100% OPEN SOURCE

SNAP4CITY

KM4CITY

SNAP4 Appliances and Dockers Installations

EU GDPR COMPLIANT

PEN Test Passed

Powered by FIWARE

SNAP4CITY on EUROPEAN OPEN SCIENCE CLOUD MARKETPLACE

Node-RED

Be smart in a SNAP!

How to Design and Develop Smart Solutions

January 2023, Course
https://www.snap4city.org/577

SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES

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

UNIVERSITÀ DEGLI STUDI FIRENZE

DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

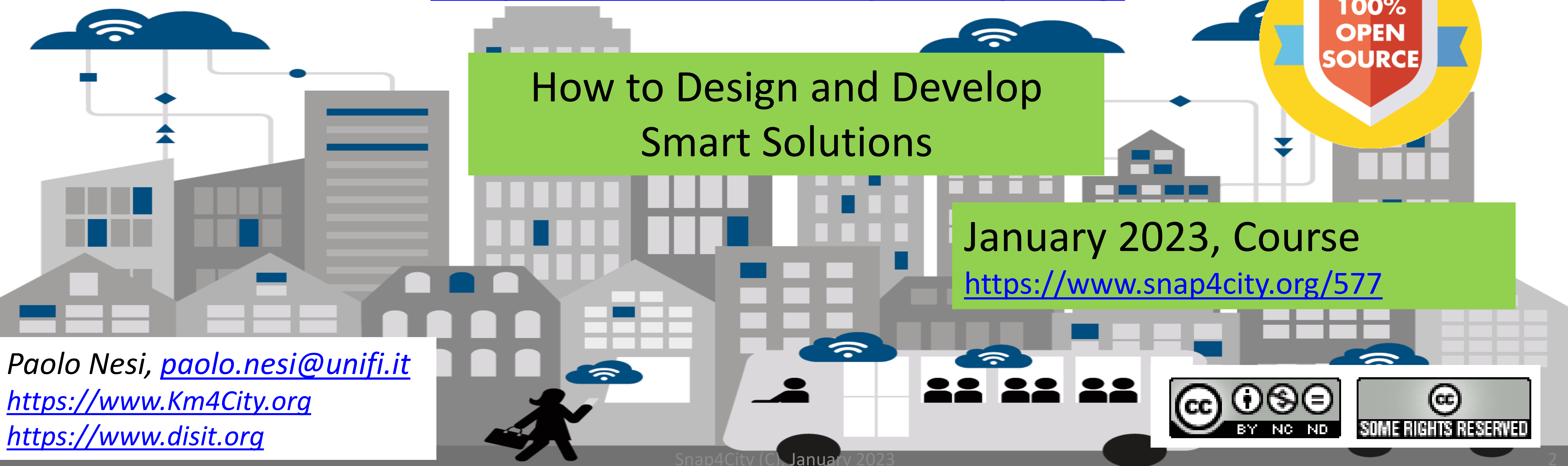
DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

SNAP4CITY

KM4CITY Powered by

scalable Smart aNalytic APplication builder for sentient Cities: for Living Lab and co-working with Stakeholders

https://www.Snap4City.org



100% OPEN SOURCE

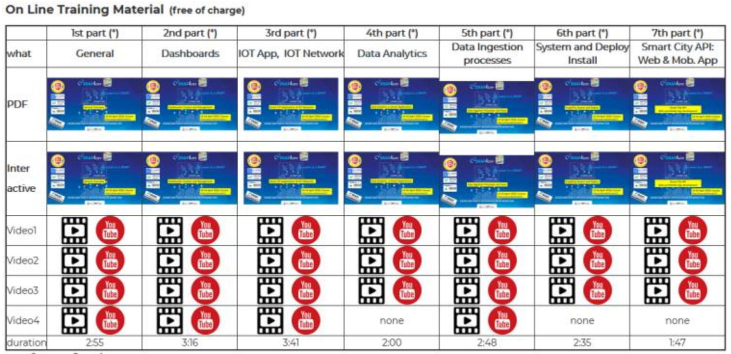
How to Design and Develop Smart Solutions

January 2023, Course
https://www.snap4city.org/577

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

ATTENTION!!!

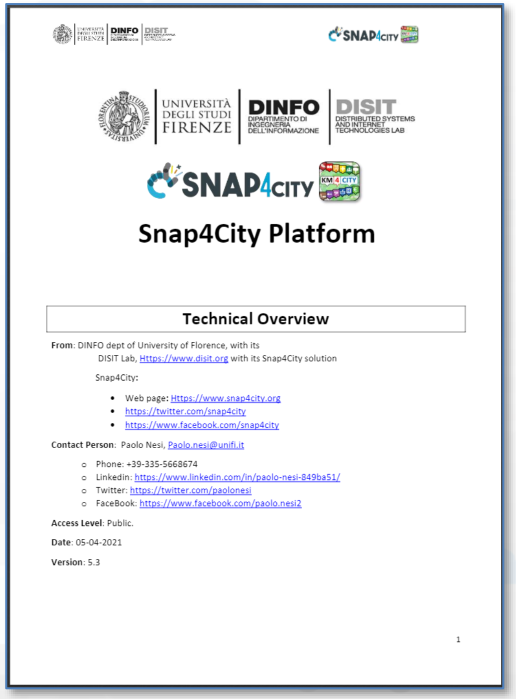
- *These slides are only a overview of the training course of Snap4City, Snap4Industry, Snap4Tech.*
- *Full training course access to dedicated web page, slide, video, documents and on line documentation which are reporting many more details, examples and functionalities.*



On Line Training Material (free of charge)

	1st part (*)	2nd part (*)	3rd part (*)	4th part (*)	5th part (*)	6th part (*)	7th part (*)
what	General	Dashboards	IOT App, IOT Network	Data Analytics	Data Ingestion processes	System and Deploy Install	Smart City API: Web & Mob. App
PDF							
Inter active							
Video1	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube
Video2	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube
Video3	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube
Video4	▶ You Tube	▶ You Tube	▶ You Tube	none	▶ You Tube	none	none
duration	2:55	3:16	3:41	2:00	2:48	2:35	1:47

https://www.snap4city.org/577



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

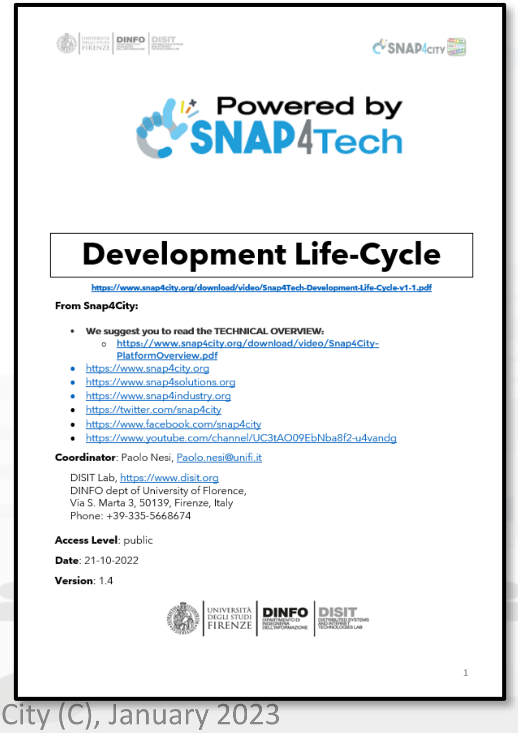
Access Level: Public.

Date: 05-04-2021

Version: 5.3

1

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

Access Level: public

Date: 21-10-2022

Version: 1.4

1

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

https://www.snap4city.org/577

On Line Training Material (free of charge)

SNAP4CITY KM4CITY

	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								

Video1	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube
Video2	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube
Video3	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube	▶ You Tube
Video4	▶ You Tube	▶ You Tube	▶ You Tube	none	▶ You Tube	none	none

General Overview of the full Course

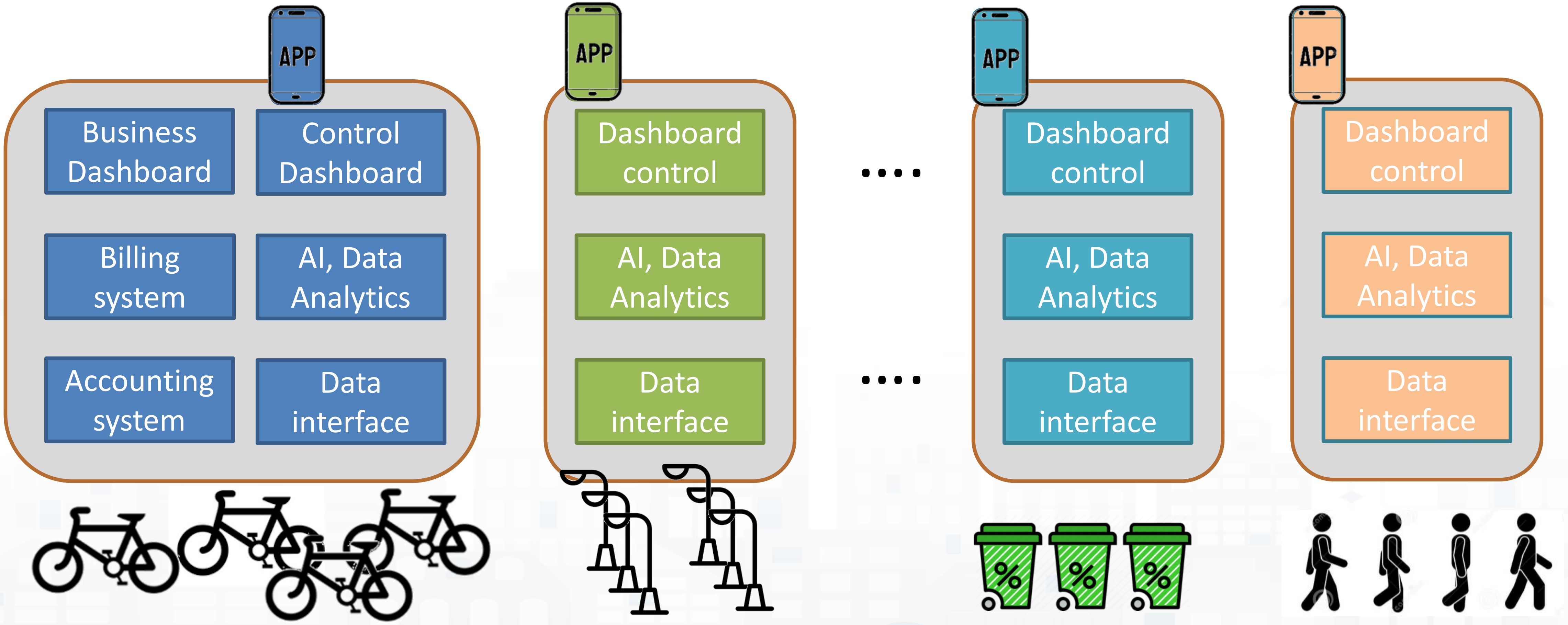
1. *General Overview*
2. ***Dashboards** Creation and Management, **Business Intelligence***
3. ***Processing Logic/IOT App** development, Entities / IOT Devices, IOT Networks*
4. ***Data Analytics**, in R Studio, in Python, how to Exploit and Manage Data Analytics in IOT Applications*
5. ***Data Ingestion**, Data Warehouse, Data Gate, IOT Device Data ingestion, IOT App for Data Ingestion, **Interoperability**, etc.*
6. *Snap4City **Installation, Extension, Administration***
7. ***Smart city API** (internal and external) **Web and Mobile App development** tool kit*
8. *How to **Design and Develop Smart Solutions***

A number of the training sections include exercitations

Updated versions on: https://www.snap4city.org/577

See also courses in ITALIANO: https://www.snap4city.org/485

Avoiding to have a collection of verticals



APP

Business Dashboard

Control Dashboard

Billing system

AI, Data Analytics

Accounting system

Data interface

APP

Dashboard control

AI, Data Analytics

Data interface

....

....

APP

Dashboard control

AI, Data Analytics

Data interface

APP

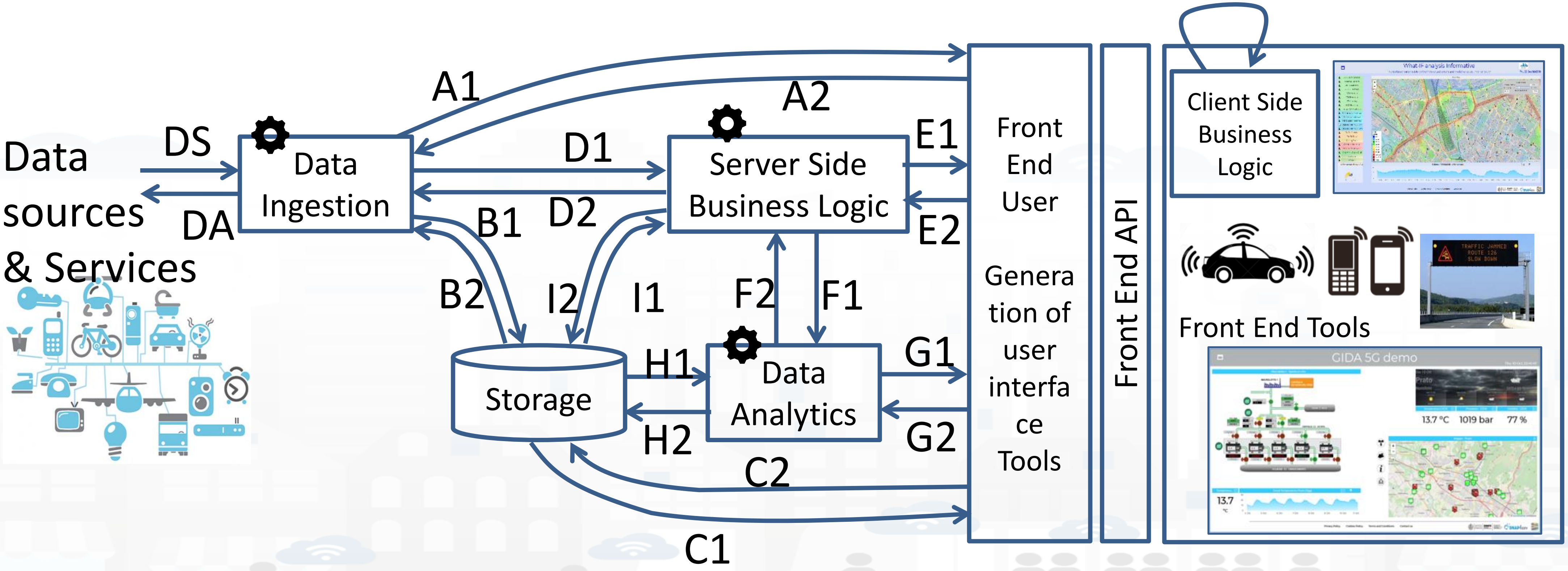
Dashboard control

AI, Data Analytics

Data interface

Simplifying the development and integration of verticals

Coverage of Data and Control Flows



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

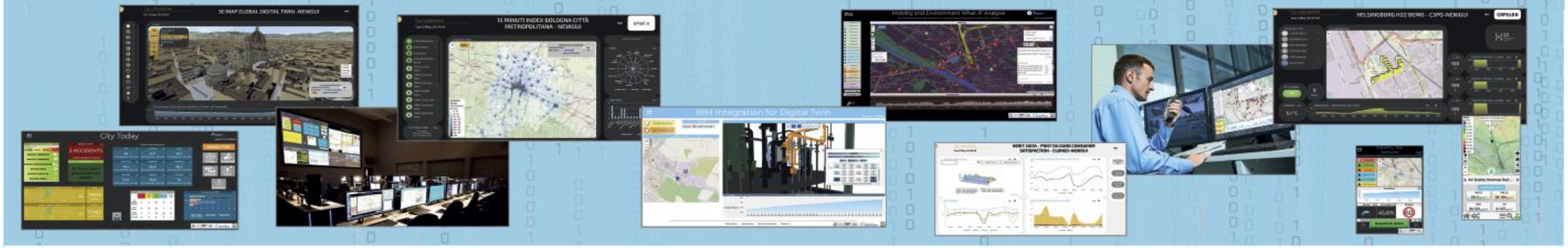
E015 digital ecosystem

NVIDIA

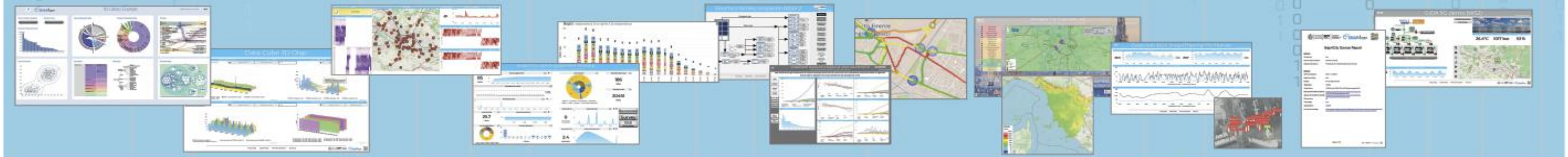
SNAP4CITY KM4CITY

SMART SOLUTIONS AND DECISION SUPPORT SYSTEMS

CONTROL ROOMS - DECISION SUPPORT SYSTEMS - WHAT-IF ANALYSIS - BUSINESS INTELLIGENCE - SIMULATIONS - SMART APPLICATIONS


3D MAP GLOBAL DIGITAL TWIN -NEWGUI
15 MINUTI INDEX BOLOGNA CITTÀ METROPOLITANA - NEWGUI
City Today
BIM Integration for Digital Twin
Mobility and Environment What-IF Analysis
HELSINGBORG H22 DEMO - C3PO-NEWGUI

DASHBOARDS - VISUAL ANALYTICS - SYNOPTICS - DIGITAL TWIN - GRAPHICAL WIDGETS - ANALYTICS - GUI CUSTOM STYLES - VISUAL PROGRAMMING

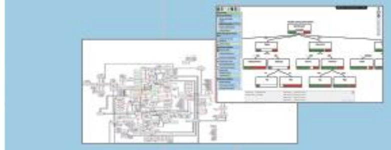


DASHBOARDS, WIDGETS TEMPLATES

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 ...

ANY: DATA, BROKER, NETWORK AND VERTICAL



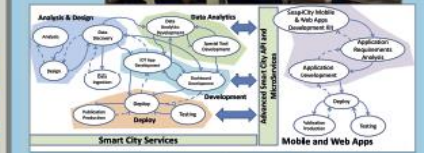
EXPERT SYSTEM, KNOWLEDGE BASE
SEMANTIC REASONING
SMART DATA MODEL
IOT DEVICE MODELS, STORAGE

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

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

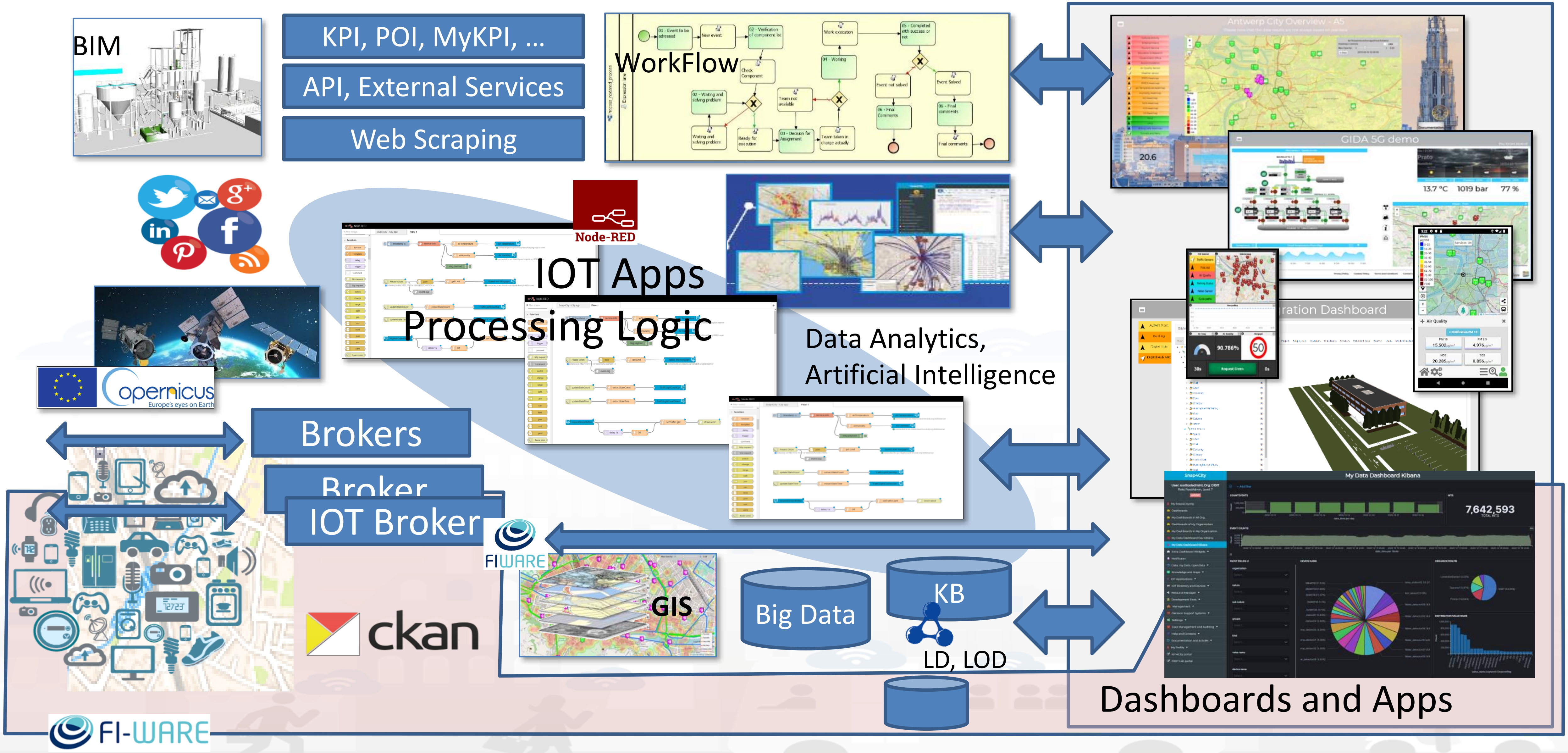
Native and External Applications

- Smart Parking
- Smart Light
- Smart Waste
- Smart Energy
- Social Media Analysis
- ...



METHODOLOGIES
LIVING LABS
COURSES AND COMMUNITY
DEVELOPMENT TOOLS

Concept - 2023



BIM

KPI, POI, MyKPI, …

API, External Services

Web Scraping

WorkFlow

Node-RED

IOT Apps

Processing Logic

Data Analytics, Artificial Intelligence

Brokers

Broker

IOT Broker

FIWARE

GIS

Big Data

KB

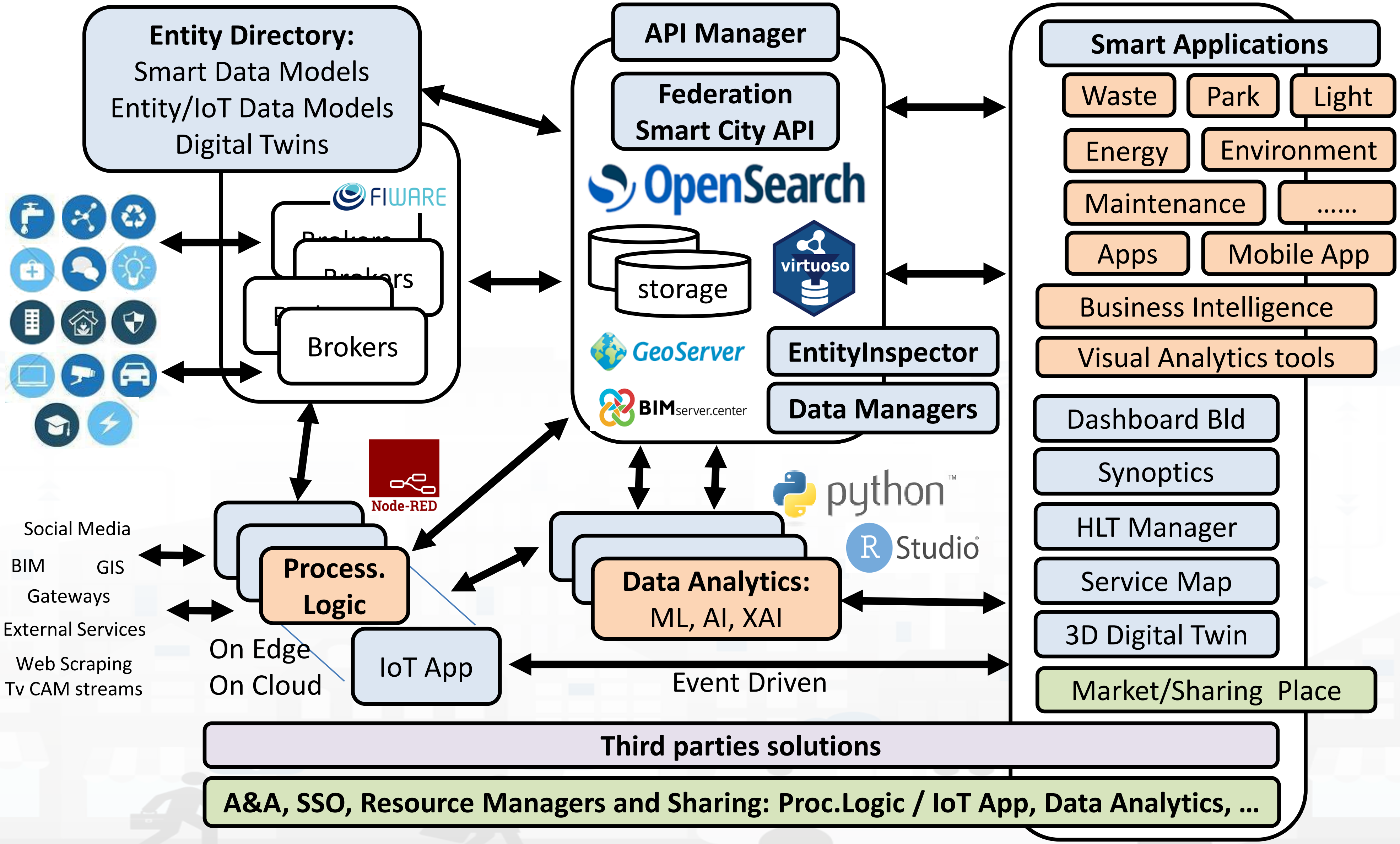
LD, LOD

ckan

Dashboards and Apps

FI-WARE

Tech Arch - 2023



Principles

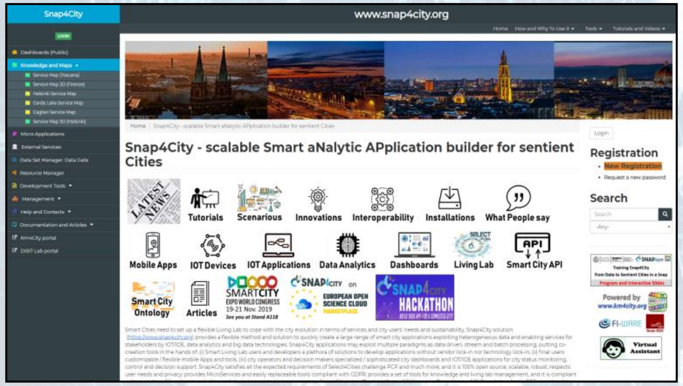
- **Smart Applications can be easily developed exploiting the cloud infrastructure by producing only:**
  - **Processing Logic / IoT App** with almost no coding activities
  - **Data Analytics** in Python or Rstudio
  - **Dashboards** with almost no coding activities.
- **→ Orange parts of the previous figure slide are those usually developed,**
  - all the rest, is part of the provided microservices and infrastructure.

- **Third party applications can dialog with the solutions via**
  - **Smart City API**, Swagger: https://www.km4city.org/swagger/external/
  - **Brokers/IoT Brokers**, for example for NGSI Orion Broker: https://www.km4city.org/swagger/external/?urls.primaryName=Orion%20Broker%20K1-K2%20Authentication%20API
  - **Processing Logic / oT App** any protocols: https://www.snap4city.org/65 They can also expose some specific API, custom made

How to adopt Snap4City

SNAP4CITY

[Https://www.snap4city.org](https://www.snap4city.org)



Snap4City

www.snap4city.org

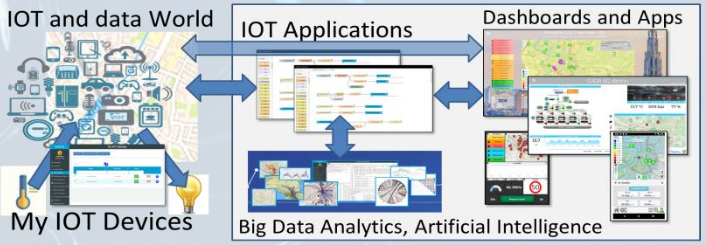
Snap4City - scalable Smart aNalytic APplication builder for sentient Cities

Registration

- New Registration
- Request a new password

Smart Cities need to set up a flexible Living Lab to cope with the city evolution in terms of services and city users' needs and sustainability. Snap4City solution (https://www.snap4city.org) provides a flexible method and solution to quickly create a large range of smart city applications exploiting heterogeneous data and enabling services for stakeholders by IOT/IOE, data analytics and big data technologies. Snap4City applications may exploit multiple paradigms as data driven, stream and batch processing, putting co-creation tools in the hands of: (i) Smart Living Lab users and developers a plethora of solutions to develop applications without vendor lock-in nor technology lock-in, (ii) final users customizable / flexible mobile Apps and tools, (iii) city operators and decision makers specialized / sophisticated city dashboards and IOT/IOE applications for city status monitoring, control and decision support. Snap4City satisfies all the expected requirements of Select4Cities challenge PCP and much more, and it is 100% open source, scalable, robust, respects user needs and privacy; provides MicroServices and easily replaceable tools compliant with GDPR; provides a set of tools for knowledge and living lab management, and it is compliant

On your premise



KM 4 CITY

Powered by SNAP4Tech

Smart City as a Service

- Supporting Org
- 100% Open-Source Platform: Github
- Further developments
- Publishing Appliances and Dockers
- Training courses, docs
- Consulting
- Forums
- Etc.

Download and deploy

Installation on your premise

- **Virtual Machines or Dockers**
- Different configurations
  - From small to scalable
  - Exploiting your legacy tools
  - Interoperable with any tool
- No vendor lock-in, No tech lock-in

Mixed solutions! For example:

- Start on Cloud as Smart City as a Service
  - Migrate on premise on the fly
- Start on Cloud into a sand box
  - Pass to install on premise what you need

SNAP4CITY on EUROPEAN OPEN SCIENCE CLOUD MARKETPLACE

https://www.Snap4City.org

100% OPEN SOURCE

PEN Test Passed

EU GDPR COMPLIANT

SNAP4CITY

Node-RED

FIWARE

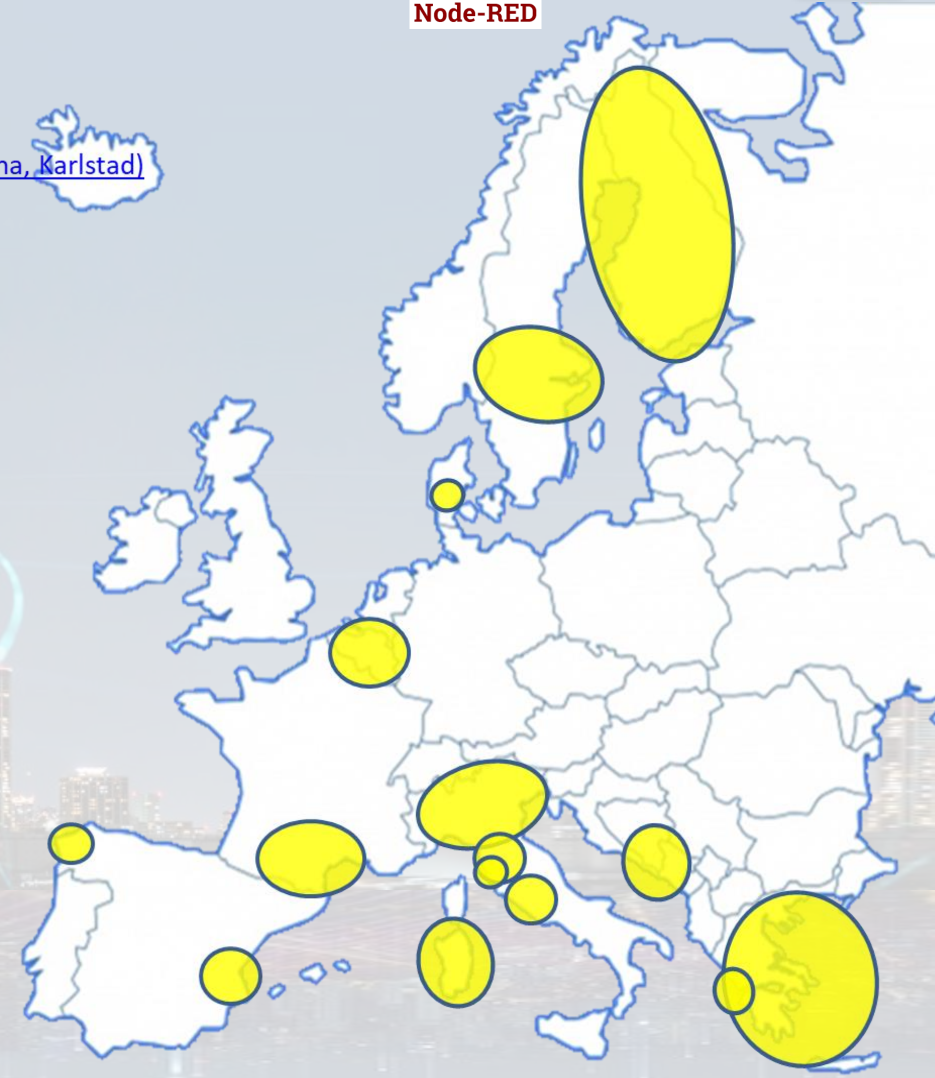
KM4CITY

- > 7 running installations
  - Toscana, Pisa, Sweden, ISPRA, Snap4.eu,
  - Altair, Italmatic, ….
- 13 projects, 12 pilots on 10 Countries
  - >40 cities/area
- **Wide MULTItenant deploy, e.g.,**
  - 18 Organizations / tenant
  - > 7400 users on
  - > 1400 Dashboards
  - > 16 mobile Apps
  - **> 2 Million of structured data per day**
  - > 520 IoT Applications/node-RED
  - > 700 web pages with training
  - > 60 videos, training videos

EUROPEAN OPEN SCIENCE CLOUD

Main Organizations/areas

- Antwerp area (Be)
- Bologna (I)
- Capelon (Sweden: Västerås, Eskilstuna, Karlstad)
- DISIT demo (multiple)
- Dubrovnik, Croatia
- Firenze area (I)
- Garda Lake area (I)
- Greece (Gr)
- Helsinki area (Fin)
- Livorno area (I)
- Lonato del Garda (I)
- Modena (I)
- Mostar, Bosnia-Herzegovina
- Oslo & Padova (Impetus)
- Pisa area (I)
- Pistoia (I)
- Pont du Gard, Occitanie (Fr)
- Prato (I)
- 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)



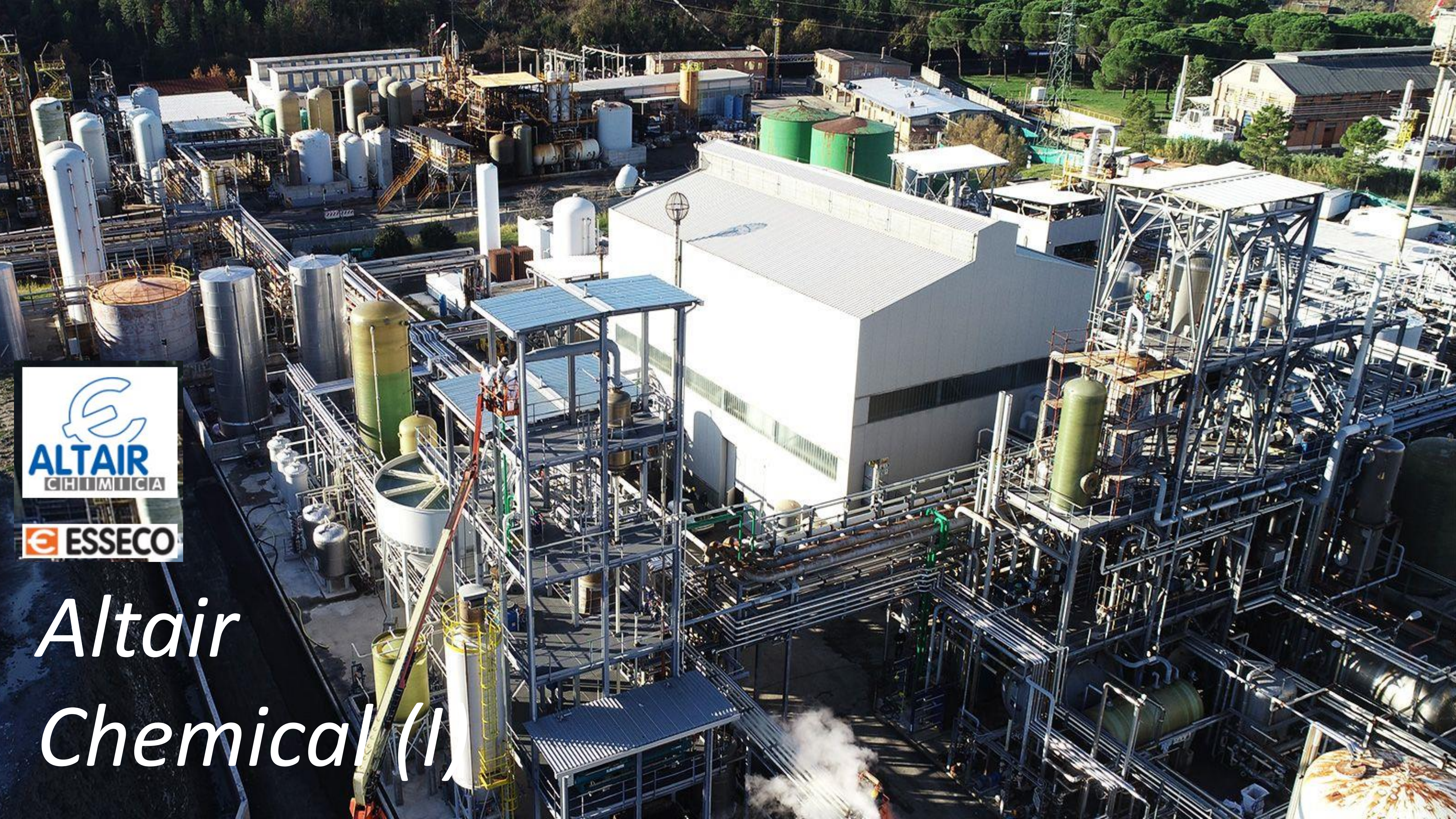
- **Trials in Israel, Brasile, Australia, India, etc.………….**

8th part Agenda

- Architectural Examples
- Smart Solution Development Life Cycle
- Analysis and Design for Innovation
- Design & Develop: Data Models & Processes
- Design & Develop: Data Processes
- Design & Develop: user interfaces, visual tools
- Design & Develop of Data Analytics
- Visual Analytic vs Data Analytics: Client-Side Business Logic Intelligence
- Design & Develop Web and mobile Apps
- Design and Control of Smart Applications
- References

TOP

Architectural Examples



ALTAIR CHIMICA

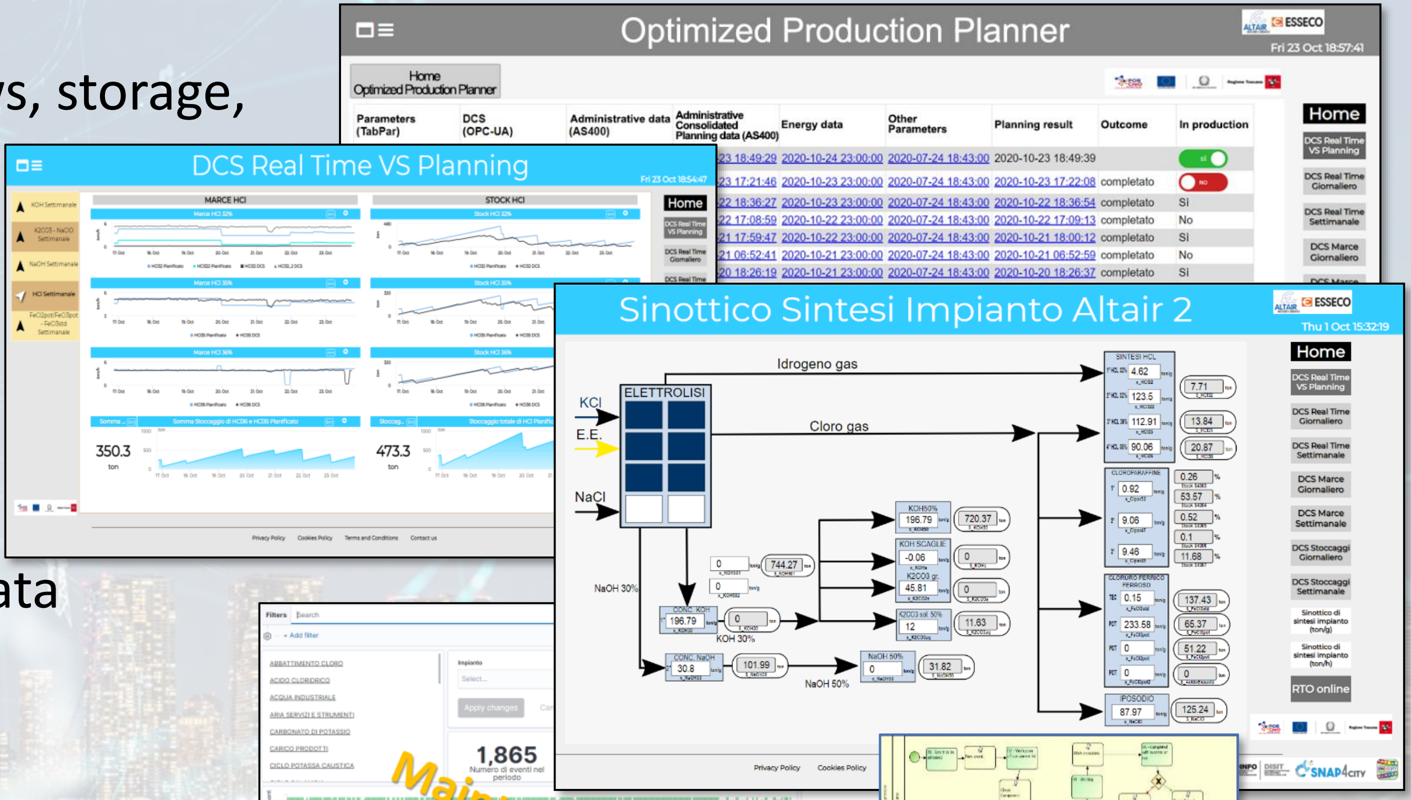
ESSECO

Altair Chemical (I)

Snap4Altair Decision Support supervision and control, Industry 4.0

SNAP4INDUSTRY 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**



Optimized Production Planner

Fri 23 Oct 18:57:41

Home
Optimized Production Planner

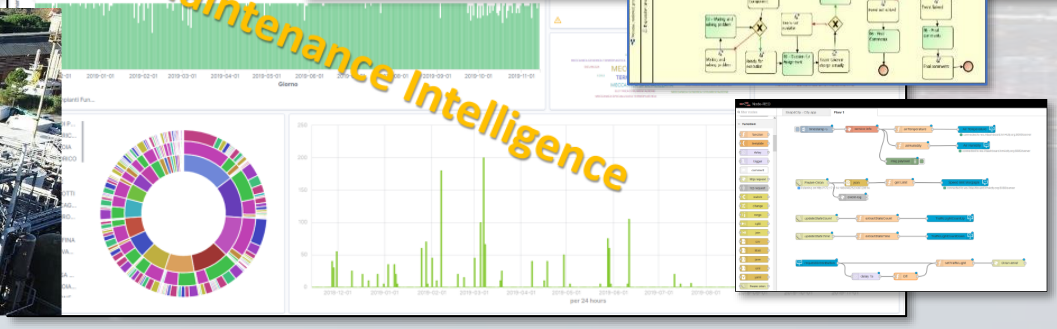
Parameters (TabPar)	DCS (OPC-UA)	Administrative data (AS400)	Administrative Consolidated Planning data (AS400)	Energy data	Other Parameters	Planning result	Outcome	In production
			23 18:49:29	2020-10-24 23:00:00	2020-07-24 18:43:00	2020-10-23 18:49:39		si
			23 17:21:46	2020-10-23 23:00:00	2020-07-24 18:43:00	2020-10-23 17:22:08	completato	NO
			22 18:36:27	2020-10-23 23:00:00	2020-07-24 18:43:00	2020-10-22 18:36:54	completato	Si
			22 17:08:59	2020-10-22 23:00:00	2020-07-24 18:43:00	2020-10-22 17:09:13	completato	No
			21 17:59:47	2020-10-22 23:00:00	2020-07-24 18:43:00	2020-10-21 18:00:12	completato	Si
			21 06:52:41	2020-10-21 23:00:00	2020-07-24 18:43:00	2020-10-21 06:52:59	completato	No
			20 18:26:19	2020-10-21 23:00:00	2020-07-24 18:43:00	2020-10-20 18:26:37	completato	Si

DCS Real Time VS Planning

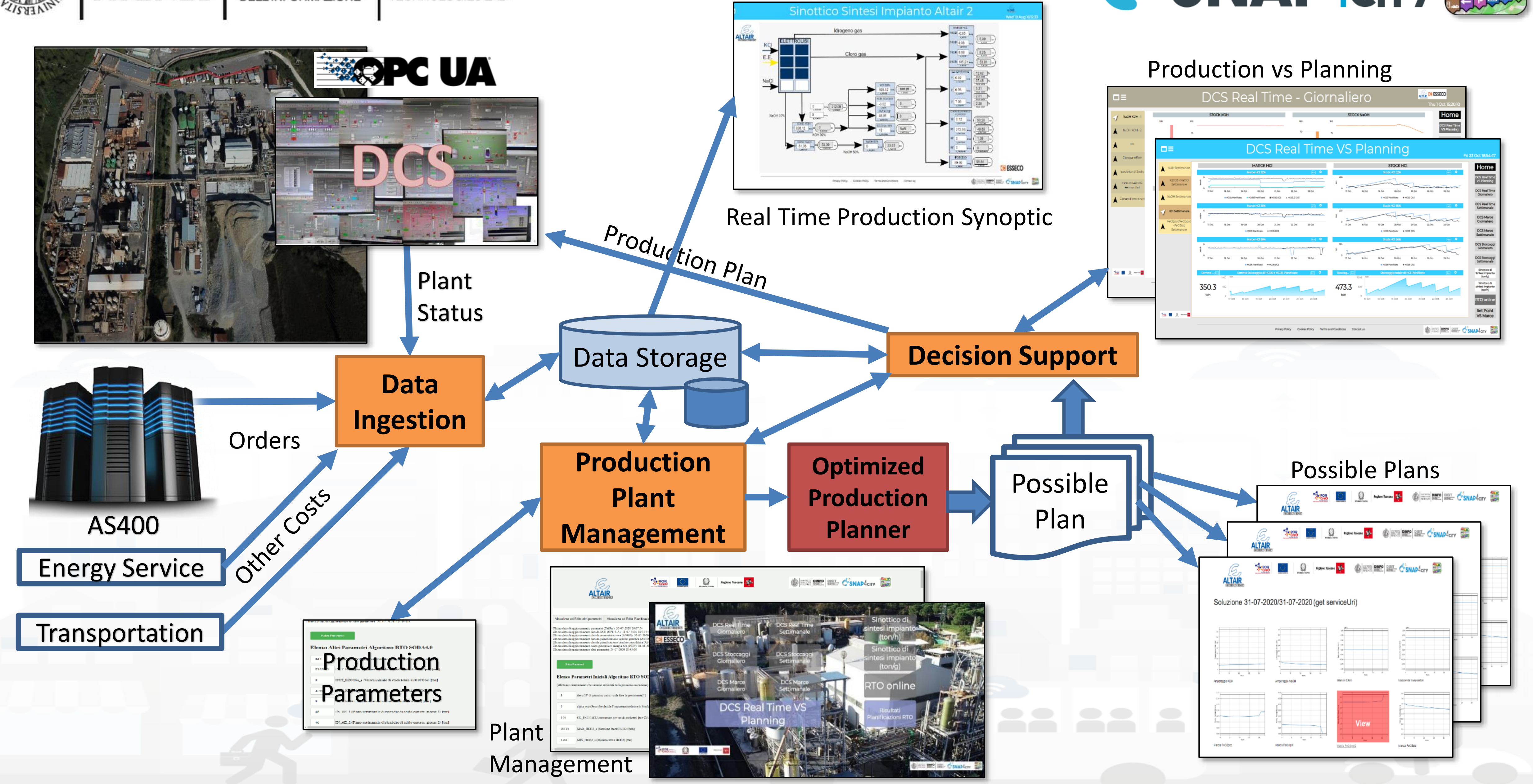
Fri 23 Oct 18:54:47

Sinottico Sintesi Impianto Altair 2

Thu 1 Oct 15:32:19



Maintenance Intelligence



OPC UA

DCS

Plant Status

Real Time Production Synoptic

Production vs Planning

Production Plan

Data Storage

Decision Support

Data Ingestion

AS400

Orders

Energy Service

Transportation

Other Costs

Production Plant Management

Optimized Production Planner

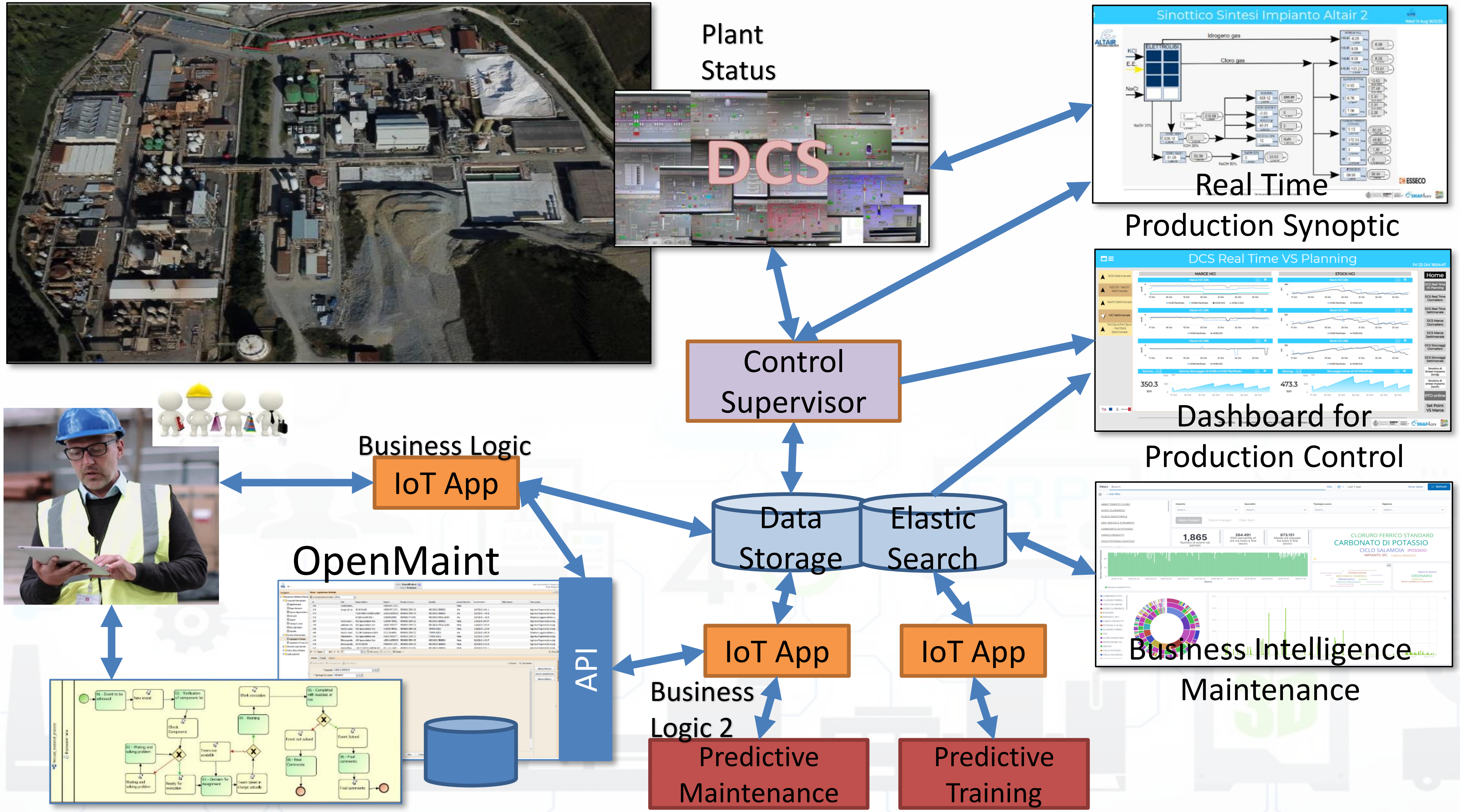
Possible Plan

Possible Plans

Production Parameters

Plant Management

Solution



Plant Status

DCS

Real Time Production Synoptic

Dashboard for Production Control

Business Intelligence Maintenance

Control Supervisor

Business Logic

IoT App

Data Storage

Elastic Search

OpenMaint

API

IoT App

IoT App

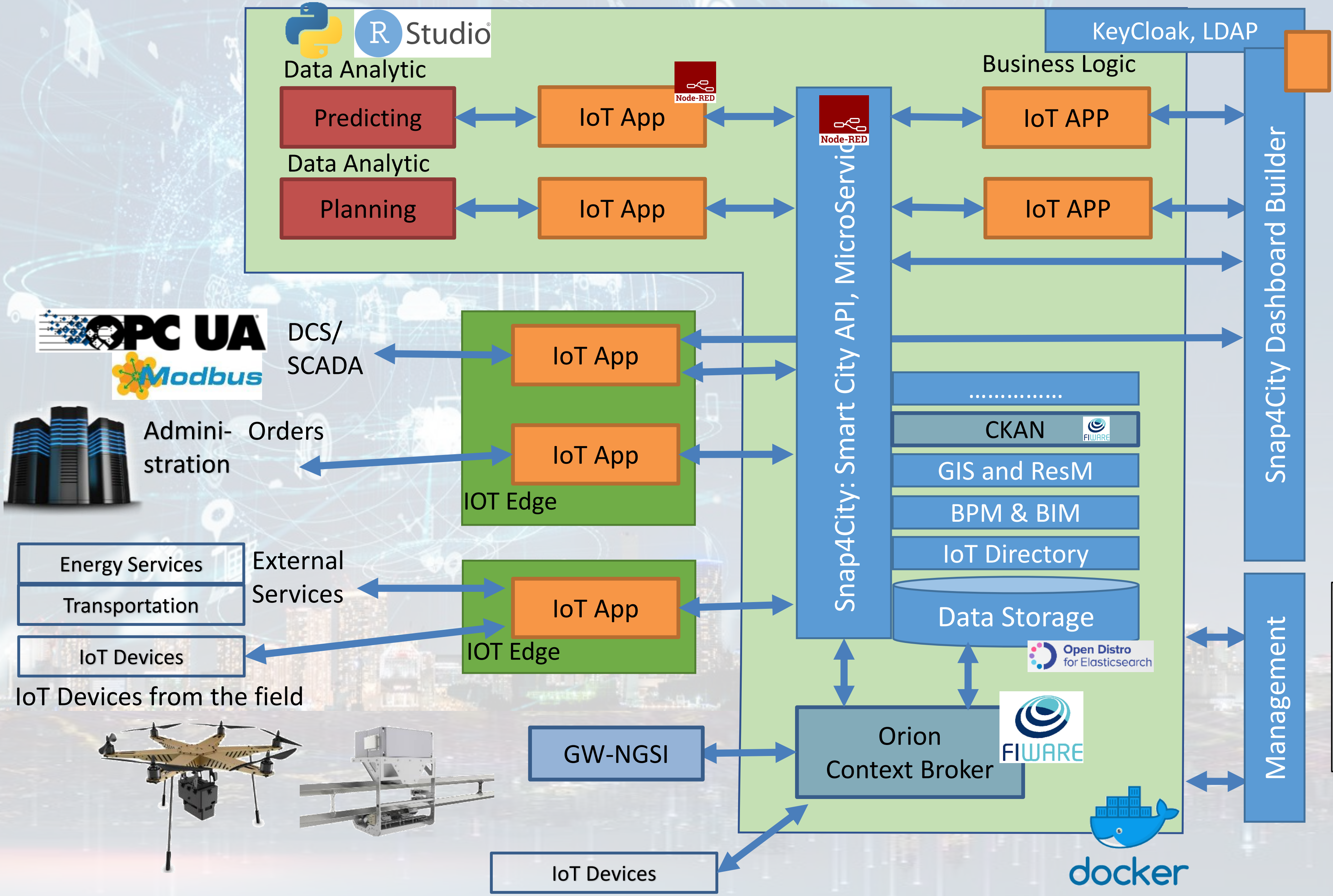
Business Logic 2

Predictive Maintenance

Predictive Training

Sinottico Sintesi Impianto Altair 2
DCS Real Time VS Planning

Snap4City/Industry Detailed Architecture



Data Analytic

Predicting

IoT App

Data Analytic

Planning

IoT App

Node-RED

Snap4City: Smart City API, MicroServices

KeyCloak, LDAP

Business Logic

IoT APP

IoT APP

Snap4City Dashboard Builder

DCS/ SCADA

Admini- stration

Orders

IoT App

IoT App

IOT Edge

Energy Services

Transportation

External Services

IoT Devices

IoT App

IOT Edge

IoT Devices from the field

....

CKAN

GIS and ResM

BPM & BIM

IoT Directory

Data Storage

Open Distro for Elasticsearch

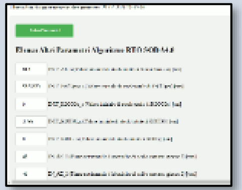
GW-NGSI

Orion Context Broker

Management

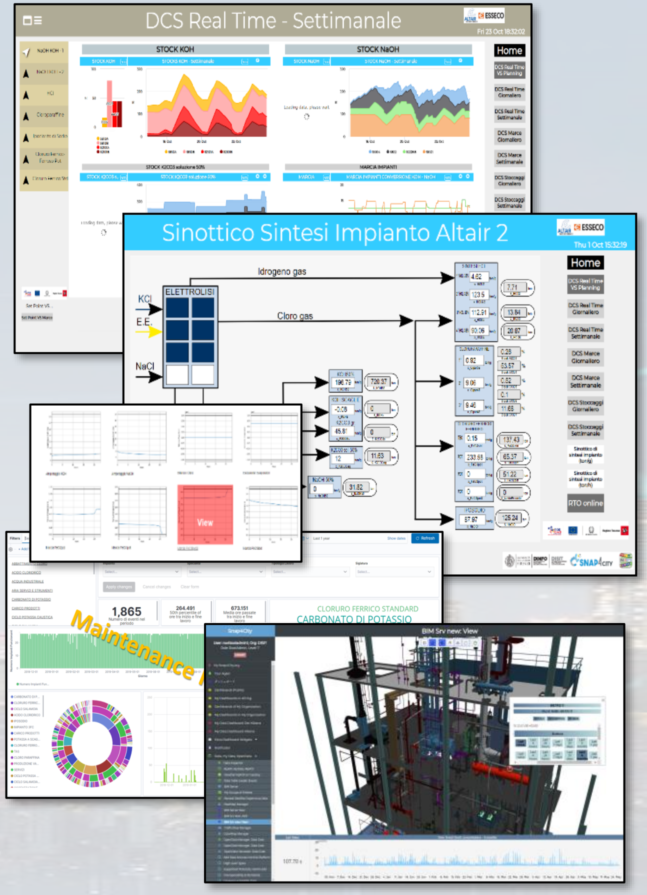
IoT Devices

docker

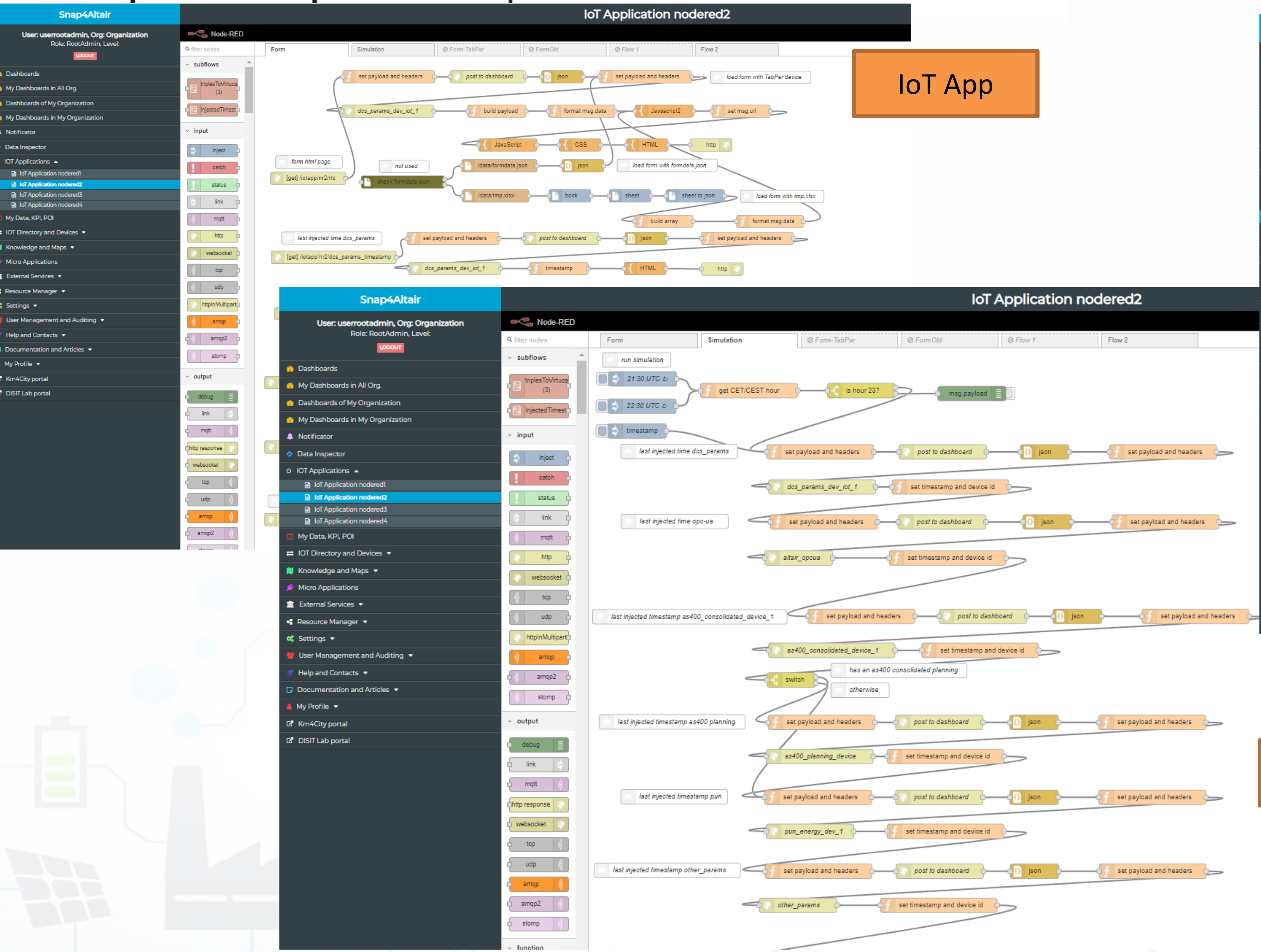


Production Parameters

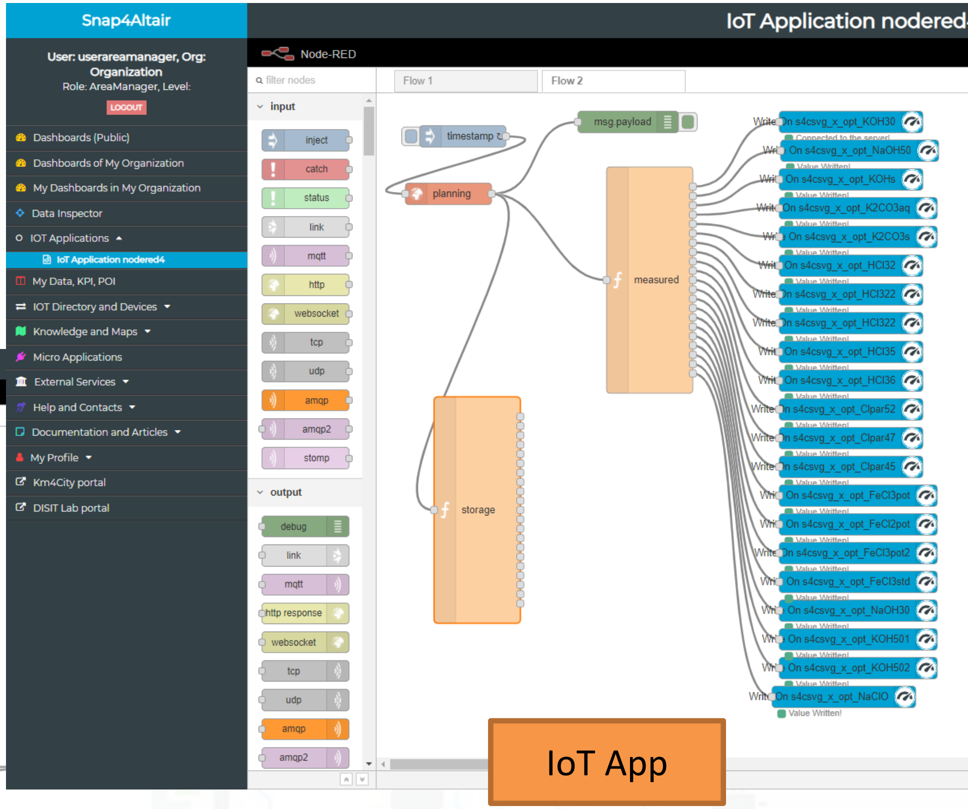
Dashboards, Visual Analytics, Synoptics, 3D, Maps



Some Flows

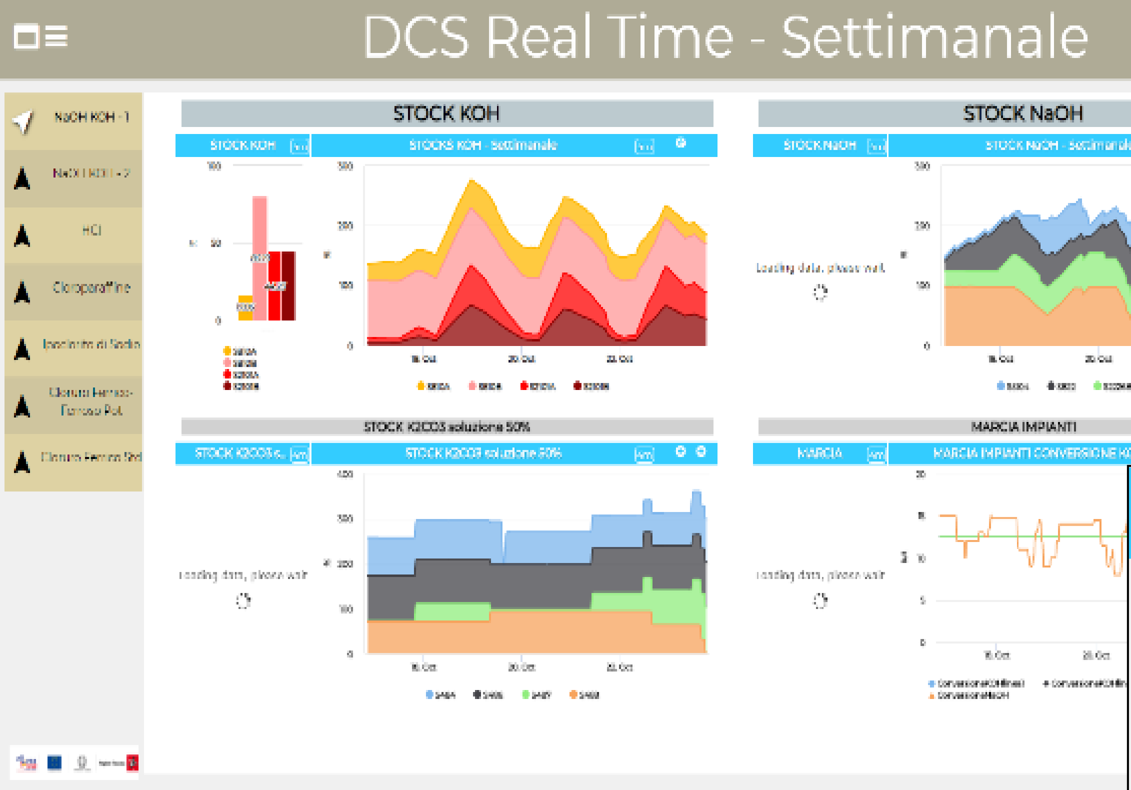


IoT App



IoT App

IoT App



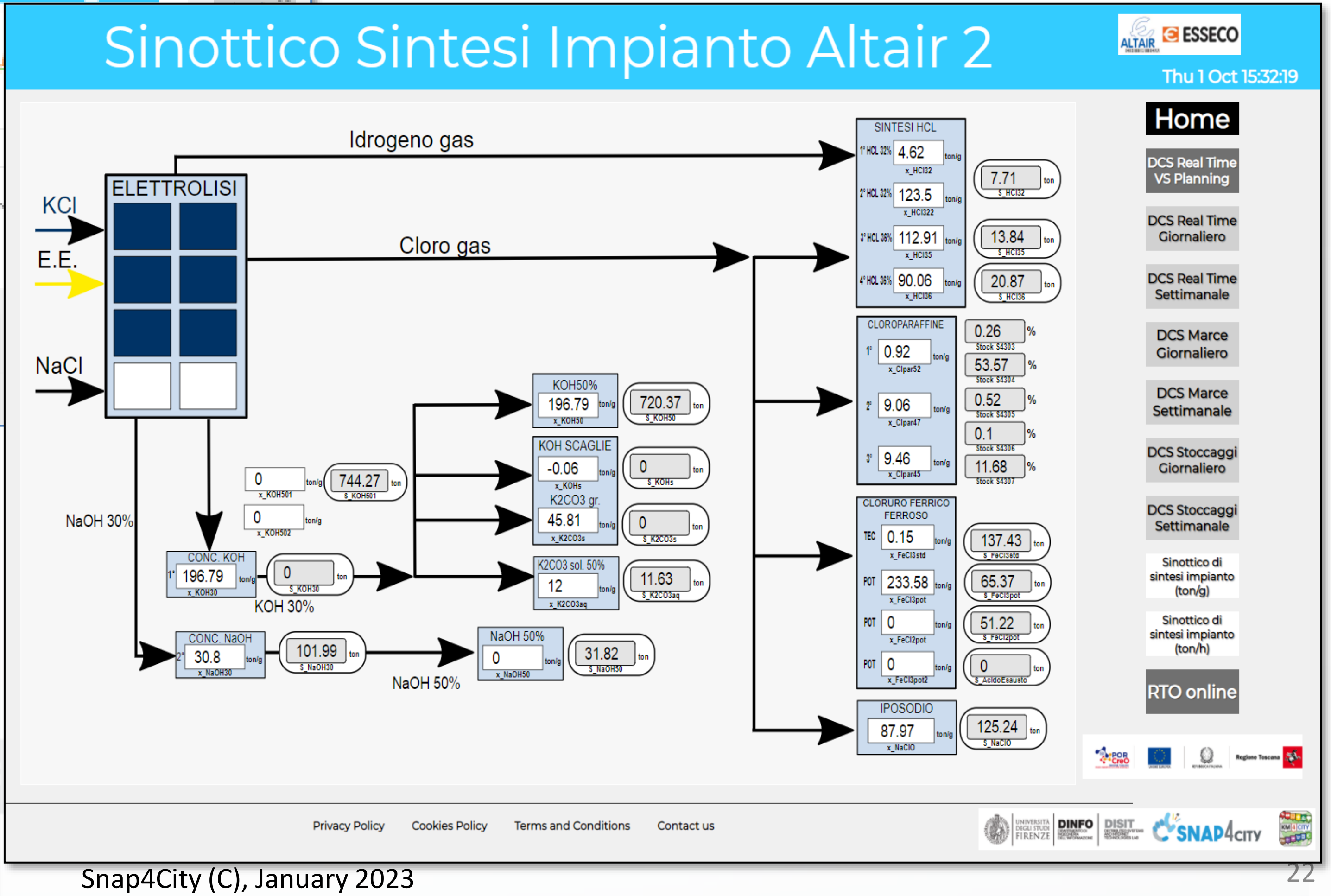
ALTAIR CHIMICA
ESSECO

SNAP4INDUSTRY

KM4CITY

Planning

SNAP4ALTAIR



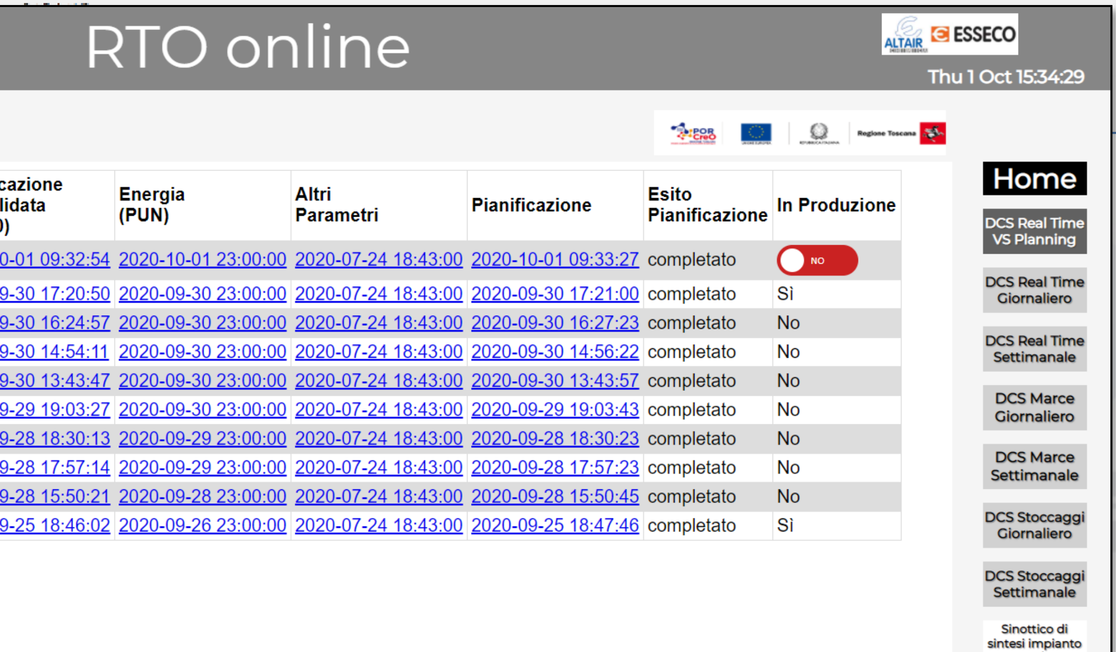
Sinottico Sintesi Impianto Altair 2

ESSECO

Thu 1 Oct 15:32:19

- 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



RTO online

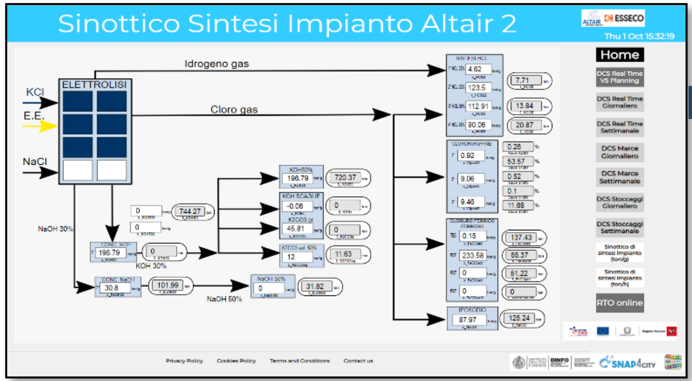
ESSECO

Thu 1 Oct 15:34:29

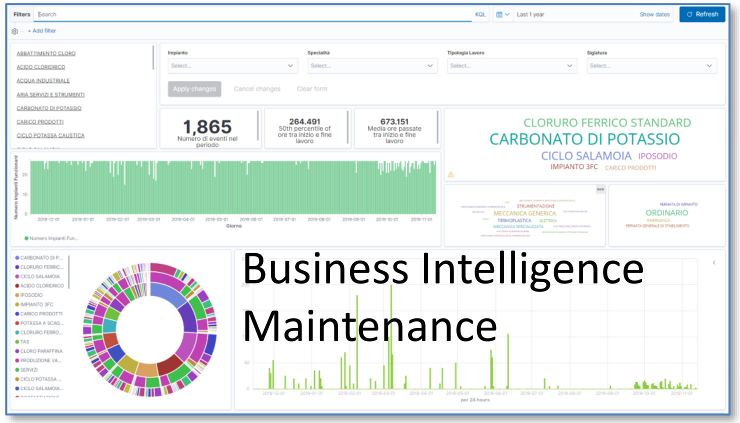
cazione lidata)	Energia (PUN)	Altri Parametri	Pianificazione	Esito Pianificazione	In Produzione
0-01 09:32:54	2020-10-01 23:00:00	2020-07-24 18:43:00	2020-10-01 09:33:27	completato	NO
9-30 17:20:50	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 17:21:00	completato	Sì
9-30 16:24:57	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 16:27:23	completato	No
9-30 14:54:11	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 14:56:22	completato	No
9-30 13:43:47	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 13:43:57	completato	No
9-29 19:03:27	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-29 19:03:43	completato	No
9-28 18:30:13	2020-09-29 23:00:00	2020-07-24 18:43:00	2020-09-28 18:30:23	completato	No
9-28 17:57:14	2020-09-29 23:00:00	2020-07-24 18:43:00	2020-09-28 17:57:23	completato	No
9-28 15:50:21	2020-09-28 23:00:00	2020-07-24 18:43:00	2020-09-28 15:50:45	completato	No
9-25 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 di sintesi impianto

Workflow for Ticket management


Sinottico Sintesi Impianto Altair 2

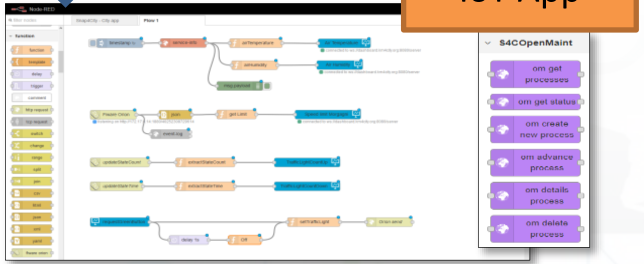
Consumptions/productions

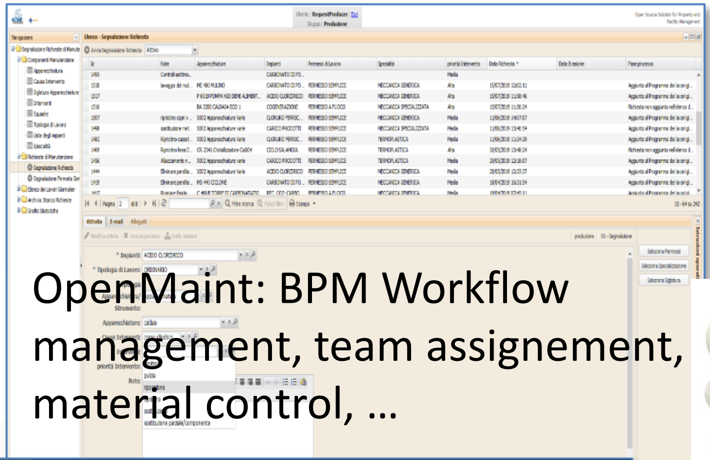


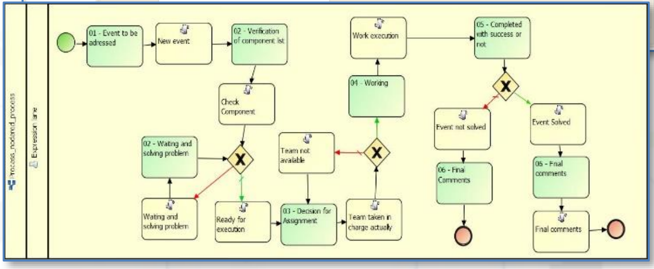
Business Intelligence Maintenance

Events/actions

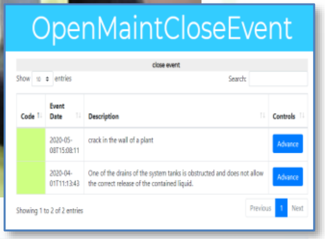
Dashboards and actions


IoT App


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


01 - Event to be adressed
New event
02 - Verification of component list
Work execution
05 - Completed with success or not
Check Component
04 - Working
Event not solved
Event Solved
02 - Waiting and solving problem
Team not available
Waiting and solving problem
Ready for execution
03 - Decision for Assignment
Team taken in charge actually
06 - Final Comments
06 - Final comments
Final comments

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


OpenMaintCloseEvent

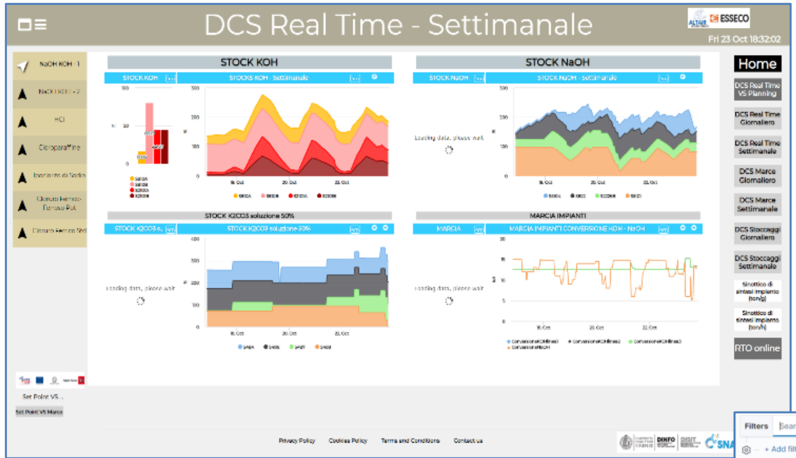
close event

Show 10 entries Search:

Code	Event Date	Description	Controls
🟩	2020-05-08T15:08:11	crack in the wall of a plant	Advance
🟩	2020-04-01T11:13:43	One of the drains of the system tanks is obstructed and does not allow the correct release of the contained liquid.	Advance

