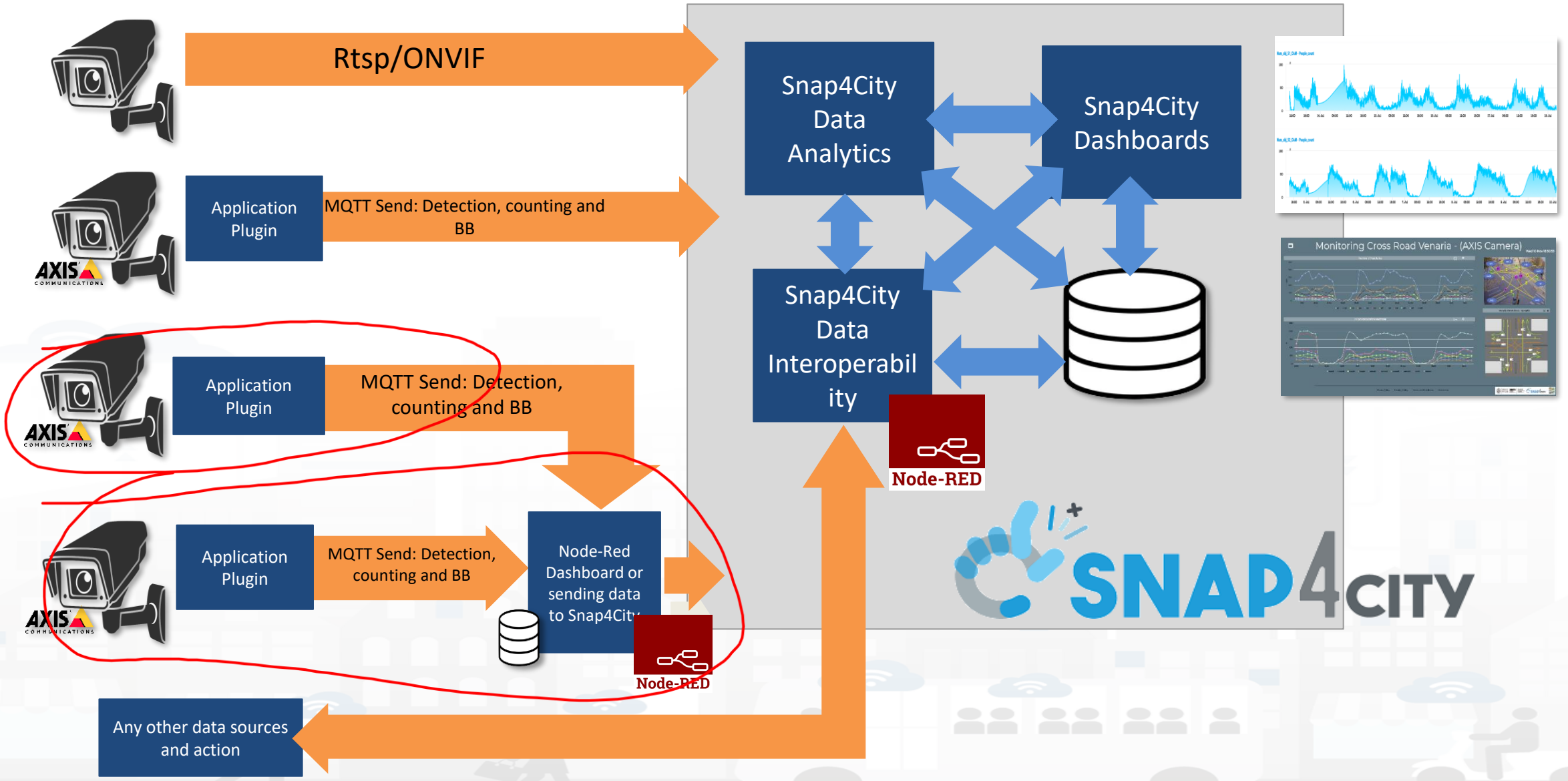




TV Cam AXIS on edge

AXIS COMMUNICATIONS




Rtsp/ONVIF
Application Plugin
MQTT Send: Detection, counting and BB
Application Plugin
MQTT Send: Detection, counting and BB
Application Plugin
MQTT Send: Detection, counting and BB
Node-Red Dashboard or sending data to Snap4City
Node-RED
Snap4City Data Analytics
Snap4City Dashboards
Snap4City Data Interoperability
Node-RED
Monitoring Cross Road Venaria - (AXIS Camera)
SNAP4CITY
Any other data sources and action

Venaria Reale

- **AXIS cameras as IoT Edge (Node-RED)**
  - **Color, Thermal and Radar: security, transport, etc.**
  - Node-RED on board
  - Snap4City Library installed
  - Image processing for trajectories
  - Sending data stream on Snap4City.org
- **Snap4City infrastructure**
  - Collecting data in real time
  - Pre-processing, clustering in real time
  - **Counting in real time**: 12 trajectories, 8 in/out flows
  - Presenting data on dashboard

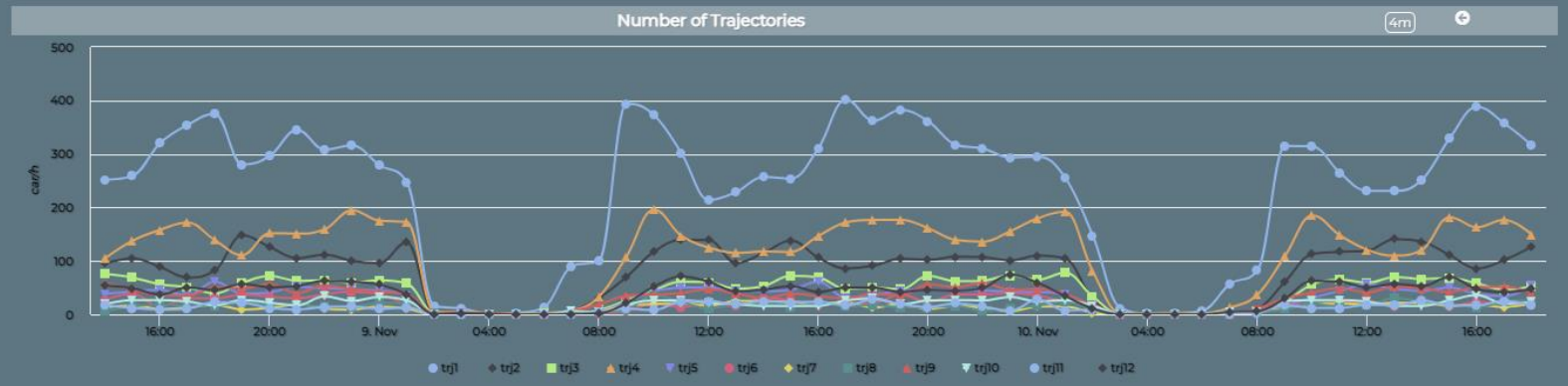


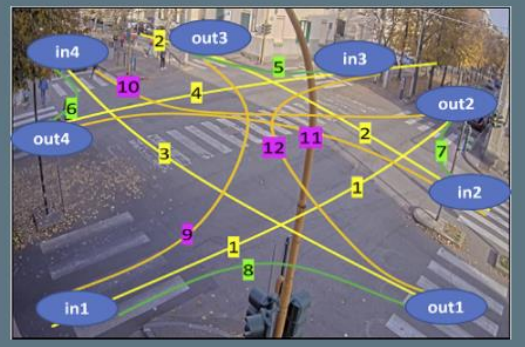
One Camera
Many results

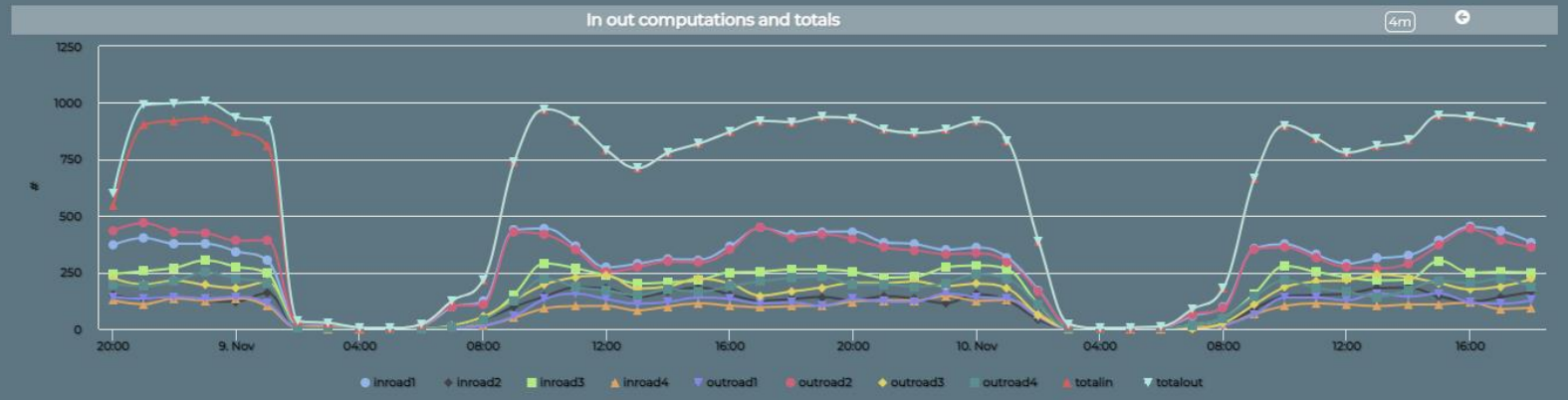
Venaria Reale

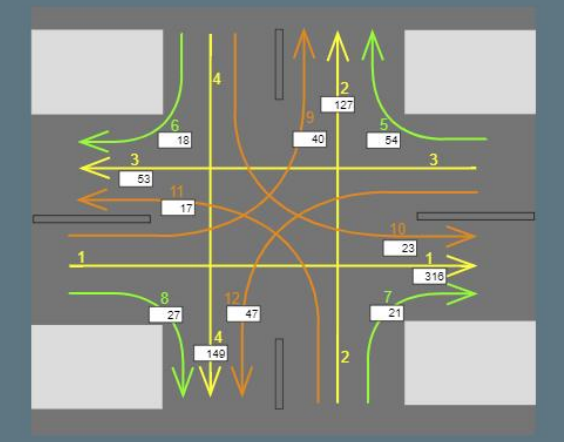
Monitoring Cross Road Venaria - (AXIS Camera)

Wed 10 Nov 18:50:53





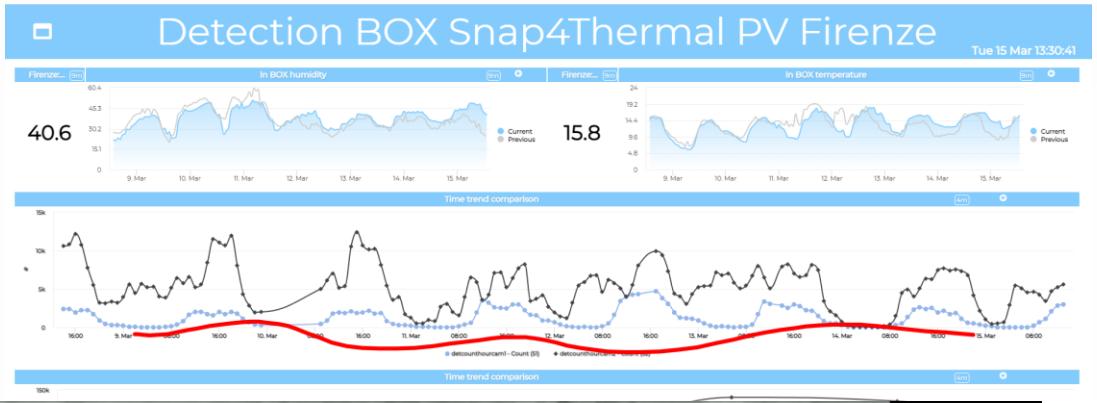


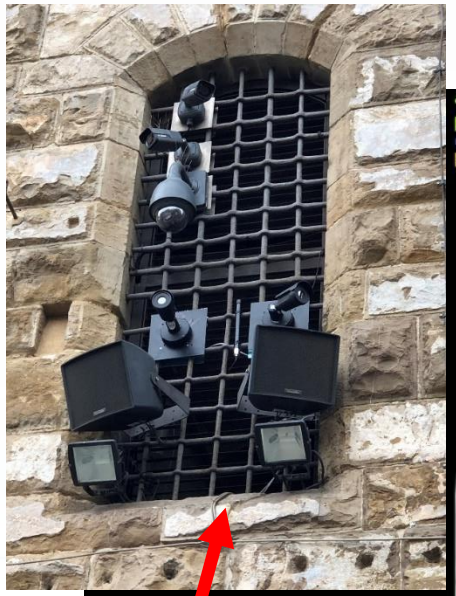


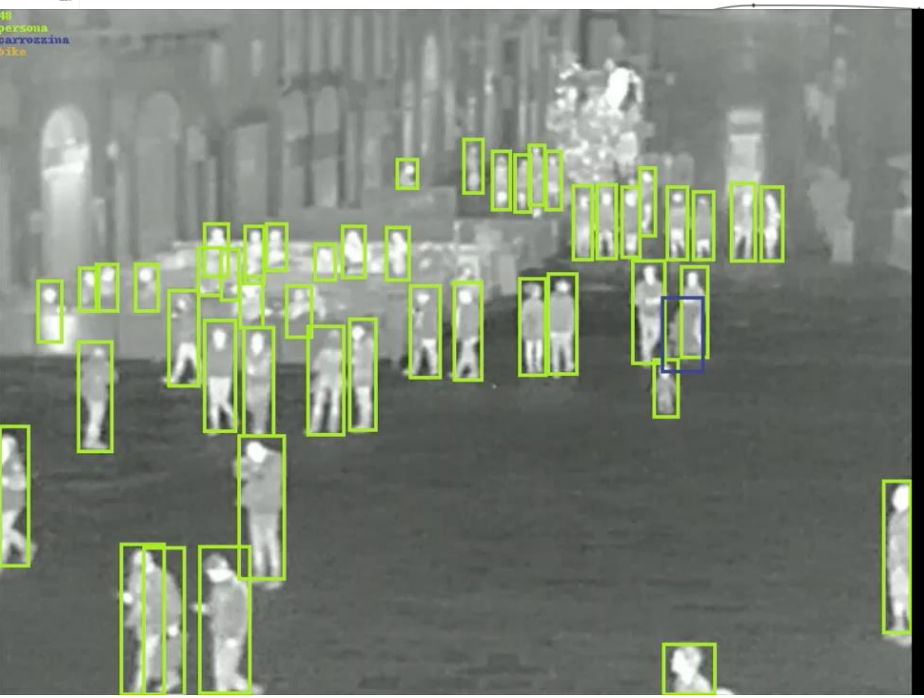
Privacy policy Cookies Policy Terms and Conditions Contact us

https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MzI5Ng==

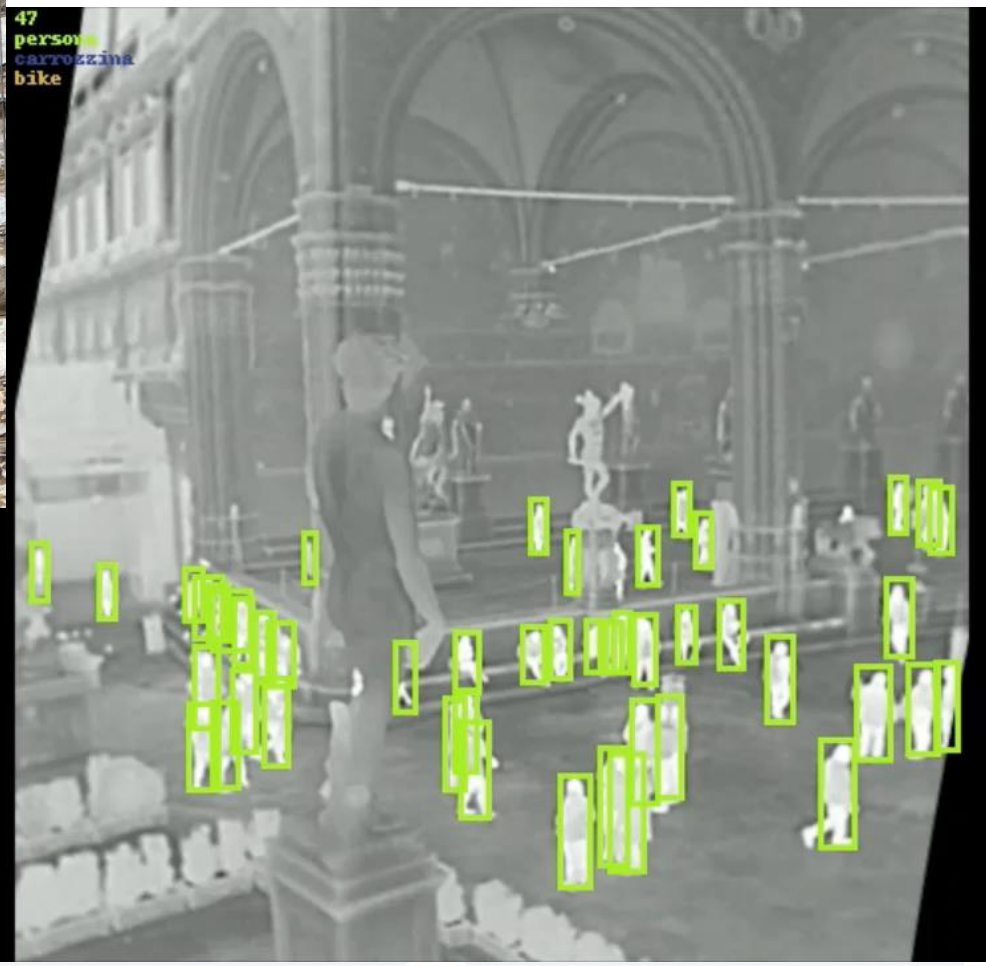
A view and data from the Thermal Camera











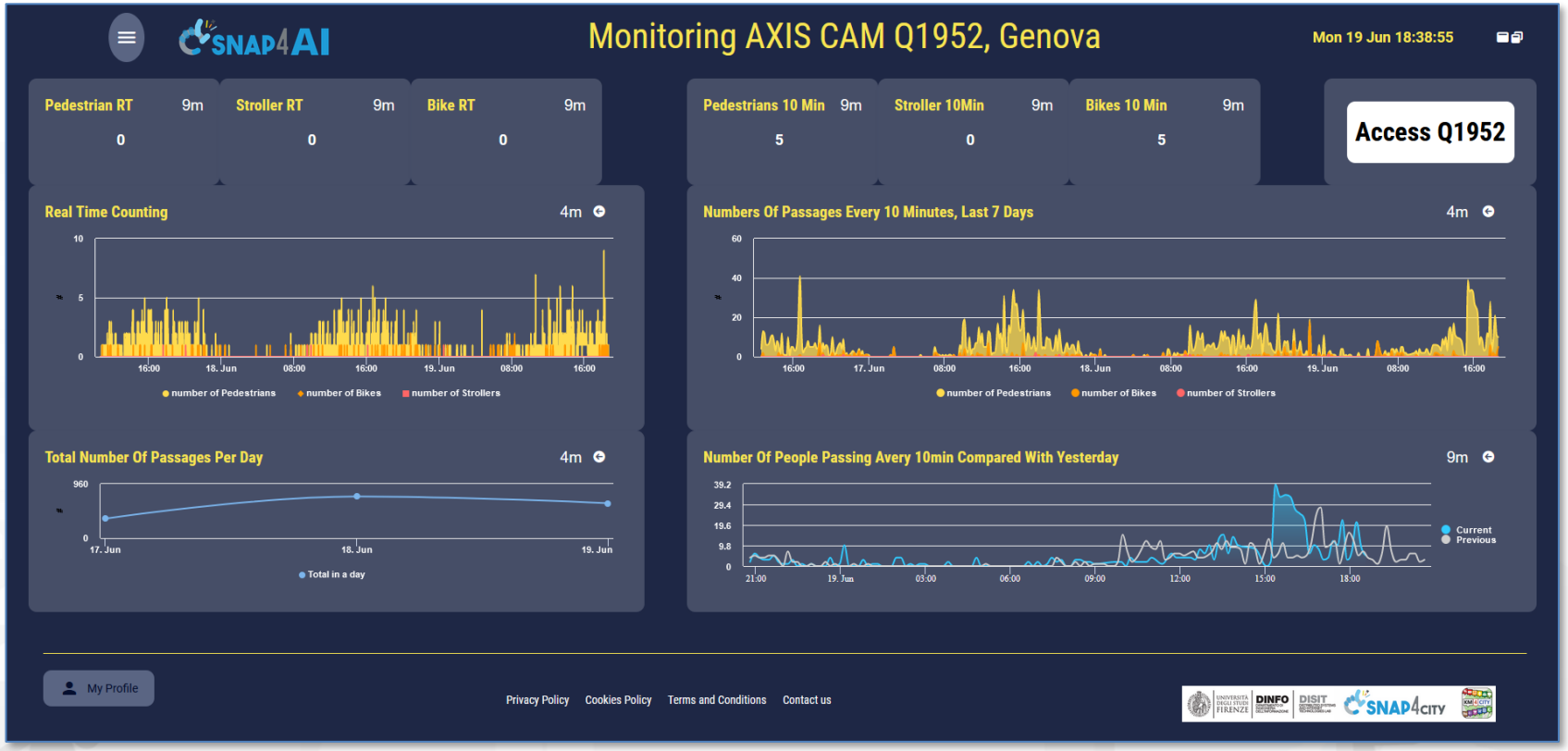




Monitoring Passages AXIS Q1952



- Genova: Ocean Race, 2023



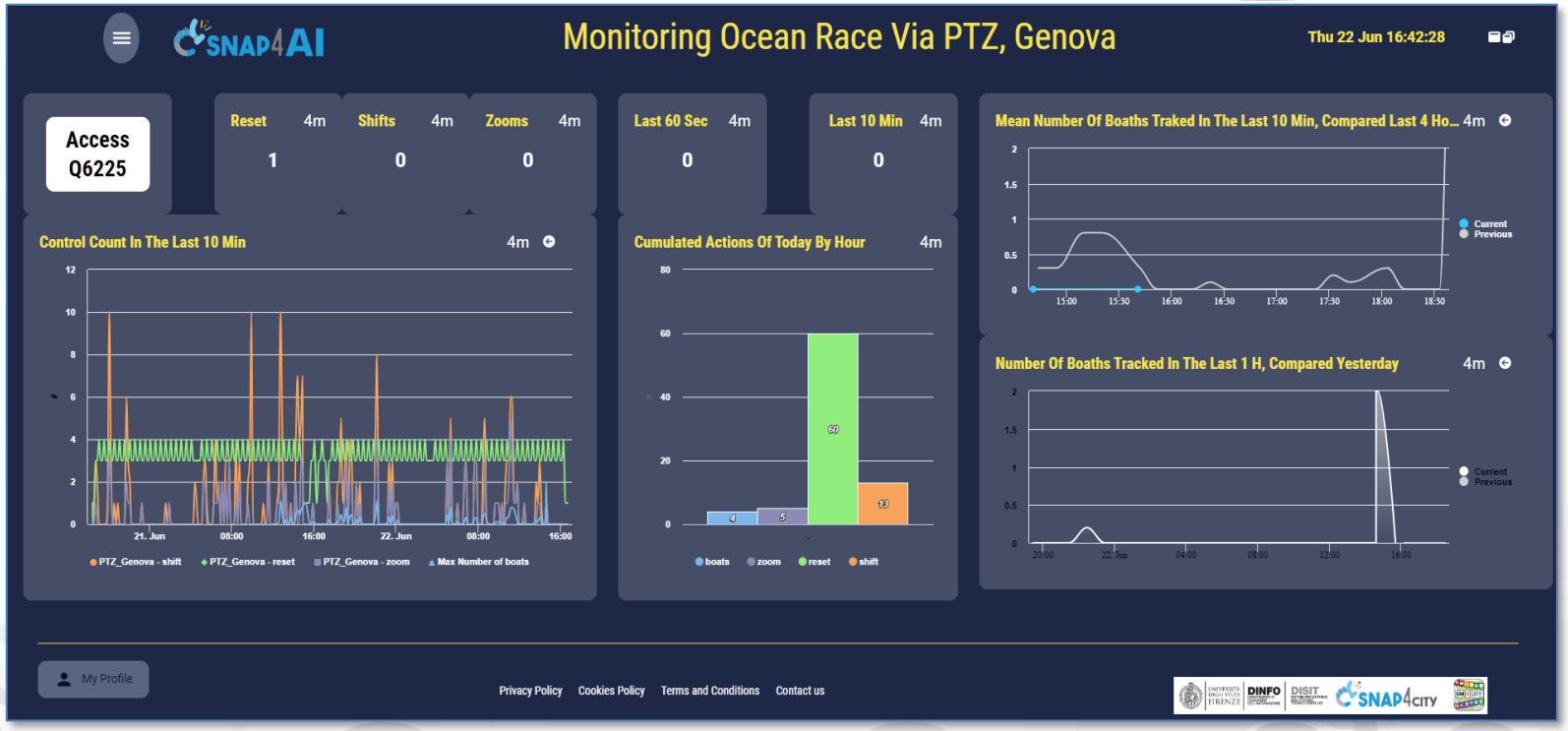
11 SUSTAINABLE CITIES AND COMMUNITIES

Monitoring Boats
AXIS Q6225, PTZ





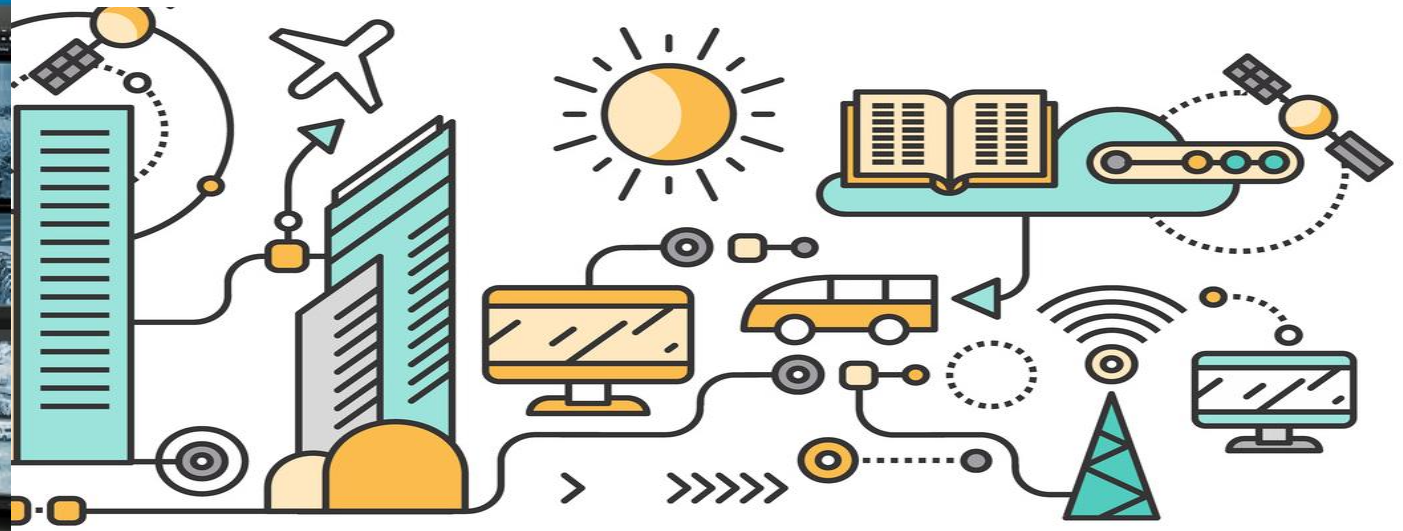
- Genova: Ocean Race, 2023



11 SUSTAINABLE CITIES AND COMMUNITIES

Integration with VMS milestone

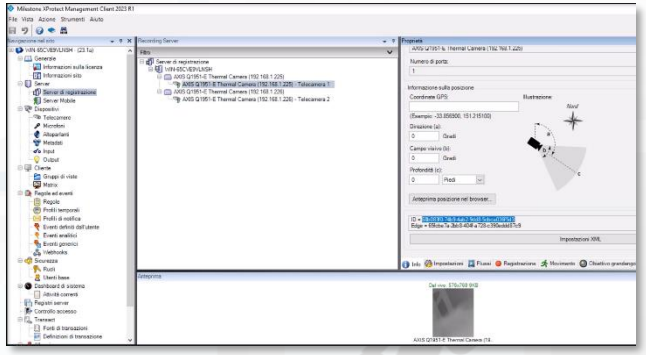


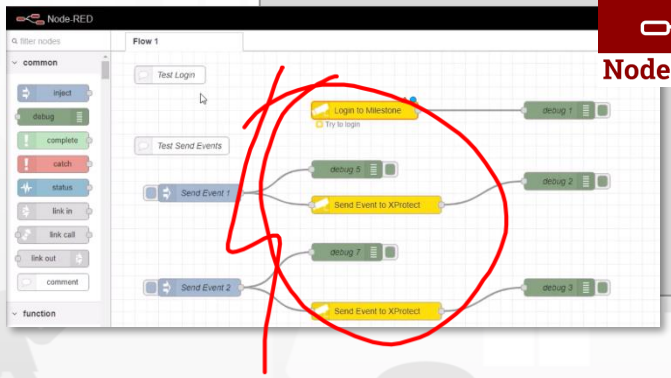


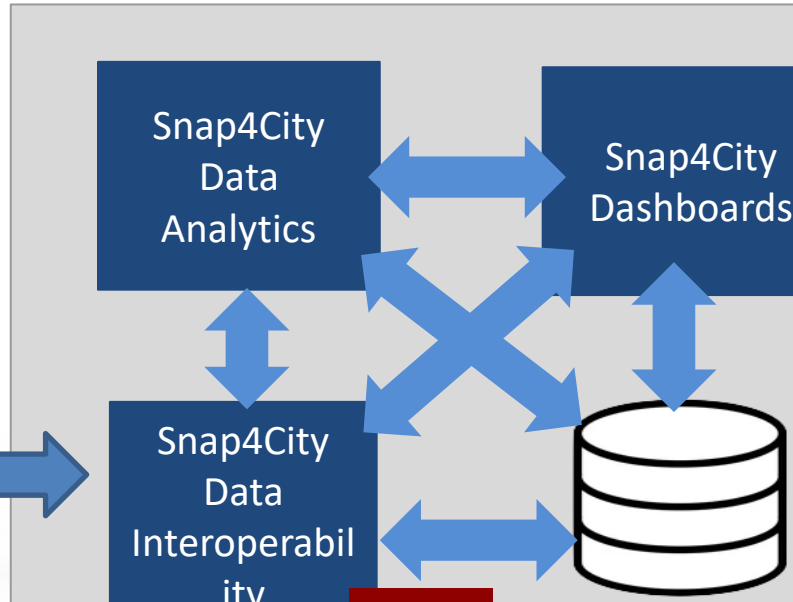
VMS vs Snap4City: sending and getting events, AI solutions