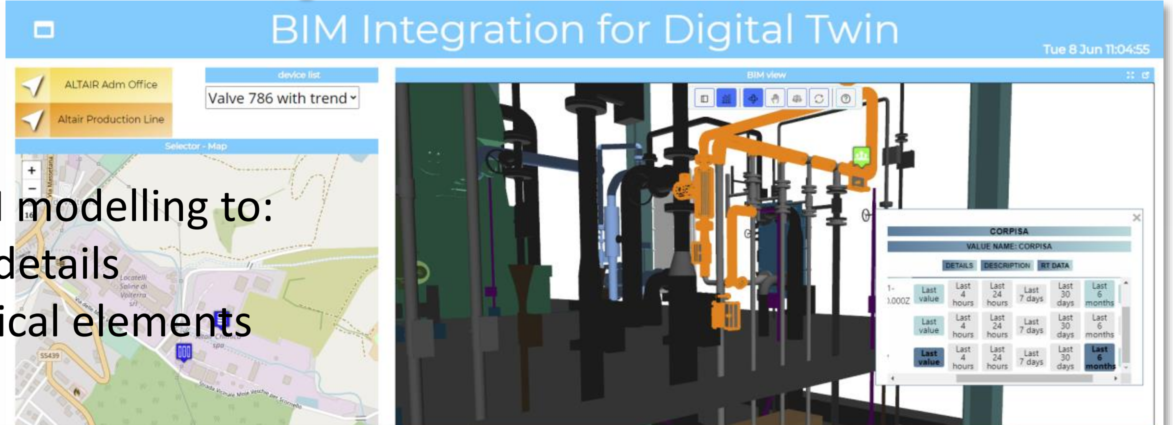
Showing 1 to 2 of 2 entries Previous 1 Next

Closing the loop

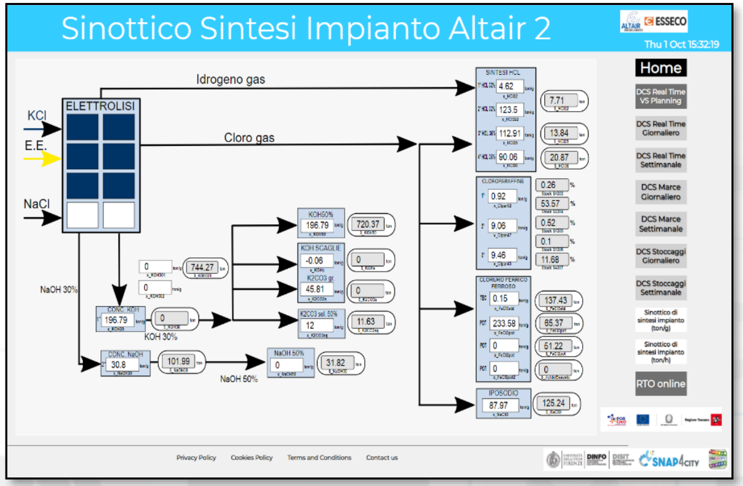


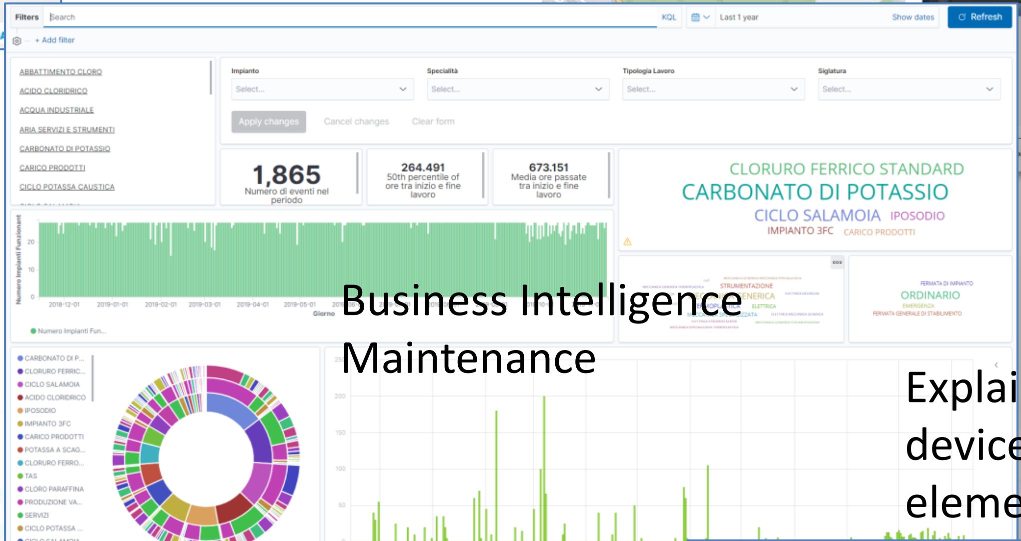
Historical and Real Time Data

Map and 3D BIM modelling to:
-- represent the details
-- associate physical elements with data



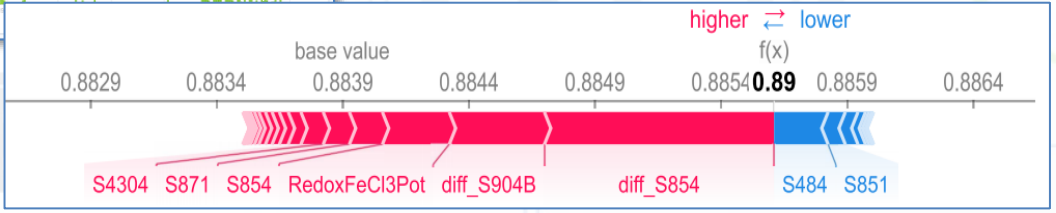
Synoptics for real time monitoring





Business Intelligence
Maintenance

Explainable AI to map critical values of devices and detection to physical elements in the plant



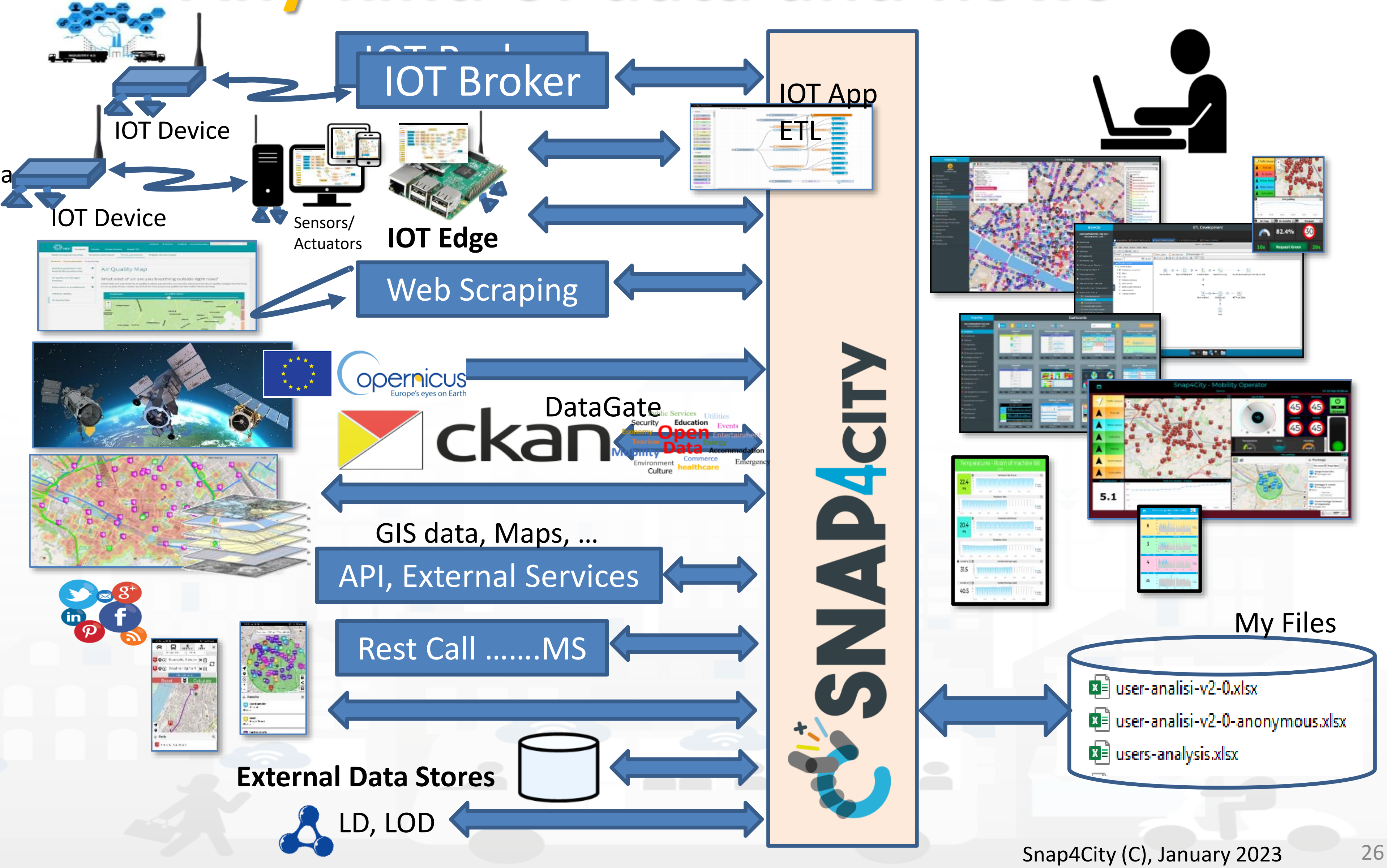
https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MzA1NA==

TOP

Snap4City: Protocols and Data Models Interoperability

Any kind of data and flows

- **Open Data:**
  - Data gate, federation of Open Data Portals
  - IOT App, ETL proc(PULL)
- **IOT Networks:**
  - IOT Application processes, data driven or PULL
  - IOT Brokers (Push) → IOT Shadow
- **Web Pages:**
  - Web scraping, crawling processes
- **Satellite data**
- **Social media: Twitter, Facebook,..**
  - Twitter Vigilance, IOT App
- **Mobile Apps**
  - Smart City API
- **Files upload: CSV, Excel, etc.**
  - IOT Applications, ETL
- **REST API, WS, FTP, LD, LOD, etc.**
  - IOT Applications, ETL
- **Data base accesses**
  - GIS: WFS, WMS
  - ETL, IOT Application



IOT Broker

IOT Device

IOT Device

Sensors/ Actuators

IOT Edge

IOT App

ETL

Web Scraping

opernicus Europe's eyes on Earth

DataGate

ckan

Open Data

GIS data, Maps, …

API, External Services

Rest Call …….MS

External Data Stores

LD, LOD

SNAP4CITY

My Files

user-analisi-v2-0.xlsx

user-analisi-v2-0-anonymous.xlsx

users-analysis.xlsx

Standards and Interoperability (9/2022)

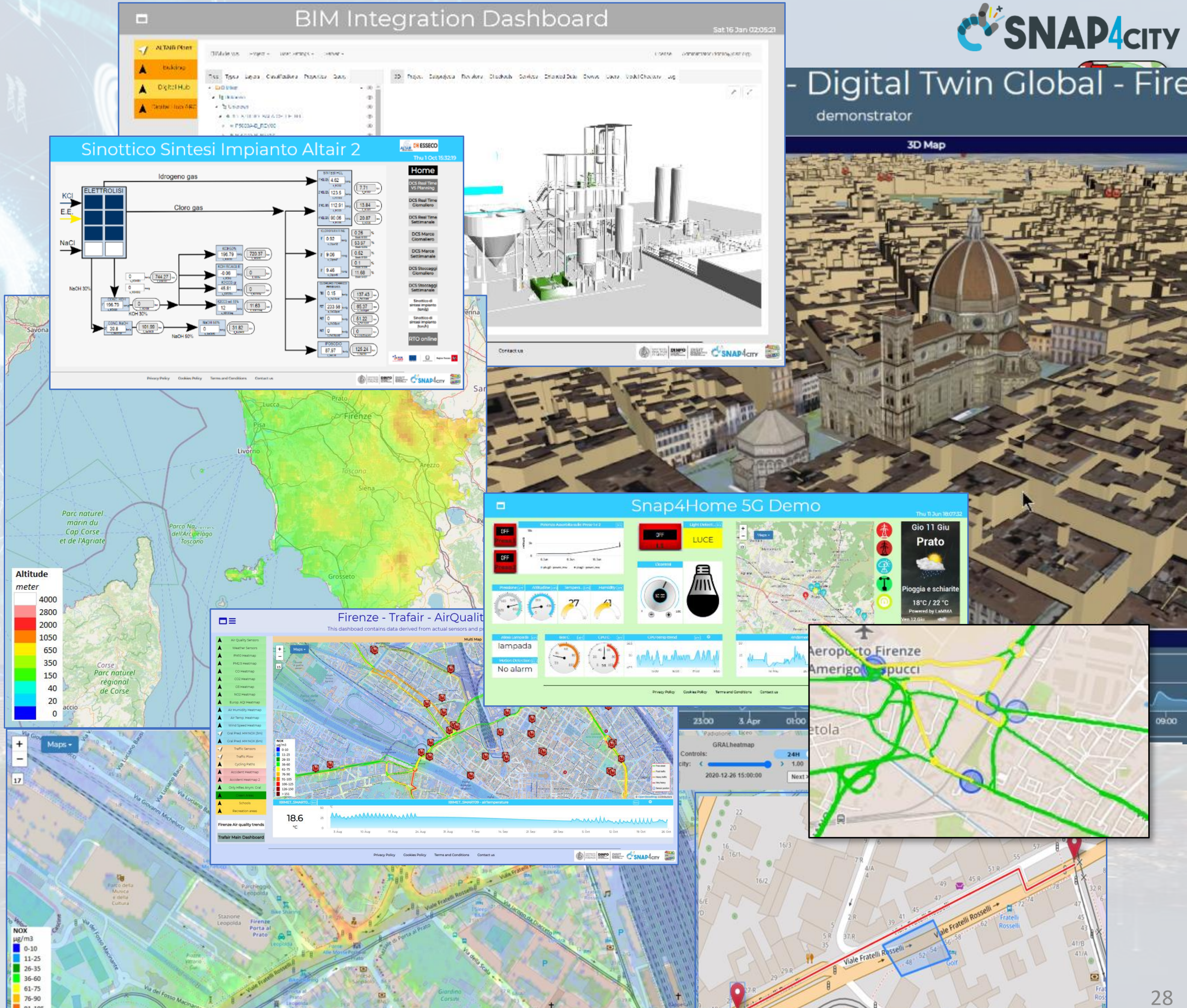
Compliant with:

- **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, SMTP, 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, ….
- **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, ..
- **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

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.

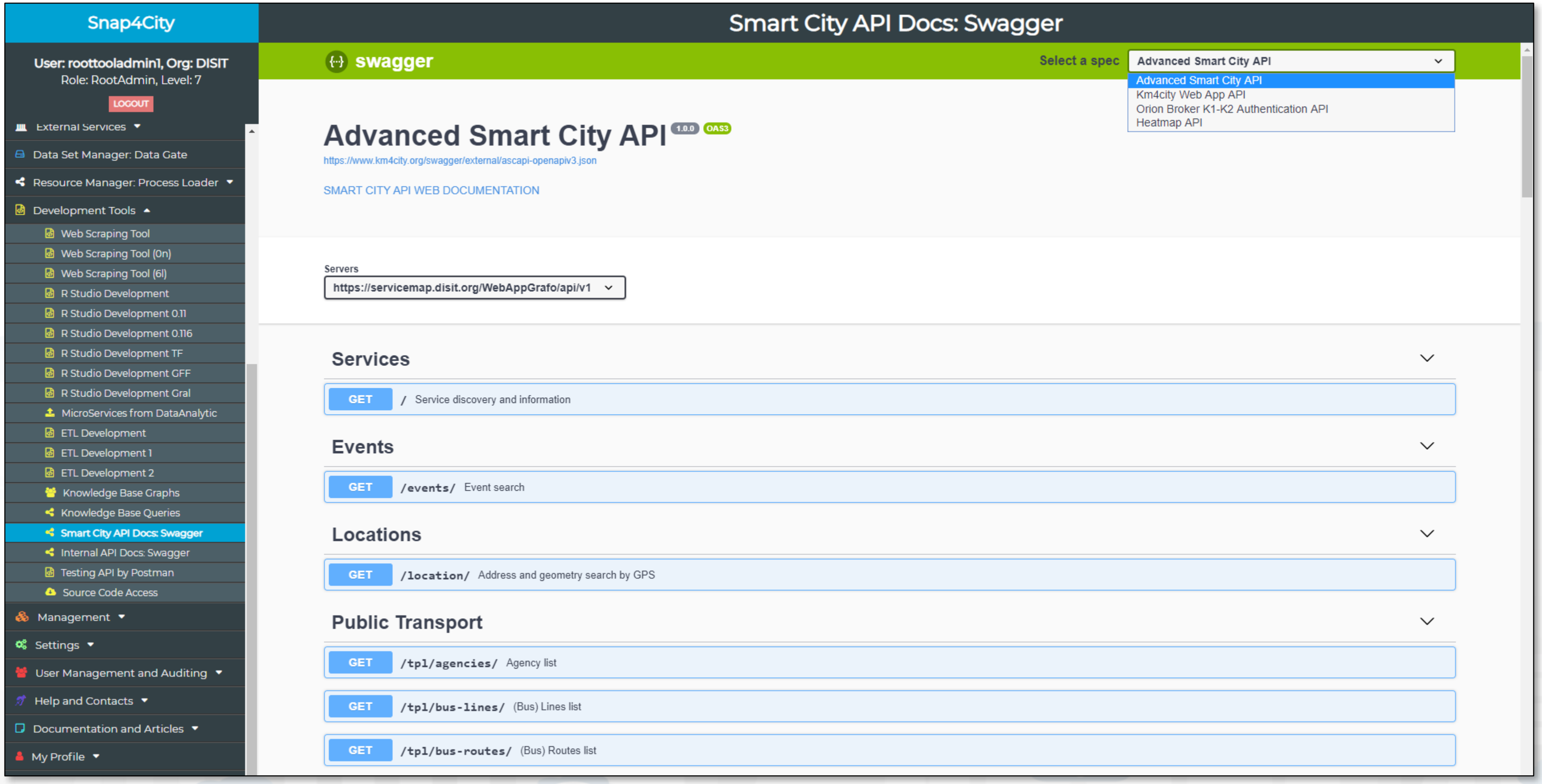


UNIVERSITÀ DEGLI STUDI FIRENZE

DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

External Smart City API



Smart City API Docs: Swagger

Advanced Smart City API 1.0.0 OAS3

https://www.km4city.org/swagger/external/ascapi-openapiv3.json

SMART CITY API WEB DOCUMENTATION

Servers
https://servicemap.disit.org/WebAppGrafo/api/v1

Services
GET / Service discovery and information

Events
GET /events/ Event search

Locations
GET /location/ Address and geometry search by GPS

Public Transport
GET /tpl/agencies/ Agency list
GET /tpl/bus-lines/ (Bus) Lines list
GET /tpl/bus-routes/ (Bus) Routes list

https://www.km4city.org/swagger/external/index.html

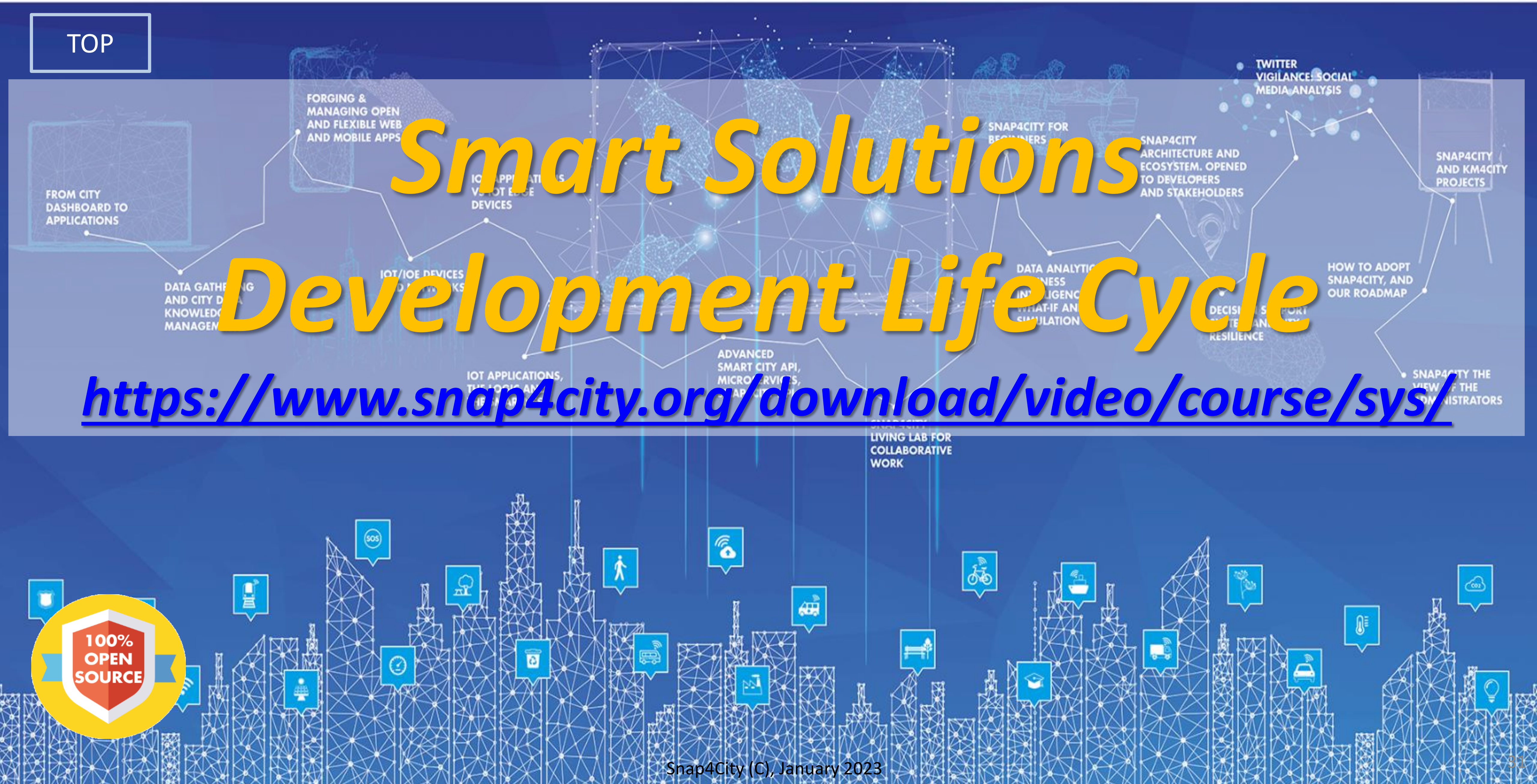
Authentication and SSO

- **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/

TOP

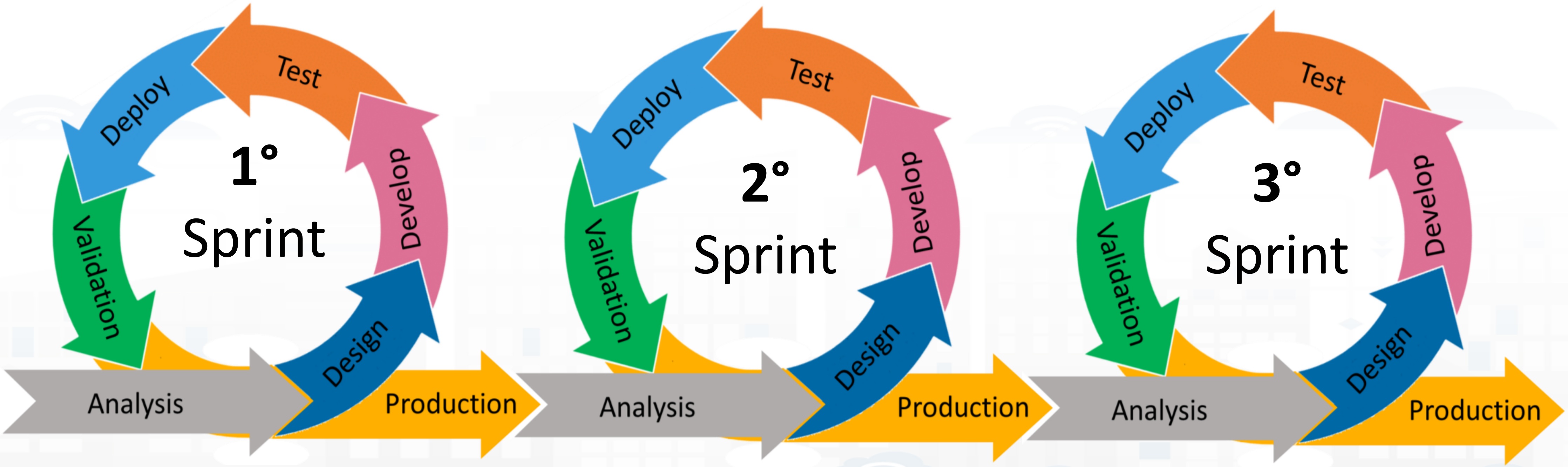
Smart Solutions Development Life Cycle

https://www.snap4city.org/download/video/course/sys/

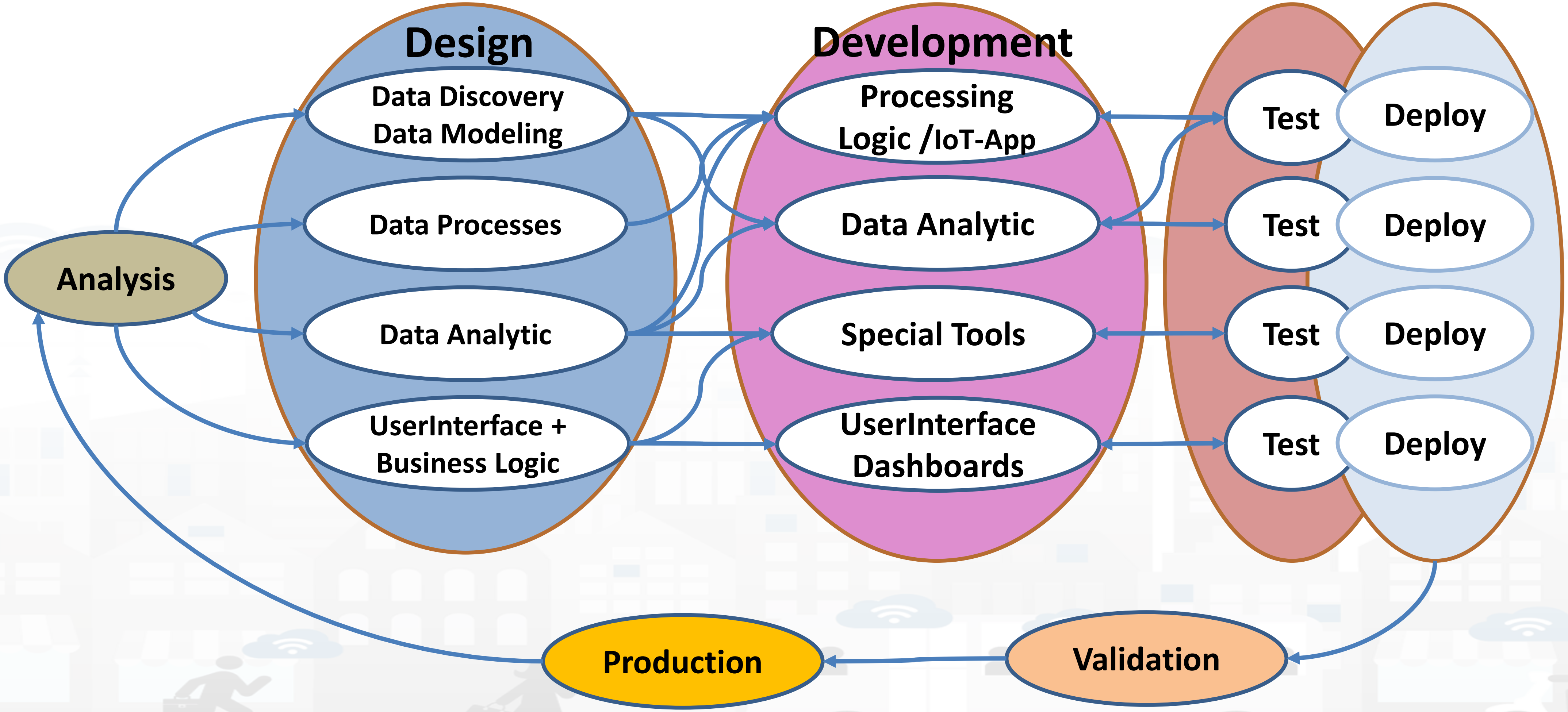

FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS
FROM CITY DASHBOARD TO APPLICATIONS
SNAP4CITY FOR BEGINNERS
SNAP4CITY ARCHITECTURE AND ECOSYSTEM, OPENED TO DEVELOPERS AND STAKEHOLDERS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY AND KM4CITY PROJECTS
DEVICES
IOT/IOE DEVICES
DATA ANALYTIC
SIMULATION
RESILIENCE
HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP
ADVANCED SMART CITY API,
IOT APPLICATIONS,
LIVING LAB FOR COLLABORATIVE WORK
100% OPEN SOURCE

Development Life Cycle
Smart Solutions

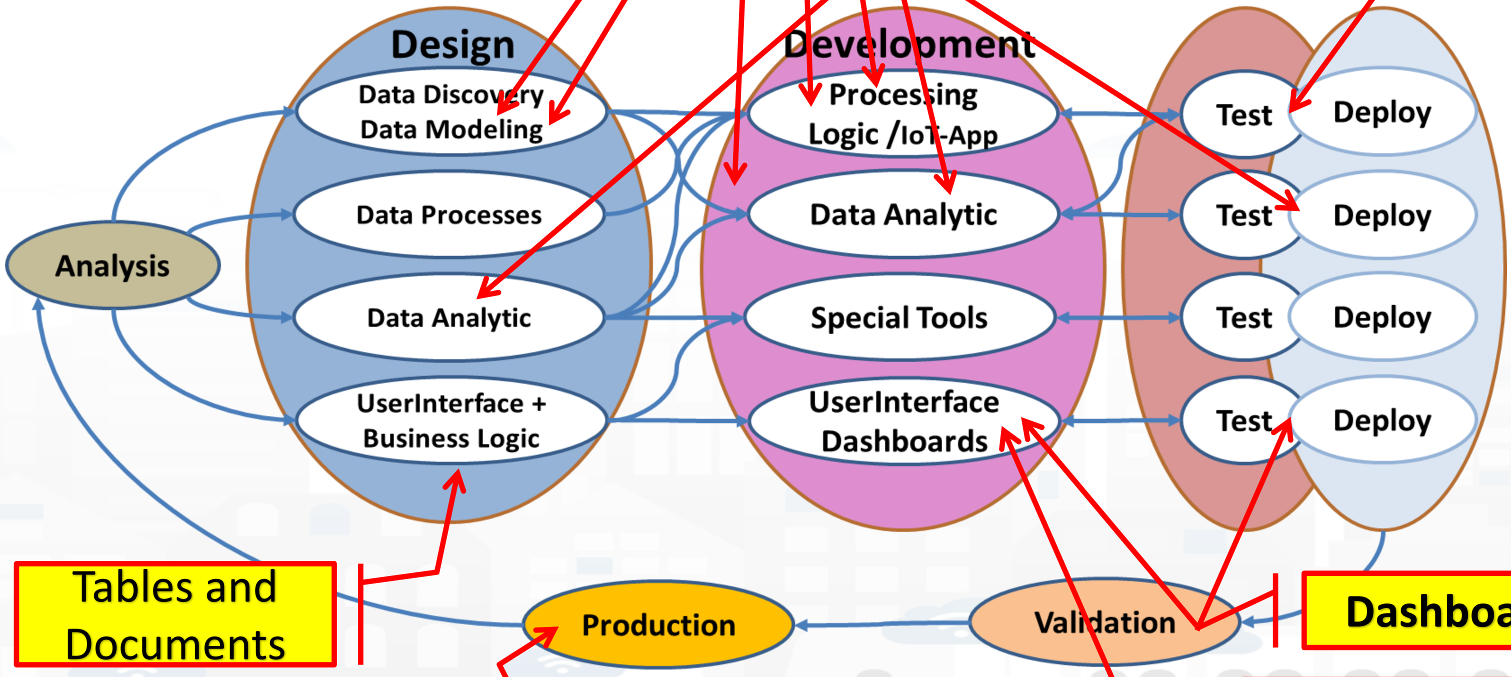
RULE OF THUMB



Development Life Cycle Smart Solutions



Snap4City Tools vs Development Life Cycle



Data Management HLT-Data Inspector

Knowledge and Map, ServiceMap

Proc.Logic/IoT App

Development Tools + IoT App

Entity Directory IoT Directory

HLT & Data Management

Tables and Documents

Development life cycle manual

Management

Dashboards

Extra Dashboard Widgets

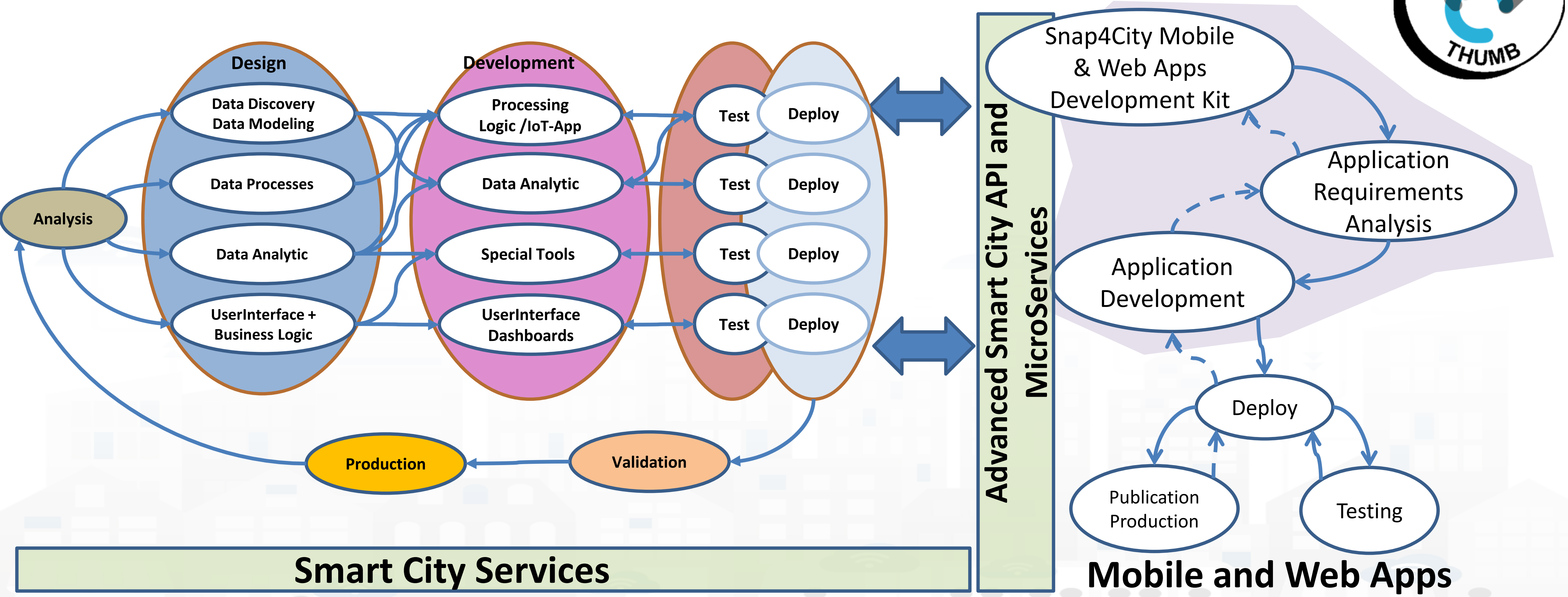
Test & Deploy

- The activities of Test and Deploy are performed into the corresponding tools
  - **Processing Logic / IoT App Editor Node-RED** provides a button for Deploy and a Debug console for testing
  - **Data Analytics** are
    - tested on development user interface on RStudio and Python
    - Tested on Deploy when they are executed as container from IoT Apps
  - **Dashboards** are tested directly into the Dashboard editor and preview

Validation and Production

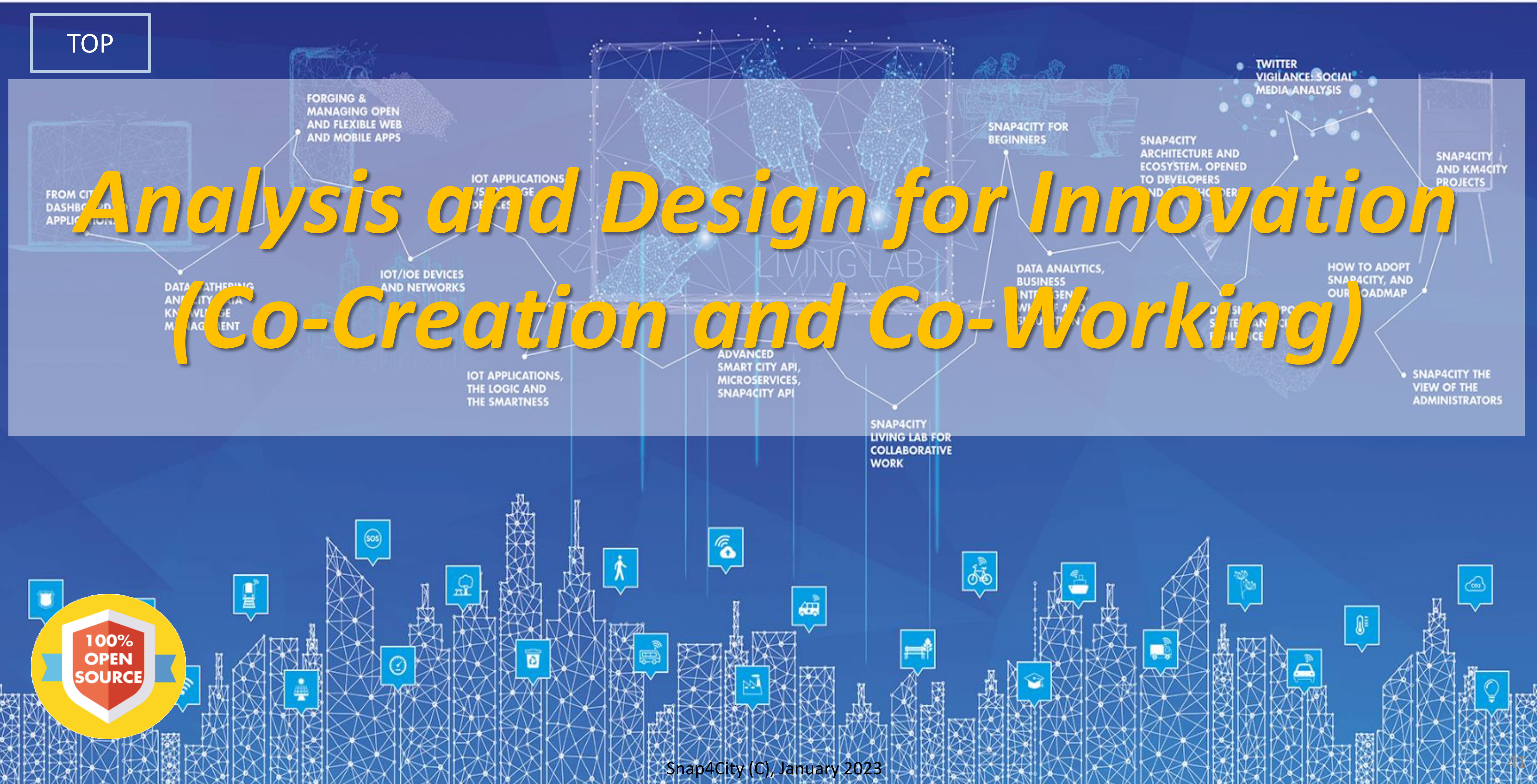
- Is the phase in which all components can be integrated and tested in their integration on the platform ready to be used in production.
- The **validation** should be performed verifying:
  - Functional Requirements
  - Non functional Requirements
- The **production** process is very easy in Snap4City since implies to provide access to the tools and services to final users you planned.
  - The grant can be performed on Dashboard Management and on IoT Directory, and on Data Management for the data.
- Once put in production the **Solution can be monitored** in deep on Dashboard usage, on data status, on IoT App, etc. See Part 6 of the training course.

Develop Mobile & Web Applications Exploiting Snap4City Smart City Services

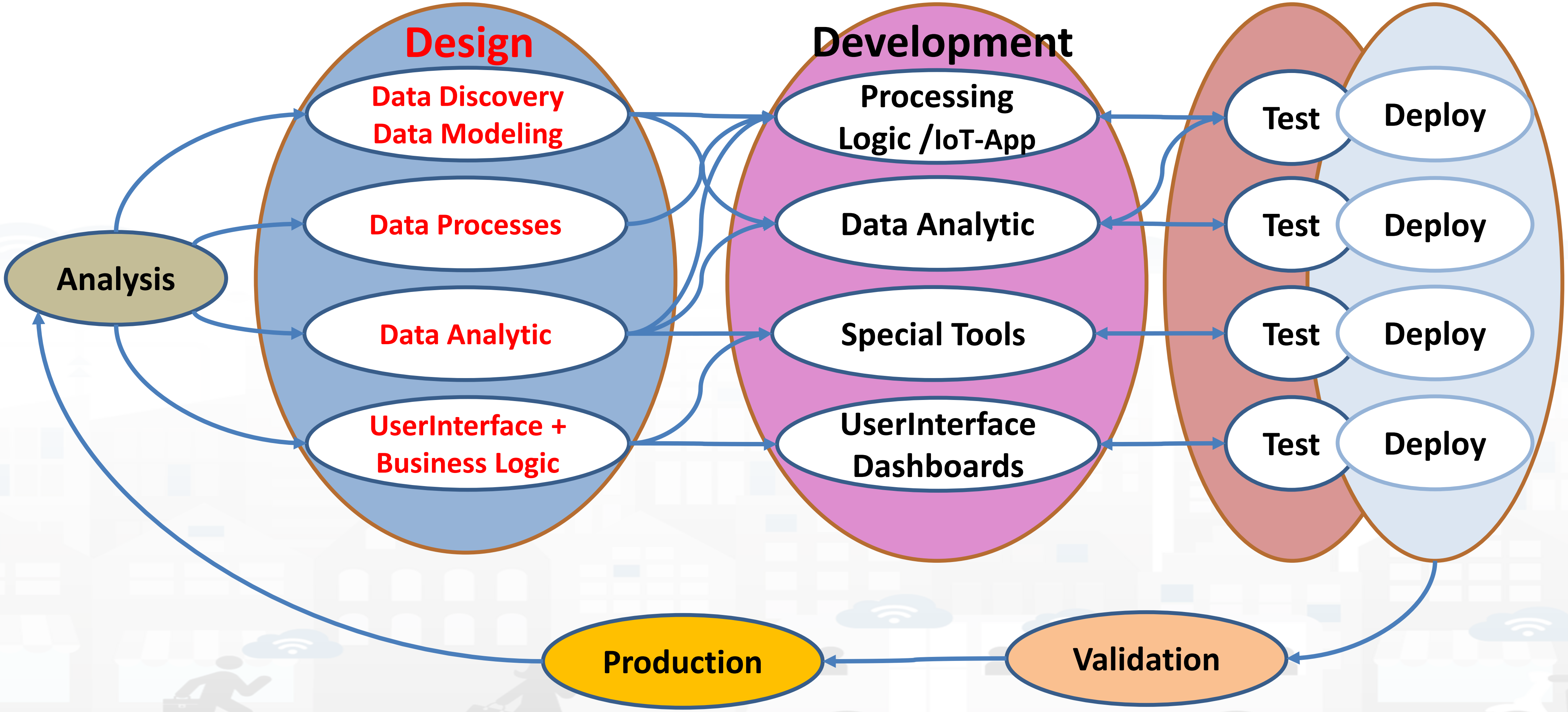


TOP

Analysis and Design for Innovation (Co-Creation and Co-Working)


FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS
FROM CITY DASHBOARD APPLICATIONS
IOT APPLICATIONS / SMART EDGE DEVICES
SNAP4CITY FOR BEGINNERS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY ARCHITECTURE AND ECOSYSTEM. OPENED TO DEVELOPERS AND STAKEHOLDER
SNAP4CITY AND KM4CITY PROJECTS
DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT
IOT/IOE DEVICES AND NETWORKS
LIVING LAB
DATA ANALYTICS, BUSINESS INTELLIGENCE
HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP
DECISION SUPPORT SYSTEM AND RESILIENCE
ADVANCED SMART CITY API, MICROSERVICES, SNAP4CITY API
IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS
SNAP4CITY THE VIEW OF THE ADMINISTRATORS
SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK
100% OPEN SOURCE

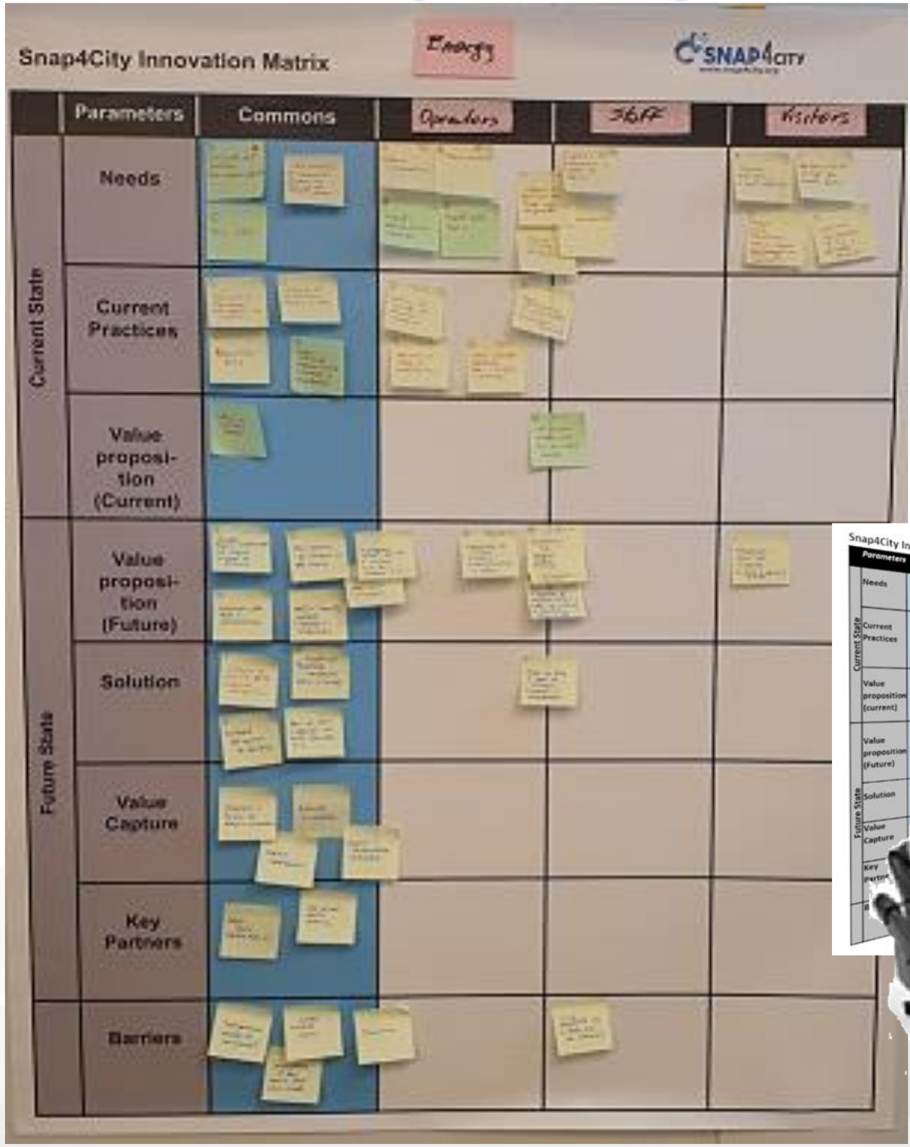
Development Life Cycle Smart Solutions



Analysis 1/2

- **Performing workshops:** Innovation Matrix by domain
- **Entity Identification**: which is the **Dictionary**
  - **Actors and their profiles (as Entity Models, IoT Device Model):** User, Operator, final user, ict expert, decision maker, doctors, driver, etc.
  - **entities and their digital counterpart (as Entity Models, IoT Device Model)** for: Vehicle, Analysis, Server, Client, Mobile App, parking area, etc.
  - **Entity Instances / IoT Devices which are instances of the models** as: City user XX, Control Room Operator, Doctor Rossi, Cop 3726, Car FI796HG, IoT Device XY, Trip 34, Patient Health Record for Robert, etc.
  - **External API:** to interoperate with any other application and service.
  - **External Services:** to host into the user interface and Dashboards elements coming from third party applications.
  - **Tools:** which can be actual software or hardware tools, and also data analytics, algorithms, procedures.
  - **Modules or Tools** of Third party or legacy tools: they are applications, servers, IoT Edge subsystems, well known services for data providing, gateway, brokers, etc., which should interact some how with your solutions. They can be on cloud or on some premise, they can provide you some External API, of some kind: WebServer, Rest Call, FTP, Web Socket, MQTT, etc.

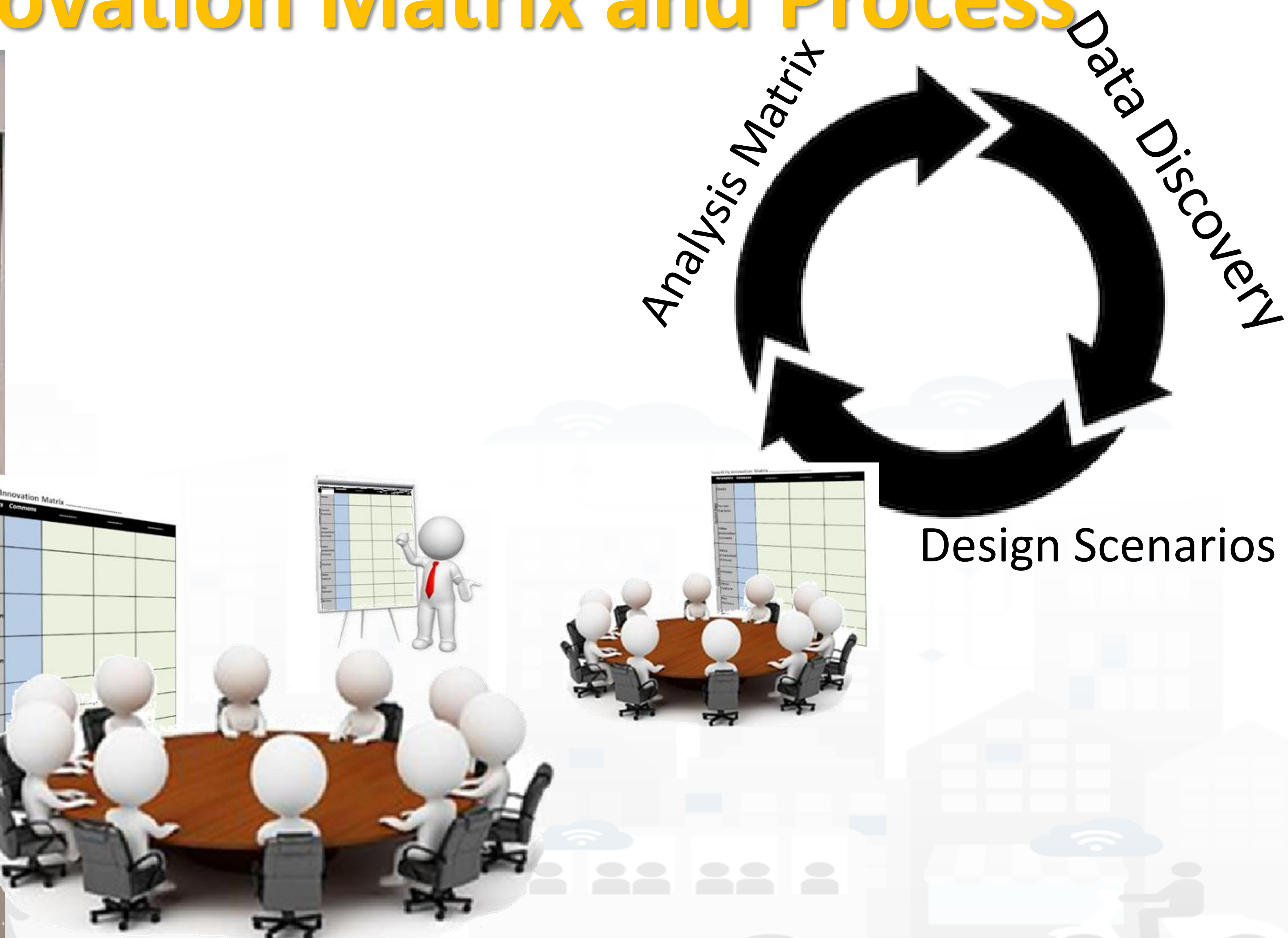
Snap4City Innovation Matrix and Process



Snap4City Innovation Matrix

Energy

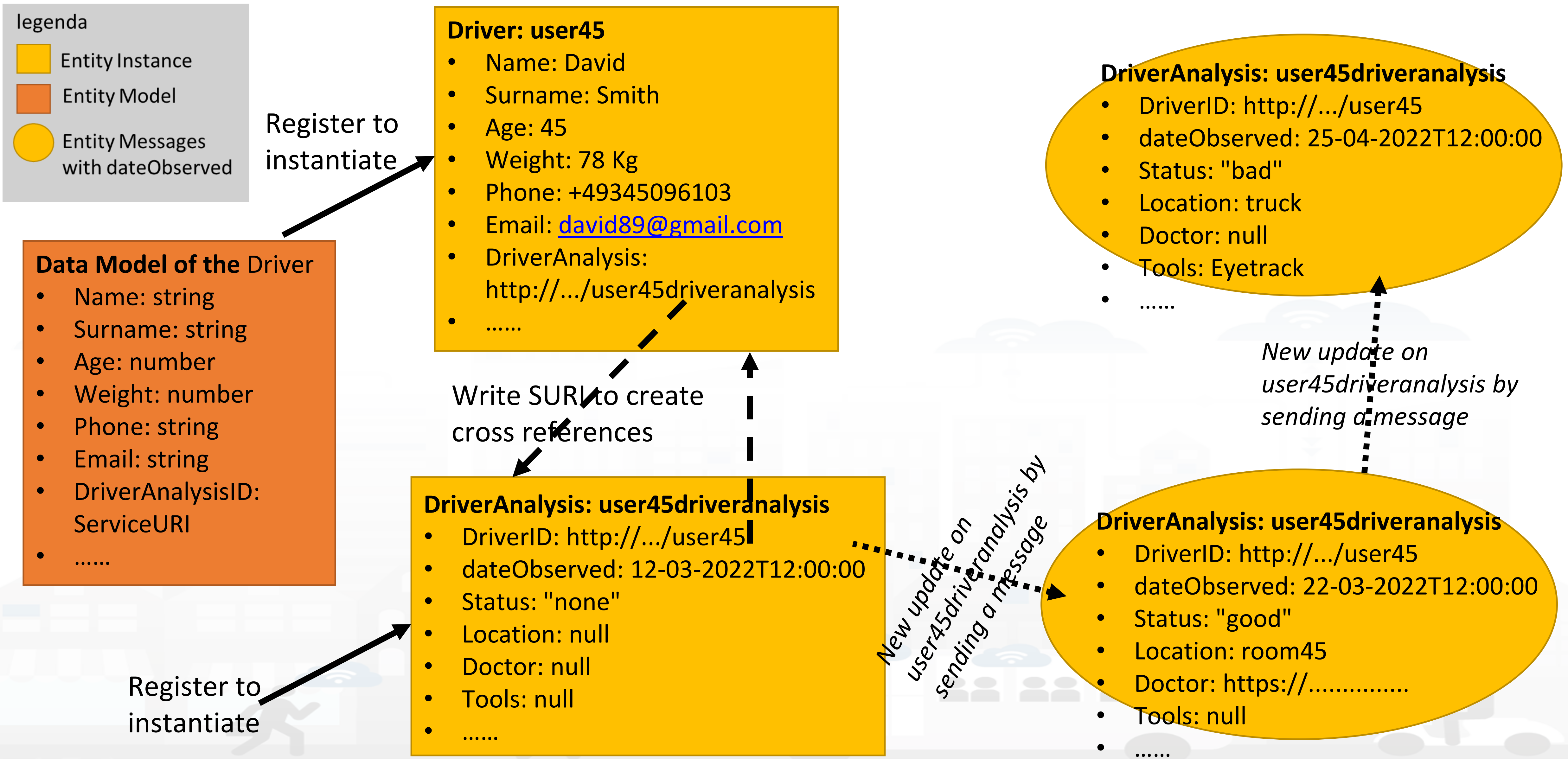
	Parameters	Commons	Operators	Staff	Visitors
Current State	Needs				
	Current Practices				
	Value proposi-tion (Current)				
Future State	Value proposi-tion (Future)				
	Solution				
	Value Capture				
	Key Partners				
	Barriers				



The Dictionary of Entities

Dictionary of Entities

Term	DataModel or Module	Kind	Responsible	Status	Spec where
Driver Healthiness	DriverHealthiness	Entity Model	Dr. Rick Ross	To be done	To be defined
User profile A	DriverA	Entity Model			
Vehicle Event	VehicleEvent	Entity Model			
Remote Consolle	MyOperation	Application	J.T. Kirk	To be done	lost



legenda

- 🟨 Entity Instance
- 🟧 Entity Model
- ● Entity Messages with dateObserved

Data Model of the Driver
- Name: string
- Surname: string
- Age: number
- Weight: number
- Phone: string
- Email: string
- DriverAnalysisID: ServiceURI
- ……

Register to instantiate

Driver: user45
- Name: David
- Surname: Smith
- Age: 45
- Weight: 78 Kg
- Phone: +49345096103
- Email: david89@gmail.com
- DriverAnalysis: http://.../user45driveranalysis
- ……

Write SURI to create cross references

Register to instantiate

DriverAnalysis: user45driveranalysis
- DriverID: http://.../user45
- dateObserved: 12-03-2022T12:00:00
- Status: "none"
- Location: null
- Doctor: null
- Tools: null
- ……

New update on user45driveranalysis by sending a message

DriverAnalysis: user45driveranalysis
- DriverID: http://.../user45
- dateObserved: 22-03-2022T12:00:00
- Status: "good"
- Location: room45
- Doctor: https://..............
- Tools: null
- ……

New update on user45driveranalysis by sending a message

DriverAnalysis: user45driveranalysis
- DriverID: http://.../user45
- dateObserved: 25-04-2022T12:00:00
- Status: "bad"
- Location: truck
- Doctor: null
- Tools: Eyetrack
- ……

API, External Services

External API						
API name	API url and shape	Kind	parameter	Credentials approach	status	Description, Swagger link, Postman, …

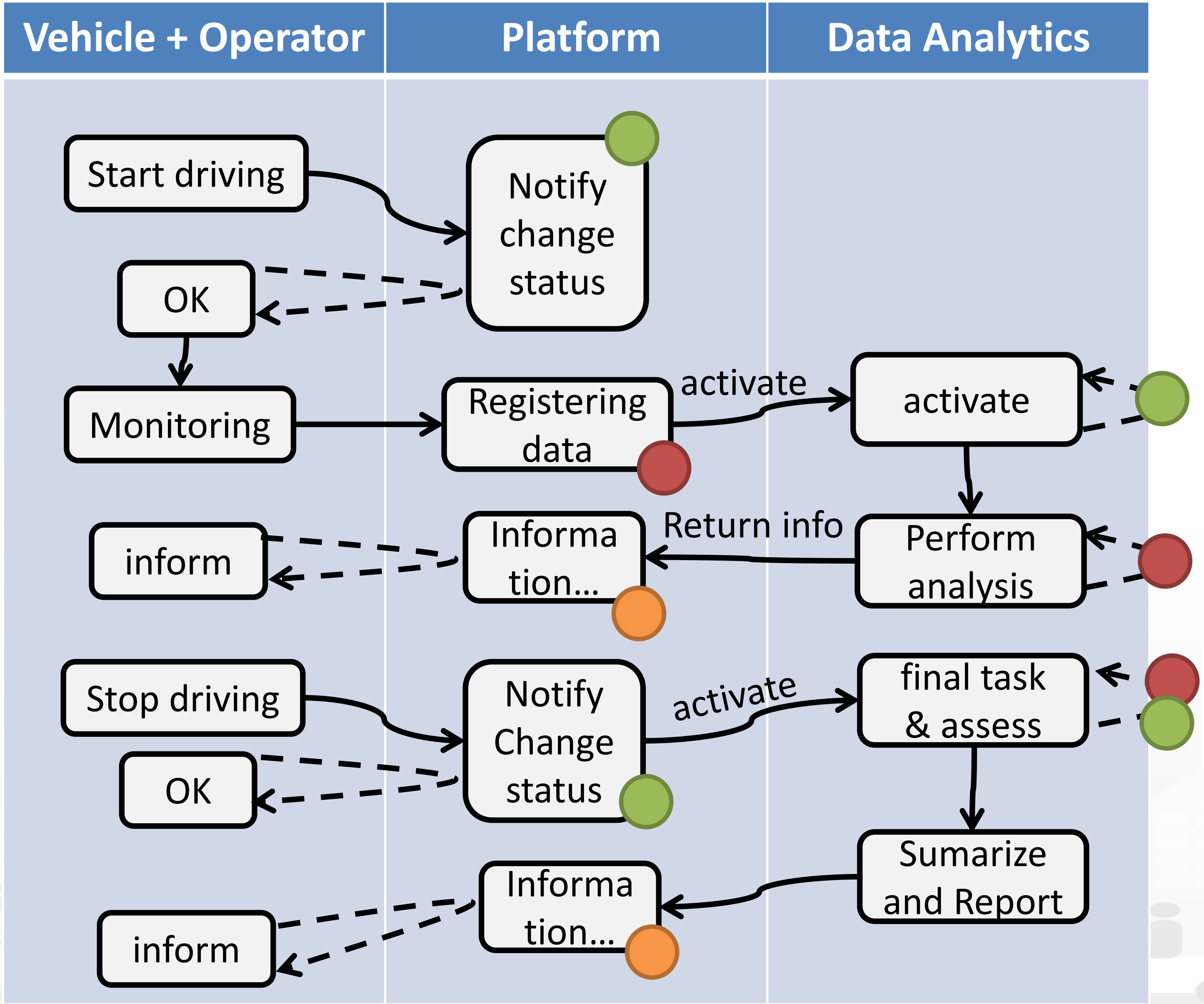
External Services				
URL	parameter	Description	Nature	Subnature

Analysis 2/2

- **Scenarios** describing the application/task, textual definition, with some standard table as UML. The scenarios have to refer to identified entities.
  - https://www.uml-diagrams.org/activity-diagrams-examples.html
- **Use Cases** describing the different cases into the single applications, by using UML formalization, there are specific Use Cases for each Scenario. Please focus on the most relevant, those that are adding value to your solutions. The others can be given for granted in a first phase.
- **Requirements** by using standard tables, using identified **Dictionary** of **Entities**, prioritizing them, setting mandatory/preferred/optional, functional and non-functional, first/second/third release, etc.
- **Sequence Diagrams:** for some of the critical aspects- For example for describing the user interaction, and/or the interaction among major entities, putting in evidence which is the Entity starting the dialogue with respect to the other **Entities** involved (e.g., a client requesting data to the server, a device sending data to the broker). UML sequence diagrams are a suitable formalization for the purpose.
  - https://en.wikipedia.org/wiki/Sequence_diagram

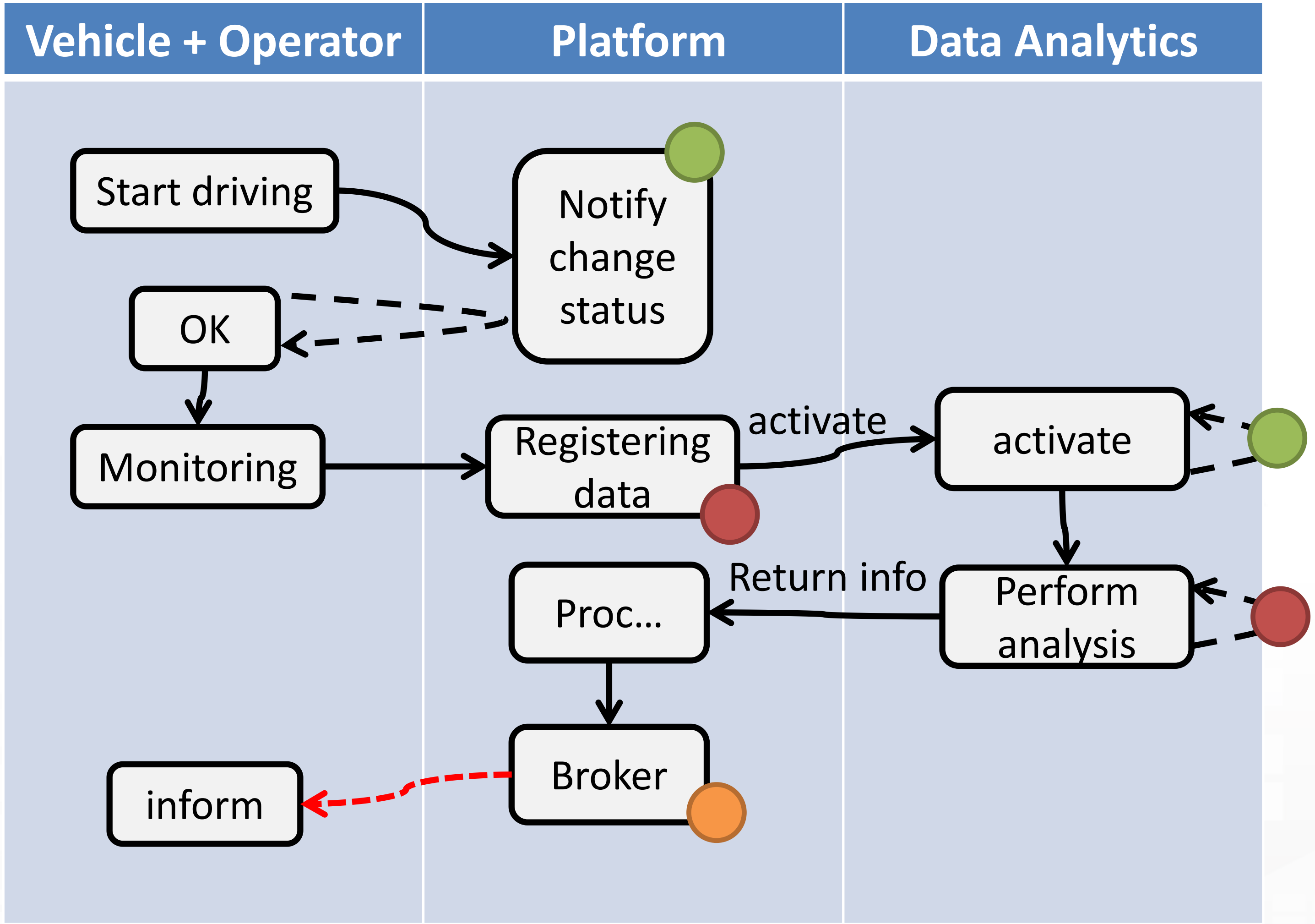
Example: Activity Diagram

- **Continuous Lines** can denote event driven, sync communications… for example by sending data on IoT Broker
- **Dashed lines** can denote Pull data collected periodically. Mainly Async. Communication from Platform to Mobile Devices
- **Coloured Dots** are the different devices data storage



A variant

- Every time a data is entered into the Storage an event occurs into the broker
- The server «Inform» can be subscribed from an IoT App to receive in push these changes **(red dashed line)**



The above figure

- The driver on its Mobile App, he/she marks the start of the driving section, and the App notifies the change of status to the platform via some broker, once performed all the needed verifications (taking some minutes, may be).
- The effective change and authorization to start is made accessible by the platform to the mobile app which is requesting the status in pull (dashed line).
- Then the mobile app starts to monitor the drive status continuously, and send new data (e.g., the level of attention, the road taken, etc.) to the platform via some broker every minute.
- The arrival of new data may activate some data analytics to perform some analysis of the collected data (red dots) and producing results on the platform data. In the case in which the process detected critical conditions for the driver, the assessment procedure on platform may decide to send an event/message (dashed red, in push from platform to clients) to the operator and driver via a Broker to warning the driving of the lack of attention or for some wrong path.
- The event in push from platform to client could be a viable approach on some platforms and may have some limitation on Mobile App in which the interaction paradigm can be changed in a periodic REST call from the Mobile to the Platform.

Legenda on REST Call 1/2

- the **black continuous line** (push) will be used to send some data on the platform broker with a REST call which has to be Authenticated and Authorized according to the OpenId Connect as explained later, and would be in the form of:
  - https://<platformdomain>:8443/orionbrokerfilter/v1/updateContext
  - Or in the form for non TSL protected interaction:
    - *http://iot-app.snap4city.org:80/orion-broker/v1/updateContext?elementid=**ELEMENTID**&k1=**K1**&k2=**K2***

- the **black dashed line** (pull) will be used to request some data from the platform by using a REST call to smart city API (Authenticated and Authorized according to the OpenId Connect as explained later), in the forms:
  - via regular Smart city API by category, etc.
    - http://svealand.snap4city.org/ServiceMap/api/v1/?selection=59.581458578537955;16.71183586120606;59.62875017053684;16.875171661376957&categories=Street_light&maxResults=100&format=json
  - Via Super
    - https://www.disit.org/superservicemap/api/v1/?......
  - Via Super by values
    - https://www.snap4city.org/superservicemap/api/v1/iot-search/?selection=43.77;11.2&maxDists=700.2&model=CarPark
  - https://www.snap4city.org/superservicemap/api/v1/iot-search/?selection=42.014990;10.217347;43.7768;11.2515&model=metrotrafficsensor&valueFilters=vehicleFlow>0.5;vehicleFlow<300

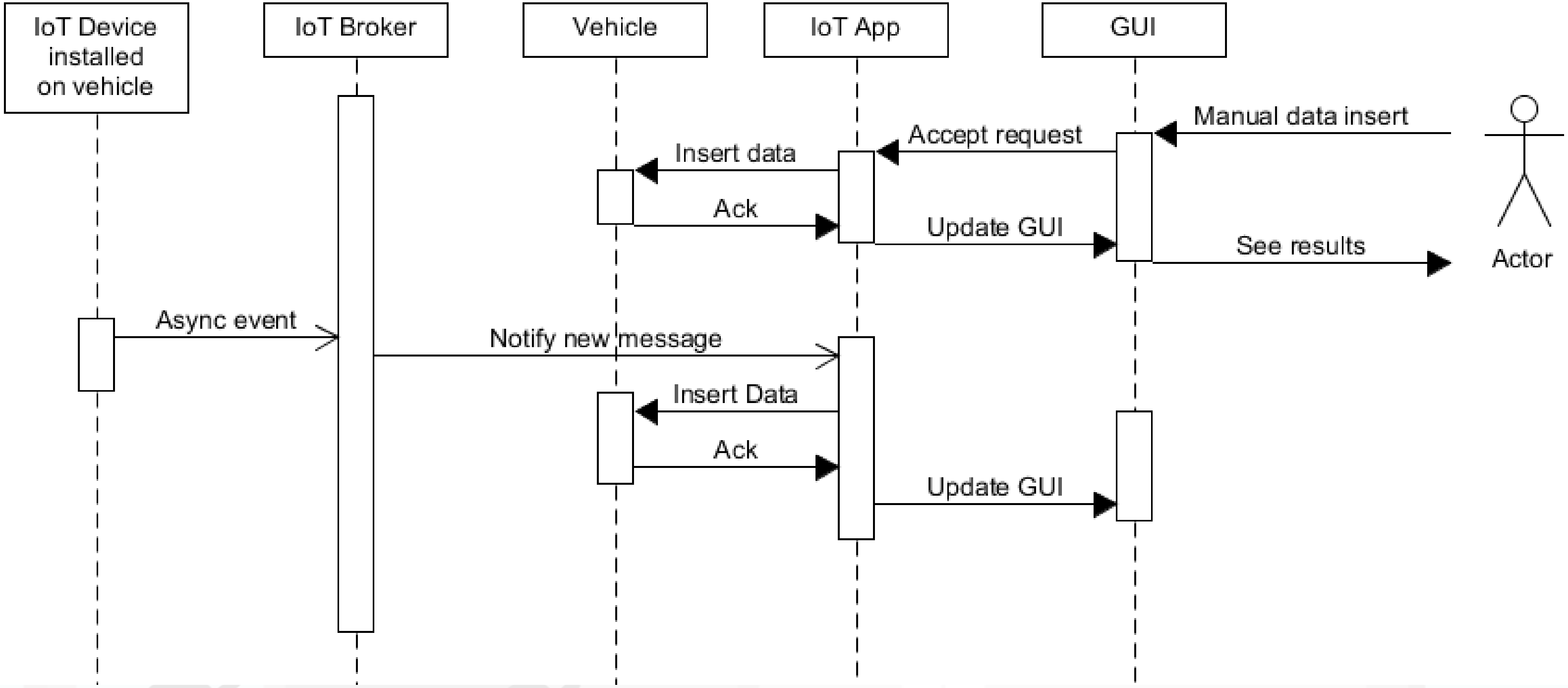
Legenda on REST Call 2/2

- the **red dashed line** (push) will be used to send some data from the platform (from an Orion broker) to some stable IP client or other machine for machine-to-machine communication
  - As a first step the client has to subscribe to some entity on the Orion Broker passing its IP where the broker will have to send the data in push
    - The POST will be in the form of /v1/subscribeContext passing as parameters: elementid (the device ID, and K1, K2) or TSL approach

    -
      ```
      curl -X POST "https://broker1.snap4city.org:8080/v1/subscribeContext?elementid=mypersonaldatatester-device&k1=4e0924a8-fdd6-49cf-8d4a-f49cb5710d8b&k2=240567da-64a4-43b3-8ac9-1265178f3cbe" -H "accept: application/json" -H "Content-Type: application/json" -d "{\"entities\":[{\"type\":\"Ambiental\",\"isPattern\":false,\"id\":\"mypersonaldatatester-device\"}],\"attributes\":[\"temperature\"],\"reference\":\"http://prova/\",\"duration\":\"P1M\",\"notifyConditions\":[{\"type\":\"ONCHANGE\",\"condValues\":\"temperature\"}],\"throttling\":\"PT10S\"}"
      ```
  - Then the broker will send the messages to the subscribed client
  - it could be possible to have this kind of push also by using Kafka and/or WebSocket, but this is possible with simple and direct exposed API to all Snap4City platforms.

- **The external APIs of Snap4City are documented in Swagger**
  - https://www.km4city.org/swagger/external/index.html

Example: Sequence Diagram



Requirements

Requirements

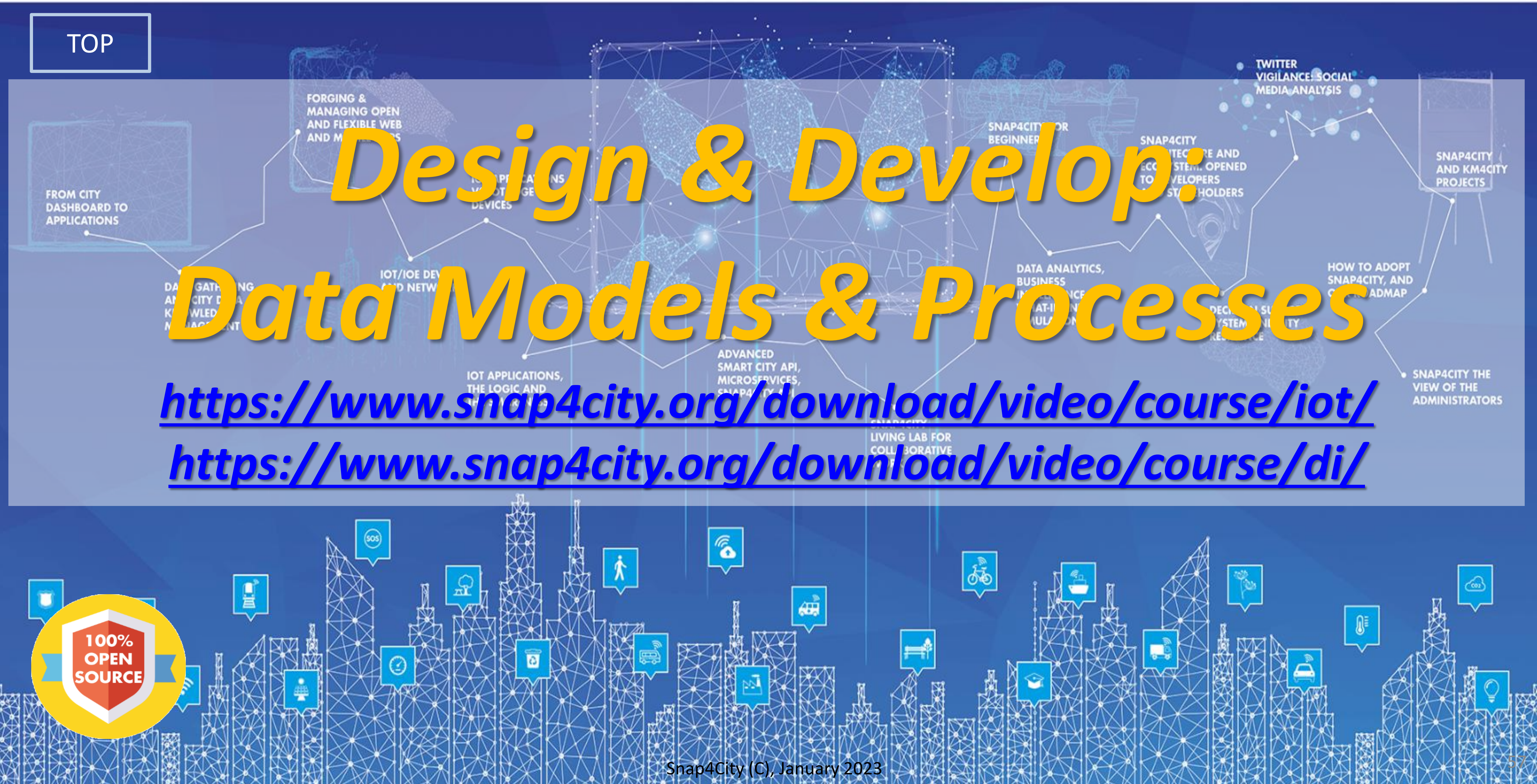
ID	Main Entity / Area	Description	Relevance / Priority	Main Tool-Module / Entity involved	Status	Source Code
D1	Operator	The Operator has to be authorized to register Drivers	mandatory	OperatorTool	Not developed	JavaScript by xxxx on GitLab ….
D2	Driver	The Drive can verify its registration by putting Password to access to its data on the solution	optional	Web and/or Mobile App accessible for the Drivers	accessible as open source	Yes In Java with AGPL licence

TOP

Design & Develop: Data Models & Processes

https://www.snap4city.org/download/video/course/iot/

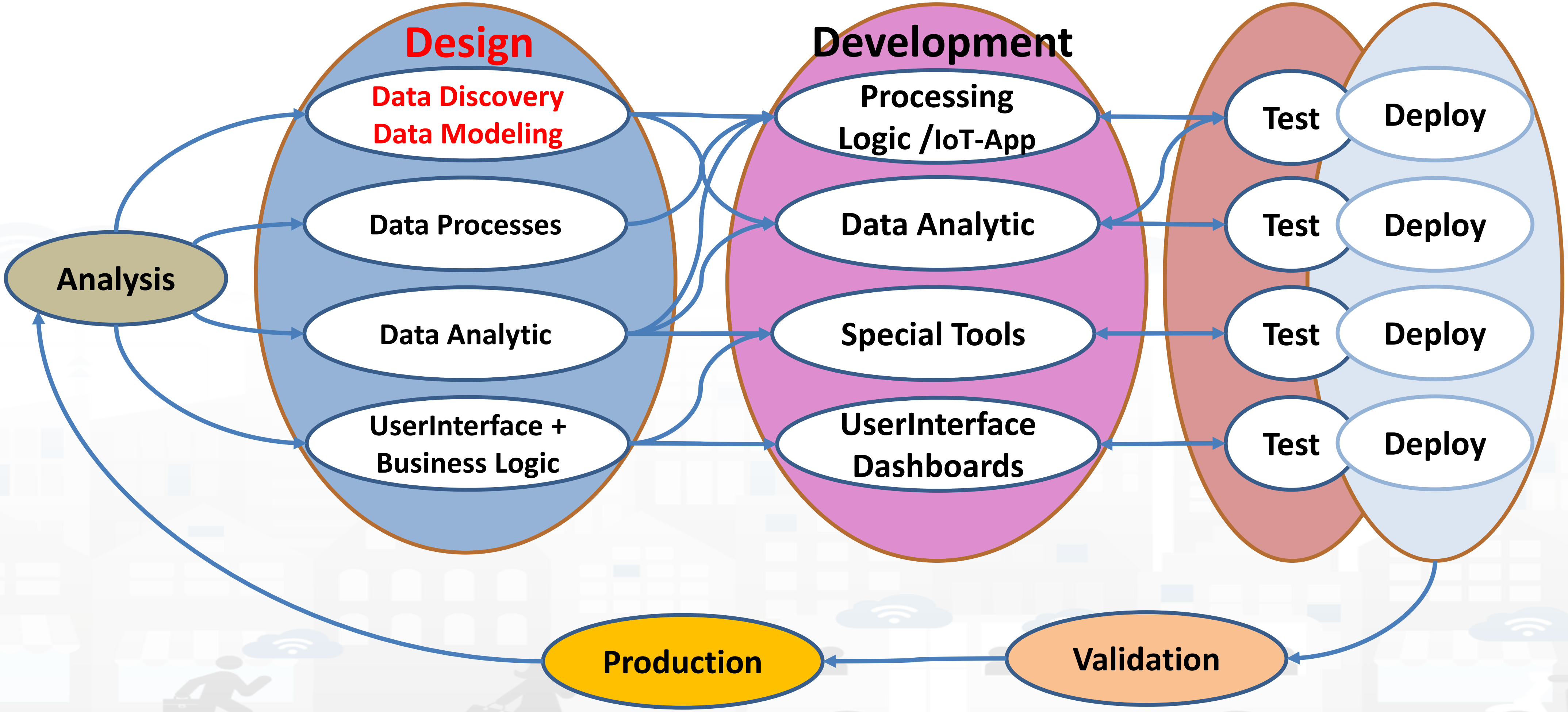
https://www.snap4city.org/download/video/course/di/


FORGING & MANAGING OPEN AND FLEXIBLE WEB AND M
FROM CITY DASHBOARD TO APPLICATIONS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY AND KM4CITY PROJECTS
DATA ANALYTICS, BUSINESS
HOW TO ADOPT SNAP4CITY, AND
IOT APPLICATIONS, THE LOGIC AND
ADVANCED SMART CITY API, MICROSERVICES,
LIVING LAB FOR COLLABORATIVE
SNAP4CITY THE VIEW OF THE ADMINISTRATORS
LIVING LAB
100% OPEN SOURCE

TOP

Design: Data Discovery

Development Life Cycle
Smart Solutions



Data Discovery

RULE OF THUMB

- Performed by analyzing data from:
  I. identified scenarios from the **Snap4City Innovation Matrix**
  II. main organizations (via interviews)
  III. other stakeholders (via interview and web pages)
  IV. regional, national and international sources:
     I. open data portals, weather sources,
     II. IOT networks, etc. via web pages and sites
  V. Mobile Applications (via Snap4City API)
  VI. Snap4City portal [Https://www.snap4city.org](https://www.snap4city.org)
  VII. etc.