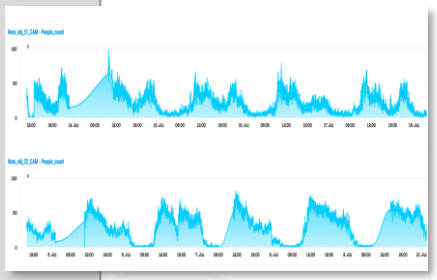


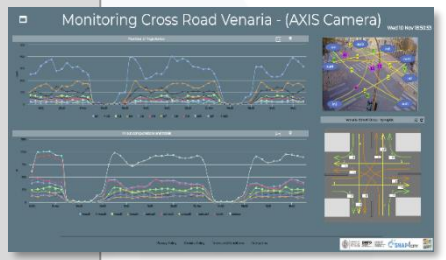
Events + info









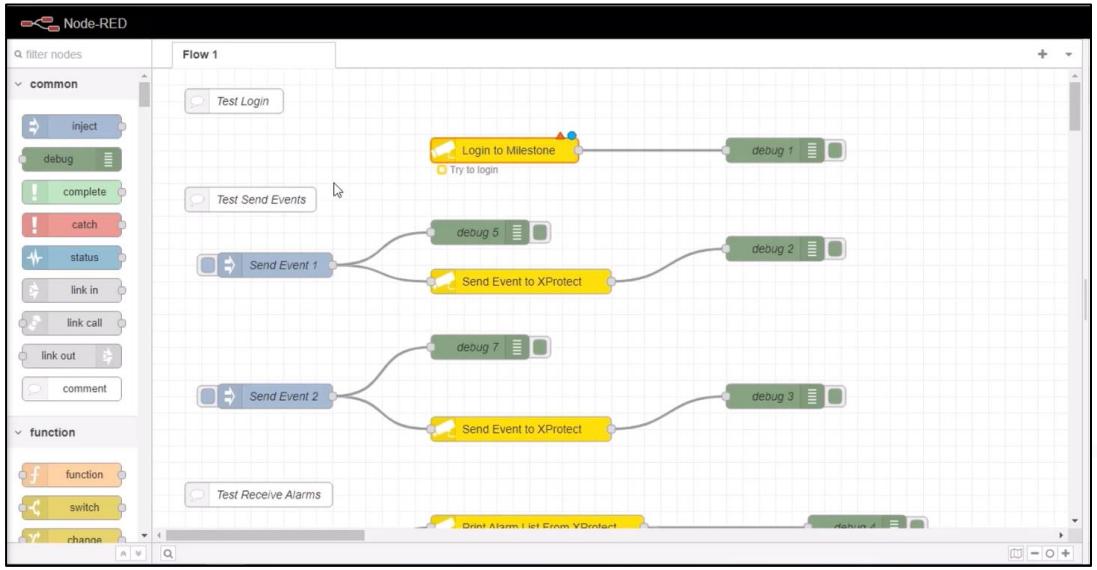


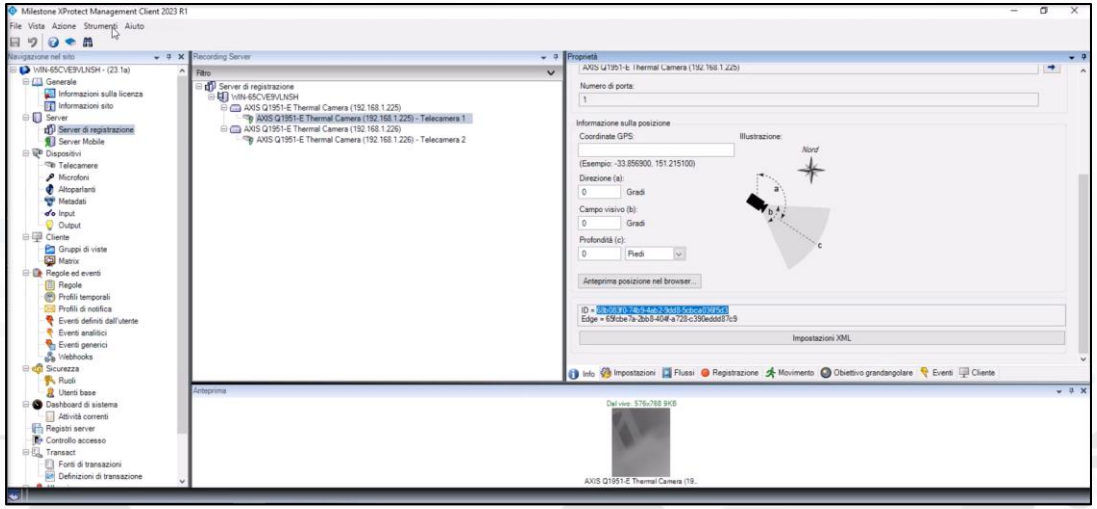
Node-RED

SNAP4CITY

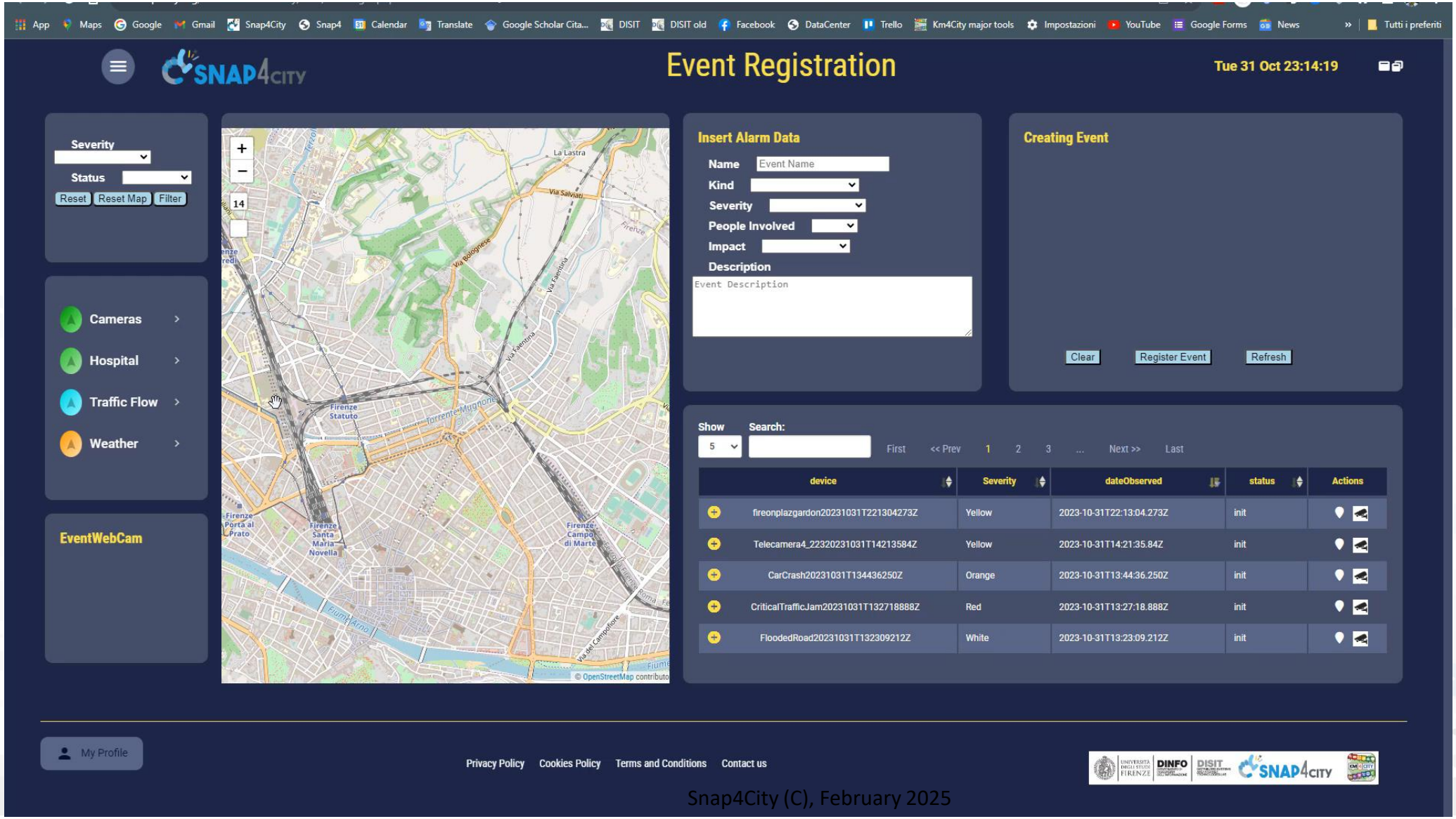
Snap4City ←→ Milestone Integration

- Snap4City VMS Library on Node-RED
- Functionalities:
  - Registering IoT App/Proc.Logic on VMS Milestone
  - Receving event of VMS into Snap4City platform via Node-RED, on cloud or on premise
  - Sending Snap4City Events into VMS Milestone

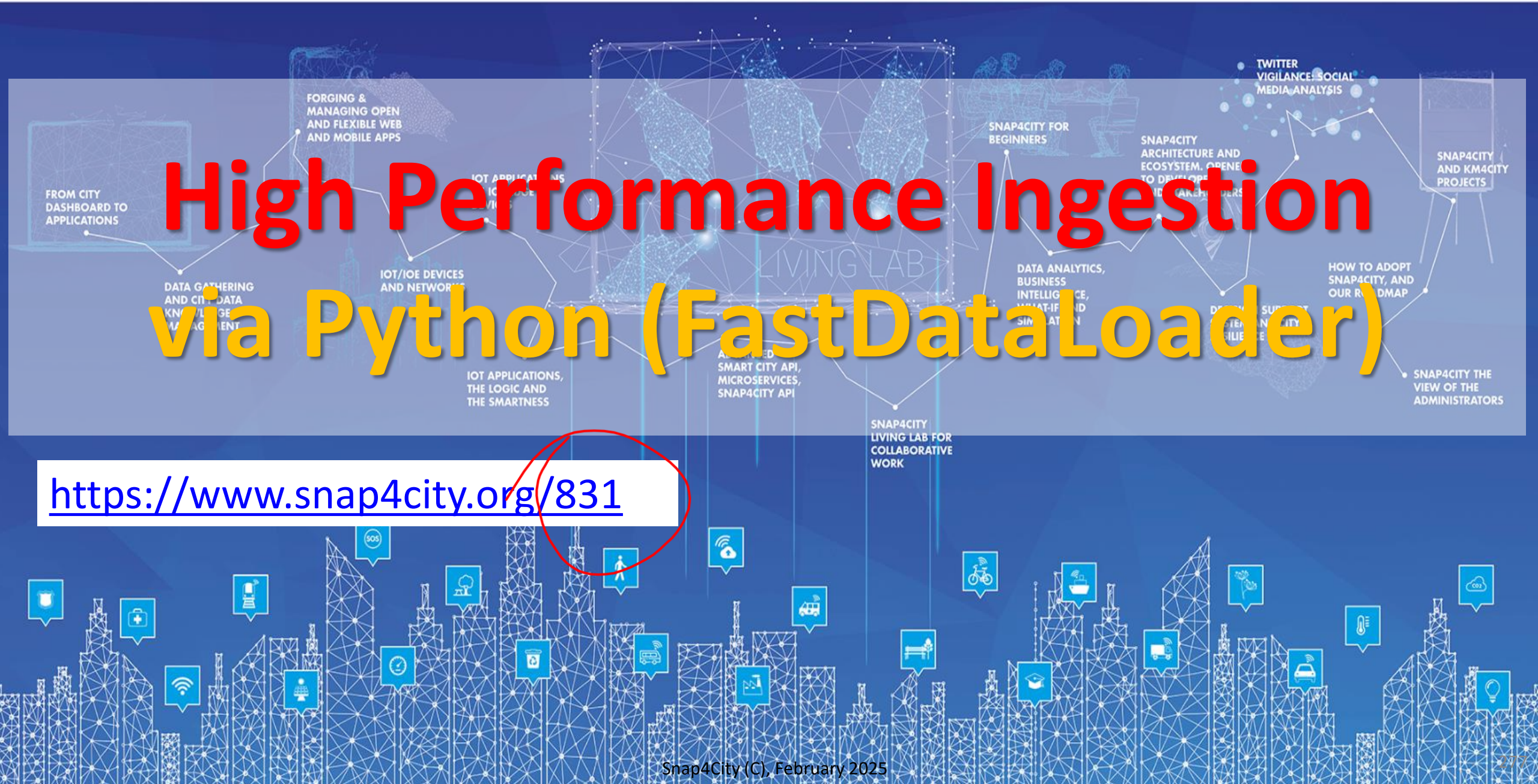




Event Management



High Performance Ingestion via Python (FastDataLoader)



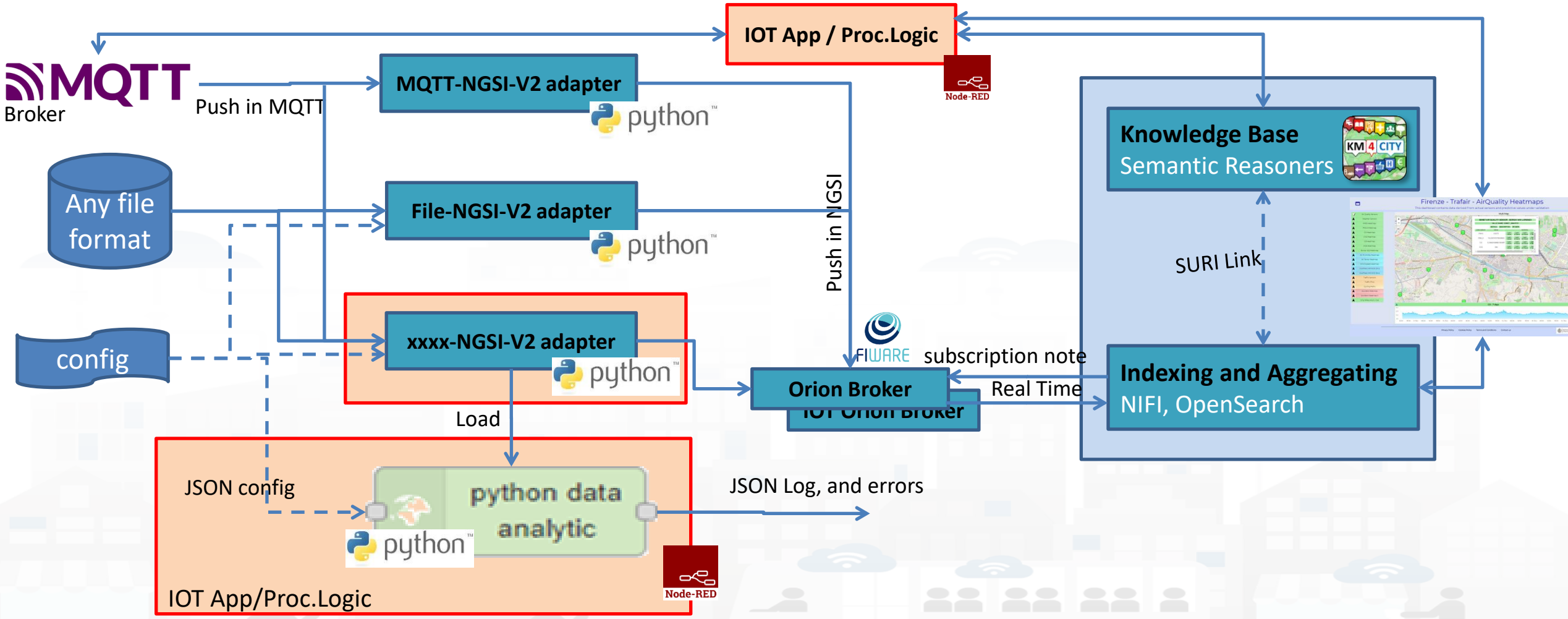
https://www.snap4city.org/831

Fast and Massive Data Ingestion

- The ingestion processes passes data to some Orion Broker when the device is registered on Directory.
  - The ingestion in this case consists in posting each single message on the broker and the broker posts them on NIFI which performs regularization exploiting KB and passes them on Open Search
  - The process can be performed by IoT App, thus in Node-RED with high flexibility (95% of cases is ok) but with some limitation on throughput
- **Faster approaches** is based on **a Python process** to get data and:
  - **Case A)** post each single message on one of the Orion Brokers
  - Case B) post each single message directly on Ni-Fi cluster (not suggested but viable)
  - For Both: *you have to register the IoT devices/Entities on Directory*

Case A)

High Performance Solutions on Snap4City Framework



Interoperability of Snap4City Platform



Snap4City
Authentication Interoperability



Authentication and SSO

- **Authentication in Snap4Tech is based on KeyCloak which is based on SAML, https://auth0.com/blog/how-saml-authentication-works/**
- **Different Versions of interoperability Authentication and Single Sign On, SSO**, are available on demand, with
  - **Spid**, Public Digital Identity System, https://www.spid.gov.it/en/
  - **EIDAS** (electronic IDentification Authentication and Signature), http://www.agid.gov.it/en/platforms/eidas, https://digital-strategy.ec.europa.eu/en/policies/eidas-regulation
  - **CIE, Electronic Identity Card https://ec.europa.eu/info/strategy/priorities-2019-2024/europe-fit-digital-age/european-digital-identity_en**
  - **RealMe** NZ, https://www.realme.govt.nz/

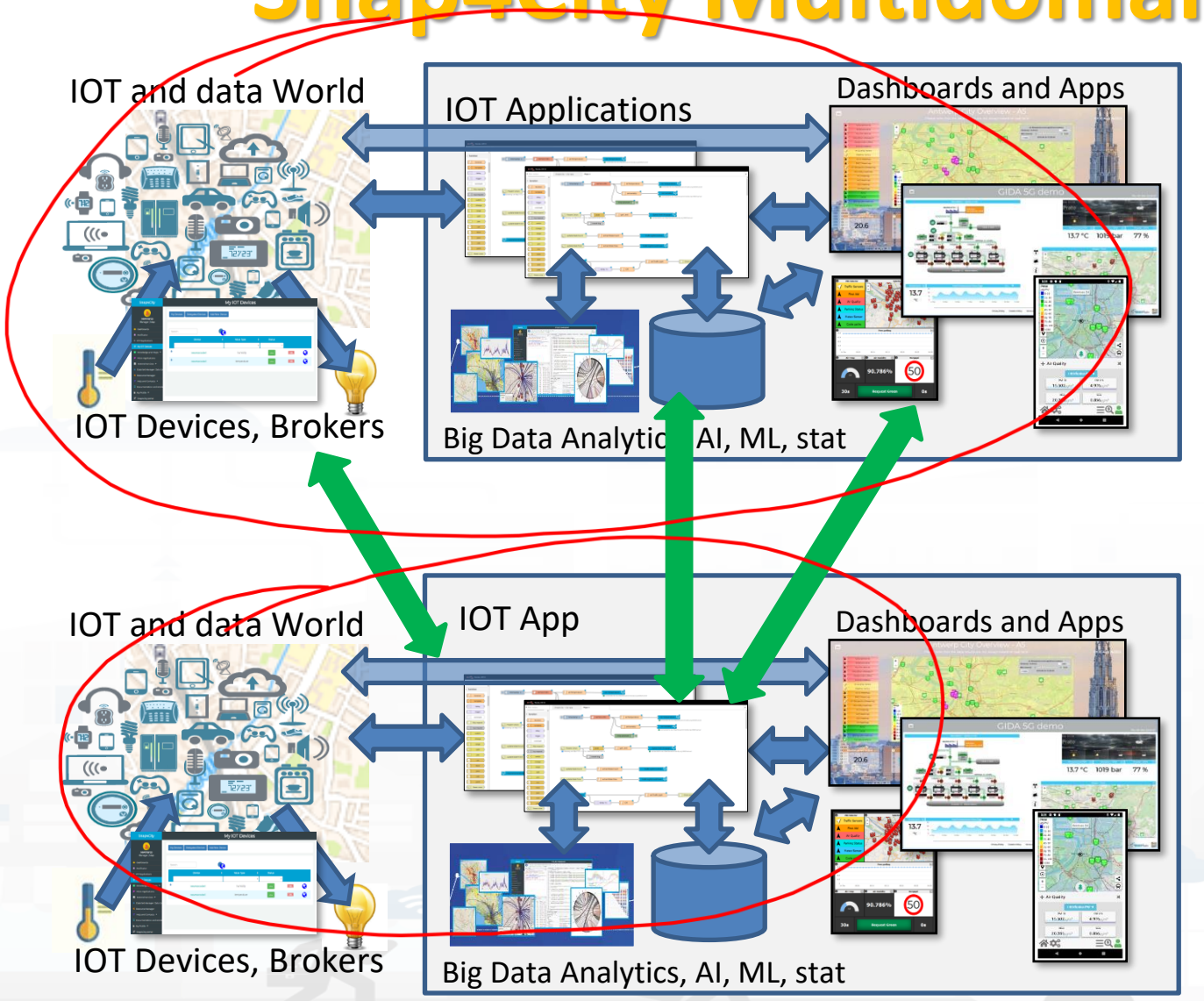
Proc.Logic/IoT App working on multiple Snap4City Platforms



Distributed Computing

- The Snap4City Libraries on Node-RED support the management of Multiple Snap4City Platforms Installations
- It is possible to:
  - Have in different Blocks/nodes, different registrations to different Snap4City Installations/platforms or Users
  - Get/Send data from/to a Snap4City Installations/Users and send/get to/from another
  - Have Multiple Brokers on multiple installations and users
  - Creating collaborative distributed processing that work and share data and processing in multiple platforms based on Snap4City or different.

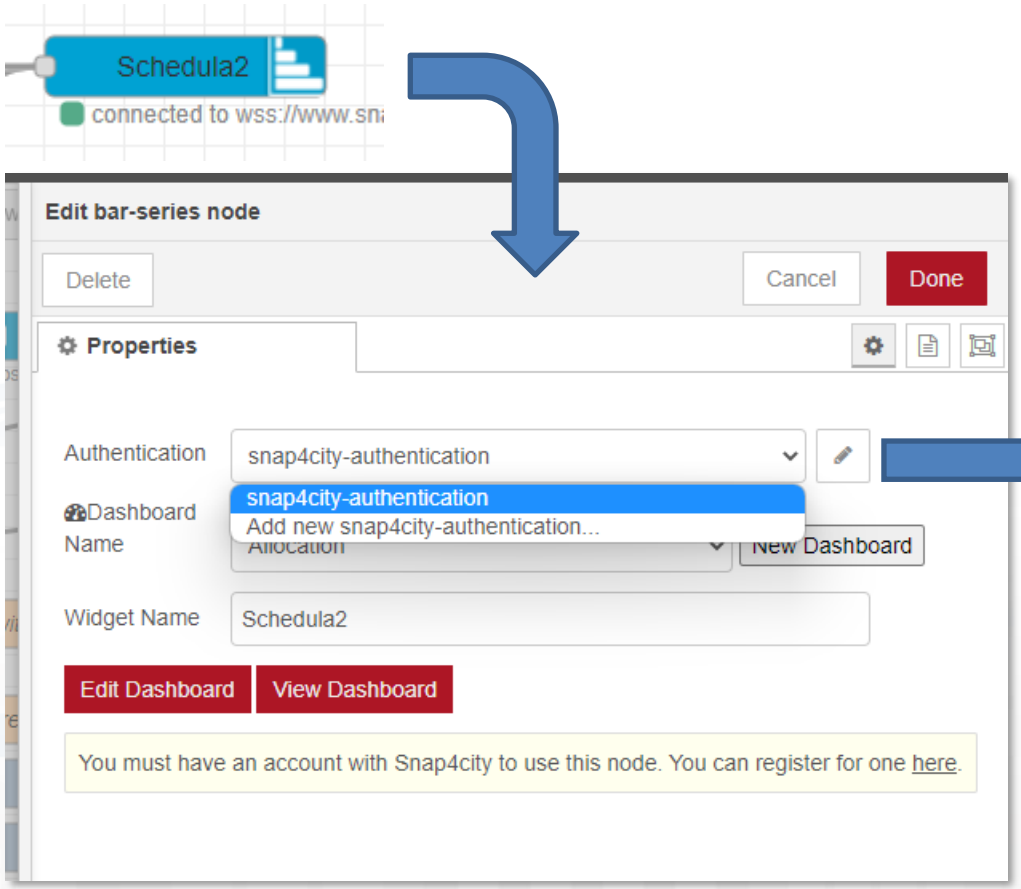
Snap4City Multidomain Applications

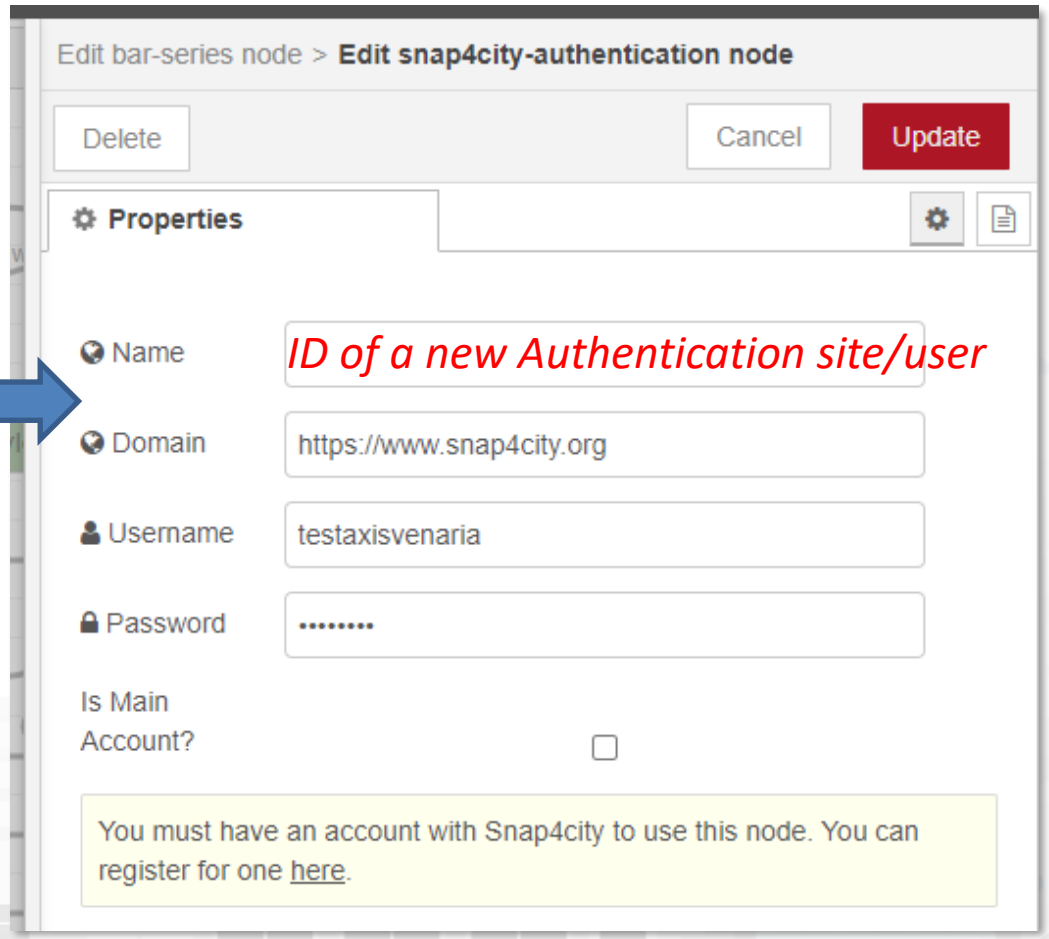


Any Snap4City Installation
Different domain
Different user
Different auth./authoriz. System
Etc..

Any Snap4City Installation
Different domain
Different user
Different auth./authoriz. System
Etc..

Example on Controlling Dashboards multiple domains





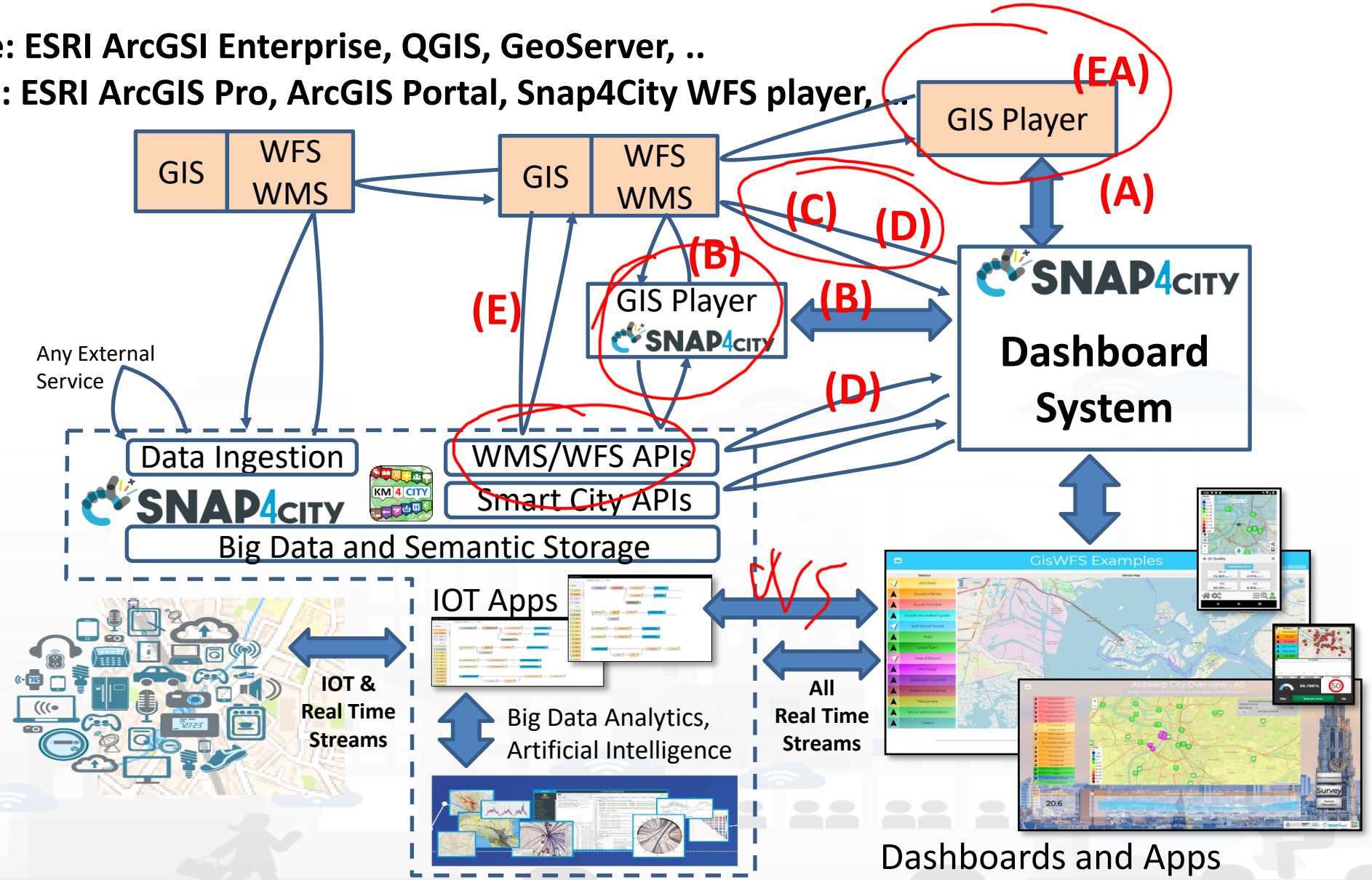
GIS Data Import and Export: WFS and WMS



https://www.snap4city.org/drupal/node/368

GIS vs Sna4City

GIS Server can be: ESRI ArcGSI Enterprise, QGIS, GeoServer, ..
GIS Player can be: ESRI ArcGIS Pro, ArcGIS Portal, Snap4City WFS player, ...



- **GIS:**
  - Geographic Information System
- **WMS:**
  - Web Map Service
- **WFS:**
  - Web Feature Services

Ingesting Service

Transport Offers



Public Transport Information/file/streams

- **Other sources as ODM and POI: parking, sharing, etc.**
- **Models used for**: busses, train, ferry, metro, tramways, etc.
- **Include**:
  - Public Transport Lines, Rides with paths and timeline, stops, polylines for paths, etc.
  - real time data about the position of the vehicles: train, busses, etc.
  - Multi operator data
- **Information is modelled as**
  - **GTSF** format: multiple files in XML
  - **Transmodel** format
  - **Netex** format
- **GTSF files can be ingested on Snap4City via**
  - **Python** which takes GTFS files and convert them in triples «.n3» file for the Knowledge Base
    - https://github.com/disit/smart-city-etl/tree/master/TrasformazioneTPLBus_new_model/Triplification/Models
    - Former version: https://www.snap4city.org/download/snap4cityETL/TPL_bus_gtfs/
  - **GTFS RT can be ingested via IoT App and sent to the Broker**
  - **Chouette** and then
    - using a Python developed by **Snap4City to converter** to produce Triples for the Knowledge Base, service map
    - https://github.com/disit/snap4city/blob/master/Snap4CityGTFS/chouette-gtfs-n3.py
- **Transmodel (EN12896) or Neptune files can be ingested in Snap4City via**
  - **Chouette** and then, with a certain level of adaptation,
    - using a Python developed by **Snap4City to converter** to produce Triples for the Knowledge Base, service map
    - https://github.com/disit/snap4city/blob/master/Snap4CityGTFS/chouette-gtfs-n3.py

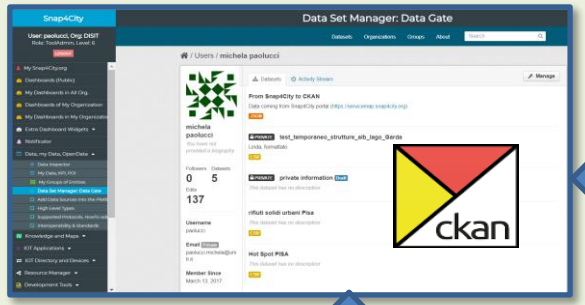
Integration with CKAN
Open Data Manager and Portal



Snap4City vs CKAN

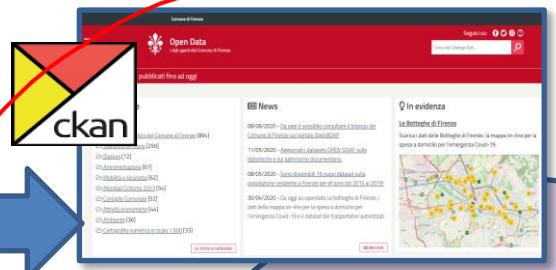
Snap4City Portal and Integrated tools

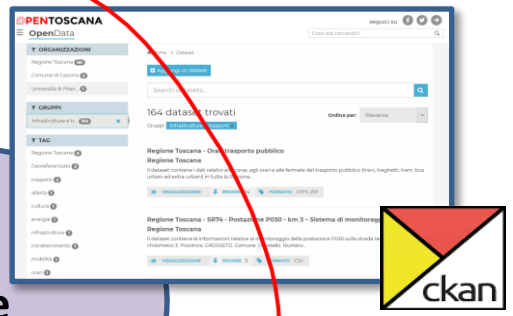
Datagate



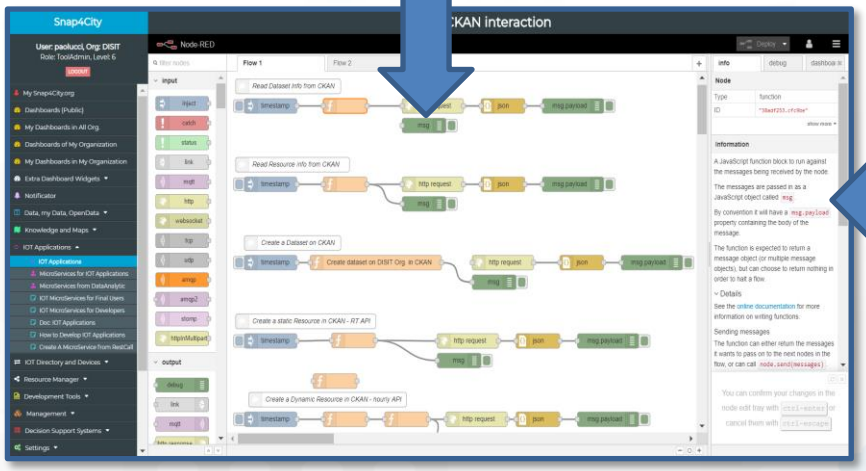
Advanced Snap4City APIs and Micro Services

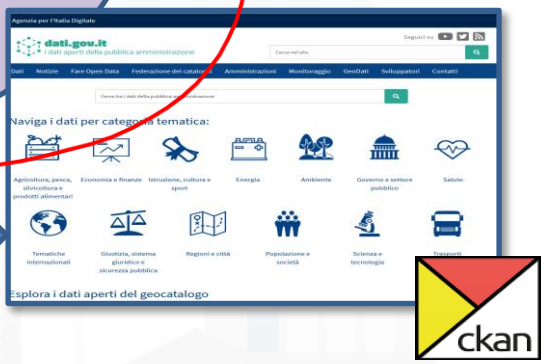
Harvesting and Publishing





Open or Private External CKAN Data Portals

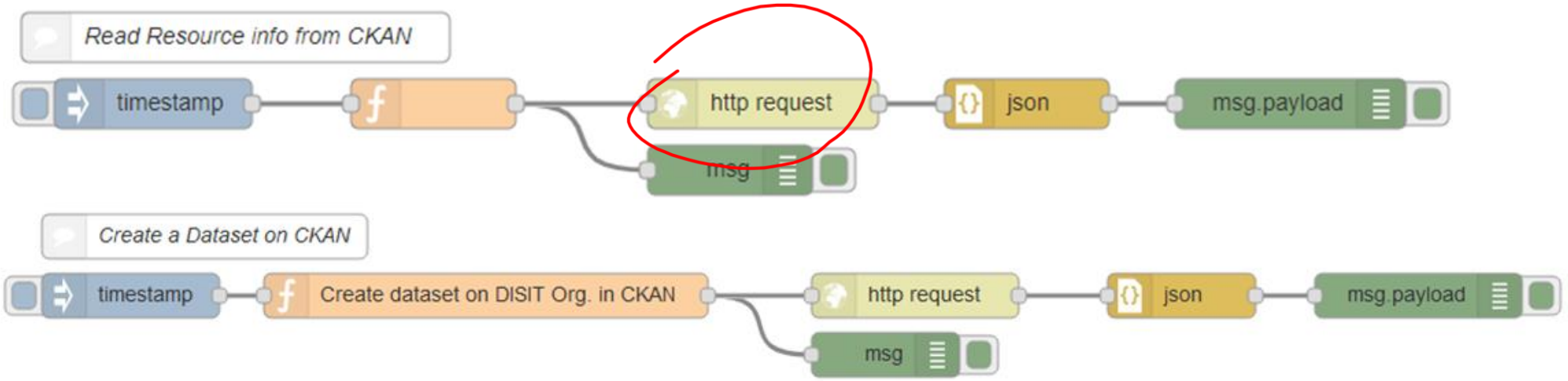




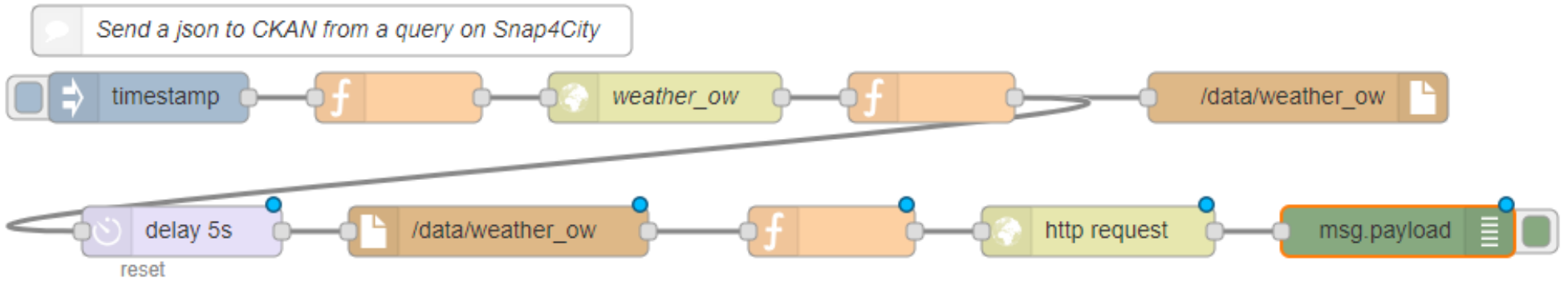
Automatize:

- Import data from CKAN to Snap4City
- Upload Public Data from Snap4City to CKAN
- Data Harvesting
- Dashboards and Mobile/Web Apps creation

Some IOT App segements



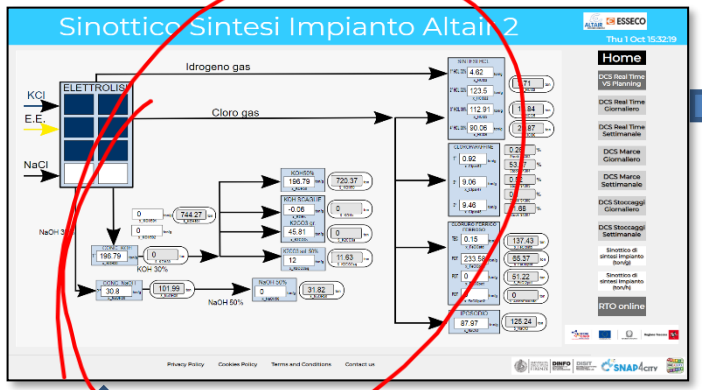
Almost all the calls to CKAN are quite similar



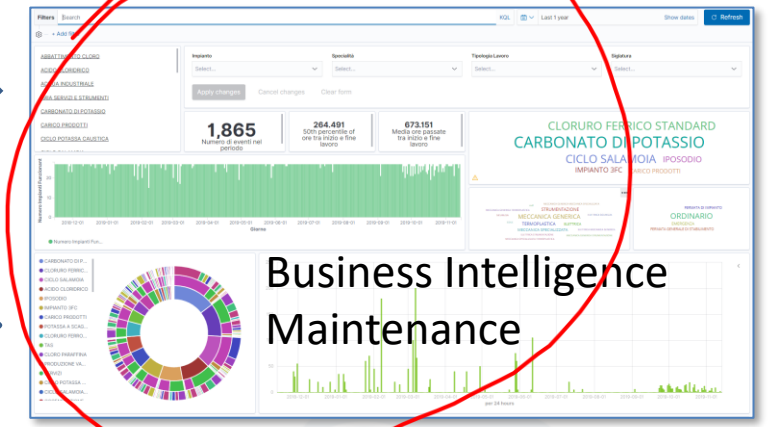
Integration with Open Maint for workflow and ticketing



Example of Integrated workflow



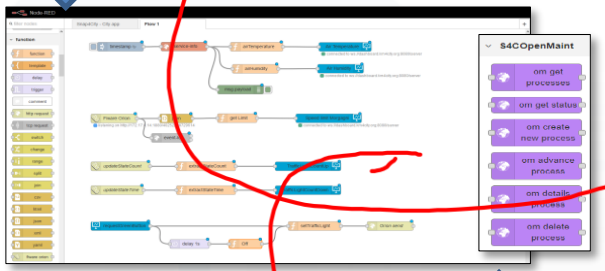
Consumptions/productions

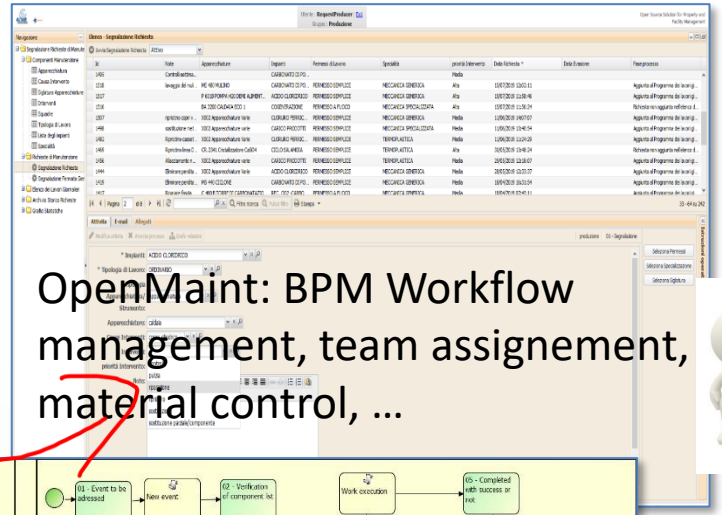


Events/actions

Business Intelligence
Maintenance

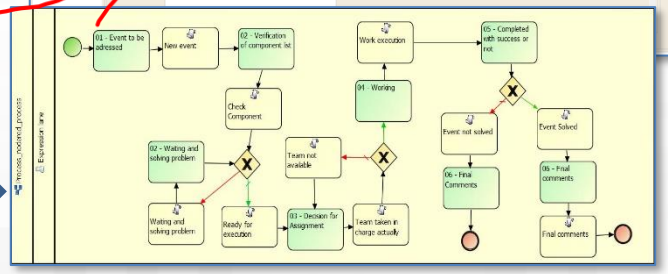
Dashboards and actions

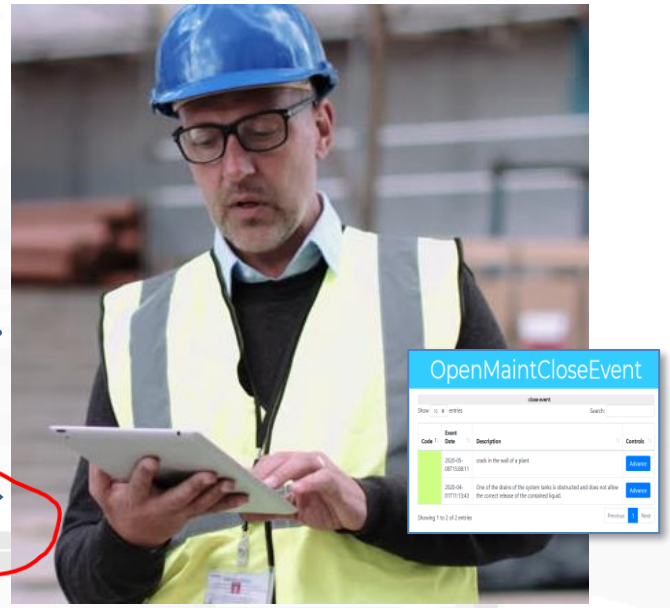




OpenMaint: BPM Workflow management, team assignement, material control, …







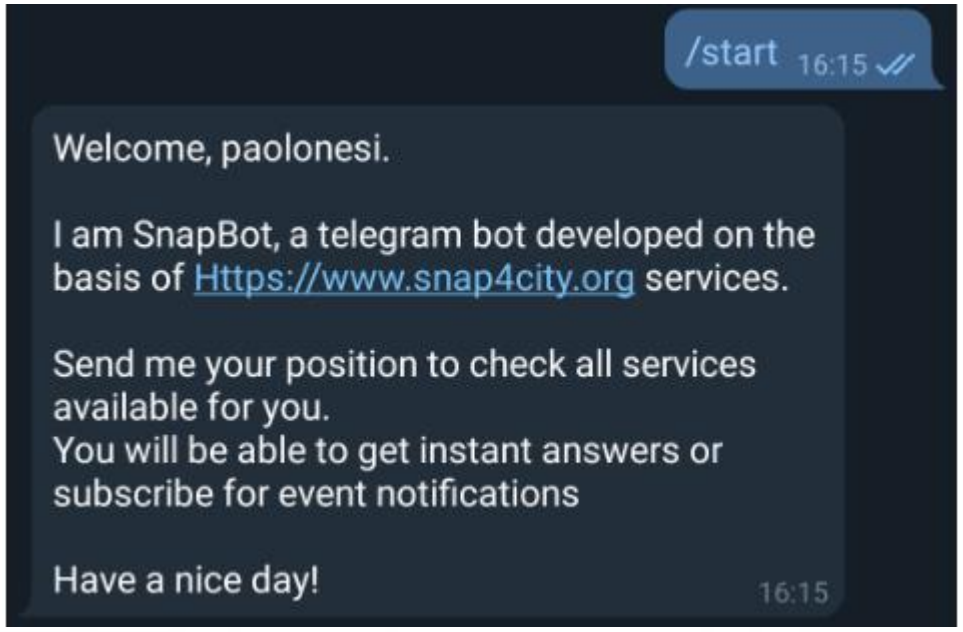
IOT App, Data event firing, event detection and firing Critical event management

Integration with Telegram

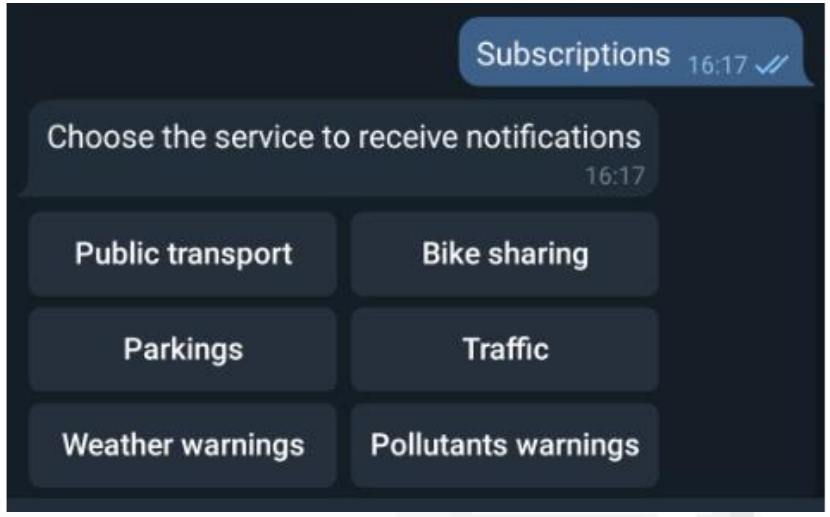


SnapBot

- provides real time smart city services to Telegram users, geolocalized, when you like, what you like
- active on Tuscany in all provinces and cities according to the data accessible on Https://www.snap4city.org
- Services on
  - Public Transport (more than 10 different operators),
  - bike sharing, parking lots,
  - traffic flow, weather warnings,
  - Air quality, pollutant,
  - find your location, etc.

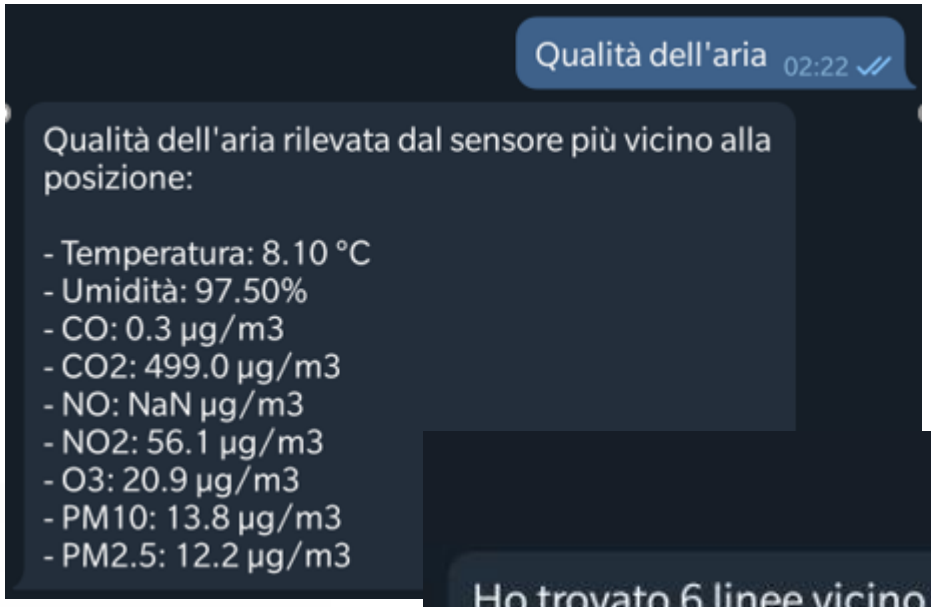






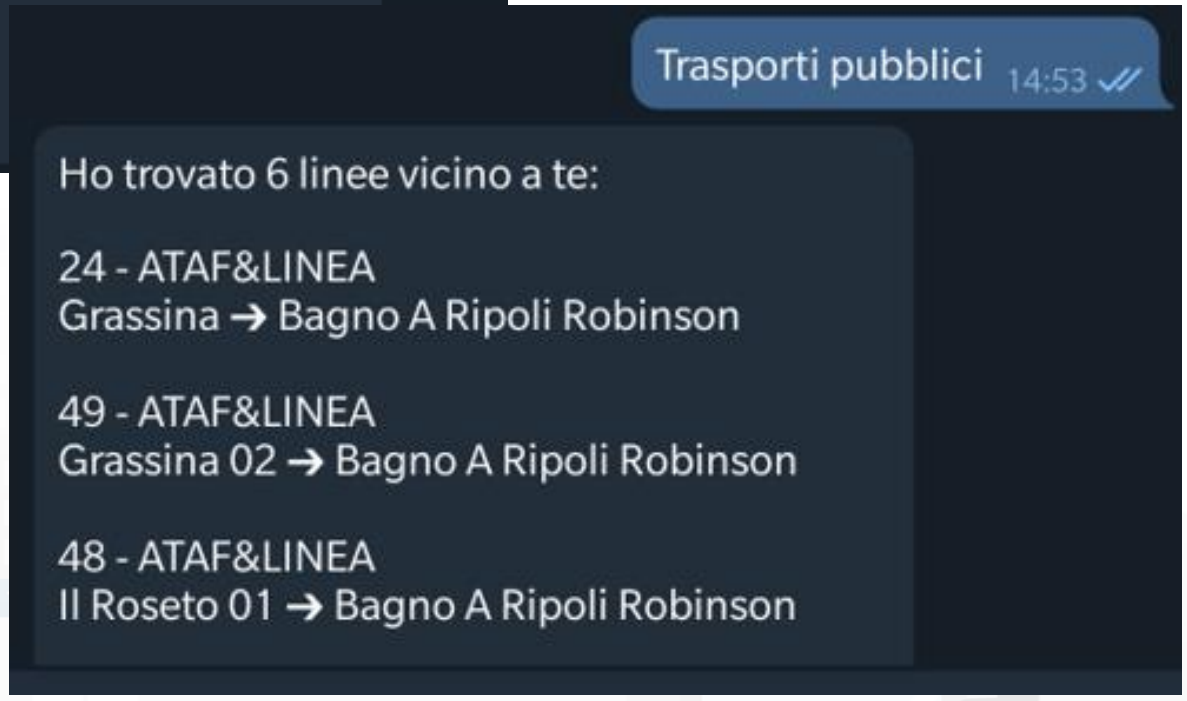
SnapBot


Tap on the hour you prefer to receive 3 notification everyday for the Bike Sharing service
16:18
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
08:00
09:00
10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
18:00
19:00
20:00
21:00
22:00
23:00