- Exploiting Snap4City experience, data and tools
- By following the Snap4City guidelines on Data Search on web and world reported in the training course and on Snap4City.org portal.

TOP

Design: from Data Modeling to Data Ingestion

What About Entity Instances / IoT Devices, Time Series

Entity / IOT Device

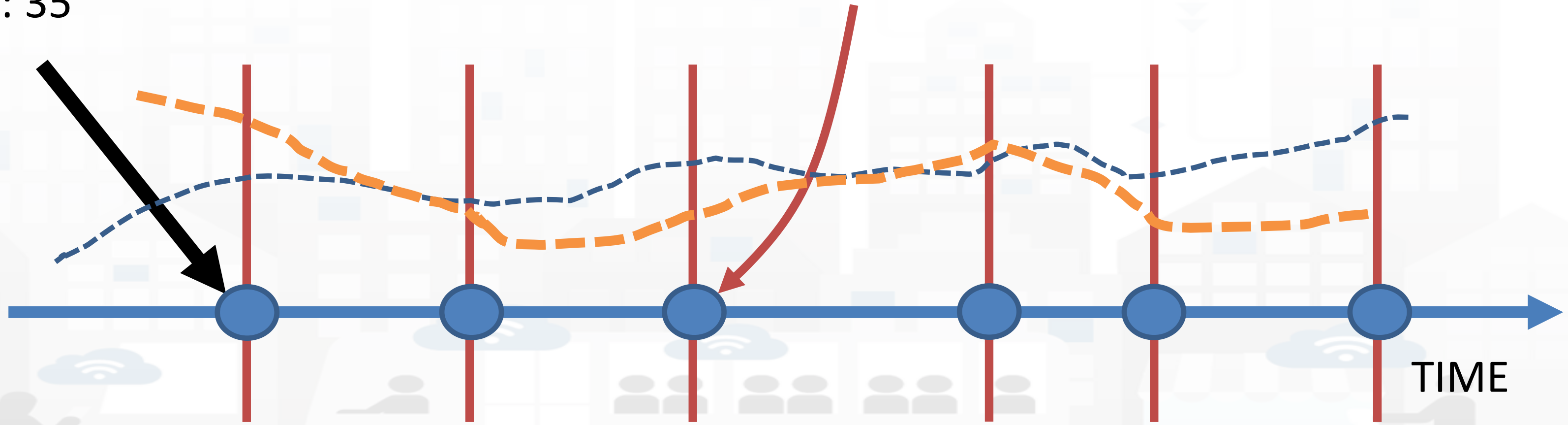
Entity: IOT Device

Sends a message

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

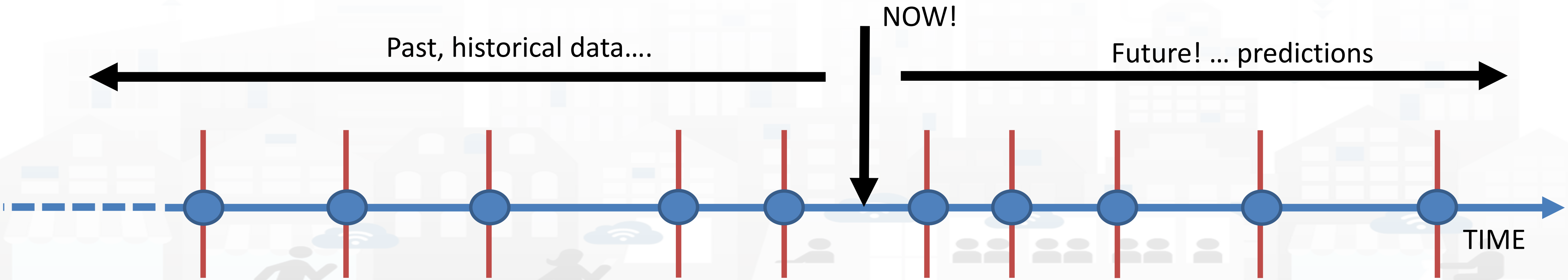
dateObserved	Temp	Humidity
02-04-2020 10:30	34.5	23
02-04-2020 10:40	36.5	24
02-04-2020 10:50	36.0	22.5

- A set of data coming from an Entity Instance / 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 registered an Entity Instance / IoT Device
  - 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.
    - Loaded in PULL with some File or Data Table Loader



Entity Instance / IoT Devices

IoT Device Models

register

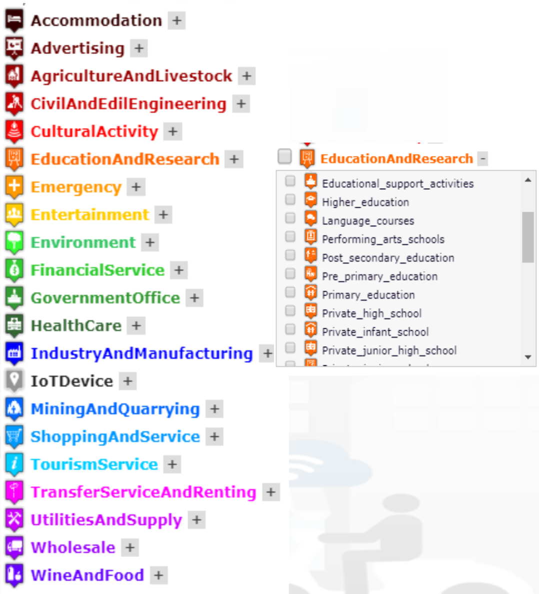
IoT Device

- Name:…..
- Model:…..
- Position: …….

IoT Device Variables

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

- Conceptually are Entity Instances / IoT 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 Entity Instances / IoT Devices can be:
  - **Entity Model/ IoT Device Models**, as: «personal coffee machine»
  - **Entity Instance / IoT Device** name, as: «mycoffemachine1», «CM23»
  - **Entity Variable / IoT Device Variable**, as: «Temperature»
  - **Entity Message / IoT Device Message**

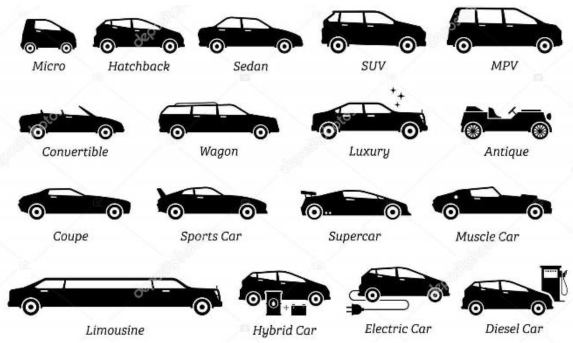


- Accommodation +
- Advertising +
- AgricultureAndLivestock +
- CivilAndEdilEngineering +
- CulturalActivity +
- EducationAndResearch +
- Emergency +
- Entertainment +
- Environment +
- FinancialService +
- GovernmentOffice +
- HealthCare +
- IndustryAndManufacturing +
- IoTDevice +
- MiningAndQuarrying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

□ 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

Mobile Entities



Mobile Entity Models

register

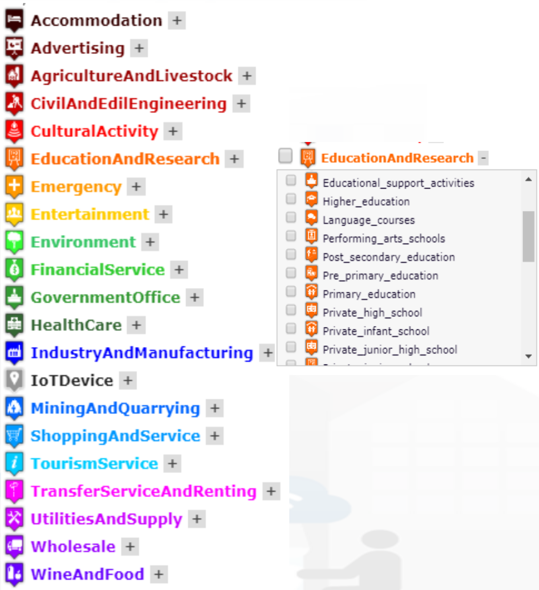
Mobile Entity Instances

- Name:…..
- Model:…..
- Spec:…

Mobile Entity Variables

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

- They are a special case of IoT Devices
  - they are managed as IoT Devices 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 can be:
  - **Mobile Entity/Device Model**, for example: «sedan»
  - **Mobile Entity Instance / Device** name, as: «BMW JD7356HD», «Ford KO786KK»
  - **Mobile Entity/Device Variable**, for example: «velocity»



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

□ 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

TOP

Trick and tips on Time Series

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

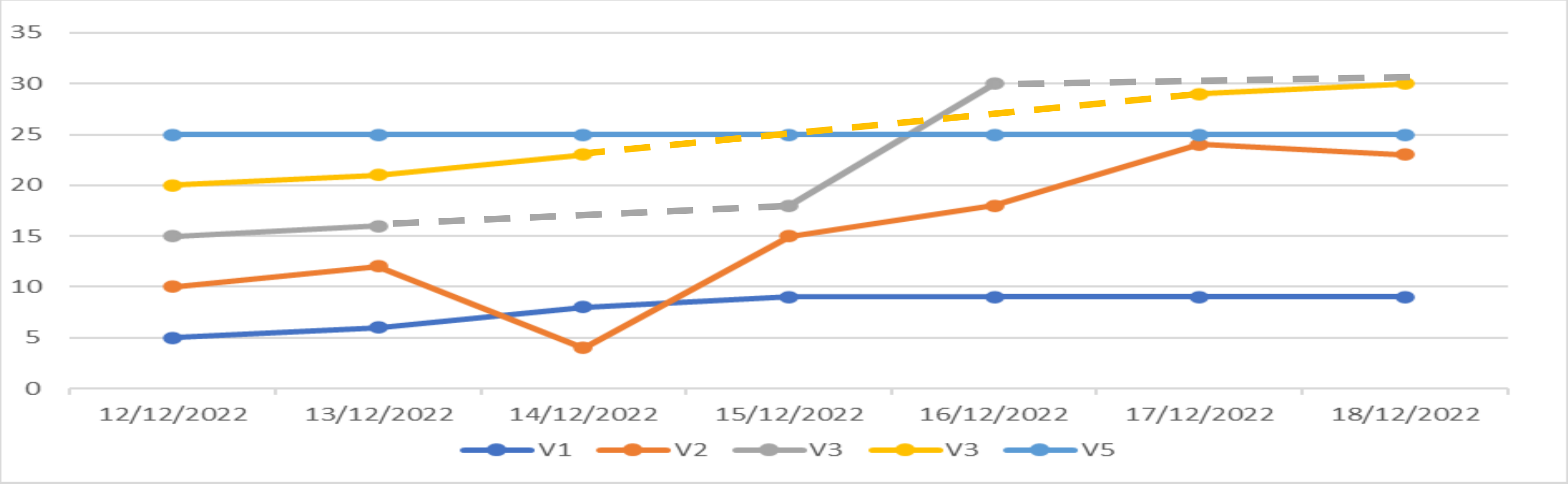
- valid in all messages on graphs
- With holes in tables

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

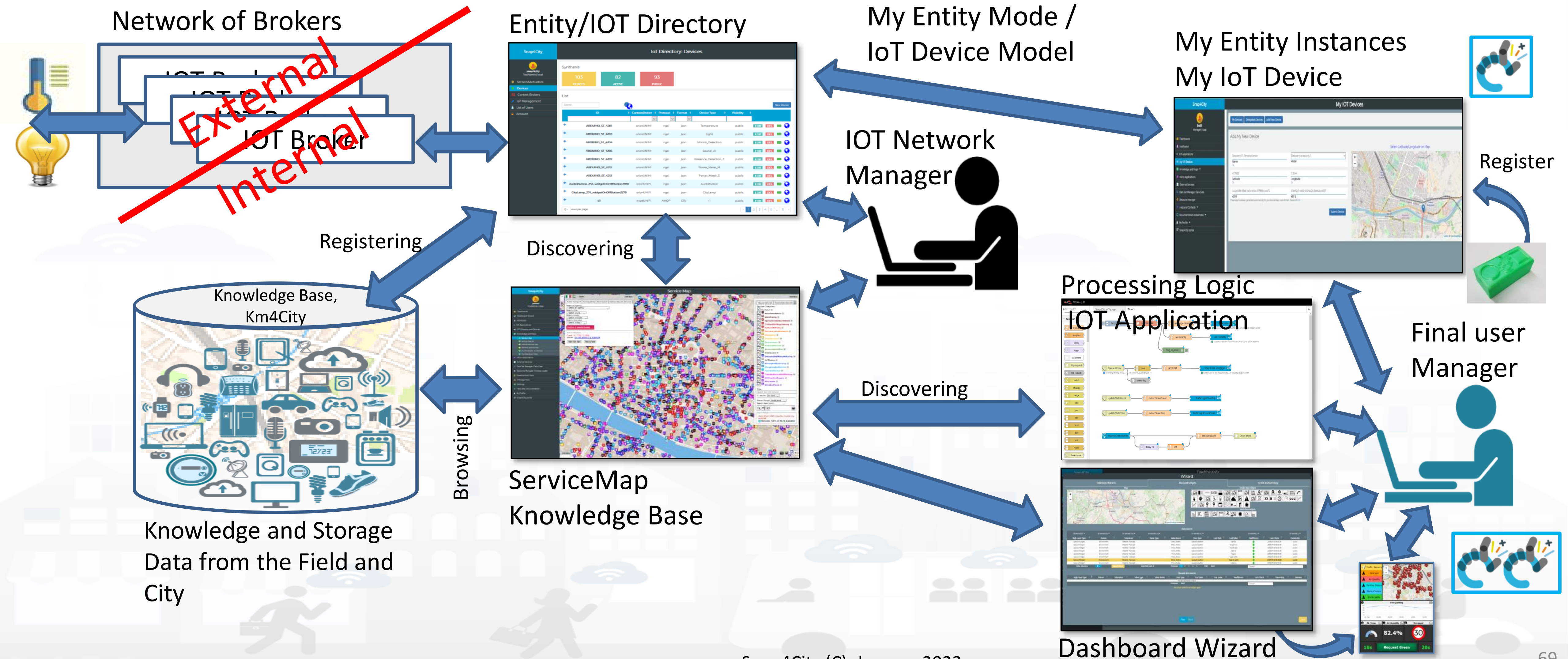
12-12-2022	13-12-22	14-12-22	15-12-22	16-12-22	17-12-22	18-12-22
V1: 5	5	6	8	9	9	9
V2: 10	10	12	4	15	18	24
V3: 16	15	16	null	18	30	null
V4: 20	20	21	23	null	null	29
V5: 25	25	25	25	25	25	25



TOP

Design: from Data Modelling to Data Ingestion

IOT Network Manager vs Final User



Network of Brokers

IOT Broker

External

Internal

Entity/IOT Directory

My Entity Mode / IoT Device Model

My Entity Instances My IoT Device

Register

IOT Network Manager

Registering

Discovering

Knowledge Base, Km4City

Knowledge and Storage Data from the Field and City

Browsing

ServiceMap Knowledge Base

Discovering

Processing Logic IOT Application

Final user Manager

Dashboard Wizard

The Data Models can be simply instantiated from

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

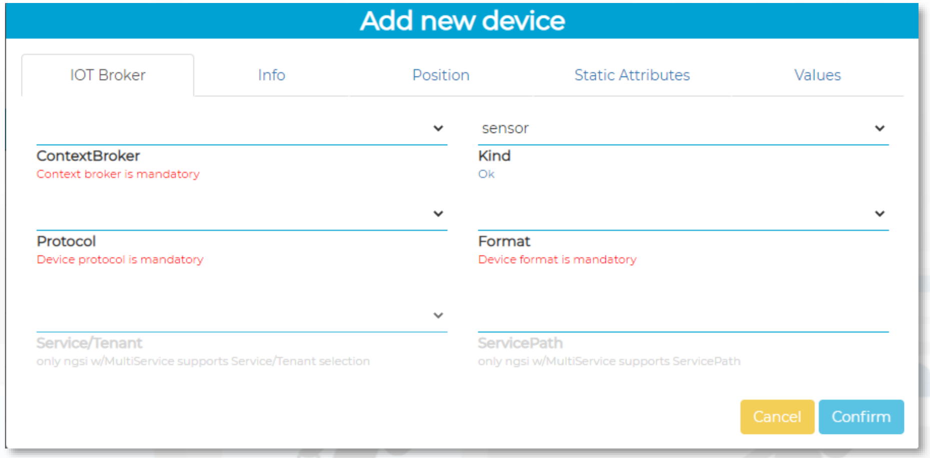
b) **Entity Model / IoT Device Model** which are accessible into the Snap4City environment

c) **Excel files by using Data Table tool**, which extracts the model from the excel table and automatically creates **Entity Model /** IoT Device Model, Entity Instances / IoT Devices and data attached to them

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

Entity Models / IoT Device Data Model (1)

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



ContextBroker
Context broker is mandatory

sensor
Kind
Ok

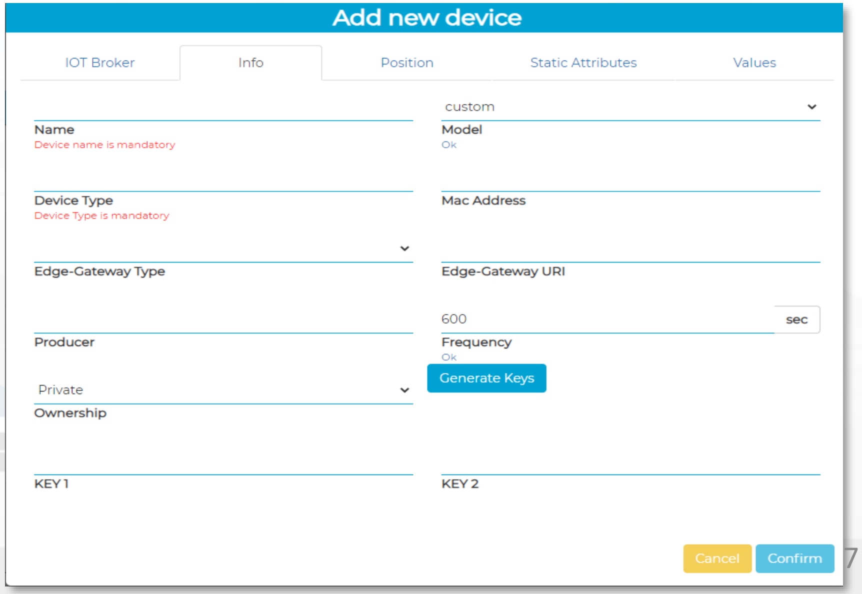
Protocol
Device protocol is mandatory

Format
Device format is mandatory

Service/Tenant
only ngsi w/MultiService supports Service/Tenant selection

ServicePath
only ngsi w/MultiService supports 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



Name
Device name is mandatory

custom
Model
Ok

Device Type
Device Type is mandatory

Mac Address

Edge-Gateway Type

Edge-Gateway URI

Producer

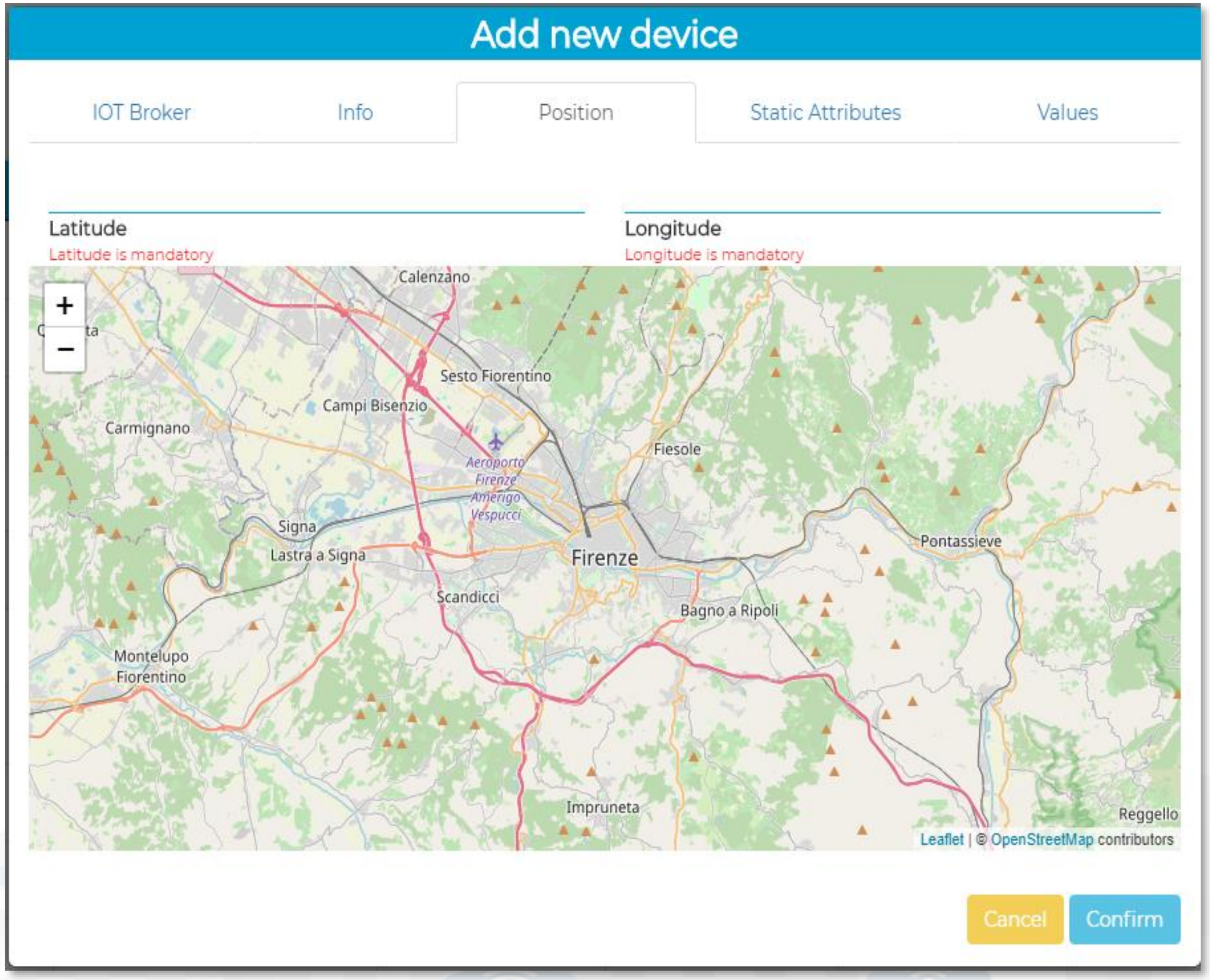
600 sec
Frequency
Ok

Private
Ownership

KEY 1

KEY 2

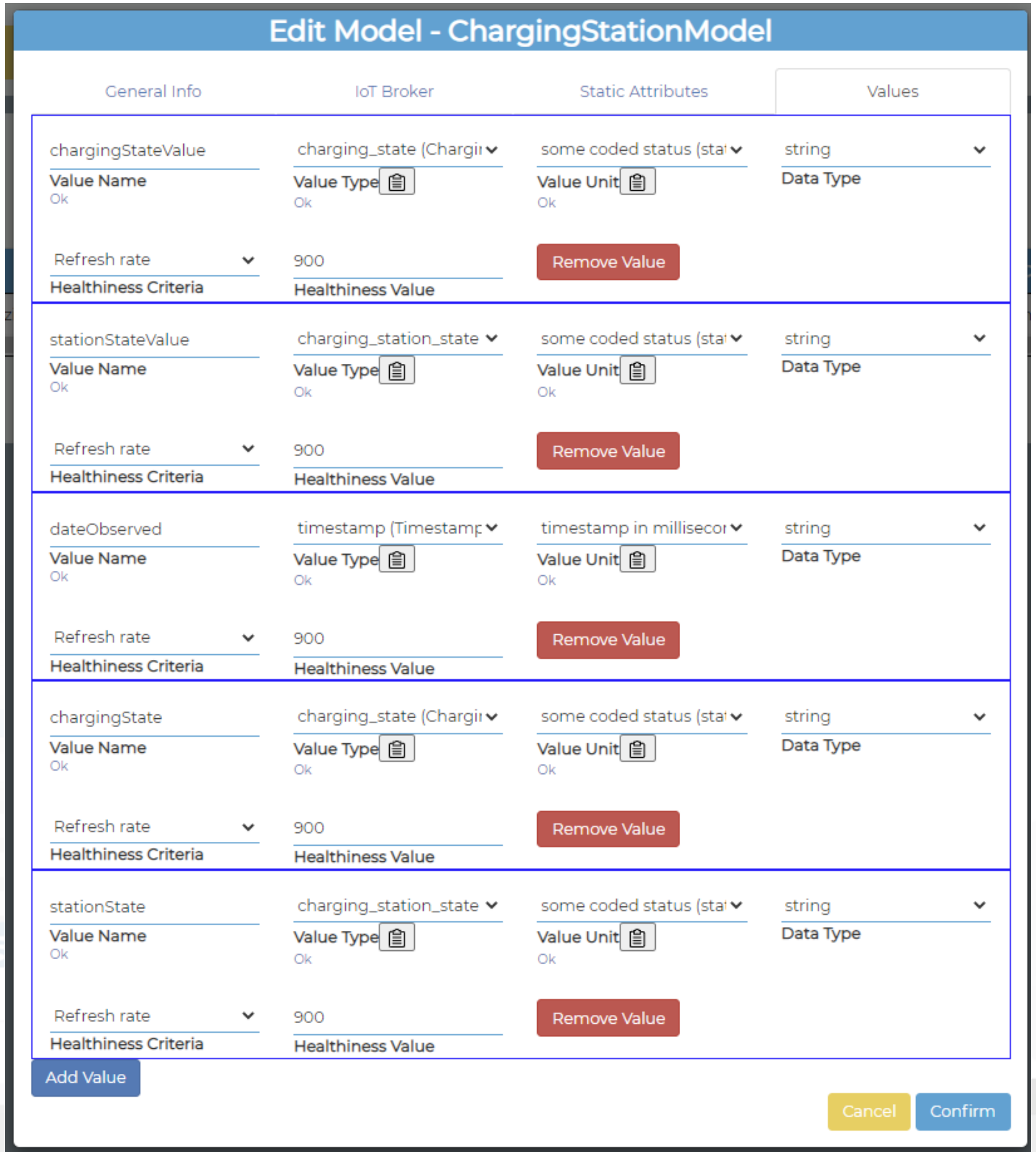
Entity Models / IoT Device Data Model (2)



Add new device

Latitude
Latitude is mandatory

Longitude
Longitude is mandatory



Edit Model - ChargingStationModel

chargingStateValue Value Name Ok	charging_state (Chargi Value Type Ok	some coded status (sta Value Unit Ok	string Data Type
Refresh rate Healthiness Criteria	900 Healthiness Value	Remove Value	
stationStateValue Value Name Ok	charging_station_state Value Type Ok	some coded status (sta Value Unit Ok	string Data Type
Refresh rate Healthiness Criteria	900 Healthiness Value	Remove Value	
dateObserved Value Name Ok	timestamp (Timestamp Value Type Ok	timestamp in millisecor Value Unit Ok	string Data Type
Refresh rate Healthiness Criteria	900 Healthiness Value	Remove Value	
chargingState Value Name Ok	charging_state (Chargi Value Type Ok	some coded status (sta Value Unit Ok	string Data Type
Refresh rate Healthiness Criteria	900 Healthiness Value	Remove Value	
stationState Value Name Ok	charging_station_state Value Type Ok	some coded status (sta Value Unit Ok	string Data Type
Refresh rate Healthiness Criteria	900 Healthiness Value	Remove Value	

Entities from models to variables

Where	Entity Model (IOT Device Model)	Entity Instance (IOT Device)	Entity Message a Temporal Instance
Broker	**Broker: OrionUNIFI**		
Broker	**Protocol: NGSI**		
Info	**ID: string**	ID: "park45"	**park45**
Position	**GPS: lat, long**	GSP Position: 43.12, 11.34	GSP Position: 44.12, 11.12
Static attribute	**Description: string**	Description: "parking massaia"	
Static attribute	**Location: string**	Location: "Via Massaia"	
Static attribute	**Civic Number: string**	Civic Number: 3	
Static attribute	**MaxCapacity: number, cars**	MaxCapacity: 456	
Values	**dateObserved: Timestamp**		23-12-2019T20:13:12…
Values	**FreeSlots: Integer, #**		345
Values	**Humidity: float, %**		25,5
Values	**Temperature: float, celsius**		34

Model meaning

- **ID:** is the unique identifier for reconnecting Temporal Instances with registered IOT 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

Using the Entity Model (IOT Device Model) notes!!!

- Once performed the **Entity Model**, a number of **Entity Instances** (IoT Devices) can be produced by **using the model as a Template**
  - **NOTE:** the produced Instances are not going to change in structure if the Model is modified.
  - *All the cookies you've made don't break when your template breaks ! ☺*

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

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*

TOP

Custom Data Modeling example

Example 1

IoT Device Model: Driver
Nature:…………….
Subnature: …………………
Lat,lon: Default (they do not need to be specified in the variables, they are provided by default, but values have to be imposed at the instantiation of the device from model), they are float
Device in Mobility: No (the variable do not need to be specified, while the value has to be set to state if the Lat,Lon are going to change, moving the device or not)

Value_name	Value Type	Value Unit	Data Type
dateObserved	**Timestamp**	**Timestamp in ms**	**String**
identifier	ID	text	String
name	entity	text	String
surname	entity	text	String
age	age	number	Integer
sex	status	some coded status	String
language	entity	text	String
email	entity	text	String
phone	entity	text	String
address	entity	text	String
locality	entity	text	String
city	entity	text	String
nationality	entity	text	String
civicNmber	entity	text	String
dateofBorn	DateTime	Timestamp in ms	String
gender	status	some coded status	String
driverHelthiness	Identifier	ServiceURI	String
driverEvent	Identifier	ServiceURI	String
driverAnalysis	Identifier	ServiceURI	String
Vechicle	Identifier	ServiceURI	String

Example 2

IoT Device Model: driverHelthiness
Nature:…………….
Subnature: …………………
Lat,lon: ……….
Device in Mobility: ……….

Value_name	Value Type	Value Unit	Data Type
dateObserved	Timestamp	Timestamp in ms	String
kind			
levelAttentionFactor1			
levelAttentionFactor2			
driver	Identifier	ServiceURI	String

Example 3

IoT Device Model: Vehicle
Nature:................
Subnature: ....
Lat,lon: ....
Device in Mobility: ....

Value_name	Value Type	Value Unit	Data Type
dateObserved	Timestamp	Timestamp in ms	String
producer	entity	text	String
model	entity	text	String
plate	entity	text	String
companyID	entity	text	String
velocity	velocity	km/h	float
acceleration	acceleration	m/s2	float
Status	status	some coded status	String
energyLevel	energy level	percentage	Float
kmTotal	distance	km	Float
thankLevel	energy level	percentage	Float
vehicleEvent	Identifier	ServiceURI	String

Example 4

IoT Device Model: VehicleEvent

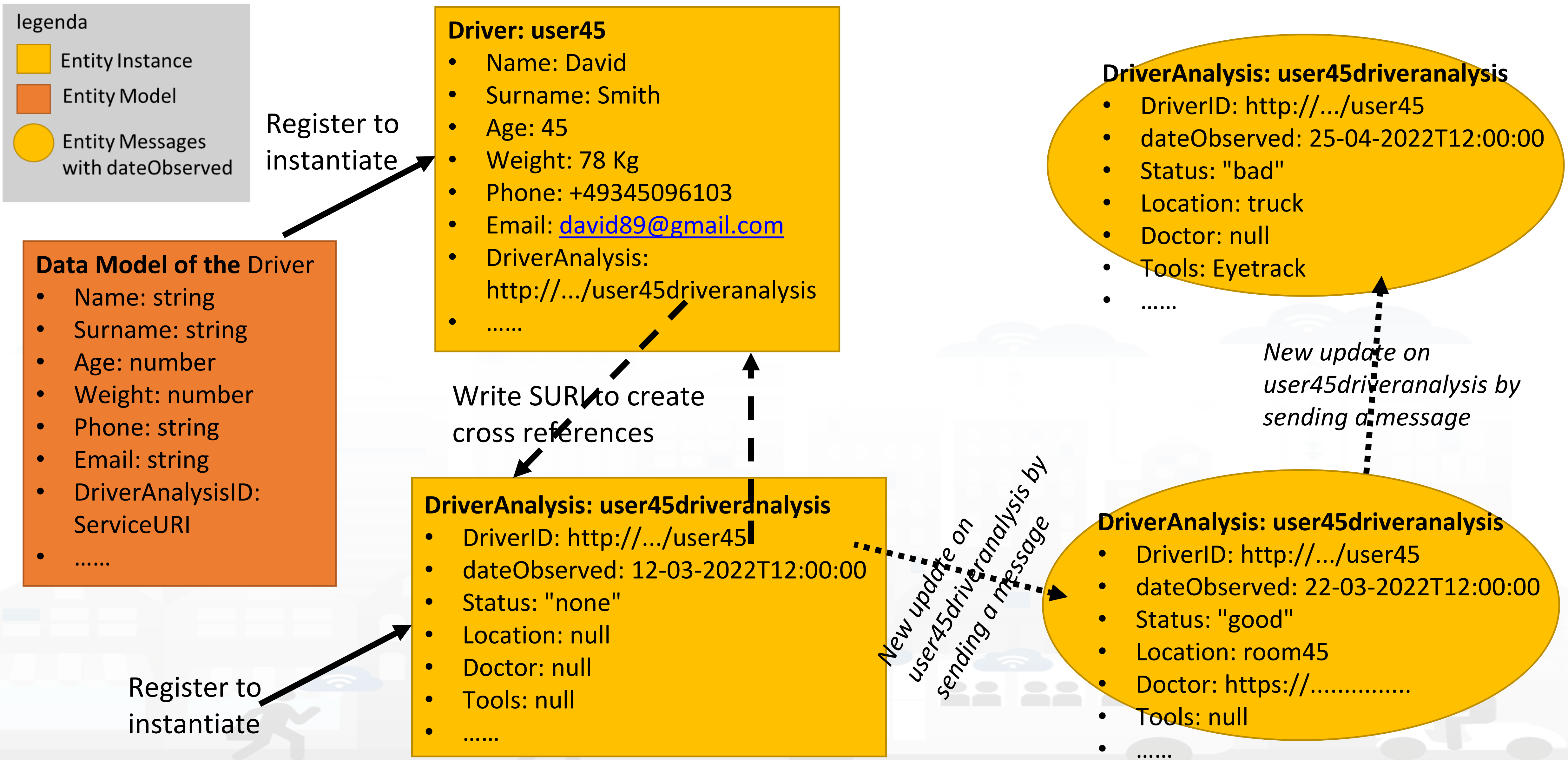
Nature:................

Subnature: ....

Lat,lon: ....

Device in Mobility: ....

Value_name	Value Type	Value Unit	Data Type
dateObserved	Timestamp	Timestamp in ms	String
eventID	ID	text	String
eventKind	status	some coded status	String
status	status	some coded status	String
vehicle	Identifier	ServiceURI	String



legenda

- 🟨 Entity Instance
- 🟧 Entity Model
- ● Entity Messages with dateObserved

Data Model of the Driver
- Name: string
- Surname: string
- Age: number
- Weight: number
- Phone: string
- Email: string
- DriverAnalysisID: ServiceURI
- ……

Register to instantiate

Driver: user45
- Name: David
- Surname: Smith
- Age: 45
- Weight: 78 Kg
- Phone: +49345096103
- Email: david89@gmail.com
- DriverAnalysis: http://.../user45driveranalysis
- ……

Write SURI to create cross references

Register to instantiate

DriverAnalysis: user45driveranalysis
- DriverID: http://.../user45
- dateObserved: 12-03-2022T12:00:00
- Status: "none"
- Location: null
- Doctor: null
- Tools: null
- ……

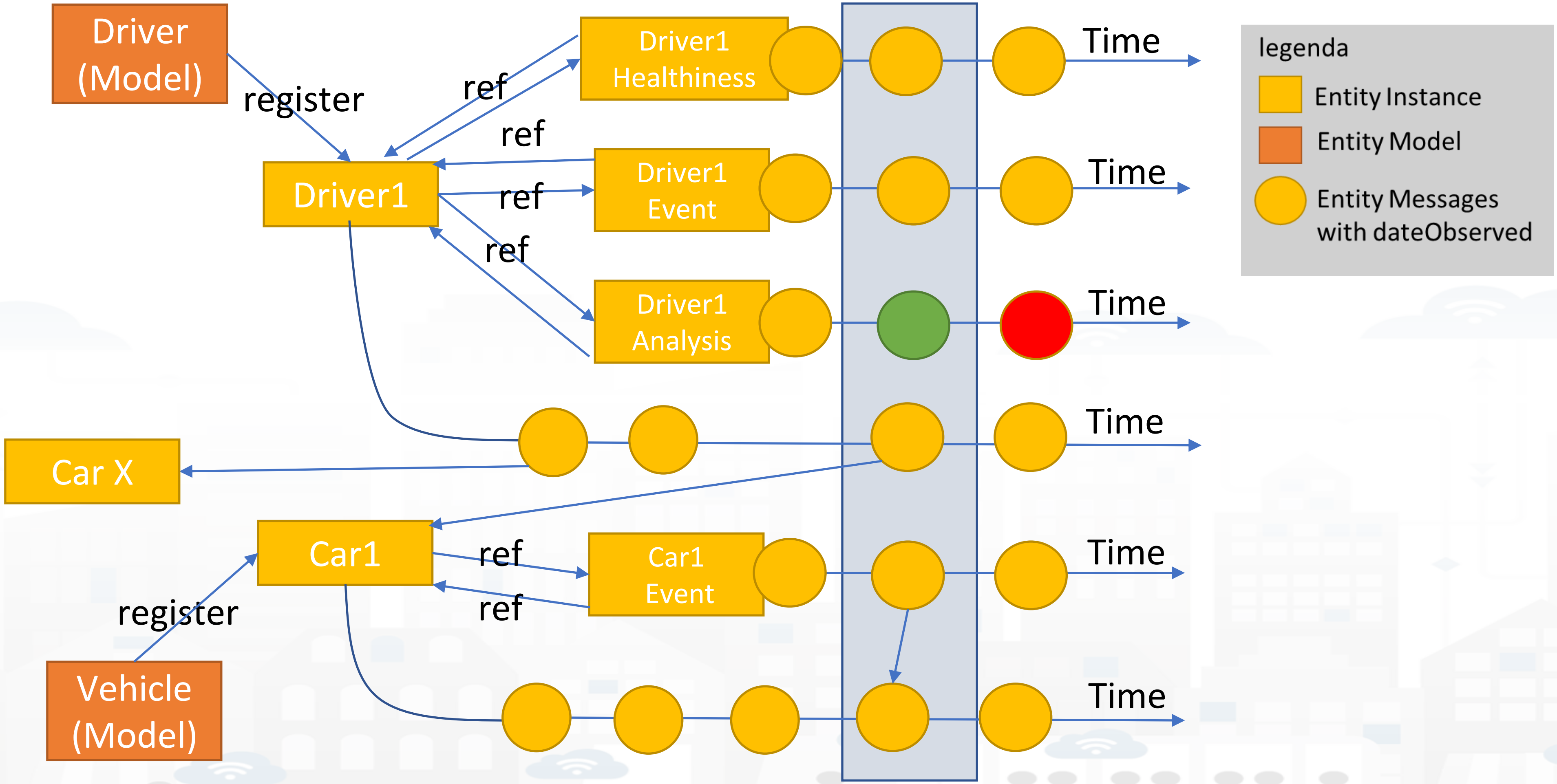
New update on user45driveranalysis by sending a message

DriverAnalysis: user45driveranalysis
- DriverID: http://.../user45
- dateObserved: 22-03-2022T12:00:00
- Status: "good"
- Location: room45
- Doctor: https://..............
- Tools: null
- ……

New update on user45driveranalysis by sending a message

DriverAnalysis: user45driveranalysis
- DriverID: http://.../user45
- dateObserved: 25-04-2022T12:00:00
- Status: "bad"
- Location: truck
- Doctor: null
- Tools: Eyetrack
- ……

Example of Data Model Diagram

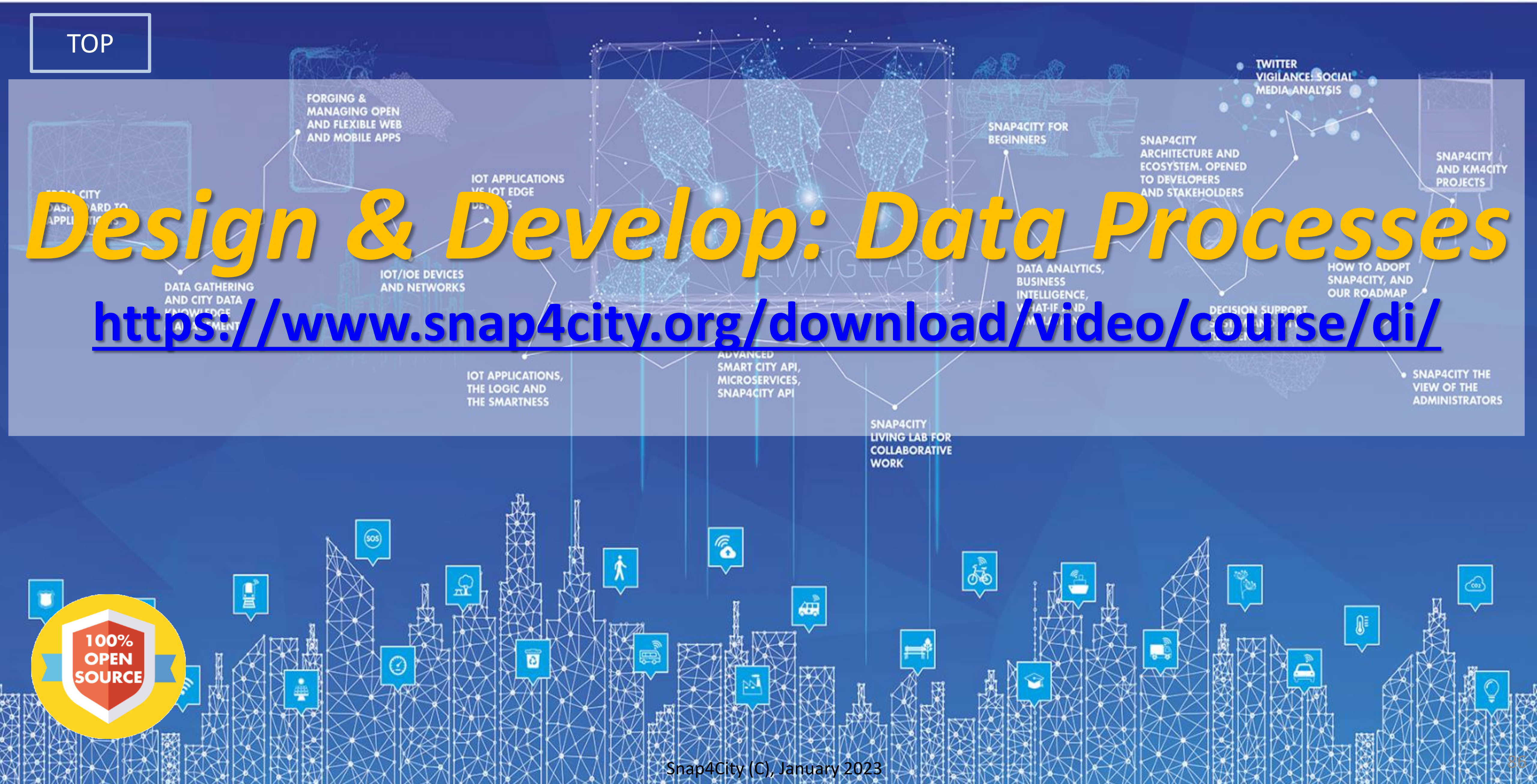


At the «same time slot»

TOP

Design & Develop: Data Processes

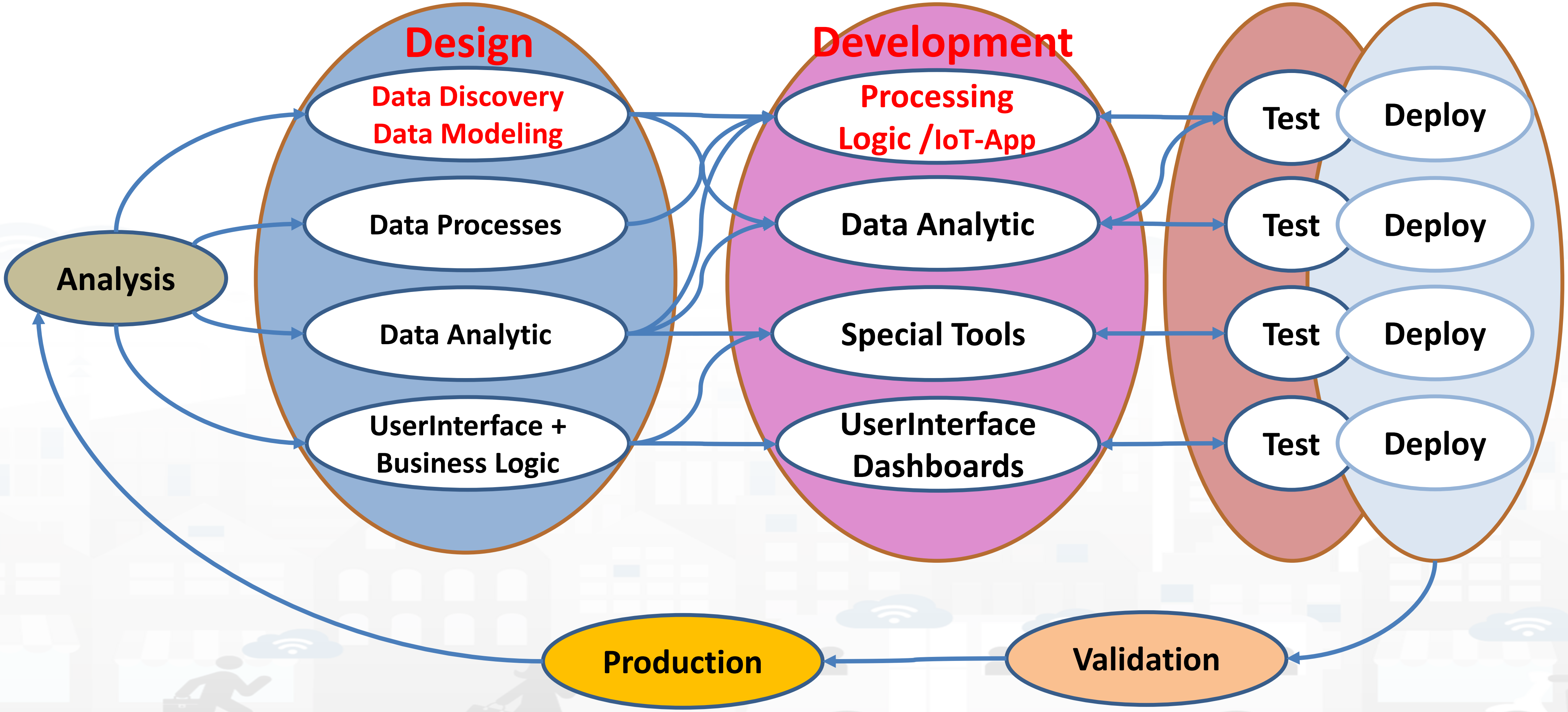
https://www.snap4city.org/download/video/course/di/


FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS
IOT APPLICATIONS VS IOT EDGE DEVICES
SNAP4CITY FOR BEGINNERS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY ARCHITECTURE AND ECOSYSTEM. OPENED TO DEVELOPERS AND STAKEHOLDERS
SNAP4CITY AND KM4CITY PROJECTS
FROM CITY DASHBOARD TO APPLICATIONS
DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT
IOT/IOE DEVICES AND NETWORKS
LIVING LAB
DATA ANALYTICS, BUSINESS INTELLIGENCE, WHAT-IF AND
HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP
DECISION SUPPORT
ADVANCED SMART CITY API, MICROSERVICES, SNAP4CITY API
IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS
SNAP4CITY THE VIEW OF THE ADMINISTRATORS
SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK
SOS
100% OPEN SOURCE

Activities for IoT App data processes

- **Data Ingestion, gathering, harvesting, grabbing**
- **Data Transformation, transcoding, decoding, converting**
- **Data load to storage, retrieve from storage**
  - the load is typically performed loading data on some Internal IoT Orion Broker V2, or on some MyKPI storage
  - the retrieval is typically performed using one of the several query / search nodes.
  - Many other kind of storage connections are accessible in Snap4City IoT App
- **Data Production, generation, reformatting, etc.**
- **Data Publication, post in other channels of any kind, etc.**

Development Life Cycle Smart Solutions



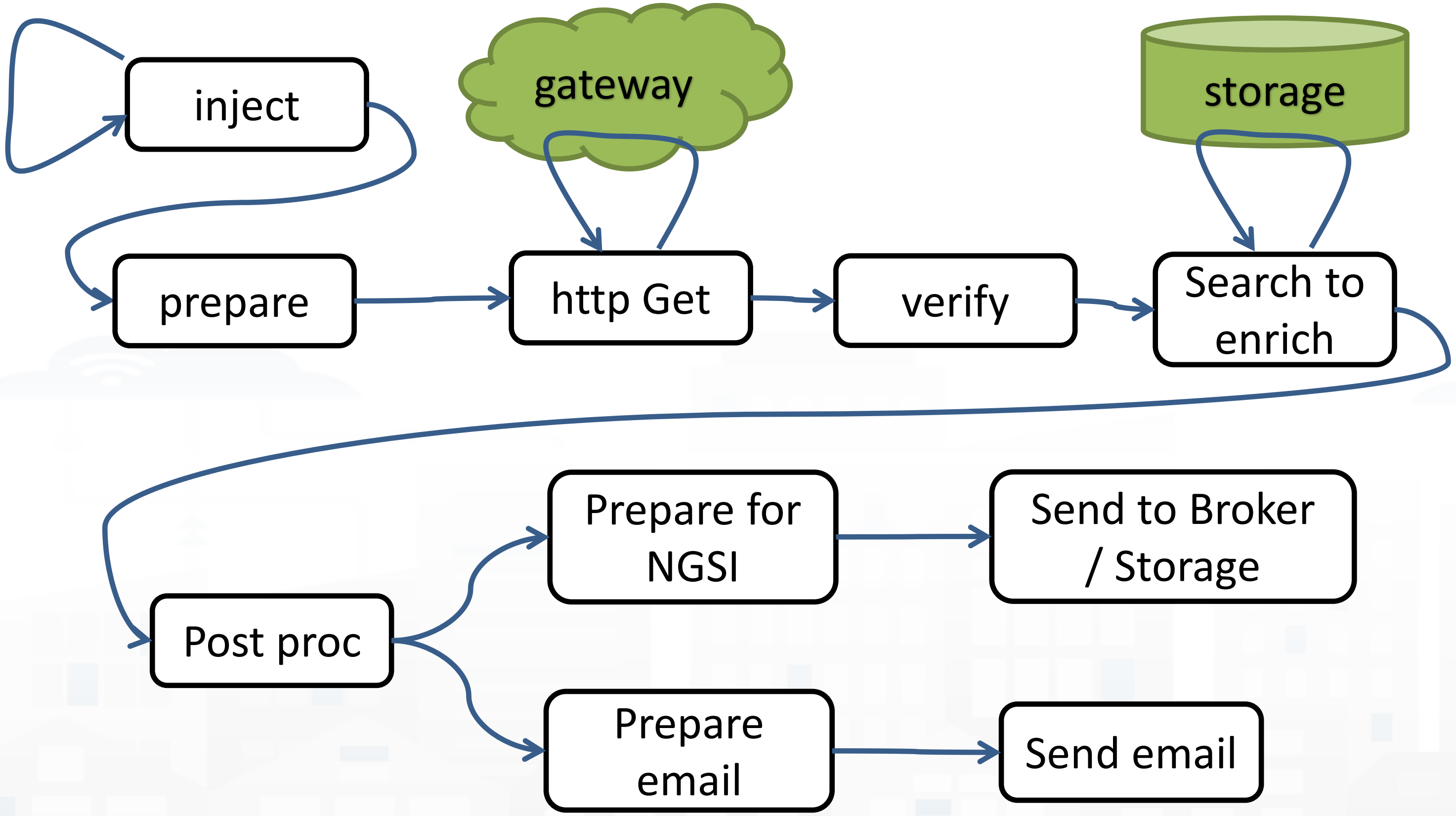
TOP

Design: Data Processes

How to Design

1. Business Logic is going to be implemented in Processing Logic (IoT App), with a set of flows.
2. Decompose you 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 Processing Logic (IoT App) Node-RED flows by adding nodes needed
7. Once obtained the Processing 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 Design, for each Data/Event Flow

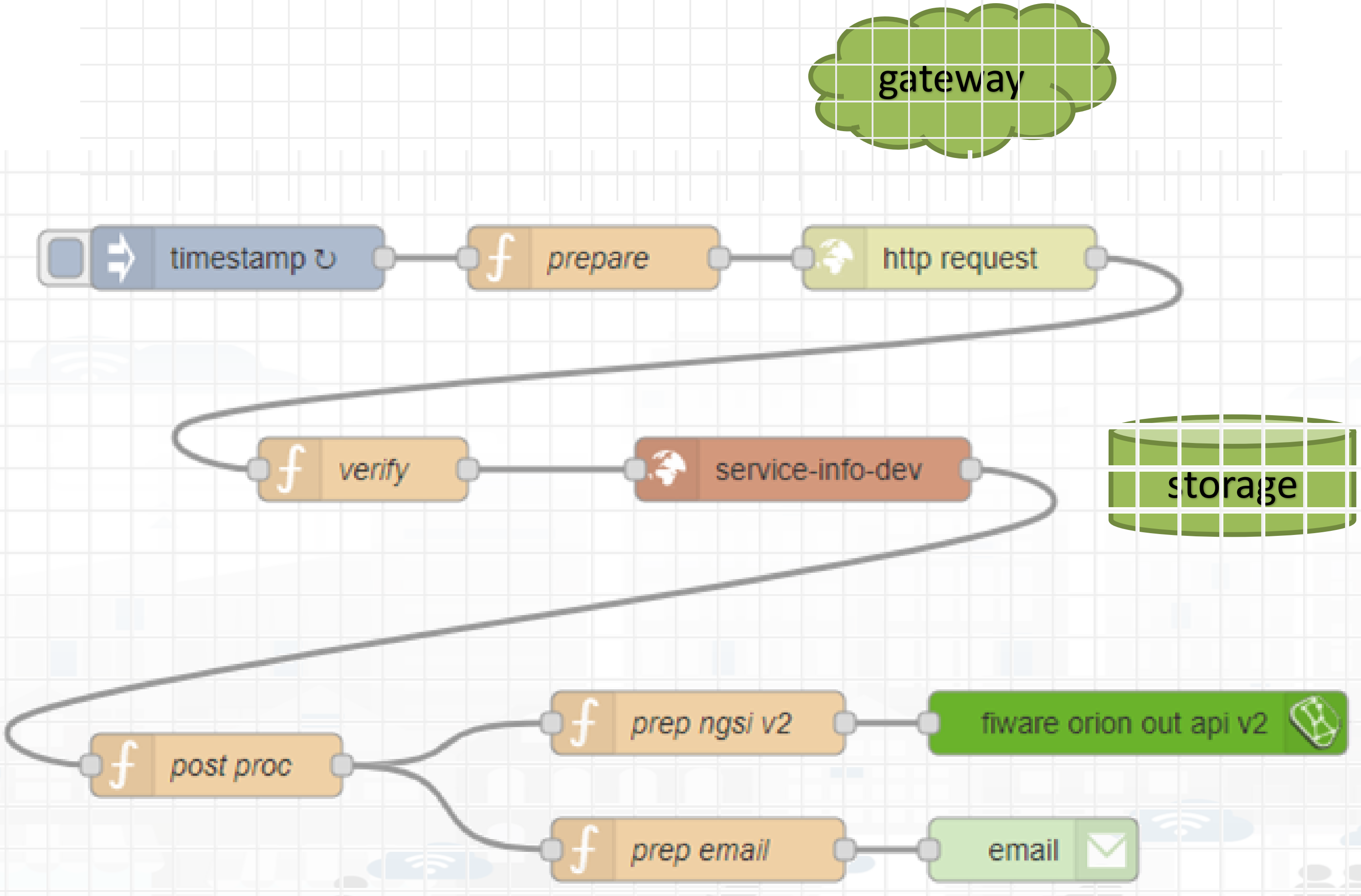


a. Periodically activate the flow
b. Call a gateway to get data
c. Verify the correctness of data
d. Enrich the data with other information coming from Cloud data into the storage
e. Transform the data in the correct forma
f. Send the data into the IoT Broker, and thus send the data in the storage on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

TOP

Develop: Data Processes

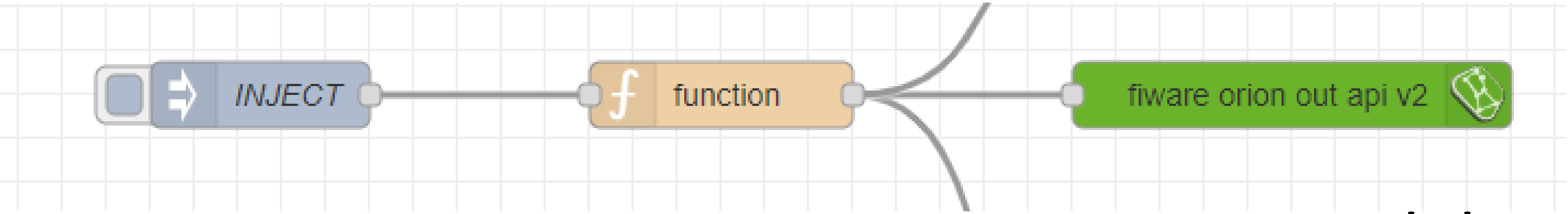
Proc.Logic (IoT App) Design, for each Data/Event Flow



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

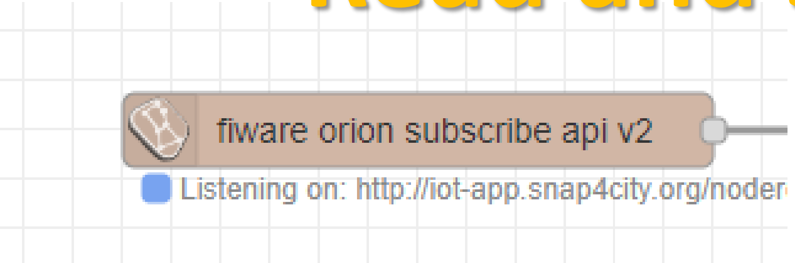


Function, example of NGSI V2 payload:

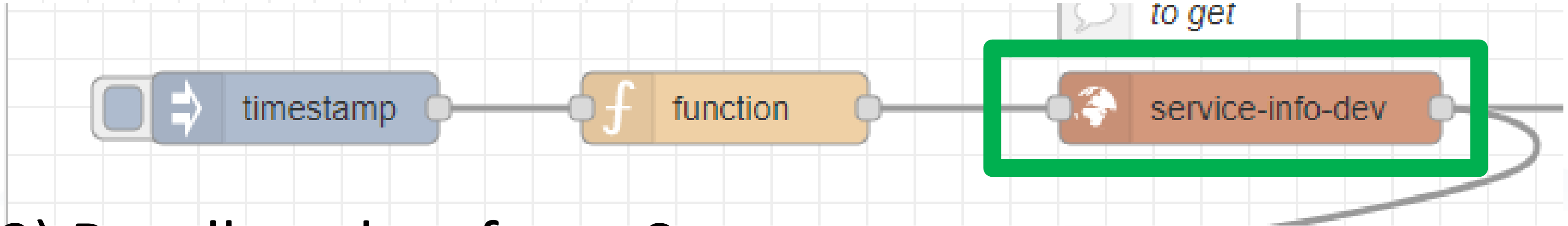
```
var time_now = new Date().toISOString();
var arandvalue = Math.random()
msg.payload =
        {"id":"mydev",
        "type":"mydevSensor",
        "anID":{"type": "integer", "value": "http://www.disit.org/km4city/resource/iot/............./anuser"},
        "VDDValue":{"type":"float","value":arandvalue},
        "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


fiware orion subscribe api v2
Listening on: http://iot-app.snap4city.org/noder

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

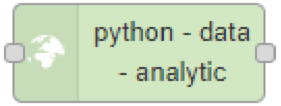


2) Recollect data from Storage

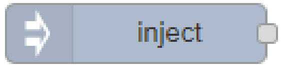
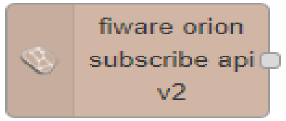
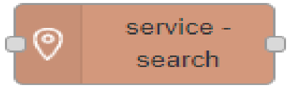
- This node uses the Smart City API

- 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)

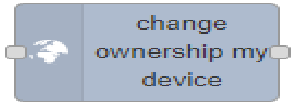
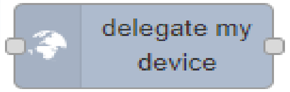
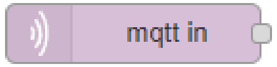
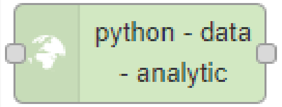
- Please note that the most important blocks nodes to interact with the platforms are reported in this table to familiarize with the main concepts. ***They are actually families of blocks/nodes*** since many others are present that allow you to perform a very large number of other features.
- YOU DO NOT 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.
- Most of the nodes can be also configured once from their user settings but the JSON is primary mode for setting parameters.


python - data - analytic

examples

Node shape	Description	Snap4City or standard
 inject	To generate injection messages into a flow, scheduled or on manual demand by click it on left.	standard
function	A java script function, from a JSON input to one or more JSON outputs, which can be produced by setting it.	standard
fiware orion out api v2	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.	Snap4city
 fiware orion subscribe api v2	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.	Snap4city
service info dev	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.	Snap4city
 service - search	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.	Snap4city
email	Send email. With other nodes you can send Telegram, SMS, etc.	standard
http request	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.	standard

examples

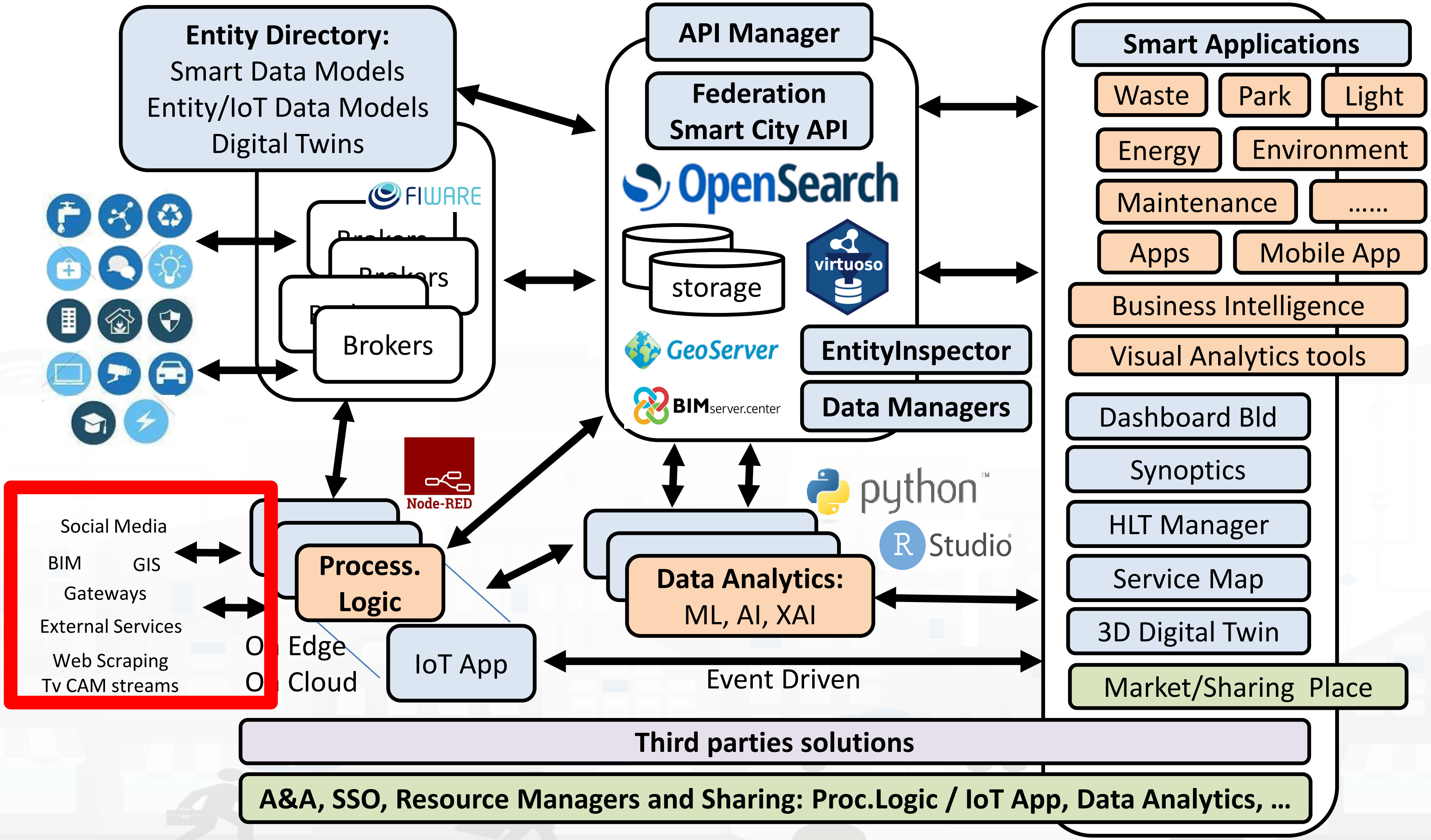
Node shape	Description	Snap4City or standard
	A block which is printing on debug view the data JSON passed in its input. Please note that the node can be tuned to provide only msg.payload or the full JSON message.	standard
	To create a Entity Instance (device instance) from a model prepared on Entity 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 simple widget. A large set of dashboard nodes to send and retrieve data to/from dashboards. This specific Nodes allows to send on dashboard HTML formatted messages with some limitations. Full HTTP widget is also accessible.	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
	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

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.

Tech Arch - 2023



Entity Directory:
Smart Data Models
Entity/IoT Data Models
Digital Twins

FIWARE

Brokers

API Manager

Federation
Smart City API

OpenSearch

storage

virtuoso

GeoServer

BIMserver.center

EntityInspector

Data Managers

Smart Applications

Waste

Park

Light

Energy

Environment

Maintenance

……

Apps

Mobile App

Business Intelligence

Visual Analytics tools

Dashboard Bld

Synoptics

HLT Manager

Service Map

3D Digital Twin

Market/Sharing Place

Social Media
BIM GIS
Gateways
External Services
Web Scraping
Tv CAM streams

Node-RED

Process. Logic

On Edge
On Cloud

IoT App

python

R Studio

Data Analytics:
ML, AI, XAI

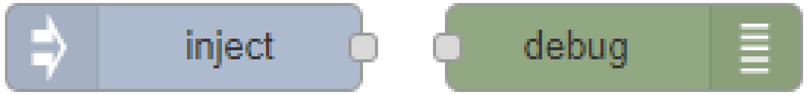
Event Driven

Third parties solutions

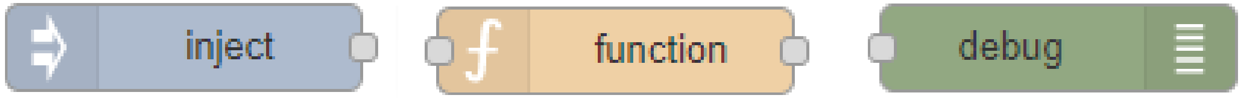
A&A, SSO, Resource Managers and Sharing: Proc.Logic / IoT App, Data Analytics, …

Some patterns

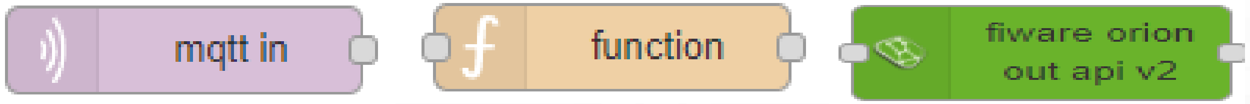
1) Hello world of node-red, the inject may provide a string to the debug.



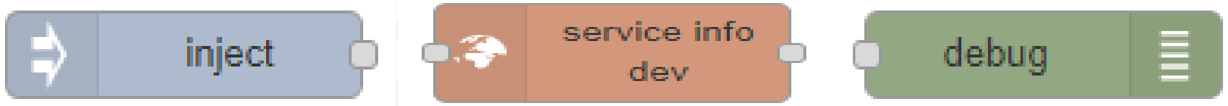
2) Hello world of node-red at two steps, the inject provides a push while a JSON is created into the function as *msg.payload = {…………}* and sent/shown to/by the debug.



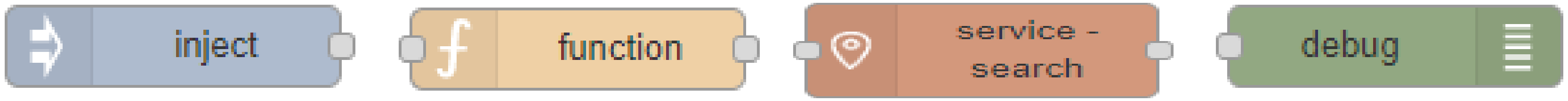
3) Event data reception from an MQTT broker, transformation and send it to the storage pushing data into the Orion Broker V2.



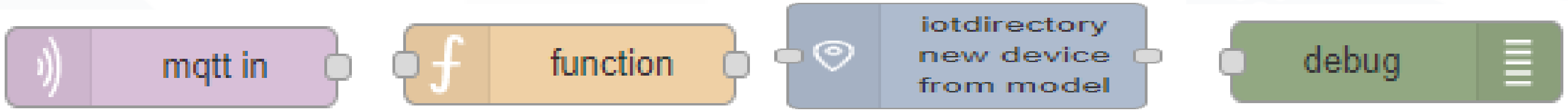
4) request on inject of a SURI to the storage to see data on debug.



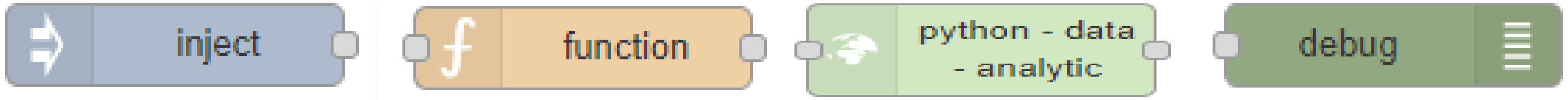
1) Preparation of data request on function, query to the storage and see data result on debug.



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):

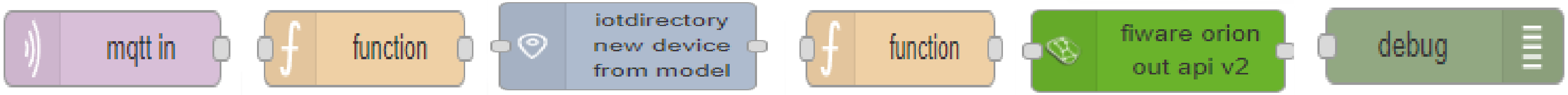


3) Preparation of data parameters on function, request computing Data Analytic, see data result on debug.

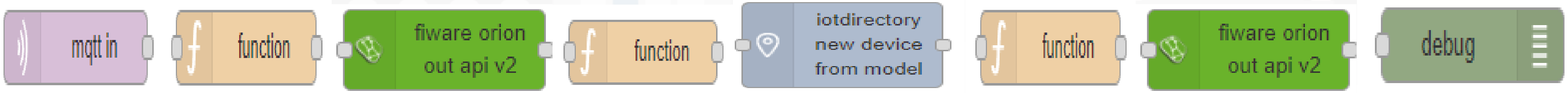


Typical strange patterns that may be not efficient in most cases:

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).



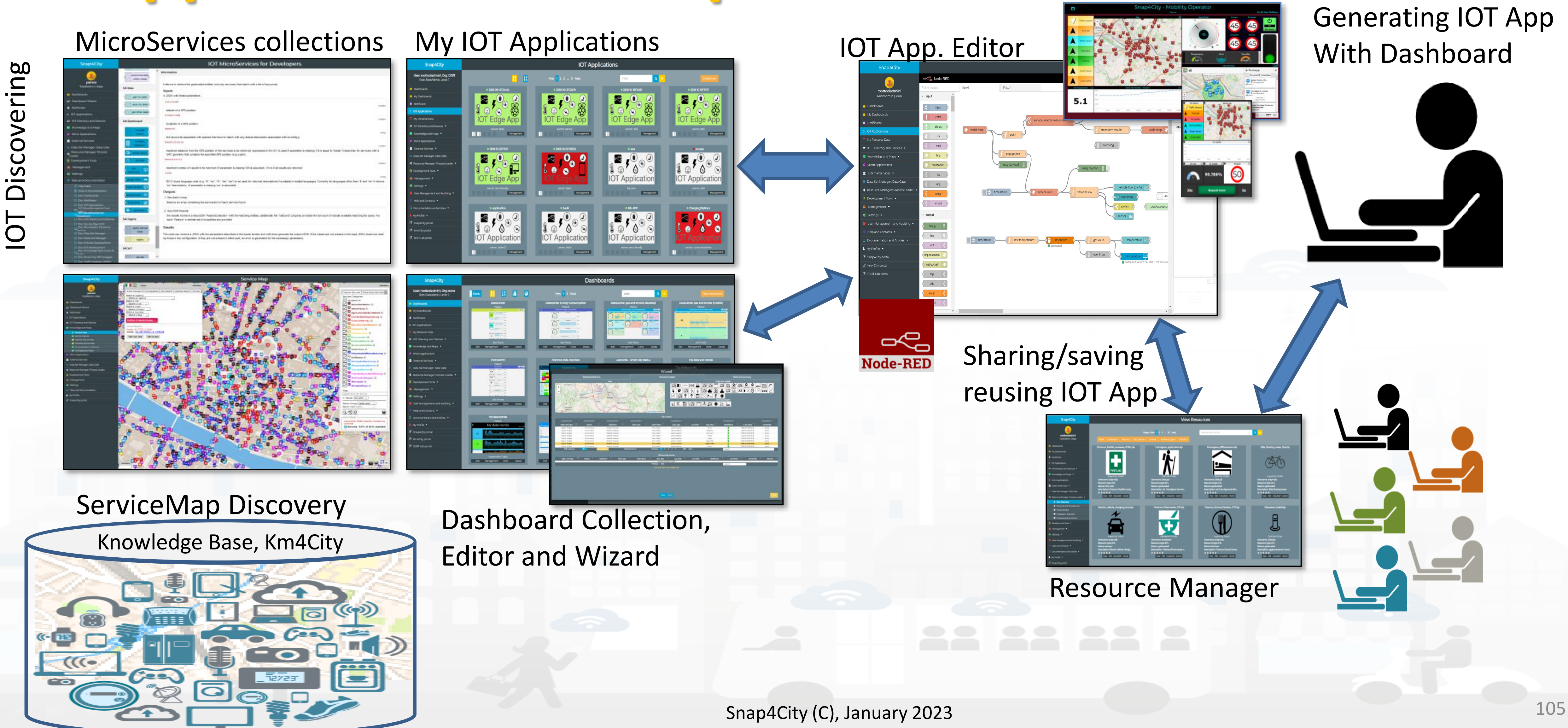
In most cases, it should be done the opposite: try to send the Entity Message, if it fails than create a new Entity Instance by known model, and if successful send again the Entity Message, or just wait for the new message to save it the first.

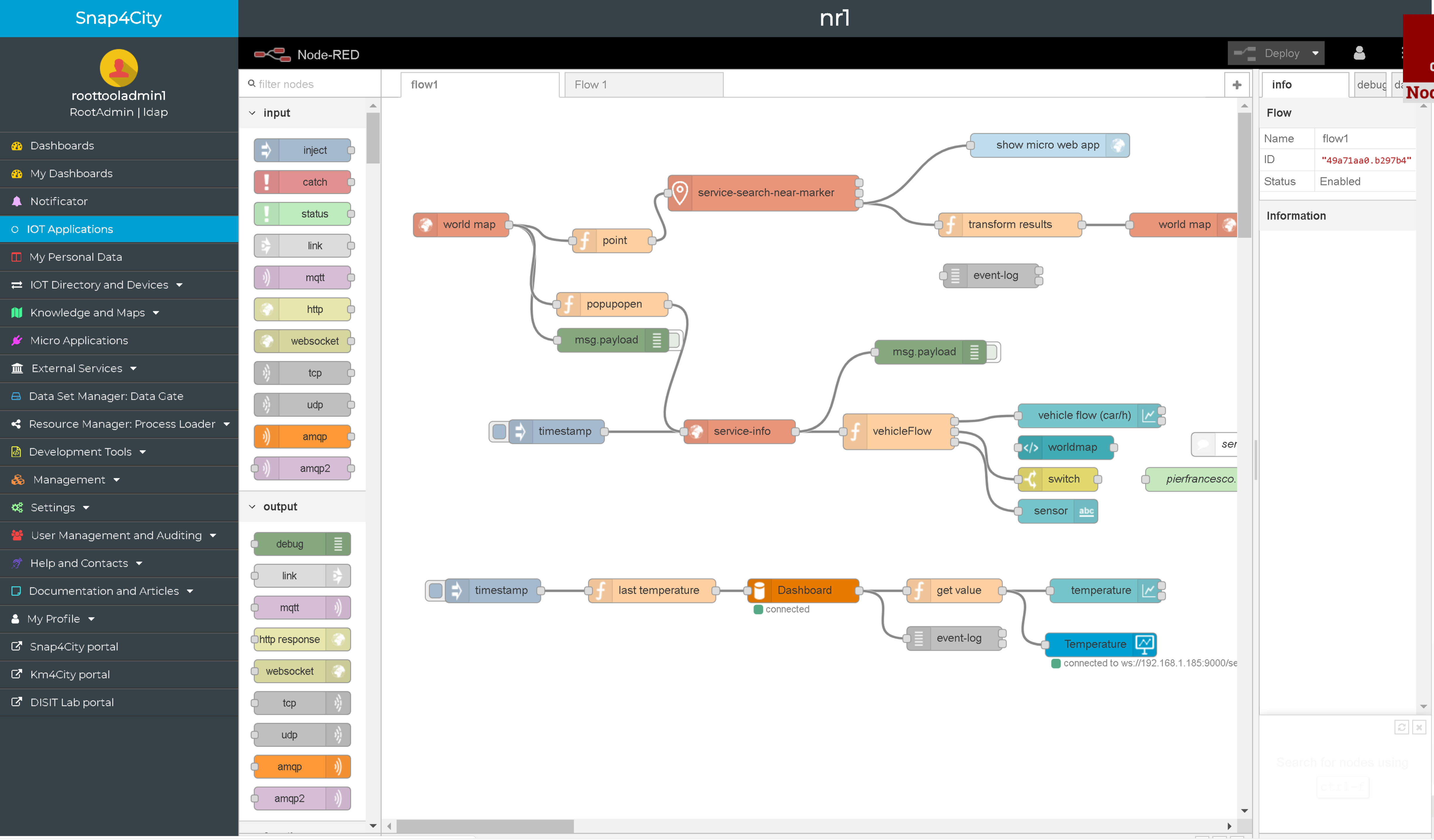


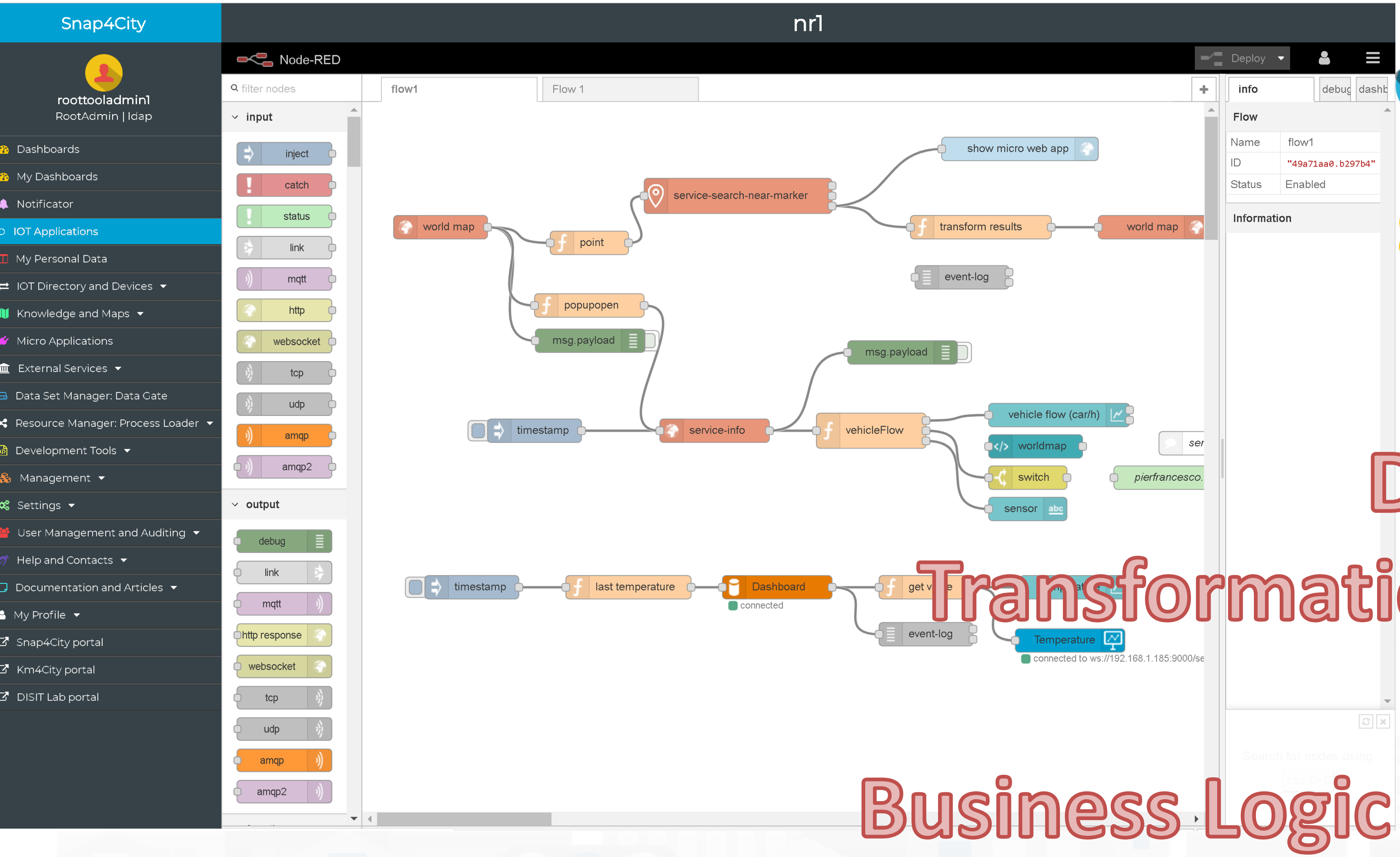
TOP

Processing Logic (IoT App) Development

IOT Applications Development




Snap4City
nr1
Node-RED
Deploy
roottooladmin1
RootAdmin | ldap
Dashboards
My Dashboards
Notificator
IOT Applications
My Personal Data
IOT Directory and Devices
Knowledge and Maps
Micro Applications
External Services
Data Set Manager: Data Gate
Resource Manager: Process Loader
Development Tools
Management
Settings
User Management and Auditing
Help and Contacts
Documentation and Articles
My Profile
Snap4City portal
Km4City portal
DISIT Lab portal
filter nodes
input
inject
catch
status
link
mqtt
http
websocket
tcp
udp
amqp
amqp2
output
debug
link
mqtt
http response
websocket
tcp
udp
amqp
amqp2
flow1
Flow 1
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
abc
ser
pierfrancesco.
timestamp
last temperature
Dashboard
connected
get value
temperature
event-log
Temperature
connected to ws://192.168.1.185:9000/se
info
debug
da
Flow
Name
flow1
ID
"49a71aa0.b297b4"
Status
Enabled
Information
Node-RED



Data Adapation

Transformation, Conversion

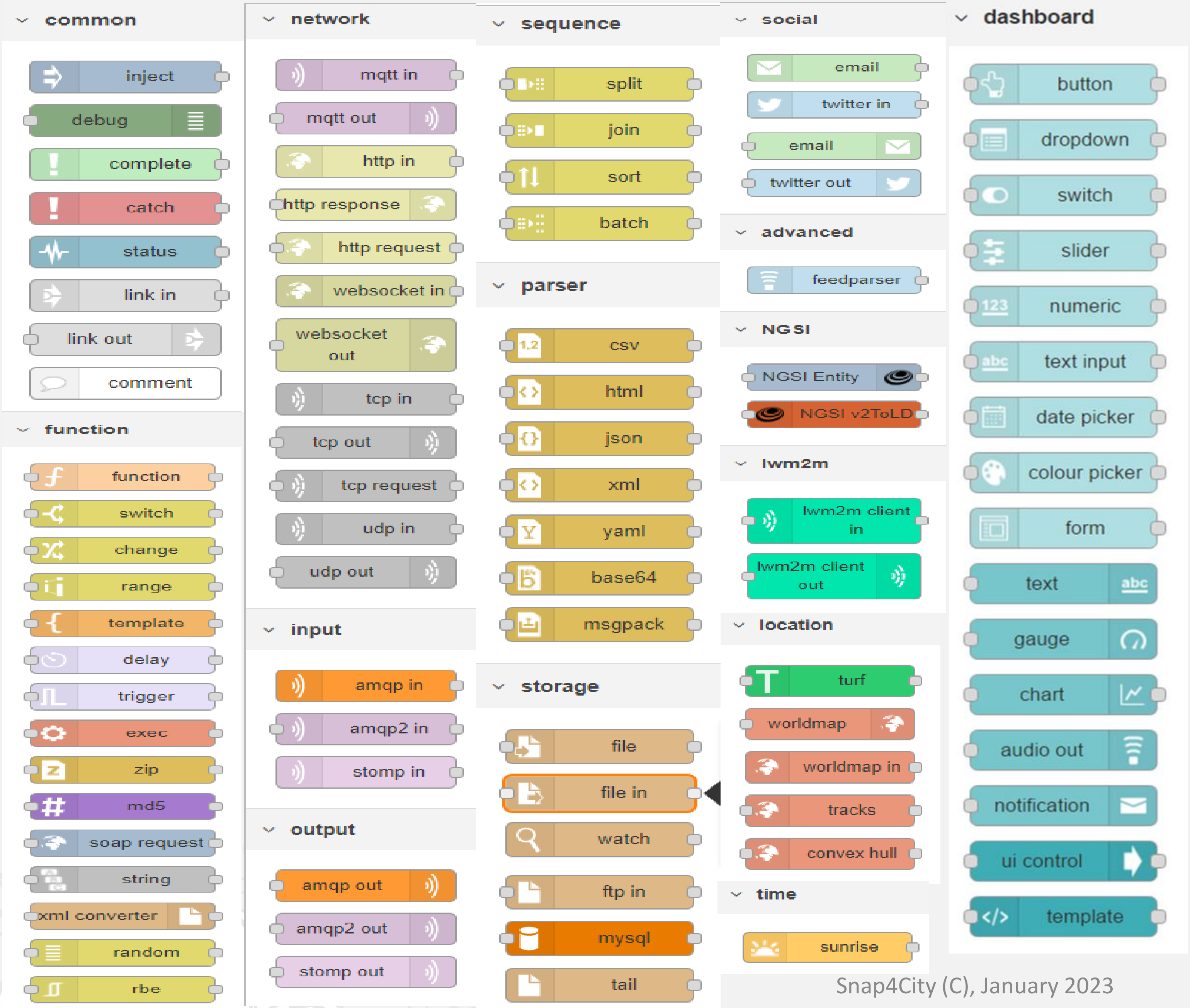
Integration

Business Logic vs Dashboards

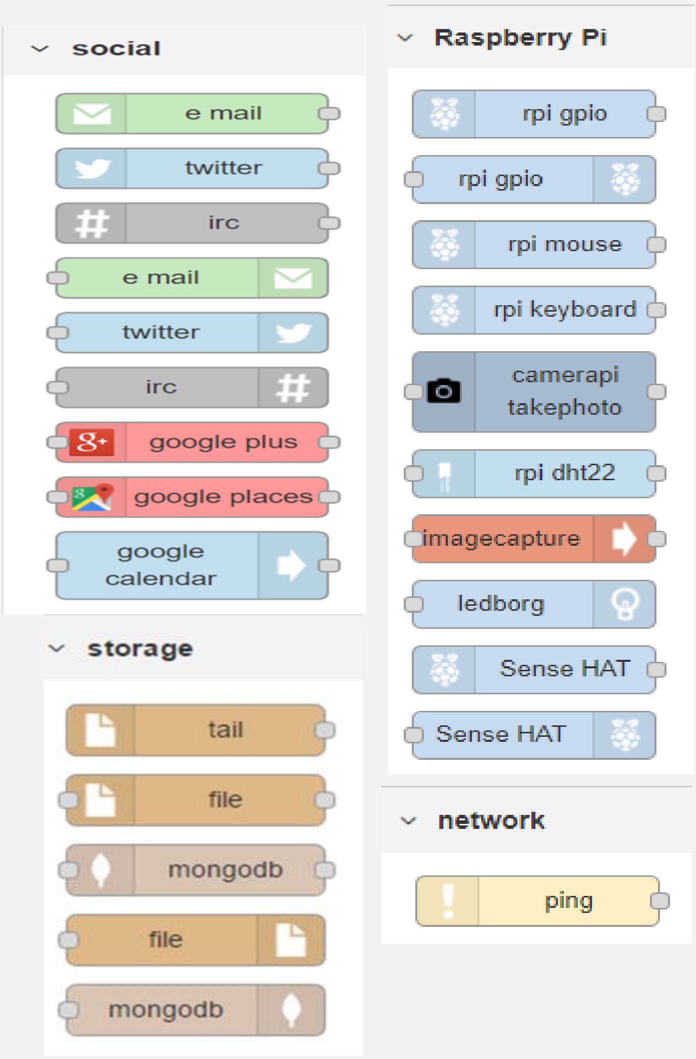
Data Analytics control

Everywhere: Cloud, on IoT Edge Devices

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



+ on IOT Edge Raspberry

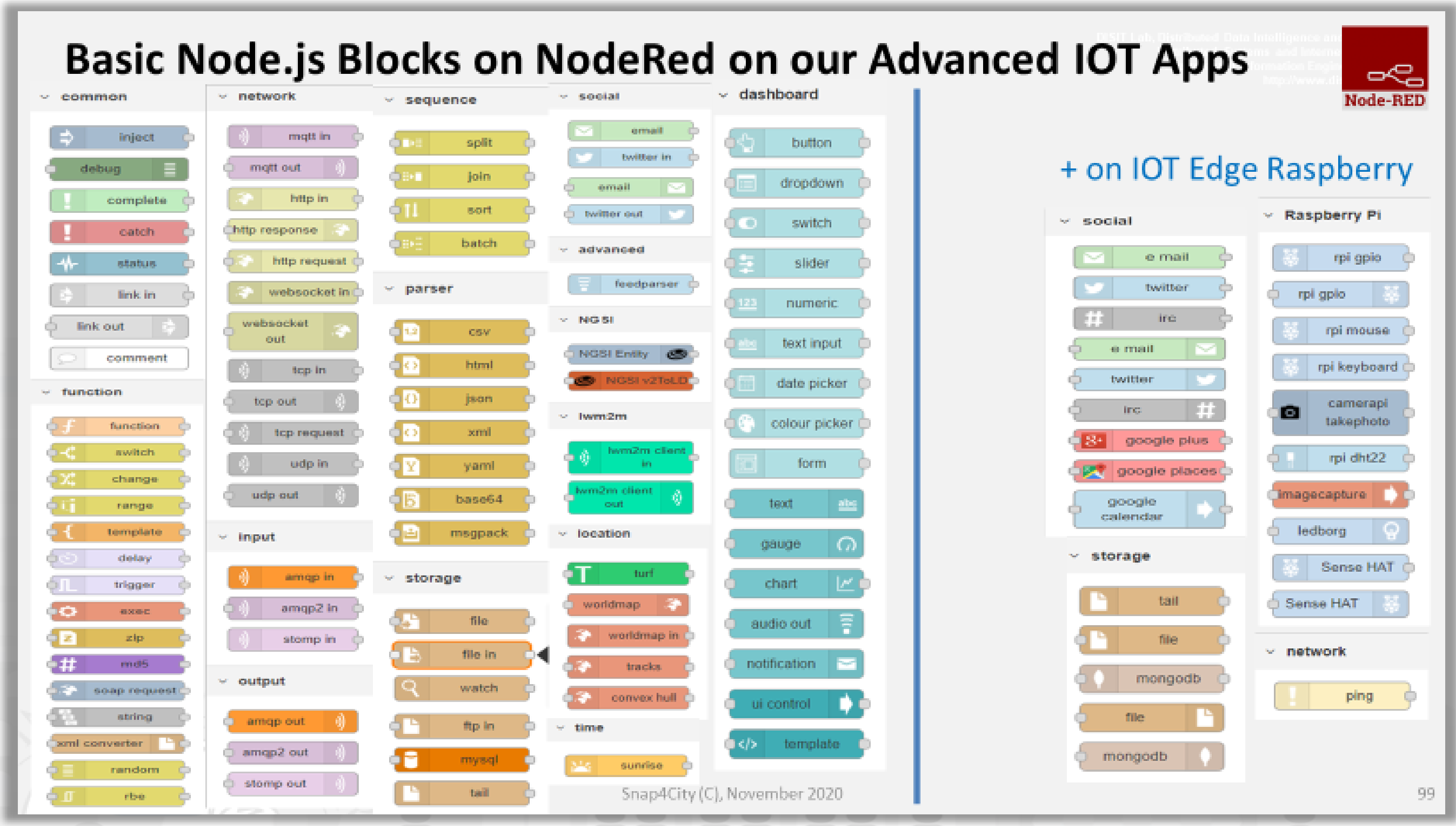


Node-RED Basic Blocks

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.



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

+ on IOT Edge Raspberry

Snap4City (C), November 2020

99

IoT Applications

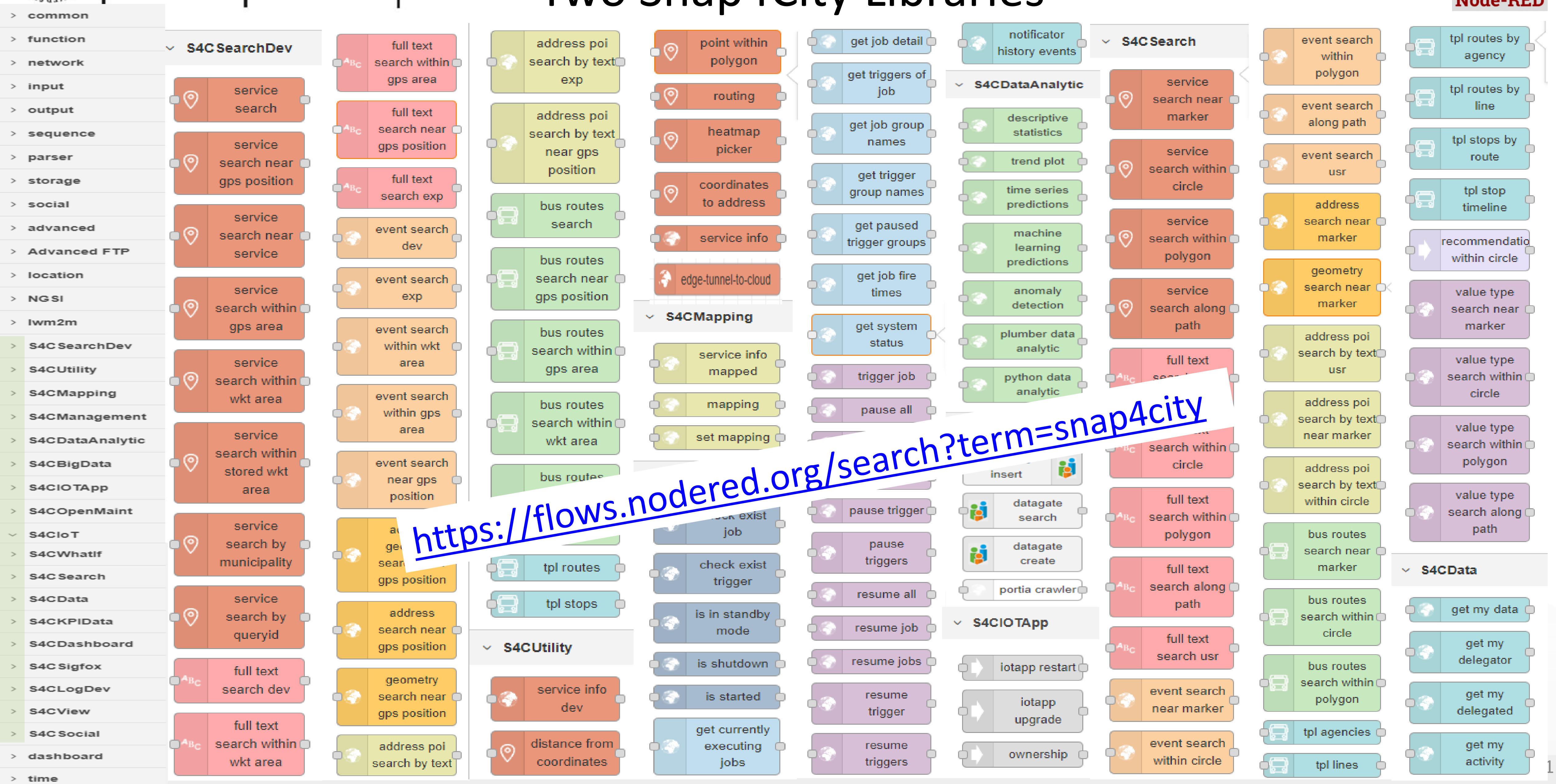
- **Data ingestion**: more than 70 protocols IOT and Industry 4.0, web Scraping, external services, any protocol database, etc.
- **Data access**: save/retrieve data, query search on expert system, georeverse solution, search on expert system Km4City ontology, etc.
- **Data Transformation/transcoding:** binary, hexadecimal, XML, JSON, String, any format
- **Integration**: CKAN, Web Scraping, FTP, Copernicus satellite, Twitter Vigilance, Workflow OpenMaint, Digital Twin BIMServer, 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.)
- **Custom Widgets**: SVG, synoptics, animations, dynamic pins on maps, etc
- **Event management**: Telegram, Twitter, Facebook, SMS, WhatsApp, CAP, etc.
- **Hardware Specific Devices**: Raspberry Pi, Android, Philips, video wall management, etc.

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

Sept 2022 collection

Two Snap4City Libraries

SNAP4CITY | Node-RED


common
function
network
input
output
sequence
parser
storage
social
advanced
Advanced FTP
location
NGSI
lwm2m
S4CSearchDev
S4CUtility
S4CMapping
S4CManagement
S4CDataAnalytic
S4CBigData
S4CIOTApp
S4COpenMaint
S4CIoT
S4CWhatIf
S4CSearch
S4CData
S4CKPIData
S4CDashboard
S4CSigfox
S4CLogDev
S4CView
S4CSocial
dashboard
time
S4CSearchDev
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
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
bus routes
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
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
S4CSearch
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

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

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

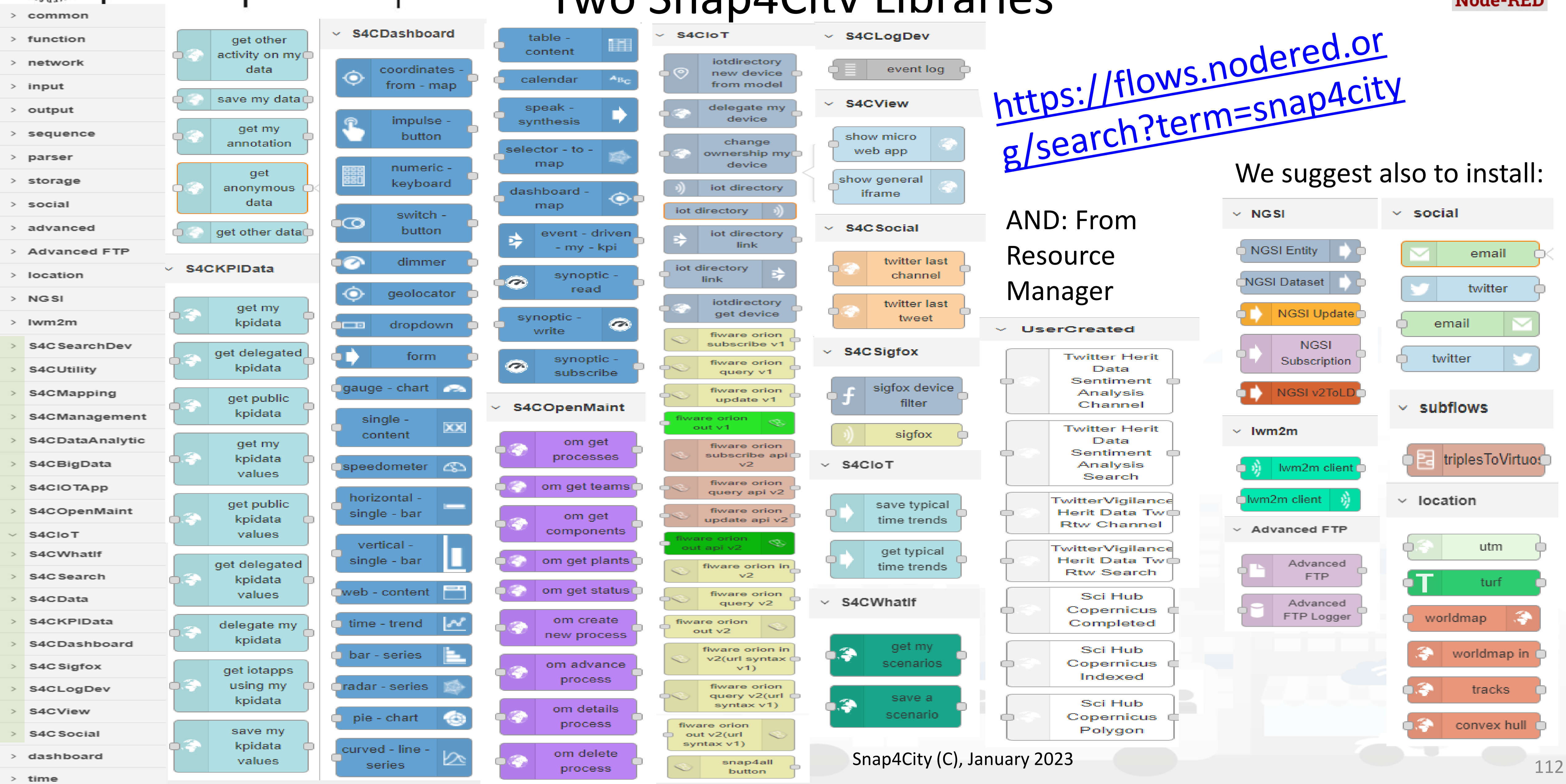
Sept 2022 collection
Two Snap4Citv Libraries

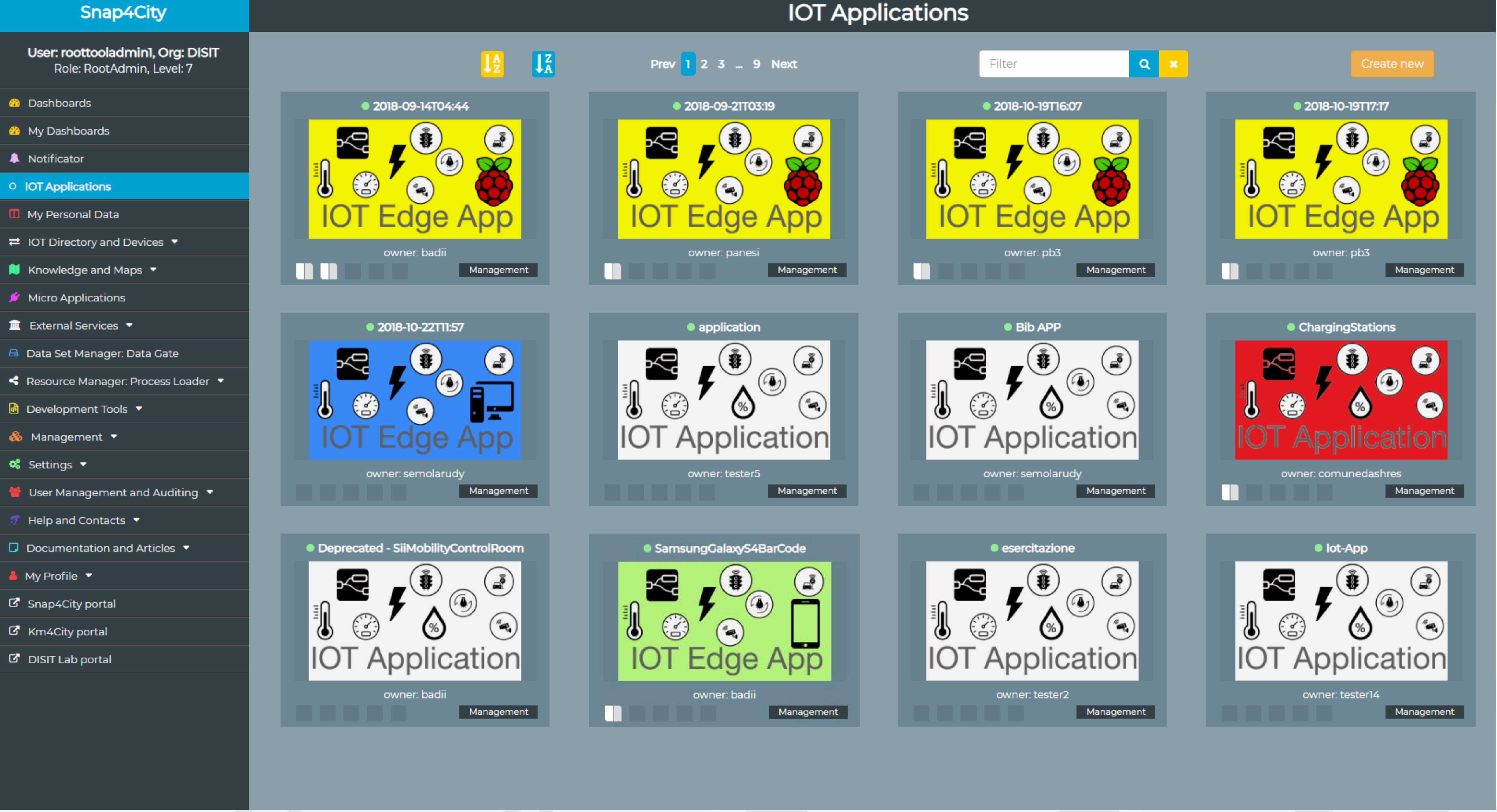
SNAP4CITY Node-RED

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

AND: From Resource Manager

We suggest also to install:





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

IOT Applications

2018-09-14T04:44
owner: badii

2018-09-21T03:19
owner: panesi

2018-10-19T16:07
owner: pb3

2018-10-19T17:17
owner: pb3

2018-10-22T11:57
owner: semolarudy

application
owner: tester5

Bib APP
owner: semolarudy

ChargingStations
owner: comunedashres

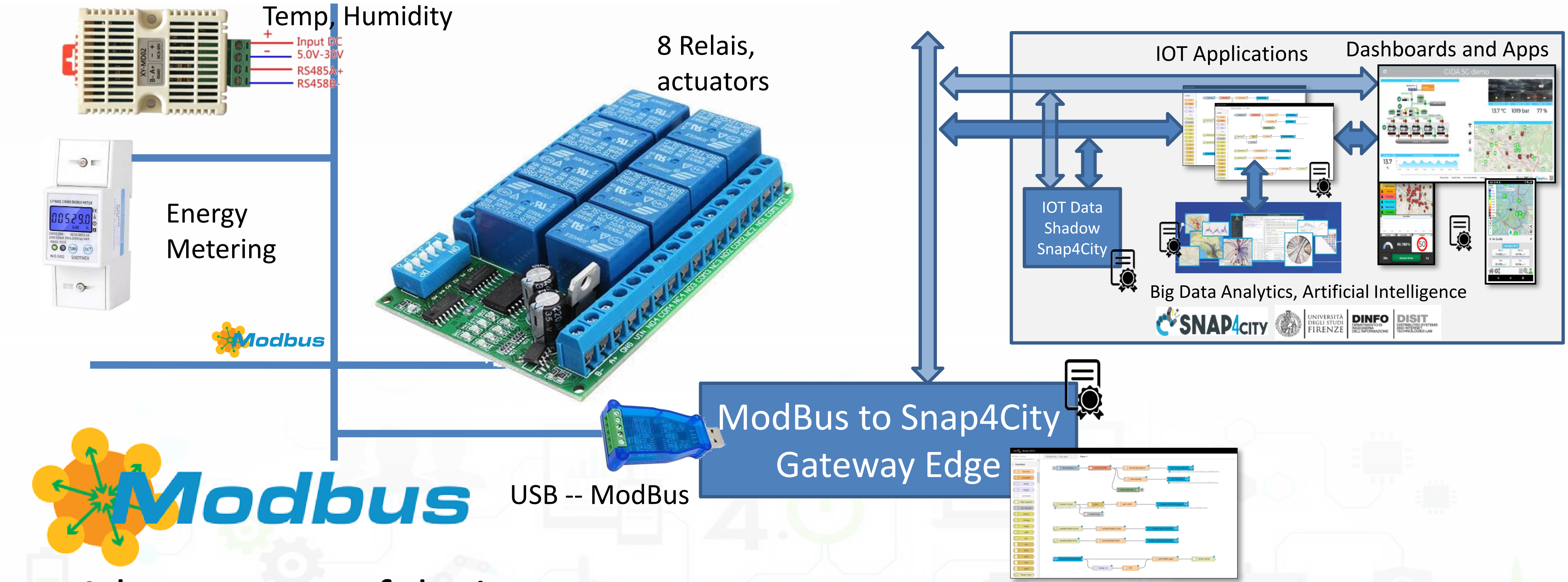
Deprecated - SiiMobilityControlRoom
owner: badii

SamsungGalaxyS4BarCode
owner: badii

esercitazione
owner: tester2

Iot-App
owner: tester14

Devices



Temp, Humidity

Input DC 5.0V-30V

RS485A+

RS458B-

8 Relais, actuators

Energy Metering

Modbus

USB -- ModBus

ModBus to Snap4City Gateway Edge

IOT Applications

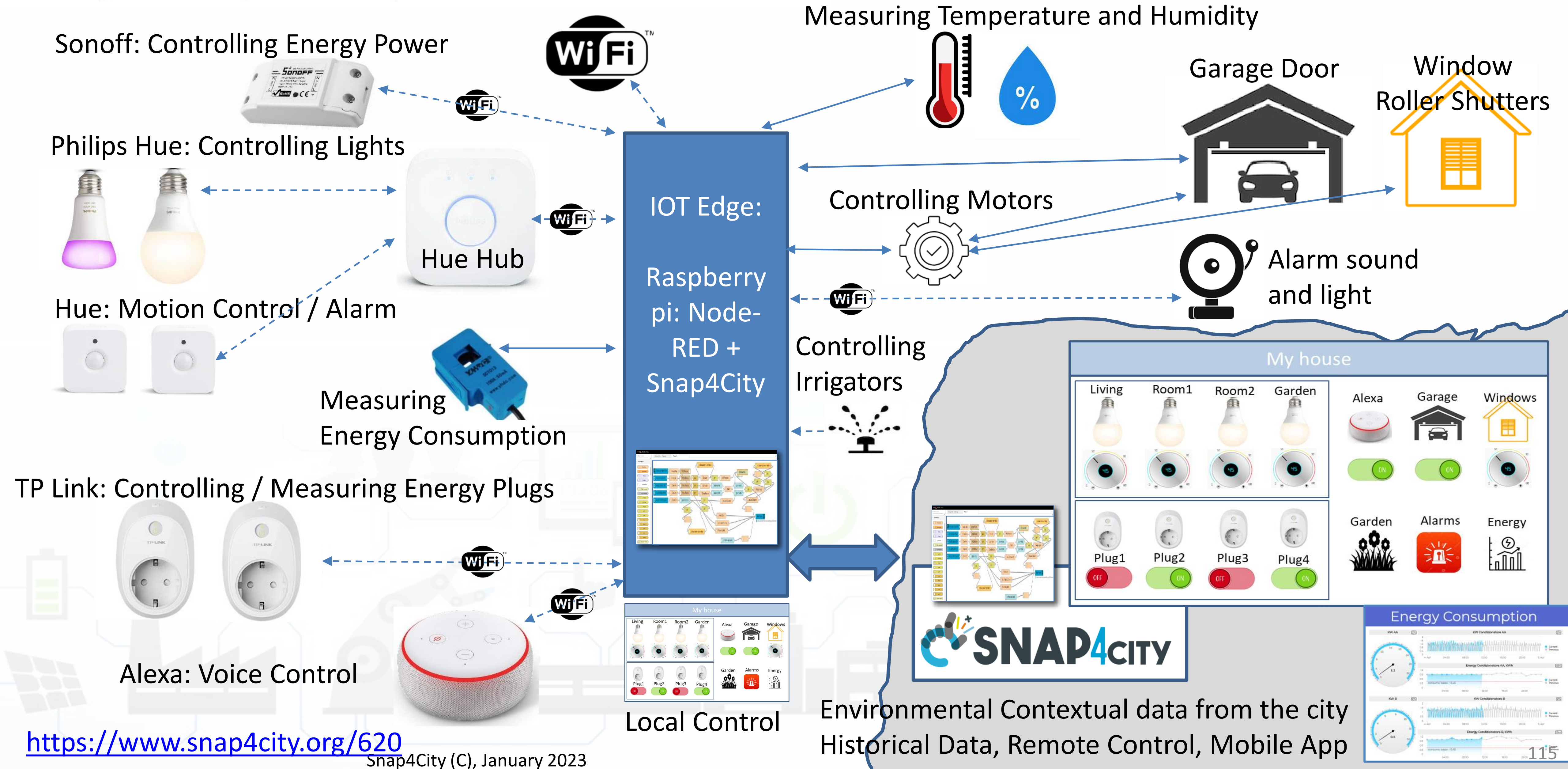
Dashboards and Apps

IOT Data Shadow Snap4City

Big Data Analytics, Artificial Intelligence

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

Snap4Home


Sonoff: Controlling Energy Power
Wi Fi
Philips Hue: Controlling Lights
Hue Hub
Hue: Motion Control / Alarm
Measuring
Energy Consumption
TP Link: Controlling / Measuring Energy Plugs
Alexa: Voice Control
IOT Edge:
Raspberry pi: Node-RED + Snap4City
Local Control
Measuring Temperature and Humidity
%
Garage Door
Window
Roller Shutters
Controlling Motors
Alarm sound and light
Controlling Irrigators
My house
Living
Room1
Room2
Garden
Alexa
Garage
Windows
Plug1
Plug2
Plug3
Plug4
OFF
ON
Alarms
Energy
SNAP4CITY
Energy Consumption
Environmental Contextual data from the city
Historical Data, Remote Control, Mobile App

https://www.snap4city.org/620

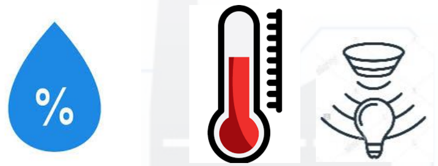
Snap4Home

Hue Hub

Motion Control / Alarm

TP Link plugs: meter

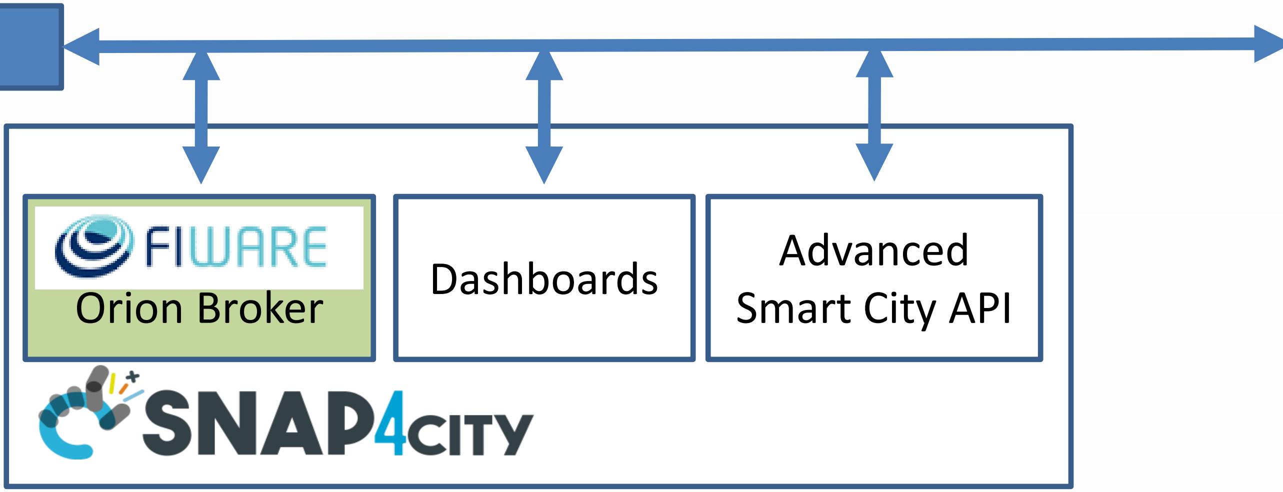
Alexa: Voice Control



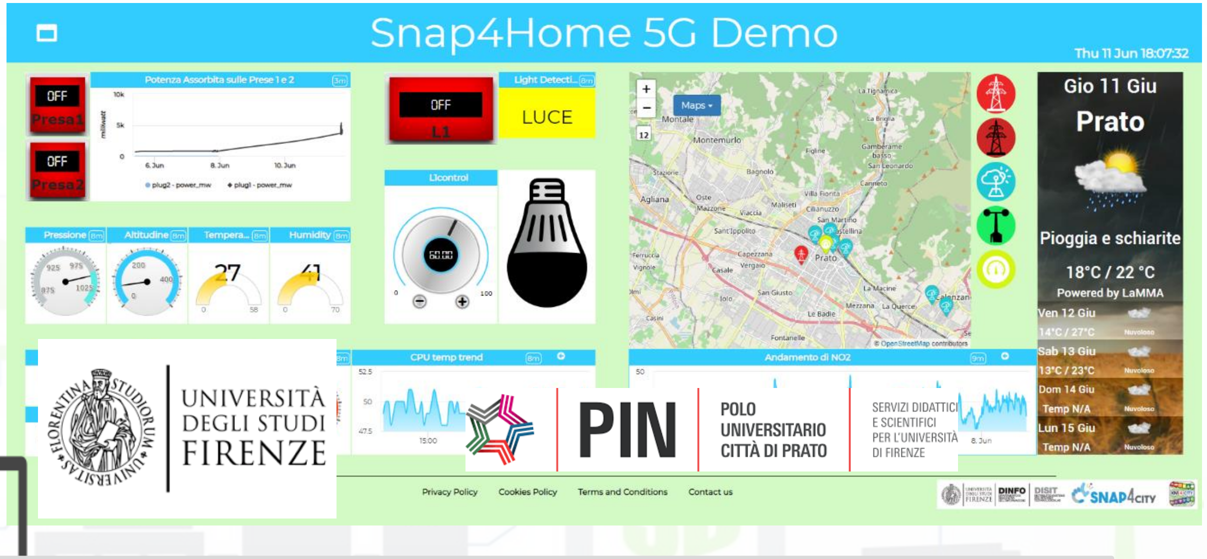
IOT Edge:

Raspberry pi: Node-RED + Snap4City MicroService Library

5G gateway



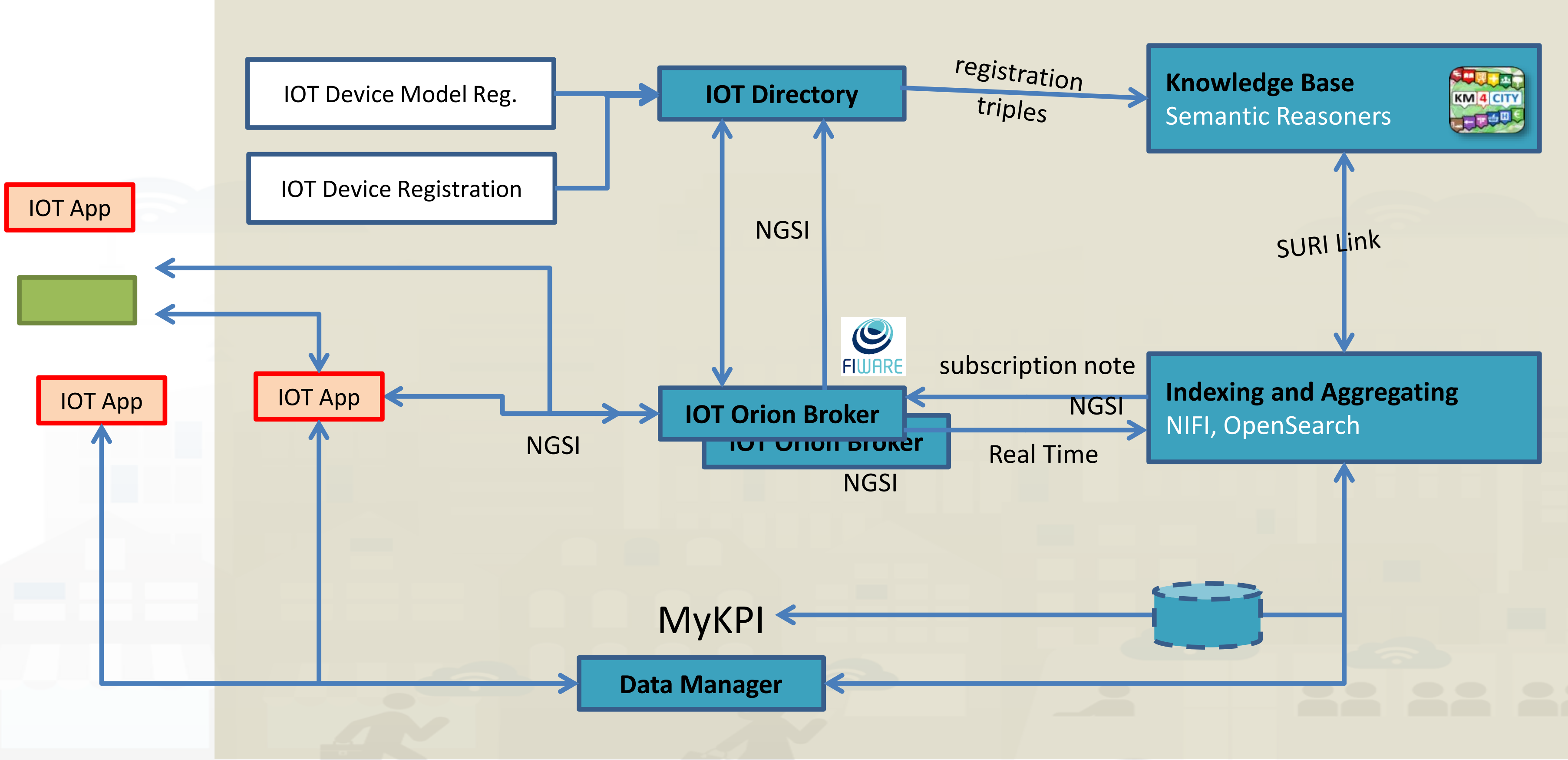
Environmental Contextual data from the city. Historical Data, Remote Control, Mobile App



open fiber WIND 3 ZTE estra

https://www.snap4city.org/369

Main IoT Data In/Out flows



IOT Device Model Reg.

IOT Device Registration

IOT Directory

registration triples

Knowledge Base
Semantic Reasoners

NGSI

SURI Link

IOT App

IOT App

IOT App

FIWARE

subscription note

NGSI

IOT Orion Broker

IOT Orion Broker

Real Time

NGSI

NGSI

Indexing and Aggregating
NIFI, OpenSearch

MyKPI

Data Manager

Checking data ingestion results

Knowledge base
Semantic reasoners

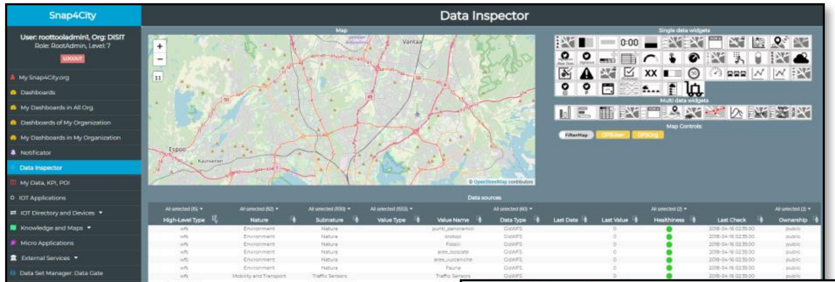
- All searches
- Metata
- Structure
- Last values of IOT Dev
- GTFS
- Only public IOT Dev

- **Data Inspector**
- **ServiceMap**, SCAPI
  - LOG / LOD viewer
  - Super Service Map
- IOT Directory
- SCAPI: Swagger
- IOT Broker

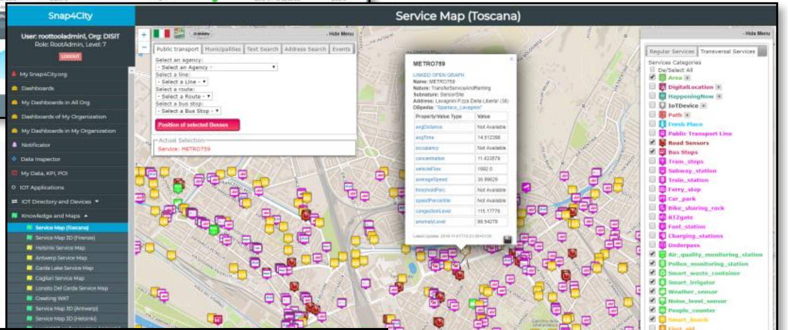
Indexing and aggregating
NIFI, OpenSearch

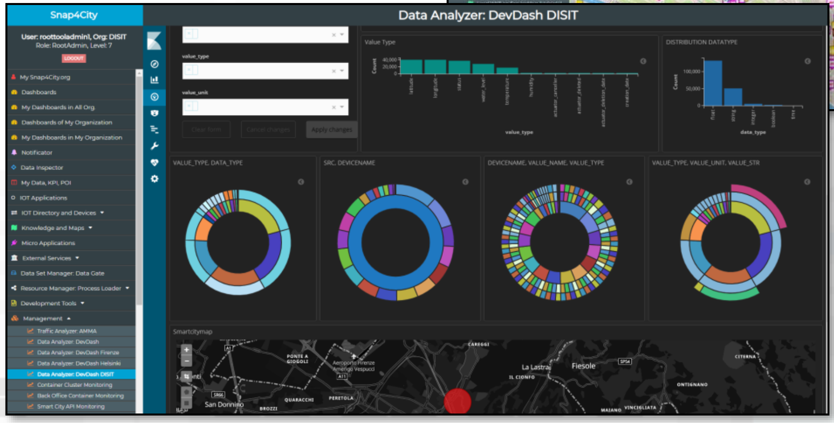
- Faceted search
- Geo search
- Time Series
- Private and Public

- **Data Inspector**
- **ServiceMap**, SCAPI
- **My Data Dashboard** (Kibana), DevDash
- OpenDistro x Elastic Search



Data Inspector
Digital Twin view



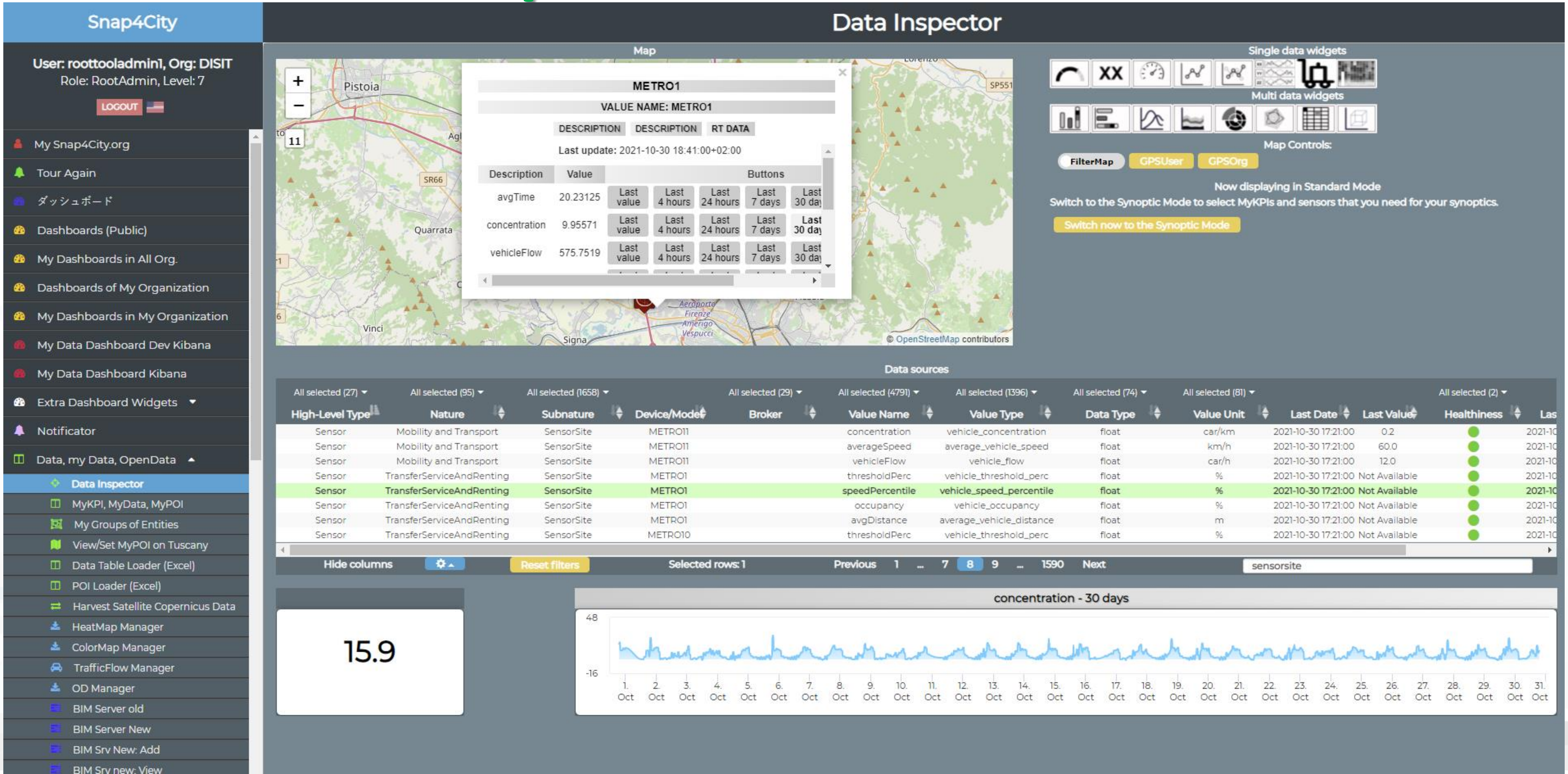


ServiceMap or
Super ServiceMap

My Data Dashboard
DevDash

Some functionalities are limited to certain roles

Data Inspector: HLT classification



Snap4City

Data Inspector

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

LOGOUT

- My Snap4City.org
- Tour Again
- 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, my Data, OpenData
  - 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
  - HeatMap Manager
  - ColorMap Manager
  - TrafficFlow Manager
  - OD Manager
  - BIM Server old
  - BIM Server New
  - BIM Srv New: Add
  - BIM Srv new: View

Map

METRO1

VALUE NAME: METRO1

DESCRIPTION | DESCRIPTION | RT DATA

Last update: 2021-10-30 18:41:00+02:00

Description	Value	Buttons				
avgTime	20.23125	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 day
concentration	9.95571	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 day
vehicleFlow	575.7519	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 day

Pistoia
Quarrata
Vinci
Signa
Aeroporto Firenze Amerigo Vespucci
SR66
SP551
11
© OpenStreetMap contributors

Single data widgets

Multi data widgets

Map Controls:

FilterMap | GPSUser | GPSOrg

Now displaying in Standard Mode

Switch to the Synoptic Mode to select MyKPIs and sensors that you need for your synoptics.

Switch now to the Synoptic Mode

Data sources

High-Level Type	Nature	Subnature	Device/Model	Broker	Value Name	Value Type	Data Type	Value Unit	Last Date	Last Value	Healthiness	Las
All selected (27)	All selected (95)	All selected (1658)		All selected (29)	All selected (4791)	All selected (1396)	All selected (74)	All selected (81)			All selected (2)	
Sensor	Mobility and Transport	SensorSite	METRO11		concentration	vehicle_concentration	float	car/km	2021-10-30 17:21:00	0.2	●	2021-10
Sensor	Mobility and Transport	SensorSite	METRO11		averageSpeed	average_vehicle_speed	float	km/h	2021-10-30 17:21:00	60.0	●	2021-10
Sensor	Mobility and Transport	SensorSite	METRO11		vehicleFlow	vehicle_flow	float	car/h	2021-10-30 17:21:00	12.0	●	2021-10
Sensor	TransferServiceAndRenting	SensorSite	METRO1		thresholdPerc	vehicle_threshold_perc	float	%	2021-10-30 17:21:00	Not Available	●	2021-10
Sensor	TransferServiceAndRenting	SensorSite	METRO1		speedPercentile	vehicle_speed_percentile	float	%	2021-10-30 17:21:00	Not Available	●	2021-10
Sensor	TransferServiceAndRenting	SensorSite	METRO1		occupancy	vehicle_occupancy	float	%	2021-10-30 17:21:00	Not Available	●	2021-10
Sensor	TransferServiceAndRenting	SensorSite	METRO1		avgDistance	average_vehicle_distance	float	m	2021-10-30 17:21:00	Not Available	●	2021-10
Sensor	TransferServiceAndRenting	SensorSite	METRO10		thresholdPerc	vehicle_threshold_perc	float	%	2021-10-30 17:21:00	Not Available	●	2021-10

Hide columns | Reset filters | Selected rows: 1 | Previous 1 … 7 8 9 … 1590 Next | sensorsite

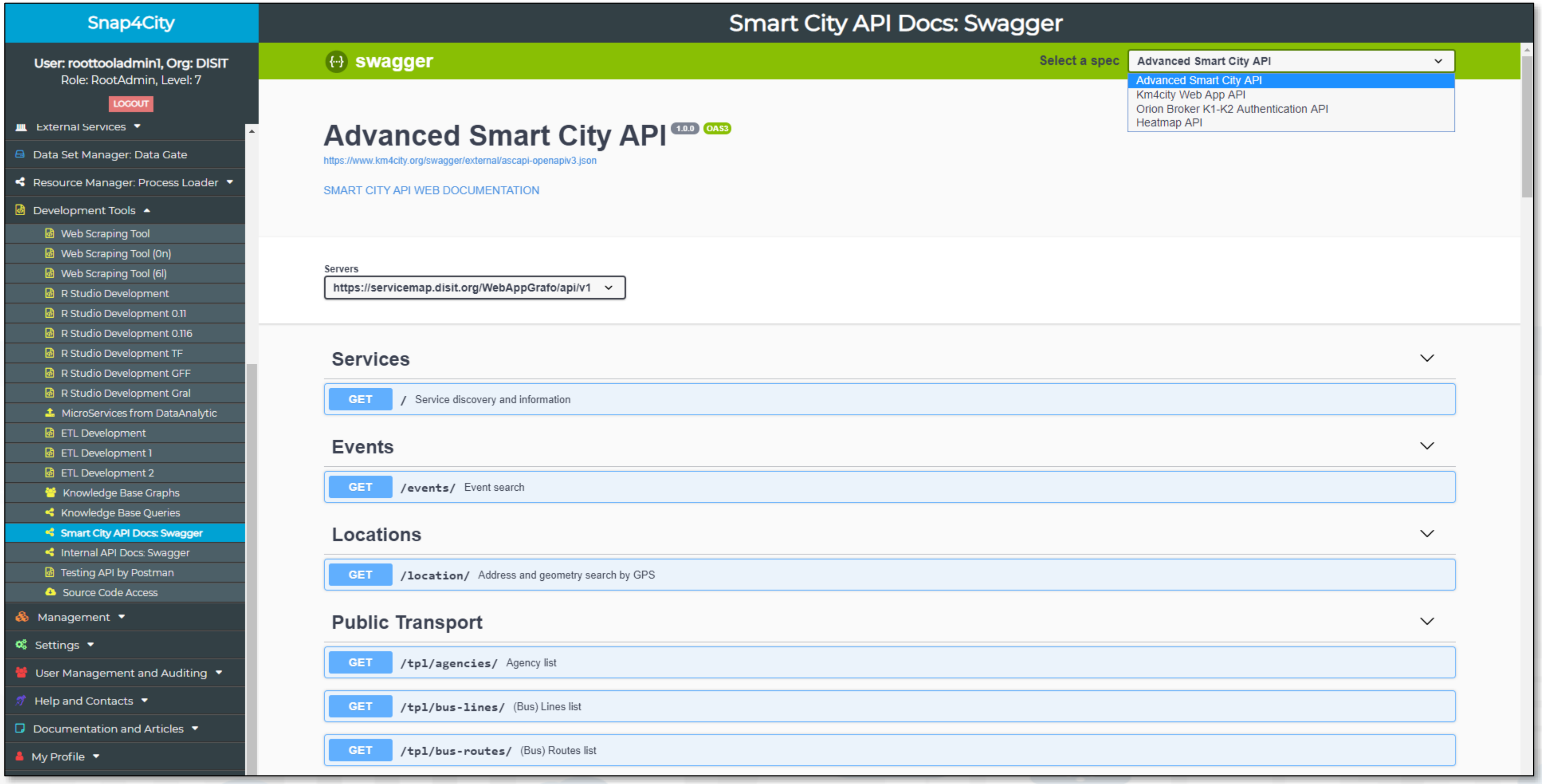
15.9

concentration - 30 days

TOP

Exploit Smart City API

External Smart City API



Smart City API Docs: Swagger

Advanced Smart City API 1.0.0 OAS3

https://www.km4city.org/swagger/external/ascapi-openapiv3.json

SMART CITY API WEB DOCUMENTATION

Servers

https://servicemap.disit.org/WebAppGrafo/api/v1

Services

GET `/` Service discovery and information

Events

GET `/events/` Event search

Locations

GET `/location/` Address and geometry search by GPS

Public Transport

GET `/tpl/agencies/` Agency list

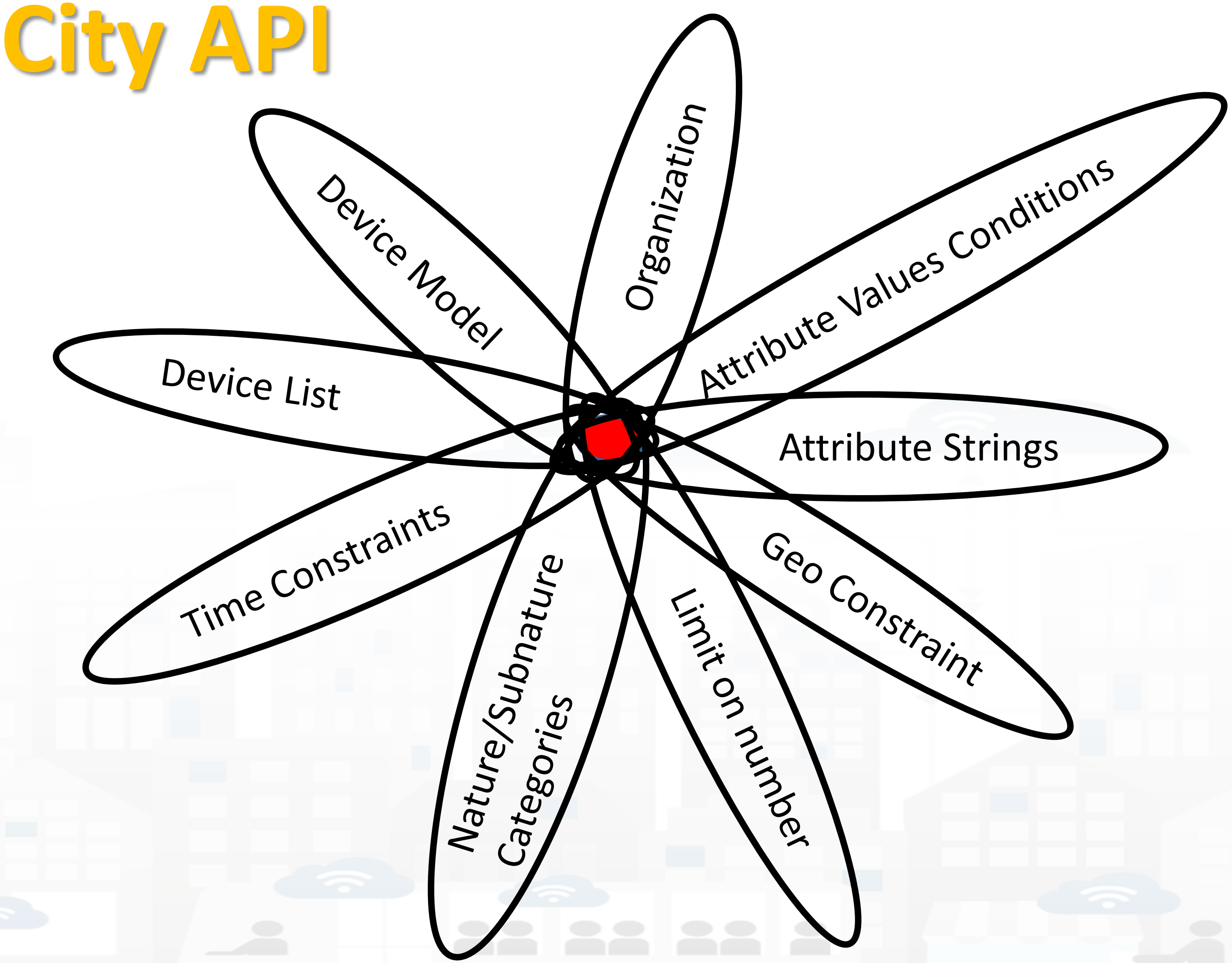
GET `/tpl/bus-lines/` (Bus) Lines list

GET `/tpl/bus-routes/` (Bus) Routes list

https://www.km4city.org/swagger/external/index.html

Selection on Smart City API

- Combining different filters for selecting entities from Smart City APIs
- ***Be care***: filtering too much may lead to empty set ☺



How to Get the «Query» used in More Options (2a)

- REST CALL by category → JSON (Options in RED), they are REST ASCAPI calls
  - Requesting a category, so that to see all Services of the same category (subNature)
    - http://svealand.snap4city.org/ServiceMap/api/v1/?selection=59.5814585785379555;16.71183586120606;59.62875017053684;16.875171661376957&categories=Street_light&maxResults=100&format=json
      - Please note that in the MoreOption dashboard the GPS area is neglected
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.64471;11.005751;43.89471;11.505751&categories=Green_areas&maxResults=200&format=json
      - Please note that in the MoreOption dashboard the GPS area is neglected
    - Custom PINS note: “selection” coordinates are used for collecting attributes in custom PINS. Other options such as “maxDists” cannot be used in custom PIN. All parameters can be used in other cases.
    - Different KB links are identified by their ASCAPI links: svealand.snap4city.org, servicemap.disit.org, ….
  - **Requests to SuperServiceMap for the network of Federated KBs** by using /api/……….. Without prefixed KB to obtain merged results from more KBs. For example as:
    - /api/v1/?categories=Air_quality_monitoring_station&format=json
    - Please note that the direct links to the superservicemap can be of the form:
      - https://www.disit.org/superservicemap/api/v1/? ……………………

How to Get the «Query» used in More Options (2b)

- **REST CALL by ServiceURI → JSON (ServiceURI in RED), they are ASCAPI calls**
  - **Requesting single Service**
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/ARPAT_QA_FI-BOBOLI&format=json
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/ARPAT_QA_FI-MOSSE_SV&format=json
    - Different KBs links are identified by their ASCAPI links: svealand.snap4city.org, servicemap.disit.org,
  - **Requesting all IoT Devices that have been produced by the same Model**
    - **https://www.disit.org/superservicemap/api/v1?selection=59.36535064975547;13.457822799682619;59.3903147426085213.566999435424806&model=SmartLightCapelon**&format=json
      - Please note that in this case the call is performed on the superservicemap, you can change to go directly on the right KB
      - You can specific both category and model to be more precise and focused.
    - **https://www.disit.org/superservicemap/api/v1/?selection=36.8092847020594;12.216796875000002;42.71473218539458;32.03613281250001&categories=Travel_information&format=json&fullCount=false&maxResults=500&model=DOMESTICMOVEMENTS2013-2018_1620304406**
      - In this case, we have a double filtering for model and for categories, plus other constraints
      - Please note that in the MoreOption dashboard the GPS area is neglected

How to Get the «Query» used in More Options (2c)

- **Requesting get data single device (view on map, if format HTML and not JSON)**

  Request to see the single device:

  - **https://svealand.snap4city.org/ServiceMap/api/v1/?serviceUri=http://www.disit.org/km4city/resource/iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFE894AF7&format=json&fromTime=3-day**
  - **With ServerURI: http://www.disit.org/km4city/resource/iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFE894AF7**
  - **From KB: https://svealand.snap4city.org**

https://svealand.snap4city.org/ServiceMap/api/v1/?serviceUri=http://www.disit.org/km4city/resource/iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFE894AF7&format=json&fromTime=3-day

```
{ "Service":
{"type": "FeatureCollection",
"features": [
        {
         "geometry": {    "type": "Point",    "coordinates": [ 13.46701, 59.37458 ] },
         "type": "Feature",
         "properties": {    "serviceUri": "http://www.disit.org/km4city/resource/iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFE894AF7",
           "serviceType": "Environment_Smart_street_light",
           "name": "5C0272FFFE894AF7",
           "typeLabel": "Smart street light",
           "protocol": "ngsi",
           "format": "json",
           "model": "SmartLightCapelon2",
           "producer": "Capelon",
           "macaddress": "",
           "brokerName": "orionCAPELON-UNIFI",
           "ownership": "public",
           "organization": "CAPELON",
           "description": "",
           "website": "",
           "maintenanceUrl": "",
           "maxCapacity": "",
           "minCapacity": "",
           "isMobile": "",
           "nature": "Environment",
        ….
        ….
```

Query by value

Queries can be complex *by geo-area, by cathegory, by IoT Device Model, a list of ServiceURI (all the same kind), with filters by value on specific Variables (numeric, and textual in AND), QUERY:*

- https://www.snap4city.org/superservicemap/api/v1/**iot-search**/?selection=43.77;11.2&maxDists=700.2&model=CarPark

- https://www.snap4city.org/superservicemap/api/v1/i**ot-search**/?selection=42.014990;10.217347;43.7768;11.2515&model=metrotrafficsensor&**valueFilters=vehicleFlow>0.5;vehicleFlow<300**

- https://www.snap4city.org/superservicemap/api/v1/**iot-search**/?selection=43.77;11.2&maxDists=200.2&model=metrotrafficsensor&valueFilters=vehicleFlow>10;vehicleFlow<400&serviceUri=http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO1;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO10;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO11;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO13;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO14;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO15;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO16;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO17;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO18;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO19;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO2;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO20;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO21;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO22;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO23;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO24;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO25;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO26

How to Get the «Query» used in More Options (3)

- **ServiceMap (specific KB) and Query service**
  - The Query performed is saved and can be recalled with a QueryID, valid for that specific KB, and not accessible via SuperServiceMap / Federated KB
  - The QueryID is communicated via email
  - Specific REST Call with HTML is also provided to change the Query in server associated with the QueryID received

- **Query ID (only Read and Read/Write of the query)**
  - https://servicemap.disit.org/WebAppGrafo/api/v1/?queryId=1c8111893d40a2bb07a2078ffe299ced&format=json
  - Cannot be used for Custom PINs.
  - **Cannot be used to get data via ServiceMap since the Query ID is KB based**

Special Commands in «Query» of More Options (4)

- **Commands for Special Tool**:
  - **Traffic Flow** tool: https://firenzetraffic.km4city.org/trafficRTDetails/roads/read.php
  - **Scenario** tool: /scenario/
  - **Whatif** tool: /whatif/
- **Heatmaps**, see Data Analytic part of the training for the several versions which can be used:
  - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=PM2_5Average24HourFlorence
  - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=denseNO2_Firenze_IDW
  - WMSServer that is a GeoServer may be different for different installations of Snap4City

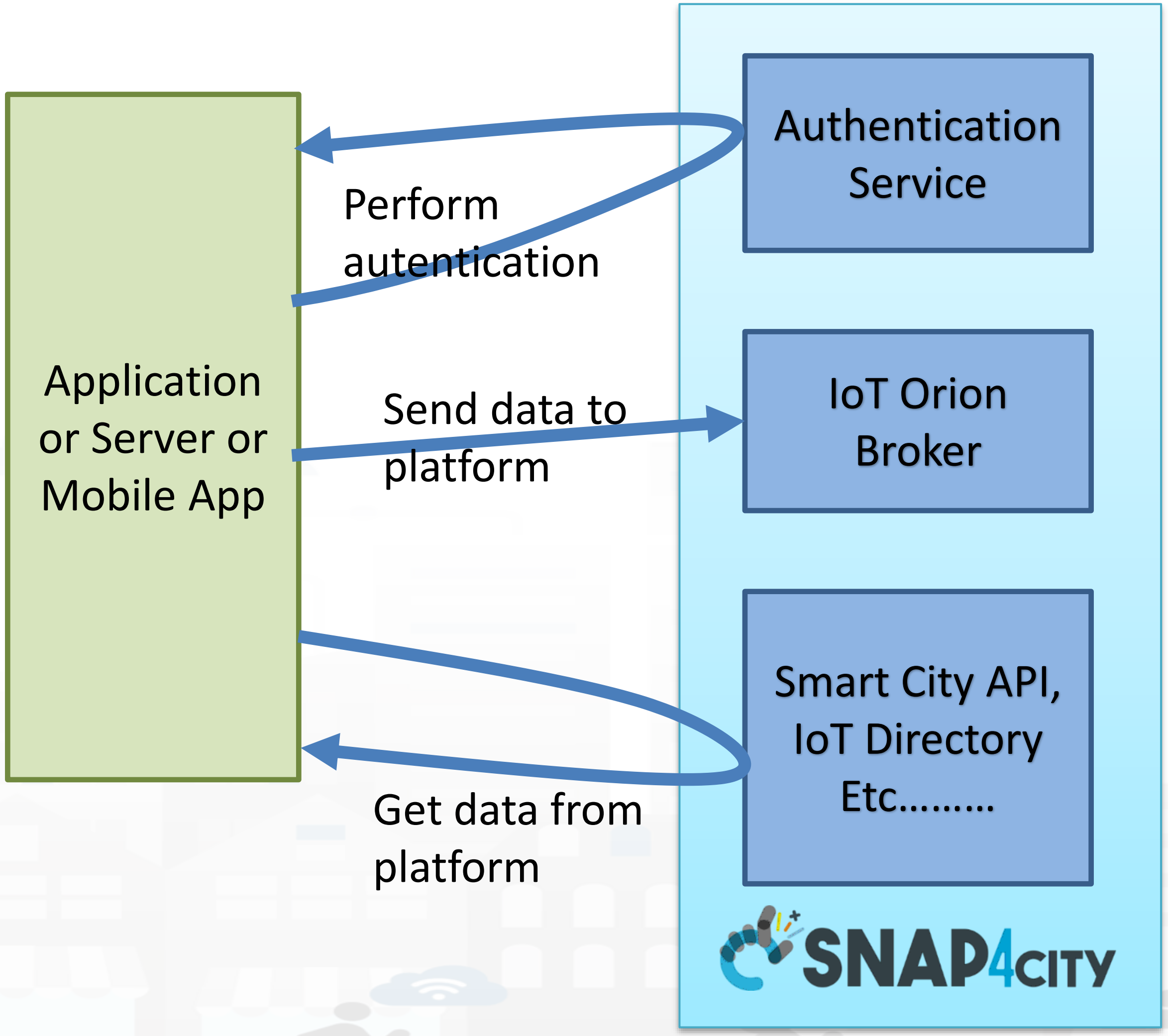
Time Series Data Access

- Time Series are attached to Devices which are identified by ServiceURI
- To **Access at the Time Series** (also called real time data) you can:
  1. From IoT App use the block «service info dev» **In this case, you automatically access to your private and delegated data. You do not need to perform the authentication since it is performed directly from the microservice IoT App context, both on cloud and on edge**
  2. From Python/Rstudio, Web and Mobile App, you can call Smart City API, see in this section and in ***Part 7 of the course***.
  3. Retrieve data from Processing Logic (IoT App) and pass them to Python/Rstudio as presented in other sections. This approach is viable for small amount of data, such as some thousands. For larger amount of data or to be more efficient we suggest to use case (2) which is a direct access to the Smart City API.

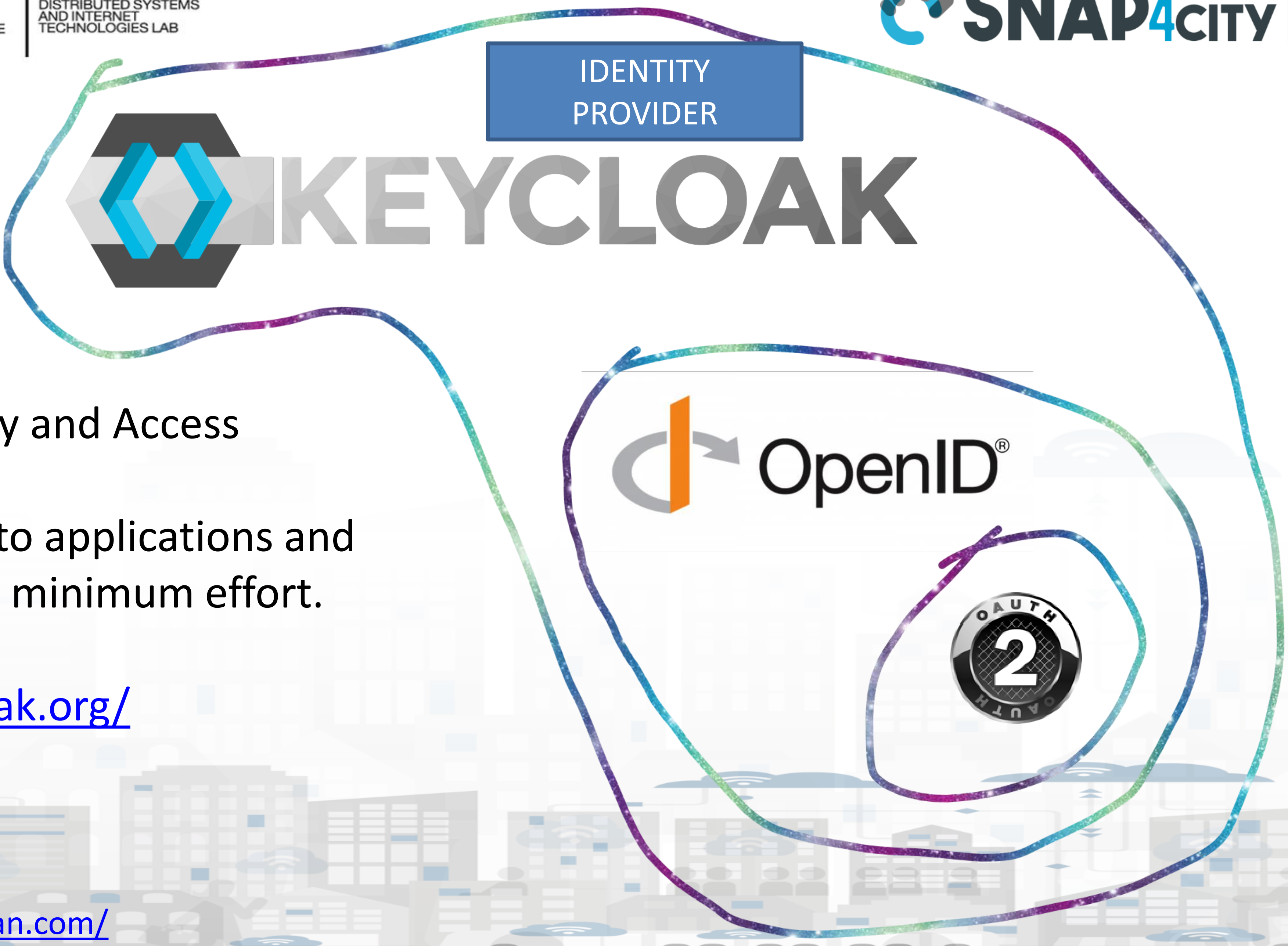
TOP

Develop: Provide Data and Access to Protected Data

Approaches



- **First**
  - authenticate
- **Second**
  - use the services to provide and/or get data


IDENTITY PROVIDER

Open Source Identity and Access Management
Add authentication to applications and secure services with minimum effort.

https://www.keycloak.org/

https://www.postman.com/

Two Possible Approaches for Authentication

Authentication Code Flow Protocol (confidential application)

- For Web Application with Server Side functionalities or native applications, including services towards mobile applications

Single Page App

- For Web Application without Server Side functionalities

Implicit Flow Protocol referred to as Direct Grant with username/password **(public applications)**

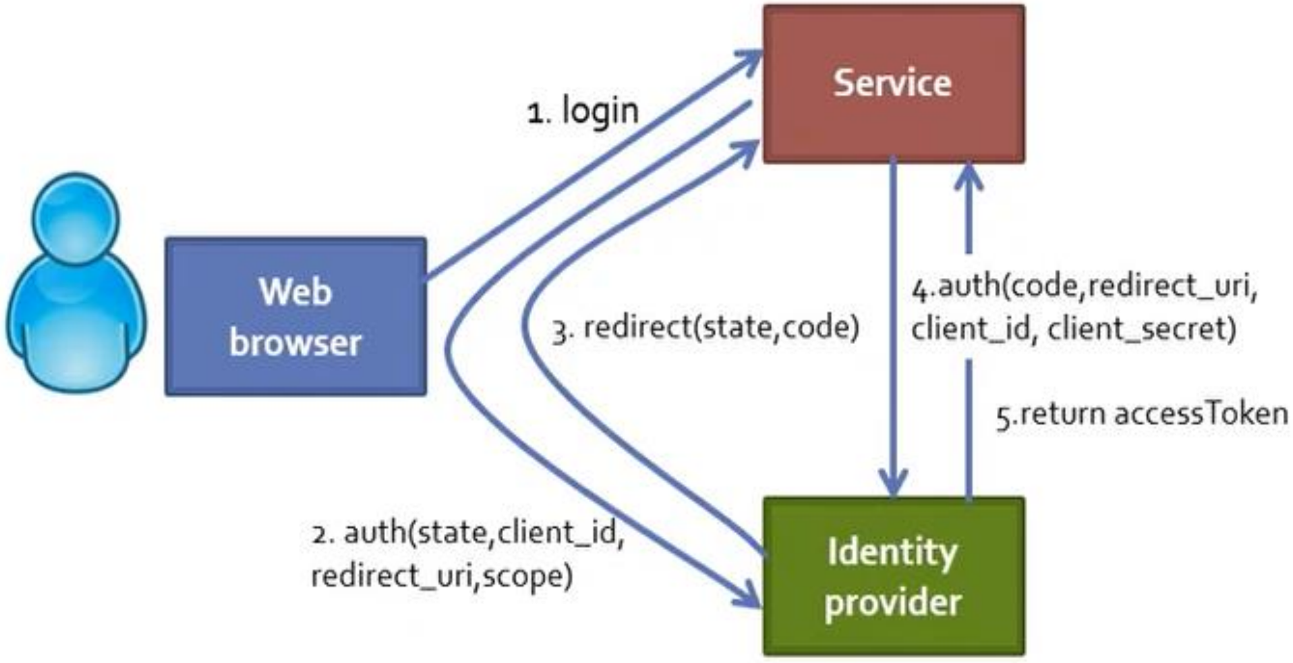
- Less secure: It’s not recommended to use this flow unless you absolutely need to
- For Front-End Web Application that do not have Server Side functionalities.
- JavaScript can do only this kind of applications

Authentication Flow Protocol (confidential application, Web Server Application)

Step1 The Web Server Application provides a way to securely store information, and provide service to your users via HTML pages

- In particular: **client_id and client_secret are secured on the WSA**
  - **They** have to be requested to the snap4City platform organization
  - Snap4City has to know the **redirect uri** of your Application Server to complete the round and provide back the information
- This approach is valid for application servers which provide html pages to your users, while this information is saved into the Applications Server which also interact with the Snap4City Platform
  - It in practice the same approch used by the Dashboard manager to provice access to the dashboard at the users.

Authentication Flow Protocol, step 2


Service
1. login
Web browser
3. redirect(state,code)
4.auth(code,redirect_uri, client_id, client_secret)
5.return accessToken
2. auth(state,client_id, redirect_uri,scope)
Identity provider

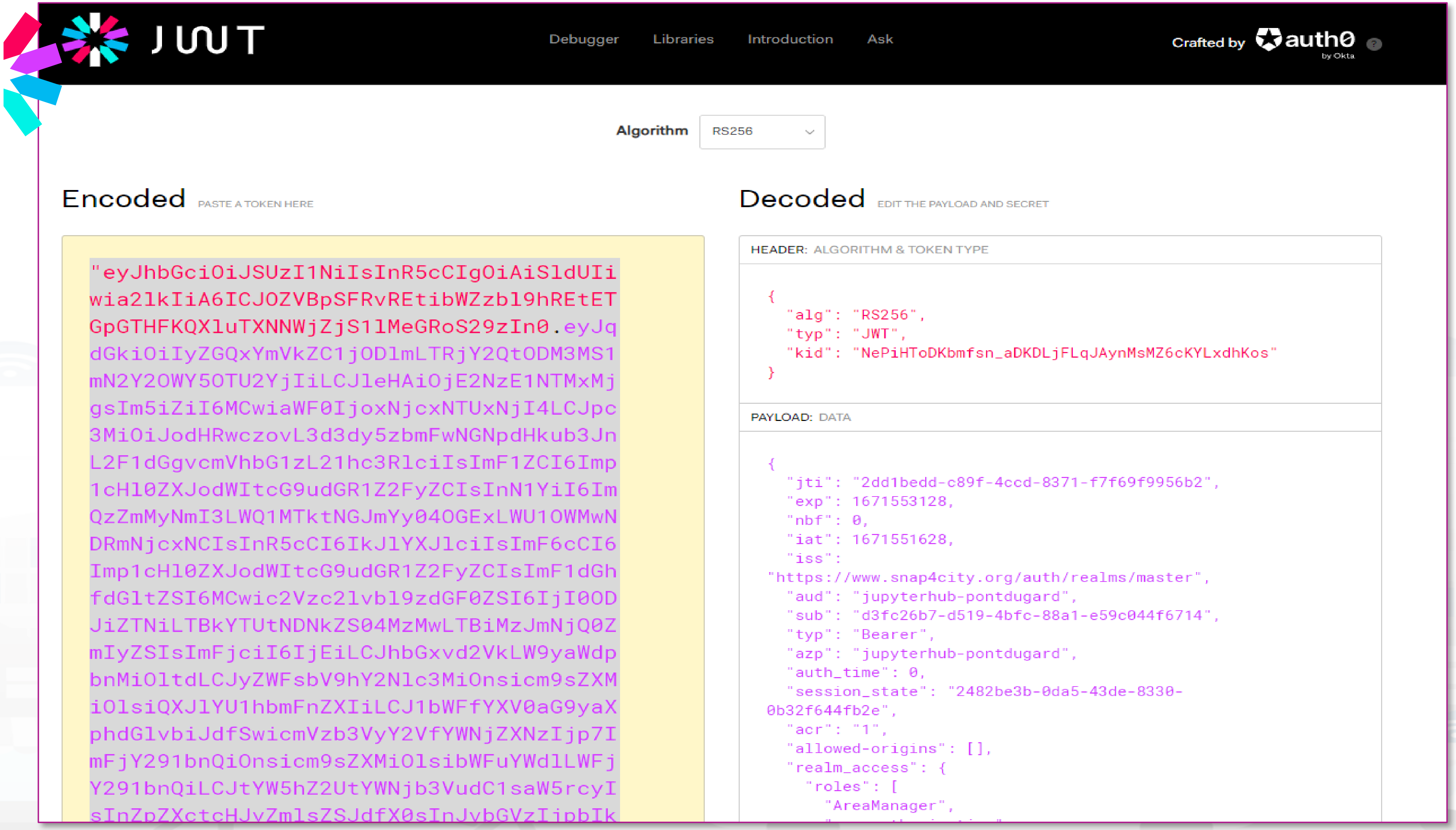
KEYCLOAK
SNAP4CITY

Step2

- the user login is redirected to the identity provider
- 2) Given the **client_id** of the application and the **client_secret** (in the diagram called code)
- 5) The Service get the AccessToken
- Then the Service can pose any API rest call to get data for the User

Access Token of OpenID®

- The access token is in the format of JSON Web Tokens
- https://jwt.io/
- The access token was:
- eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtETGpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.eyJqdGkiOiIyZGQxYmVkZC1jODlmLTRjY2QtODM3MS1mN2Y2OWY5OTU2YjIiLCJleHAiOjE2NzE1NTMxMjgsIm5iZiI6MCwiaWF0IjoxNjcxNTUxNjI4LCJpc3MiOiJodHRwczovL3d3dy5zbmFwNGNpdHkub3JnL2F1dGgvcmVhbG1zL21hc3RlciIsImF1ZCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsInN1YiI6ImQzZmMyNmI3LWQ1MTktNGJmYy04OGExLWU1OWMwNDRmNjcxNCIsInR5cCI6IkJlYXJlciIsImF6cCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsImF1dGhfdGltZSI6MCwic2Vzc2lvbl9zdGF0ZSI6IjI0ODJiZTNiLTBkYTUtNDNkZS04MzMwLTBiMzJmNjQ0ZmIyZSIsImFjciI6IjEiLCJhbGxvd2VkLW9yaWdpbnMiOltdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiQXJlYU1hbmFnZXIiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VyY2VfYWNjZXNzIjp7ImFjY291bnQiOnsicm9sZXMiOlsibWFuYWdlLWFjY291bnQiLCJtYW5hZ2UtYWNjb3VudC1saW5rcyIsInZpZXctcHJvZmlsZSJdfX0sInJvbGVzIjpbIkFyZWFNYW5hZ2VyIiwidW1hX2F1dGhvcml6YXRpb24iLCJvZmZsaW5lX2FjY2VzcyJdLCJuYW1lIjoic3VybmFtZSIsInByZWZlcnJlZF91c2VybmFtZSI6InRlc3R3YXN0ZSIsImZhbWlseV9uYW1lIjoic3VybmFtZSIsImVtYWlsIjoidGVzdHdhc3RlQGdtYWlsLmNvbSJ9.RE7whLSrXRpf3uXFV32rVb90YHY4GW0g087OS_k-p79Q84twdQswu-8OaAT0bV1RKep0qpRKZpWAsBWcHwrWEeDDNadUbv6n-GmUT0qfZRTpRzn2N8JfpqHGaI2sC4-ThstKxgH99fkI6e_9ubZOz4G9zWHQRrIHTcEmReYfazOnutdmgSX0F-ErM8eO9vRPmUmWBn5y7ZUm8re7CH6UPZNb15P4dXUnfR_zZK3gw8tyUyXdkHSSYTZrtj3fFbVjq3zxzV1Do2aI-BpqS7quiCyHMG0qInYriWZKSARUyzjulL1QqBoSS6_xTe25wyizvwZ1BwHoeak40oRc0IqAgw



Algorithm RS256

Encoded PASTE A TOKEN HERE

```
"eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIi
wia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtET
GpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.eyJq
dGkiOiIyZGQxYmVkZC1jODlmLTRjY2QtODM3MS1
mN2Y2OWY5OTU2YjIiLCJleHAiOjE2NzE1NTMxMj
gsIm5iZiI6MCwiaWF0IjoxNjcxNTUxNjI4LCJpc
3MiOiJodHRwczovL3d3dy5zbmFwNGNpdHkub3Jn
L2F1dGgvcmVhbG1zL21hc3RlciIsImF1ZCI6Imp
1cHl0ZXJodWItcG9udGR1Z2FyZCIsInN1YiI6Im
QzZmMyNmI3LWQ1MTktNGJmYy04OGExLWU1OWMwN
DRmNjcxNCIsInR5cCI6IkJlYXJlciIsImF6cCI6
Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsImF1dGh
fdGltZSI6MCwic2Vzc2lvbl9zdGF0ZSI6IjI0OD
JiZTNiLTBkYTUtNDNkZS04MzMwLTBiMzJmNjQ0Z
mIyZSIsImFjciI6IjEiLCJhbGxvd2VkLW9yaWdp
bnMiOltdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXM
iOlsiQXJlYU1hbmFnZXIiLCJ1bWFfYXV0aG9yaX
phdGlvbiJdfSwicmVzb3VyY2VfYWNjZXNzIjp7I
mFjY291bnQiOnsicm9sZXMiOlsibWFuYWdlLWFj
Y291bnQiLCJtYW5hZ2UtYWNjb3VudC1saW5rcyI
sInZpZXctcHJvZmlsZSJdfX0sInJvbGVzIjpbIk
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

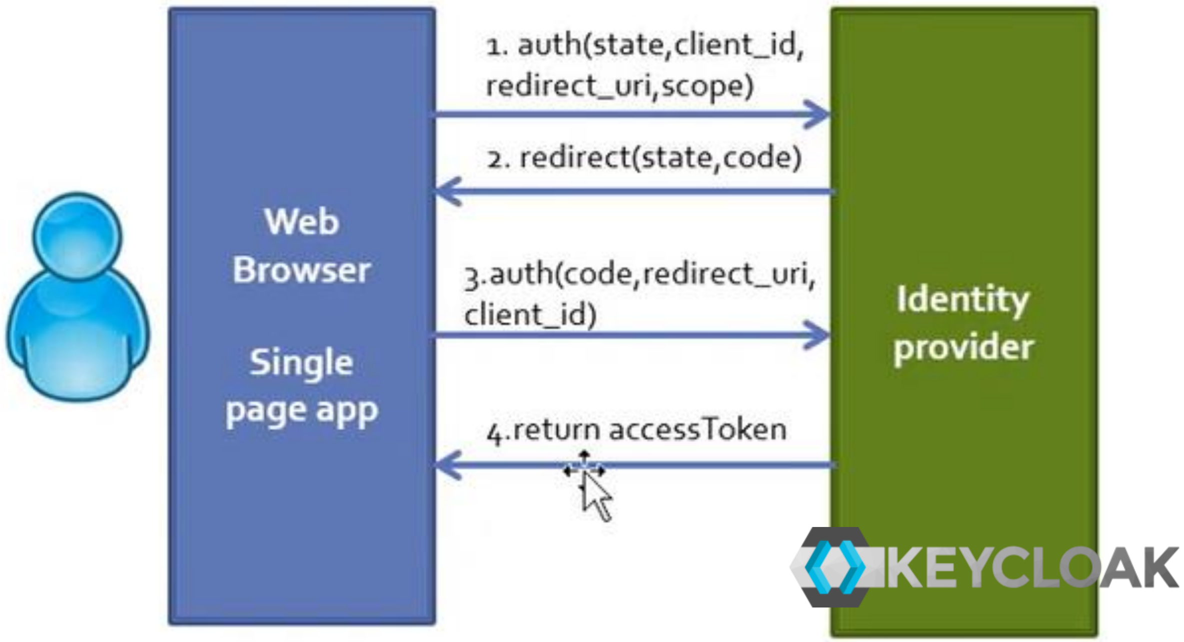
```
{
  "alg": "RS256",
  "typ": "JWT",
  "kid": "NePiHToDKbmfsn_aDKDLjFLqJAynMsMZ6cKYLxdhKos"
}
```

PAYLOAD: DATA

```
{
  "jti": "2dd1bedd-c89f-4ccd-8371-f7f69f9956b2",
  "exp": 1671553128,
  "nbf": 0,
  "iat": 1671551628,
  "iss":
"https://www.snap4city.org/auth/realms/master",
  "aud": "jupyterhub-pontdugard",
  "sub": "d3fc26b7-d519-4bfc-88a1-e59c044f6714",
  "typ": "Bearer",
  "azp": "jupyterhub-pontdugard",
  "auth_time": 0,
  "session_state": "2482be3b-0da5-43de-8330-
0b32f644fb2e",
  "acr": "1",
  "allowed-origins": [],
  "realm_access": {
    "roles": [
      "AreaManager",
```

Single Page App

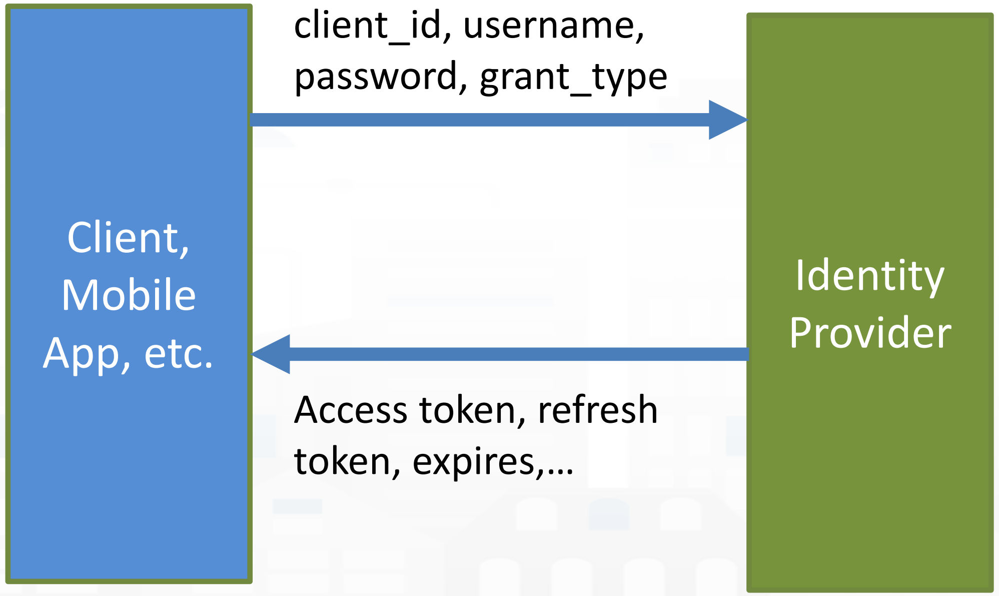
- The Single Page has no secure way to store information on client side
- All the secret information is maintained on the Identity Provider side



- The Single Page has to bring the login on the Identify Provider, which redirect on the applications
- The process follow the above presented approach
- Given the **client_id** of the application, the users can get the accessToken to make requests.