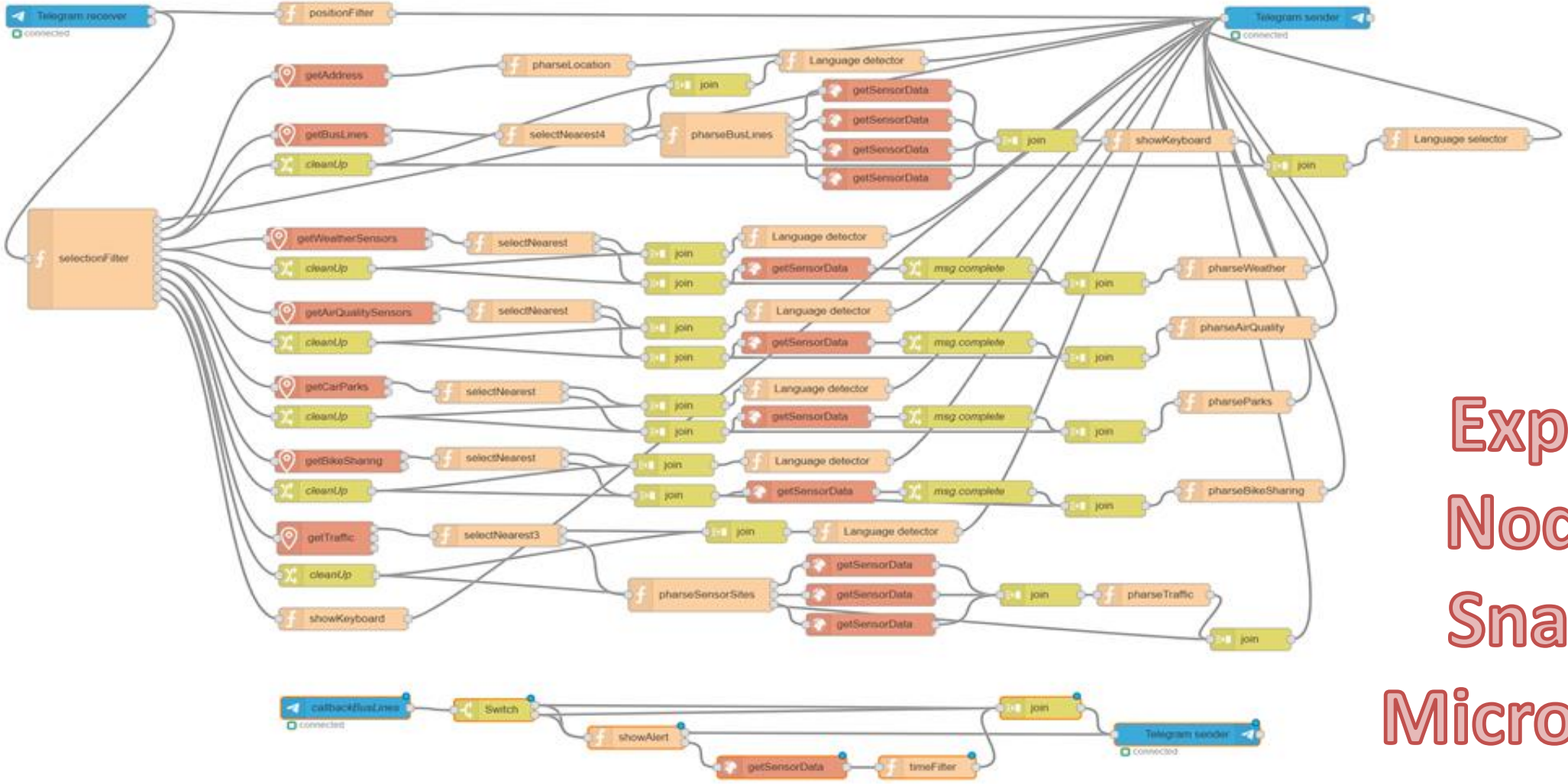

Qualità dell'aria 02:22
Qualità dell'aria rilevata dal sensore più vicino alla posizione:
- Temperatura: 8.10 °C
- Umidità: 97.50%
- CO: 0.3 µg/m3
- CO2: 499.0 µg/m3
- NO: NaN µg/m3
- NO2: 56.1 µg/m3
- O3: 20.9 µg/m3
- PM10: 13.8 µg/m3
- PM2.5: 12.2 µg/m3


SnapBot


Public transport 16:41
Choose a bus stop: 16:42
Giorgini
Giorgini
Vittorio Emanuele
Montelatici
Giorgini - FM0256
17:12 - [55] → Cappuccini
17:29 - [55] → Cappuccini
17:45 - [55] → Cappuccini
18:01 - [55] → Cappuccini
18:17 - [55] → Cappuccini
18:33 - [55] → Cappuccini 16:43


Trasporti pubblici 14:53
Ho trovato 6 linee vicino a te:
24 - ATAF&LINEA
Grassina → Bagno A Ripoli Robinson
49 - ATAF&LINEA
Grassina 02 → Bagno A Ripoli Robinson
48 - ATAF&LINEA
Il Roseto 01 → Bagno A Ripoli Robinson

IOT App of SnapBot: OneShot Services



Exploiting Node-RED Snap4City MicroServices

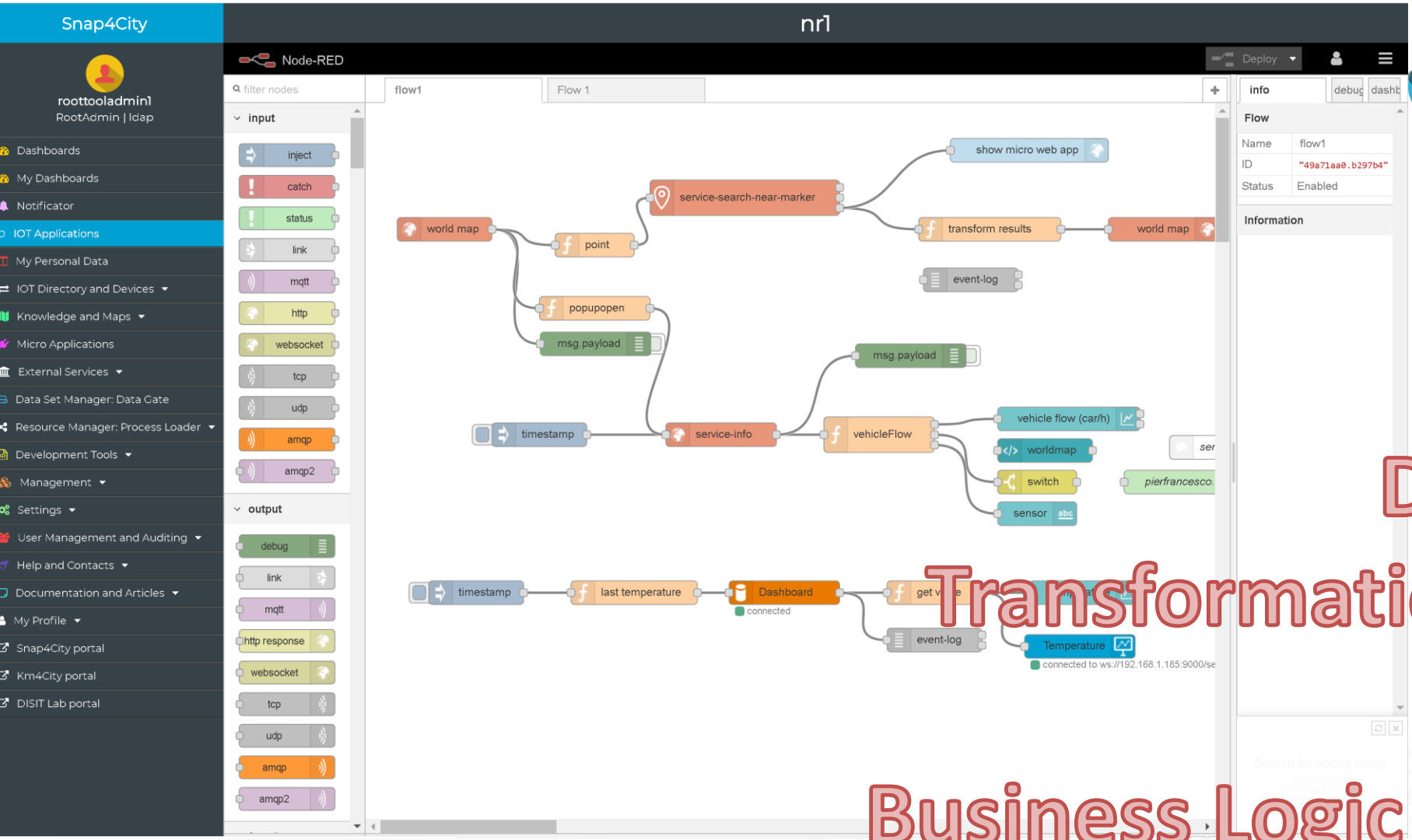
Examples of IoT App/Proc.Logic for Smartening Solutions





Agenda of this integrated overview part

- Recall of Snap4City Architecture
- Data Ingestion Strategy and Orientation
  - Basic entity elements
  - Knowledge Base: Modelling and ServiceURI as Entity Identifier
  - Models vs Devices/Entities and Registration
- Develop: Data Processes Proc.Logic / IoT App
  - Proc.Logic = Node-RED + Snap4City
  - An Integrated Example for Time Series
  - Verification of Data Ingestion
  - Exploiting Storage data by using: IoT App / Proc. Logic
  - Interoperability with respect to Hardware staff
  - Interoperability of Snap4City Platform
- Some Applicative examples
- Training Material
- Acknowledgements



Editing IOT Applications

Data Adapation

Transformation, Conversion

Integration

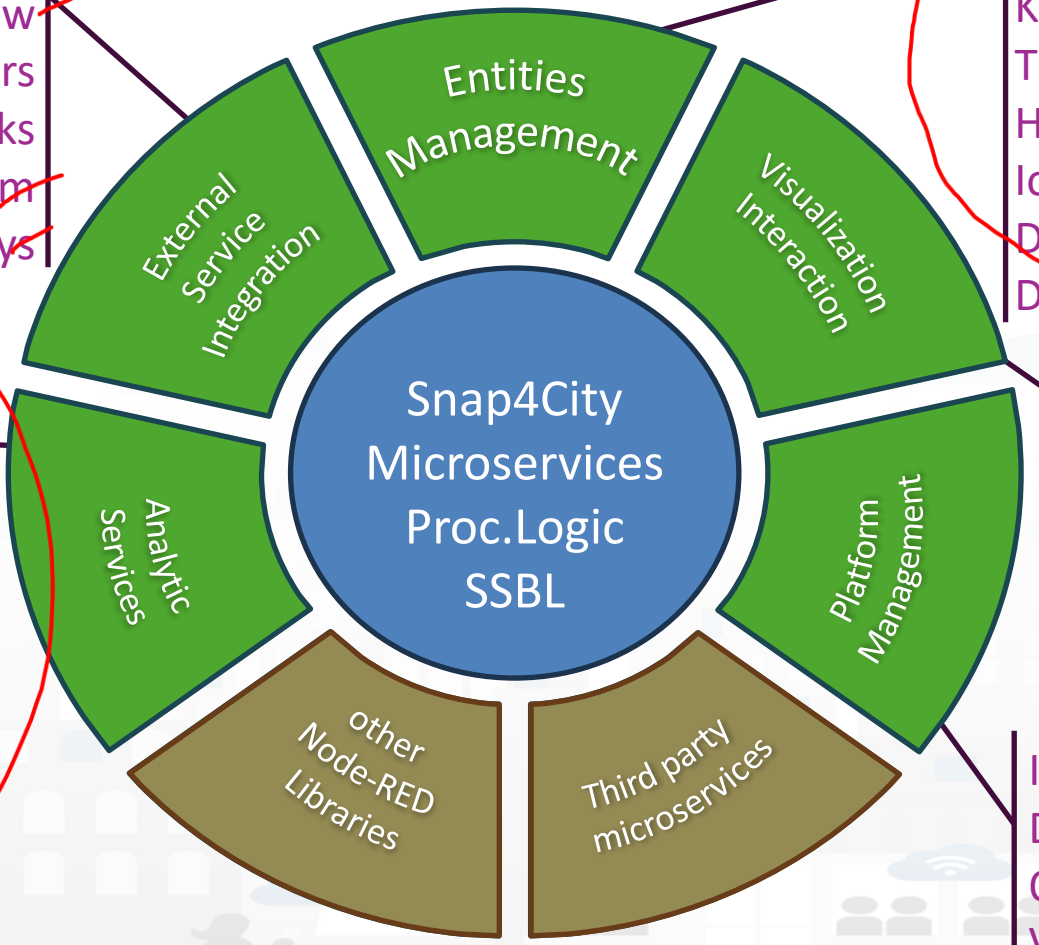
Business Logic vs Dashboards

Data Analytics control

Everywhere: Cloud, on IoT Edge Devices

MicroServices Areas

> 60.000 downloads



Open Data CKAN
Ticket Management, workflow
BIM Servers
Social Networks
Video Management system
Gateways

Data Load / Search / Retrieval
KPI, POI, GIS Data, Scenarios
Time Series, Public transport
High Level Types: heatmaps, ODM,…
IoT / Entity Discovery
Delegation Management
Data Mapping

Data Analytics
Statistic, Optimization
Simulation
Artificial Intelligence
What-if Analysis Support
Geo Utilities Support
Routing & Traffic Flow
MLOps support
Python support
R Studio Support

Dashboards
Widgets: Graphic Libraries
Interactive Widgets
Maps, 3D representations
Synoptics, External Content
Micro Web App

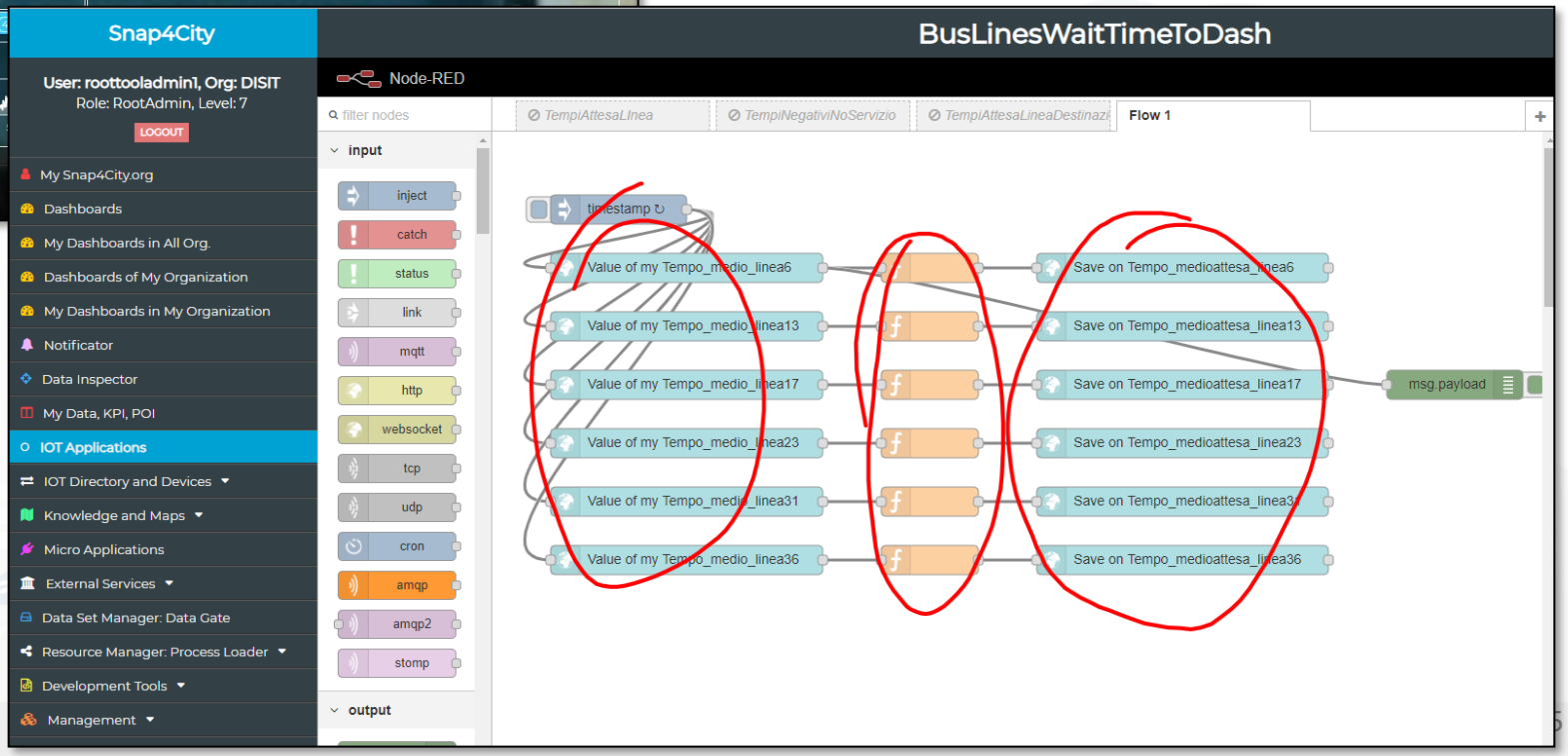
IoTApp Management
Data Logs, A&A, Security
Ownership Management
VPN remote access

Node-RED

Estimation of the mean waiting time at bus stops







PaxCounter devices







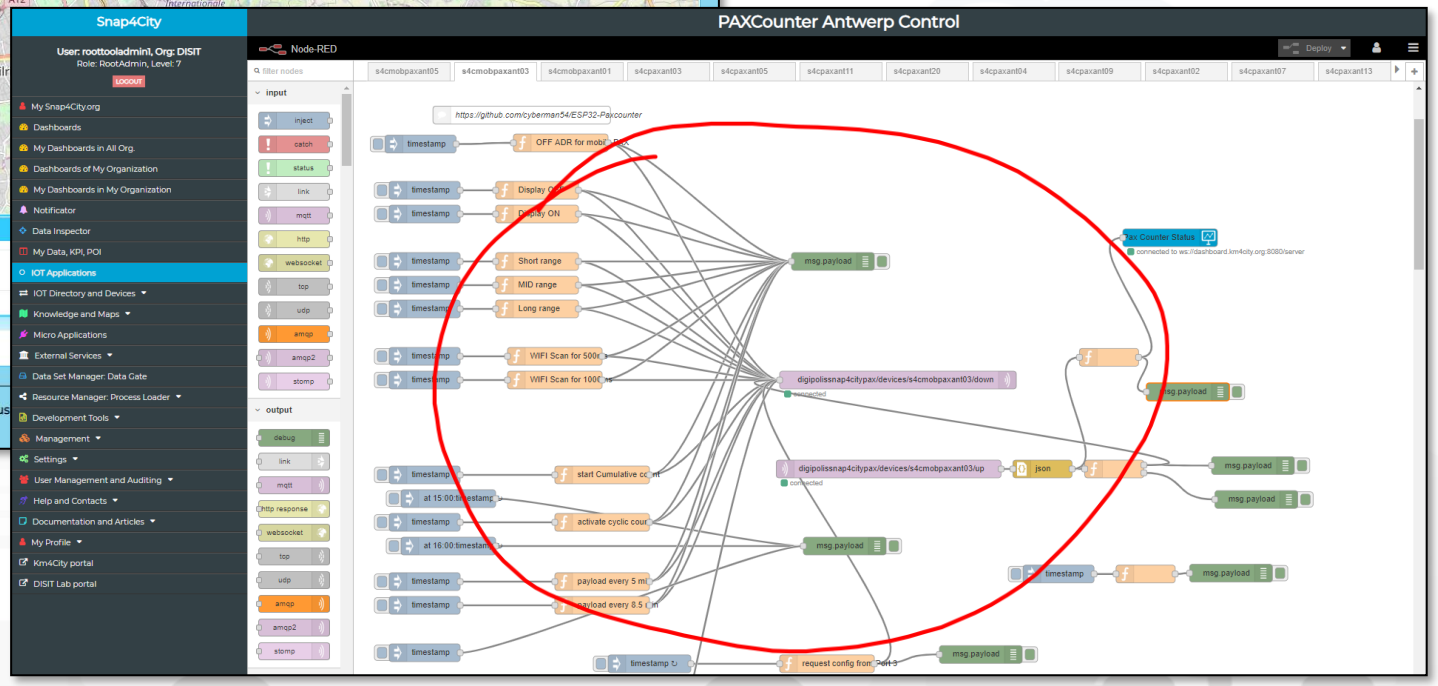
- **Fix PaxCounter LoraWan**
  - sniffing on: Wi-Fi, Bluetooth
  - Sending data via LoraWan
- **Mobile PaxCounter LoraWan**
  - sniffing on: Wi-Fi, Bluetooth
  - Sending data via LoraWan
- **Fix PaxCounter, multiple out**
  - Sending data via LoraWan and Wi-Fi
  - sniffing on: Wi-Fi, Bluetooth

https://www.snap4city.org/456

Programmable PAX counting

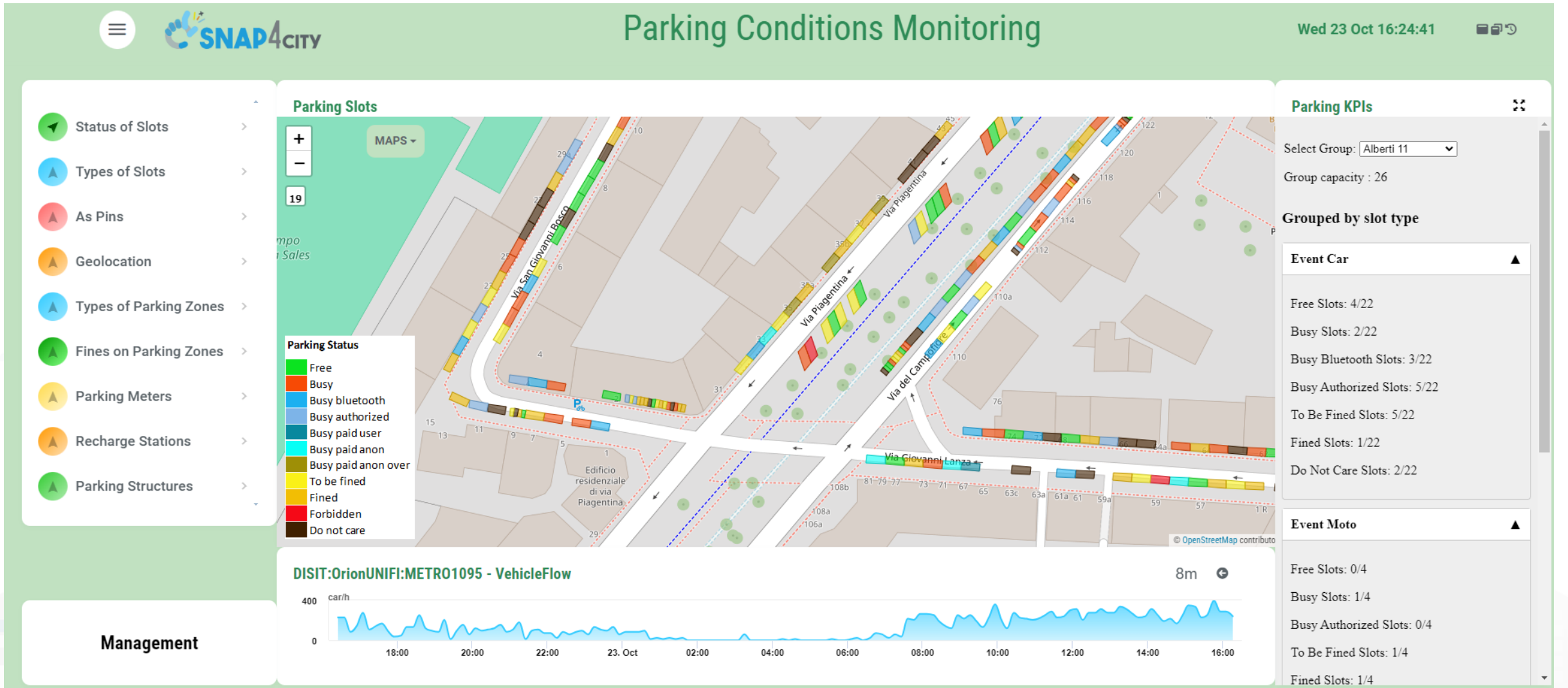




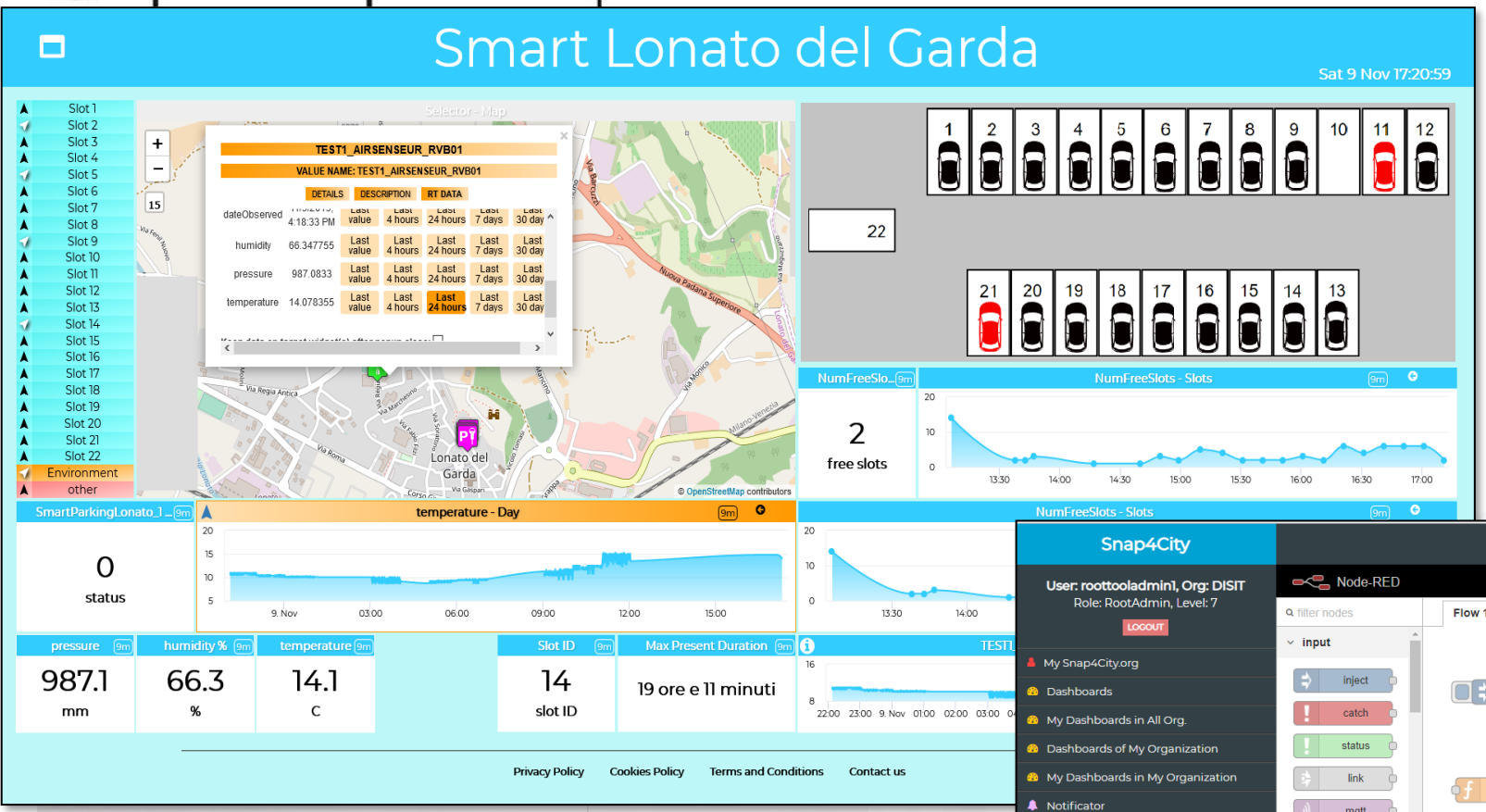


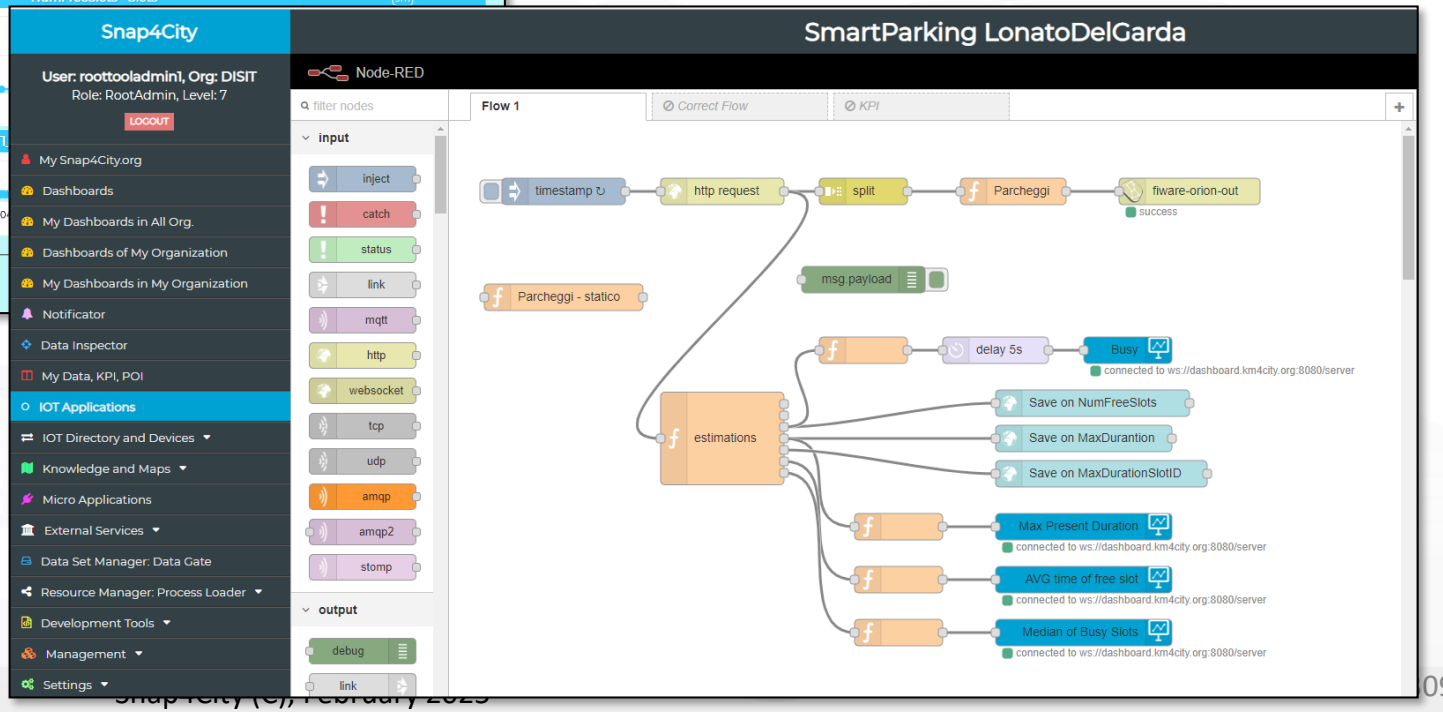
Antwerp

Road Parking



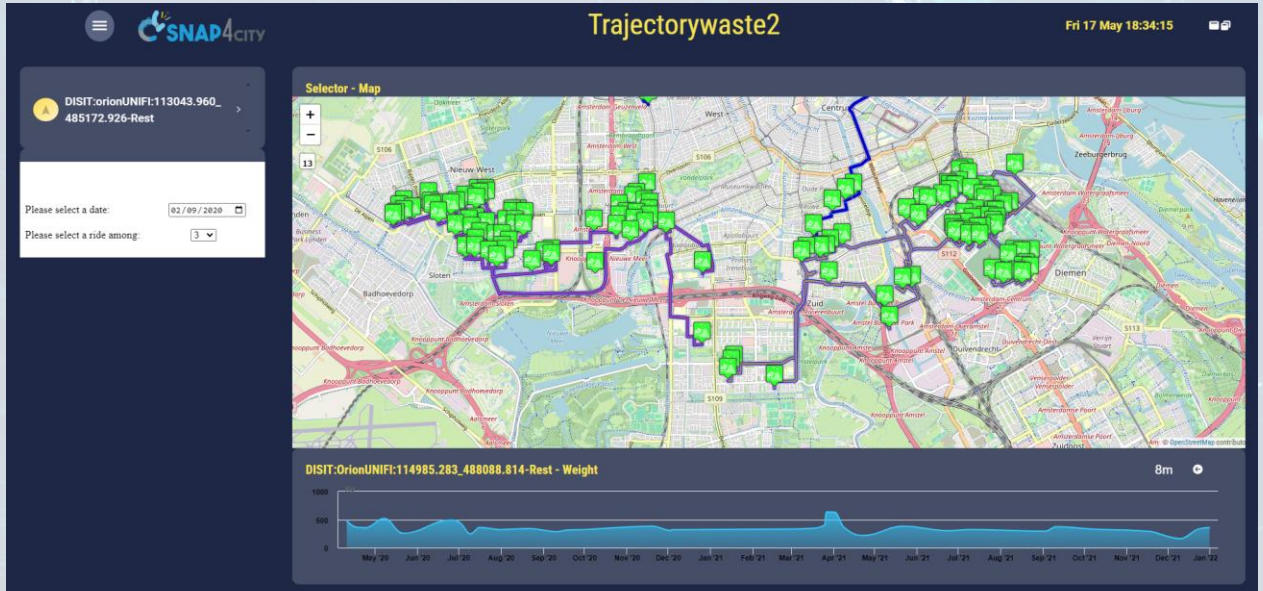
Smart Parking Monitoring (SVG, statistic, overparking)

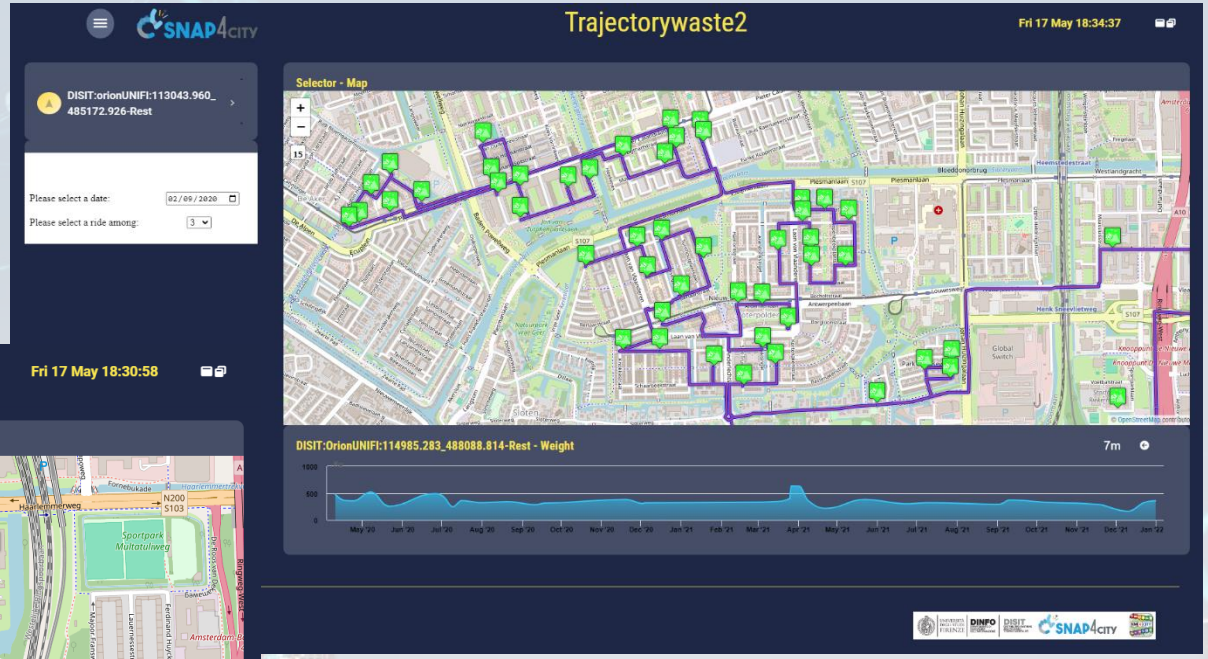


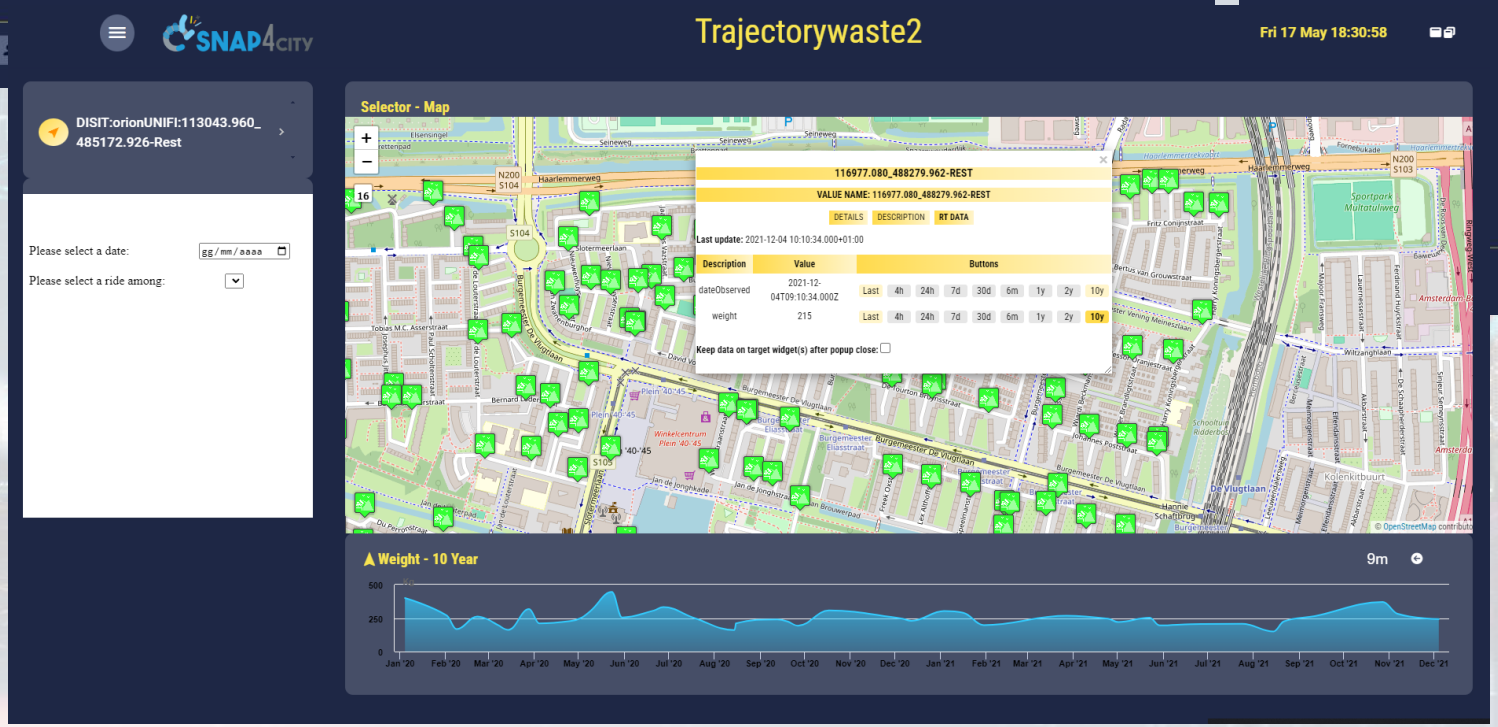




Smart Waste


Trajectorywaste2
Fri 17 May 18:34:15
DISIT:orionUNIFI:113043.960_485172.926-Rest
Please select a date:
02/09/2020
Please select a ride among:
3
Selector - Map
DISIT:OrionUNIFI:114985.283_488088.814-Rest - Weight
8m


Trajectorywaste2
Fri 17 May 18:34:37
DISIT:orionUNIFI:113043.960_485172.926-Rest
Please select a date:
02/09/2020
Please select a ride among:
3
Selector - Map
DISIT:OrionUNIFI:114985.283_488088.814-Rest - Weight
7m


Trajectorywaste2
Fri 17 May 18:30:58
DISIT:orionUNIFI:113043.960_485172.926-Rest
Please select a date:
gg/mm/aaaa
Please select a ride among:
Selector - Map
116977.080_488279.962-REST
VALUE NAME: 116977.080_488279.962-REST
DETAILS
DESCRIPTION
RT DATA
Last update: 2021-12-04 10:10:34.000+01:00
Description
Value
Buttons
dateObserved
2021-12-04T09:10:34.000Z
weight
215
Keep data on target widget(s) after popup close:
Weight - 10 Year
9m


11
SUSTAINABLE CITIES AND COMMUNITIES

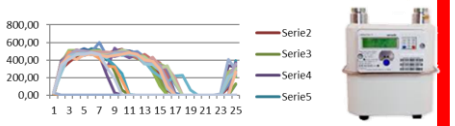

3
GOOD HEALTH AND WELL-BEING

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

https://www.snap4city.org/369

SNAP4CITY KM4CITY

External Services



Fleet management



IoT Devices/Edge

IoT Broker



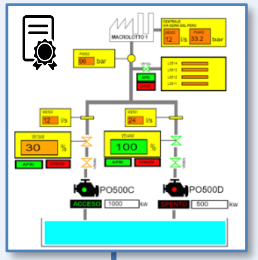
Internet

SECURE





Admin





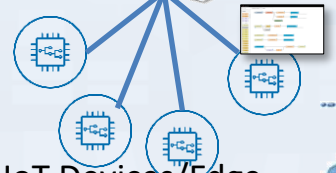
DCS

IoT Broker

PLC



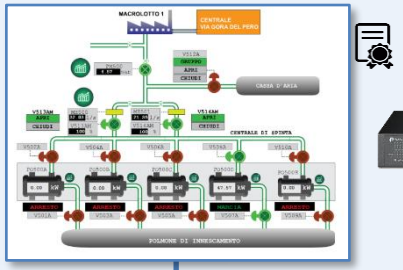




IoT Devices/Edge



Industry Plant1.....





SCADA



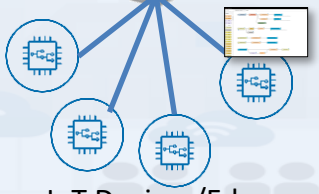


PLC/ RTU



IoT Broker



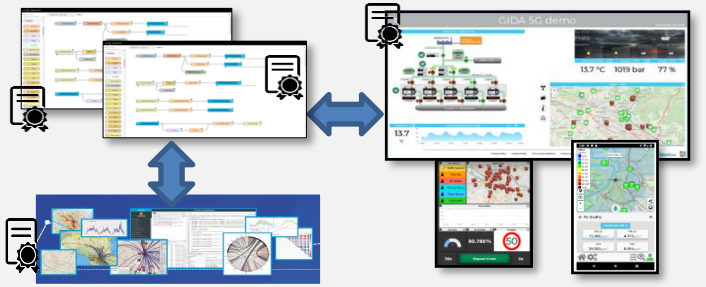


IoT Devices/Edge

Industry Plant2.....

IOT Applications

Dashboards and Apps



Big Data Analytics, Artificial Intelligence

Control and Supervision on Multiple Supply Chains

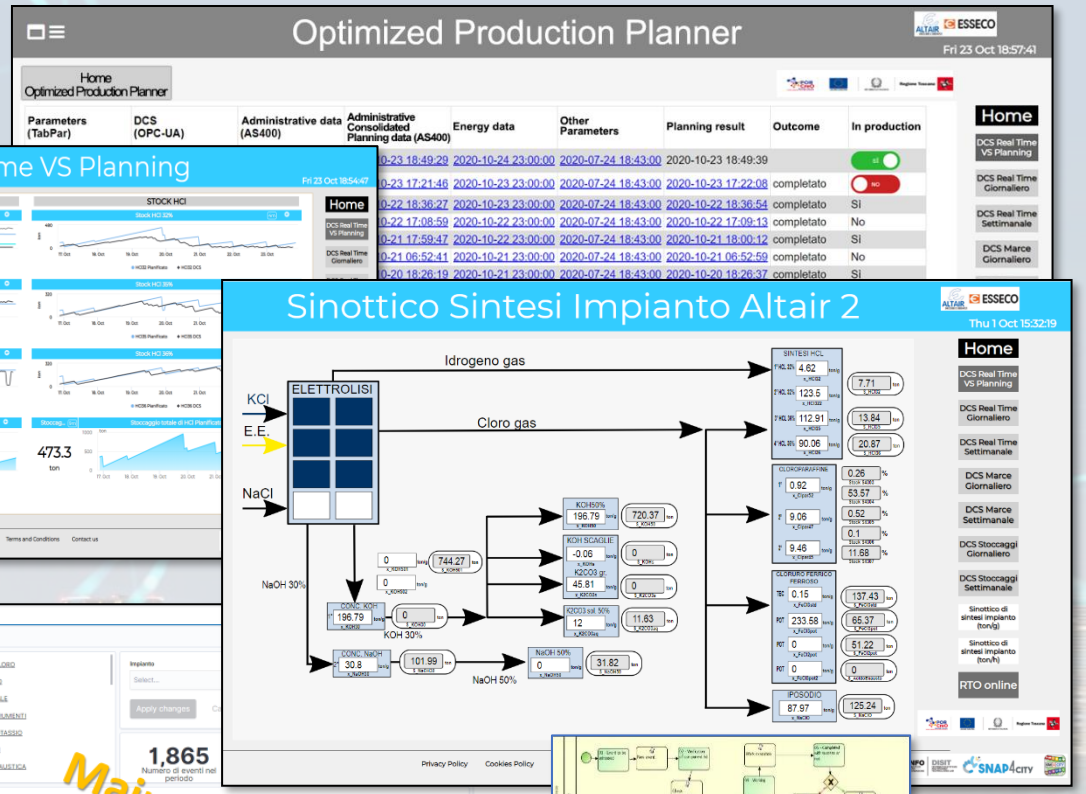
Industry 4.0 as a Service

Snap4Altair Decision Support supervision and control, Industry 4.0

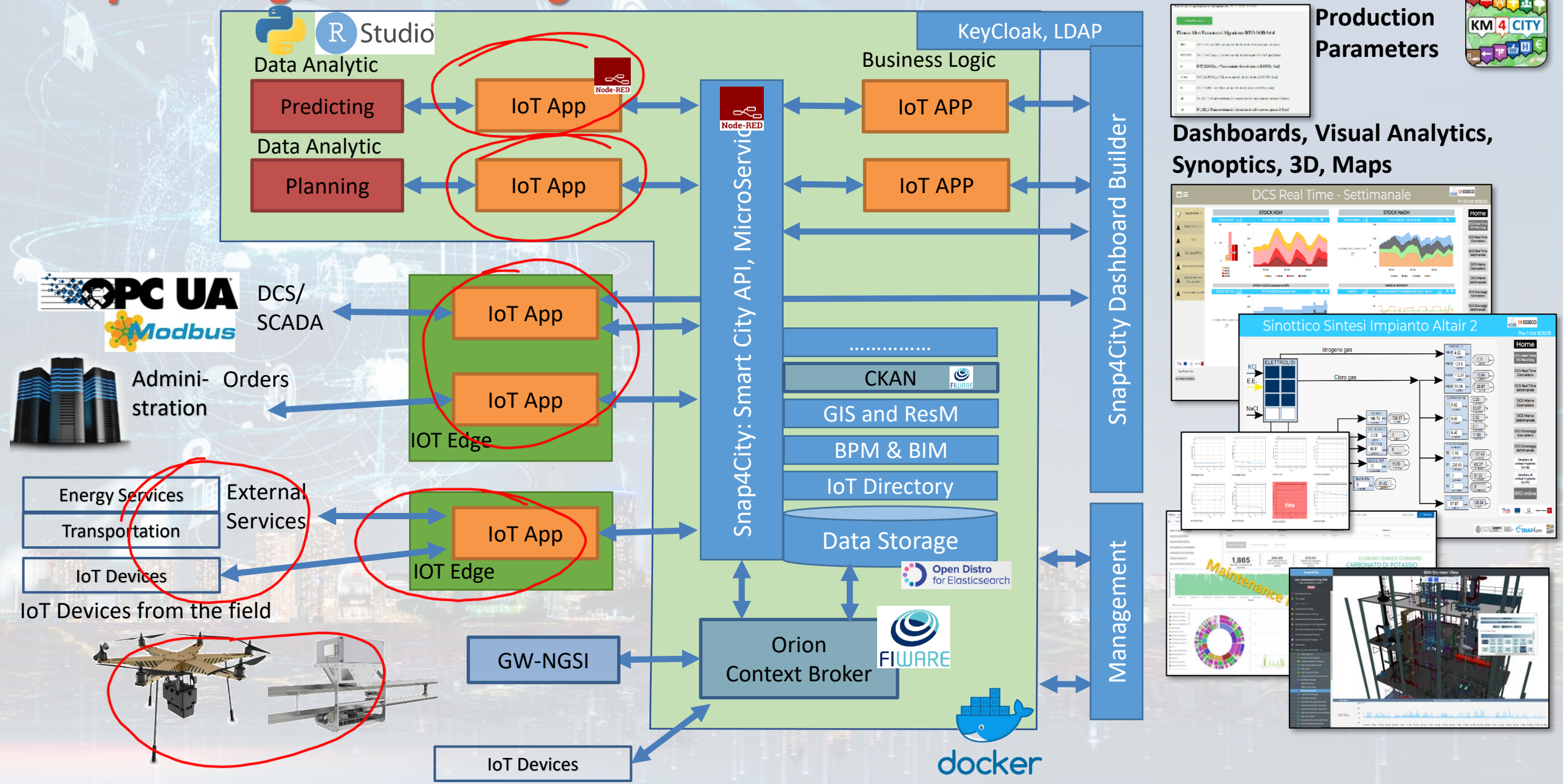
reference

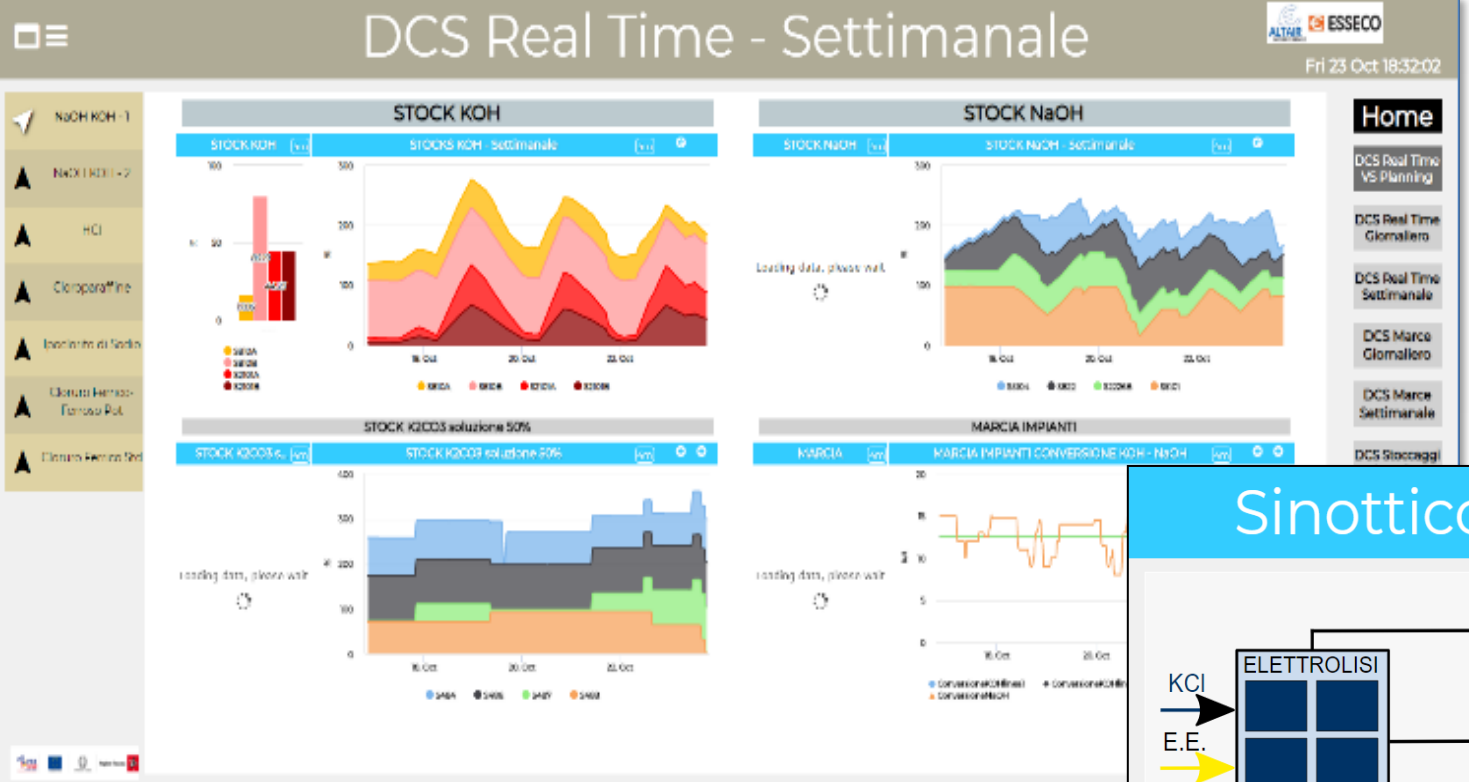
- **Multiple Domain Data**
  - Distributed Control System: energy, flows, storage, chemical data, settings, ..
  - Cost of energy, Orders,
  - Production Parameters
  - Maintenance data
- **Multiple Levels & Decision Makers**
  - Optimized planning on chemical model
  - Business Intelligence on Maintenance data
- **Historical and Real Time data**
  - Billions of Data
- **Services Exploited on:**
  - Multiple Levels, Mobile Apps, API
- **Since 2020**