Two Simpler Cases

- Public Applications



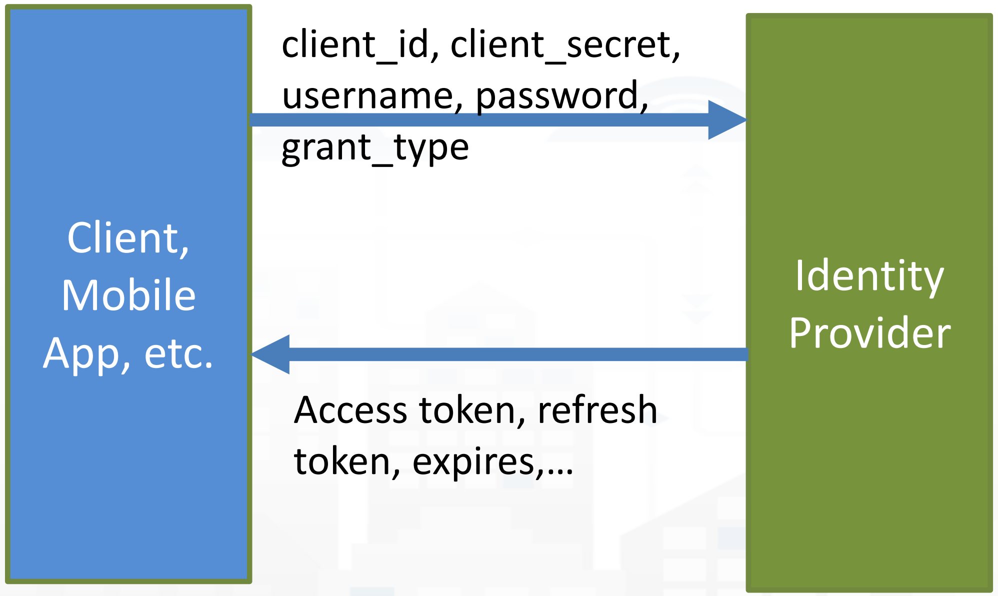
Client, Mobile App, etc.

client_id, username, password, grant_type

Identity Provider

Access token, refresh token, expires,…

- Implicit Flow Protocol



Client, Mobile App, etc.

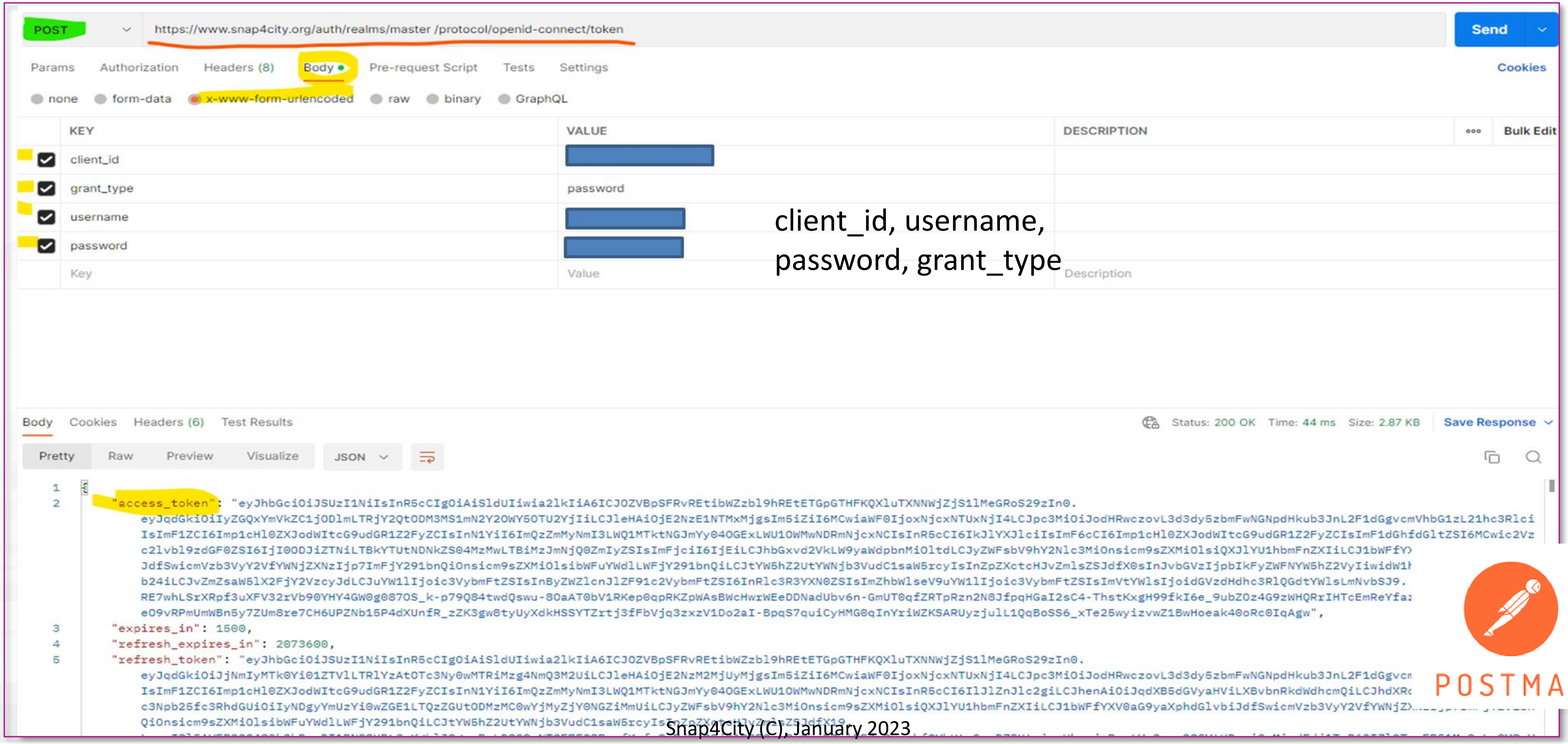
client_id, client_secret, username, password, grant_type

Identity Provider

Access token, refresh token, expires,…

Public Applications

https://www.snap4city.org/auth/realms/master/protocol/openid-connect/token



POST https://www.snap4city.org/auth/realms/master /protocol/openid-connect/token

○ none ○ form-data ● x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

	KEY	VALUE	DESCRIPTION
☑	client_id	[redacted]	
☑	grant_type	password	
☑	username	[redacted]	
☑	password	[redacted]	
	Key	Value	Description

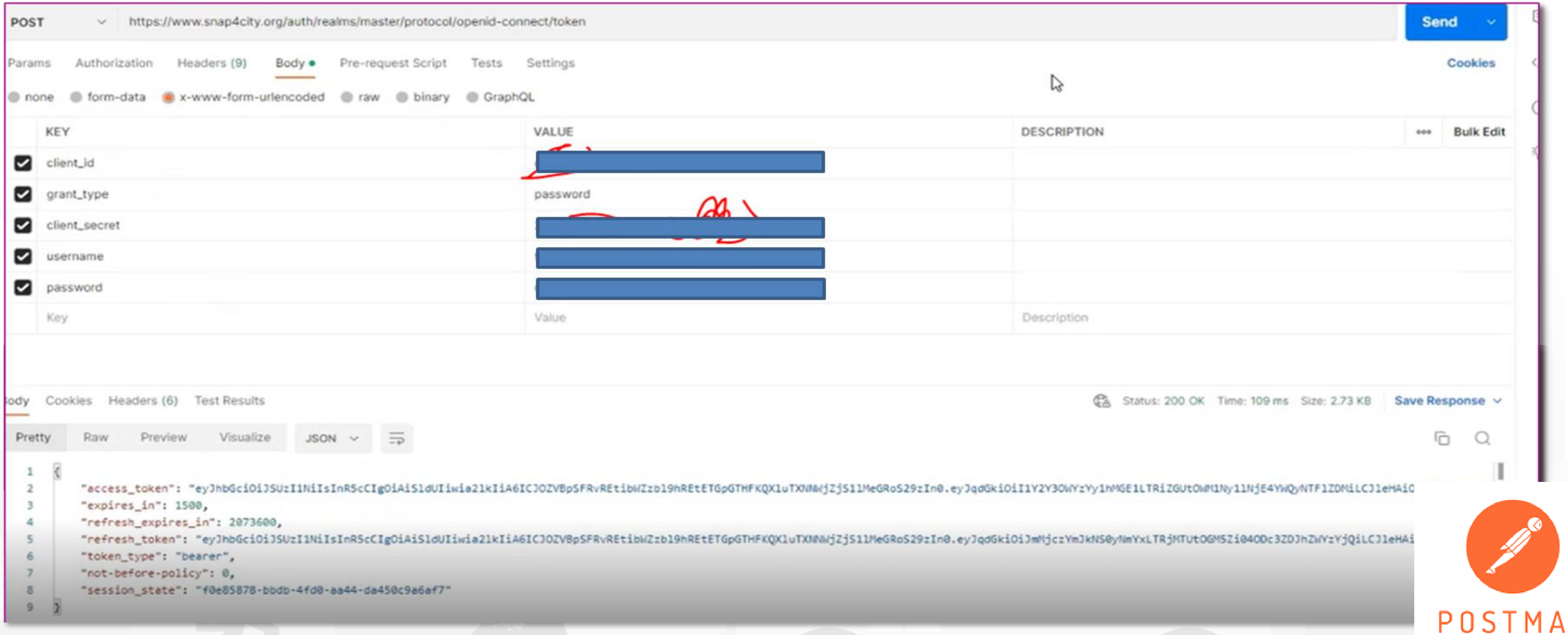
client_id, username, password, grant_type

Status: 200 OK Time: 44 ms Size: 2.87 KB

```
{
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtETGpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.
        eyJqdGkiOiIyZGQxYmVkZC1jODlmLTRjY2QtODM3MS1mN2Y2OWY5OTU2YjIiLCJleHAiOjE2NzE1NTMxMjgsIm5iZiI6MCwiaWF0IjoxNjcxNTUxNjI4LCJpc3MiOiJodHRwczovL3d3dy5zbmFwNGNpdHkub3JnL2F1dGgvcmVhbG1zL21hc3Rlci
        IsImF1ZCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsInN1YiI6ImQzZmMyNmI3LWQ1MTktNGJmYy04OGExLWU1OWMwNDRmNjcxNCIsInR5cCI6IkJlYXJlciIsImF6cCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsImF1dGhfdGltZSI6MCwic2Vz
        c2lvbl9zdGF0ZSI6IjI0ODJiZTNiLTBkYTUtNDNkZS04MzMwLTBiMzJmNjQ0ZmIyZSIsImFjciI6IjEiLCJhbGxvd2VkLW9yaWdpbnMiOltdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiQXJlYU1hbmFnZXIiLCJ1bWFfY
        JdfSwicmVzb3VyY2VfYWNjZXNzIjp7ImFjY291bnQiOnsicm9sZXMiOlsibWFuYWdlLWFjY291bnQiLCJtYW5hZ2UtYWNjb3VudC1saW5rcyIsInZpZXctcHJvZmlsZSJdfX0sInJvbGVzIjpbIkFyZWFNYW5hZ2VyIiwidW1h
        b24iLCJvZmZsaW5lX2FjY2VzcyJdLCJuYW1lIjoic3VybmFtZSIsInByZWZlcnJlZF91c2VybmFtZSI6InRlc3R3YXN0ZSIsImZhbWlseV9uYW1lIjoic3VybmFtZSIsImVtYWlsIjoidGVzdHdhc3RlQGdtYWlsLmNvbSJ9.
        RE7whLSrXRpf3uXFV32rVb90YHY4GW0g087OS_k-p79Q84twdQswu-8OaAT0bV1RKep0qpRKZpWAsBWcHwrWEeDDNadUbv6n-GmUT0qfZRTpRzn2N8JfpqHGaI2sC4-ThstKxgH99fkI6e_9ubZOz4G9zWHQRrIHTcEmReYfa
        eO9vRPmUmWBn5y7ZUm8re7CH6UPZNb15P4dXUnfR_zZK3gw8tyUyXdkHSSYTZrtj3fFbVjq3zxzV1Do2aI-BpqS7quiCyHMG0qInYriWZKSARUyzjulL1QqBoSS6_xTe25wyizvwZ1BwHoeak40oRc0IqAgw",
    "expires_in": 1500,
    "refresh_expires_in": 2073600,
    "refresh_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtETGpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.
        eyJqdGkiOiJjNmIyMTk0Yi01ZTVlLTRlYzAtOTc3Ny0wMTRiMzg4NmQ3M2UiLCJleHAiOjE2NzM2MjUyMjgsIm5iZiI6MCwiaWF0IjoxNjcxNTUxNjI4LCJpc3MiOiJodHRwczovL3d3dy5zbmFwNGNpdHkub3JnL2F1dGgvcm
        IsImF1ZCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsInN1YiI6ImQzZmMyNmI3LWQ1MTktNGJmYy04OGExLWU1OWMwNDRmNjcxNCIsInR5cCI6IlJlZnJlc2giLCJhenAiOiJqdXB5dGVyaHViLXBvbnRkdWdhcmQiLCJhdXRo
        c3Npb25fc3RhdGUiOiIyNDgyYmUzYi0wZGE1LTQzZGUtODMzMC0wYjMyZjY0NGZiMmUiLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiQXJlYU1hbmFnZXIiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VyY2VfYWNjZX
        QiOnsicm9sZXMiOlsibWFuYWdlLWFjY291bnQiLCJtYW5hZ2UtYWNjb3VudC1saW5rcyIsInZpZXctcHJvZmlsZSJdfX19.
```

Implicit Flow Protocol

- For some client_id the client_secret are needed



POST https://www.snap4city.org/auth/realms/master/protocol/openid-connect/token

○ none ○ form-data ● x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

	KEY	VALUE	DESCRIPTION
☑	client_id	[redacted]	
☑	grant_type	password	
☑	client_secret	[redacted]	
☑	username	[redacted]	
☑	password	[redacted]	
	Key	Value	Description

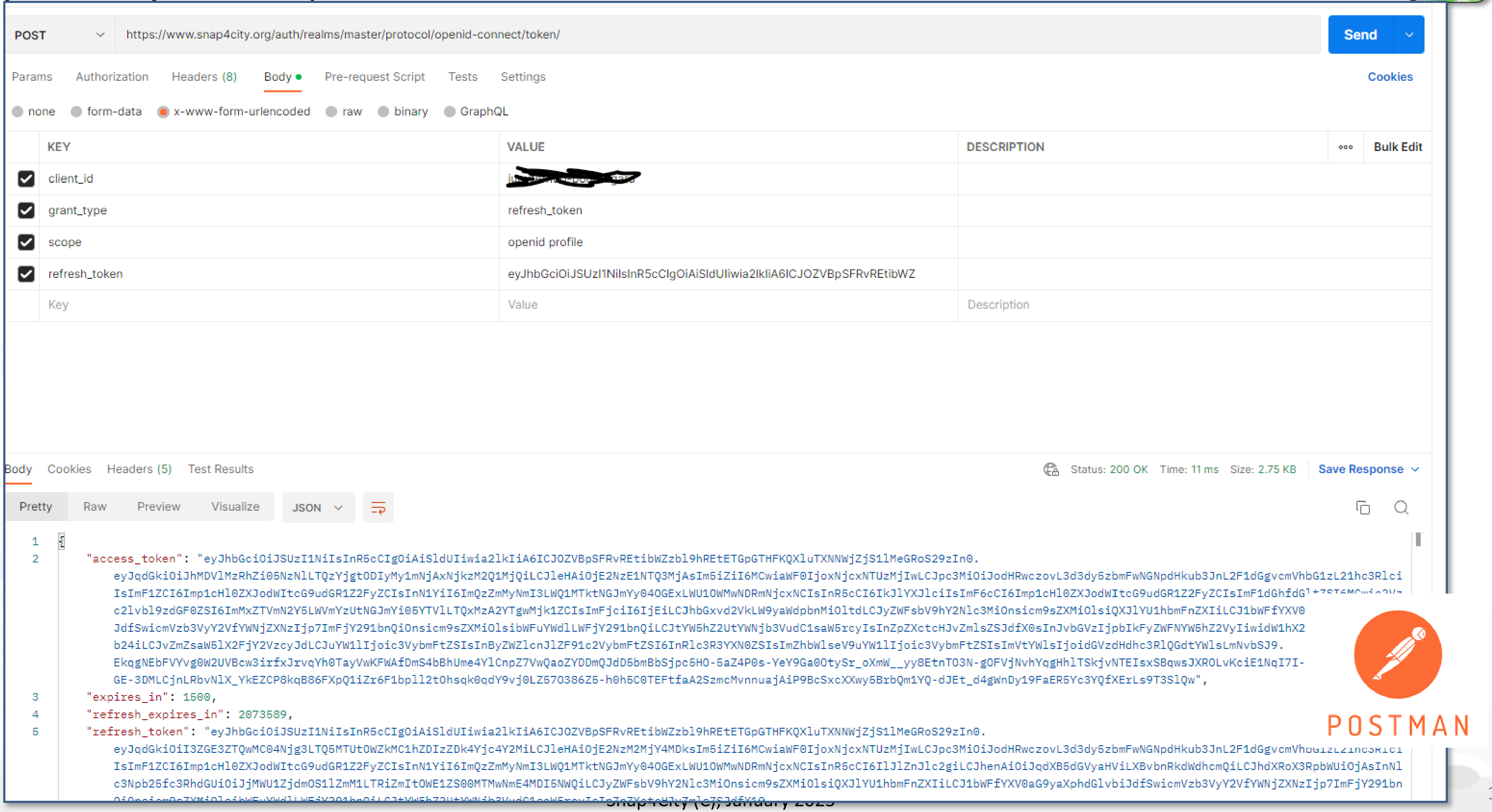
Status: 200 OK Time: 109 ms Size: 2.73 KB

```
{
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtETGpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.eyJqdGkiOiI1Y2Y3OWYzYy1hMGE1LTRiZGUtOWM1Ny1lNjE4YWQyNTF1ZDMiLCJleHAiO
    "expires_in": 1500,
    "refresh_expires_in": 2073600,
    "refresh_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtETGpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.eyJqdGkiOiJmMjczYmJkNS0yNmYxLTRjMTUtOGM5Zi04ODc3ZDJhZWYzYjQiLCJleHAi
    "token_type": "bearer",
    "not-before-policy": 0,
    "session_state": "f0e85878-bbdb-4fd0-aa44-da450c9a6af7"
}
```

POSTMAN

Access Token & Refresh Token

- Access tokens have typically of short duration
  - Once the access token is expired,
  - The refresh token can be used to request another fresh access token and this can be done at the endpoint
- https://www.snap4city.org/auth/realms/master/protocol/openid-connect/token/
- With the parameters reported in the next slide



POST https://www.snap4city.org/auth/realms/master/protocol/openid-connect/token/

○ none ○ form-data ● x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

	KEY	VALUE	DESCRIPTION
☑	client_id	[redacted]	
☑	grant_type	refresh_token	
☑	scope	openid profile	
☑	refresh_token	eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZ	
	Key	Value	Description

Status: 200 OK Time: 11 ms Size: 2.75 KB

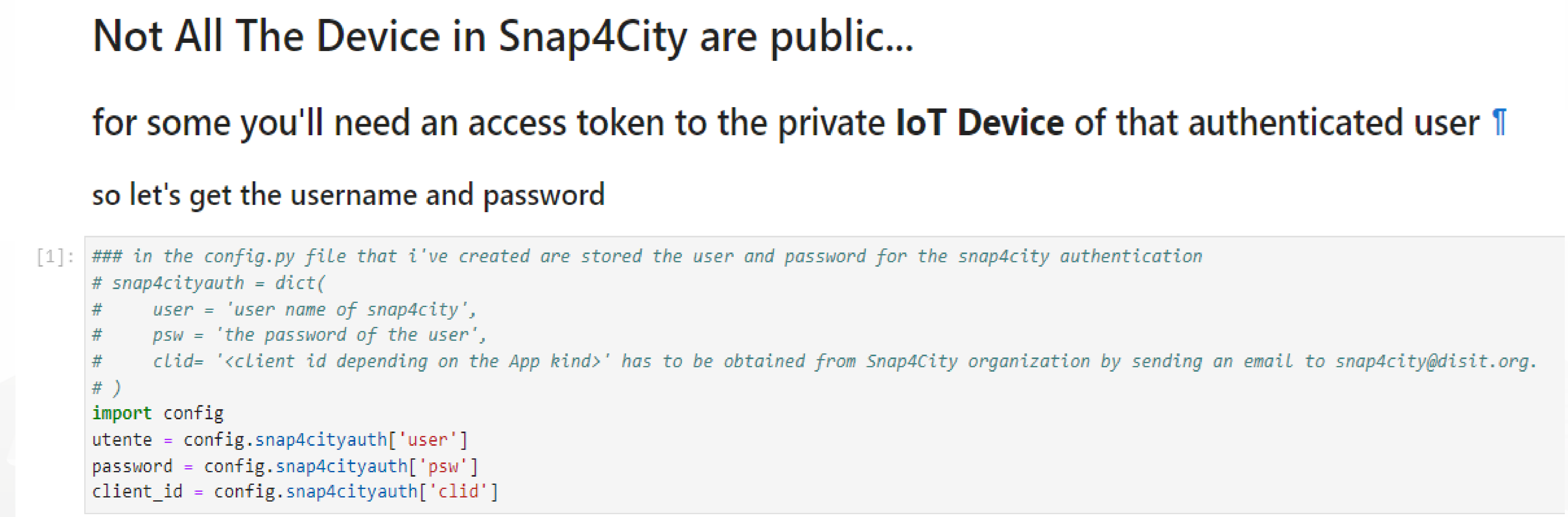
```
{
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtETGpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.
        eyJqdGkiOiJhMDVlMzRhZi05NzNlLTQzYjgtODIyMy1mNjAxNjkzM2Q1MjQiLCJleHAiOjE2NzE1NTQ3MjAsIm5iZiI6MCwiaWF0IjoxNjcxNTUzMjIwLCJpc3MiOiJodHRwczovL3d3dy5zbmFwNGNpdHkub3JnL2F1dGgvcmVhbG1zL21hc3Rlci
        IsImF1ZCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsInN1YiI6ImQzZmMyNmI3LWQ1MTktNGJmYy04OGExLWU1OWMwNDRmNjcxNCIsInR5cCI6IkJlYXJlciIsImF6cCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsImF1dGhfdG
        c2lvbl9zdGF0ZSI6ImMxZTVmN2Y5LWVmYzUtNGJmYi05YTVlLTQxMzA2YTgwMjk1ZCIsImFjciI6IjEiLCJhbGxvd2VkLW9yaWdpbnMiOltdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiQXJlYU1hbmFnZXIiLCJ1bWFfYXV0
        JdfSwicmVzb3VyY2VfYWNjZXNzIjp7ImFjY291bnQiOnsicm9sZXMiOlsibWFuYWdlLWFjY291bnQiLCJtYW5hZ2UtYWNjb3VudC1saW5rcyIsInZpZXctcHJvZmlsZSJdfX0sInJvbGVzIjpbIkFyZWFNYW5hZ2VyIiwidW1hX2
        b24iLCJvZmZsaW5lX2FjY2VzcyJdLCJuYW1lIjoic3VybmFtZSIsInByZWZlcnJlZF91c2VybmFtZSI6InRlc3R3YXN0ZSIsImZhbWlseV9uYW1lIjoic3VybmFtZSIsImVtYWlsIjoidGVzdHdhc3RlQGdtYWlsLmNvbSJ9.
        EkqgNEbFVYvg0W2UVBcw3irfxJrvqYh0TayVwKFWAfDmS4bBhUme4YlCnpZ7VwQaoZYDDmQJdD5bmBbSjpc5HO-5aZ4P0s-YeY9Ga0OtySr_oXmW__yy8EtnTO3N-gOFVjNvhYqgHhlTSkjvNTEIsxSBqwsJXROLvKciE1NqI7I-
        GE-3DMLCjnLRbvNlX_YkEZCP8kqB86FXpQ1iZr6F1bpll2tOhsqk0qdY9vj0LZ570386Z5-h0h5C0TEFtfaA2SzmcMvnnuajAiP9BcSxcXXwy5BrbQm1YQ-dJEt_d4gWnDy19FaER5Yc3YQfXErLs9T3SlQw",
    "expires_in": 1500,
    "refresh_expires_in": 2073589,
    "refresh_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJOZVBpSFRvREtibWZzbl9hREtETGpGTHFKQXluTXNNWjZjS1lMeGRoS29zIn0.
        eyJqdGkiOiI3ZGE3ZTQwMC04Njg3LTQ5MTUtOWZkMC1hZDIzZDk4Yjc4Y2MiLCJleHAiOjE2NzM2MjY4MDksIm5iZiI6MCwiaWF0IjoxNjcxNTUzMjIwLCJpc3MiOiJodHRwczovL3d3dy5zbmFwNGNpdHkub3JnL2F1dGgvcmVhbG1zL21hc3Rlci
        IsImF1ZCI6Imp1cHl0ZXJodWItcG9udGR1Z2FyZCIsInN1YiI6ImQzZmMyNmI3LWQ1MTktNGJmYy04OGExLWU1OWMwNDRmNjcxNCIsInR5cCI6IlJlZnJlc2giLCJhenAiOiJqdXB5dGVyaHViLXBvbnRkdWdhcmQiLCJhdXRoX3RpbWUiOjAsInNl
        c3Npb25fc3RhdGUiOiJjMWU1ZjdmOS1lZmM1LTRiZmItOWE1ZS00MTMwNmE4MDI5NWQiLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiQXJlYU1hbmFnZXIiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VyY2VfYWNjZXNzIjp7ImFjY291bn
```

TOP

Develop: How Cloud Containers may Access to Protected Data (example of Python)

Private Device Data Retrieval

- We'll use the cloud installation of jupyterhub
- https://www.snap4city.org/650



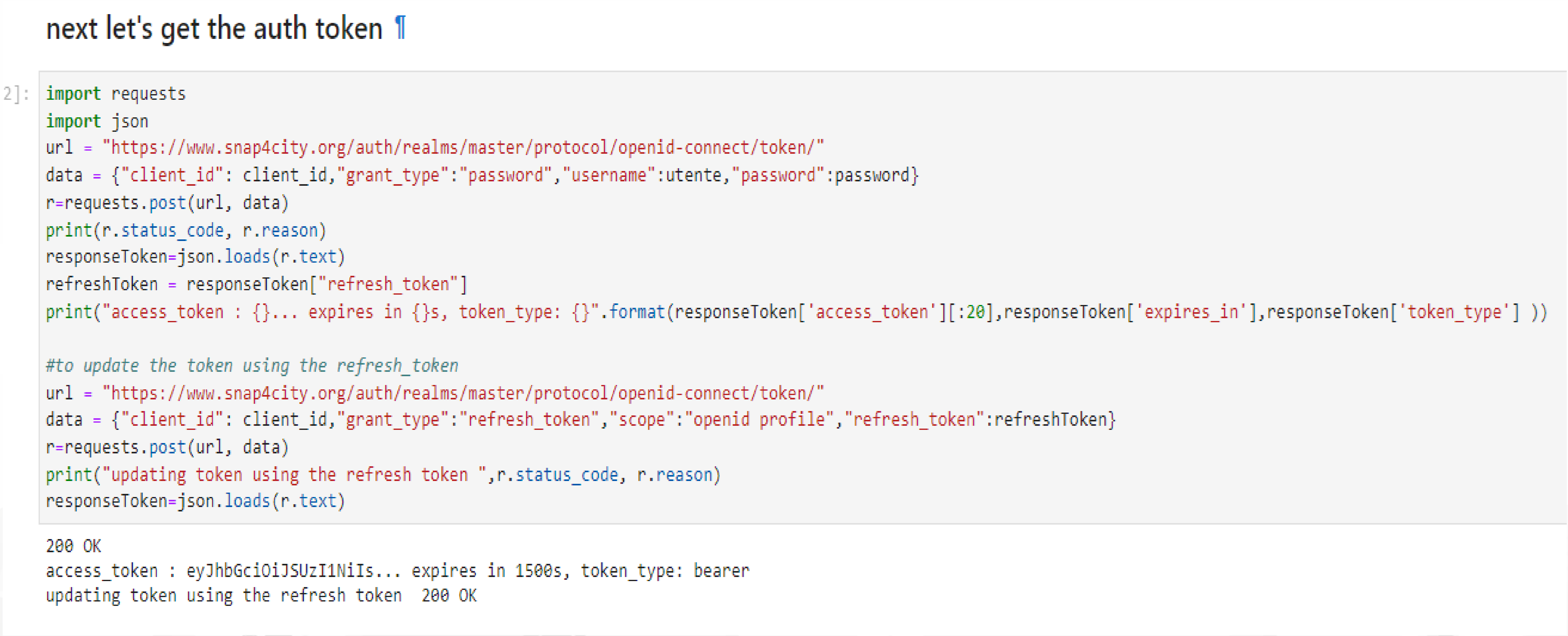
Not All The Device in Snap4City are public...

for some you'll need an access token to the private **IoT Device** of that authenticated user ¶

so let's get the username and password

```
[1]: ### in the config.py file that i've created are stored the user and password for the snap4city authentication
     # snap4cityauth = dict(
     #     user = 'user name of snap4city',
     #     psw = 'the password of the user',
     #     clid= '<client id depending on the App kind>' has to be obtained from Snap4City organization by sending an email to snap4city@disit.org.
     # )
     import config
     utente = config.snap4cityauth['user']
     password = config.snap4cityauth['psw']
     client_id = config.snap4cityauth['clid']
```

Private Device Data Retrieval



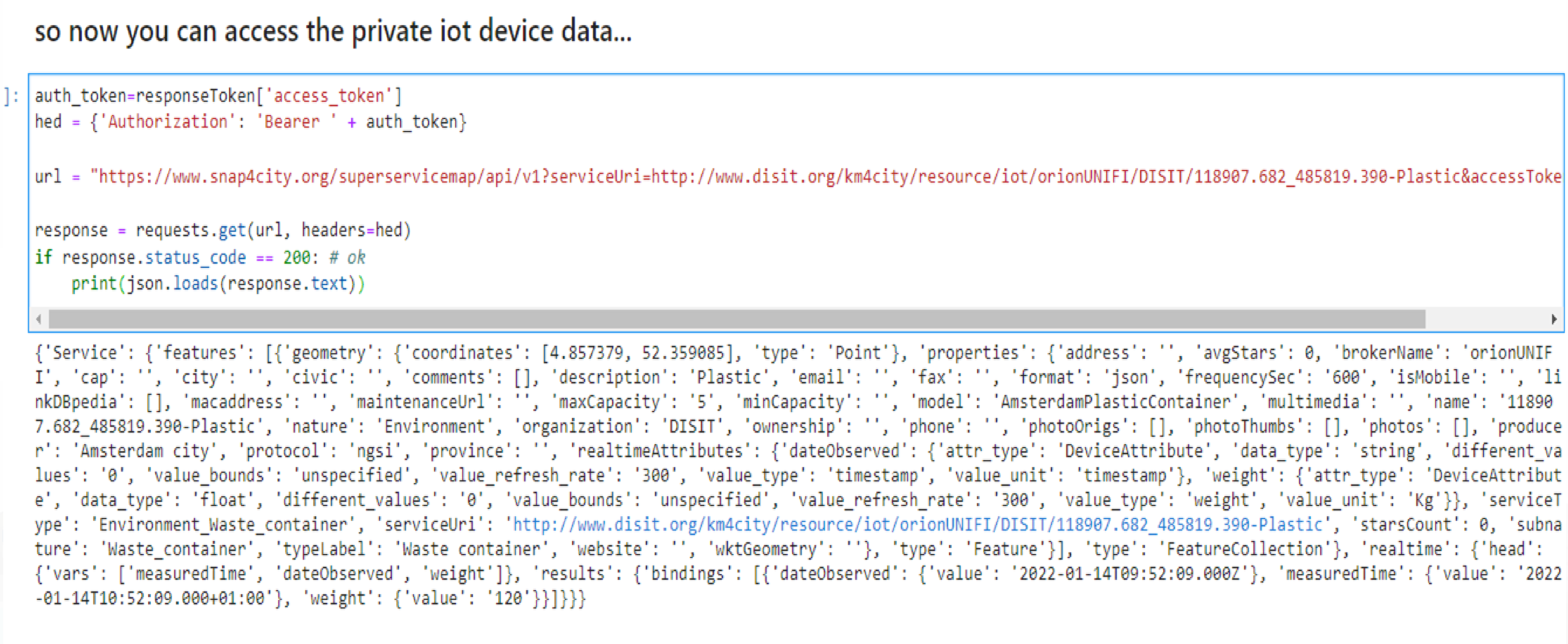
next let's get the auth token ¶

2]:

```python
import requests
import json
url = "https://www.snap4city.org/auth/realms/master/protocol/openid-connect/token/"
data = {"client_id": client_id,"grant_type":"password","username":utente,"password":password}
r=requests.post(url, data)
print(r.status_code, r.reason)
responseToken=json.loads(r.text)
refreshToken = responseToken["refresh_token"]
print("access_token : {}... expires in {}s, token_type: {}".format(responseToken['access_token'][:20],responseToken['expires_in'],responseToken['token_type'] ))

#to update the token using the refresh_token
url = "https://www.snap4city.org/auth/realms/master/protocol/openid-connect/token/"
data = {"client_id": client_id,"grant_type":"refresh_token","scope":"openid profile","refresh_token":refreshToken}
r=requests.post(url, data)
print("updating token using the refresh token ",r.status_code, r.reason)
responseToken=json.loads(r.text)
```

```
200 OK
access_token : eyJhbGciOiJSUzI1NiIs... expires in 1500s, token_type: bearer
updating token using the refresh token  200 OK
```

Private Device Data Retrieval



so now you can access the private iot device data...

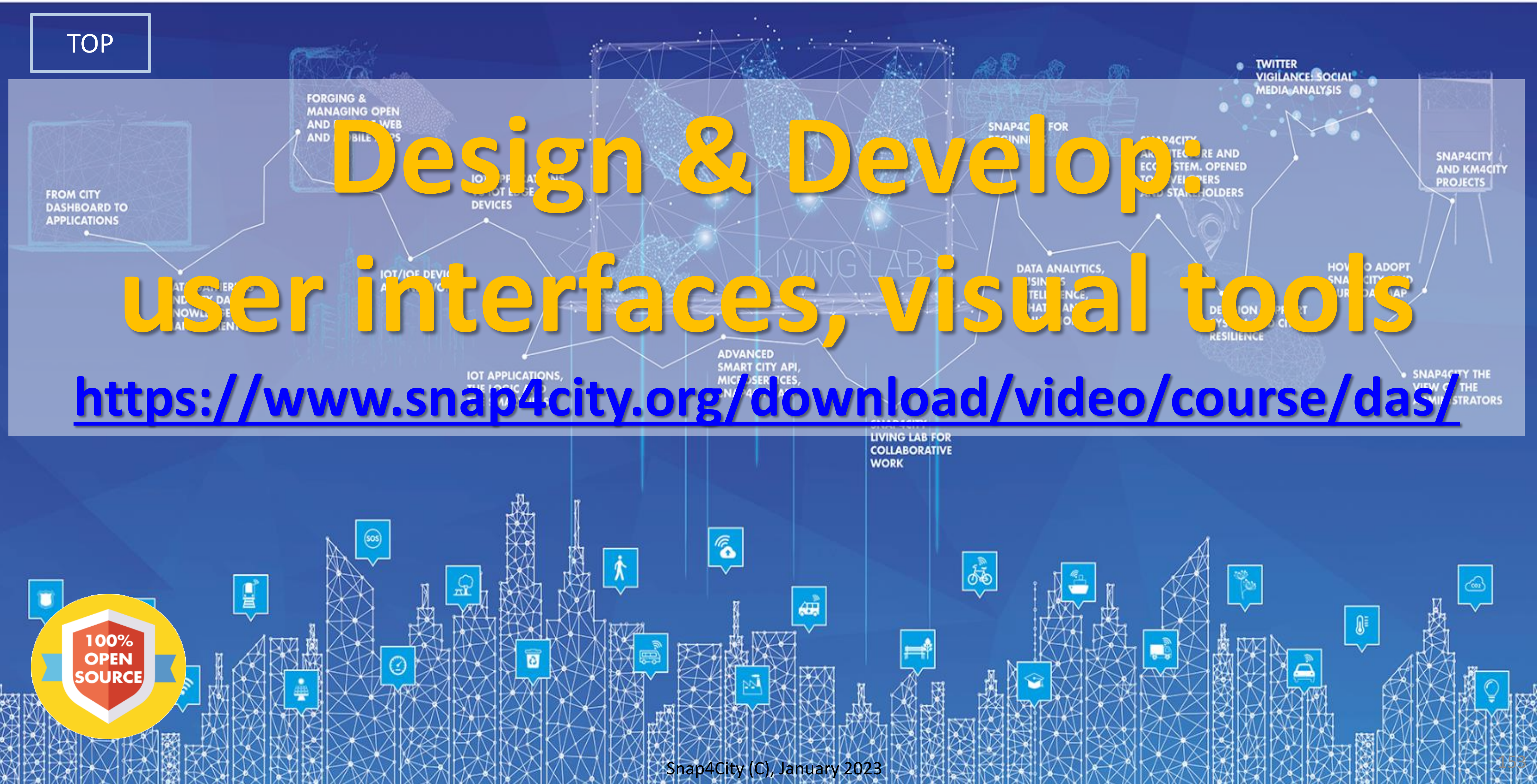
```
]: auth_token=responseToken['access_token']
   hed = {'Authorization': 'Bearer ' + auth_token}

   url = "https://www.snap4city.org/superservicemap/api/v1?serviceUri=http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/118907.682_485819.390-Plastic&accessToke

   response = requests.get(url, headers=hed)
   if response.status_code == 200: # ok
       print(json.loads(response.text))
```

```
{'Service': {'features': [{'geometry': {'coordinates': [4.857379, 52.359085], 'type': 'Point'}, 'properties': {'address': '', 'avgStars': 0, 'brokerName': 'orionUNIF
I', 'cap': '', 'city': '', 'civic': '', 'comments': [], 'description': 'Plastic', 'email': '', 'fax': '', 'format': 'json', 'frequencySec': '600', 'isMobile': '', 'li
nkDBpedia': [], 'macaddress': '', 'maintenanceUrl': '', 'maxCapacity': '5', 'minCapacity': '', 'model': 'AmsterdamPlasticContainer', 'multimedia': '', 'name': '11890
7.682_485819.390-Plastic', 'nature': 'Environment', 'organization': 'DISIT', 'ownership': '', 'phone': '', 'photoOrigs': [], 'photoThumbs': [], 'photos': [], 'produce
r': 'Amsterdam city', 'protocol': 'ngsi', 'province': '', 'realtimeAttributes': {'dateObserved': {'attr_type': 'DeviceAttribute', 'data_type': 'string', 'different_va
lues': '0', 'value_bounds': 'unspecified', 'value_refresh_rate': '300', 'value_type': 'timestamp', 'value_unit': 'timestamp'}, 'weight': {'attr_type': 'DeviceAttribut
e', 'data_type': 'float', 'different_values': '0', 'value_bounds': 'unspecified', 'value_refresh_rate': '300', 'value_type': 'weight', 'value_unit': 'Kg'}}, 'serviceT
ype': 'Environment_Waste_container', 'serviceUri': 'http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/118907.682_485819.390-Plastic', 'starsCount': 0, 'subna
ture': 'Waste_container', 'typeLabel': 'Waste container', 'website': '', 'wktGeometry': ''}, 'type': 'Feature'}], 'type': 'FeatureCollection'}, 'realtime': {'head':
{'vars': ['measuredTime', 'dateObserved', 'weight']}, 'results': {'bindings': [{'dateObserved': {'value': '2022-01-14T09:52:09.000Z'}, 'measuredTime': {'value': '2022
-01-14T10:52:09.000+01:00'}, 'weight': {'value': '120'}}]}}}
```

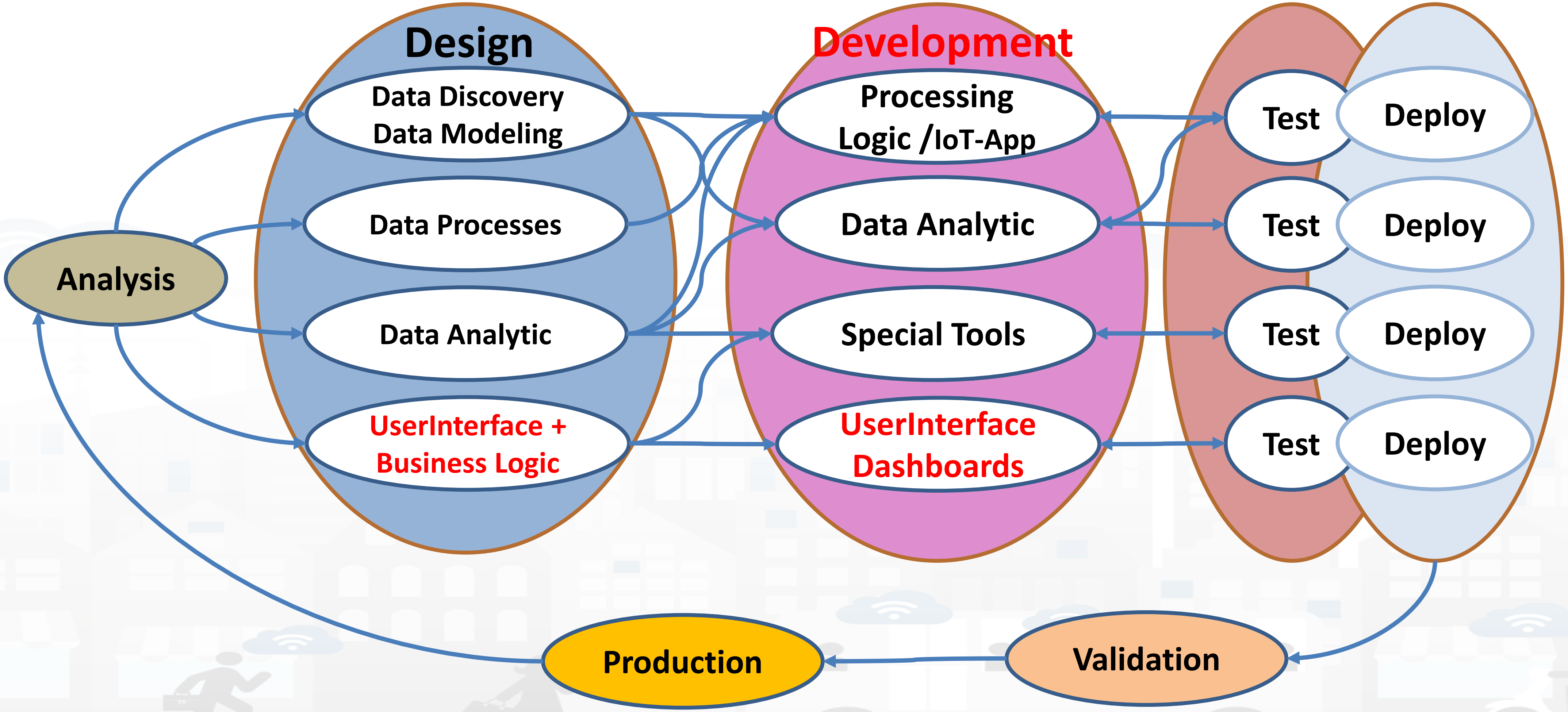
TOP

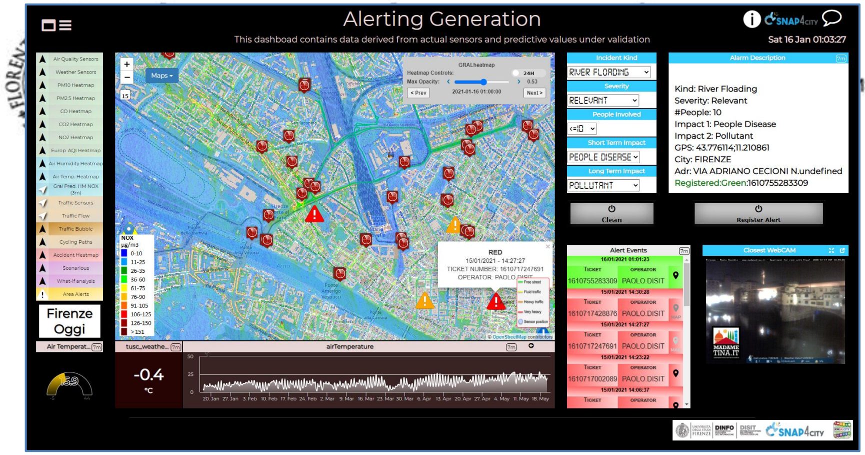
Design & Develop: user interfaces, visual tools

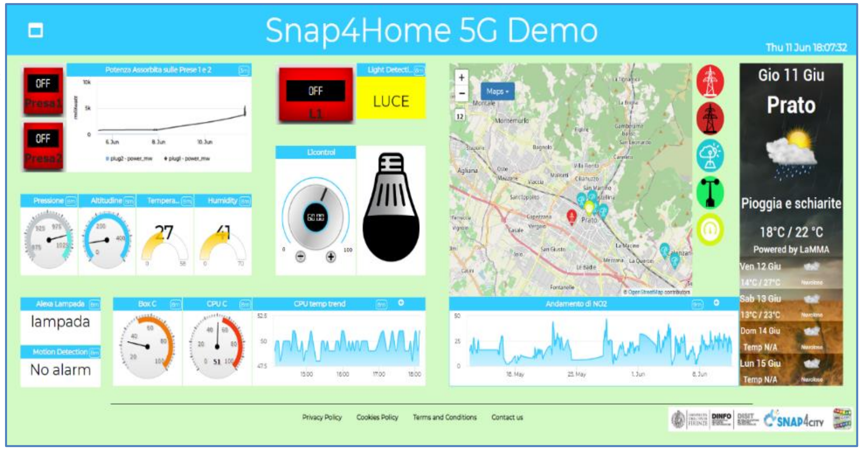
https://www.snap4city.org/download/video/course/das/

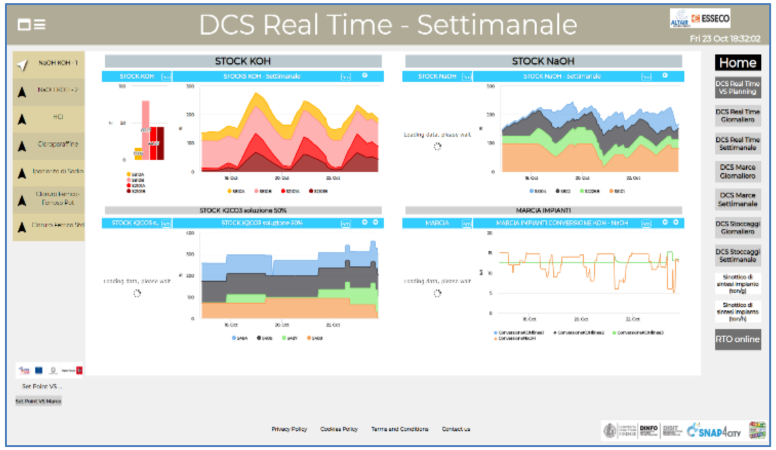

FORGING & MANAGING OPEN AND WEB AND MOBILE APPS
FROM CITY DASHBOARD TO APPLICATIONS
IOT APPLICATIONS IOT EDGE DEVICES
IOT/IOE DEVICES
LIVING LAB
SNAP4CITY FOR BEGINNERS
SNAP4CITY ARCHITECTURE AND ECOSYSTEM. OPENED TO DEVELOPERS AND STAKEHOLDERS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY AND KM4CITY PROJECTS
DATA ANALYTICS, USING INTELLIGENCE
HOW TO ADOPT SNAP4CITY
DECISION SUPPORT RESILIENCE
ADVANCED SMART CITY API, MICROSERVICES
IOT APPLICATIONS, THE LOGIC
SNAP4CITY THE VIEW OF THE ADMINISTRATORS
LIVING LAB FOR COLLABORATIVE WORK
100% OPEN SOURCE

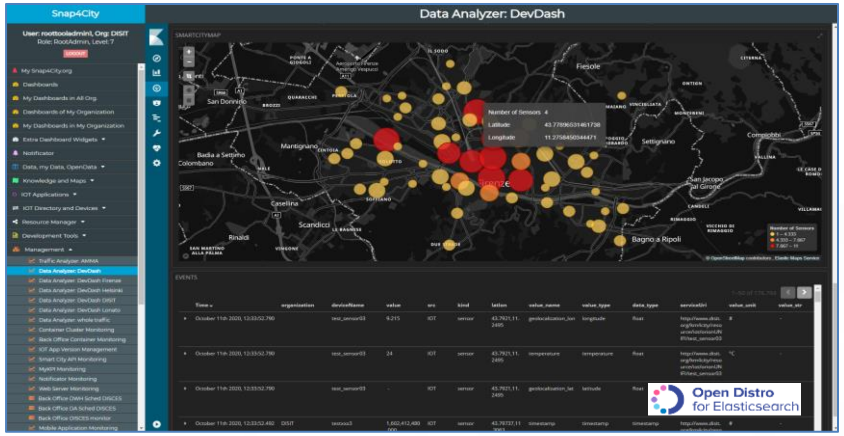
Development Life Cycle
Smart Solutions

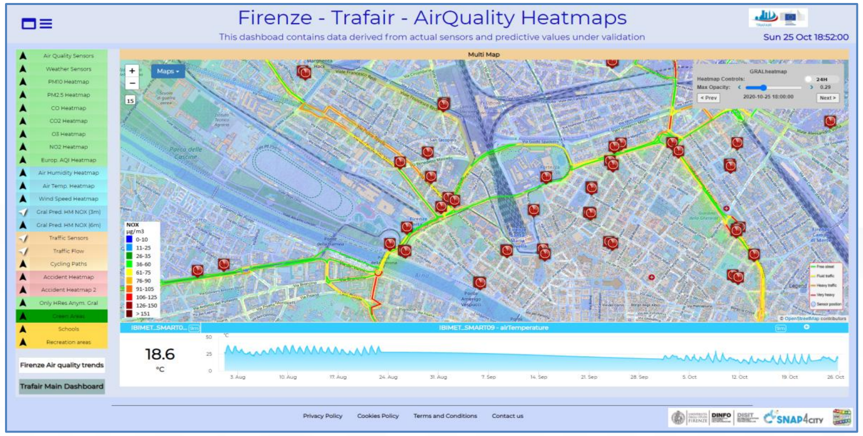


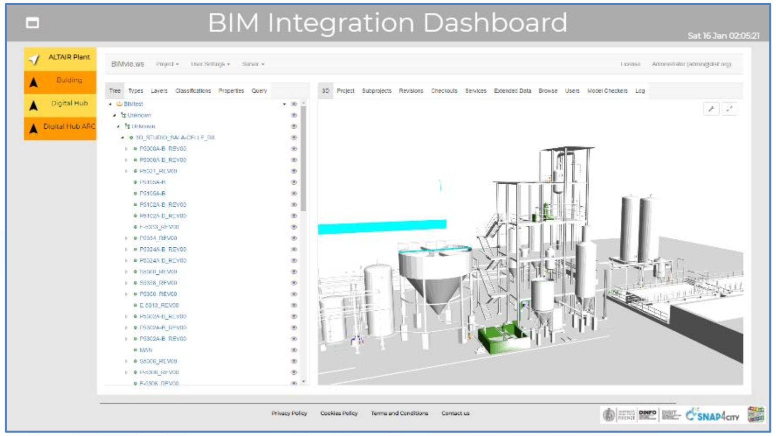

Alerting Generation
This dashboard contains data derived from actual sensors and predictive values under validation
Sat 16 Jan 01:03:27
Firenze Oggi
-0.4
Kind: River Flooding
Severity: Relevant
#People: 10
Impact 1: People Disease
Impact 2: Pollutant
City: FIRENZE
Clean
Register Alert
Alert Events

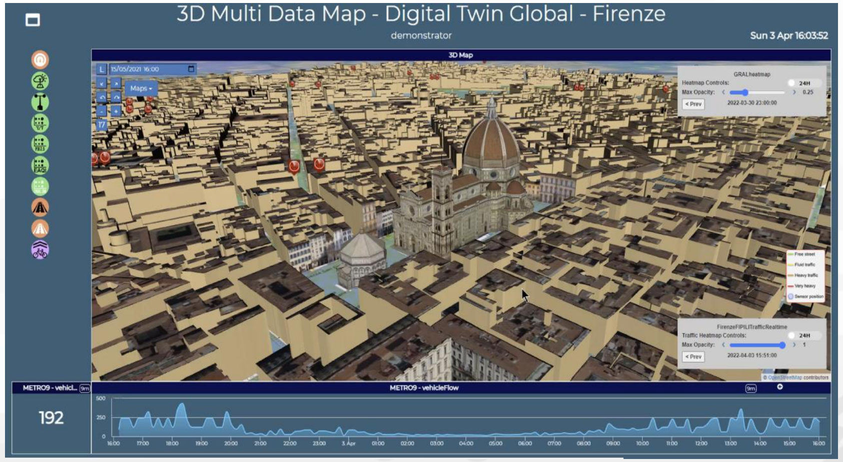

Snap4Home 5G Demo
Thu 11 Jun 18:07:32
OFF
LUCE
Gio 11 Giu
Prato
Pioggia e schiarite
18°C / 22 °C
lampada
No alarm
Privacy Policy
Cookies Policy
Terms and Conditions
Contact us

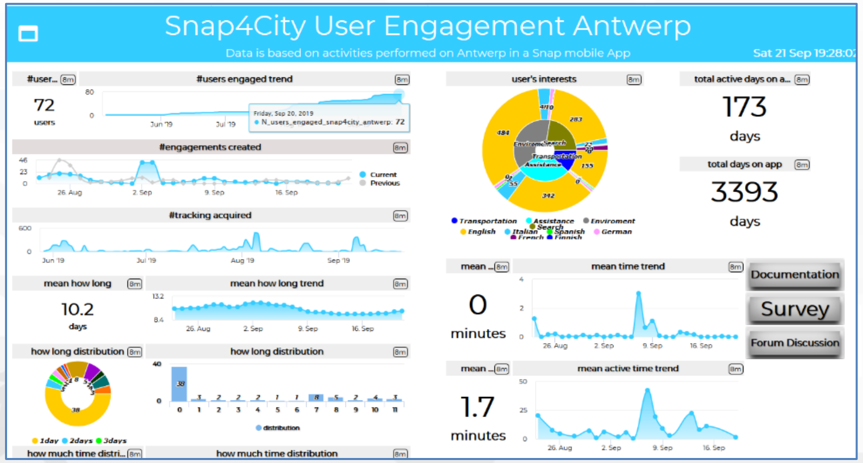

DCS Real Time - Settimanale
STOCK KOH
STOCK NaOH
Home
RTO online

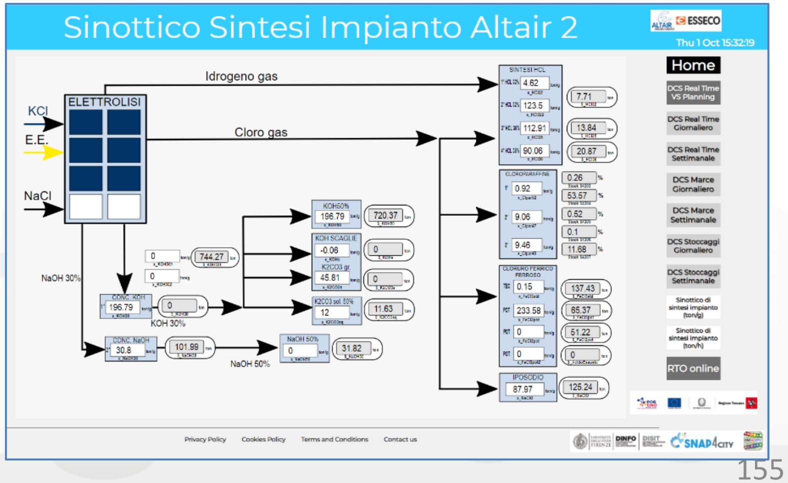

Snap4City
Data Analyzer: DevDash
Open Distro for Elasticsearch


Firenze - Trafair - AirQuality Heatmaps
This dashboard contains data derived from actual sensors and predictive values under validation
Sun 25 Oct 18:52:00
Multi Map
18.6
Firenze Air quality trends
Trafair Main Dashboard

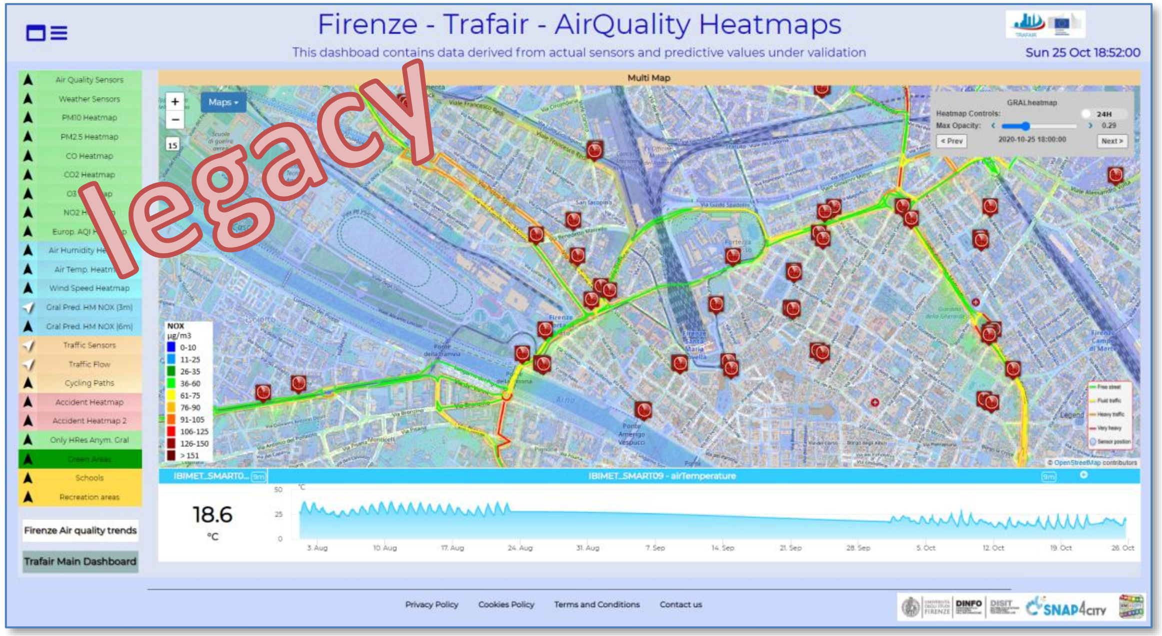

BIM Integration Dashboard
Sat 16 Jan 02:05:21
ALTAIR Plant
Building
Digital Hub


3D Multi Data Map - Digital Twin Global - Firenze
demonstrator
Sun 3 Apr 16:03:52
3D Map
METRO9 - vehicleFlow
192


Snap4City User Engagement Antwerp
Data is based on activities performed on Antwerp in a Snap mobile App
Sat 21 Sep 19:28:02
#users engaged trend
72
users
#engagements created
#tracking acquired
user's interests
total active days on app
173
days
total days on app
3393
days
mean how long
10.2
days
mean how long trend
how long distribution
mean time trend
0
minutes
mean active time trend
1.7
minutes
Documentation
Survey
Forum Discussion


Sinottico Sintesi Impianto Altair 2
Thu 1 Oct 15:32:19
ELETTROLISI
KCl
E.E.
NaCl
Idrogeno gas
Cloro gas
NaOH 30%
KOH 30%
NaOH 50%
Home
RTO online

Different Themes

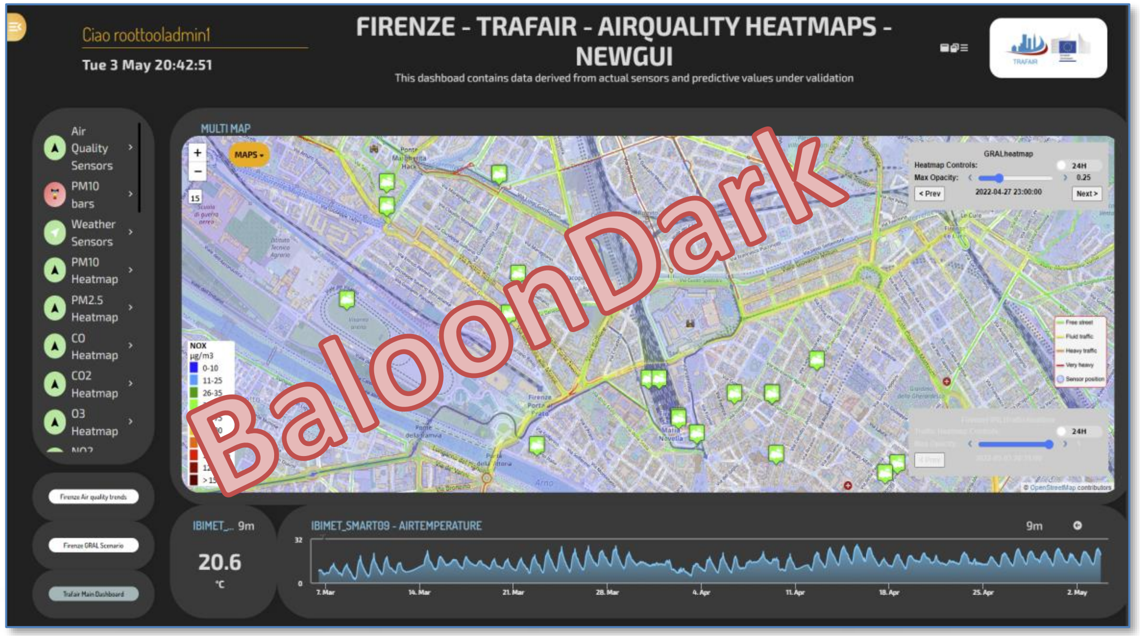


Firenze - Trafair - AirQuality Heatmaps

This dashboard contains data derived from actual sensors and predictive values under validation

Sun 25 Oct 18:52:00

legacy



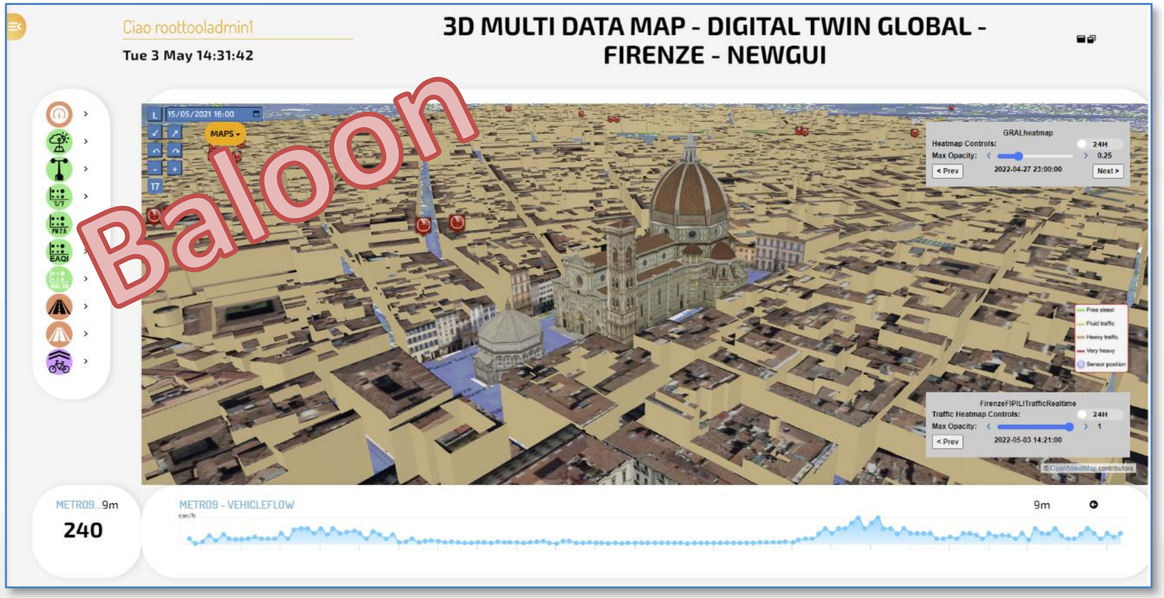
FIRENZE - TRAFAIR - AIRQUALITY HEATMAPS - NEWGUI

This dashboard contains data derived from actual sensors and predictive values under validation

Ciao roottooladmin1

Tue 3 May 20:42:51

BaloonDark

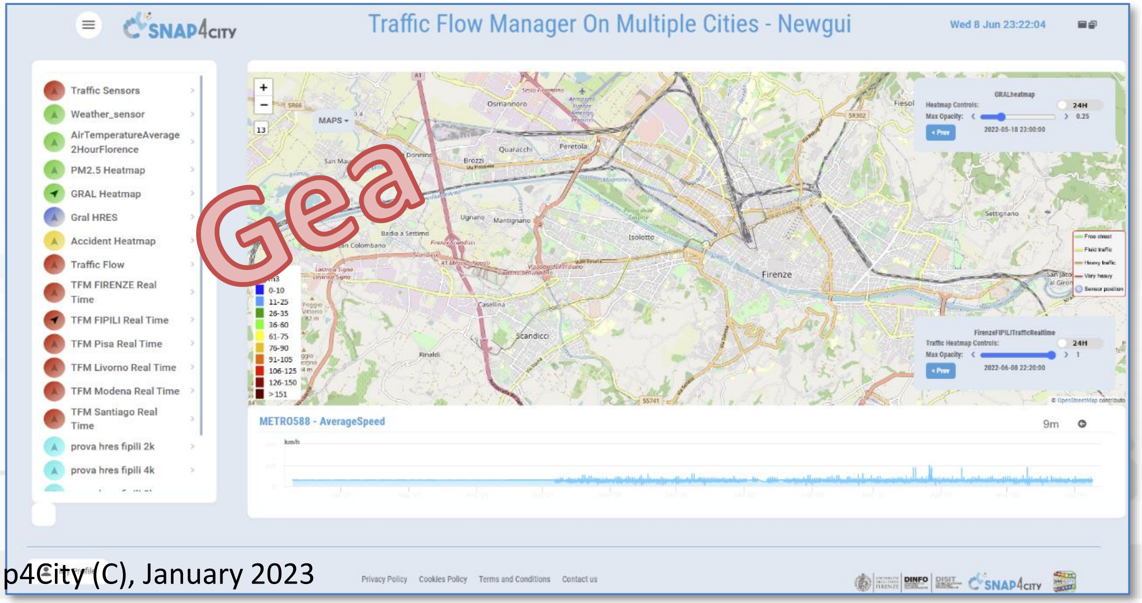


3D MULTI DATA MAP - DIGITAL TWIN GLOBAL - FIRENZE - NEWGUI

Ciao roottooladmin1

Tue 3 May 14:31:42

Baloon

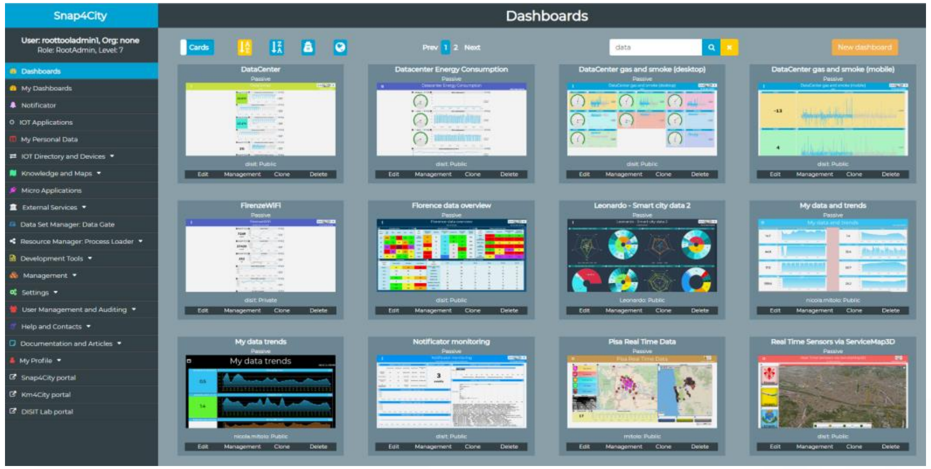


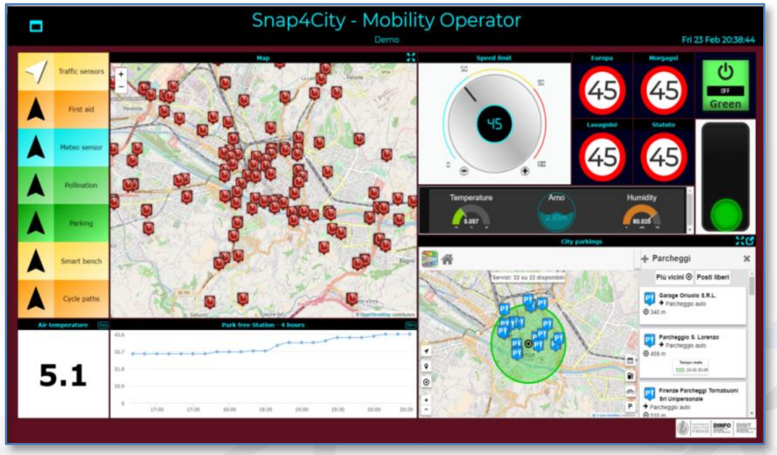
Traffic Flow Manager On Multiple Cities - Newgui

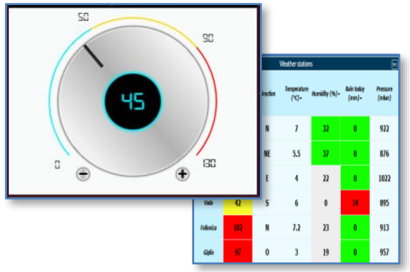
Wed 8 Jun 23:22:04

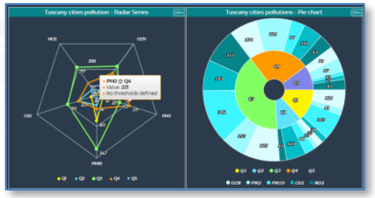
Gea

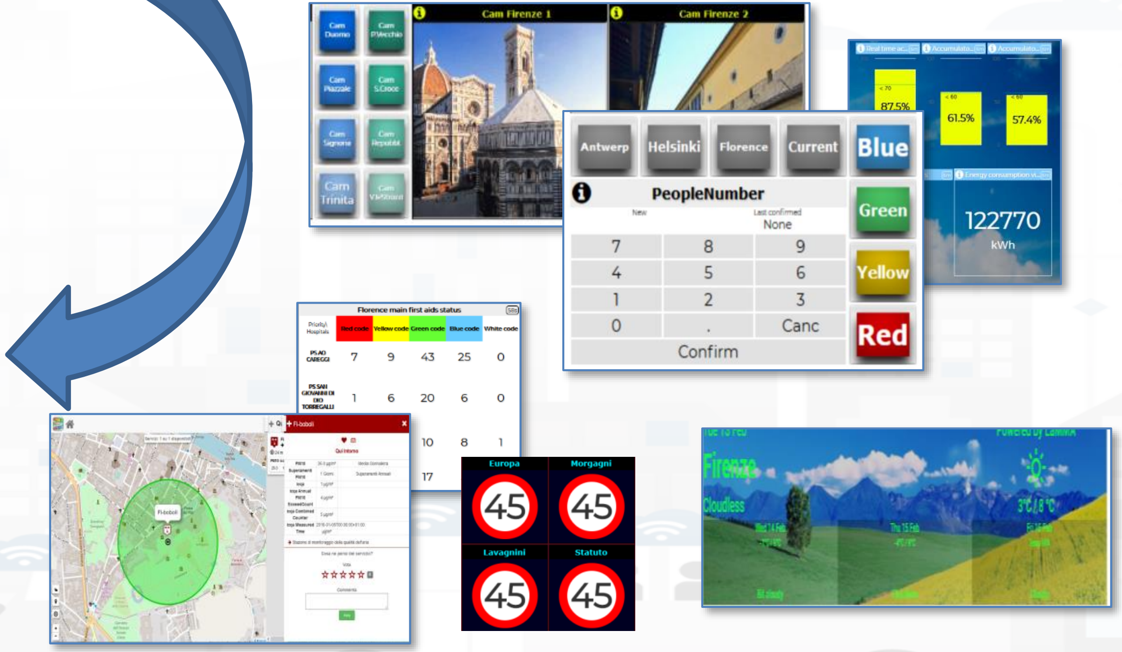
Dashboard List and Editor

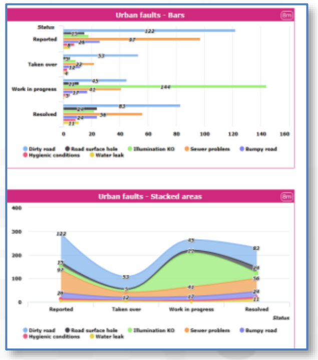




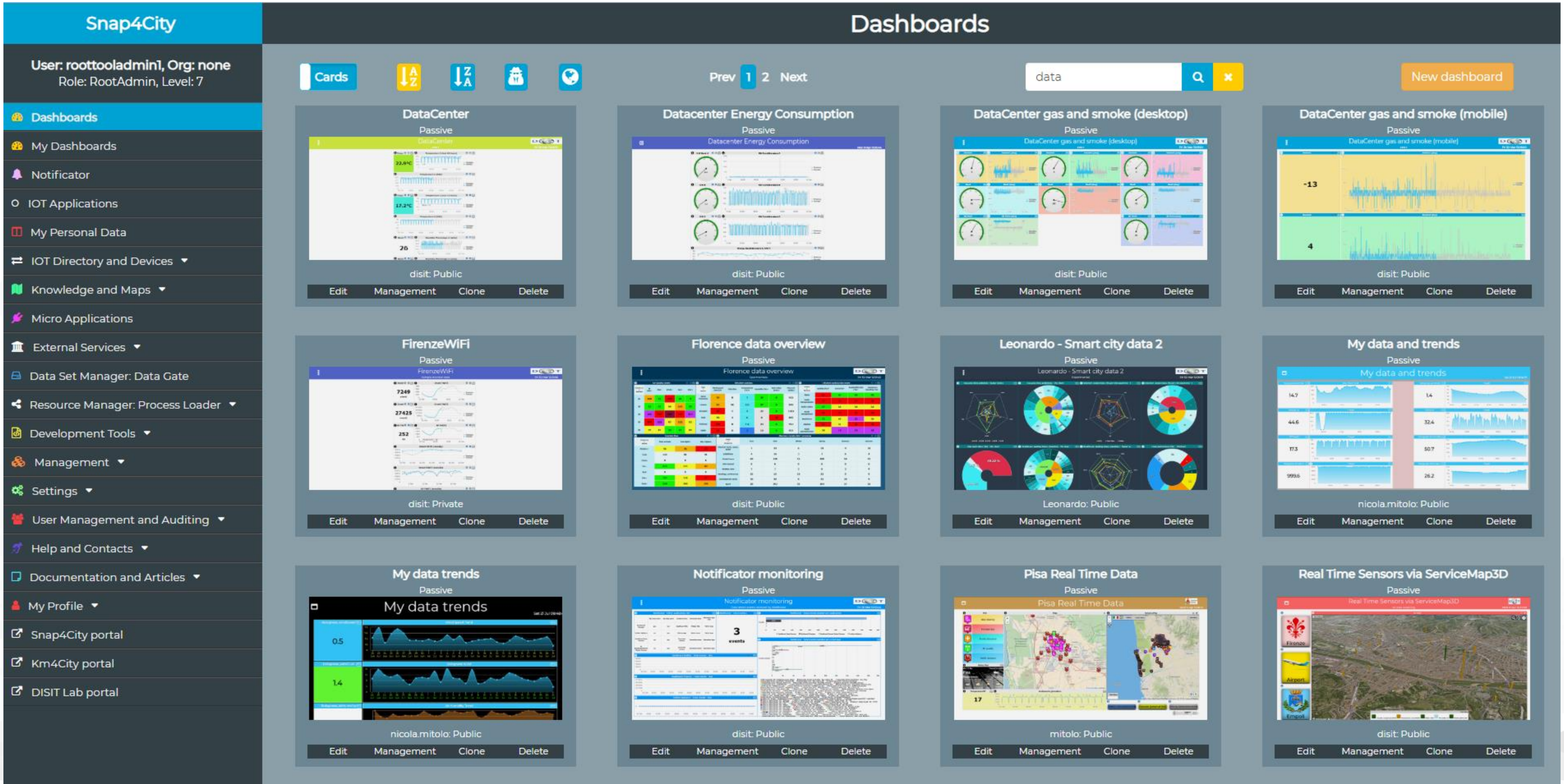



Tuscany cities pollution - Radar Series
Tuscany cities pollutions - Pie chart





Dashboard List and Editor



Snap4City

Dashboards

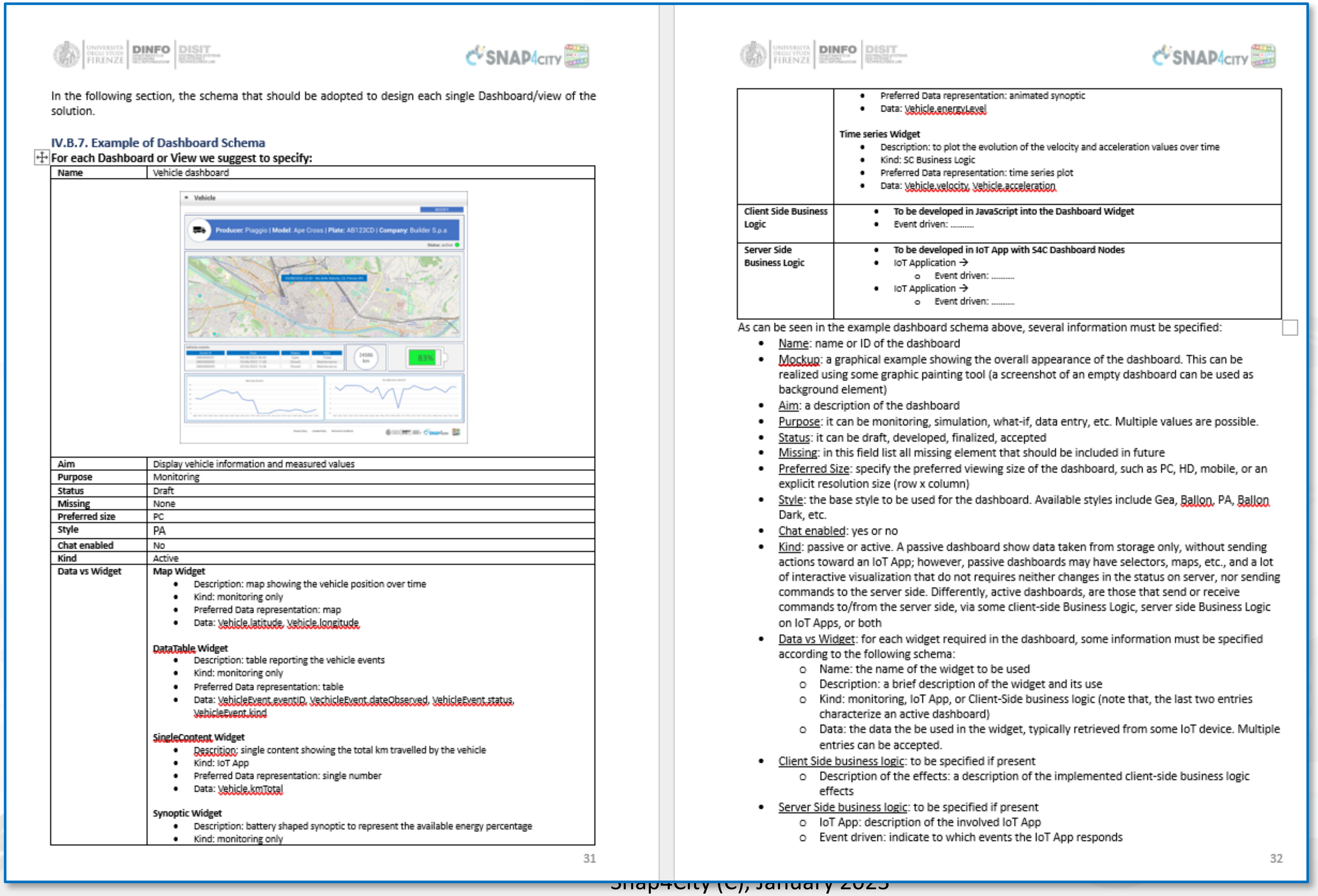
User: roottooladmin1, Org: none
Role: RootAdmin, Level: 7

- Dashboards
- My Dashboards
- Notificator
- IOT Applications
- My Personal Data
- IOT Directory and Devices ▾
- Knowledge and Maps ▾
- Micro Applications
- External Services ▾
- Data Set Manager: Data Gate
- Resource Manager: Process Loader ▾
- Development Tools ▾
- Management ▾
- Settings ▾
- User Management and Auditing ▾
- Help and Contacts ▾
- Documentation and Articles ▾
- My Profile ▾
- Snap4City portal
- Km4City portal
- DISIT Lab portal

Cards | Prev 1 2 Next | data | New dashboard

Title	Type	Owner: visibility	Actions
DataCenter	Passive	disit: Public	Edit Management Clone Delete
Datacenter Energy Consumption	Passive	disit: Public	Edit Management Clone Delete
DataCenter gas and smoke (desktop)	Passive	disit: Public	Edit Management Clone Delete
DataCenter gas and smoke (mobile)	Passive	disit: Public	Edit Management Clone Delete
FirenzeWiFi	Passive	disit: Private	Edit Management Clone Delete
Florence data overview	Passive	disit: Public	Edit Management Clone Delete
Leonardo - Smart city data 2	Passive	Leonardo: Public	Edit Management Clone Delete
My data and trends	Passive	nicola.mitolo: Public	Edit Management Clone Delete
My data trends	Passive	nicola.mitolo: Public	Edit Management Clone Delete
Notificator monitoring	Passive	disit: Public	Edit Management Clone Delete
Pisa Real Time Data	Passive	mitolo: Public	Edit Management Clone Delete
Real Time Sensors via ServiceMap3D	Passive	disit: Public	Edit Management Clone Delete

A Dashboard Design Schema is provided



In the following section, the schema that should be adopted to design each single Dashboard/view of the solution.

IV.B.7. Example of Dashboard Schema

For each Dashboard or View we suggest to specify:

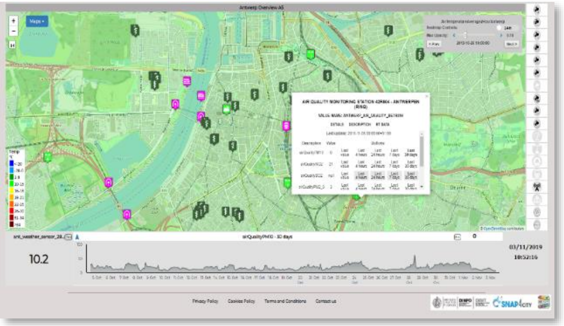
Name	Vehicle dashboard
	Vehicle Producer: Piaggio | Model: Ape Cross | Plate: AB123CD | Company: Builder S.p.a 24586 km 83%
Aim	Display vehicle information and measured values
Purpose	Monitoring
Status	Draft
Missing	None
Preferred size	PC
Style	PA
Chat enabled	No
Kind	Active
Data vs Widget	**Map Widget** • Description: map showing the vehicle position over time • Kind: monitoring only • Preferred Data representation: map • Data: Vehicle.latitude, Vehicle.longitude **DataTable Widget** • Description: table reporting the vehicle events • Kind: monitoring only • Preferred Data representation: table • Data: VehicleEvent.eventID, VechicleEvent.dateObserved, VehicleEvent.status, VehicleEvent.kind **SingleContent Widget** • Descrition: single content showing the total km travelled by the vehicle • Kind: IoT App • Preferred Data representation: single number • Data: Vehicle.kmTotal **Synoptic Widget** • Description: battery shaped synoptic to represent the available energy percentage • Kind: monitoring only • Preferred Data representation: animated synoptic • Data: Vehicle.energyLevel **Time series Widget** • Description: to plot the evolution of the velocity and acceleration values over time • Kind: SC Business Logic • Preferred Data representation: time series plot • Data: Vehicle.velocity, Vehicle.acceleration
Client Side Business Logic	• **To be developed in JavaScript into the Dashboard Widget** • Event driven: ………
Server Side Business Logic	• **To be developed in IoT App with S4C Dashboard Nodes** • IoT Application → ○ Event driven: ………. • IoT Application → ○ Event driven: ……….

As can be seen in the example dashboard schema above, several information must be specified:

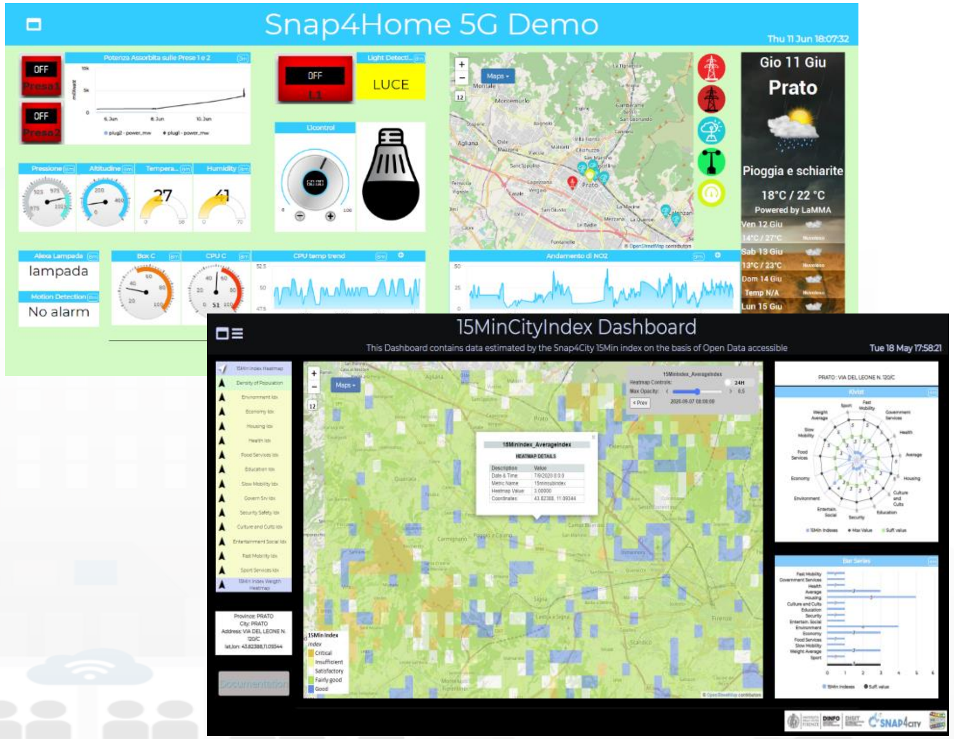
- Name: name or ID of the dashboard
- Mockup: a graphical example showing the overall appearance of the dashboard. This can be realized using some graphic painting tool (a screenshot of an empty dashboard can be used as background element)
- Aim: a description of the dashboard
- Purpose: it can be monitoring, simulation, what-if, data entry, etc. Multiple values are possible.
- Status: it can be draft, developed, finalized, accepted
- Missing: in this field list all missing element that should be included in future
- Preferred Size: specify the preferred viewing size of the dashboard, such as PC, HD, mobile, or an explicit resolution size (row x column)
- Style: the base style to be used for the dashboard. Available styles include Gea, Ballon, PA, Ballon Dark, etc.
- Chat enabled: yes or no
- Kind: passive or active. A passive dashboard show data taken from storage only, without sending actions toward an IoT App; however, passive dashboards may have selectors, maps, etc., and a lot of interactive visualization that do not requires neither changes in the status on server, nor sending commands to the server side. Differently, active dashboards, are those that send or receive commands to/from the server side, via some client-side Business Logic, server side Business Logic on IoT Apps, or both
- Data vs Widget: for each widget required in the dashboard, some information must be specified according to the following schema:
  - Name: the name of the widget to be used
  - Description: a brief description of the widget and its use
  - Kind: monitoring, IoT App, or Client-Side business logic (note that, the last two entries characterize an active dashboard)
  - Data: the data the be used in the widget, typically retrieved from some IoT device. Multiple entries can be accepted.
- Client Side business logic: to be specified if present
  - Description of the effects: a description of the implemented client-side business logic effects
- Server Side business logic: to be specified if present
  - IoT App: description of the involved IoT App
  - Event driven: indicate to which events the IoT App responds

Dashboard Kind

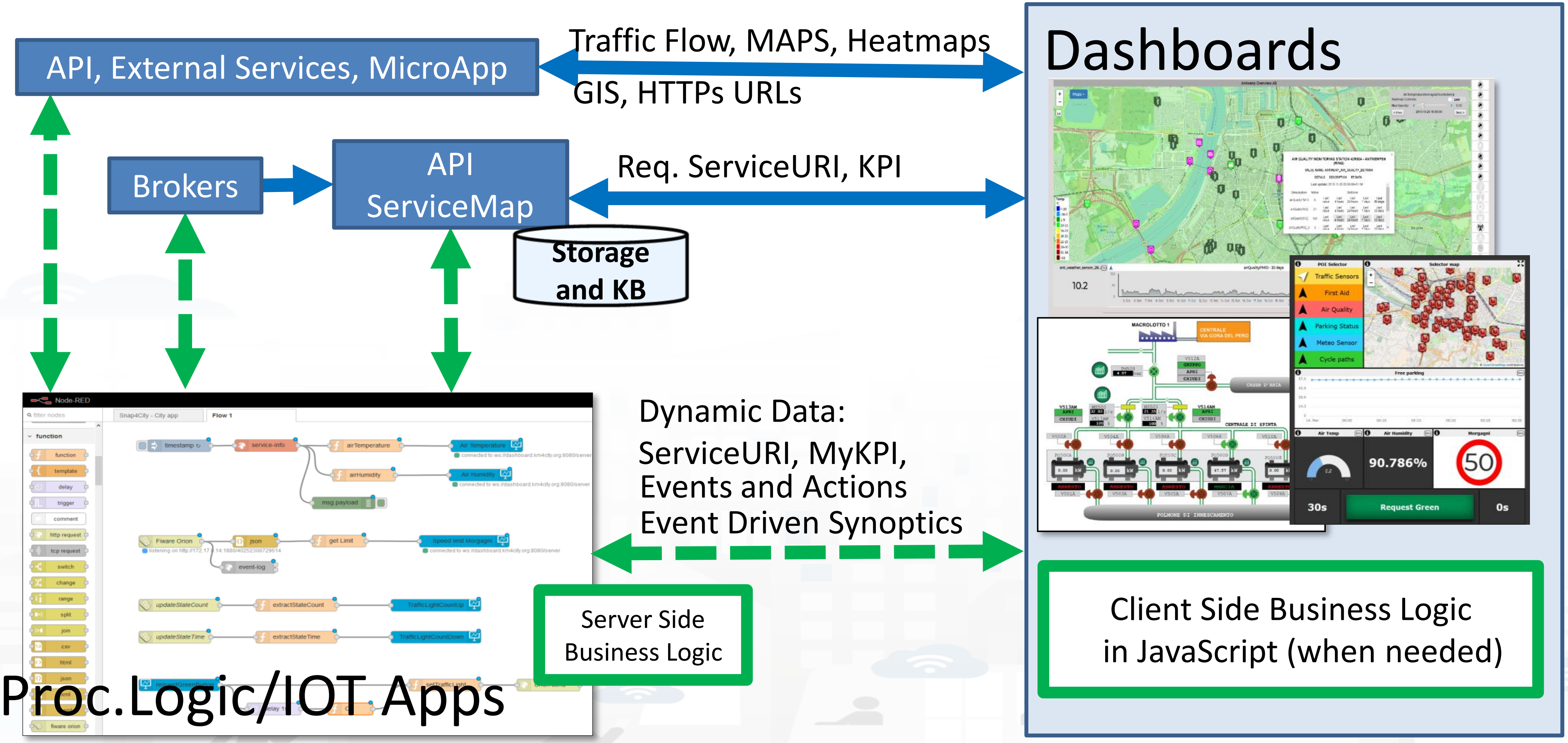
- **Passive Dashboards**: showing data taken from Storage only, no actions toward IoT App
  - Passive dashboards may have Selectors, maps, etc., and a lot of visualization without changing the status on Server, no sending commands to the Server Side.



- **Active Dashboards, which can be those sending or receiving commands to/from the logic coded somehow and in particular for**
  - **Server Side Business Logic →** logic on IoT Apps with Snap4City Dashboard Nodes, which is easier to be programmed begin based on Node-RED visual programming.
  - **Client Side Business Logic →** logic on JavaScript on specific Dashboard Widgets only for skilled developers of Snap4City Platform. We suggest first prototype by using Server Side Business Logic, then pass to Client Side Business Logic in JavaScript.
  - Both kind of Business Logics may be active on the same Active Dashboard.



How the Dashboards exchange data



API, External Services, MicroApp

Traffic Flow, MAPS, Heatmaps

GIS, HTTPs URLs

Brokers

API ServiceMap

Storage and KB

Req. ServiceURI, KPI

Dashboards

Dynamic Data:

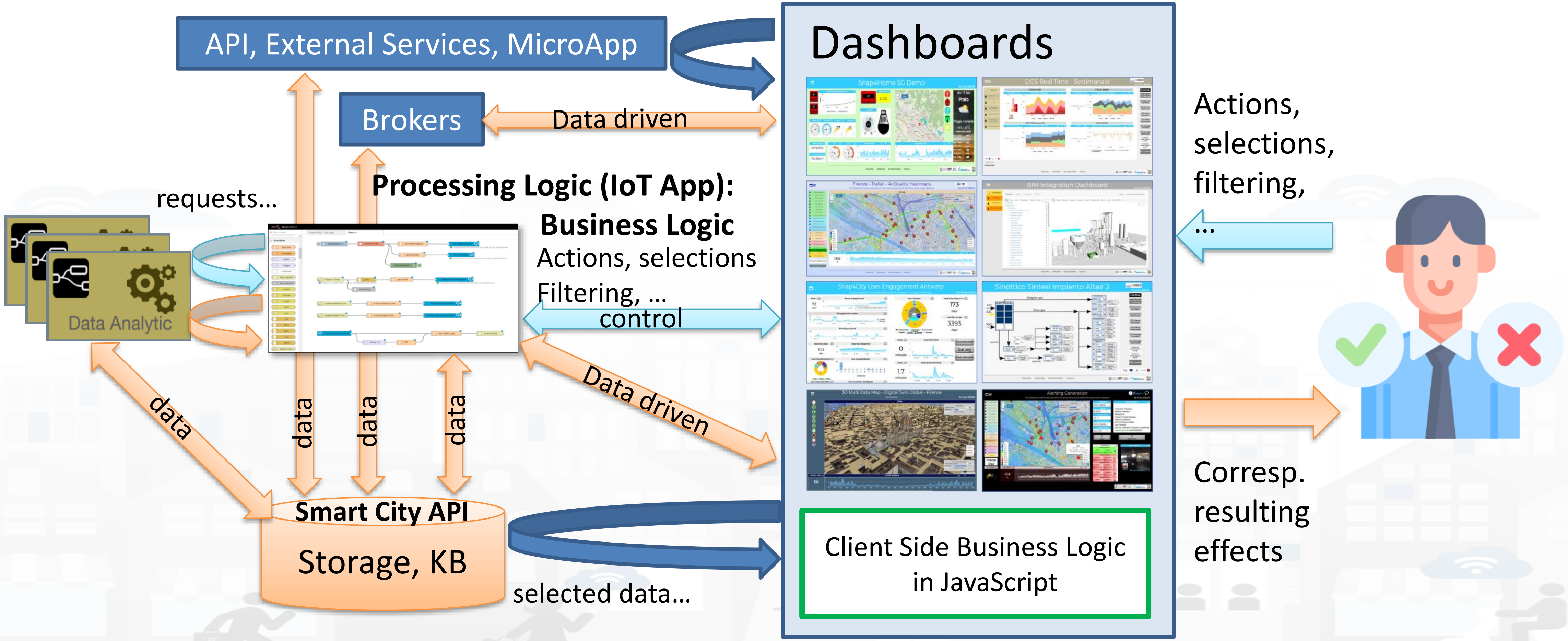
ServiceURI, MyKPI, Events and Actions Event Driven Synoptics

Server Side Business Logic

Client Side Business Logic in JavaScript (when needed)

Proc.Logic/IOT Apps

How the Dashboards exchange data/controls



API, External Services, MicroApp

Brokers

Data driven

Dashboards

Actions, selections, filtering, …

requests…

Processing Logic (IoT App): Business Logic

Actions, selections Filtering, …

control

Data Analytic

data

data

data

data

Data driven

Smart City API

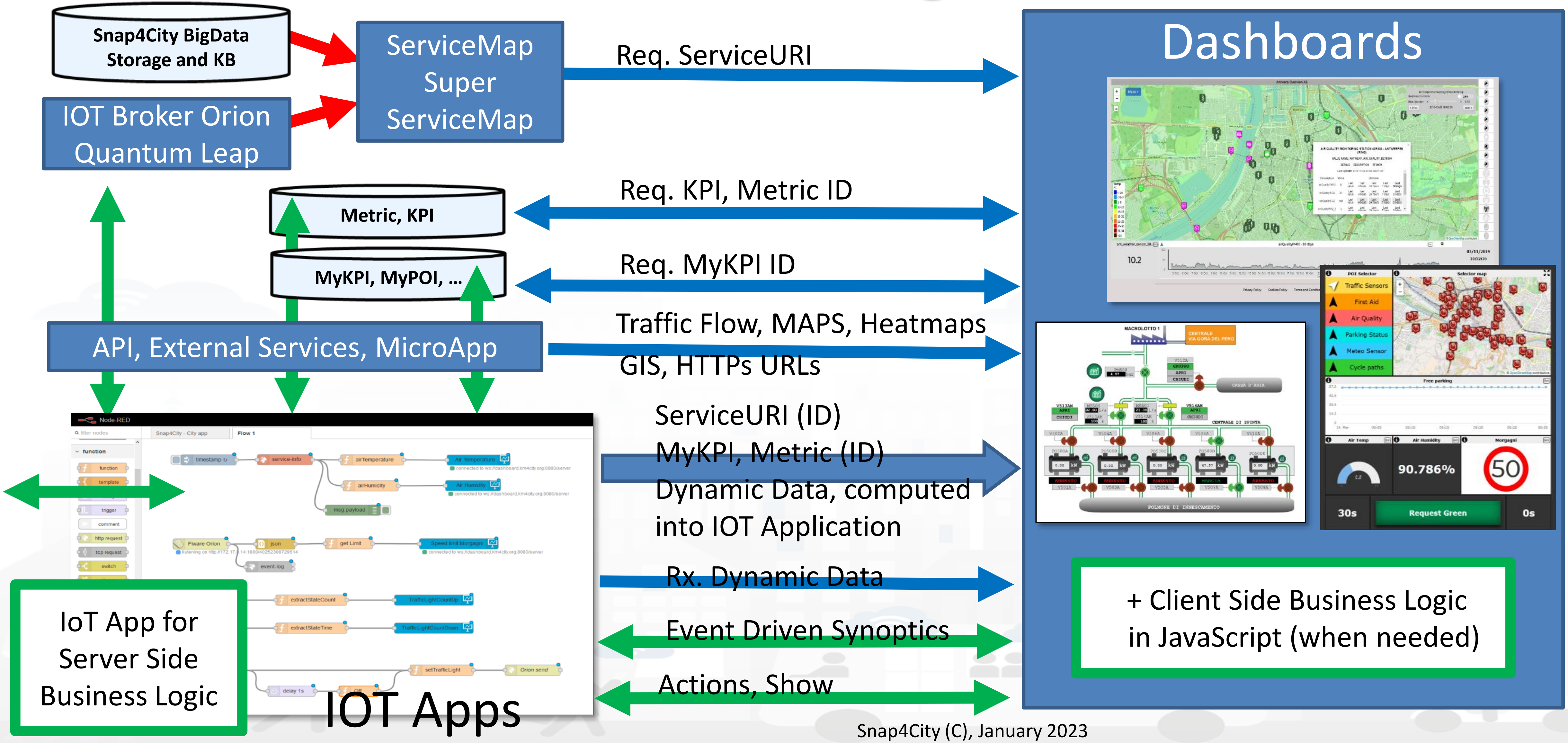
Storage, KB

selected data…

Client Side Business Logic in JavaScript

Corresp. resulting effects

How the Dashboards exchange data



Snap4City BigData Storage and KB

IOT Broker Orion Quantum Leap

ServiceMap Super ServiceMap

Req. ServiceURI

Dashboards

Metric, KPI

MyKPI, MyPOI, …

Req. KPI, Metric ID

Req. MyKPI ID

API, External Services, MicroApp

Traffic Flow, MAPS, Heatmaps GIS, HTTPs URLs

ServiceURI (ID)
MyKPI, Metric (ID)
Dynamic Data, computed into IOT Application

Rx. Dynamic Data

Event Driven Synoptics

Actions, Show

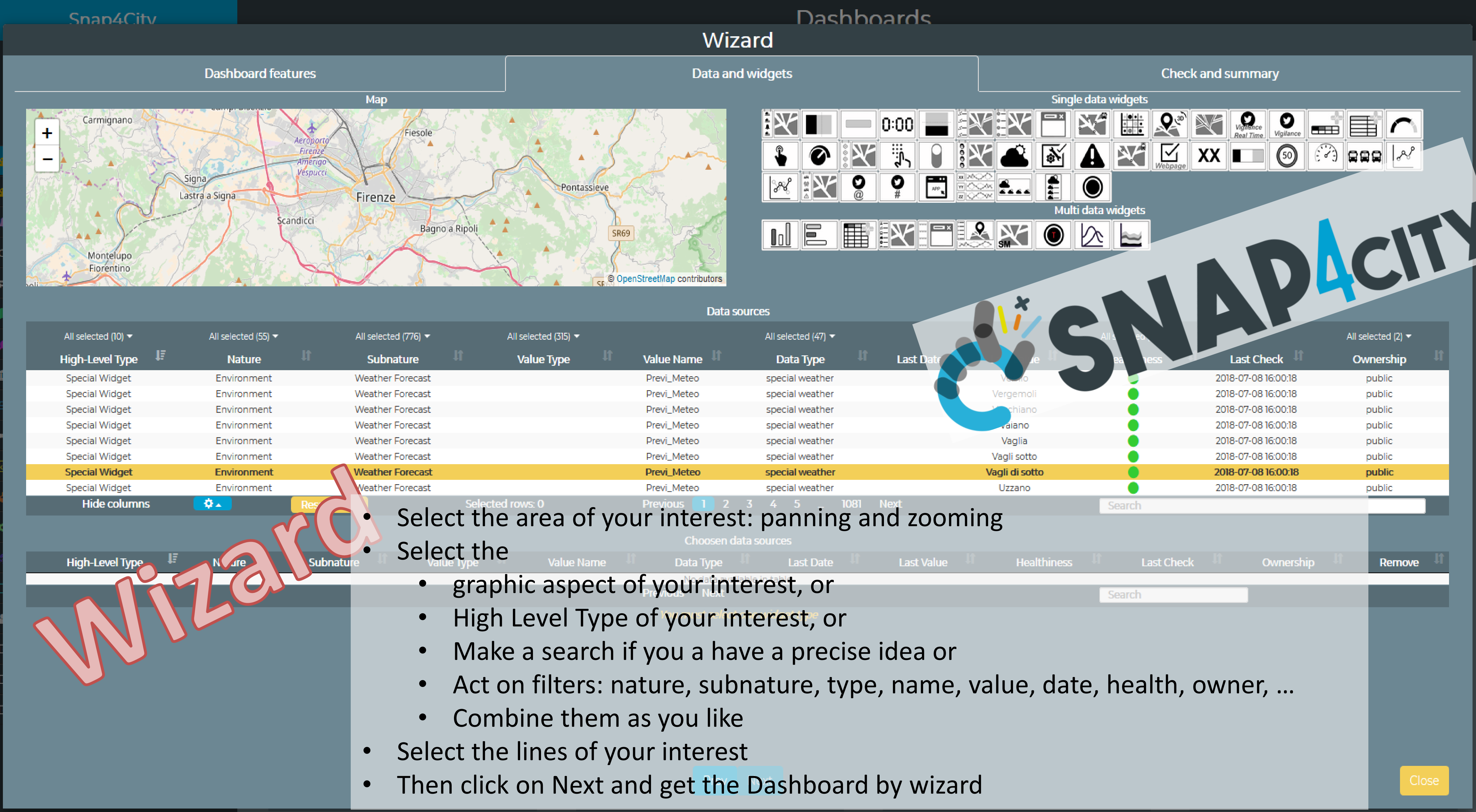
IoT App for Server Side Business Logic

IOT Apps

+ Client Side Business Logic in JavaScript (when needed)

TOP

Develop: via Dashboard Wizard



Data sources

All selected (10) ▾ High-Level Type	All selected (55) ▾ Nature	All selected (776) ▾ Subnature	All selected (315) ▾ Value Type	Value Name	All selected (47) ▾ Data Type	Last Date	Last Value	Healthiness	Last Check	All selected (2) ▾ Ownership
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		[illegible]	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vergemoli	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		[illegible]chiano	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vaiano	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vaglia	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vagli sotto	●	2018-07-08 16:00:18	public
Special Widget	**Environment**	**Weather Forecast**		**Previ_Meteo**	**special weather**		**Vagli di sotto**	●	**2018-07-08 16:00:18**	**public**
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Uzzano	●	2018-07-08 16:00:18	public

Hide columns ⚙ Res[illegible] Selected rows: 0 Previous 1 2 3 4 5 … 1081 Next Search

Choosen data sources

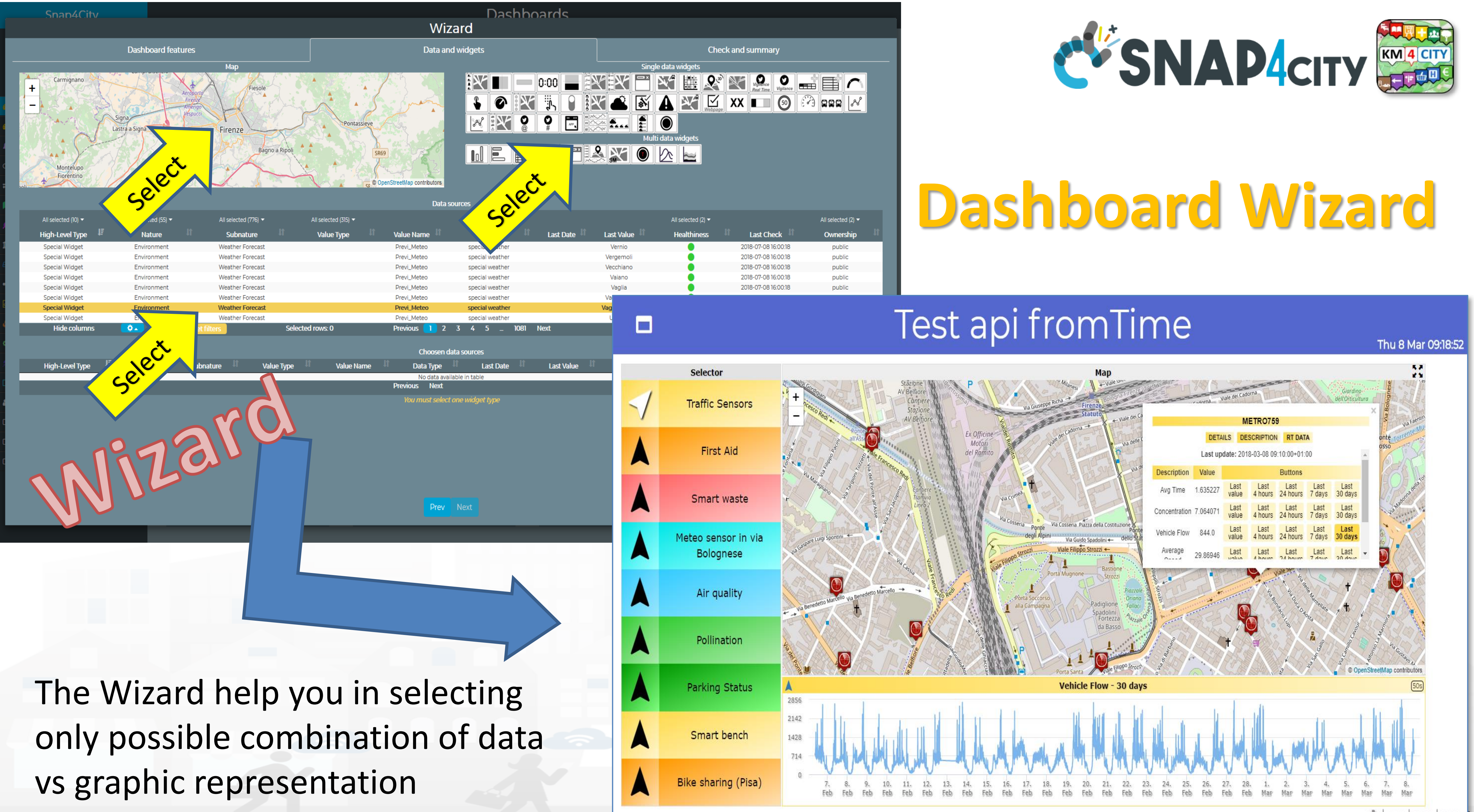
High-Level Type	Nature	Subnature	Value Type	Value Name	Data Type	Last Date	Last Value	Healthiness	Last Check	Ownership	Remove

Previous Next Search

Wizard

- Select the area of your interest: panning and zooming
- Select the
  - graphic aspect of your interest, or
  - High Level Type of your interest, or
  - Make a search if you a have a precise idea or
  - Act on filters: nature, subnature, type, name, value, date, health, owner, …
  - Combine them as you like
- Select the lines of your interest
- Then click on Next and get the Dashboard by wizard

Dashboard Wizard



Data sources

High-Level Type	Nature	Subnature	Value Type	Value Name		Last Date	Last Value	Healthiness	Last Check	Ownership
All selected (10)	…ted (55)	All selected (776)	All selected (315)					All selected (2)		All selected (2)
Special Widget	Environment	Weather Forecast		Previ_Meteo	speci…ther		Vernio	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vergemoli	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vecchiano	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vaiano	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vaglia	●	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Va			
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		Vag			
Special Widget	E	Weather Forecast		Previ_Meteo	special weather		U			

Selected rows: 0

Choosen data sources

High-Level Type		…ubnature	Value Type	Value Name	Data Type	Last Date	Last Value
No data available in table							

You must select one widget type

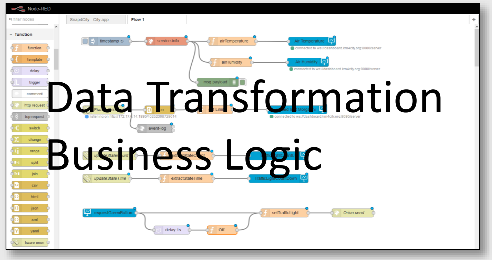
METRO759

Last update: 2018-03-08 09:10:00+01:00

Description	Value	Buttons				
Avg Time	1.635227	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 days
Concentration	7.064071	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 days
Vehicle Flow	844.0	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 days
Average Speed	29.86946	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 days

The Wizard help you in selecting only possible combination of data vs graphic representation

Dashboard Builder: Development



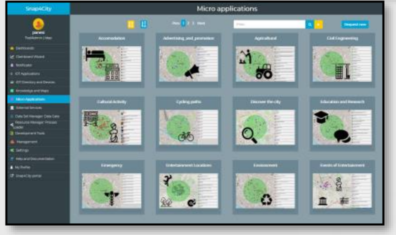
Data Transformation
Business Logic

IOT Applications

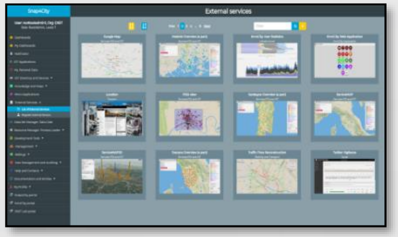
Knowledge Base,
Km4City

Knowledge and Storage Data from the Field and City + MyKPI ++

Widget Collection



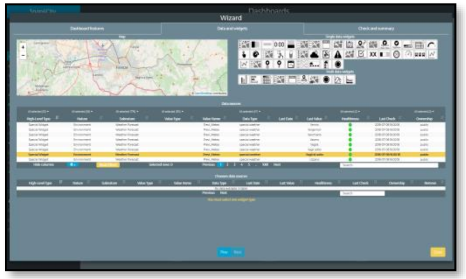
Micro Applications



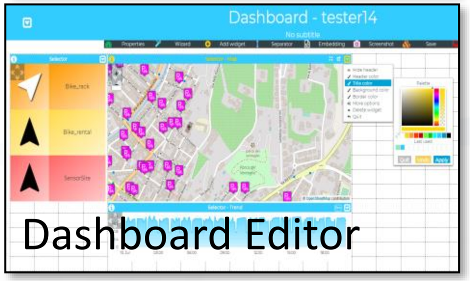
External Services

Custom Widgets/ Synoptics

Dashboard Wizard



Create, save, load, delegate, grant access, change ownerhip

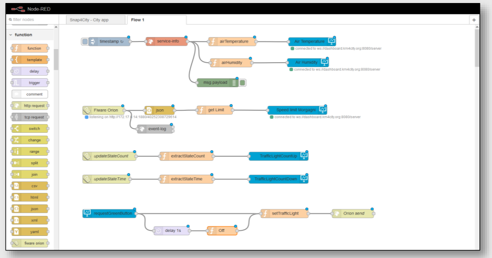


Dashboard Editor

Public Dashboard Collection

My Own Dash/App

Custom Widget / Synoptic / PIN Development



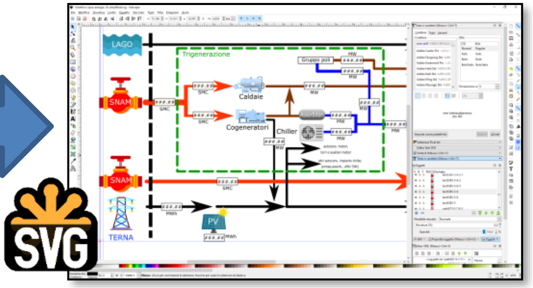
IOT Applications

Knowledge and Storage Data from the Field and City

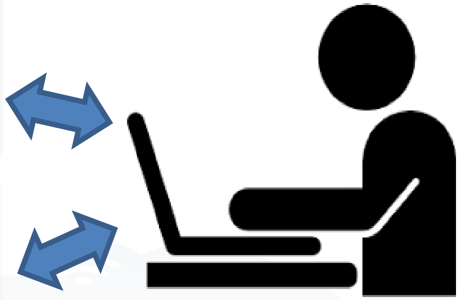
SVG Symbols Collection

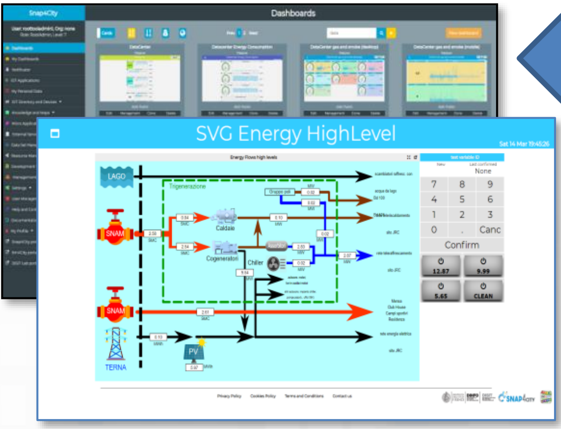
0.04
SMC
49.25 kW
SVG

Inkscape editor on your computer



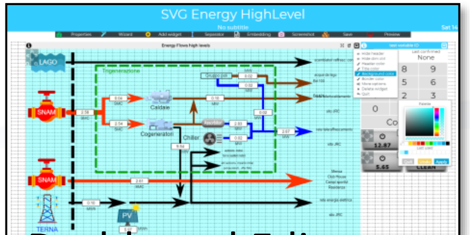
Create, save a Custom Widget in SVG





Public Dashboard Collection

My Own Dash/App



Dashboard Editor

Create, save, load, delegate, grant access

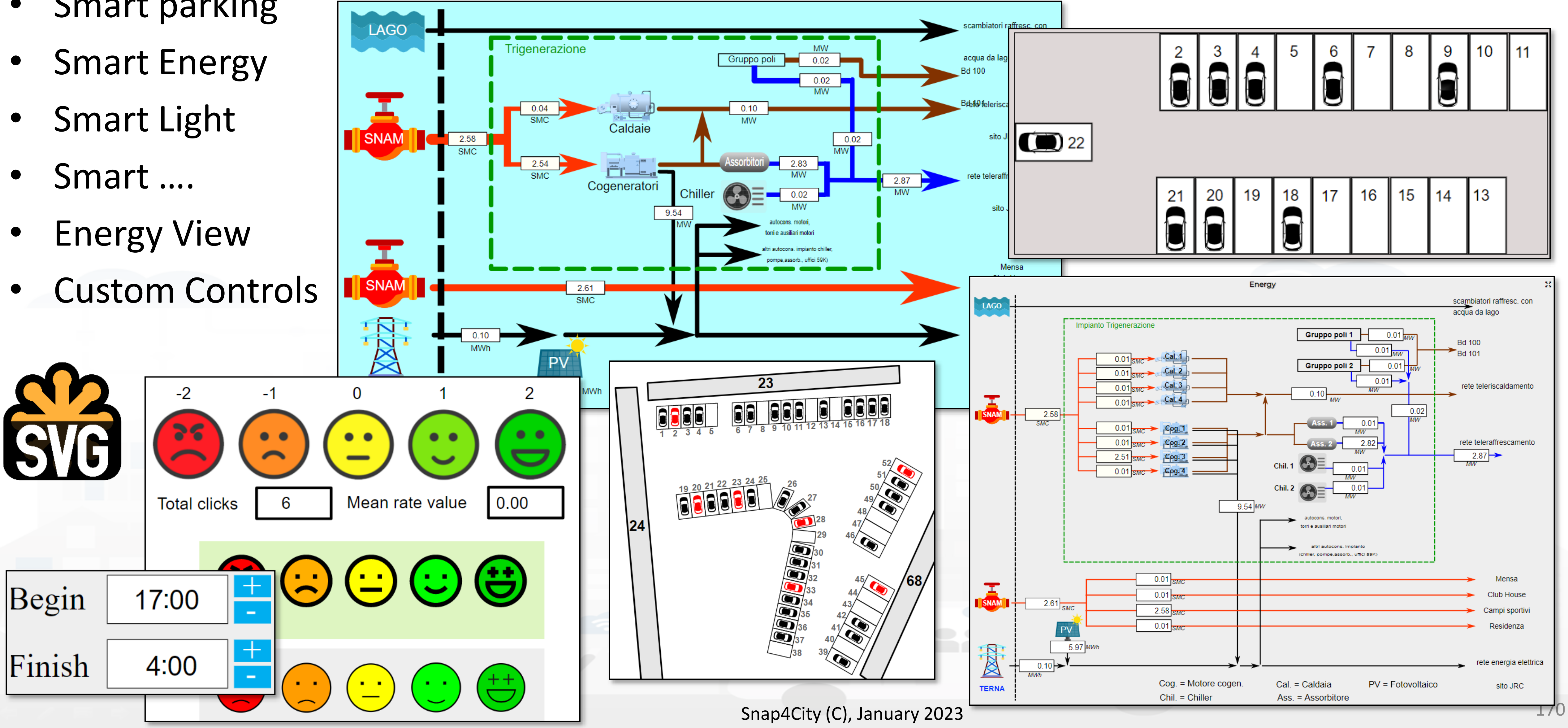
1. Create and Load a Custom SVG
2. Select/Reuse an SVG
3. Make and Instance of Synoptic by Associate Variables with MyKPI
4. Create on Dashboard a Widget based on Synoptic HLT such as Ext. Srv.:
   - https://www.snap4city.org/synoptic/v2/synoptic.html?id=xxxx

TOP

Develop: Dashboards with Synoptics

Special Custom Widgets

- Smart parking
- Smart Energy
- Smart Light
- Smart ….
- Energy View
- Custom Controls



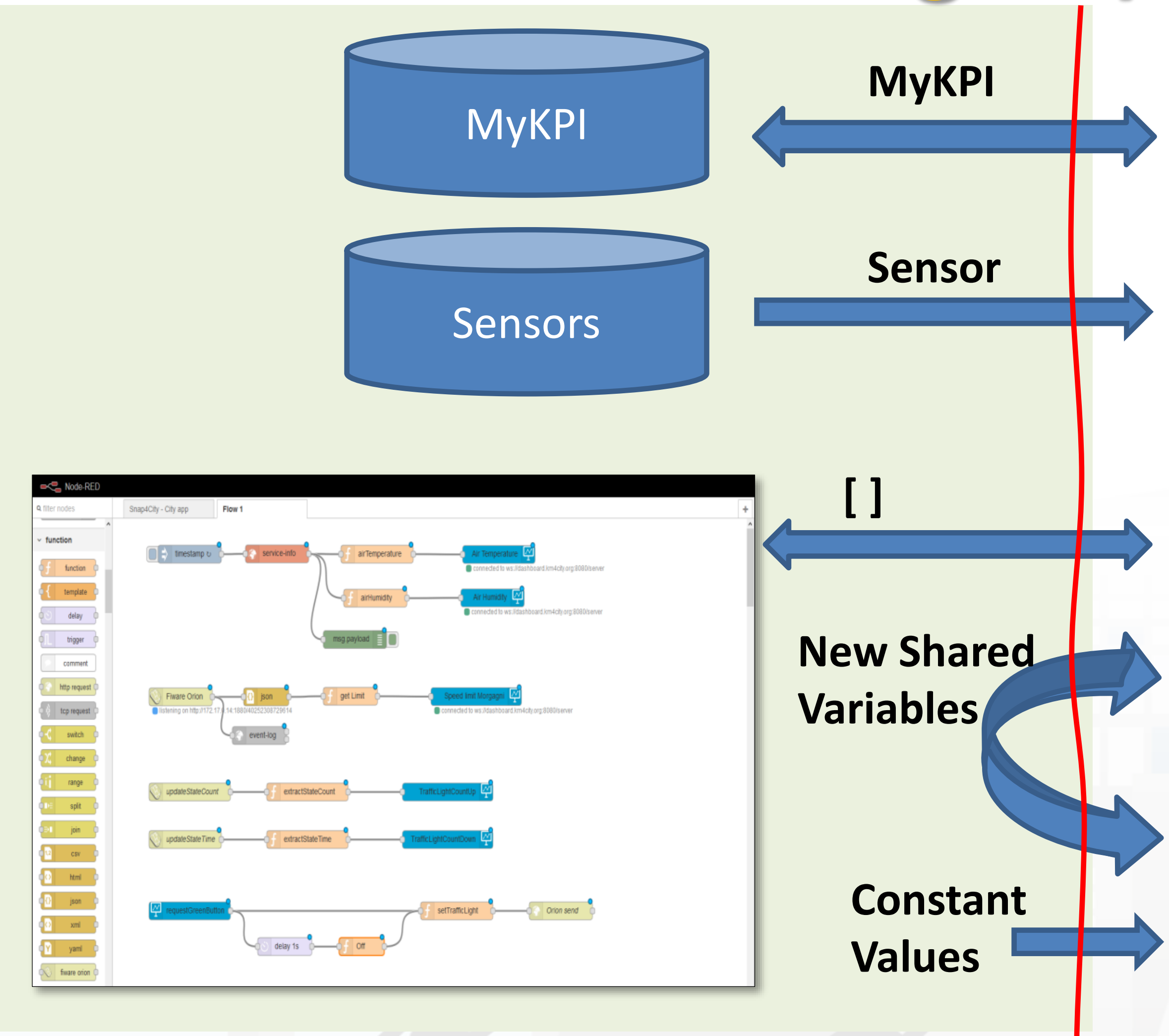
Total clicks 6

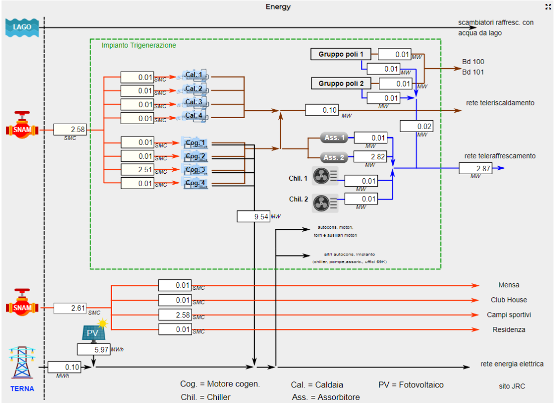
Mean rate value 0.00

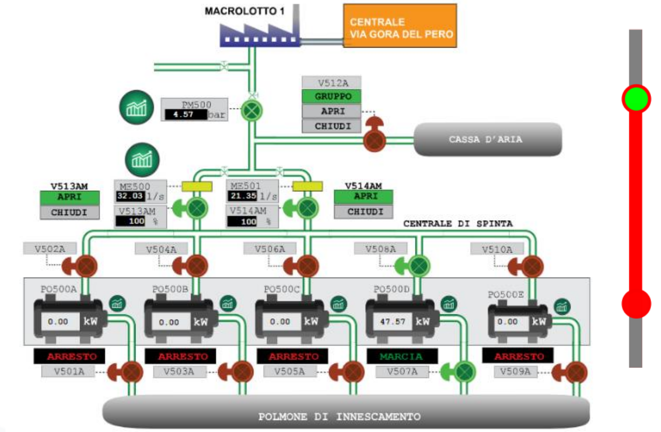
Begin 17:00 + -

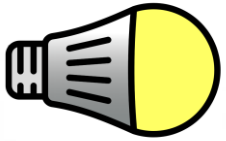
Finish 4:00 + -

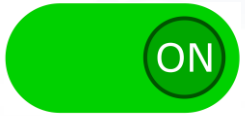
From-To Custom Widgets / Synoptics to Storage in WS

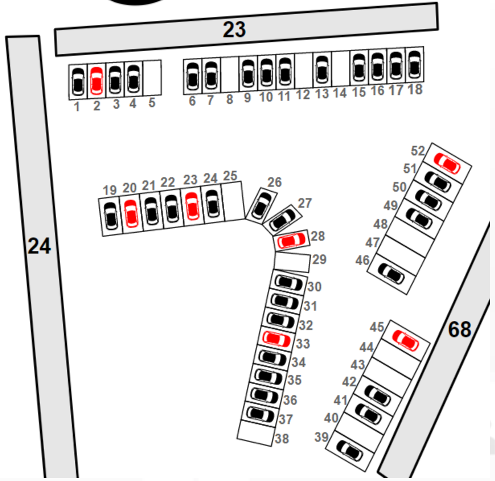


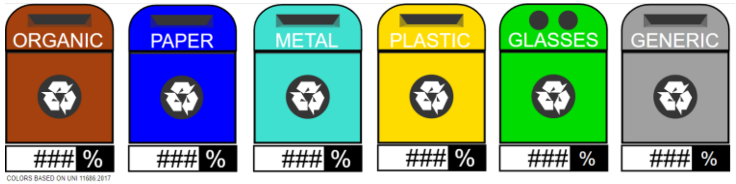


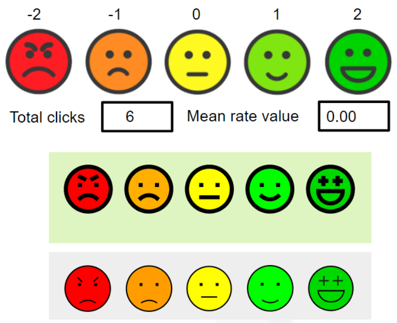


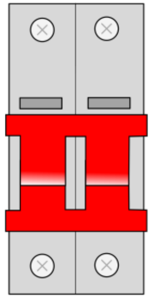


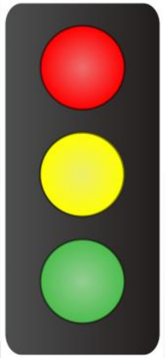








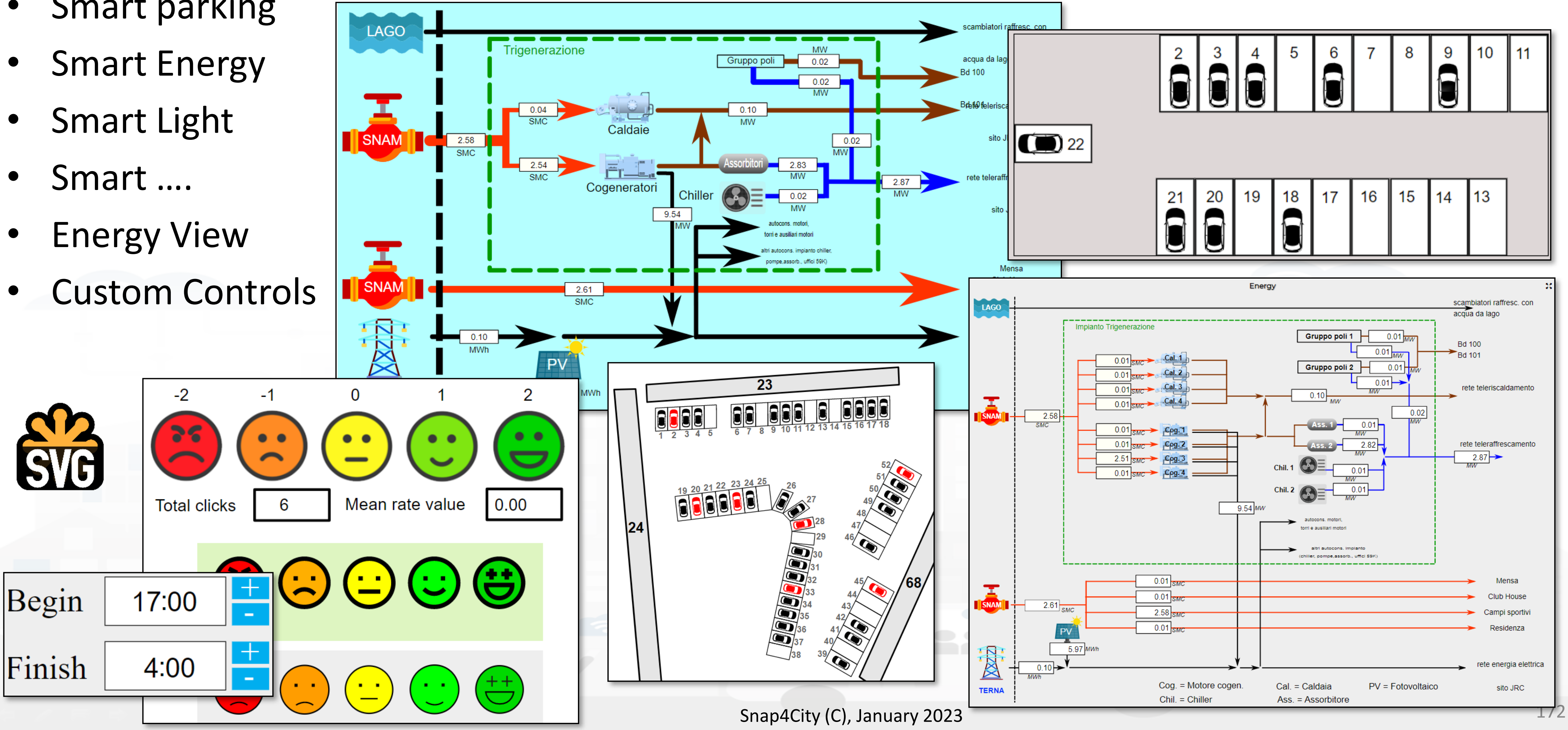




Web Socket Secure

Special Custom Widgets

- Smart parking
- Smart Energy
- Smart Light
- Smart ….
- Energy View
- Custom Controls



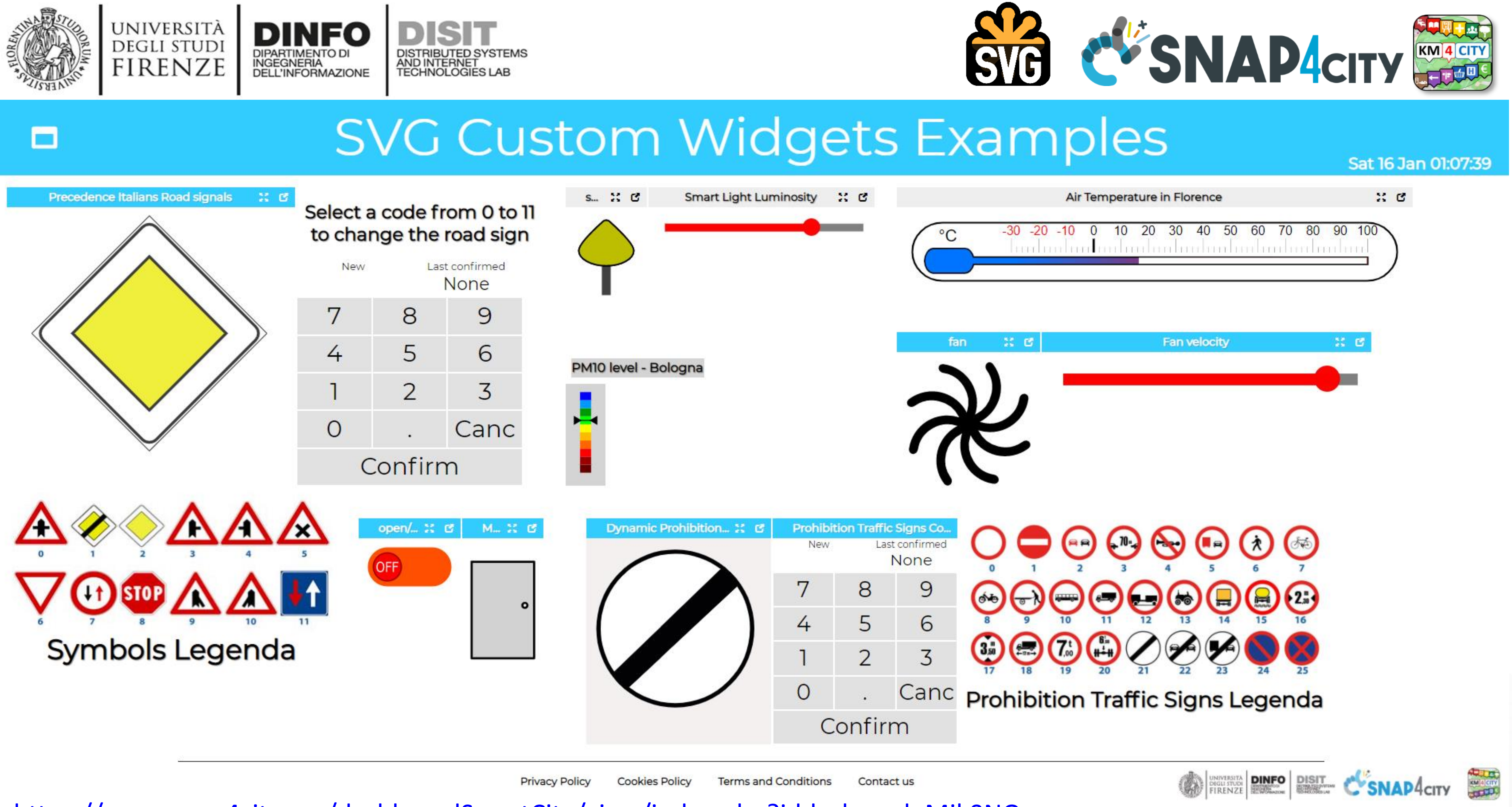
Total clicks 6

Mean rate value 0.00

Begin 17:00 + -

Finish 4:00 + -

Snap4City (C), January 2023

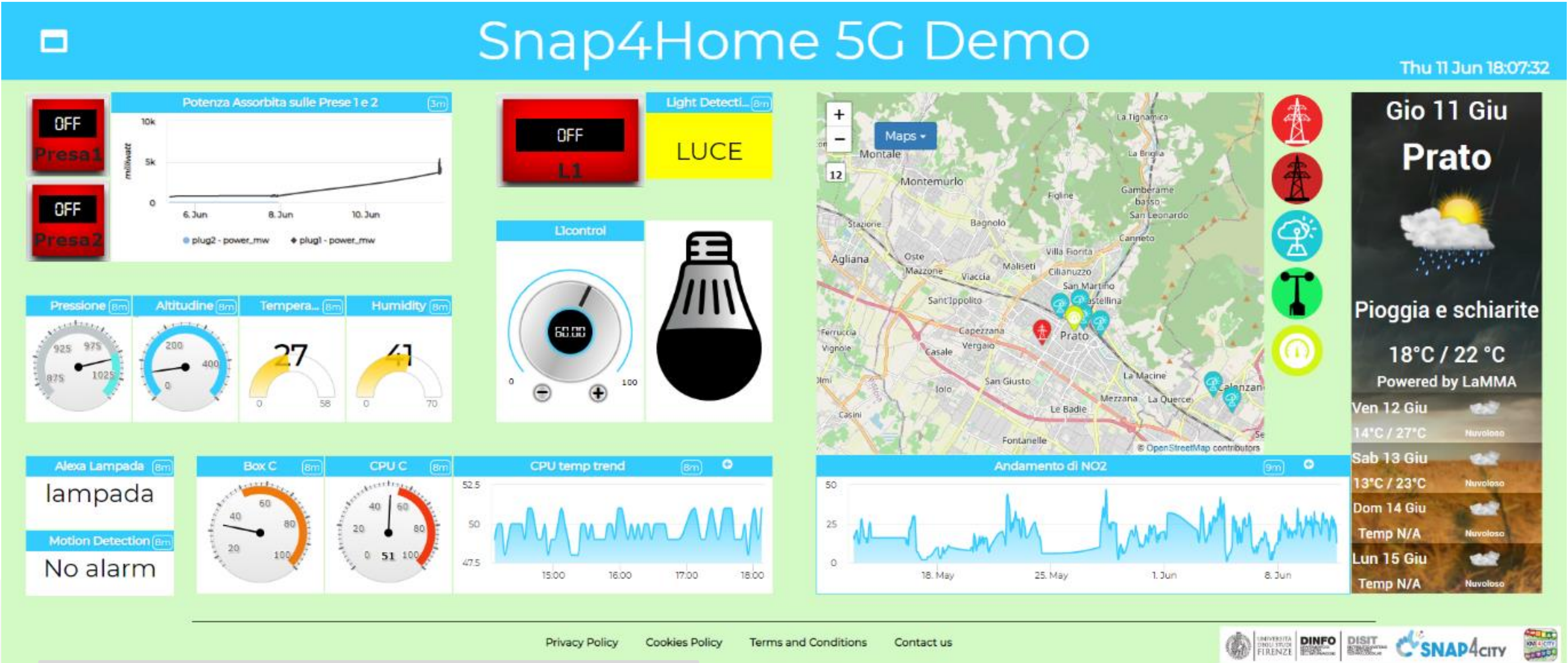


SVG Custom Widgets Examples

Sat 16 Jan 01:07:39

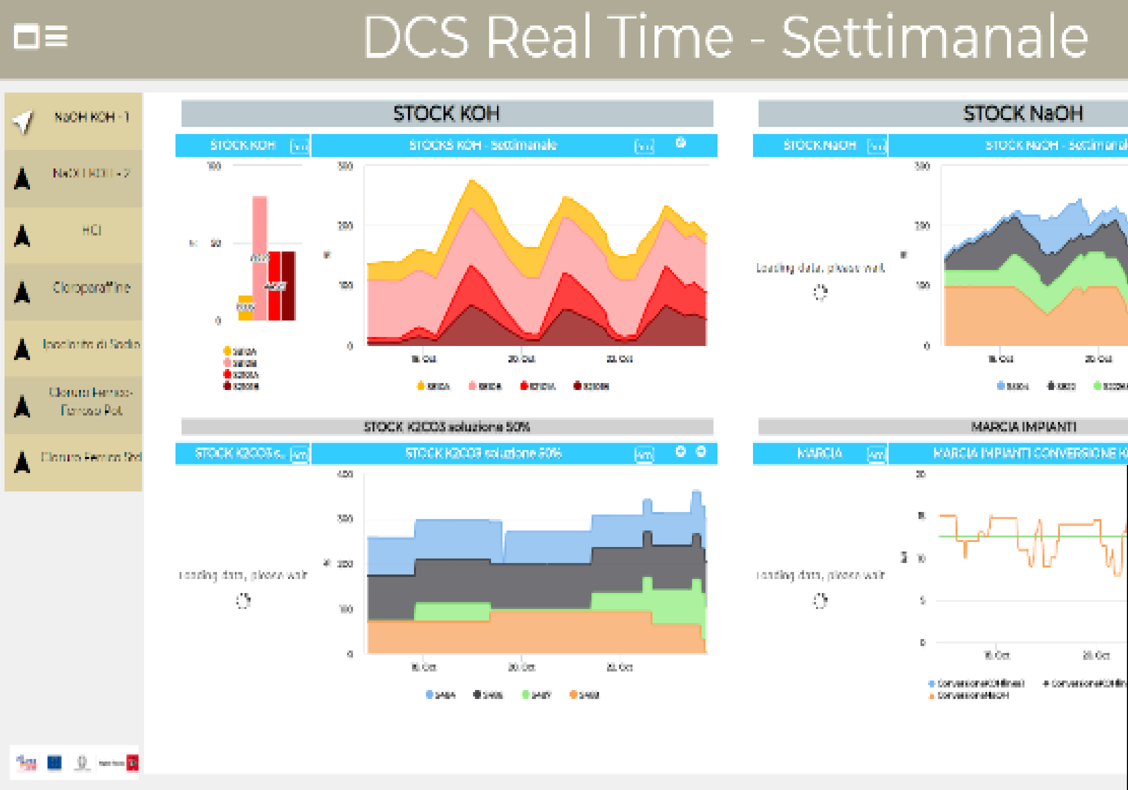
Select a code from 0 to 11 to change the road sign

https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=Mjk0NQ==

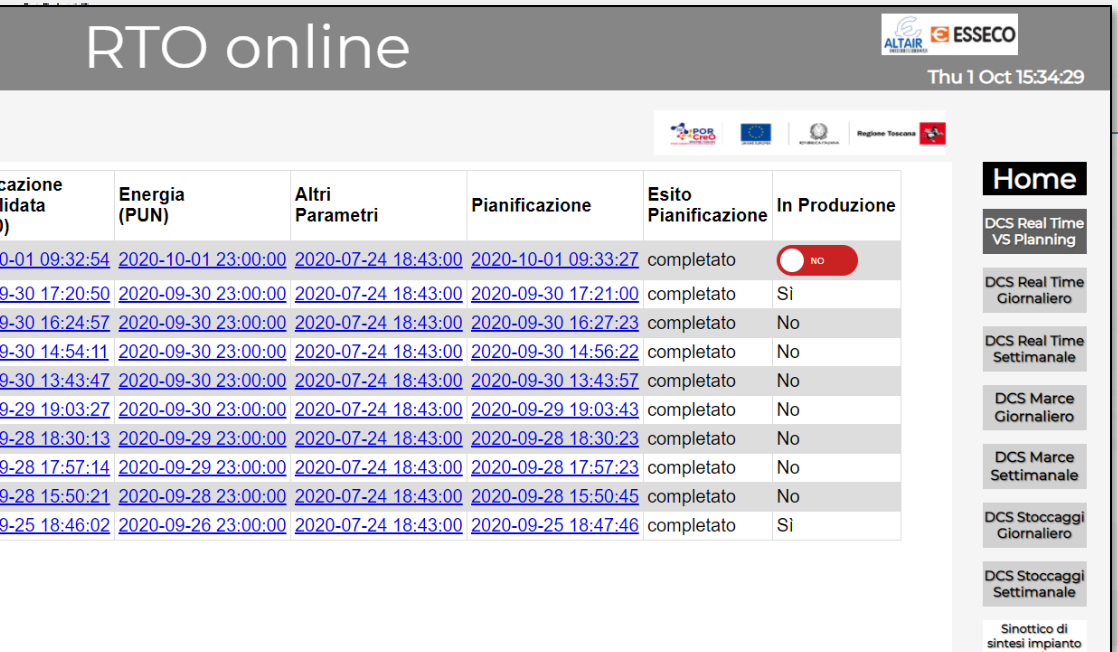


Snap4Home 5G Demo

open fiber WIND 3 ZTE estra



RTO online



ALTAIR ESSECO

Thu 1 Oct 15:34:29

cazione lidata)	Energia (PUN)	Altri Parametri	Pianificazione	Esito Pianificazione	In Produzione
0-01 09:32:54	2020-10-01 23:00:00	2020-07-24 18:43:00	2020-10-01 09:33:27	completato	NO
9-30 17:20:50	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 17:21:00	completato	Sì
9-30 16:24:57	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 16:27:23	completato	No
9-30 14:54:11	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 14:56:22	completato	No
9-30 13:43:47	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 13:43:57	completato	No
9-29 19:03:27	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-29 19:03:43	completato	No
9-28 18:30:13	2020-09-29 23:00:00	2020-07-24 18:43:00	2020-09-28 18:30:23	completato	No
9-28 17:57:14	2020-09-29 23:00:00	2020-07-24 18:43:00	2020-09-28 17:57:23	completato	No
9-28 15:50:21	2020-09-28 23:00:00	2020-07-24 18:43:00	2020-09-28 15:50:45	completato	No
9-25 18:46:02	2020-09-26 23:00:00	2020-07-24 18:43:00	2020-09-25 18:47:46	completato	Sì

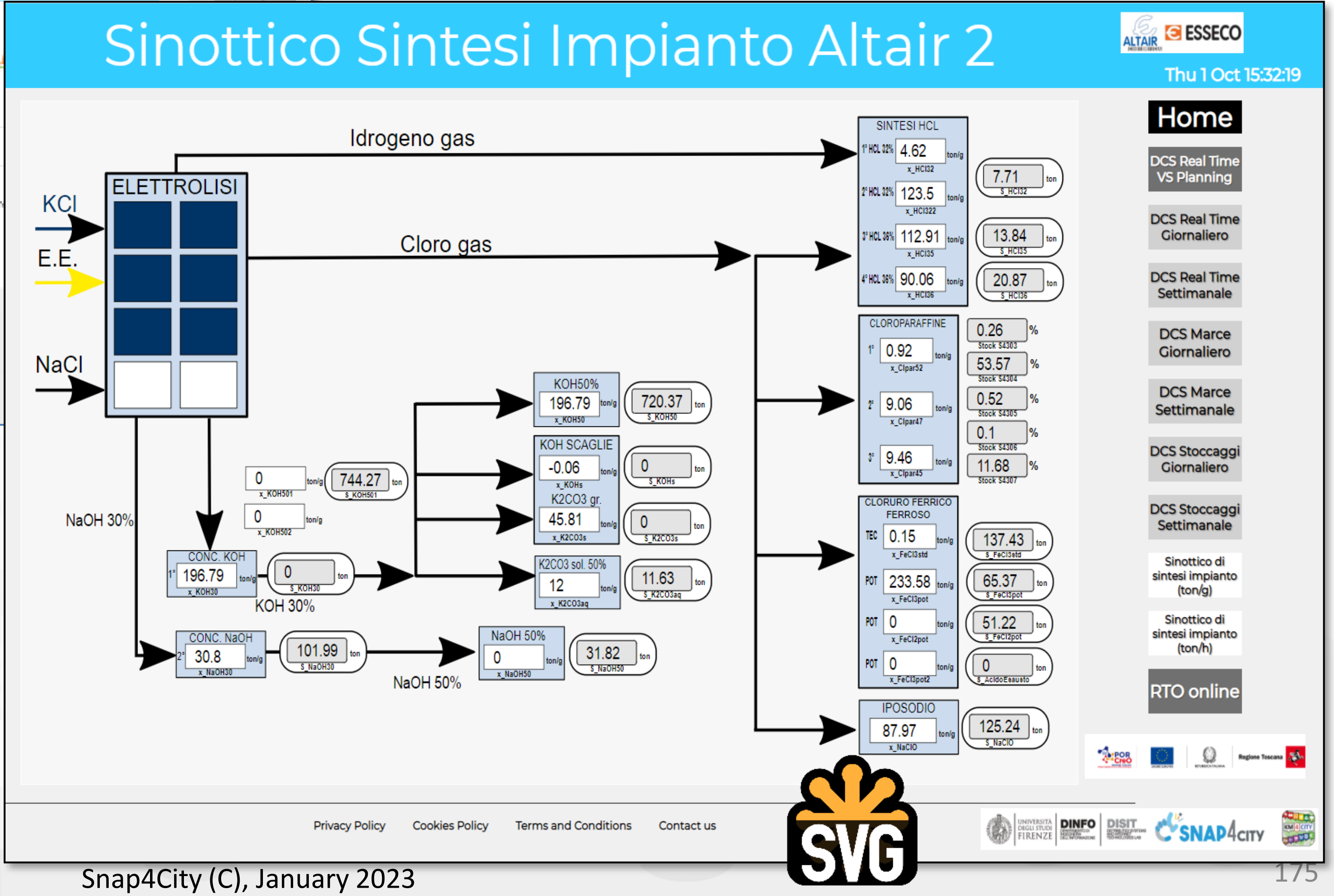
Home
DCS Real Time VS Planning
DCS Real Time Giornaliero
DCS Real Time Settimanale
DCS Marce Giornaliero
DCS Marce Settimanale
DCS Stoccaggi Giornaliero
DCS Stoccaggi Settimanale
Sinottico di sintesi impianto

ALTAIR CHIMICA ESSECO

SNAP4CITY

SNAP4ALTAIR

Sinottico Sintesi Impianto Altair 2



ALTAIR ESSECO

Thu 1 Oct 15:32:19

SVG

Snap4City (C), January 2023

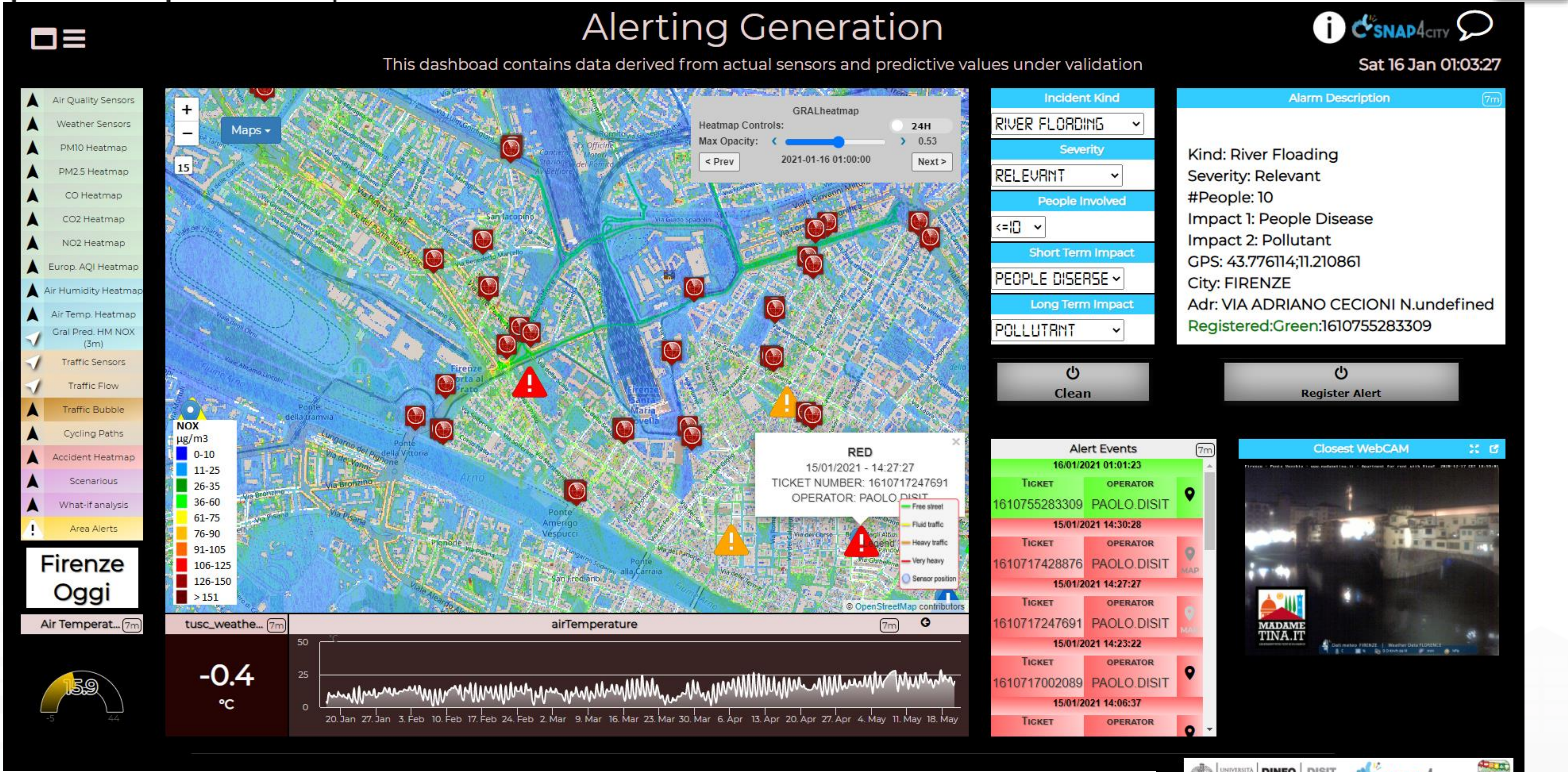
TOP

Develop: Dashboards with Server-Side Business Intelligence

Principles of Server Side Business Logic

- It is possible to have one Processing Logic (IoT App) referring to multiple Dashboards, and one Dashboard referring multiple Processing Logic (IoT Apps)
- Let see a 1:1 relationship from Proc.Logic and Dashboard
  - Any Action performed on Dashboard is provided to the Proc.Logic, which may produce reactions on Dashboard.
  - The context of Proc.Logic ←→Dashboard is a singleton, thus any user connected to the Dashboard will observe the evolutions performed. So that all the users will see the same story and view
  - This is good for control rooms, and single/few users prototypes

Alert Registration



Alerting Generation

This dashboad contains data derived from actual sensors and predictive values under validation

Sat 16 Jan 01:03:27

RED
15/01/2021 - 14:27:27
TICKET NUMBER: 1610717247691
OPERATOR: PAOLO.DISIT

Incident Kind: RIVER FLOADING

Severity: RELEVANT

People Involved: <=10

Short Term Impact: PEOPLE DISEASE

Long Term Impact: POLLUTANT

Alarm Description

Kind: River Floading
Severity: Relevant
#People: 10
Impact 1: People Disease
Impact 2: Pollutant
GPS: 43.776114;11.210861
City: FIRENZE
Adr: VIA ADRIANO CECIONI N.undefined
Registered:Green:1610755283309

Alert Events

16/01/2021 01:01:23
TICKET 1610755283309 OPERATOR PAOLO.DISIT

15/01/2021 14:30:28
TICKET 1610717428876 OPERATOR PAOLO.DISIT

15/01/2021 14:27:27
TICKET 1610717247691 OPERATOR PAOLO.DISIT

15/01/2021 14:23:22
TICKET 1610717002089 OPERATOR PAOLO.DISIT

15/01/2021 14:06:37
TICKET OPERATOR

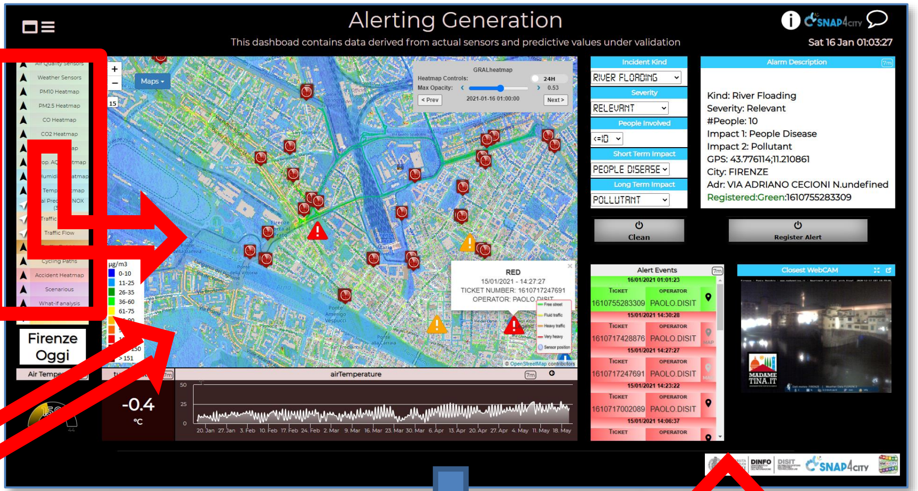
https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MzA0OQ==

Maps Server Side Business Logic vs IOT Apps

Any Snap4City data and sources: IoT Devices, Variables, Heatmaps, traffic, tools, KPI, etc.

data

Selector



Alerting Generation

This dashboard contains data derived from actual sensors and predictive values under validation

Sat 16 Jan 01:03:27

Incident Kind: RIVER FLOADING

Severity: RELEVANT

People Involved: <=10

Short Term Impact: PEOPLE DISEASE

Long Term Impact: POLLUTANT

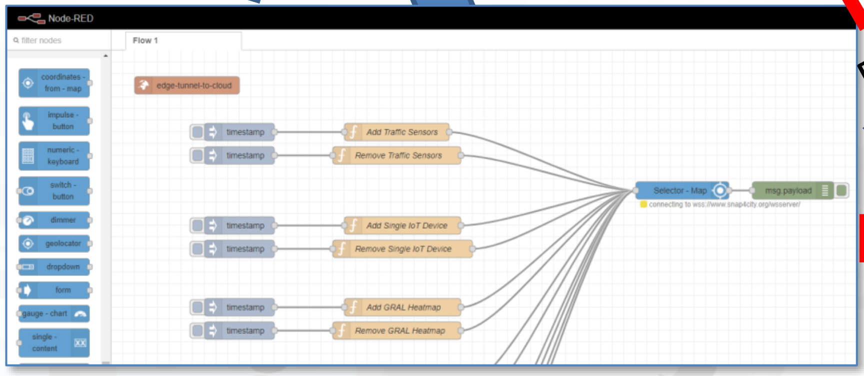
Alarm Description

Kind: River Floading
Severity: Relevant
#People: 10
Impact 1: People Disease
Impact 2: Pollutant
GPS: 43.776114;11.210861
City: FIRENZE
Adr: VIA ADRIANO CECIONI N.undefined
Registered:Green:1610755283309

Data, changed data

References, commands, directives, selections

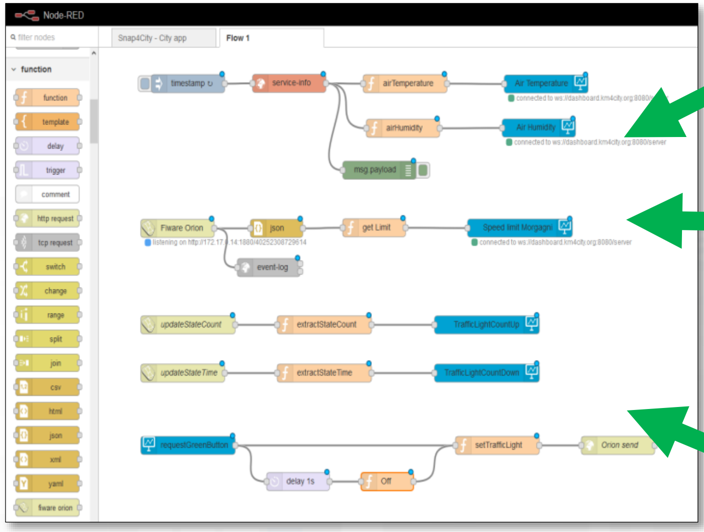
data



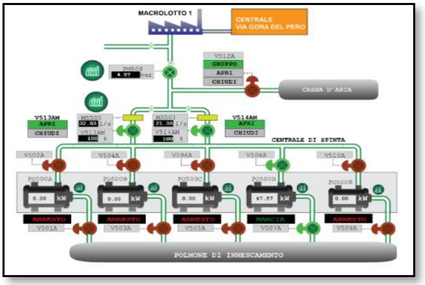
selections, positions, ServiceURI

Advanced Proc.Logic (IoT Apps)

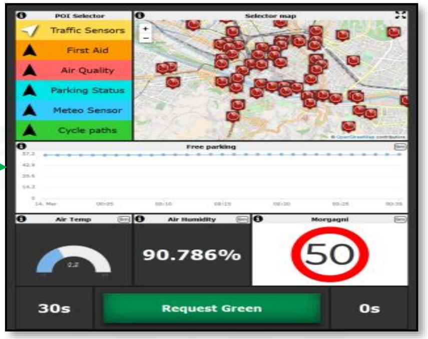
- **Synoptics can ……**
  - **do all ☺**
- **Widgets can**
  - send/receive dynamic data,
  - change data sources, etc.
  - Provide interactive maps
- **HTML pages can**
  - be dynamically generated
  - provide forms to produce data for Proc. Logic (IoT App)
  - Collect files on web and system
  - produce files on web ad system
  - have CSS and AJAX control



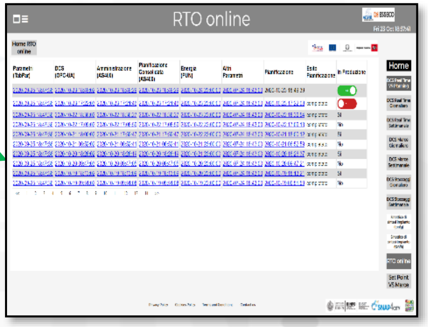
Synoptics
Custom
Widgets



Widgets
Maps
Buttons
Keypads
Controls
..