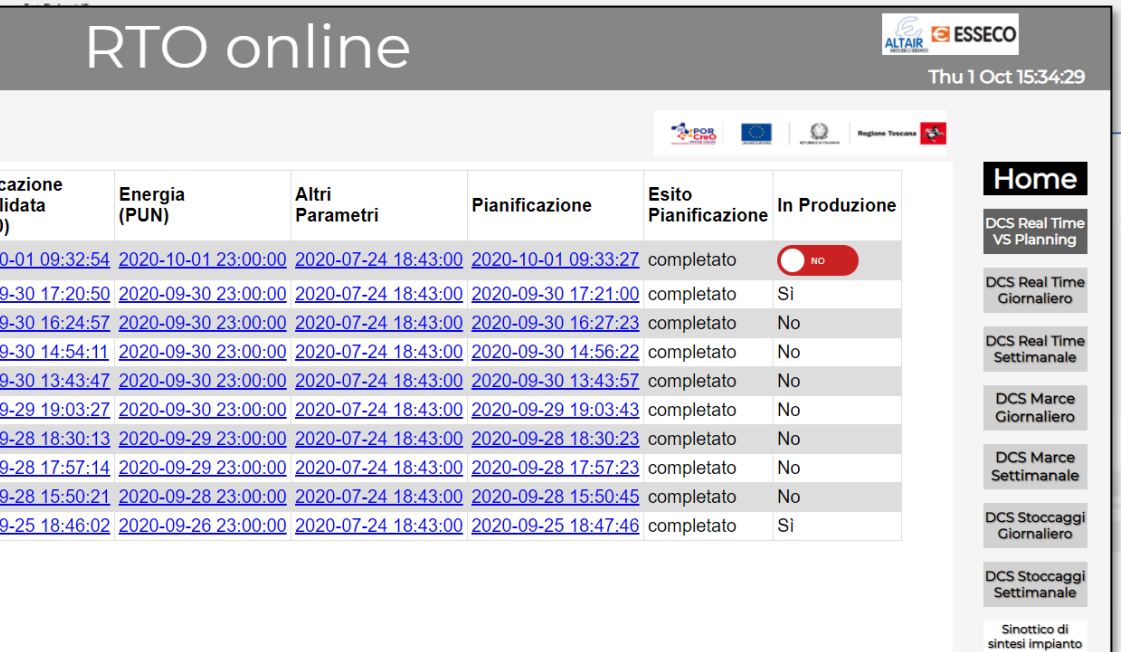


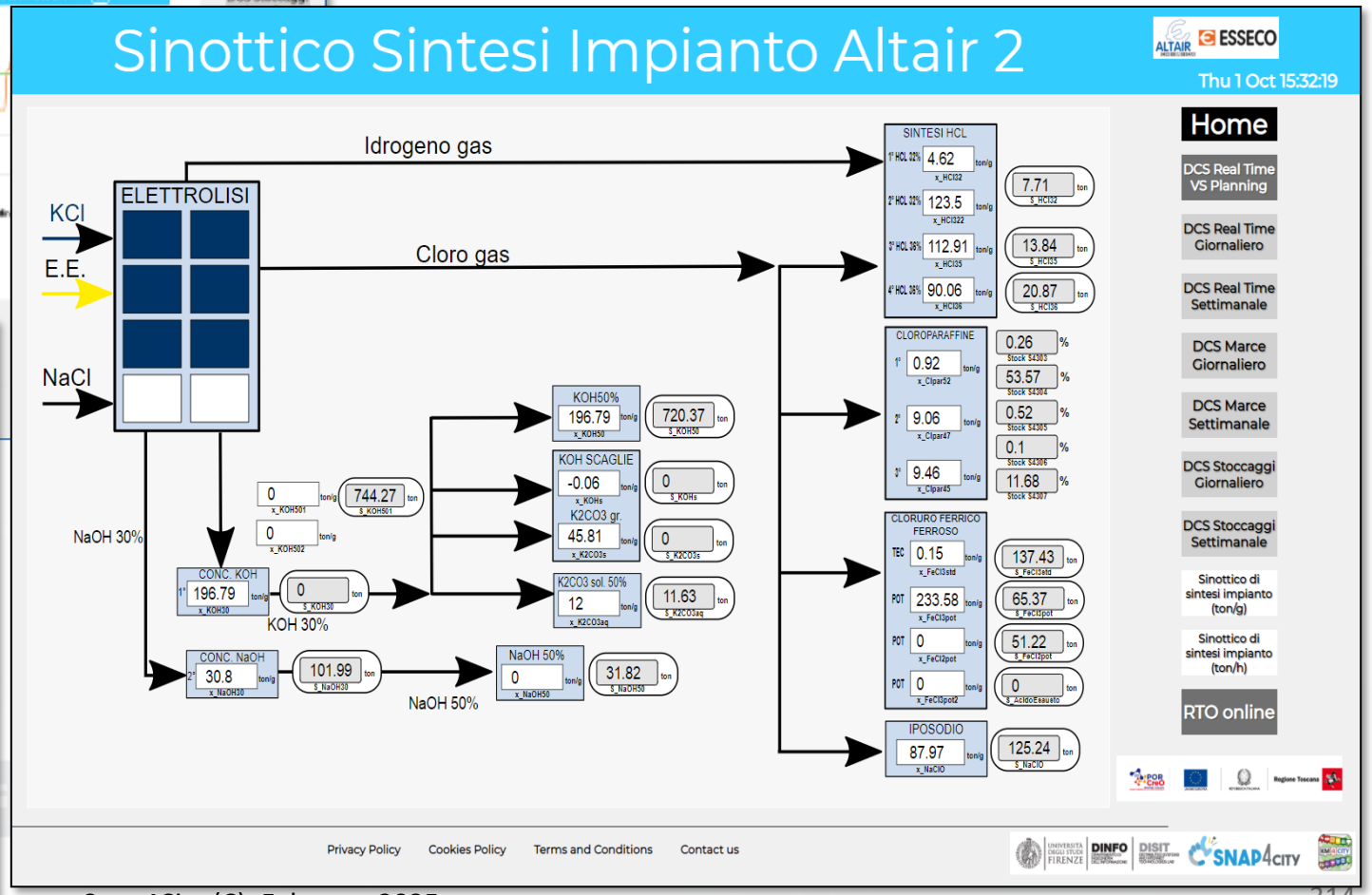


Snap4City/Industry Detailed Architecture

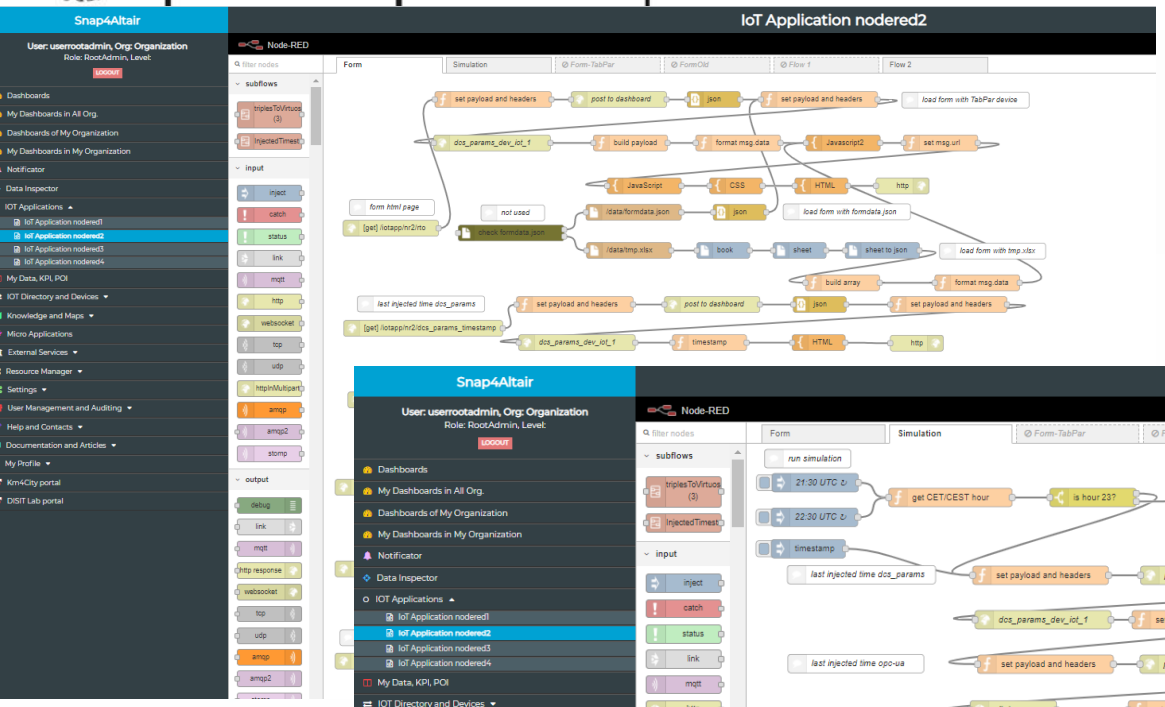


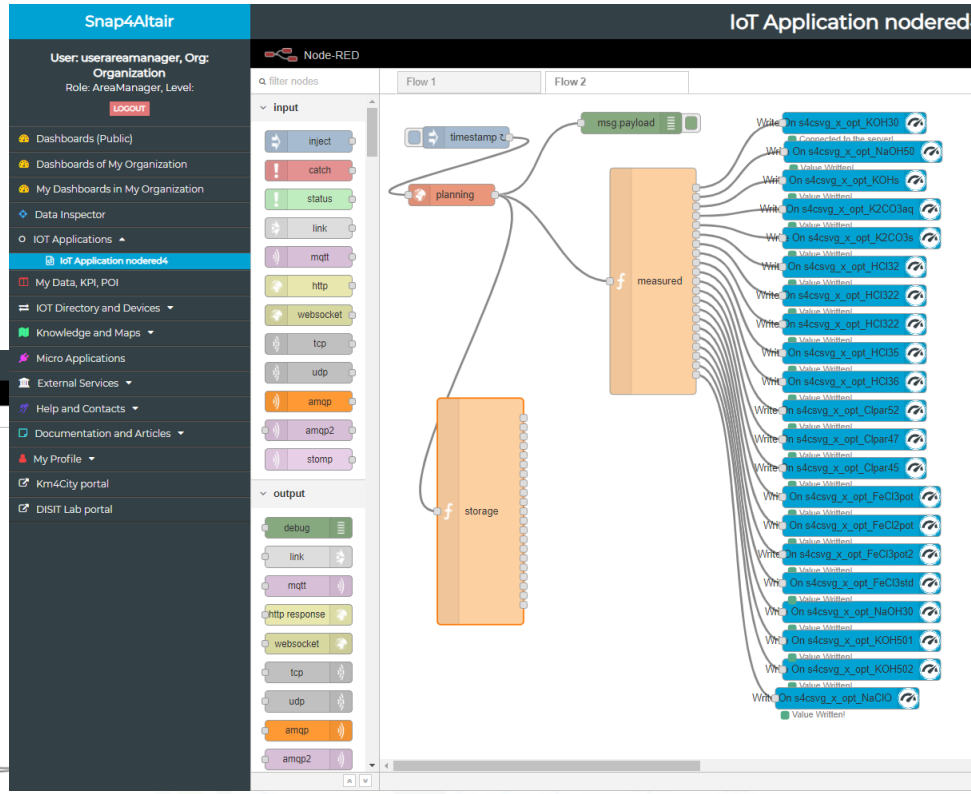

DCS Real Time - Settimanale
ESSECO
Fri 23 Oct 18:32:02
STOCK KOH
STOCK NaOH
STOCK K2CO3 soluzione 50%
MARCIA IMPIANTI
Home
DCS Real Time VS Planning
DCS Real Time Giornaliero
DCS Real Time Settimanale
DCS Marce Giornaliero
DCS Marce Settimanale
DCS Stoccaggi

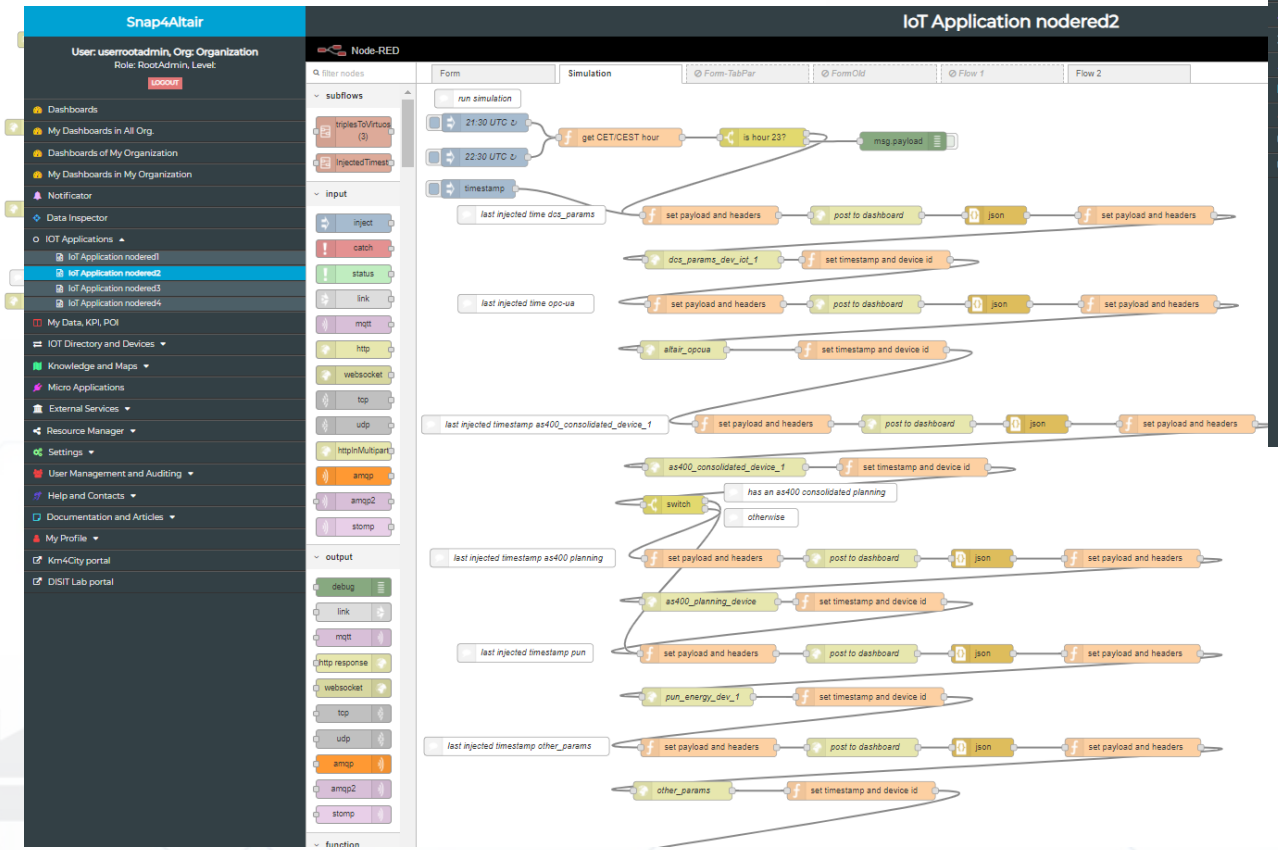

RTO online
ESSECO
Thu 1 Oct 15:34:29
Energia (PUN)
Altri Parametri
Pianificazione
Esito Pianificazione
In Produzione
2020-10-01 09:32:54 2020-10-01 23:00:00 2020-07-24 18:43:00 2020-10-01 09:33:27 completato NO
2020-09-30 17:20:50 2020-09-30 23:00:00 2020-07-24 18:43:00 2020-09-30 17:21:00 completato Sì
2020-09-30 16:24:57 2020-09-30 23:00:00 2020-07-24 18:43:00 2020-09-30 16:27:23 completato No
2020-09-30 14:54:11 2020-09-30 23:00:00 2020-07-24 18:43:00 2020-09-30 14:56:22 completato No
2020-09-30 13:43:47 2020-09-30 23:00:00 2020-07-24 18:43:00 2020-09-30 13:43:57 completato No
2020-09-29 19:03:27 2020-09-30 23:00:00 2020-07-24 18:43:00 2020-09-29 19:03:43 completato No
2020-09-28 18:30:13 2020-09-29 23:00:00 2020-07-24 18:43:00 2020-09-28 18:30:23 completato No
2020-09-28 17:57:14 2020-09-29 23:00:00 2020-07-24 18:43:00 2020-09-28 17:57:23 completato No
2020-09-28 15:50:21 2020-09-28 23:00:00 2020-07-24 18:43:00 2020-09-28 15:50:45 completato No
2020-09-25 18:46:02 2020-09-26 23:00:00 2020-07-24 18:43:00 2020-09-25 18:47:46 completato Sì
Home
DCS Real Time VS Planning
DCS Real Time Giornaliero
DCS Real Time Settimanale
DCS Marce Giornaliero
DCS Marce Settimanale
DCS Stoccaggi Giornaliero
DCS Stoccaggi Settimanale


Sinottico Sintesi Impianto Altair 2
ESSECO
Thu 1 Oct 15:32:19
Idrogeno gas
KCl
E.E.
NaCl
ELETTROLISI
Cloro gas
NaOH 30%
KOH 30%
NaOH 50%
SINTESI HCL
CLOROPARAFFINE
CLORURO FERRICO FERROSO
IPOSODIO
Home
DCS Real Time VS Planning
DCS Real Time Giornaliero
DCS Real Time Settimanale
DCS Marce Giornaliero
DCS Marce Settimanale
DCS Stoccaggi Giornaliero
DCS Stoccaggi Settimanale
Sinottico di sintesi impianto (ton/g)
Sinottico di sintesi impianto (ton/h)
RTO online
Privacy Policy
Cookies Policy
Terms and Conditions
Contact us

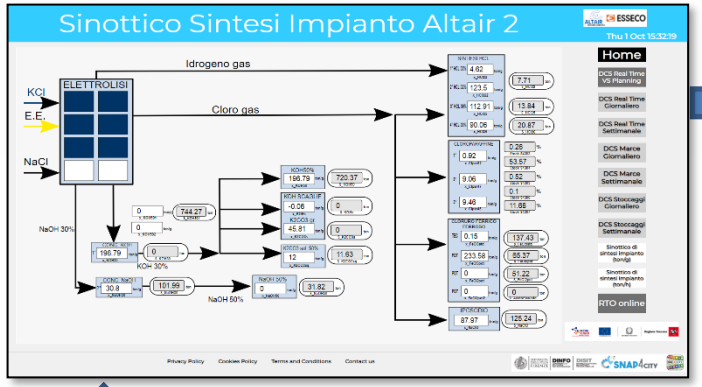
Some Flows


Snap4Altair
IoT Application nodered2
Node-RED


Snap4Altair
IoT Application nodered
Node-RED

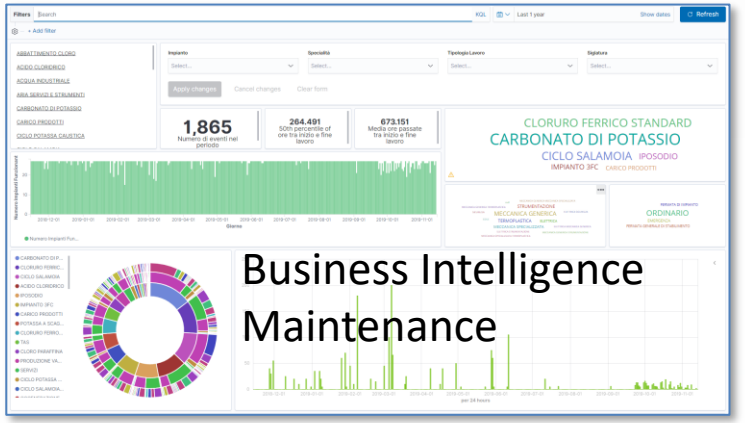

Snap4Altair
IoT Application nodered2
Node-RED

Workflow for Ticket management

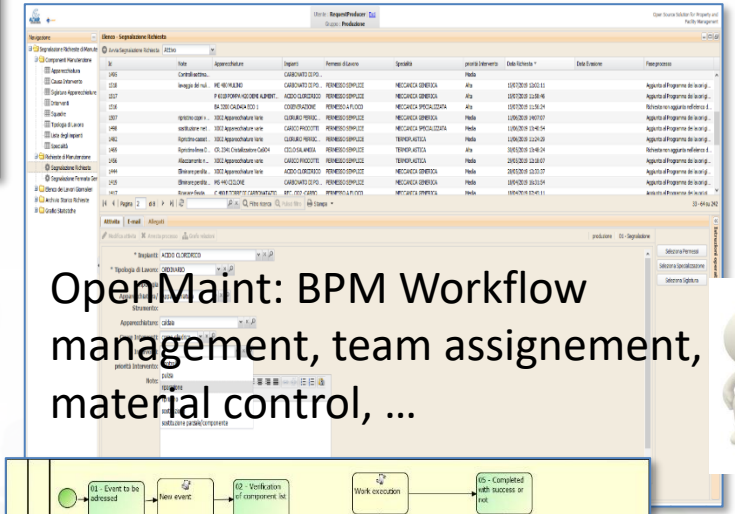


Dashboards and actions

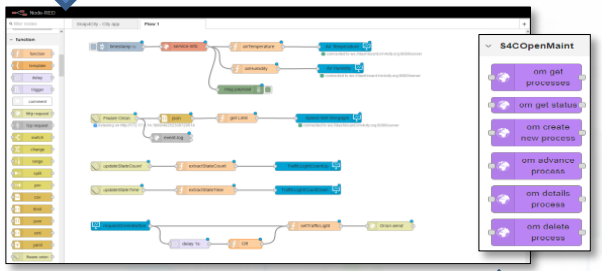
Consumptions/productions



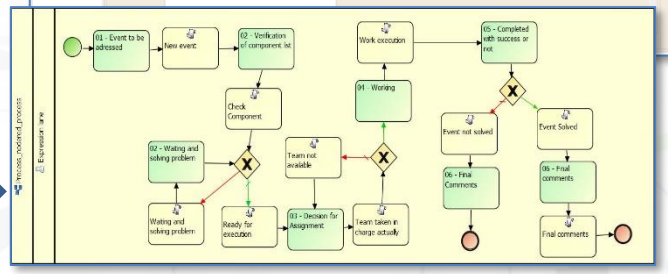
Events/actions



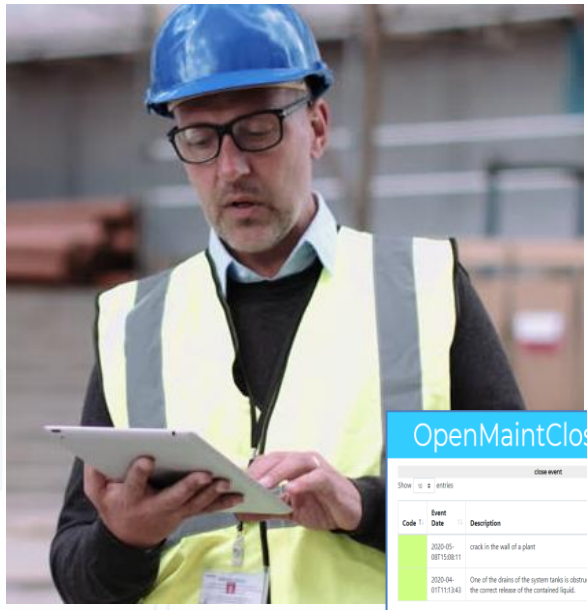
OpenMaint: BPM Workflow management, team assignement, material control, …

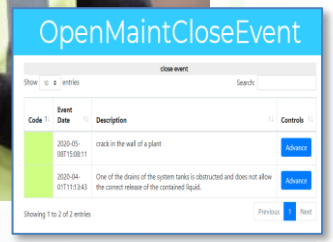


IOT App, Data event firing, event detection and firing Critical event management



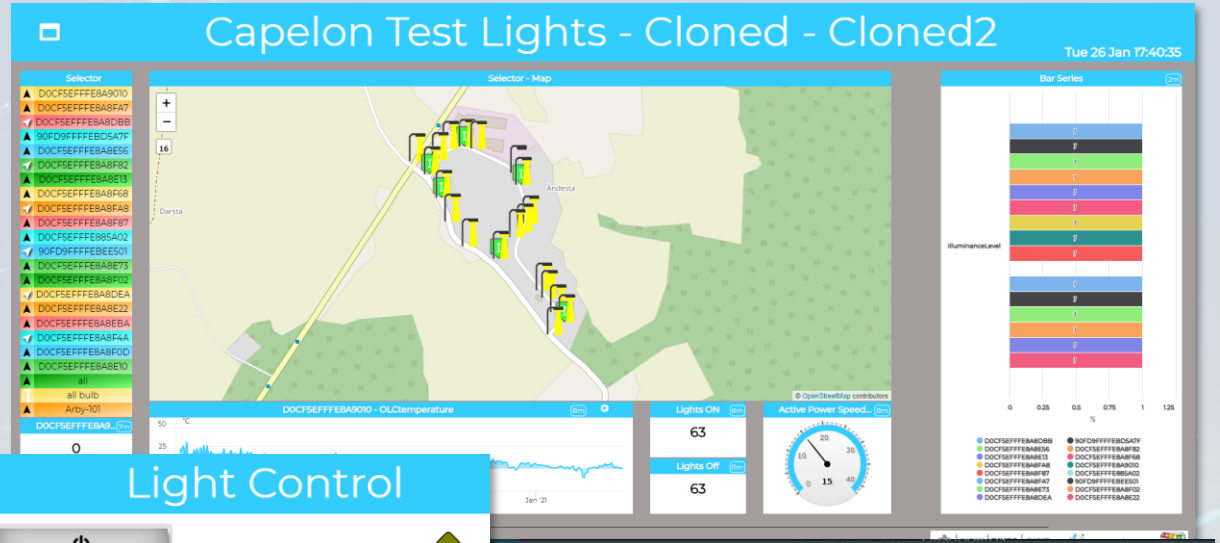




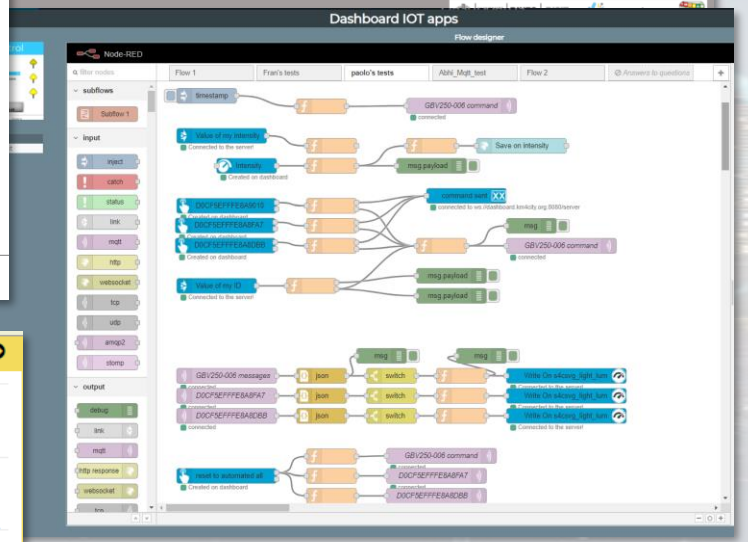


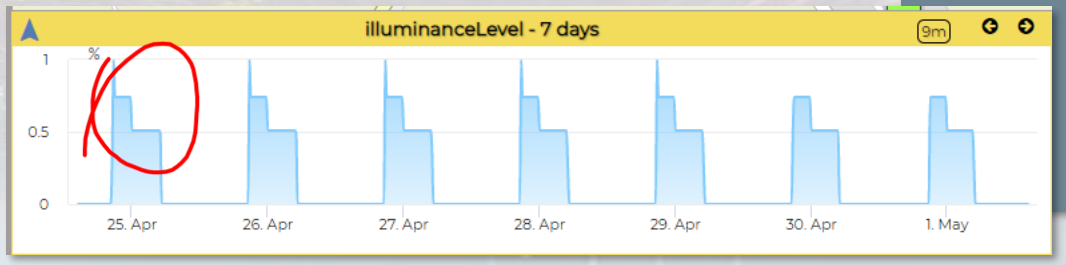
Smart Light Control of CAPELON *reference*

- **Energy Domain**
  - Smart Light, MQTT, ….
  - IoT Orion Broker FIWARE
- **Dashboards**
  - Map coverage on Sweden
  - Monitoring and real time control
  - Energy control, analytics
  - Direct control
- **Historical and Real Time data**
- **Services Exploited on:**
  - Multiple Levels, API
  - Dashboards
- **Since 2020**

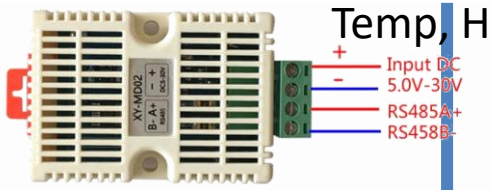








Devices



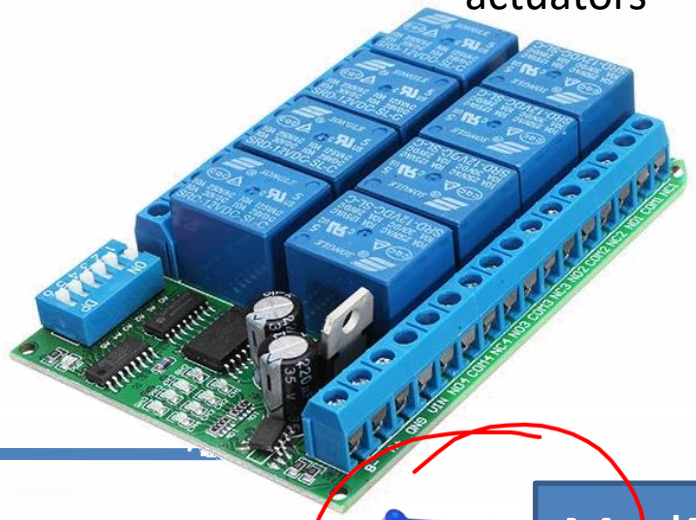
Temp, Humidity



Energy Metering

Modbus

8 Relais, actuators





USB -- ModBus

ModBus to Snap4City Gateway Edge





IOT Applications

Dashboards and Apps

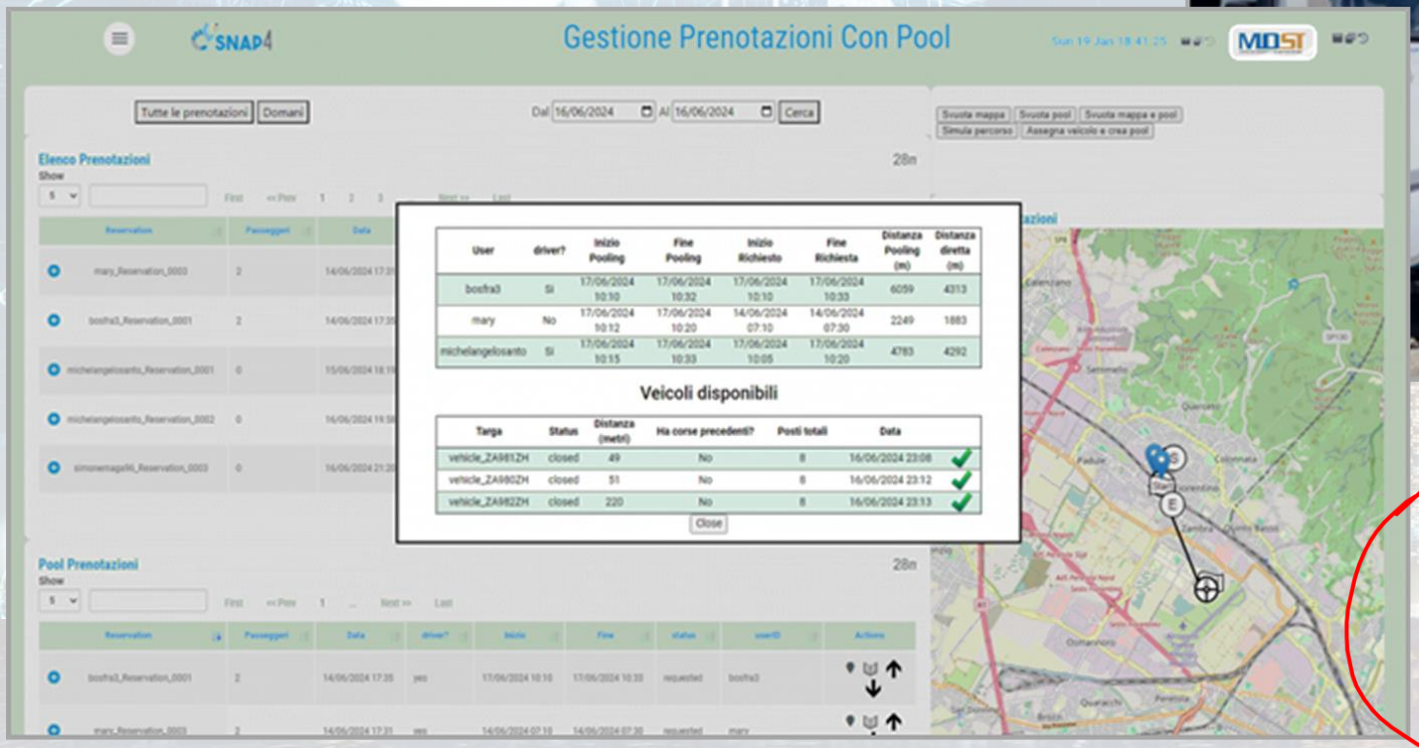
IOT Data Shadow Snap4City

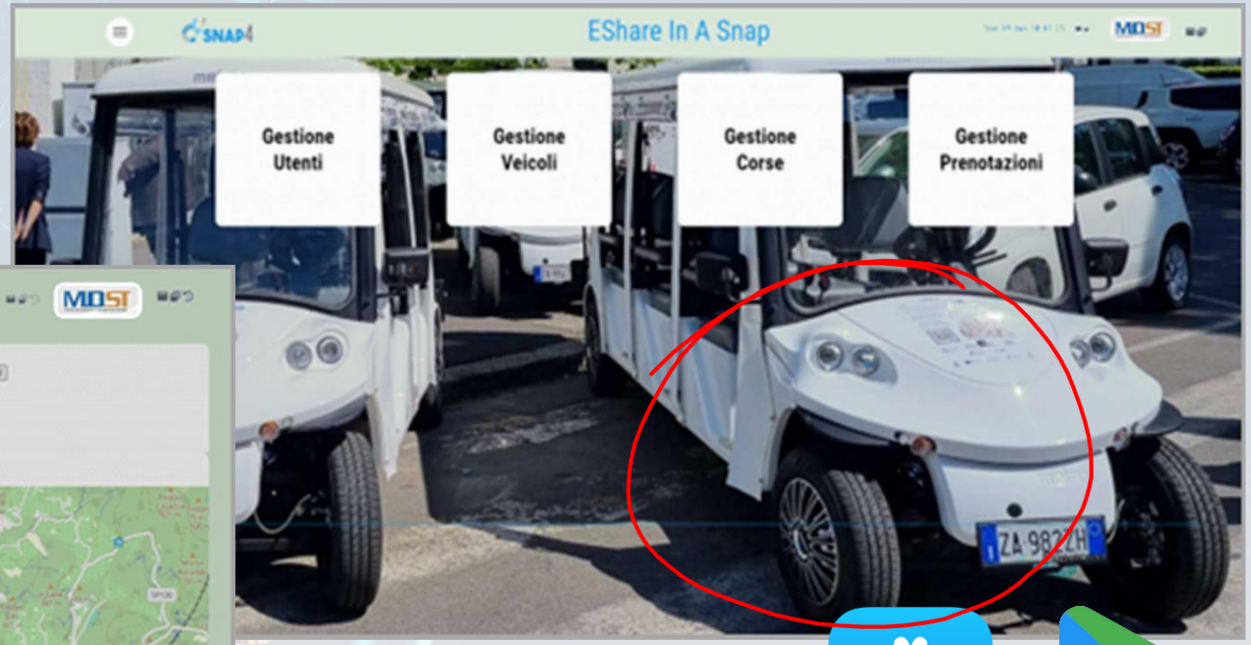
Big Data Analytics, Artificial Intelligence

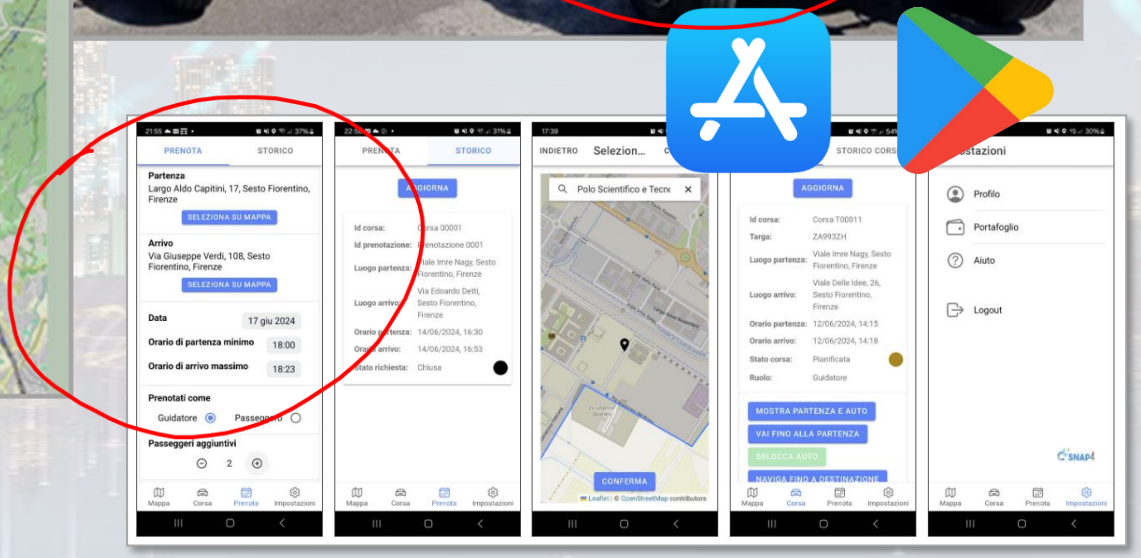
- A large range of devices: sensors and actuators
- Over serial as RS485 and/or IP

Car Sharing and Polling, Fleet Management

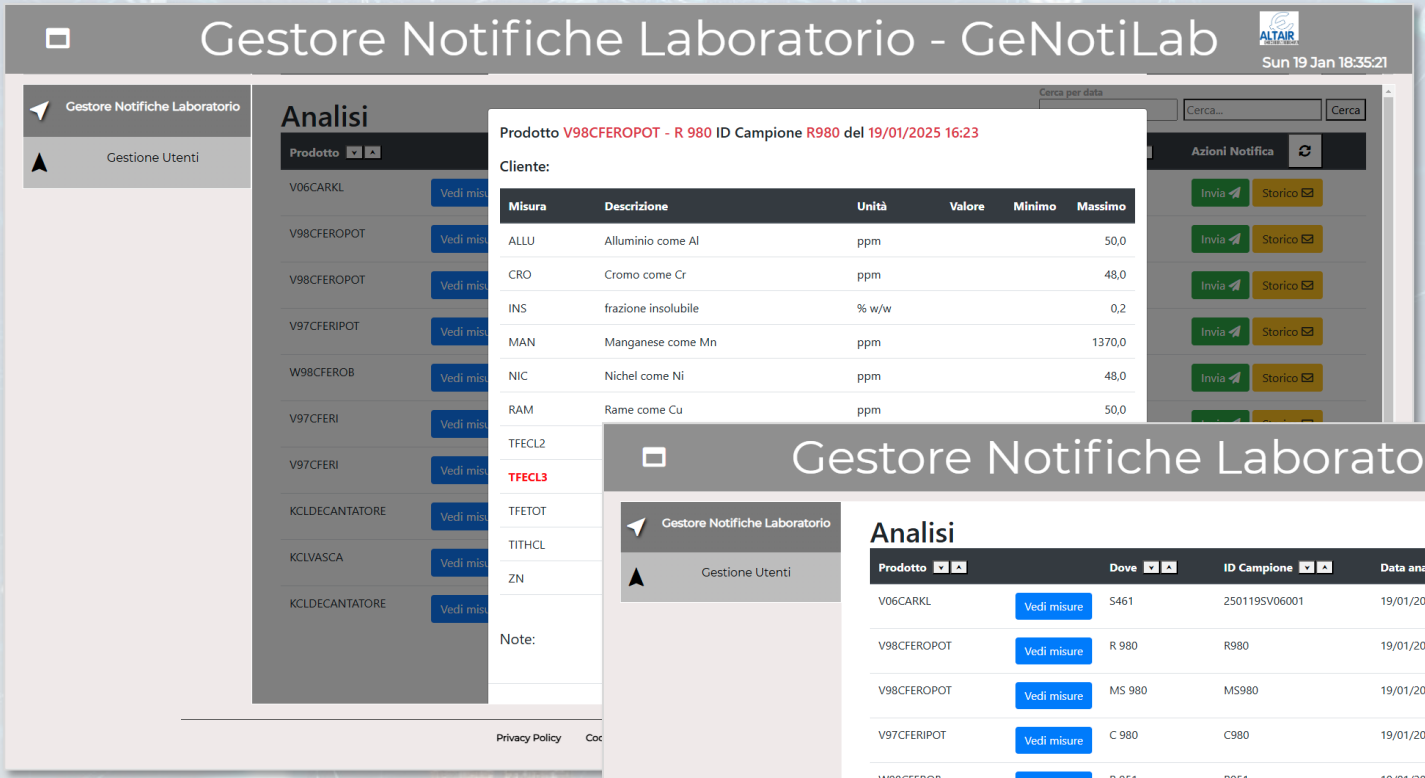
- **eShare in a Snap (TUSS)**



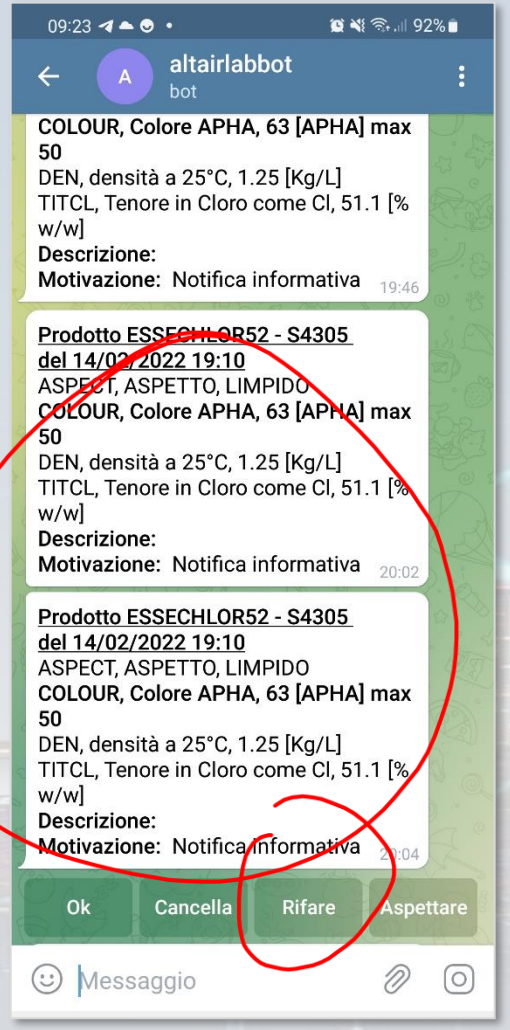




GENOTILAB: Smart production process control



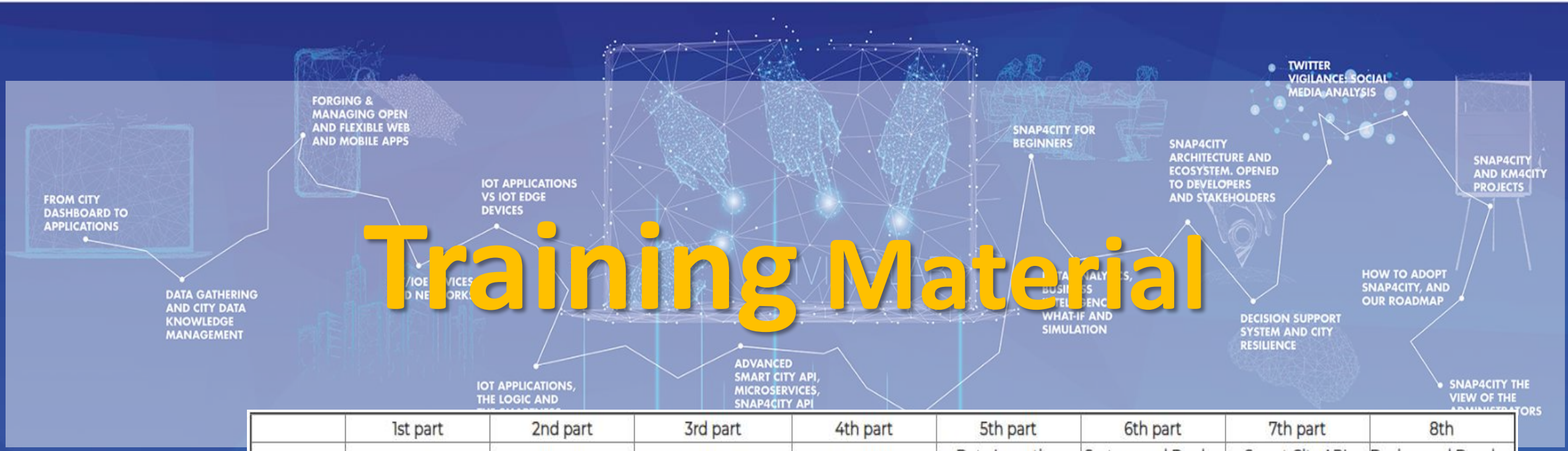




SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT

SNAP4INDUSTRY

KM4CITY



Training Material

	1st part	2nd part	3rd part	4th part	5th part	6th part	7th part	8th
what	Overview	Dashboards	IOT App, IOT Network	Data Analytics	Data Ingestion processes	System and Deploy Install	Smart City API: Web & Mob. App	Design and Develop Smart Solutions
PDF 2022								
Interactive (2022) with video and animations								

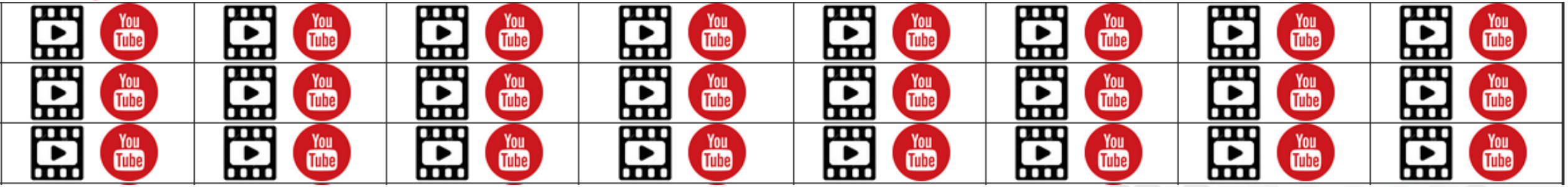
MORE ON:

- Main missing parts which can be taken from **P3, P5 and P6**, are:
  - **Loading triples on KB, converting OSM data to KB**
  - **Get Satellite data, exploit heatmap on GPS data**
  - **Server Side Business Logic** vs Dashboards and Views
  - **Digital Twin data inspector**
  - **BlockChain support to certify Entities Models, Instances, messages**
  - **Harvesting external brokers**
  - **Moving devices, tracking fleets, etc.**
  - **Ingestion of GTFS, ….**
  - **Ingestion of complex High Level Types: ……**
  - **Etc.**

https://www.snap4city.org/944

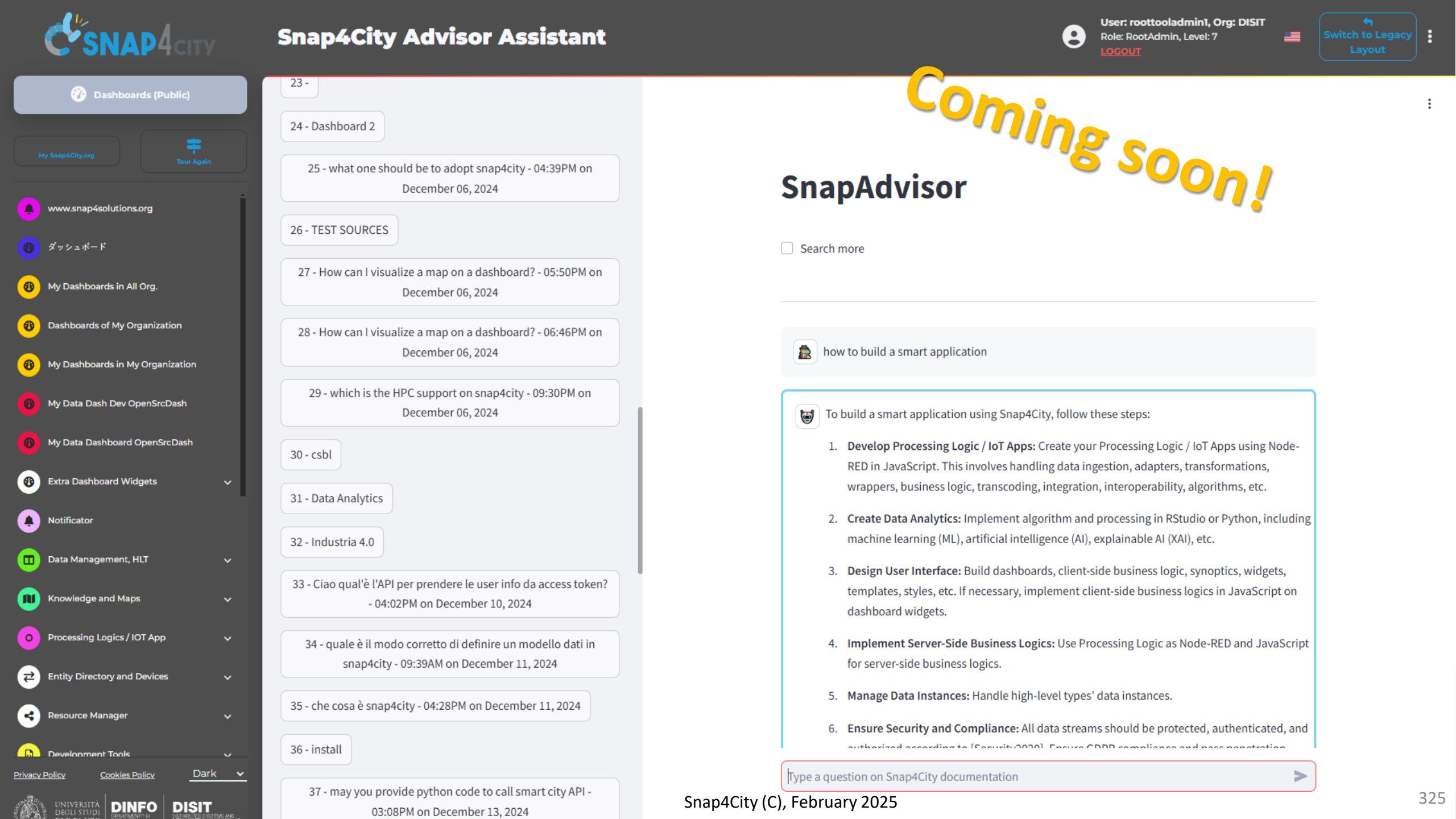
On Line Training Material (free of charge)

1st part	2nd part	3rd part	4th part	5th part	6th part	7th part	8th
Overview	Dashboards	IOT App, IOT Network	Data Analytics	Data Ingestion processes	System and Deploy Install	Smart City API: Web & Mob. App	Design and Develop Smart Solutions
							
							



Note on Training Material

- **Course**: https://www.snap4city.org/944
  - Introductionary course to Snap4City technology
  - Italian version CN MOST: https://www.snap4city.org/1068
- **Older version Course** https://www.snap4city.org/577
  - Full training course with much more details on mechanisms and a wider set of cases/solutions of the Snap4City Technology
- **Documentation** includes a deeper round of details
  - Snap4City Platform Overview:
    - https://www.snap4city.org/drupal/sites/default/files/files/Snap4City-PlatformOverview.pdf
  - Development Life Cycle:
    - https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf
  - Client Side Business Logic:
    - https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf
- **On line cases and documentation**:
  - https://www.snap4city.org/108
  - https://www.snap4city.org/78
  - https://www.snap4city.org/426


Snap4City Advisor Assistant
User: roottooladmin1, Org: DISIT
Role: RootAdmin, Level: 7
LOGOUT
Switch to Legacy Layout
Dashboards (Public)
My Snap4City.org
Tour Again
www.snap4solutions.org
ダッシュボード
My Dashboards in All Org.
Dashboards of My Organization
My Dashboards in My Organization
My Data Dash Dev OpenSrcDash
My Data Dashboard OpenSrcDash
Extra Dashboard Widgets
Notificator
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Entity Directory and Devices
Resource Manager
Development Tools
Privacy Policy
Cookies Policy
Dark
UNIVERSITÀ DEGLI STUDI
DINFO
DISIT
23 -
24 - Dashboard 2
25 - what one should be to adopt snap4city - 04:39PM on December 06, 2024
26 - TEST SOURCES
27 - How can I visualize a map on a dashboard? - 05:50PM on December 06, 2024
28 - How can I visualize a map on a dashboard? - 06:46PM on December 06, 2024
29 - which is the HPC support on snap4city - 09:30PM on December 06, 2024
30 - csbl
31 - Data Analytics
32 - Industria 4.0
33 - Ciao qual'è l'API per prendere le user info da access token? - 04:02PM on December 10, 2024
34 - quale è il modo corretto di definire un modello dati in snap4city - 09:39AM on December 11, 2024
35 - che cosa è snap4city - 04:28PM on December 11, 2024
36 - install
37 - may you provide python code to call smart city API - 03:08PM on December 13, 2024
Coming soon!
SnapAdvisor
Search more
how to build a smart application
To build a smart application using Snap4City, follow these steps:
1. Develop Processing Logic / IoT Apps: Create your Processing Logic / IoT Apps using Node-RED in JavaScript. This involves handling data ingestion, adapters, transformations, wrappers, business logic, transcoding, integration, interoperability, algorithms, etc.
2. Create Data Analytics: Implement algorithm and processing in RStudio or Python, including machine learning (ML), artificial intelligence (AI), explainable AI (XAI), etc.
3. Design User Interface: Build dashboards, client-side business logic, synoptics, widgets, templates, styles, etc. If necessary, implement client-side business logics in JavaScript on dashboard widgets.
4. Implement Server-Side Business Logics: Use Processing Logic as Node-RED and JavaScript for server-side business logics.
5. Manage Data Instances: Handle high-level types' data instances.
6. Ensure Security and Compliance: All data streams should be protected, authenticated, and
Type a question on Snap4City documentation

Dashboards (Public)

My Snap4City.org | Tour Again

www.snap4solutions.org
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
Extra Dashboard Widgets
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Entity Directory and Devices
Resource Manager
Development Tools
Management
Decision Support Systems
Deploy and Installation
Help and Contacts
Documentation and Articles

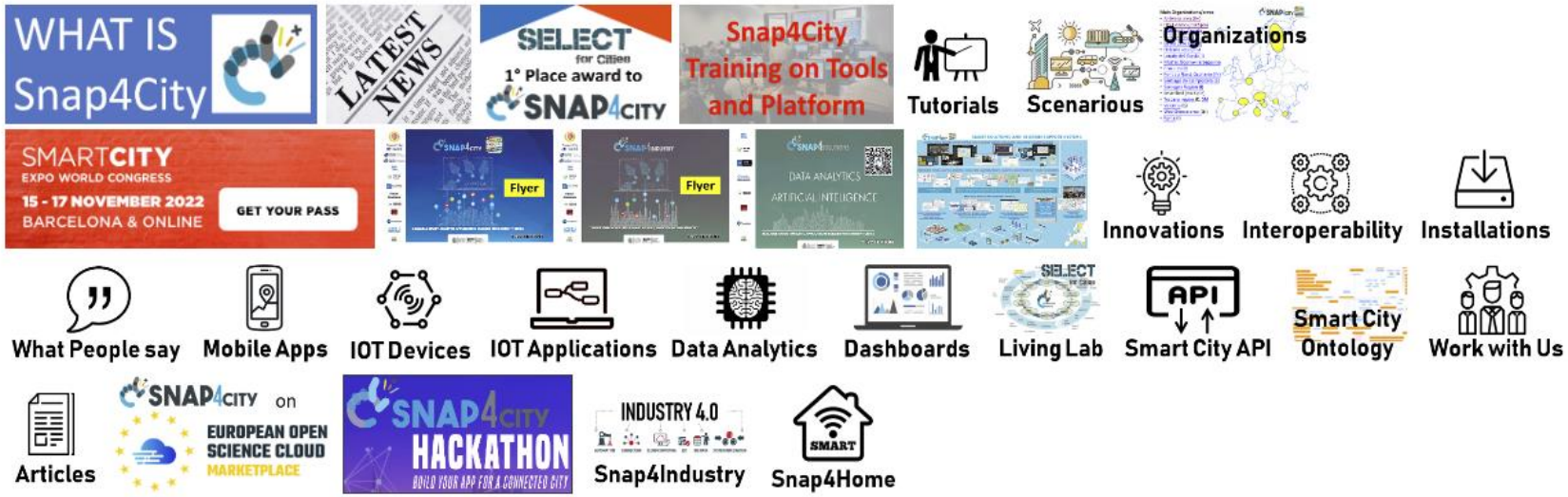
Privacy Policy | Cookies Policy



Home / Snap4City: Smart aNalytic APp builder for sentient Cities and IOT

Snap4City: Smart aNalytic APp builder for sentient Cities and IOT

You can't delete this newsletter because it has not been sent to all its subscribers.