HTML pages
HTML Forms
Tables
..

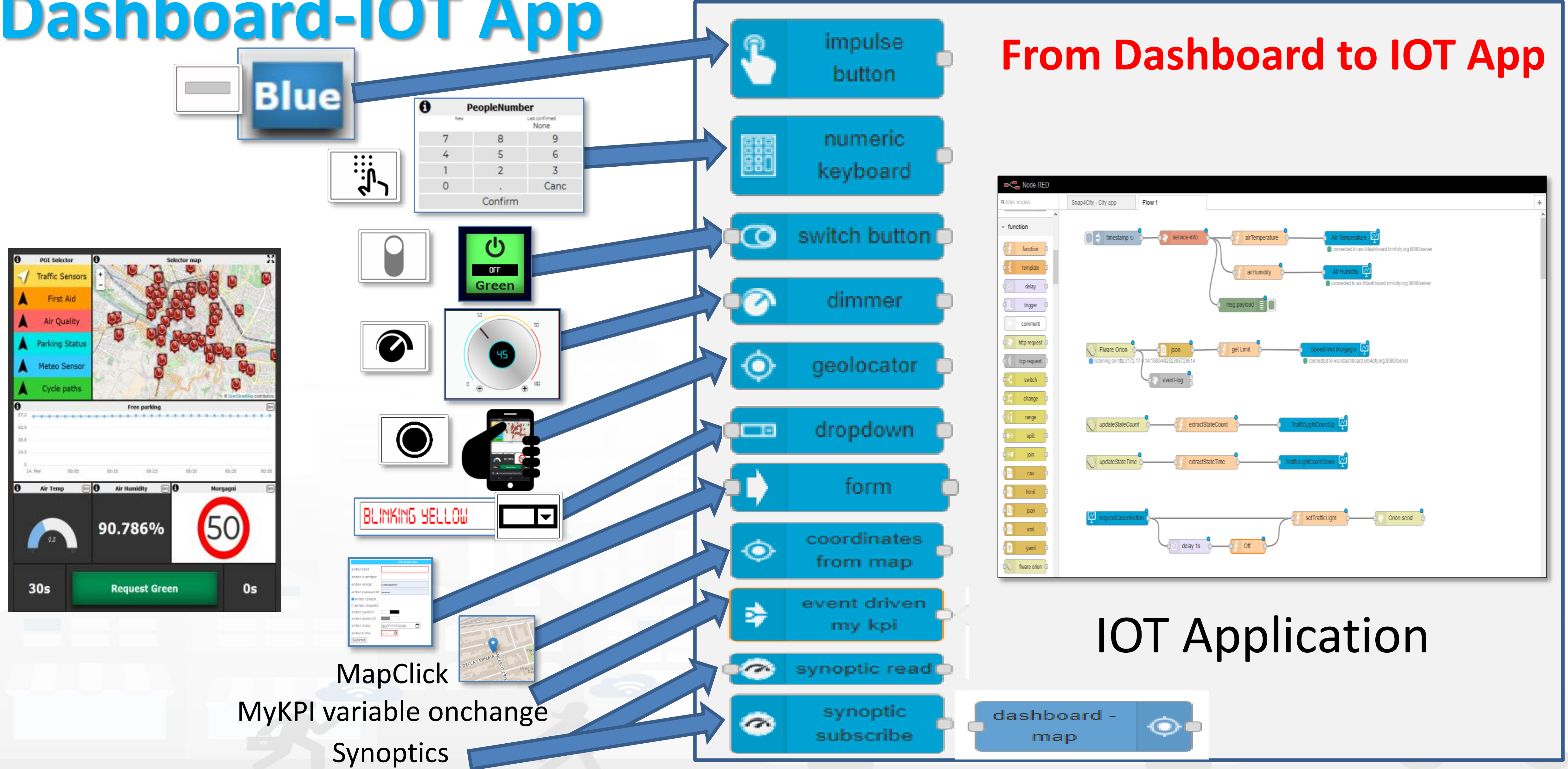


https://www.snap4city.org/394

https://www.snap4city.org/596

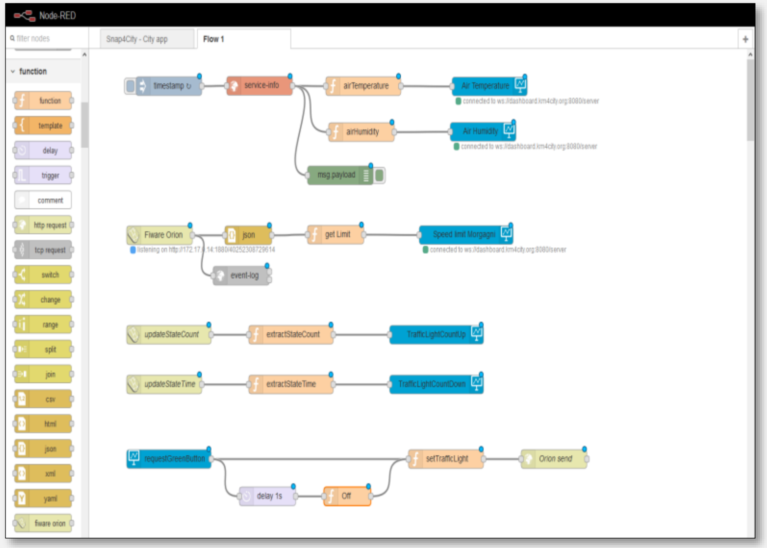
Nature

Dashboard-IOT App



Dashboard-IOT App

From IOT App to Dashboard



IOT Application

- gauge chart
- single content
- speedometer
- horizontal single bar
- vertical single bar
- web content
- time trend
- bar series
- radar series
- pie chart
- curved line series
- table content
- calendar
- speak synthesis
- synoptic write
- Selector - Map
- Snap4D3
- dashboard - map
- event table
- device table

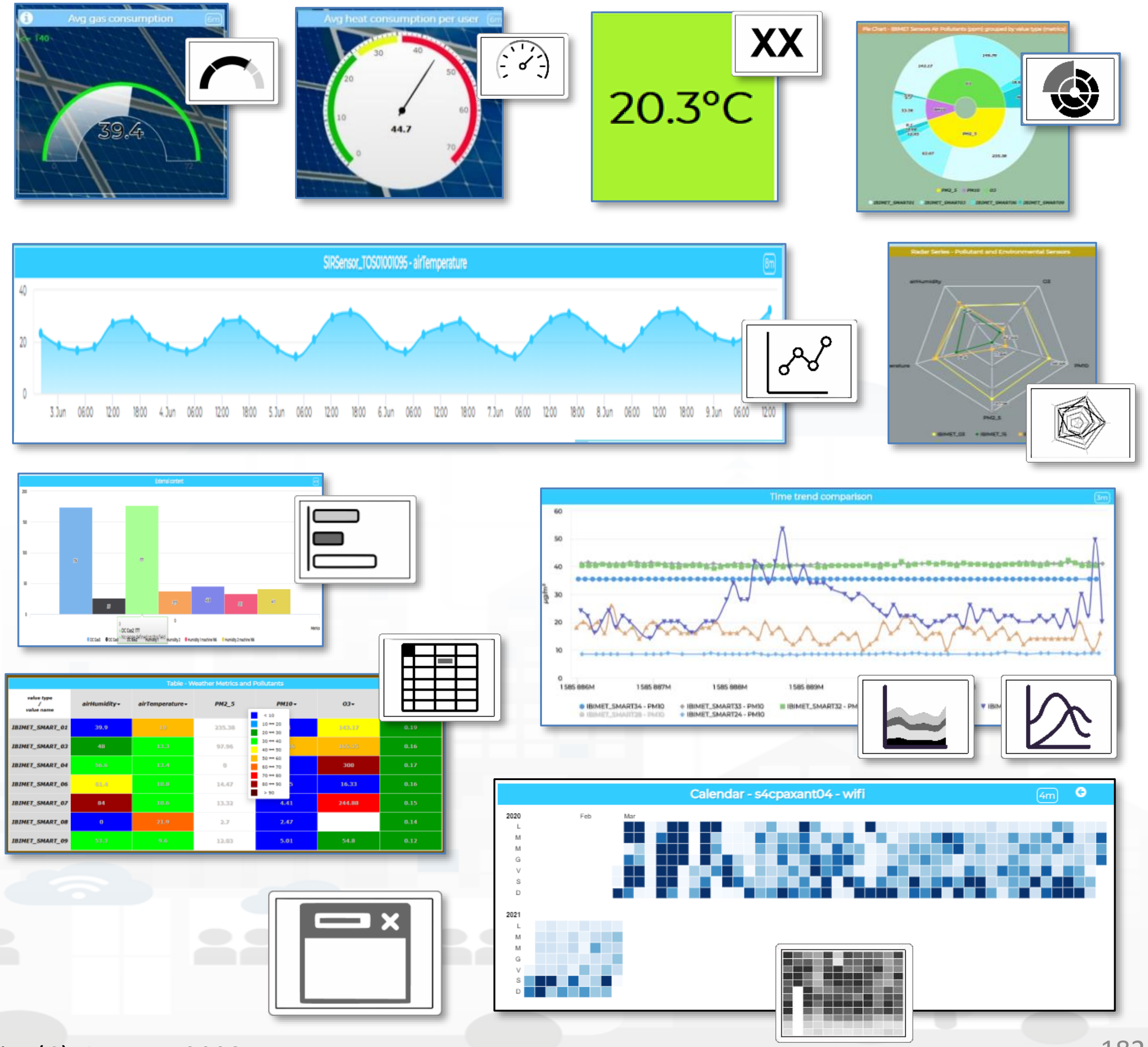
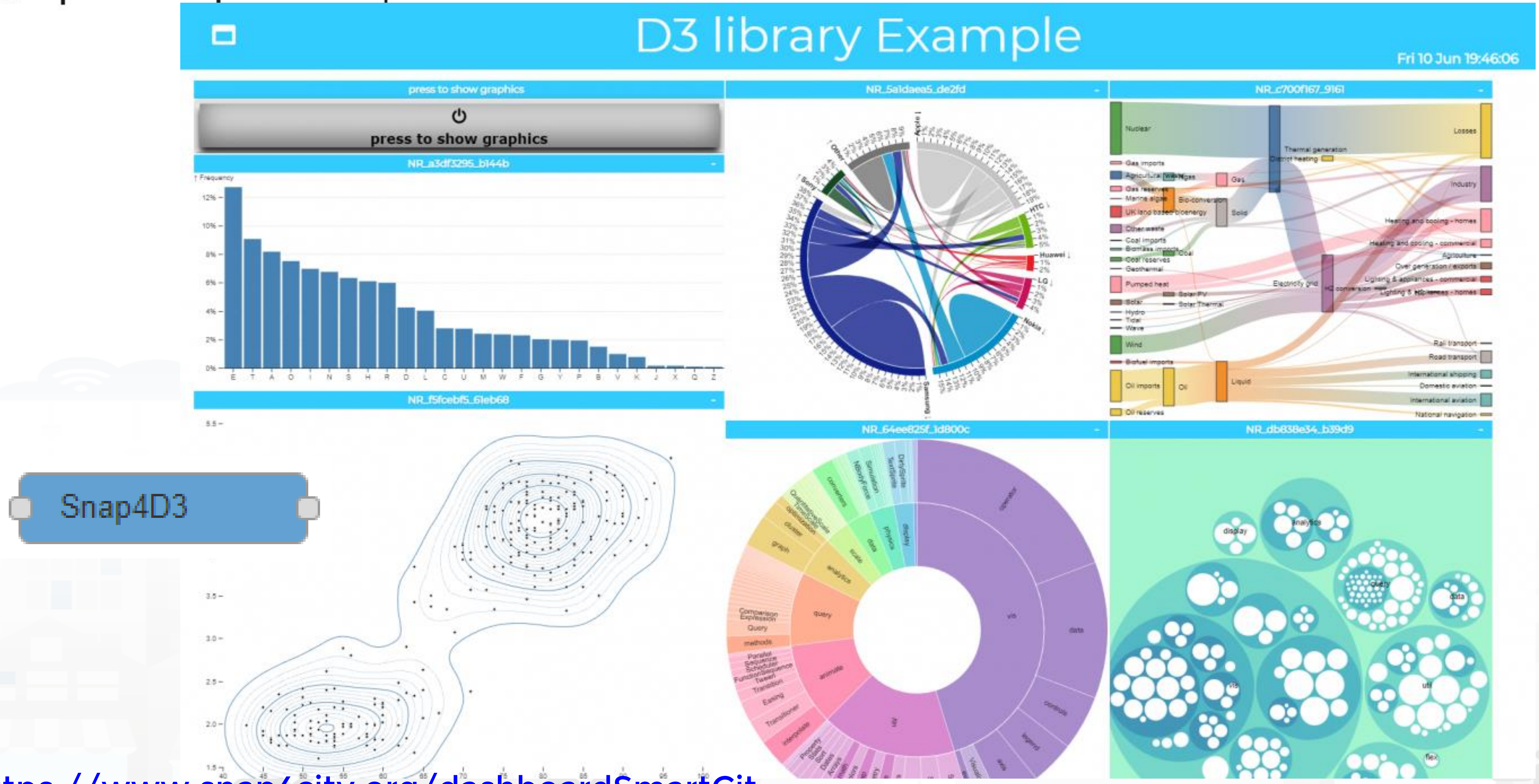


Table - Weather Metrics and Pollutants

value type / value name	airHumidity	airTemperature	PM2_5	PM10	O3	[illegible]
IBIMET_SMART_01	39.9	10	235.38	[illegible]	143.17	0.19
IBIMET_SMART_03	48	13.3	97.96	[illegible]	165.35	0.16
IBIMET_SMART_04	56.6	13.4	0	[illegible]	300	0.17
IBIMET_SMART_06	61.4	10.8	14.47	[illegible]	16.33	0.16
IBIMET_SMART_07	84	10.6	13.32	4.41	244.88	0.15
IBIMET_SMART_08	0	21.9	2.7	2.47		0.14
IBIMET_SMART_09	53.3	9.6	12.03	5.01	54.8	0.12

🟦	< 10
🟦	10 ⟷ 20
🟩	20 ⟷ 30
🟩	30 ⟷ 40
🟨	40 ⟷ 50
🟧	50 ⟷ 60
🟧	60 ⟷ 70
🟥	70 ⟷ 80
🟥	80 ⟷ 90
🟫	> 90

+ D3.JS Widgets



D3 library Example

Fri 10 Jun 19:46:06

press to show graphics

⏻ press to show graphics

NR_a3df3295_b144b

NR_f5fcebf5_61eb68

NR_5a1daea5_de2fd

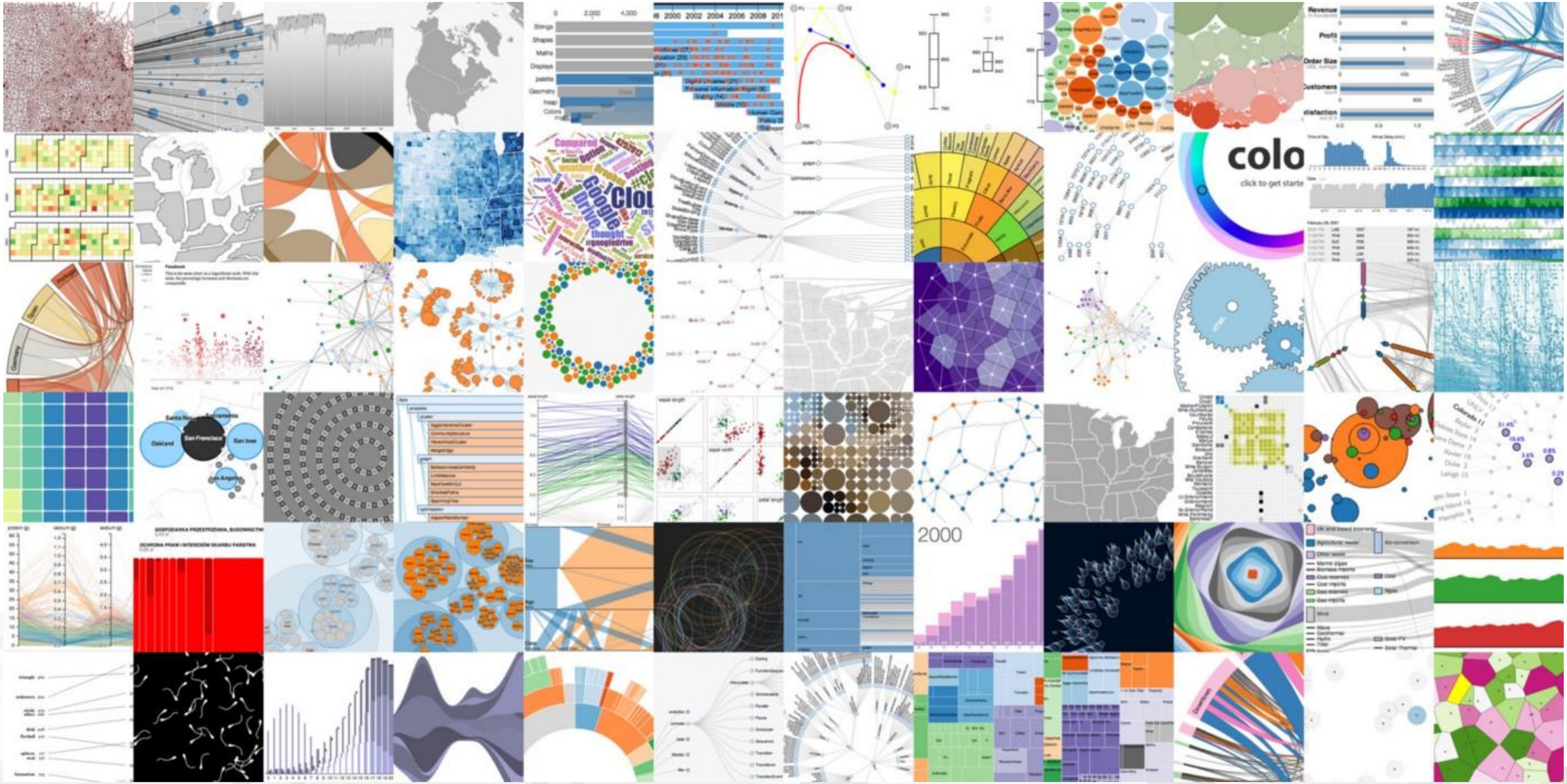
NR_c700f167_9161

NR_64ee825f_1d800c

NR_db838e34_b39d9

https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MzQ4OQ==

D3.js graphs

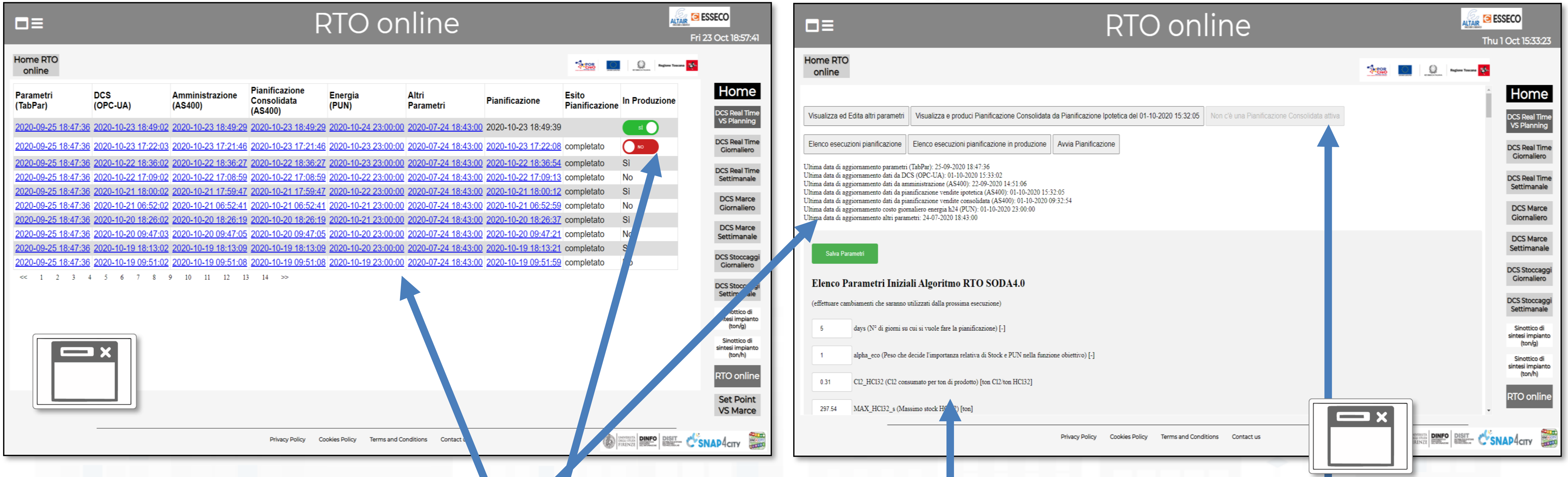


Widgets and their counterpart Nodes

- **Send** information and commands to the Dashboard Widget, for example for an action produced by the users. (**in widget/node**)
- **Receive** information and commands from the Dashboard Widget, for example presenting a dashboard change to the users. (**out widget/node**).
- **Send/receive** information and commands to/from the Dashboard Widget, for example for collecting users' actions and presenting a change to the users on the same widget (**in/out widget/node**).

On Server-Side (into Proc.Logic) the developer can even create some HTML pages and provide them into a Dashboard Widget. And a mixt of Widgets in, out, in/out

Proc.Logic (IOT App) with Dynamic Web Pages



Home RTO online

Parametri (TabPar)	DCS (OPC-UA)	Amministrazione (AS400)	Pianificazione Consolidata (AS400)	Energia (PUN)	Altri Parametri	Pianificazione	Esito Pianificazione	In Produzione
2020-09-25 18:47:36	2020-10-23 18:49:02	2020-10-23 18:49:29	2020-10-23 18:49:29	2020-10-24 23:00:00	2020-07-24 18:43:00	2020-10-23 18:49:39		sì
2020-09-25 18:47:36	2020-10-23 17:22:03	2020-10-23 17:21:46	2020-10-23 17:21:46	2020-10-23 23:00:00	2020-07-24 18:43:00	2020-10-23 17:22:08	completato	NO
2020-09-25 18:47:36	2020-10-22 18:36:02	2020-10-22 18:36:27	2020-10-22 18:36:27	2020-10-23 23:00:00	2020-07-24 18:43:00	2020-10-22 18:36:54	completato	Sì
2020-09-25 18:47:36	2020-10-22 17:09:02	2020-10-22 17:08:59	2020-10-22 17:08:59	2020-10-22 23:00:00	2020-07-24 18:43:00	2020-10-22 17:09:13	completato	No
2020-09-25 18:47:36	2020-10-21 18:00:02	2020-10-21 17:59:47	2020-10-21 17:59:47	2020-10-22 23:00:00	2020-07-24 18:43:00	2020-10-21 18:00:12	completato	Sì
2020-09-25 18:47:36	2020-10-21 06:52:02	2020-10-21 06:52:41	2020-10-21 06:52:41	2020-10-21 23:00:00	2020-07-24 18:43:00	2020-10-21 06:52:59	completato	No
2020-09-25 18:47:36	2020-10-20 18:26:02	2020-10-20 18:26:19	2020-10-20 18:26:19	2020-10-21 23:00:00	2020-07-24 18:43:00	2020-10-20 18:26:37	completato	Sì
2020-09-25 18:47:36	2020-10-20 09:47:03	2020-10-20 09:47:05	2020-10-20 09:47:05	2020-10-20 23:00:00	2020-07-24 18:43:00	2020-10-20 09:47:21	completato	No
2020-09-25 18:47:36	2020-10-19 18:13:02	2020-10-19 18:13:09	2020-10-19 18:13:09	2020-10-20 23:00:00	2020-07-24 18:43:00	2020-10-19 18:13:21	completato	Sì
2020-09-25 18:47:36	2020-10-19 09:51:02	2020-10-19 09:51:08	2020-10-19 09:51:08	2020-10-19 23:00:00	2020-07-24 18:43:00	2020-10-19 09:51:59	completato	No

<< 1 2 3 4 5 6 7 8 9 10 11 12 13 14 >>

Home RTO online

Visualizza ed Edita altri parametri | Visualizza e produci Pianificazione Consolidata da Pianificazione Ipotetica del 01-10-2020 15:32:05 | Non c'è una Pianificazione Consolidata attiva

Elenco esecuzioni pianificazione | Elenco esecuzioni pianificazione in produzione | Avvia Pianificazione

Ultima data di aggiornamento parametri (TabPar): 25-09-2020 18:47:36
Ultima data di aggiornamento dati da DCS (OPC-UA): 01-10-2020 15:33:02
Ultima data di aggiornamento dati da amministrazione (AS400): 22-09-2020 14:51:06
Ultima data di aggiornamento dati da pianificazione vendite ipotetica (AS400): 01-10-2020 15:32:05
Ultima data di aggiornamento dati da pianificazione vendite consolidata (AS400): 01-10-2020 09:32:54
Ultima data di aggiornamento costo giornaliero energia h24 (PUN): 01-10-2020 23:00:00
Ultima data di aggiornamento altri parametri: 24-07-2020 18:43:00

Salva Parametri

Elenco Parametri Iniziali Algoritmo RTO SODA4.0

(effettuare cambiamenti che saranno utilizzati dalla prossima esecuzione)

5 days (N° di giorni su cui si vuole fare la pianificazione) [-]

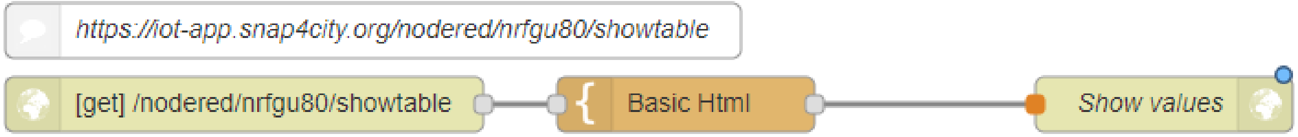
1 alpha_eco (Peso che decide l'importanza relativa di Stock e PUN nella funzione obiettivo) [-]

0.31 Cl2_HCl32 (Cl2 consumato per ton di prodotto) [ton Cl2/ton HCl32]

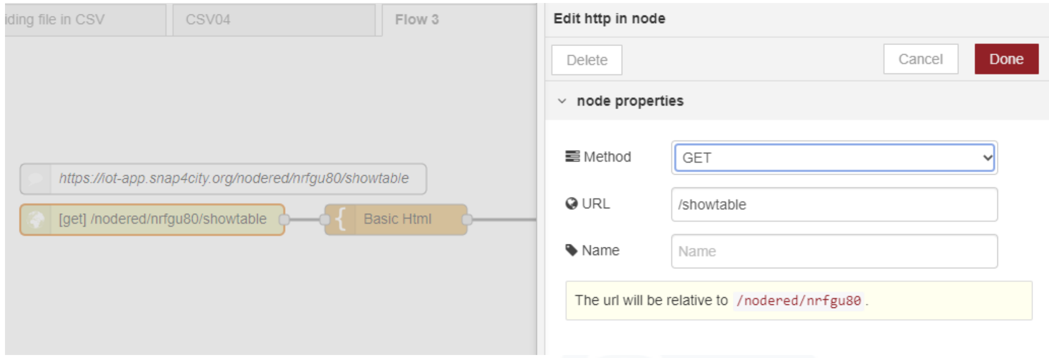
297.54 MAX_HCl32_s (Massimo stock HCl32) [ton]

- **HTML pages** can
  - be dynamically generated from the Proc.Logic / IoT App
  - provide forms to produce data to the Proc.Logic / IoT App, also including interactive elements
  - collect file from users, and produce files to web and to the system
  - have CSS and AJAX controls

From IoT App to generate HTML pages, forms


https://iot-app.snap4city.org/nodered/nrfgu80/showtable
[get] /nodered/nrfgu80/showtable
Basic Html
Show values

```
<html>
<style>table, th, td {  border:1px solid black;}</style>
<head></head>
<body>
<h1>This is a table</h1><table style="width:100%">
      <tr>
         <th>Person 1</th>   <th>Person 2</th>   <th>Person 3</th>
      </tr>
      <tr>
         <td>Emil</td>   <td>Tobias</td>   <td>Linus</td>
      </tr>
      <tr>
        <td>16</td>   <td>14</td>   <td>10</td>
      </tr>
</table>
</body> </html>
```


ding file in CSV
CSV04
Flow 3
https://iot-app.snap4city.org/nodered/nrfgu80/showtable
[get] /nodered/nrfgu80/showtable
Basic Html
Delete
Cancel
Done

Edit http in node

node properties

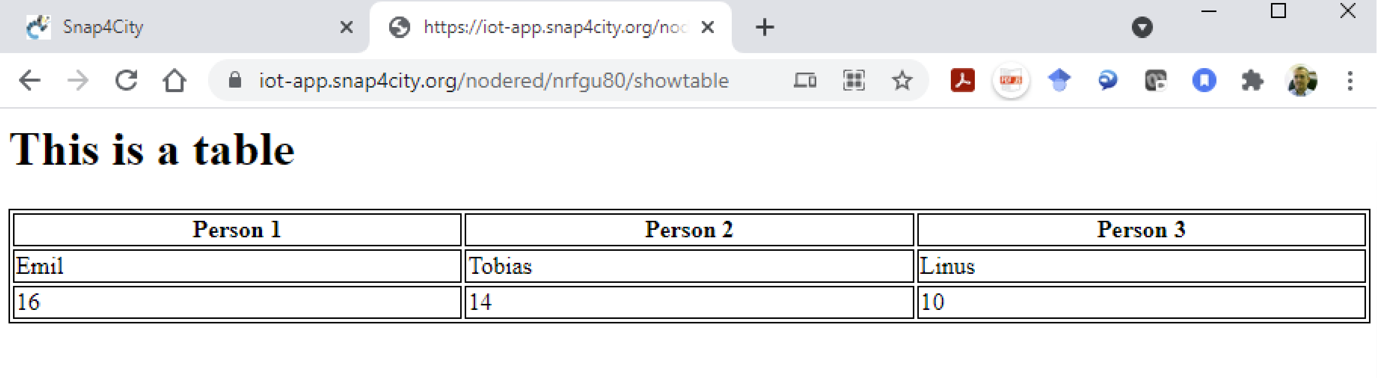
Method: GET

URL: /showtable

Name: Name

The url will be relative to /nodered/nrfgu80.

Edit "HTTP IN NODE" as above, also to get the ID of your IoT APP.
The ID is used in the URL of the generated page


Snap4City
https://iot-app.snap4city.org/noc
iot-app.snap4city.org/nodered/nrfgu80/showtable

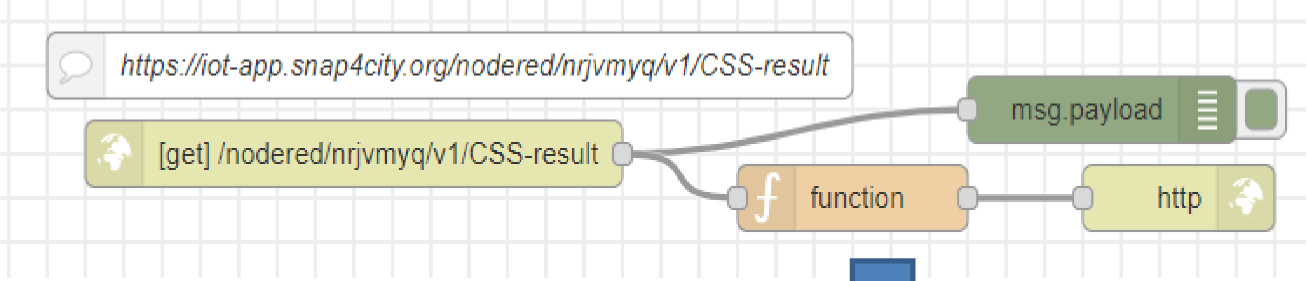
This is a table

Person 1	Person 2	Person 3
Emil	Tobias	Linus
16	14	10

HTML & Tables on Dashboards

- HTML page can expose forms to collect data for the IoT App.
- The table can be
  - constructed with the style you prefer according to HTML, CSS, etc.
  - dynamically generated on the basis of the values you collect/generate, receive, recover from storage in the flow
  - updated by send a message on the node
  - show on Dashboard by using the link (URL) into an External Content Widget
- In alternative there is to the Widget Table with less flexibility

From IoT App to API Get


https://iot-app.snap4city.org/nodered/nrjvmyq/v1/CSS-result
msg.payload
[get] /nodered/nrjvmyq/v1/CSS-result
function
http

Function receives:

{"prova":"1","test":"mio"}

It can interpretes the REST call to provide at the next Node the result

Call on Browser:

https://iot-app.snap4city.org/nodered/nrjvmyq/v1/CSS-result/?prova=1&test=mio

Domain Prefix

IoT App ID

Your custom

TOP

Develop: Dashboards with Client-Side Business Intelligence

Client Side Business Logic, CSBL

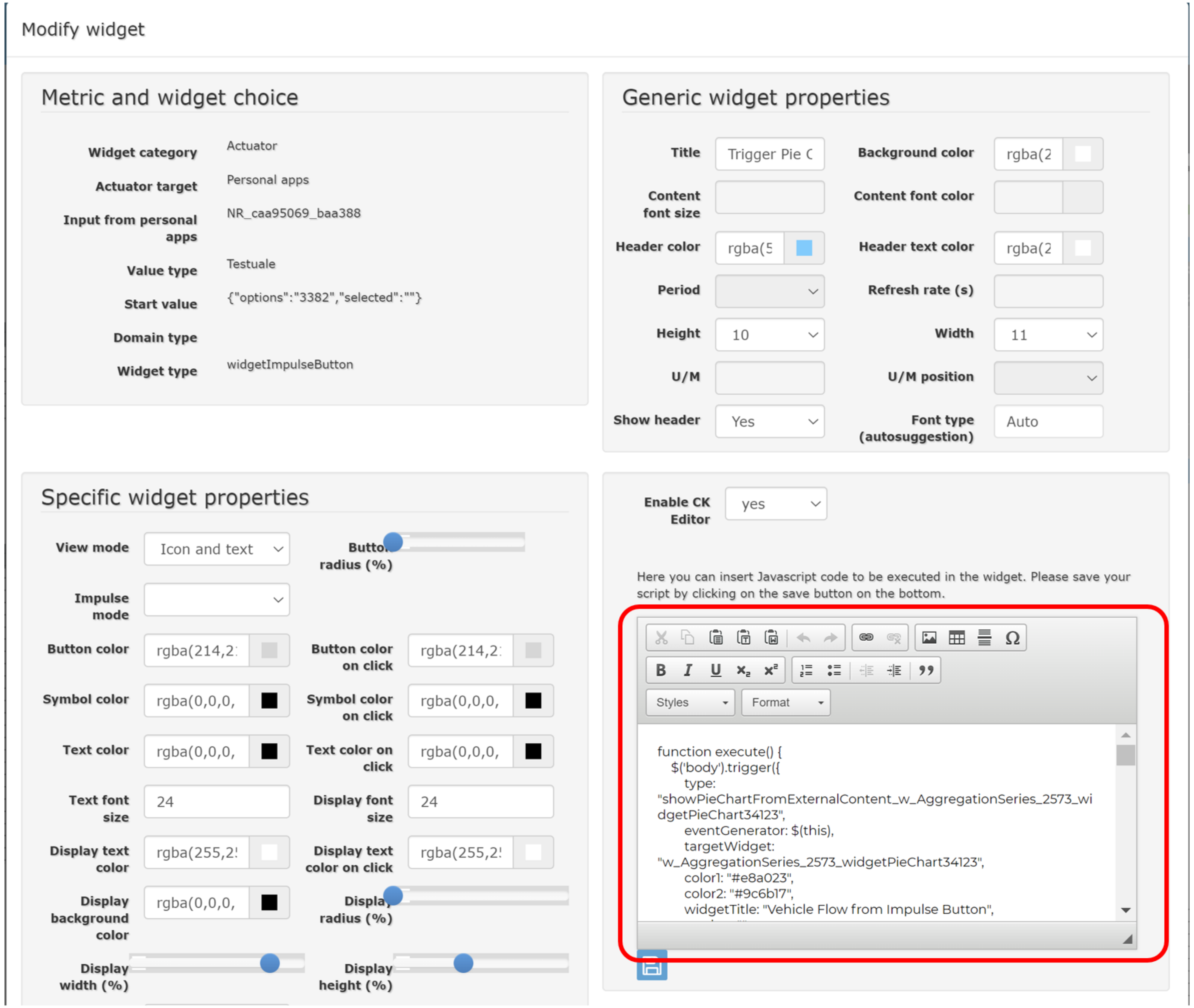
- solution to close the loop from user actions and effects on widgets directly on the client side, on the browser
- **Client-Side Business Logic, CSBL**, and **Server-Side Business Logics, SSBL**, may be present at the same time behind a Dashboard and thus behind a Business Intelligence / Smart Application
- CSBL the logic code is formalized in JavaScript only, while in SSBL the logic is formalized in Proc.Logic which is Node-RED plus some JavaScript.
- Developers that would like to develop CSBL have to be singularly authorized, please ask to snap4city@disit.org
- When working in SSBL, widgets can be created and edited from Node-Red Processing Logic. When working in CSBL context, widgets can be created through the Dashboard Wizard

Client Side Business Logic, CSBL

- **IN Widgets** are those that are prepared to receive some actions/commands from the Users. For example, a click on a button, a click on the map, etc. These IN Widgets can be regarded as Virtual Sensors.
- **OUT Widgets** are those that are prepared to provide some changes to be shown into the Users' interface. For example, a view of a barseries on some other data, a rendering of a time series, a rendering of a set of Entities on the map, etc. These OUT Widgets can be regarded as Virtual Actuators.
- **IN/OUT Widgets** are those that provide capabilities of both IN and OUT Widgets. For example, a map can receive an IN command about a selected PIN, and can receive an OUT command to show a selection of services, devices, etc. These IN/OUT Widgets can be regarded as Virtual Sensors/Actuators.

Formalization of SSBL on In Widget
More Options

- CK EDITOR



Modify widget

Metric and widget choice

Widget category	Actuator
Actuator target	Personal apps
Input from personal apps	NR_caa95069_baa388
Value type	Testuale
Start value	{"options":"3382","selected":""}
Domain type	
Widget type	widgetImpulseButton

Generic widget properties

Title	Trigger Pie C	Background color	rgba(2
Content font size		Content font color	
Header color	rgba(5	Header text color	rgba(2
Period		Refresh rate (s)	
Height	10	Width	11
U/M		U/M position	
Show header	Yes	Font type (autosuggestion)	Auto

Specific widget properties

View mode	Icon and text	Button radius (%)	
Impulse mode			
Button color	rgba(214,2:	Button color on click	rgba(214,2:
Symbol color	rgba(0,0,0,	Symbol color on click	rgba(0,0,0,
Text color	rgba(0,0,0,	Text color on click	rgba(0,0,0,
Text font size	24	Display font size	24
Display text color	rgba(255,2!	Display text color on click	rgba(255,2!
Display background color	rgba(0,0,0,	Display radius (%)	
Display width (%)		Display height (%)	

Enable CK Editor: yes

Here you can insert Javascript code to be executed in the widget. Please save your script by clicking on the save button on the bottom.

```
function execute() {
   $('body').trigger({
      type:
"showPieChartFromExternalContent_w_AggregationSeries_2573_wi
dgetPieChart34123",
      eventGenerator: $(this),
      targetWidget:
"w_AggregationSeries_2573_widgetPieChart34123",
      color1: "#e8a023",
      color2: "#9c6b17",
      widgetTitle: "Vehicle Flow from Impulse Button",
```

IN IN/OUT Widgets

IN IN/OUT Widgets	Users' Action Description and effects
widgetTimeTrend	Drill-Down on time interval selection (zoom), providing, SURI, value name, start and end time stamp
widgetTimeTrend	Reset Drill-Down (under development)
widgetTimeTrend	Click on a single time instant, providing time stamp, SURI, value name
widgetTimeTrend	Click on legend, providing the status (e.g.: "checked" or "unchecked") of the metric/SURI which has been clicked (under development)
widgetMap	Click on a generic point on the map, providing coordinates (under development, currently it only works for SSBL)
widgetMap	Click on a PIN, providing coordinates and ServiceURI of the clicked PIN
widgetMap	Select the bounding box area shown on the map, and the zoom level in order to perform geographical Drill-Down on the entities (devices identified by SURIs, Points of Interest etc.) which are currently shown on map
widgetPieChart	Click on a sector that identifies the name of a metric, providing: value, timestamp, value name, value type (SURI can be reconstructed)
widgetPieChart	click on a sector that identifies a device ID or MyKPI ID, providing: value, timestamp, value name, value type (SURI can be reconstructed)
widgetPieChart	Click on legend, providing the status (e.g.: "checked" or "unchecked") of the metric/SURI which has been clicked (under development)
widgetBarSeries	Click on a bar, providing: value, timestamp, value name, value type (SURI can be reconstructed)
widgetBarSeries	Click on legend, providing the status (e.g.: "checked" or "unchecked") of the metric/SURI which has been clicked (under development)
widgetCurvedLineSeries (multi series)	Drill-Down on time interval selection (zoom), providing: start and end time stamp, and list of SURI
widgetCurvedLineSeries (multi series)	Click on a single time instant, providing: time stamp, and list of SURI
widgetCurvedLineSeries (multi series)	Click on legend, providing the status (e.g.: "checked" or "unchecked") of the metric/SURI which has been clicked (under development)
widgetDeviceTable	Click on the action buttons, providing the action type, the corresponding SURI and a list of attributes with their corresponding values
widgetImpulseButton	Click on button as a trigger (no parameters are provided)
widgetOnOffButton	Click on button, providing the new status
widgetKnob	Drag on knob, providing the value selected on the knob
widgetNumericKeyboard	Click on the confirm button, providing the numeric value typed on the keyboard
widgetEventTable	Click on the action buttons, providing the action type, the corresponding event SURI and the ordering criteria
widgetRadarSeries	Click on a radar axis related to a specific metric of a specific Entity Instance (device), providing: value, timestamp, value name, value type (SURI can be reconstructed) – under development
widgetExternalContent	It can support HTML pages and SVG Synoptics, in addition to JavaScript, so that it can perform a wide range of actions that can be defined in the HTML/SVG/JS code by the users.

OUT Widgets

OUT IN/OUT Widgets	Commands which are ready to execute from JavaScript
widgetPieChart	Receive a JSON containing a list of SURI, metric names and/or values, and show their corresponding values on a Pie Chart graph.
widgetBarSeries	Receive a JSON object containing a list of SURI, metric names and/or values, and show their corresponding values on a Bar graph.
widgetSingleContent	• Receive a SURI and a metric name, or a value, or a text string, and show the corresponding value. • Receive and show a HTML/JS page
widgetSpeedometer	Receive a SURI and a metric name, or a value, and show the corresponding value on a speedometer graph.
widgetGaugeChart	Receive a SURI and a metric name, or a value, and show the corresponding value on a gauge graph.
widgetTimeTrend	Receive a SURI and a metric name, or a value, and show the corresponding time-series on a line, spline, area or stacked area graph.
widgetTable	Receive a JSON containing a list SURI, metric names and/or values, and show the corresponding the time-series on a HTML static table.
widgetCurvedLineSeries	Receive a JSON containing a list of SURI, metric names and/or values, and show the corresponding time-series on a line, spline, area or stacked area graph.
widgetDeviceTable	Receive a JSON containing a list of SURI representing Entity Instances (IoT devices), and show their related attributes and values on an interactive table which provides action buttons.
widgetEvent	Receive a JSON containing a list of SURI representing events as virtual devices, and show their related attributes (e.g., start and end date) and values on an interactive table which provides action buttons.
widgetMap	Receive a JSON containing a list of SURI or entities (such as heatmaps, categories of Points of Interest etc.) and show them on an interactive map as clickable markers, dynamic SVG pins, traffic flows, heatmaps etc.
widgetOnOffButton	Receive and show a value representing the status (under development), possible via SSBL
widgetKnob	Receive and show a value (under development), possible via SSBL
widgetNumericKeyboard	Receive and show a value (under development), possible via SSBL

Other useful functions

functions on Actions JavaScript segments:

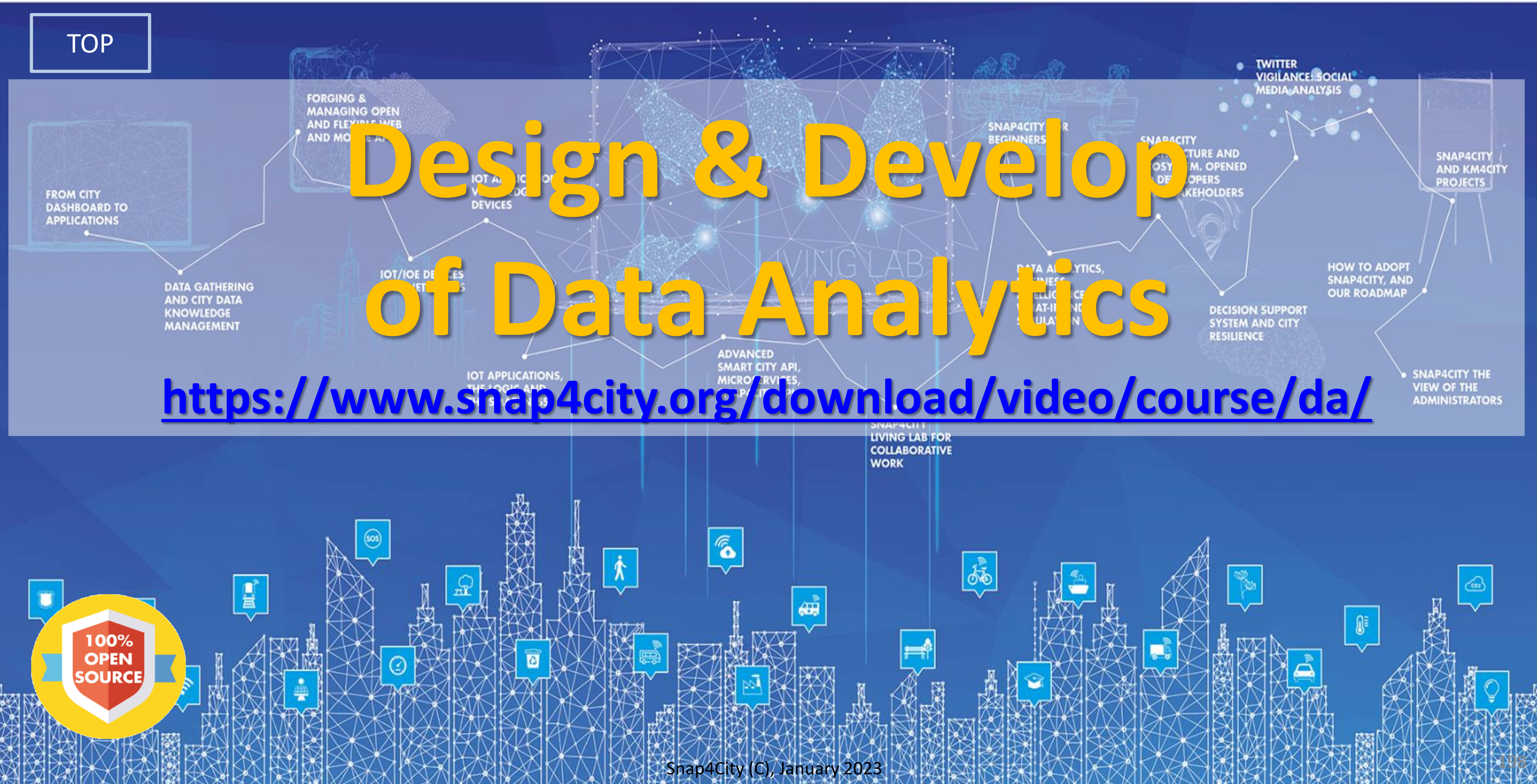
- Open a New Dashboard: openNewDashboard()
- Get parameters: getParams()

As a result, it is possible to activate in a new dashboard some actions on specific elements.

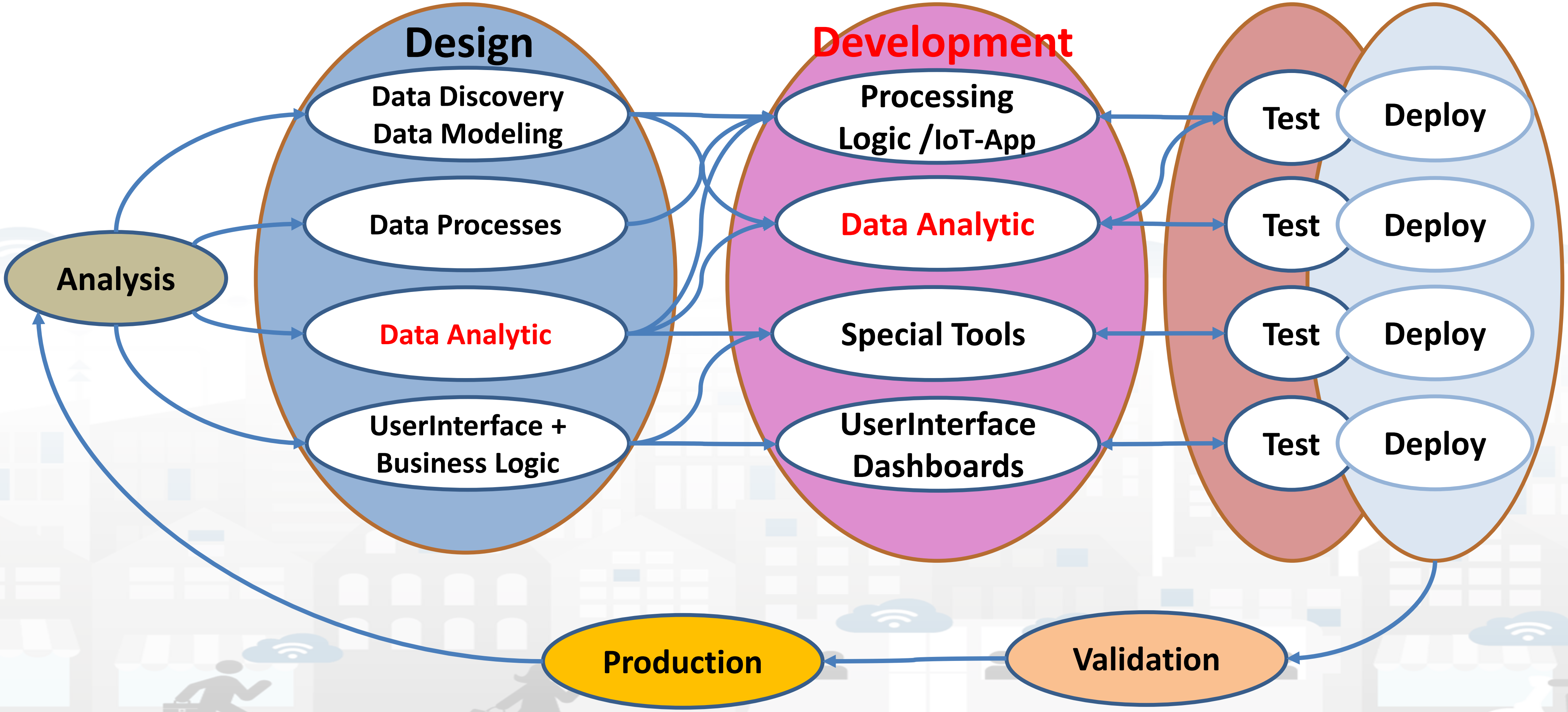
TOP

Design & Develop of Data Analytics

https://www.snap4city.org/download/video/course/da/

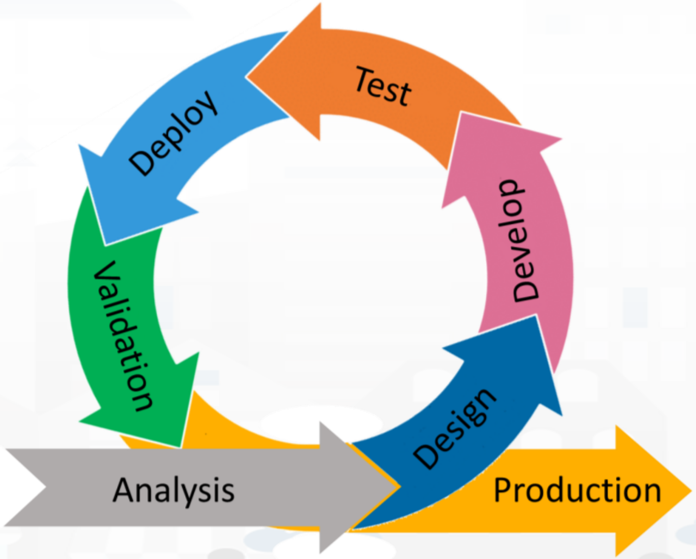

FROM CITY DASHBOARD TO APPLICATIONS
DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT
FORGING & MANAGING OPEN AND FLEXIBLE WEB
IOT/IOE DEVICES
DEVICES
IOT APPLICATIONS,
ADVANCED SMART CITY API, MICROSERVICES,
LIVING LAB
LIVING LAB FOR COLLABORATIVE WORK
SNAP4CITY FOR BEGINNERS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY AND KM4CITY PROJECTS
DECISION SUPPORT SYSTEM AND CITY RESILIENCE
HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP
SNAP4CITY THE VIEW OF THE ADMINISTRATORS
100% OPEN SOURCE

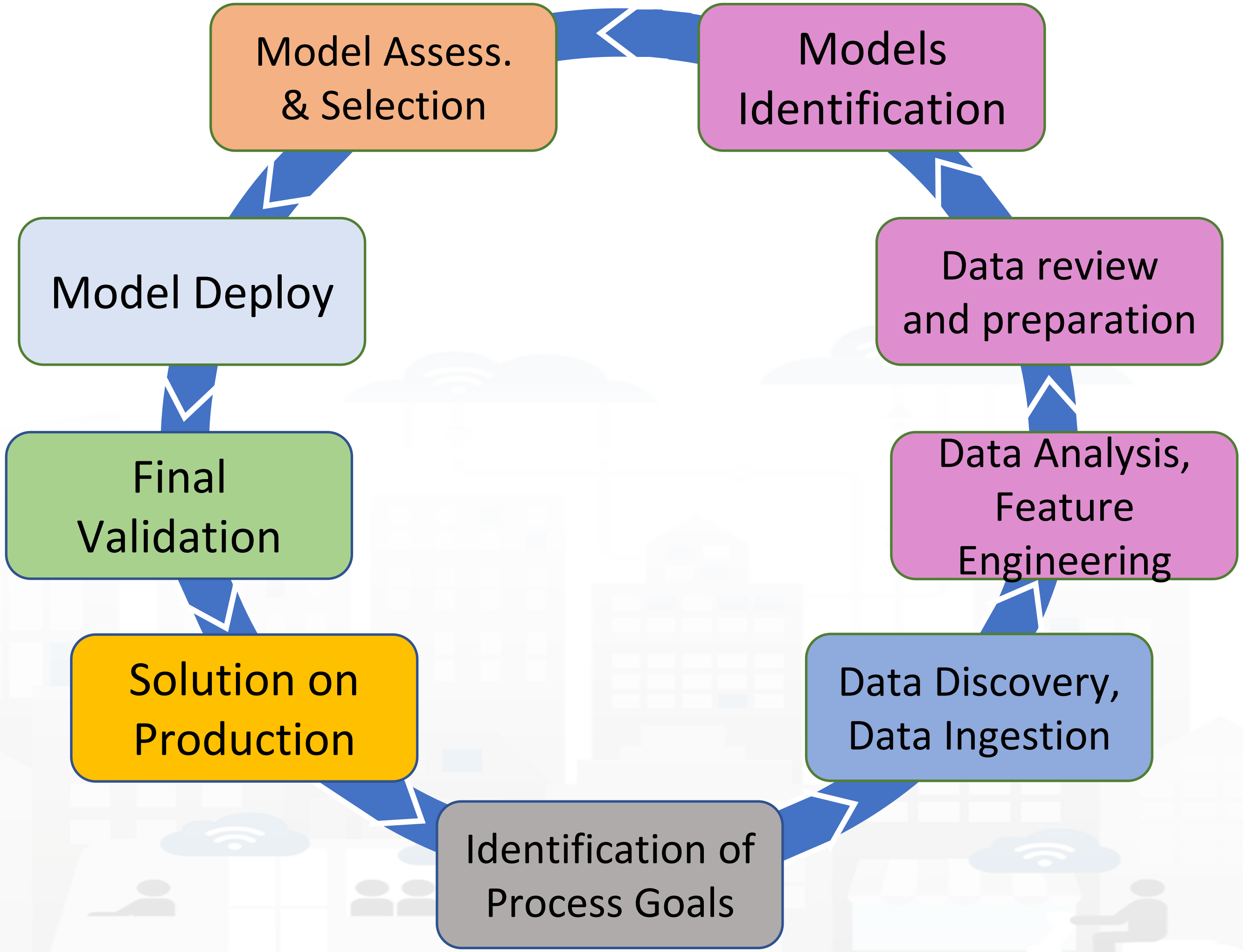
Development Life Cycle Smart Solutions



Data Analytics Development Life Cycle

- Detailed development process on specific training course slides

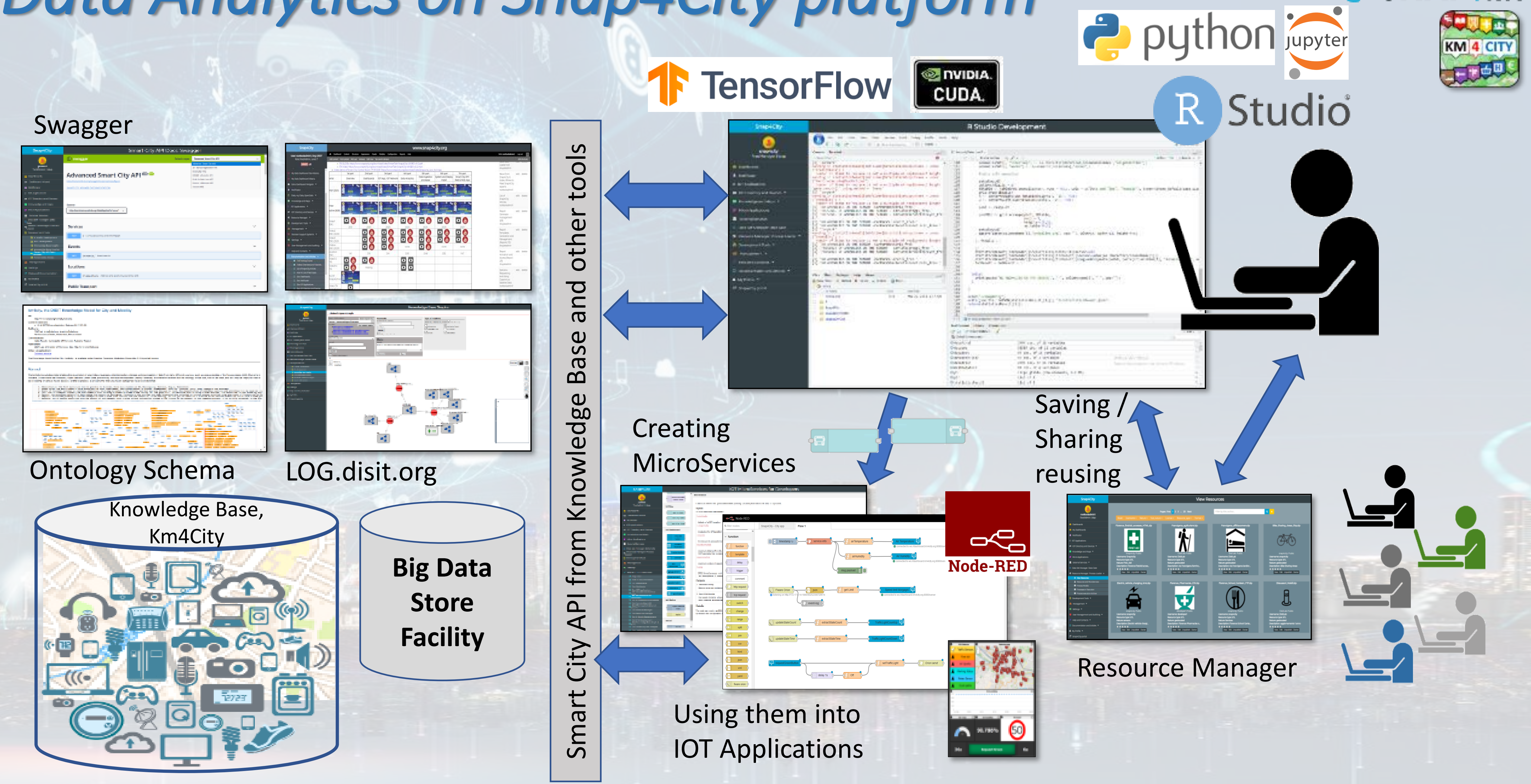



Model Assess. & Selection
Models Identification
Data review and preparation
Data Analysis, Feature Engineering
Data Discovery, Data Ingestion
Identification of Process Goals
Solution on Production
Final Validation
Model Deploy

TOP

Data Analytics on Cloud: Snap4City Infrastructures

Data Analytics on Snap4City platform



Swagger

Ontology Schema

LOG.disit.org

Knowledge Base, Km4City

Big Data Store Facility

Smart City API from Knowledge Base and other tools

Creating MicroServices

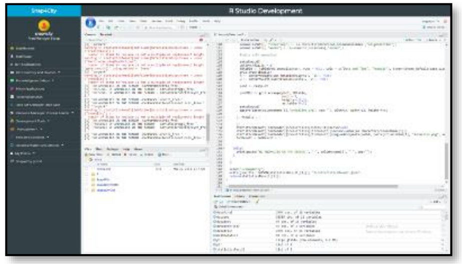
Saving / Sharing reusing

Using them into IOT Applications

Resource Manager

Development

python jupyter

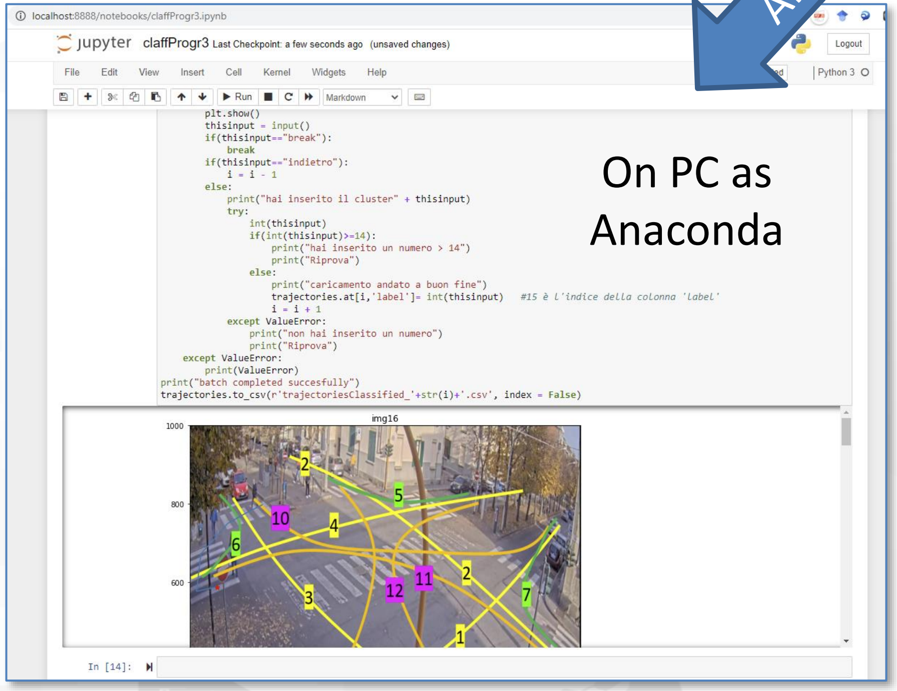


API

Big Data Store Facility

API

On Server
Or
On PC


localhost:8888/notebooks/claffProgr3.ipynb
jupyter claffProgr3 Last Checkpoint: a few seconds ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Python 3
Run
Markdown
img16
1000
800
600
2
5
10
4
6
3
12
11
2
7
1
In [14]:

```
        plt.show()
        thisinput = input()
        if(thisinput=="break"):
            break
        if(thisinput=="indietro"):
            i = i - 1
        else:
            print("hai inserito il cluster" + thisinput)
            try:
                int(thisinput)
                if(int(thisinput)>=14):
                    print("hai inserito un numero > 14")
                    print("Riprova")
                else:
                    print("caricamento andato a buon fine")
                    trajectories.at[i,'label']= int(thisinput)   #15 è l'indice della colonna 'label'
                    i = i + 1
            except ValueError:
                print("non hai inserito un numero")
                print("Riprova")
    except ValueError:
        print(ValueError)
print("batch completed succesfully")
trajectories.to_csv(r'trajectoriesClassified_'+str(i)+'.csv', index = False)
```

On PC as Anaconda

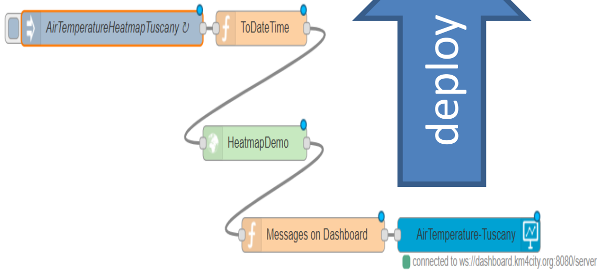
Once finalized

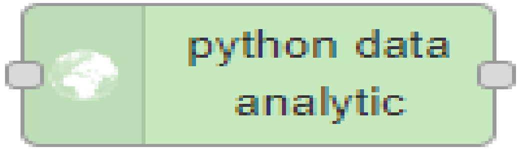
File.py
AI Model
Mapping
Data..

ZIP

Load
File.py
or .zip

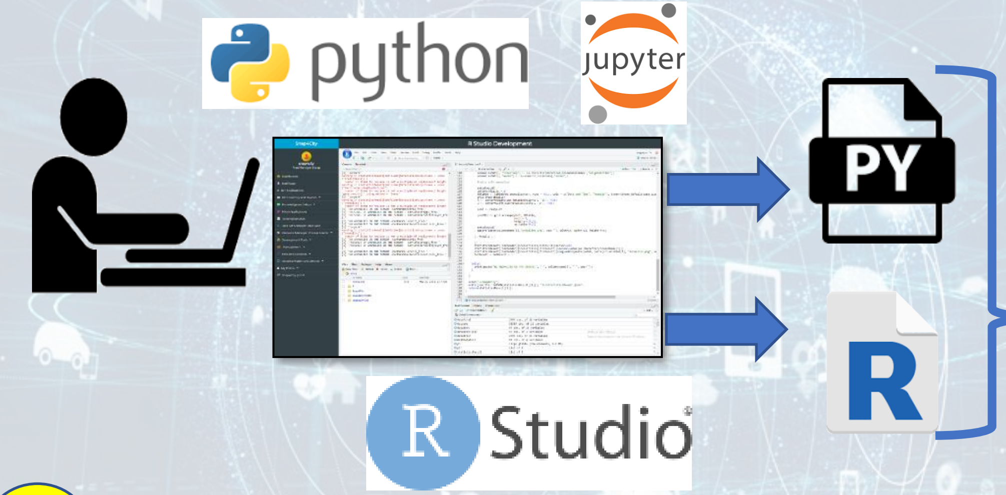
deploy


AirTemperatureHeatmapTuscany
ToDateTime
HeatmapDemo
Messages on Dashboard
AirTemperature-Tuscany
connected to ws://dashboard.km4city.org:8080/server



To make the .PY usable as MicroService you need to adapt it to get and send data in/out with Node-RED from a Container.
If you provide a .zip file the main .py inside has to be called doScript.py

Data Analytic Container

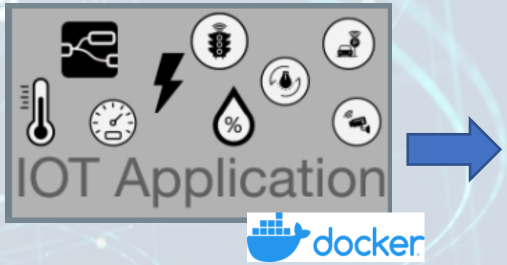


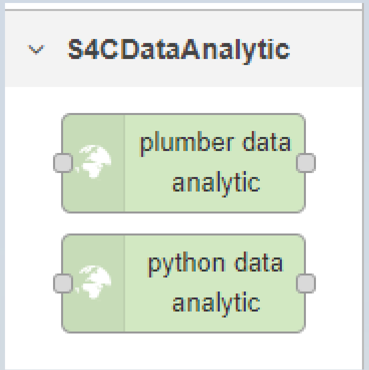
1

Develop .py **or** .r program on (i) Snap4City platform online, or (ii) your Development Machine.
The code has to respect the guidelines provided. For example see:
https://www.snap4city.org/641
https://www.snap4city.org/645

2

Open an Advanced IoT App / Node-RED

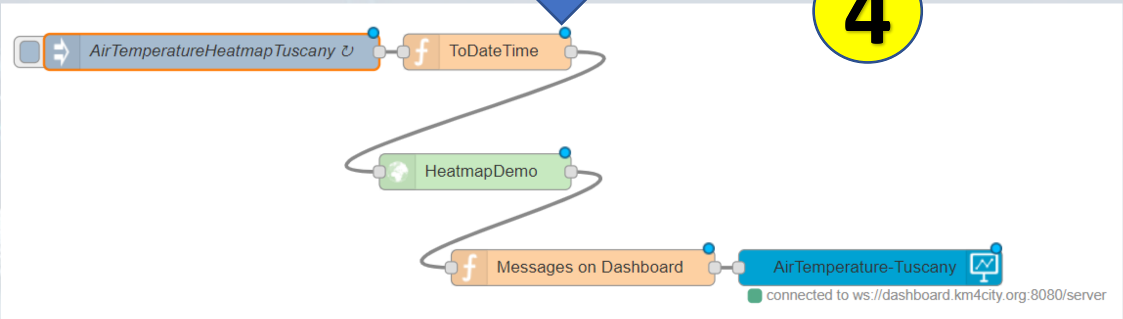




3

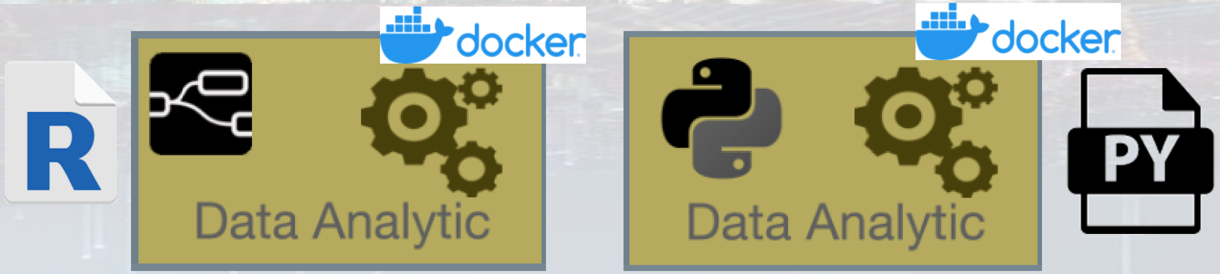
Use Snap4City Data Analytic Node, and load in the code you developed

4


AirTemperatureHeatmapTuscany
ToDateTime
HeatmapDemo
Messages on Dashboard
AirTemperature-Tuscany
connected to ws://dashboard.km4city.org:8080/server

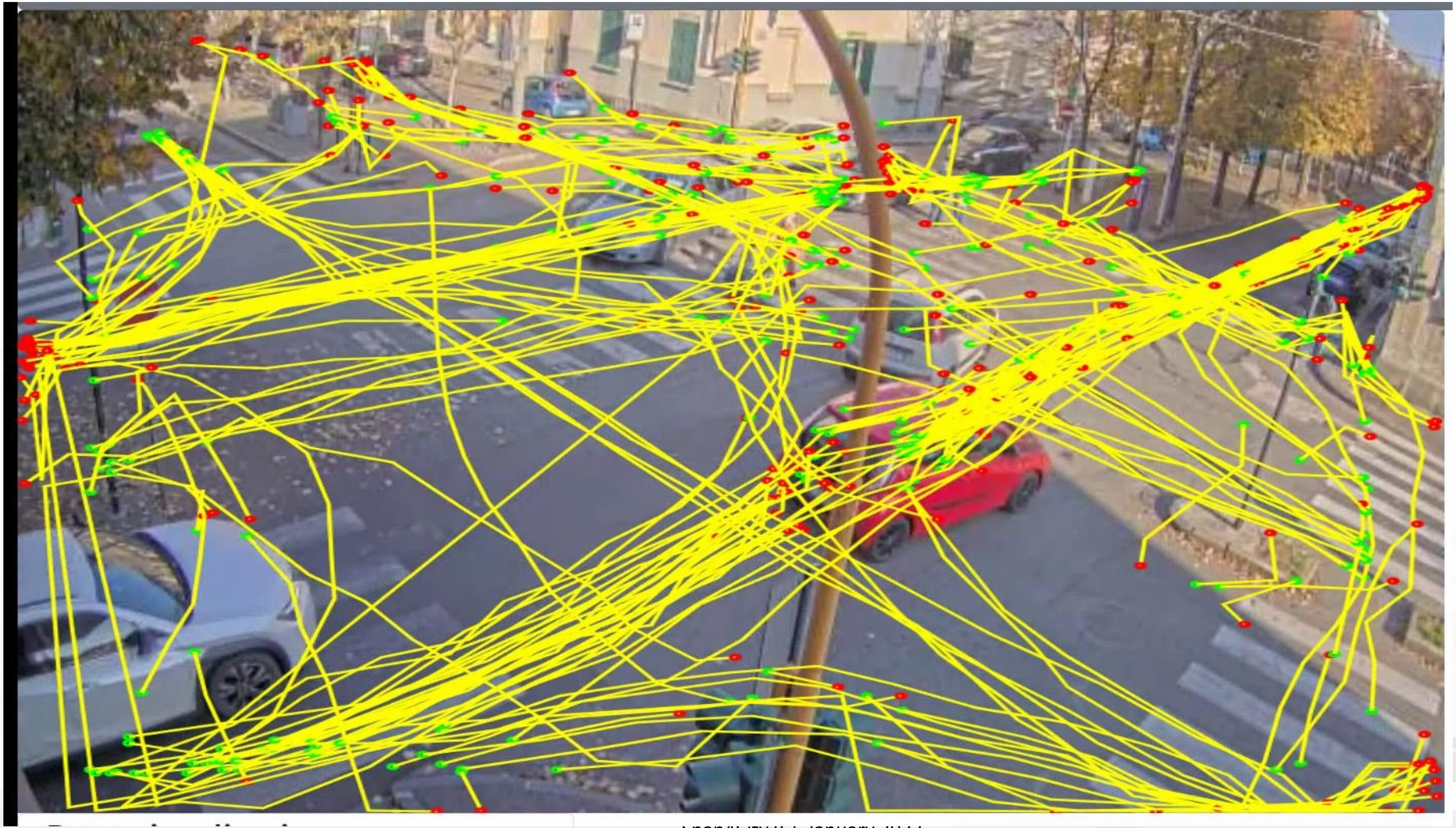
5

Deploy the IoT App → Snap4City Container Manager based on Marathon/Mesos is creating a Container for your Data Analytic code



TOP

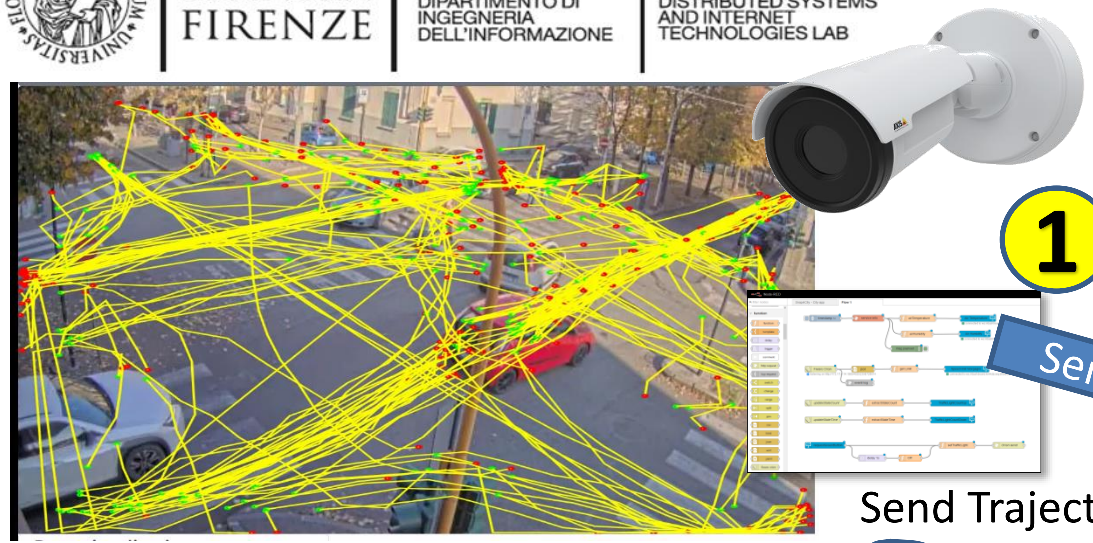
analytics example



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

AXIS COMMUNICATIONS

SNAP4CITY



IoT edge on TV Camera

1

Send Trajectories

Send data to Broker

2

Device: CrossVenaria2 with trajectories

IOT Broker

Save data

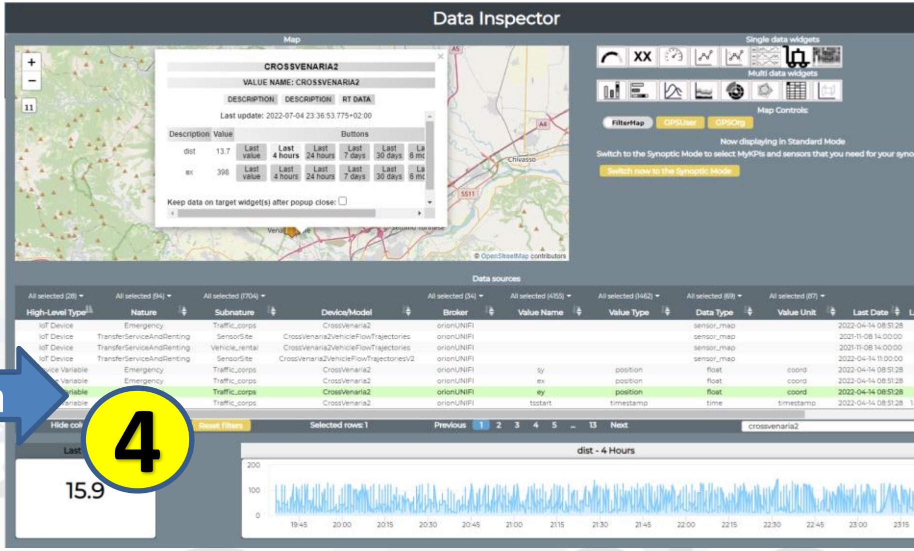
3

Big Data Store Facility

show data

4

Data Inspector



Data Inspector

CROSSVENARIA2

VALUE NAME: CROSSVENARIA2

DESCRIPTION DESCRIPTION RT DATA

Last update: 2022-07-04 23:36:53.775+02:00

Description	Value	Buttons					
dist	13.7	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 days	La 6 mo
ex	398	Last value	Last 4 hours	Last 24 hours	Last 7 days	Last 30 days	La 6 mo

Keep data on target widget(s) after popup close: □

Now displaying in Standard Mode

Switch to the Synoptic Mode to select MyKPIs and sensors that you need for your syno

Switch now to the Synoptic Mode

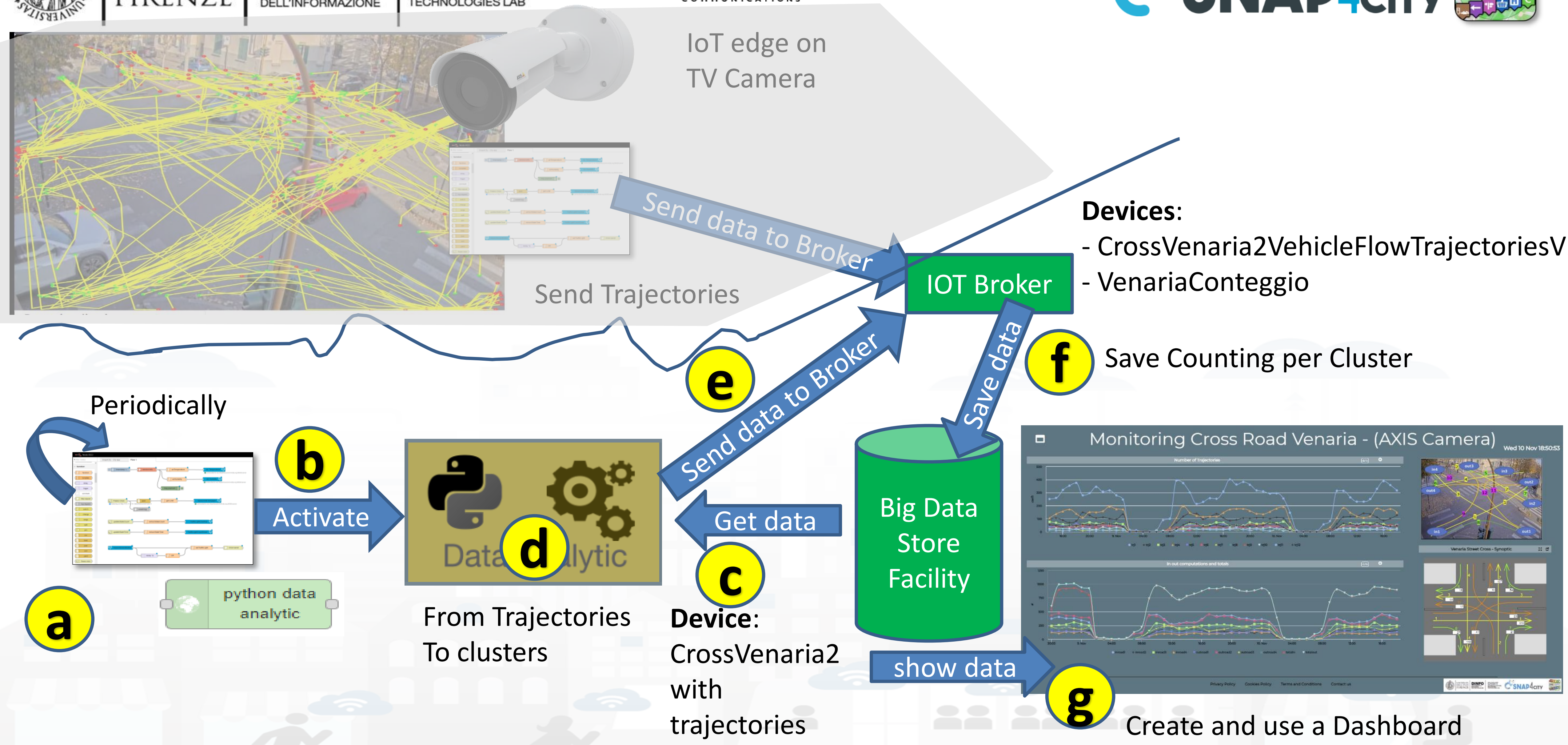
Data sources

High-Level Type	Nature	Subnature	Device/Model	Broker	Value Name	Value Type	Data Type	Value Unit	Last Date
All selected (28)	All selected (94)	All selected (1704)		All selected (34)	All selected (4155)	All selected (1462)	All selected (69)	All selected (87)	
IoT Device	Emergency	Traffic_corps	CrossVenaria2	orionUNIFI			sensor_map		2022-04-14 08:51:28
IoT Device	TransferServiceAndRenting	SensorSite	CrossVenaria2VehicleFlowTrajectories	orionUNIFI			sensor_map		2021-11-08 14:00:00
IoT Device	TransferServiceAndRenting	Vehicle_rental	CrossVenaria2VehicleFlowTrajectories	orionUNIFI			sensor_map		2021-11-08 14:00:00
IoT Device	TransferServiceAndRenting	SensorSite	CrossVenaria2VehicleFlowTrajectoriesV2	orionUNIFI			sensor_map		2022-04-14 11:00:00
ice Variable	Emergency	Traffic_corps	CrossVenaria2	orionUNIFI	sy	position	float	coord	2022-04-14 08:51:28
Variable	Emergency	Traffic_corps	CrossVenaria2	orionUNIFI	ex	position	float	coord	2022-04-14 08:51:28
riable		Traffic_corps	CrossVenaria2	orionUNIFI	ey	position	float	coord	2022-04-14 08:51:28
ariable		Traffic_corps	CrossVenaria2	orionUNIFI	tsstart	timestamp	time	timestamp	2022-04-14 08:51:28

Hide col Reset Filters Selected rows: 1 Previous 1 2 3 4 5 … 13 Next crossvenaria2

Last

15.9



Devices:
- CrossVenaria2VehicleFlowTrajectoriesV2
- VenariaConteggio

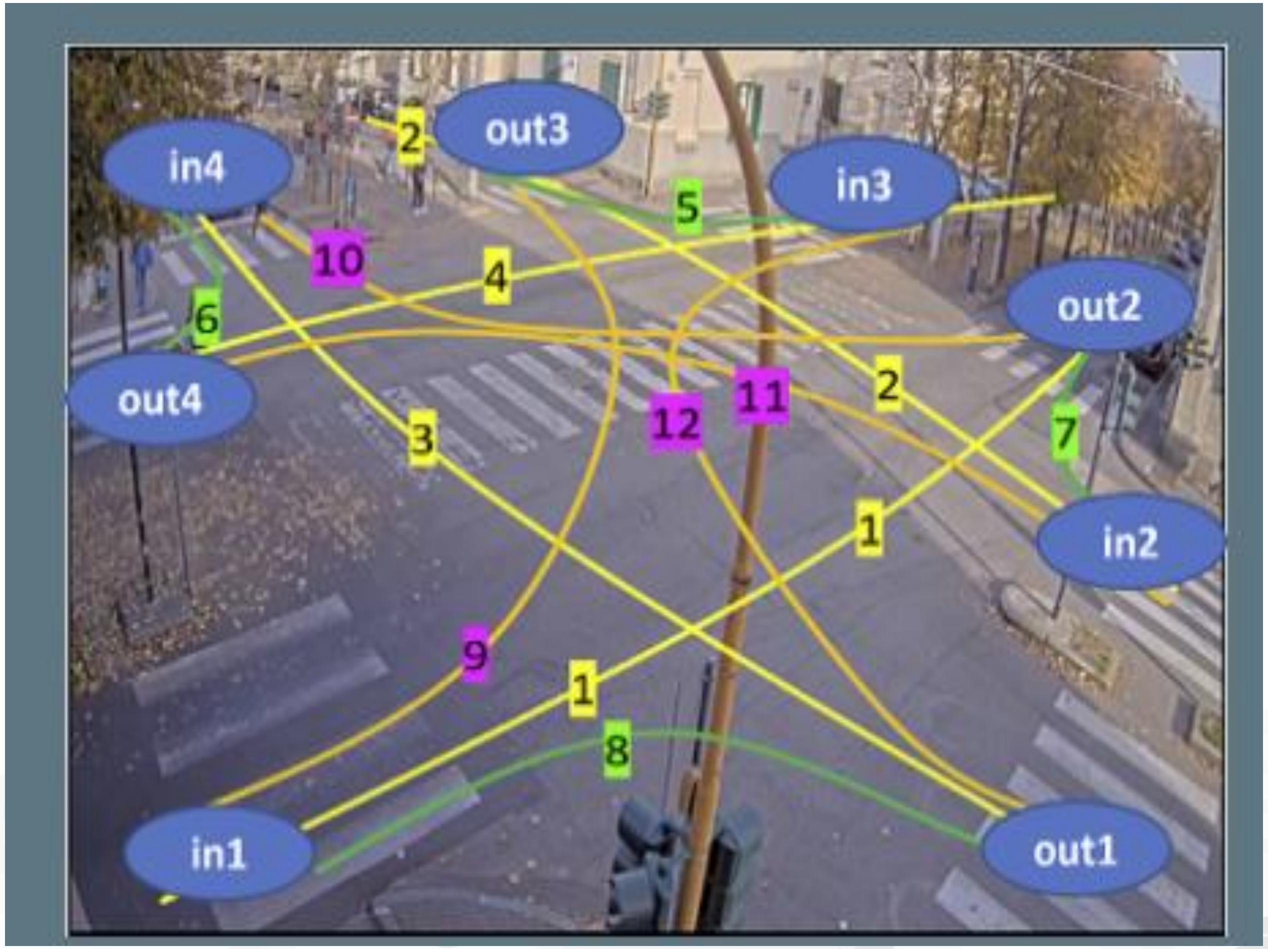
Save Counting per Cluster

From Trajectories
To clusters

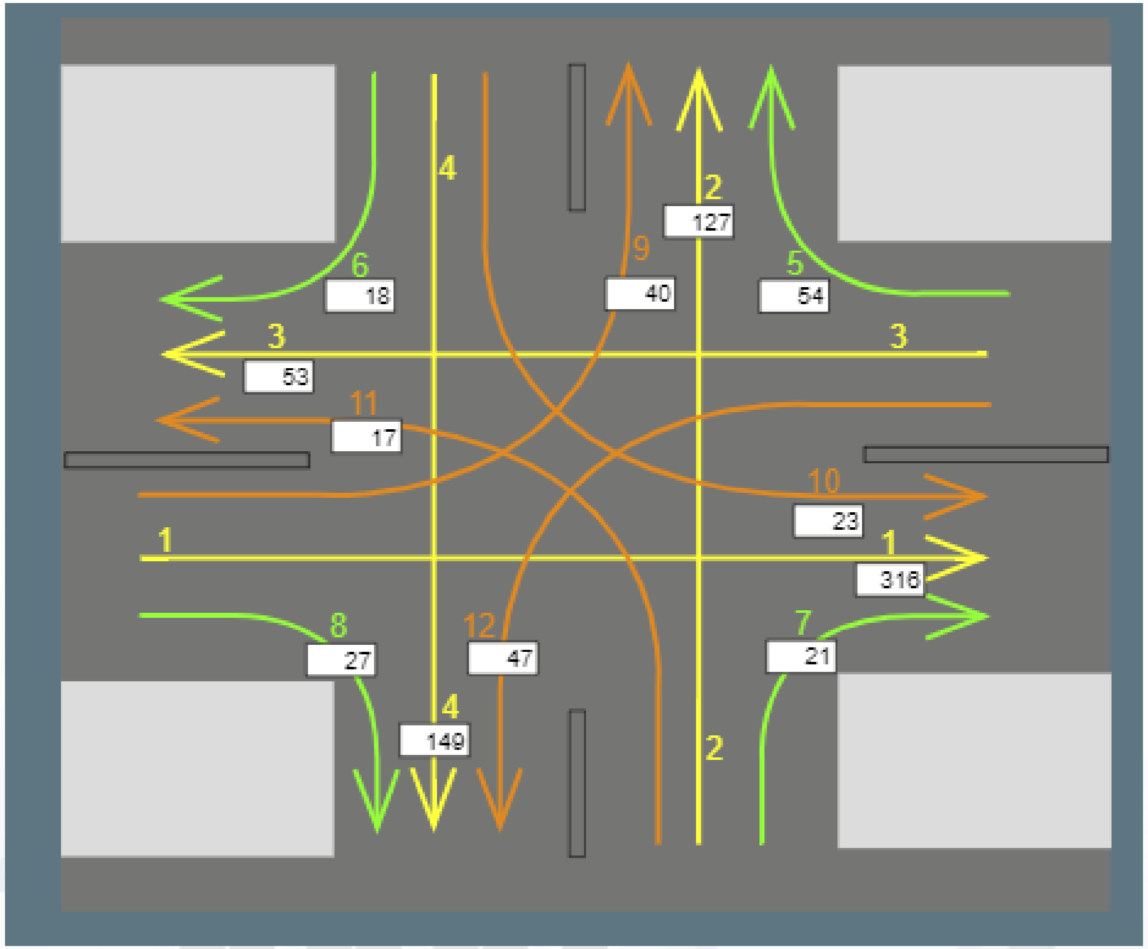
Device:
CrossVenaria2
with
trajectories

Create and use a Dashboard

Real time Clustering: legenda and synoptic



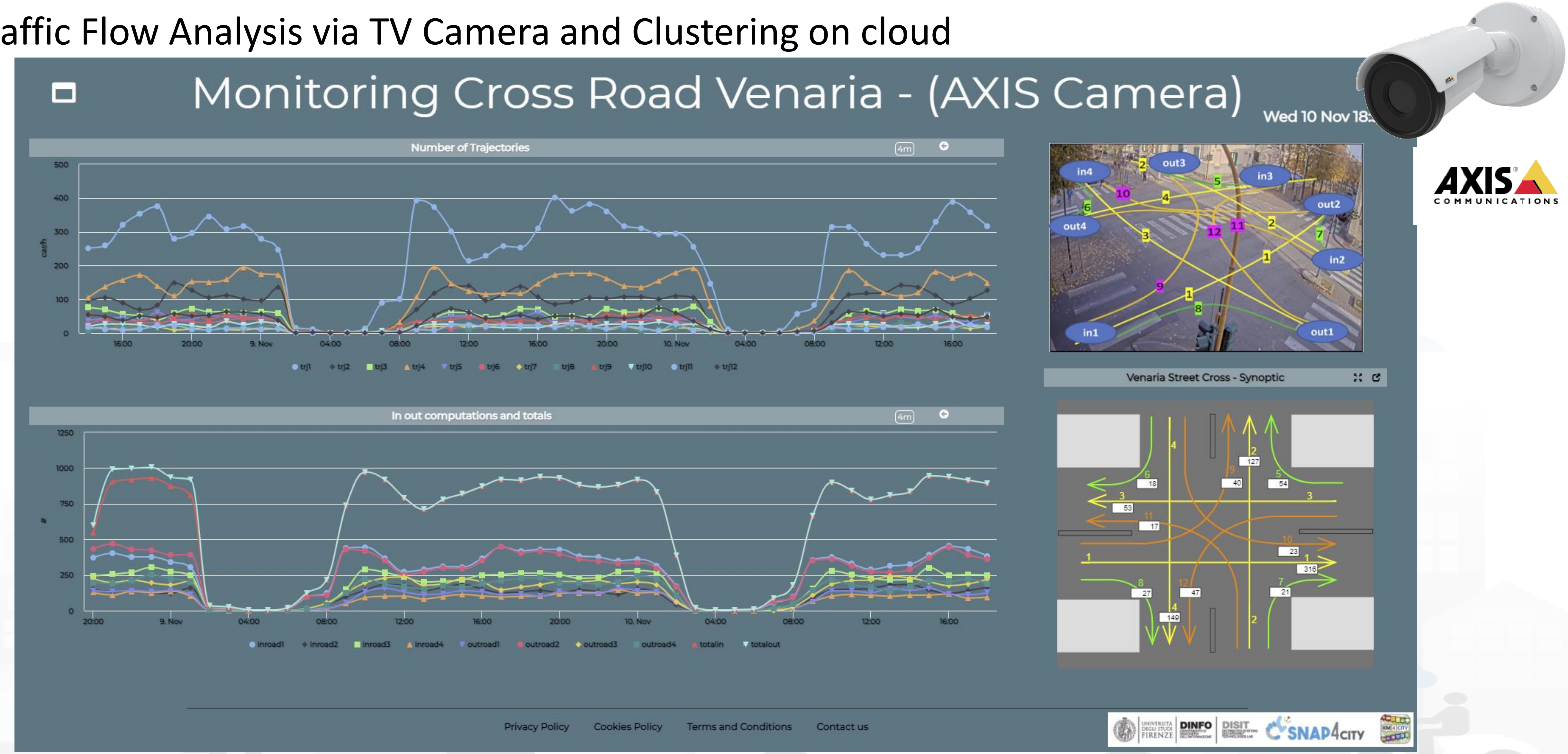
Legenda



Synoptic with real time data

An example

Traffic Flow Analysis via TV Camera and Clustering on cloud



Monitoring Cross Road Venaria - (AXIS Camera)

Wed 10 Nov 18:

Number of Trajectories

In out computations and totals

Venaria Street Cross - Synoptic

Privacy Policy Cookies Policy Terms and Conditions Contact us

TOP

Data Analytics on Dedicated Machine

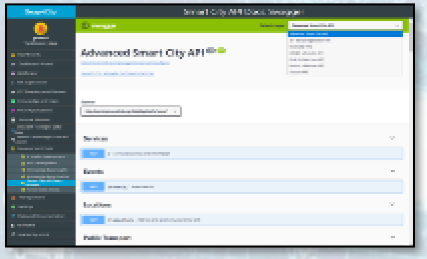
Data Analytics on Snap4City platform

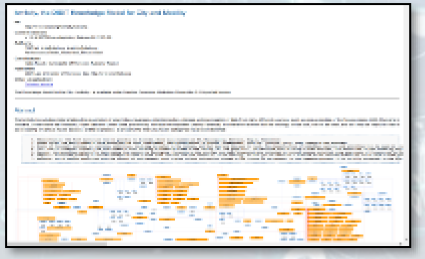
python jupyter

TensorFlow NVIDIA CUDA

R Studio

Swagger

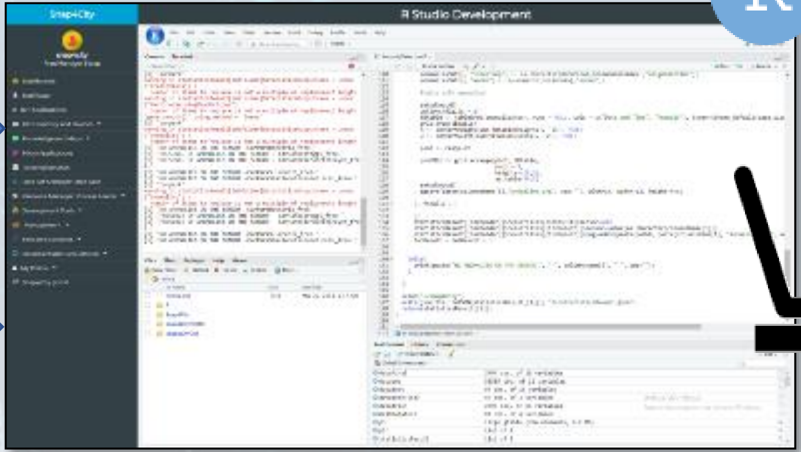




Ontology Schema

LOG.disit.org

Smart City API from Knowledge Base and other tools



Knowledge Base, Km4City

Big Data Store Facility

Saving / Sharing reusing

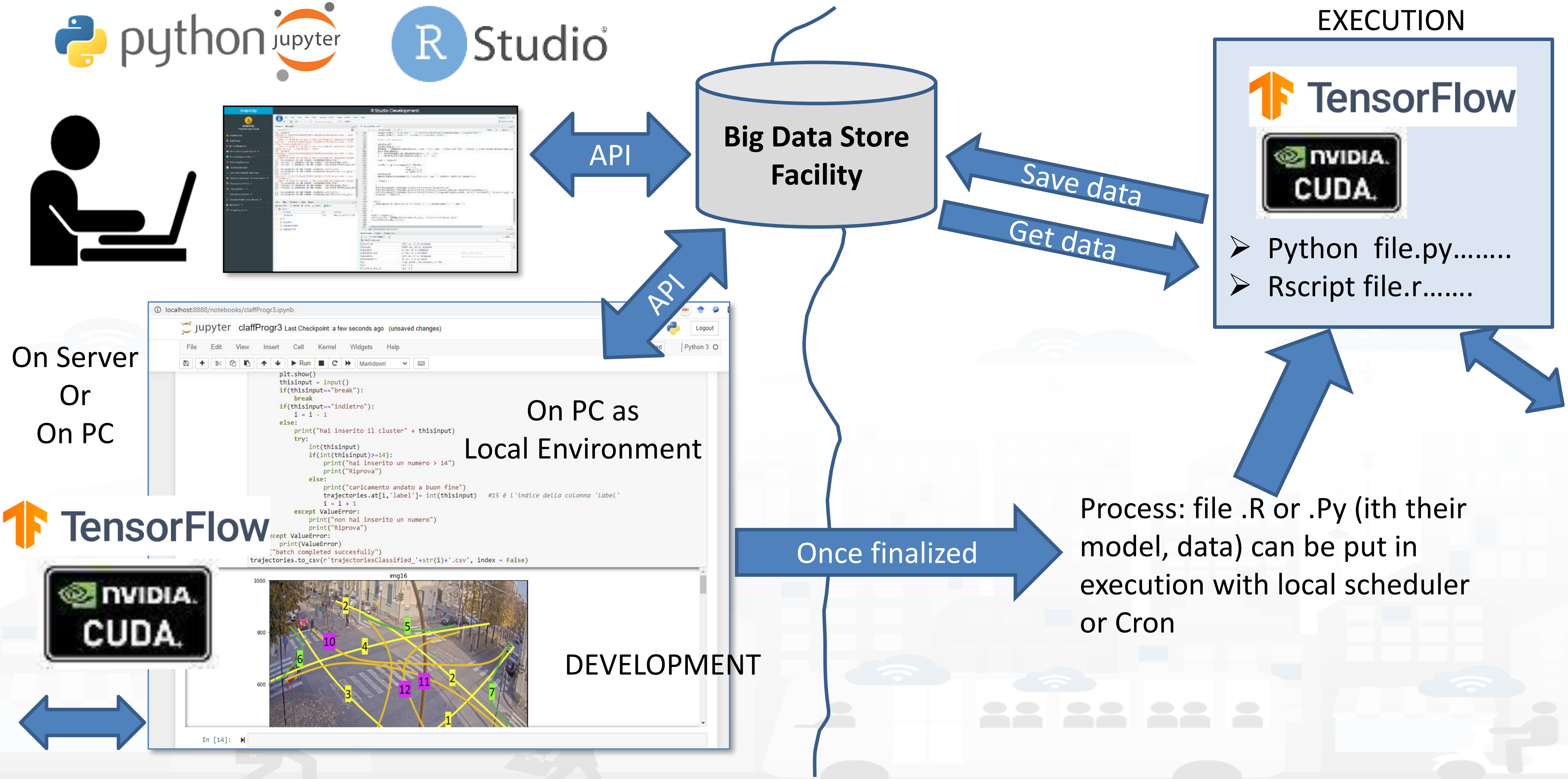
TensorFlow

NVIDIA CUDA

- Python *file.py* ....
- Rscript *file.r* ....

Resource Manager

Development



- Python file.py…….
- Rscript file.r…….

localhost:8888/notebooks/claffProgr3.ipynb

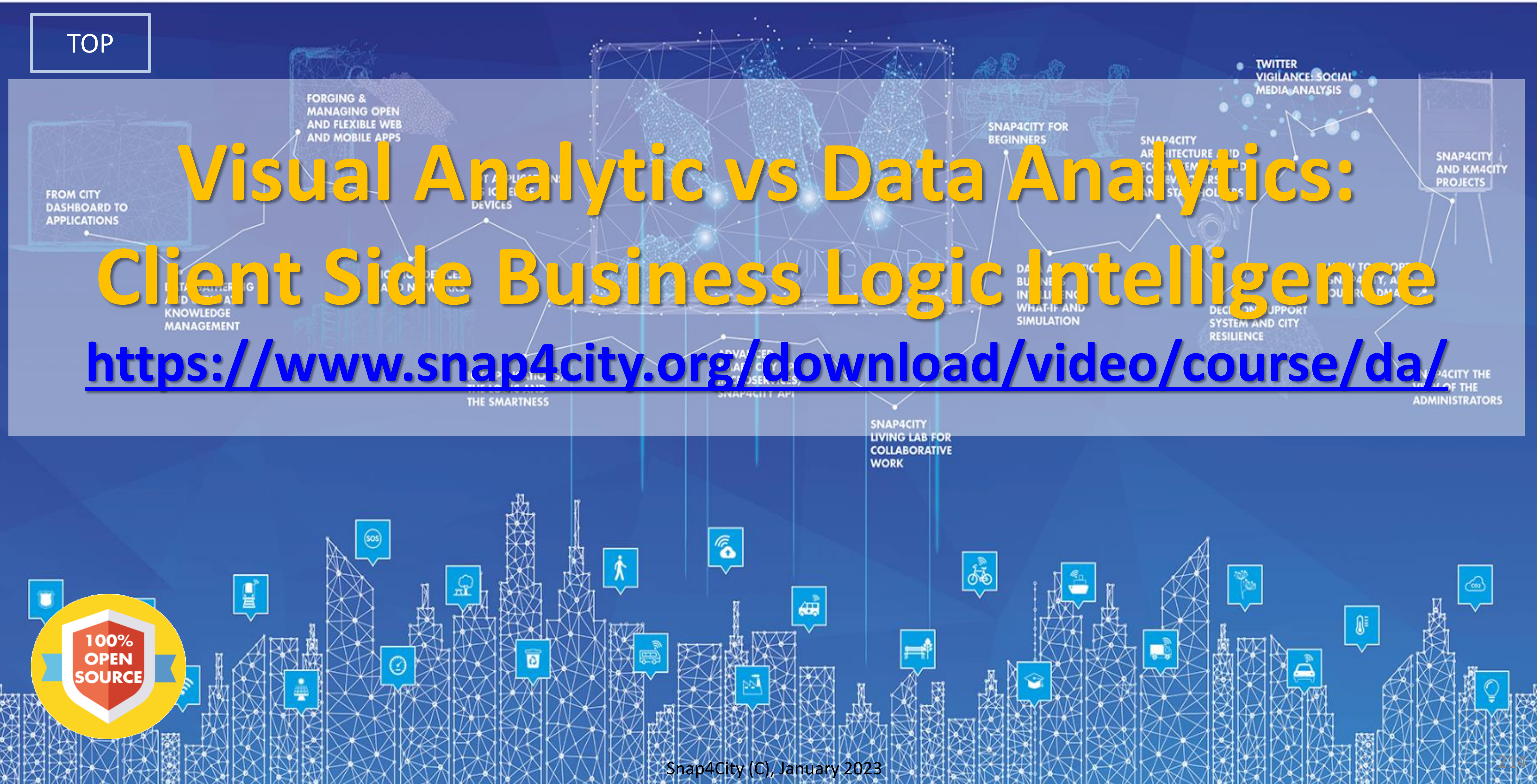
jupyter claffProgr3 Last Checkpoint: a few seconds ago (unsaved changes)

```
plt.show()
thisinput = input()
if(thisinput=="break"):
    break
if(thisinput=="indietro"):
    i = i - 1
else:
    print("hai inserito il cluster" + thisinput)
    try:
        int(thisinput)
        if(int(thisinput)>=14):
            print("hai inserito un numero > 14")
            print("Riprova")
        else:
            print("caricamento andato a buon fine")
            trajectories.at[i,'label']= int(thisinput)   #15 è l'indice della colonna 'label'
            i = i + 1
    except ValueError:
        print("non hai inserito un numero")
        print("Riprova")
ccept ValueError:
    print(ValueError)
("batch completed succesfully")
trajectories.to_csv(r'trajectoriesClassified_'+str(i)+'.csv', index = False)
```

```
In [14]:
```

Process: file .R or .Py (ith their model, data) can be put in execution with local scheduler or Cron

TOP

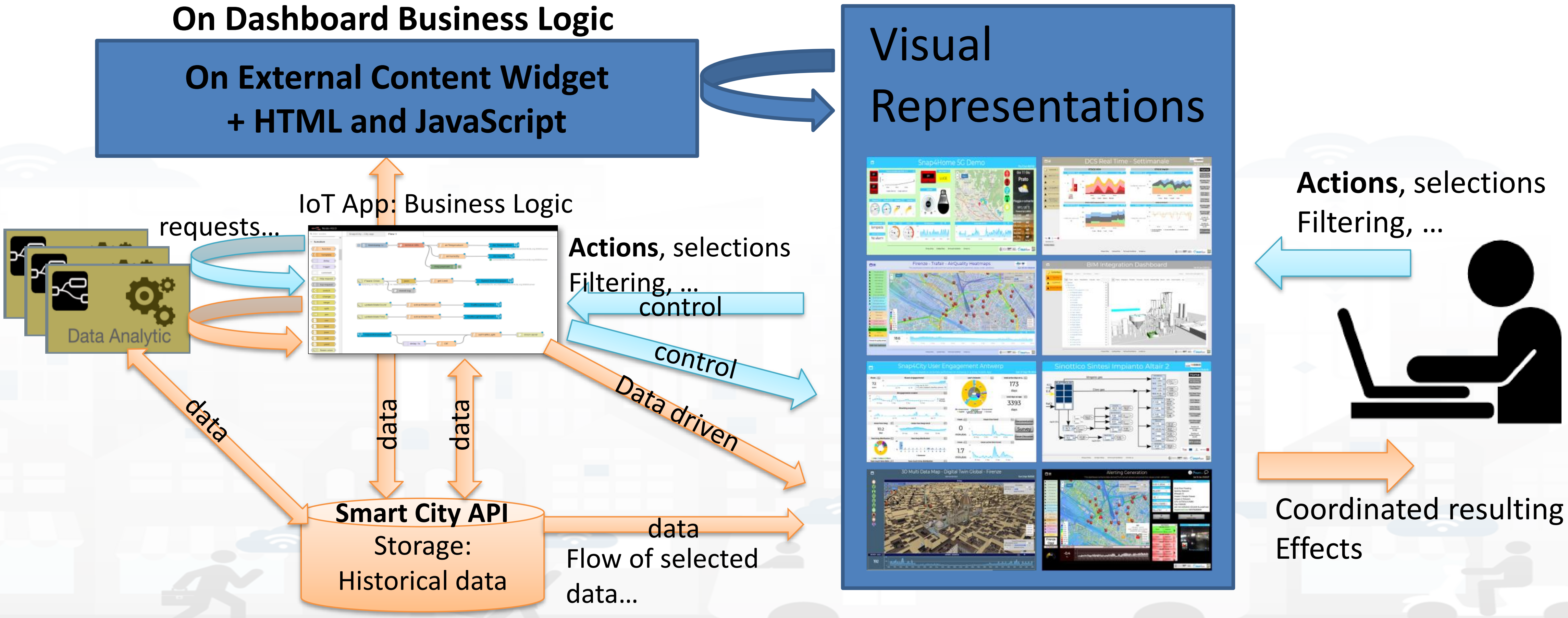

FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS
FROM CITY DASHBOARD TO APPLICATIONS
KNOWLEDGE MANAGEMENT
DEVICES
THE SMARTNESS
SNAP4CITY API
SNAP4CITY FOR BEGINNERS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY AND KM4CITY PROJECTS
WHAT-IF AND SIMULATION
SYSTEM AND CITY RESILIENCE
ADMINISTRATORS
SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK
100% OPEN SOURCE

Visual Analytic vs Data Analytics: Client Side Business Logic Intelligence

https://www.snap4city.org/download/video/course/da/

Visual Analytics

- implementing sophisticated ***Business Intelligence Tools***
- Open to receive a range of possible Actions, to produce a large combination of results in terms of data and representations.



On Dashboard Business Logic

On External Content Widget + HTML and JavaScript

Visual Representations

IoT App: Business Logic

requests…

Data Analytic

Actions, selections Filtering, …

control

control

Data driven

data

data

data

Smart City API
Storage:
Historical data

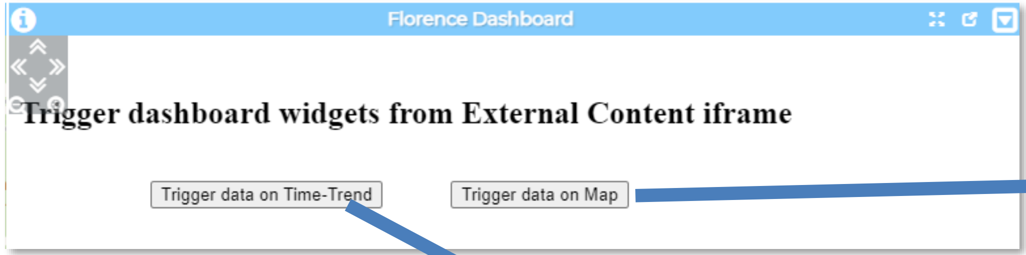
data

Flow of selected data…

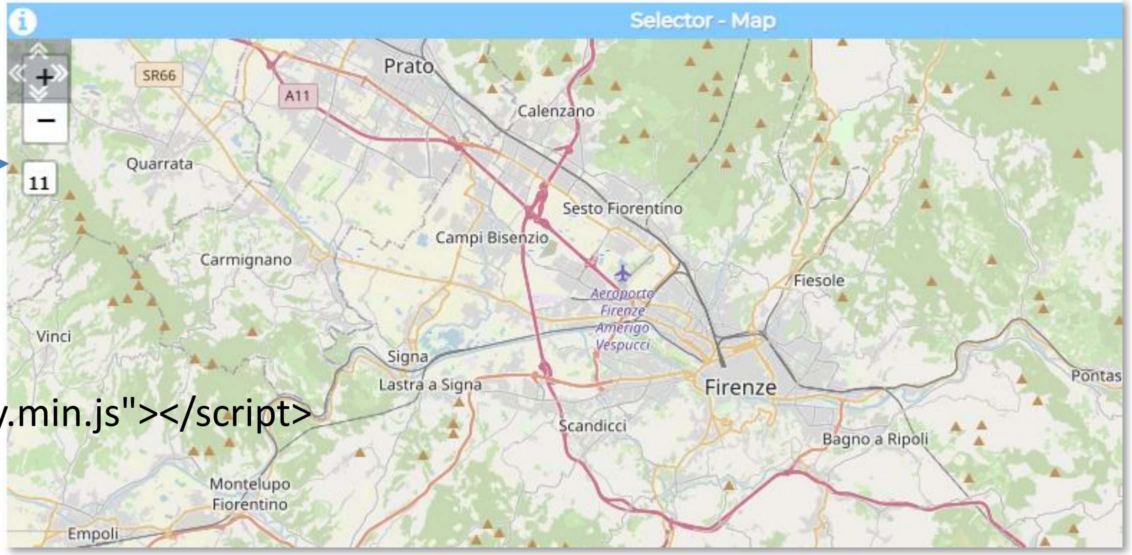
Actions, selections Filtering, …

Coordinated resulting Effects

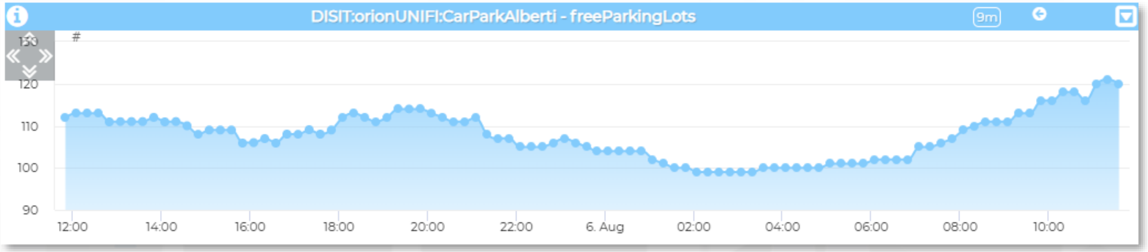
Snap4Home 5G Demo
DCS Real Time - Settimanale
Firenze - Traffic - AirQuality Heatmaps
BIM Integration Dashboard
Snap4City User Engagement Antwerp
Sinottico Sintesi Impianto Altair 2
3D Multi Data Map - Digital Twin Global - Firenze
Alerting Generation



Trigger dashboard widgets from External Content iframe



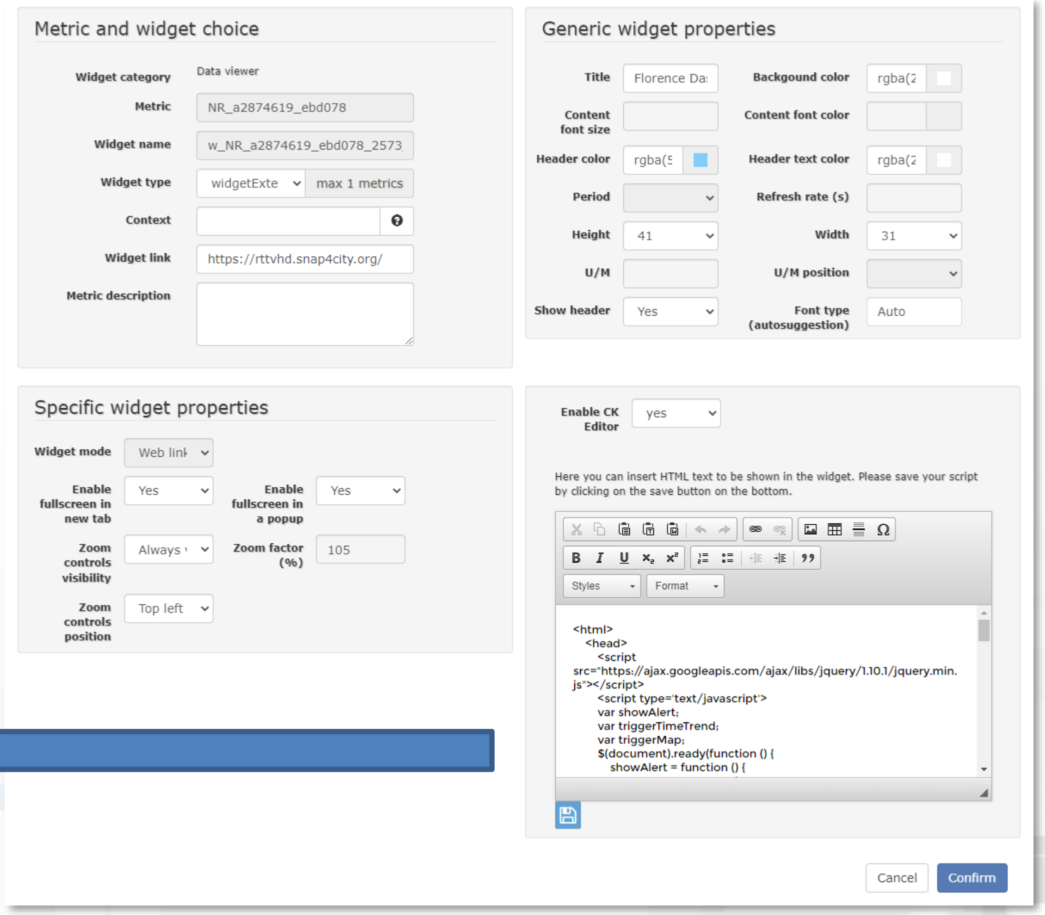
```
<html>
   <head>
      <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js"></script>
      <script type='text/javascript’>
                 …….
      </script>
   </head>
   <body>
      <h2>Trigger dashboard widgets from External Content iframe</h2>
      <div>
         <!-- <button onclick="showAlert()">Alert Button GP</button>    -->
         <button id="triggerTTrend">Trigger data on Time-Trend</button>
         <button id="triggerMap">Trigger data on Map</button>
      </div>
   </body>
</html>
```



Trigger based

Enforcing HTML and JavaScript on MoreOptions of the External Content Widget

```
<script type='text/javascript'>
var showAlert;
var triggerTimeTrend;
var triggerMap;
$(document).ready(function () {
   showAlert = function () {
      var myText = "Test alert";
      alert (myText);
   }
   $('#triggerTTrend').click(function (event) {
      ................
      parent.$('body').trigger({      ................          });
   });
   $('#triggerMap').click(function (event) {
      ................
      parent.$('body').trigger({       ................          });
   });
});
</script>
```



Metric and widget choice

Widget category: Data viewer

Metric: NR_a2874619_ebd078

Widget name: w_NR_a2874619_ebd078_2573

Widget type: widgetExte | max 1 metrics

Context:

Widget link: https://rttvhd.snap4city.org/

Metric description:

Generic widget properties

Title: Florence Da

Backgound color: rgba(2

Content font size:

Content font color:

Header color: rgba(5

Header text color: rgba(2

Period:

Refresh rate (s):

Height: 41

Width: 31

U/M:

U/M position:

Show header: Yes

Font type (autosuggestion): Auto

Specific widget properties

Widget mode: Web link

Enable fullscreen in new tab: Yes

Enable fullscreen in a popup: Yes

Zoom controls visibility: Always v

Zoom factor (%): 105

Zoom controls position: Top left

Enable CK Editor: yes

Here you can insert HTML text to be shown in the widget. Please save your script by clicking on the save button on the bottom.

```
<html>
  <head>
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.
js"></script>
    <script type='text/javascript'>
    var showAlert;
    var triggerTimeTrend;
    var triggerMap;
    $(document).ready(function () {
      showAlert = function () {
```

Trigger map

```
$('#triggerMap').click(function (event) {
        let coordsAndType = {};
        coordsAndType.eventGenerator = $(this);
        coordsAndType.desc = "CarPark";
        coordsAndType.query =
"https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.64471;11.005751;43.89471;11.505751&categories=Car_park&maxResults=200&format=json&model=CarPark";
        coordsAndType.color1 = "#ebb113";
        coordsAndType.color2 = "#eb8a13";
        coordsAndType.targets = "w_DISIT_orionUNIFI_CarParkAlberti_2573_widgetTimeTrend33703";    // the Time Trend Widget ID once pop up open
        coordsAndType.display = "pins";
        coordsAndType.queryType = "Default";
        coordsAndType.iconTextMode = "text";
        coordsAndType.pinattr = "square";
        coordsAndType.pincolor = "#959595";
        coordsAndType.symbolcolor = "undefined";
        // coordsAndType.altViewMode = altViewMode;
        coordsAndType.bubbleSelectedMetric = "";
        parent.$('body').trigger({
          type: "addSelectorPin",
          target: "w_Map_2573_widgetMap33705", // the Time Trend Widget ID of the event performed on clik
          passedData: coordsAndType
        });
      });
```

Trigger Time trend

```
    $('#triggerTTrend').click(function (event) {
        parent.$('body').trigger({
            type:
"showTimeTrendFromExternalContentGis_w_DISIT_orionUNIFI_CarParkAlberti_2573_widgetTimeTrend33703",
            eventGenerator: $(this),
            targetWidget: "w_DISIT_orionUNIFI_CarParkAlberti_2573_widgetTimeTrend33703",
            range: "7/DAY",
            color1: "#34eb6e",
            color2: "#114a23",
            widgetTitle: "Free Parking Lots data from External Content",
            field: "freeParkingLots",
            serviceUri: "http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/CarParkPal.Giustizia",
            marker: "",
            mapRef: "",
            fake: false
        });
    });
```

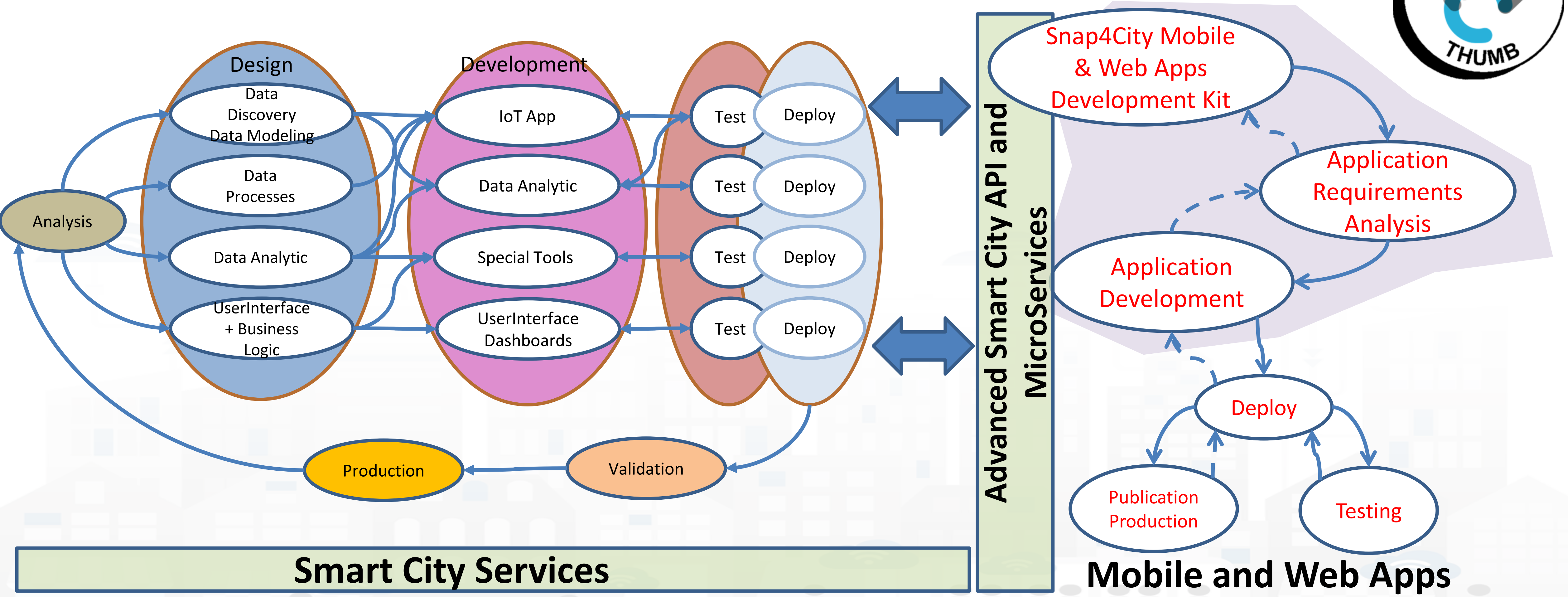
TOP

Design & Develop Web and mobile Apps

https://www.snap4city.org/download/video/course/app/

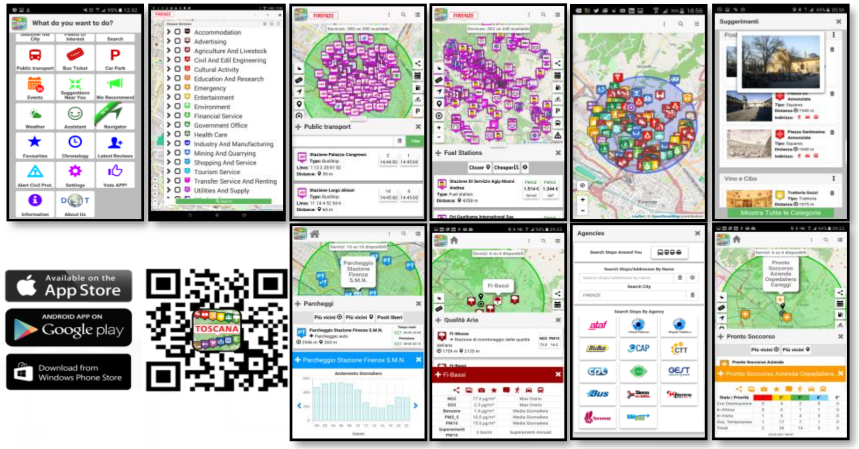
FROM CITY DASHBOARD TO APPLICATIONS
DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT
FORGING & MANAGING OPEN AND FLEX
IOT/I DEVICES
IOT APPLICATIONS,
ADVANCED SMART CITY API, MICROSERVICES,
LIVING LAB
LIVING LAB FOR COLLABORATIVE WORK
DATA ANALYTICS, BUSINESS INTELLIGENCE, WHAT-IF SIMULATION
RESILIENCE
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY AND KM4CITY PROJECTS
HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP
100% OPEN SOURCE

Develop Mobile & Web Applications Exploiting Snap4City Smart City Services

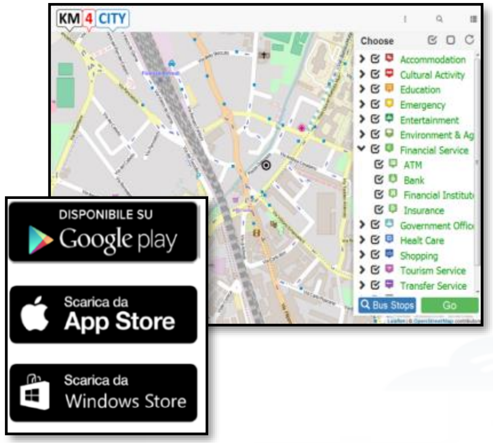


Developing Web and Mobile Apps, MicroApps,..

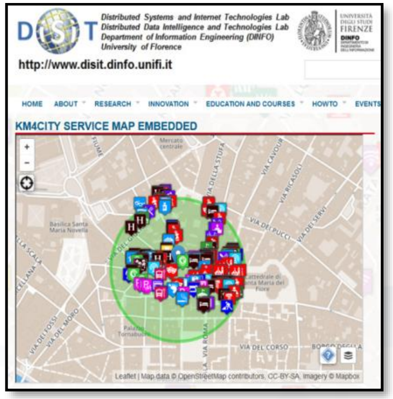
Mobile Apps


What do you want to do?
Available on the App Store
ANDROID APP ON Google play
Download from Windows Phone Store
TOSCANA

Web App HTML5, MicroApplications

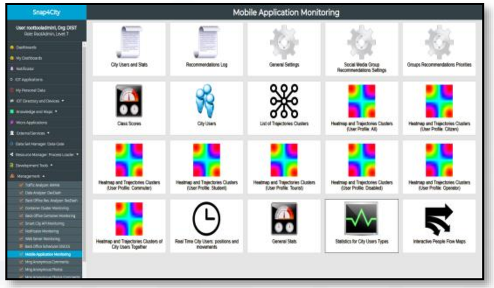


Embed into Web pages

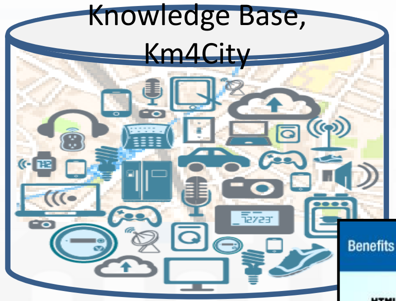

Distributed Systems and Internet Technologies Lab
Distributed Data Intelligence and Technologies Lab
Department of Information Engineering (DINFO)
University of Florence
http://www.disit.dinfo.unifi.it
HOME ABOUT RESEARCH INNOVATION EDUCATION AND COURSES HOWTO EVENTS
KM4CITY SERVICE MAP EMBEDDED

City User

Advanced Smart City API



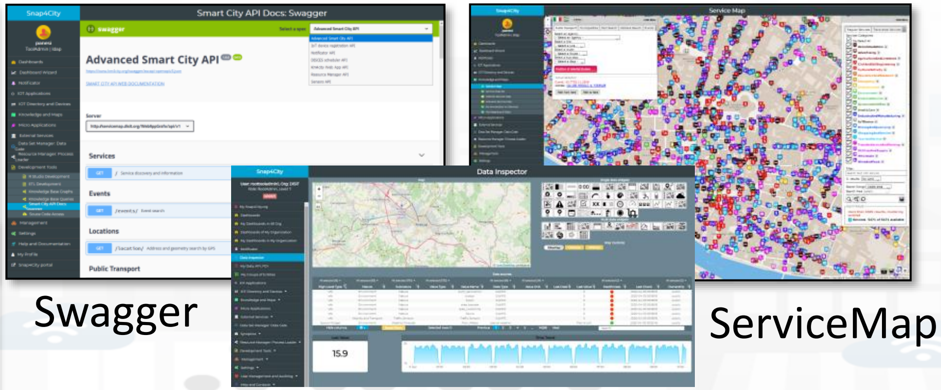
Mobile Application Monitoring Administrator



GitHub

Snap/Km4City Open Source development tool kit

Benefits of CORDOVA on Cross Platform Mobile App Development
PhoneGap
HTML 5
CSS 3
JS
APACHE CORDOVA™



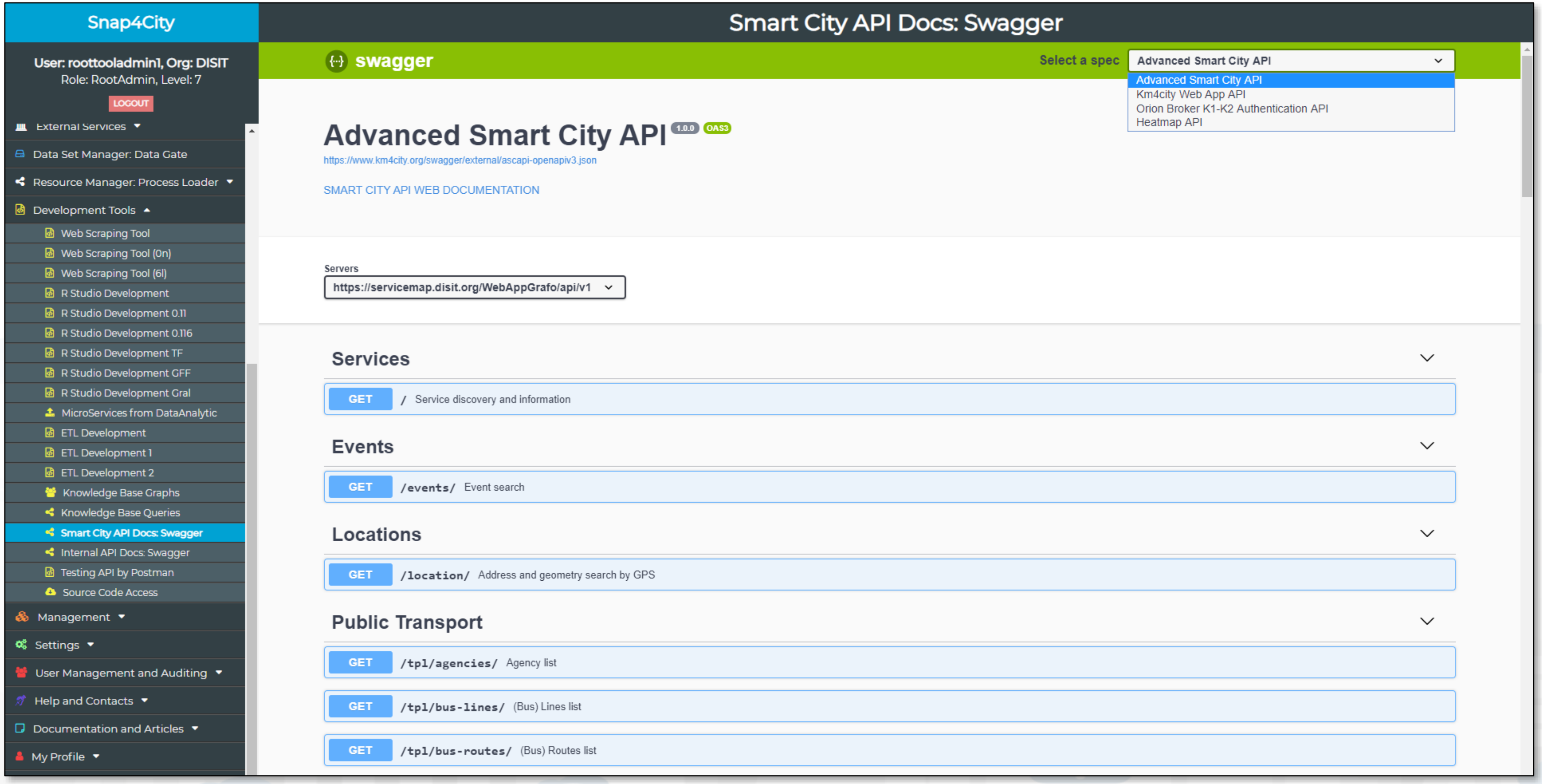
Swagger

ServiceMap

DataInspector

Developer

External Smart City API



Smart City API Docs: Swagger

Advanced Smart City API 1.0.0 OAS3

https://www.km4city.org/swagger/external/ascapi-openapiv3.json

SMART CITY API WEB DOCUMENTATION

Servers
https://servicemap.disit.org/WebAppGrafo/api/v1

Services

GET `/` Service discovery and information

Events

GET `/events/` Event search

Locations

GET `/location/` Address and geometry search by GPS

Public Transport

GET `/tpl/agencies/` Agency list

GET `/tpl/bus-lines/` (Bus) Lines list

GET `/tpl/bus-routes/` (Bus) Routes list

https://www.km4city.org/swagger/external/index.html

TOP

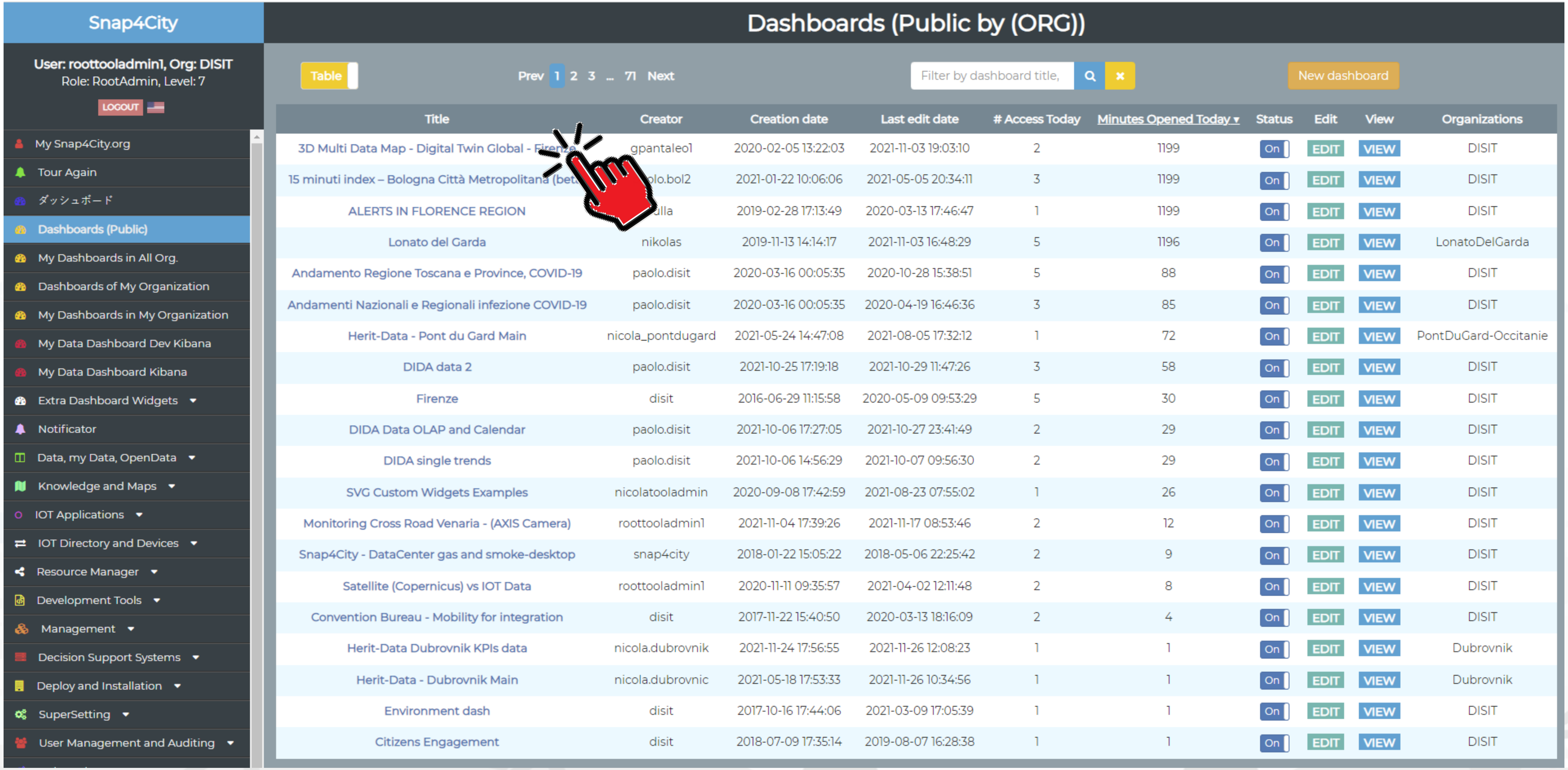
Design and Control of Smart Applications

only for user with RootAdmin role

partially accessible also for all Dashboard owners

FORGING & MANAGING OPEN AND FLEXIBLE WEB
FROM CITY DASHBOARD TO APPLICATIONS
DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT
IOT APPLICATIONS
ADVANCED SMART CITY API
SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK
DATA ANALYTICS, BUSINESS INTELLIGENCE, WHAT-IF AND SIMULATION
SNAP4CITY ARCHITECTURE AND ECOSYSTEM OPEN TO DEVELOPERS AND STAKEHOLDERS
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
SNAP4CITY AND KM4CITY PROJECTS
HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP
DECISION SUPPORT SYSTEM AND CITY RESILIENCE
SNAP4CITY THE VIEW OF THE ADMINISTRATORS
LIVING LAB

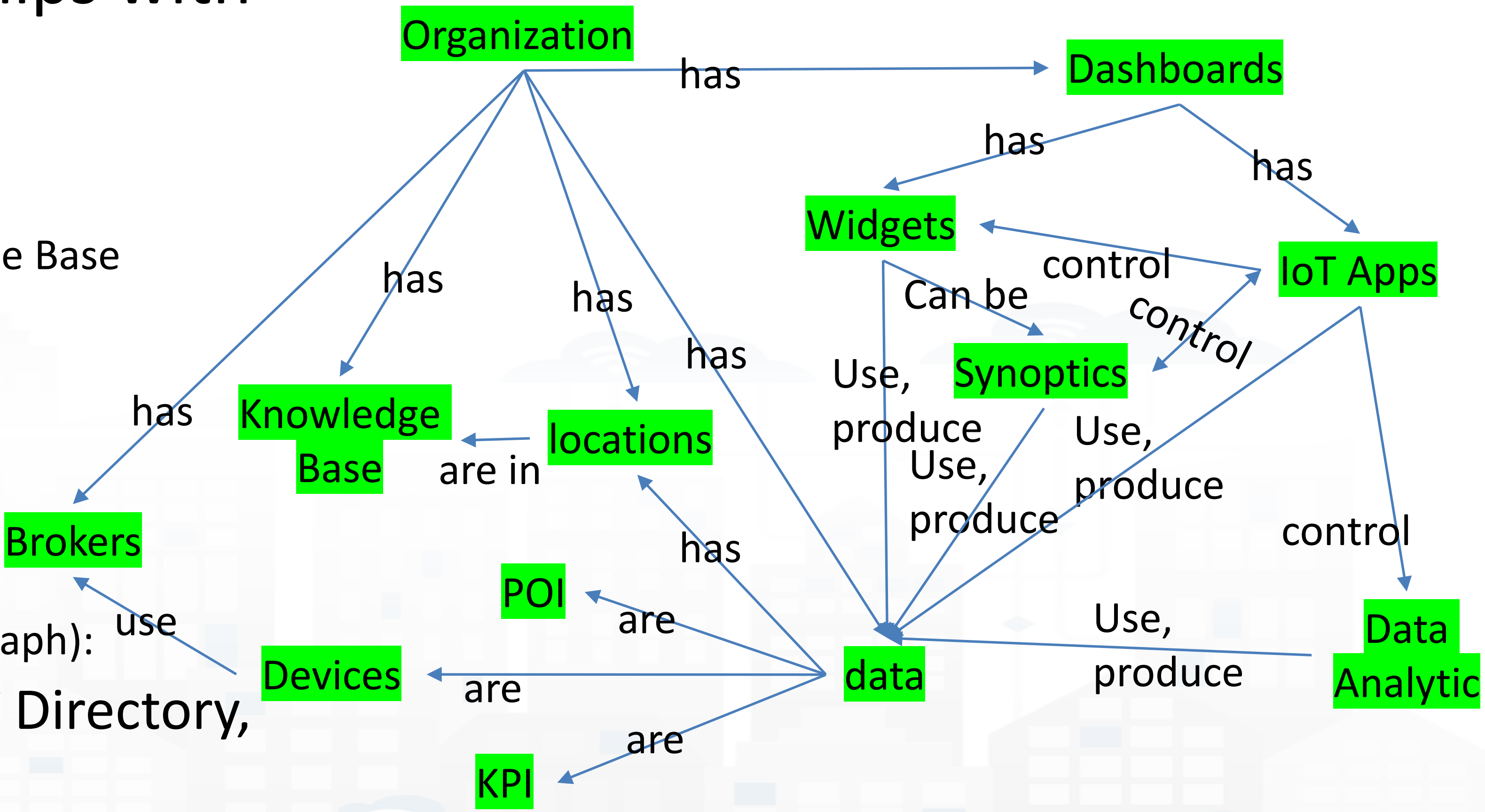
Dashboard manager for RootAdmin



Title	Creator	Creation date	Last edit date	# Access Today	Minutes Opened Today ▾	Status	Edit	View	Organizations
3D Multi Data Map - Digital Twin Global - Firenze	gpantaleo1	2020-02-05 13:22:03	2021-11-03 19:03:10	2	1199	On	EDIT	VIEW	DISIT
15 minuti index – Bologna Città Metropolitana (bet	olo.bol2	2021-01-22 10:06:06	2021-05-05 20:34:11	3	1199	On	EDIT	VIEW	DISIT
ALERTS IN FLORENCE REGION	ulla	2019-02-28 17:13:49	2020-03-13 17:46:47	1	1199	On	EDIT	VIEW	DISIT
Lonato del Garda	nikolas	2019-11-13 14:14:17	2021-11-03 16:48:29	5	1196	On	EDIT	VIEW	LonatoDelGarda
Andamento Regione Toscana e Province, COVID-19	paolo.disit	2020-03-16 00:05:35	2020-10-28 15:38:51	5	88	On	EDIT	VIEW	DISIT
Andamenti Nazionali e Regionali infezione COVID-19	paolo.disit	2020-03-16 00:05:35	2020-04-19 16:46:36	3	85	On	EDIT	VIEW	DISIT
Herit-Data - Pont du Gard Main	nicola_pontdugard	2021-05-24 14:47:08	2021-08-05 17:32:12	1	72	On	EDIT	VIEW	PontDuGard-Occitanie
DIDA data 2	paolo.disit	2021-10-25 17:19:18	2021-10-29 11:47:26	3	58	On	EDIT	VIEW	DISIT
Firenze	disit	2016-06-29 11:15:58	2020-05-09 09:53:29	5	30	On	EDIT	VIEW	DISIT
DIDA Data OLAP and Calendar	paolo.disit	2021-10-06 17:27:05	2021-10-27 23:41:49	2	29	On	EDIT	VIEW	DISIT
DIDA single trends	paolo.disit	2021-10-06 14:56:29	2021-10-07 09:56:30	2	29	On	EDIT	VIEW	DISIT
SVG Custom Widgets Examples	nicolatooladmin	2020-09-08 17:42:59	2021-08-23 07:55:02	1	26	On	EDIT	VIEW	DISIT
Monitoring Cross Road Venaria - (AXIS Camera)	roottooladmin1	2021-11-04 17:39:26	2021-11-17 08:53:46	2	12	On	EDIT	VIEW	DISIT
Snap4City - DataCenter gas and smoke-desktop	snap4city	2018-01-22 15:05:22	2018-05-06 22:25:42	2	9	On	EDIT	VIEW	DISIT
Satellite (Copernicus) vs IOT Data	roottooladmin1	2020-11-11 09:35:57	2021-04-02 12:11:48	2	8	On	EDIT	VIEW	DISIT
Convention Bureau - Mobility for integration	disit	2017-11-22 15:40:50	2020-03-13 18:16:09	2	4	On	EDIT	VIEW	DISIT
Herit-Data Dubrovnik KPIs data	nicola.dubrovnik	2021-11-24 17:56:55	2021-11-26 12:08:23	1	1	On	EDIT	VIEW	Dubrovnik
Herit-Data - Dubrovnik Main	nicola.dubrovnic	2021-05-18 17:53:33	2021-11-26 10:34:56	1	1	On	EDIT	VIEW	Dubrovnik
Environment dash	disit	2017-10-16 17:44:06	2021-03-09 17:05:39	1	1	On	EDIT	VIEW	DISIT
Citizens Engagement	disit	2018-07-09 17:35:14	2019-08-07 16:28:38	1	1	On	EDIT	VIEW	DISIT

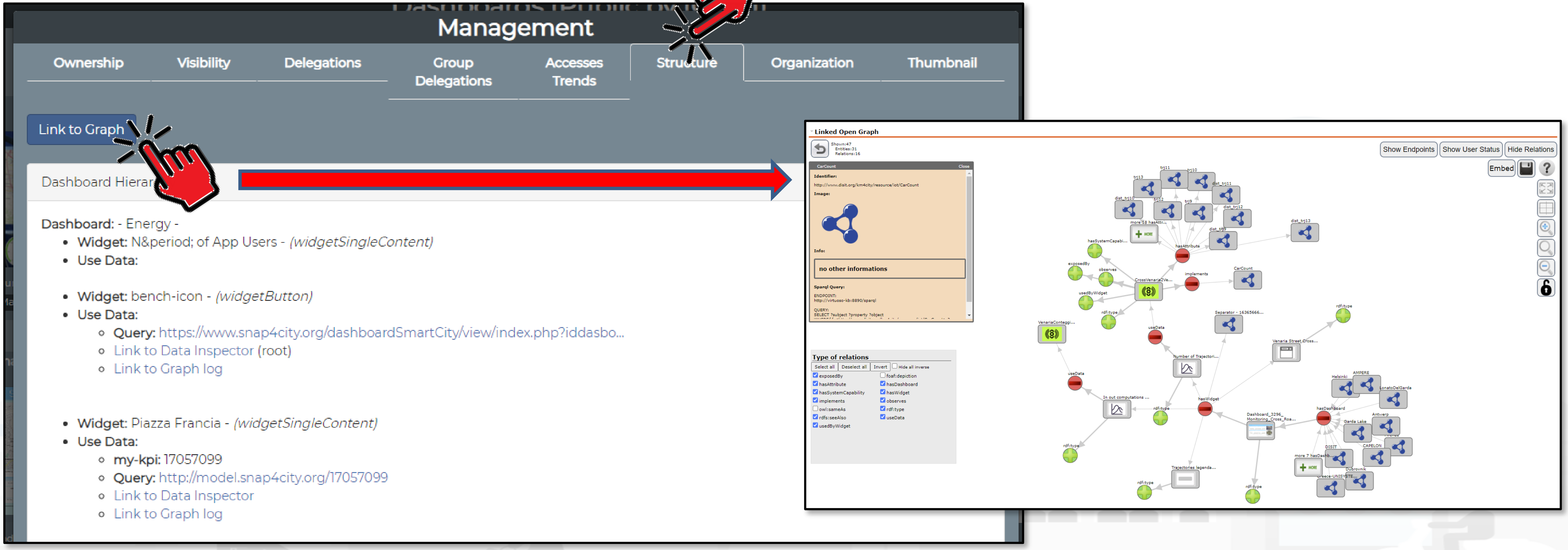
Semantic Reasoning on Smart Applications

- Dashboards have relationships with
  - Org. at which they belong
  - Widgets with
    - data they use, and each of which
      - is connected with the Knowledge Base
      - May be: device, kpi, etc.
  - IoT Apps with
    - Data they use
    - Data Analytic
    - Widget they control
- Processes are (not in the simplified graph):
  - Data, Broker, Data Analytic, IoT Directory, Device, IoT App, UserInterface
  - owned, and delegated in some manner from the owner to other users



For All Dashboard owners: Graph and Structure

- Go on Dashboard Management



Dashboard: - Energy -

- Widget: N&period; of App Users - *(widgetSingleContent)*
- Use Data:

- Widget: bench-icon - *(widgetButton)*
- Use Data:
  - Query: https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasbo...
  - Link to Data Inspector (root)
  - Link to Graph log

- Widget: Piazza Francia - *(widgetSingleContent)*
- Use Data:
  - my-kpi: 17057099
  - Query: http://model.snap4city.org/17057099
  - Link to Data Inspector
  - Link to Graph log

Linked Open Graph

Shown:47
Entities:31
Relations:16

CarCount Close

Identifier:
http://www.disit.org/km4city/resource/iot/CarCount

Image:

Info:

no other informations

Sparql Query:

ENDPOINT:
http://virtuoso-kb:8890/sparql

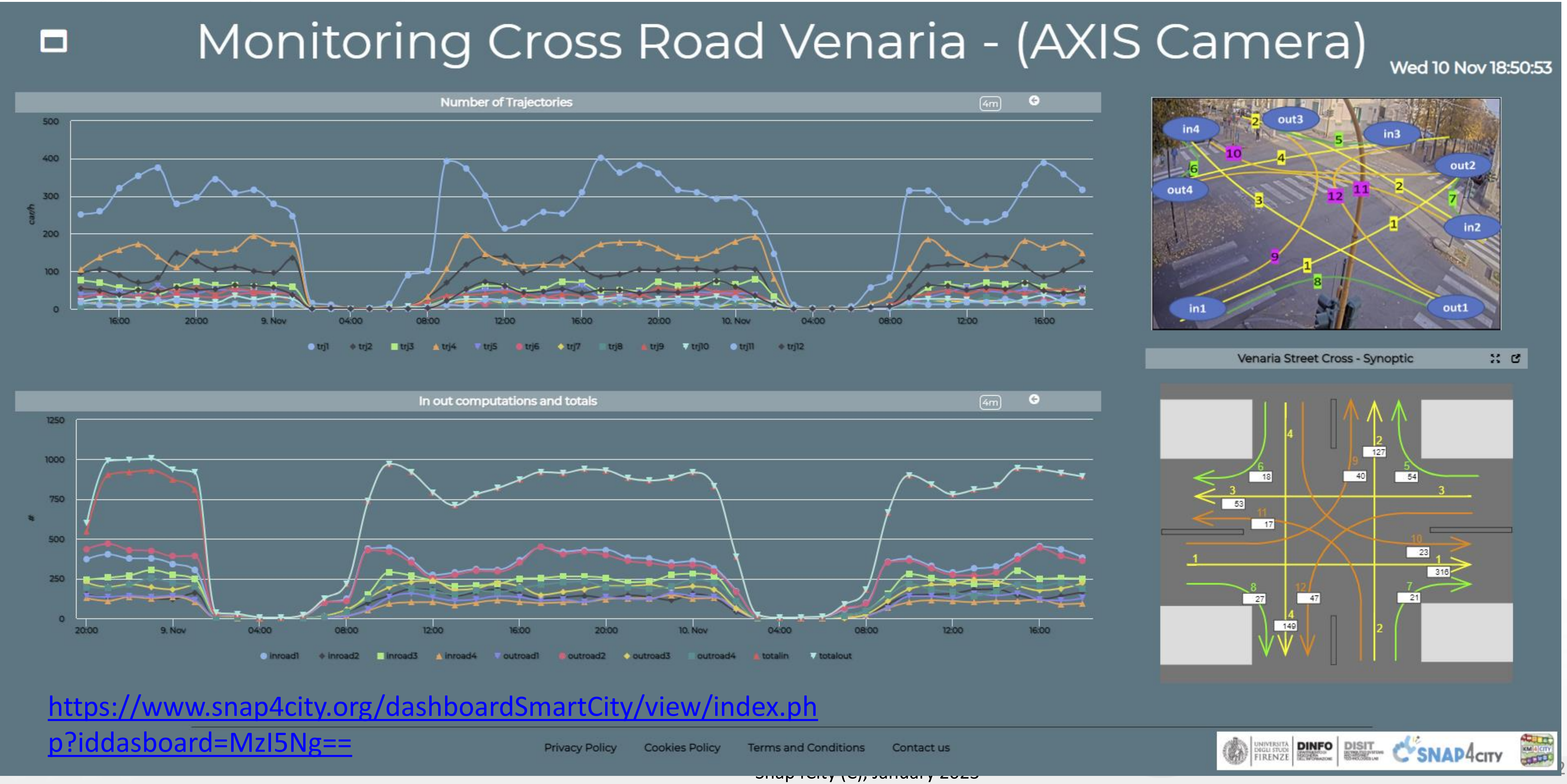
QUERY:
SELECT ?subject ?property ?object

Type of relations

Select all | Deselect all | Invert | □ Hide all inverse

☑ exposedBy	□ foaf:depiction
☑ hasAttribute	☑ hasDashboard
☑ hasSystemCapability	☑ hasWidget
☑ implements	☑ observes
□ owl:sameAs	☑ rdf:type
☑ rdfs:seeAlso	☑ useData
☑ usedByWidget	

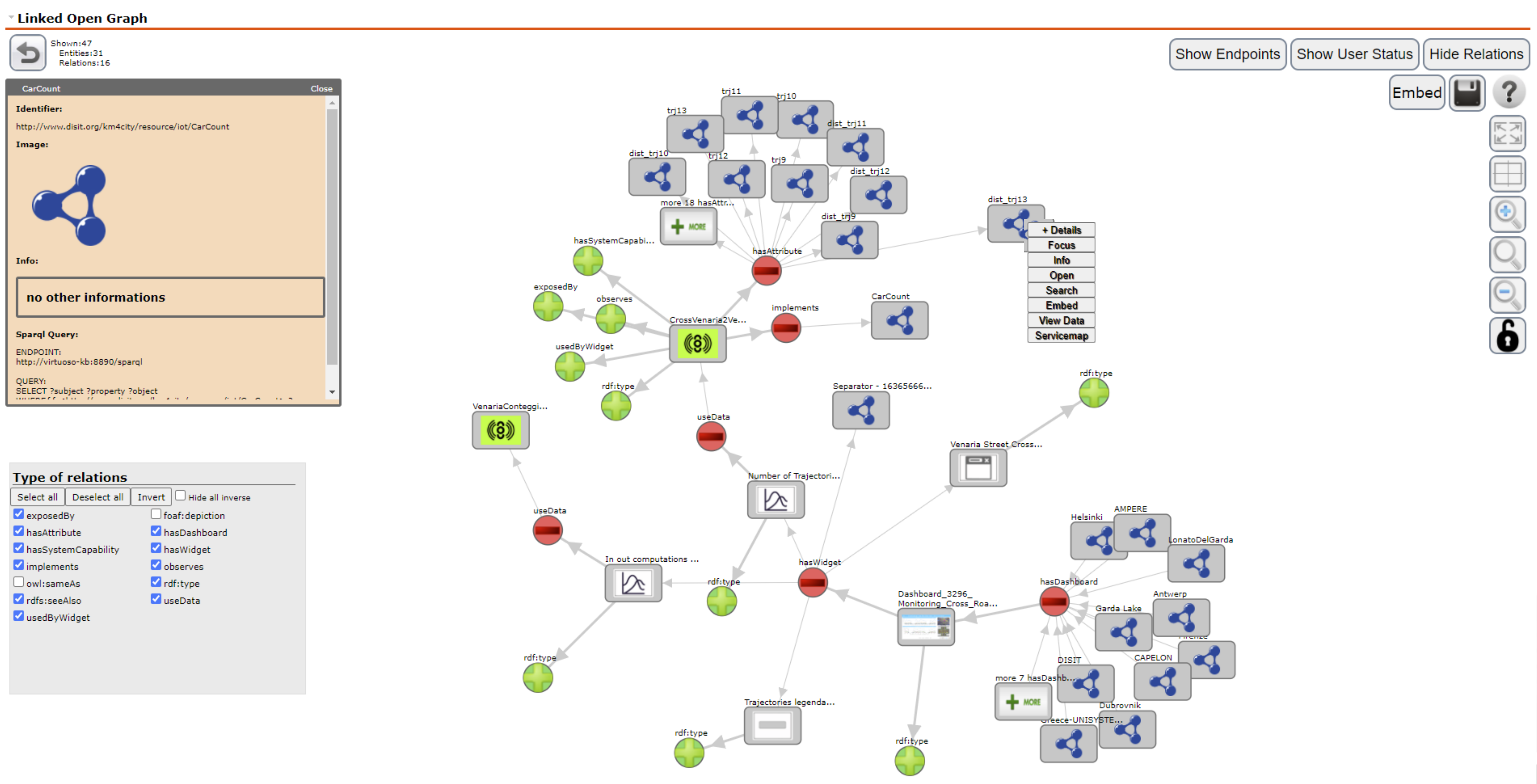
Venaria Reale



Monitoring Cross Road Venaria - (AXIS Camera)

Wed 10 Nov 18:50:53

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



Linked Open Graph

Shown:47
Entities:31
Relations:16

CarCount Close

Identifier:

http://www.disit.org/km4city/resource/iot/CarCount

Image:

Info:

no other informations

Sparql Query:

ENDPOINT:
http://virtuoso-kb:8890/sparql