- TECHNICAL OVERVIEW: https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- Development Life Cycle: https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf
- Client-Side Business Logic Widget Manual: https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf
- Booklet Data Analytics Snap4Solutions: https://www.snap4city.org/download/video/DRL_SNAP4SOLU.pdf

Username: paolo.disit

Search

Search

-Any-



Organization Groups

DISIT
- Developer
- Operativo

Updates on

booklets

- Smart City



https://www.snap4city.org/download/video/DPL_SNAP4CITY.pdf

- Industry



https://www.snap4city.org/download/video/DPL_SNAP4INDUSTRY.pdf

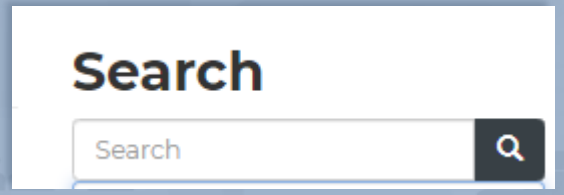
- Artificial Intelligence

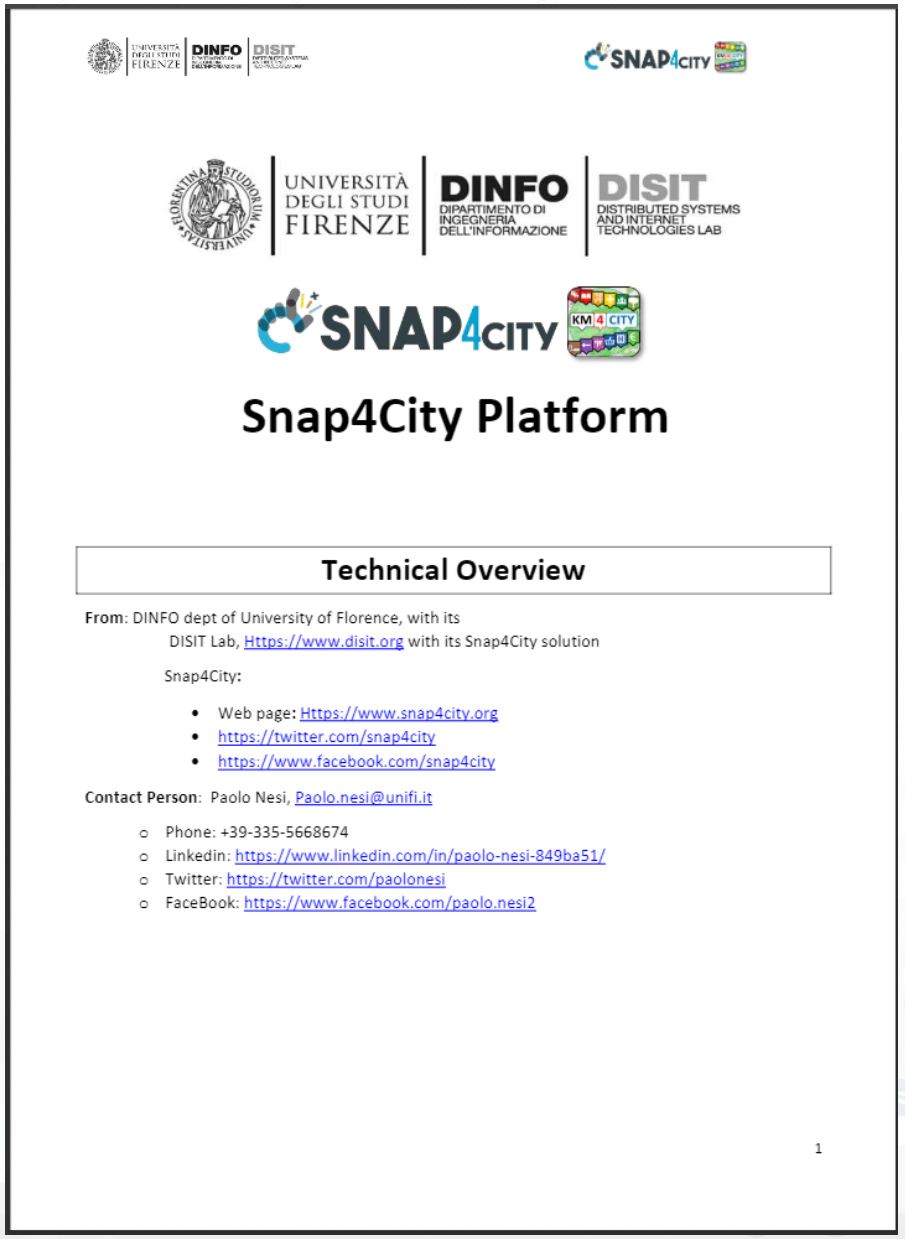


https://www.snap4city.org/download/video/DPL_SNAP4SOLU.pdf

The Platform

- **Free Registration on Snap4City.org**
  - Please select DISIT ORG to be sure to access at the examples
  - Most of the cities / tenant are private and they do not left much visible
- **What you get** is probably the 10% of what is on the platform ☺
- **Training**: https://www.snap4city.org/944
- **Scenarious**: https://www.snap4city.org/4
- **Publications**: https://www.snap4city.org/426
- **WEB pages**: https://www.snap4city.org/78
- ***SEARCH on the right side***





UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

SNAP4CITY KM4CITY

Snap4City Platform

Technical Overview

From: DINFO dept of University of Florence, with its
DISIT Lab, Https://www.disit.org with its Snap4City solution

Snap4City:

- Web page: Https://www.snap4city.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city

Contact Person: Paolo Nesi, Paolo.nesi@unifi.it

- Phone: +39-335-5668674
- Linkedin: https://www.linkedin.com/in/paolo-nesi-849ba51/
- Twitter: https://twitter.com/paolonesi
- FaceBook: https://www.facebook.com/paolo.nesi2

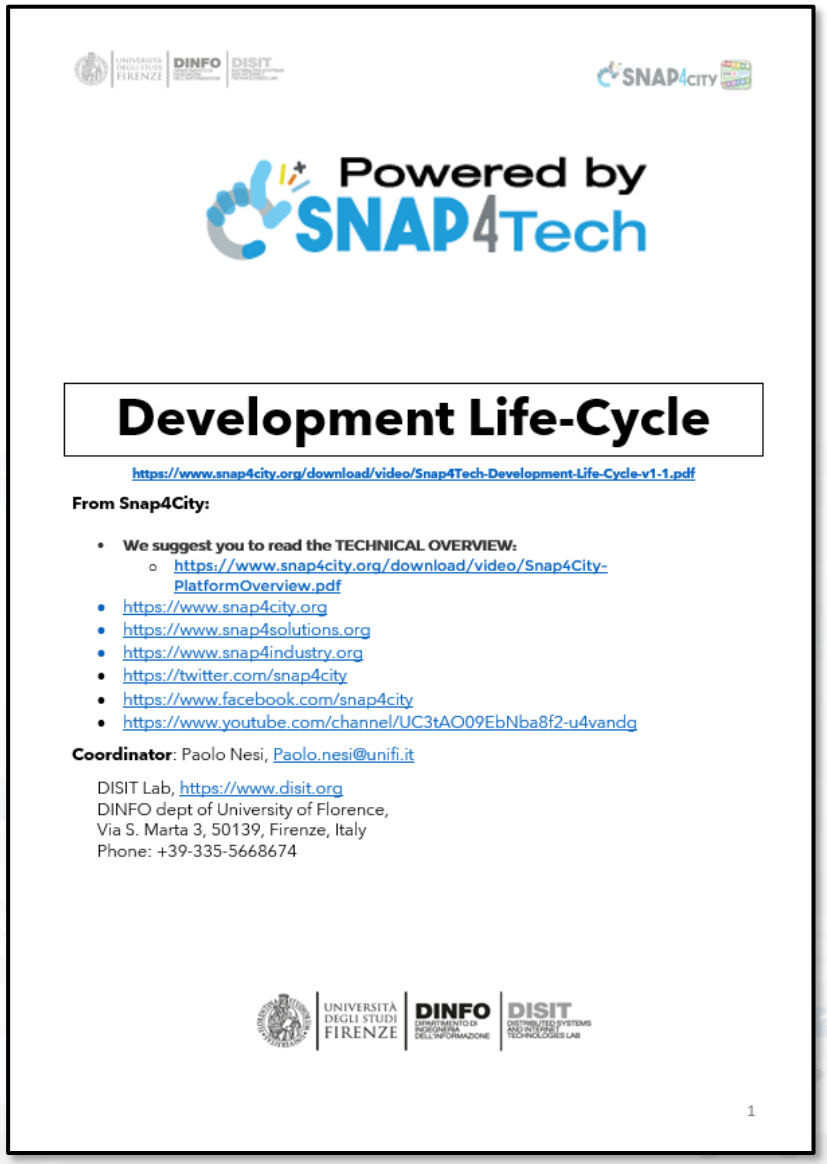
1

Tech Overview

- https://www.snap4city.org/drupal/sites/default/files/files/Snap4City-PlatformOverview.pdf

Development

https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf



Powered by SNAP4Tech

Development Life-Cycle

https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle-v1-1.pdf

From Snap4City:

- **We suggest you to read the TECHNICAL OVERVIEW:**
  - https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- https://www.snap4city.org
- https://www.snap4solutions.org
- https://www.snap4industry.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city
- https://www.youtube.com/channel/UC3tAO09EbNba8f2-u4vandg

Coordinator: Paolo Nesi, Paolo.nesi@unifi.it

DISIT Lab, https://www.disit.org
DINFO dept of University of Florence,
Via S. Marta 3, 50139, Firenze, Italy
Phone: +39-335-5668674

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

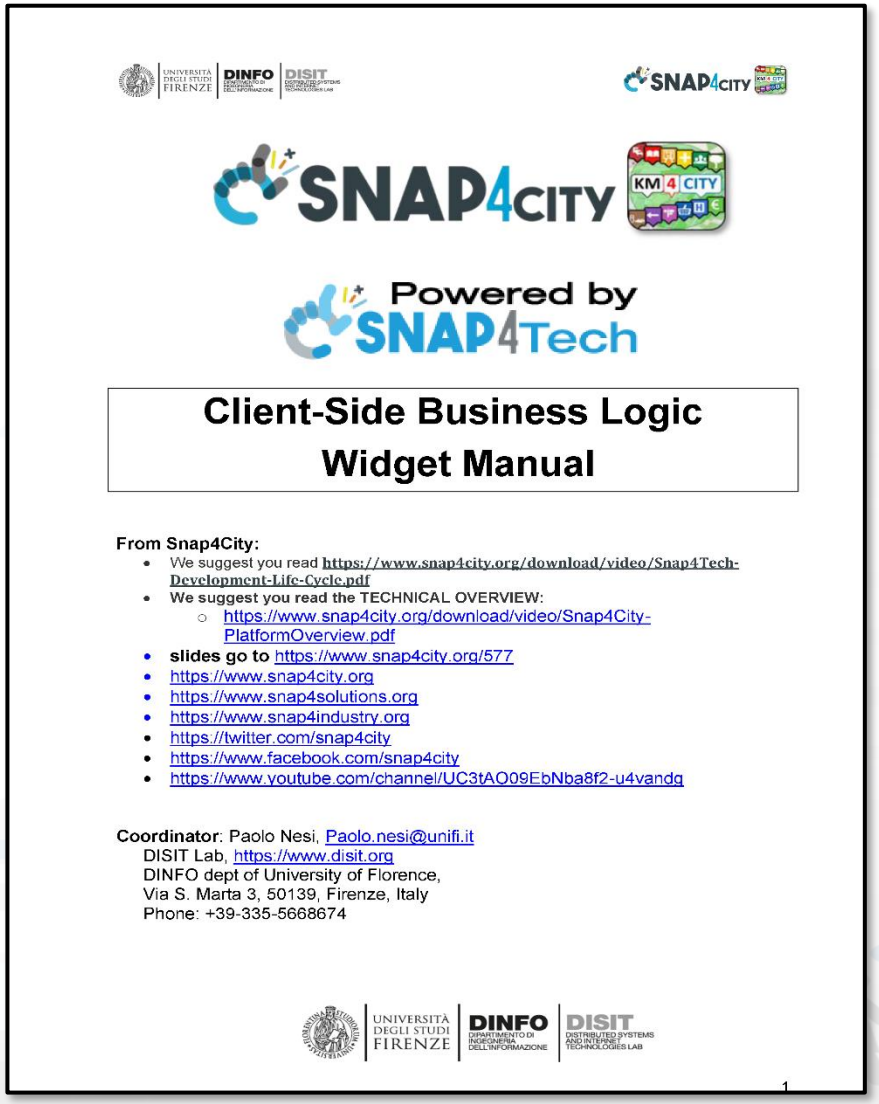
1

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB DISTRIBUTED DATA INTELLIGENCE AND TECHNOLOGIES LAB

CSBL

SNAP4CITY KM4CITY

Client Side Business Logic



SNAP4CITY KM4CITY

Powered by SNAP4Tech

Client-Side Business Logic Widget Manual

From Snap4City:

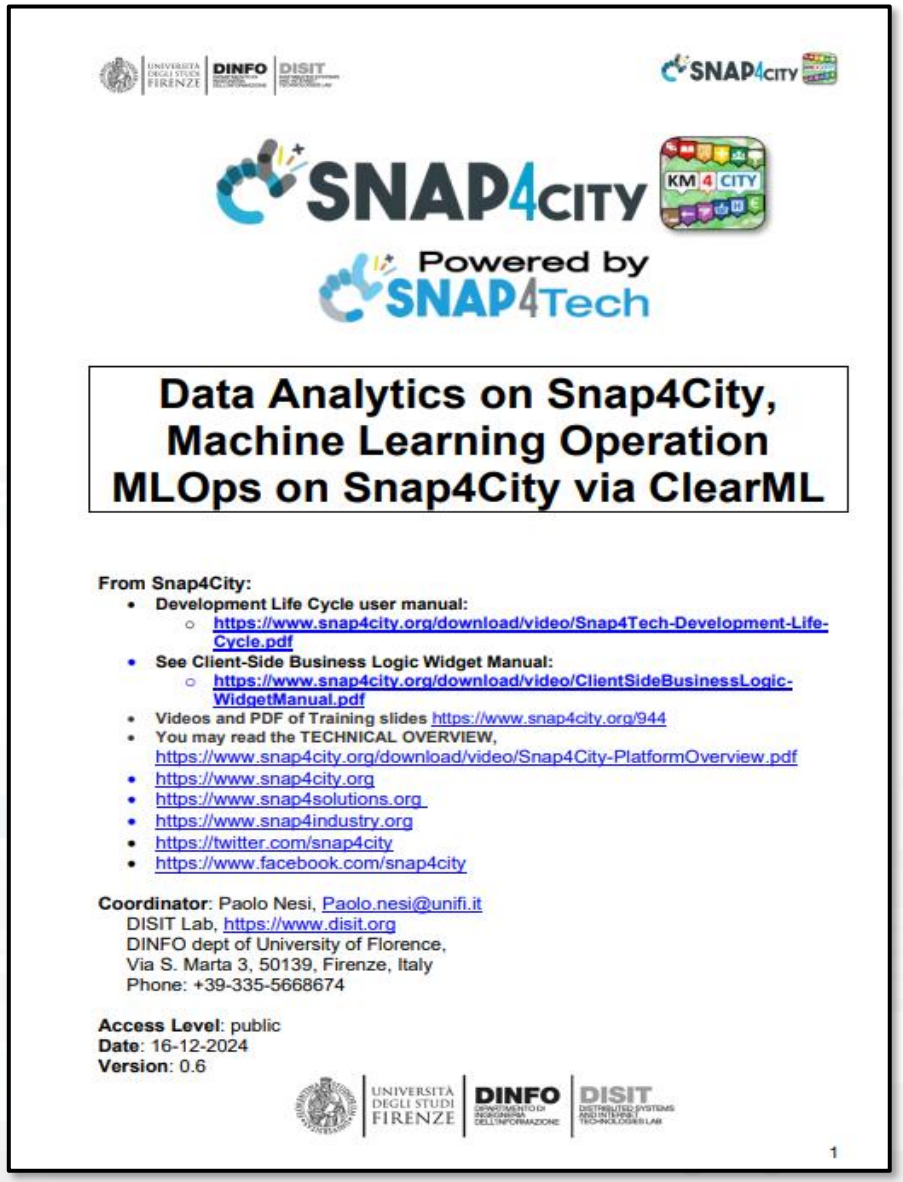
- We suggest you read **https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf**
- **We suggest you read the TECHNICAL OVERVIEW:**
  - https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- **slides go to** https://www.snap4city.org/577
- https://www.snap4city.org
- https://www.snap4solutions.org
- https://www.snap4industry.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city
- https://www.youtube.com/channel/UC3tAQ09EbNba8f2-u4vandg

Coordinator: Paolo Nesi, Paolo.nesi@unifi.it
DISIT Lab, https://www.disit.org
DINFO dept of University of Florence,
Via S. Marta 3, 50139, Firenze, Italy
Phone: +39-335-5668674

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf



SNAP4CITY

Powered by SNAP4Tech

Data Analytics on Snap4City, Machine Learning Operation MLOps on Snap4City via ClearML

From Snap4City:

- **Development Life Cycle user manual:**
  - https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf
- **See Client-Side Business Logic Widget Manual:**
  - https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf
- **Videos and PDF of Training slides** https://www.snap4city.org/944
- **You may read the TECHNICAL OVERVIEW,** https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- https://www.snap4city.org
- https://www.snap4solutions.org
- https://www.snap4industry.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city

Coordinator: Paolo Nesi, Paolo.nesi@unifi.it
DISIT Lab, https://www.disit.org
DINFO dept of University of Florence,
Via S. Marta 3, 50139, Firenze, Italy
Phone: +39-335-5668674

Access Level: public
Date: 16-12-2024
Version: 0.6

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

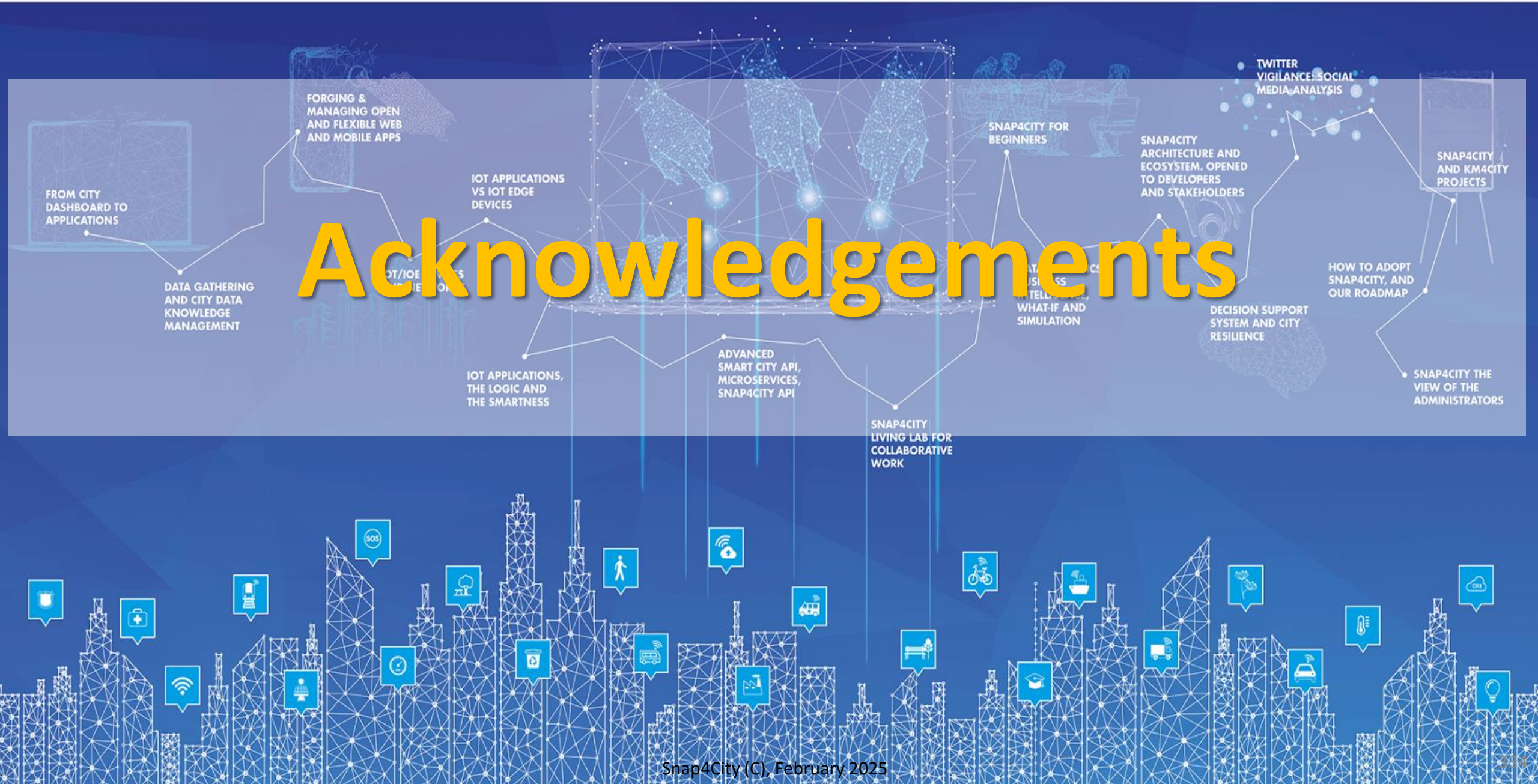
MLOperation

https://www.snap4city.org/download/video/Snap4City-MLOps-Manual.pdf

Overview



- https://fiware-foundation.medium.com/snap4city-fiware-powered-smart-app-builder-for-sentient-cities-acfe24df49d5
- https://www.snap4city.org/drupal/sites/default/files/files/FF_ImpactStories_Snap4City.pdf



Acknowledgements

DISIT lab roadmap vs model and tools' usage

2013
Km4City Ontology 1.1
- Tuscany, Road Graph
- Mobility
- culture, tourism
- Events
- Parking
- Services
- Linked open graph

2014
- Weather Forecast
- Real Time Wi-Fi
- Entertainment
- LOD

- Twitter Vigilance
- Social Media Analytics, Sentiment Analysis

Km4City 1.4

2015
Km4City 1.5
- Resilience Decision Support
- Smart First Aid
- User Behaviour Analysis, predictions
- Risk Analysis

ckan, resolute, H2020 (2015-18)

2016
(2016-21) H2020 REPLICATE

Km4City 1.6.4
- Smart Energy
- Sustainable Mobility
- Control Room
- Dashboard

Km4City 1.6.2

Sii-Mobility (2016-21) MIUR
SII-MOBILITY SCN
- Infomobility
- Mobile App
- Routing
- Multimodality

FIWARE

2017
Sii-Mobility
- Origin-Destination and trajectories
- Traffic Reconstruction
- Offer Analysis
- OBU, smart devices

E015 digital ecosystem (Powered by EXPO MILANO 2015)

(2016-19) MIUR
GHOST SIR
- Sardinia Region Smart City Strategies and plan

GREEN IMPACT C.A.P.A.C.I.T.Y.
GREEN IMPACT
POR FESR 2014-2020
- Industry 4.0
- Critical Plant
- Monitoring

- IOT/IOE, IOT App
- Living Lab
- Maker Support
- IOT Edge
- Smart City IOT
- GDPR,
- Privacy & Security

life weee (2017-20)
- Smart Waste

2018
MOSAiC MObility 4.0 for SmArt (i) City (2018-20)
- Mobility Demand / Offer Analytics and Strategy

Sii-Mobility
- User engagement
- Bike Sharing
- Data Analytics ++
- Social Predictions
- OBD2

IOT/IOE
Km4City 1.6.6
Node-RED

SNAP4CITY SELECT for Cities H2020 (2017-19)

2019
5G (2018-21)
- 5G tech
- Energy
- Industry 4.0
- Synoptics

TRAFAIR
CEF (2018-21)
- Traffic and Mobility Impact on Pollution
- NOX predictions

EUROPEAN OPEN SCIENCE CLOUD

SIRIUS

LAID Smart Bed
- Smart Health

GREEN FIELD PEAS Soda4.0 Optimization and Automated Decision System 4.0
- Industry 4.0

bee smart city
- Smart Lonato

Reverberi Enetec Gruppo MPES

SNAP4CITY

SODA

Winner of Select4Cities PCP



SNAP4CITY roadmap

2020

- Smart Tourism
- 6 Pilots
- Data Analytics
- Extended platform

- Smart Mobility
- PISA, PUMS
- Living lab

Km4City 1.6.7

2021

Smart Ambulance (2021-22)

Enterprise (2021-22) Industry 4.0

Sii-Mobility Contract

JRC Contract

PC4City (2020-21) Monitoring Terrain

Winner of Open Data Challenge of enel x

- Smart Light
- Sweden

2022

Almafluida Industry 4.0 (2021-22)

AMPERE (2021-22) Industry 4.0

SYN-RG-AI SmartCity

GRUPPO PRETTO Industry 4.0

uni.systems SmartCity, 2021-23

AXIS collab SmartCity

Asymmetrica Smart City, 2022-23

2023

JRC Contract, 2022-23

ART-ER Contract, 2022-23

Filippi 2022-2023

enel x Contract, 15min

IMPETUS Security and Risk

Smartea

Italferr, Smart City

CN MOST, 2022-26

EI THE, 2022-26

G. Agile, 2021-23

TUSCANY XD EDIH 2023-26 — Finanziato dall'Unione europea NextGenerationEU

Merano, smart light

OceanRace, Genova, AWS

Cuneo, smart city

2024

Km4City 1.6.8

TOURISMO

AMMIRARE

ELLIE IA 2025-2027

UrbanDT4TF

JRC Contract, 2024-25

CAI4DSA

FAIR Future Artificial Intelligence Research

OPTIFaaS

SASUAM

Rhodes, smart city

eShare UNIFI TUSS

Interreg Euro-MED — Co-funded by the European Union

Interreg MARITTIMO-IT FR-MARITIME

2024/25

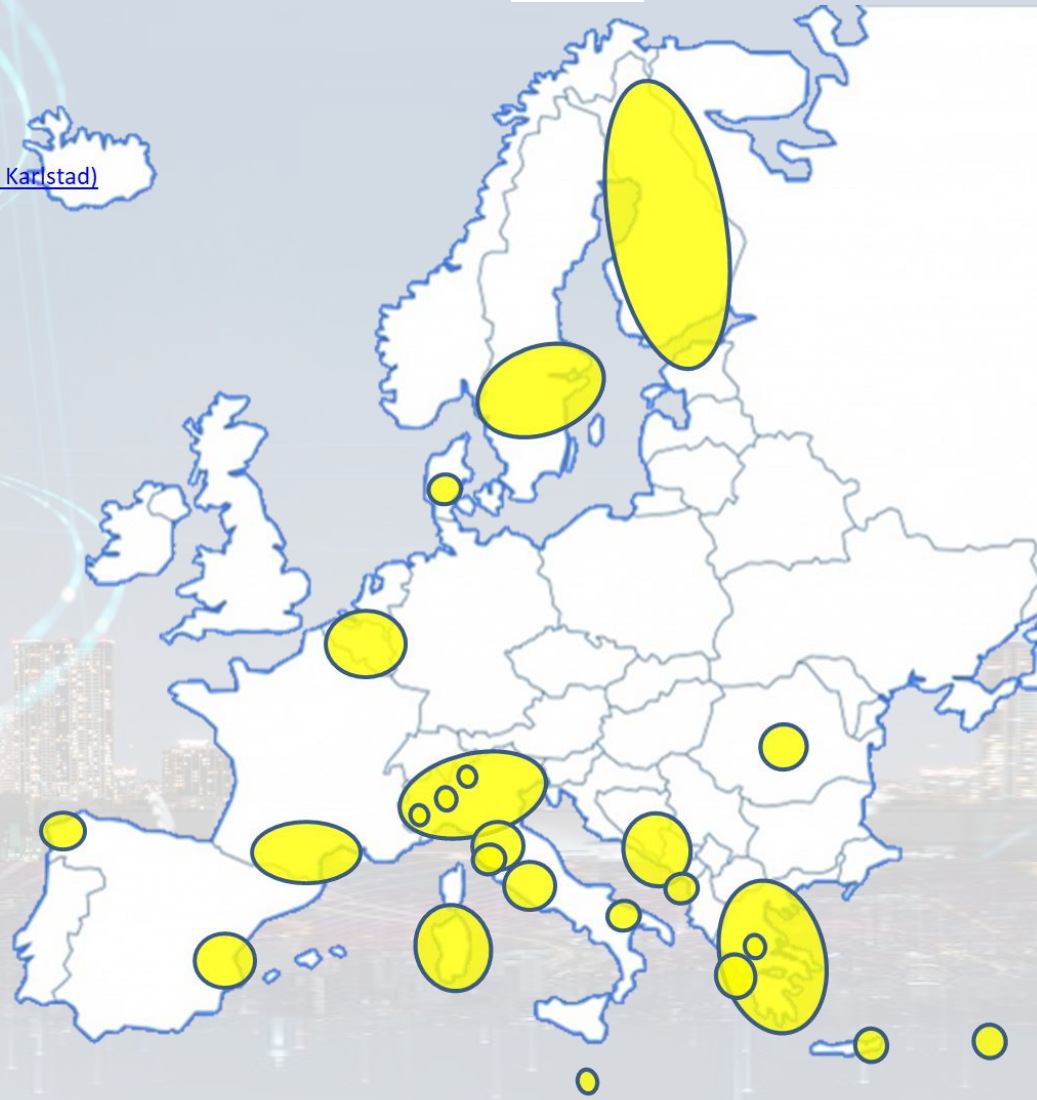
- **UrbanDT4TF**, CN HPC: Digital Twin mobility
- **DI-DTPlatform**, CN HPC: Digital Twin, mobility, environment
- **Sasuam**, CN MOST, PNRR: AI, mobility
- **OPTIFaaS**, CN MOST, PNRR: AI, mobility, DSS
- **LeverageOPTIFaaS**, CN MOST: PNRR, mobility
- **TOURISMO**, Interreg, EC: Tourism, NLP, DSS
- **ELLIE**, Horizon Europe, EC: AI, VR
- **CN MOST**, PNRR: sustainable mobility, platform
- **ISPRA JRC contract**, EC: DSS, SOC, control room, energy
- **The IE**, PNRR: AI, NLP, Legal Aspects
- **AMMIRARE**, Interreg, EC: AI, environment, Big Data
- **CAI4DSA**, FAIR PE1, PNRR: AI, Neuro-Symbolic, PINN, NG-DSS
- **SADI-MIAC**, RT, partner: AI, Tourism, Retail, Computer Vision
- **Energia**, RT, conv: AI, PINN, DSS
- **RFI contract**: mobility, AI, DSS
- **PRIN UNICagliari**: mobility, DSS
- **Talent Hub**, ECRF, conv: NLP, match demand vs offer

https://www.Snap4City.org

- 11 running installations in Europe
  - Snap4.city.org, Greece, Merano, Cuneo, …
  - Toscana, Pisa, Sweden, ISPRA, Snap4.eu,
  - Altair, Italmatic, Romania, ….
- 16 projects, 12 pilots on 10 Countries
  - >40 cities/area
- **Widest MULTI-tenant deploy has**
  - 24 Organizations / tenant
  - > 8850 users on
  - > 1800 Dashboards
  - > 17 mobile Apps
  - **> 2.2 Million of structured data per day**
  - > 580 IoT Applications/node-RED
  - > 750 web pages with training
  - > 75 videos, training videos

Main Organizations/areas

- Antwerp area (Be)
- Bari (I)
- Bisevo, Croatia
- Bologna (I)
- Brasov (Ro), by ICEBERG
- Capelon (Sweden: Västerås, Eskilstuna, Karlstad)
- Cuneo (I)
- DISIT demo (multiple)
- Dubrovnik, Croatia
- Firenze area (I)
- Garda Lake area (I)
- Greece (Gr)
- Helsinki area (Fin)
- Limassol (Cy)
- Livorno area (I)
- Lonato del Garda (I)
- Malta (Malta)
- Merano (I)
- Modena (I)
- Mostar, Bosnia-Herzegovina
- Oslo & Padova (Impetus)
- Pisa area (I)
- Pistoia (I)
- Pont du Gard, Occitanie (Fr)
- Prato (I)
- Rhodes (Gr)
- Roma (I)
- Santiago de Compostela (S)
- Sardegna Region (I)
- Siena (I)
- SmartBed (multiple)
- Toscana Region (I), SM
- Valencia (S)
- Venezia area (I)
- WestGreece area (Gr)



- **+ Israel, Colombia, Brasile, Australia, India, China, etc.**

SNAP4CITY

KM 4 CITY

100% OPEN SOURCE

Be smart in a SNAP!

SMARTCITY EXPO WORLD CONGRESS

SNAP4 Appliances and Dockers **Installations**

CONTACT

DISIT Lab, DINFO: Department of Information Engineering
Università degli Studi di Firenze - School of Engineering

Via S. Marta, 3 - 50139 Firenze, ITALY
https://www.disit.org

www.snap4city.org

Email: snap4city@disit.org

Office: +39-055-2758-515 / 517
Cell: +39-335-566-86-74
Fax.: +39-055-2758570

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB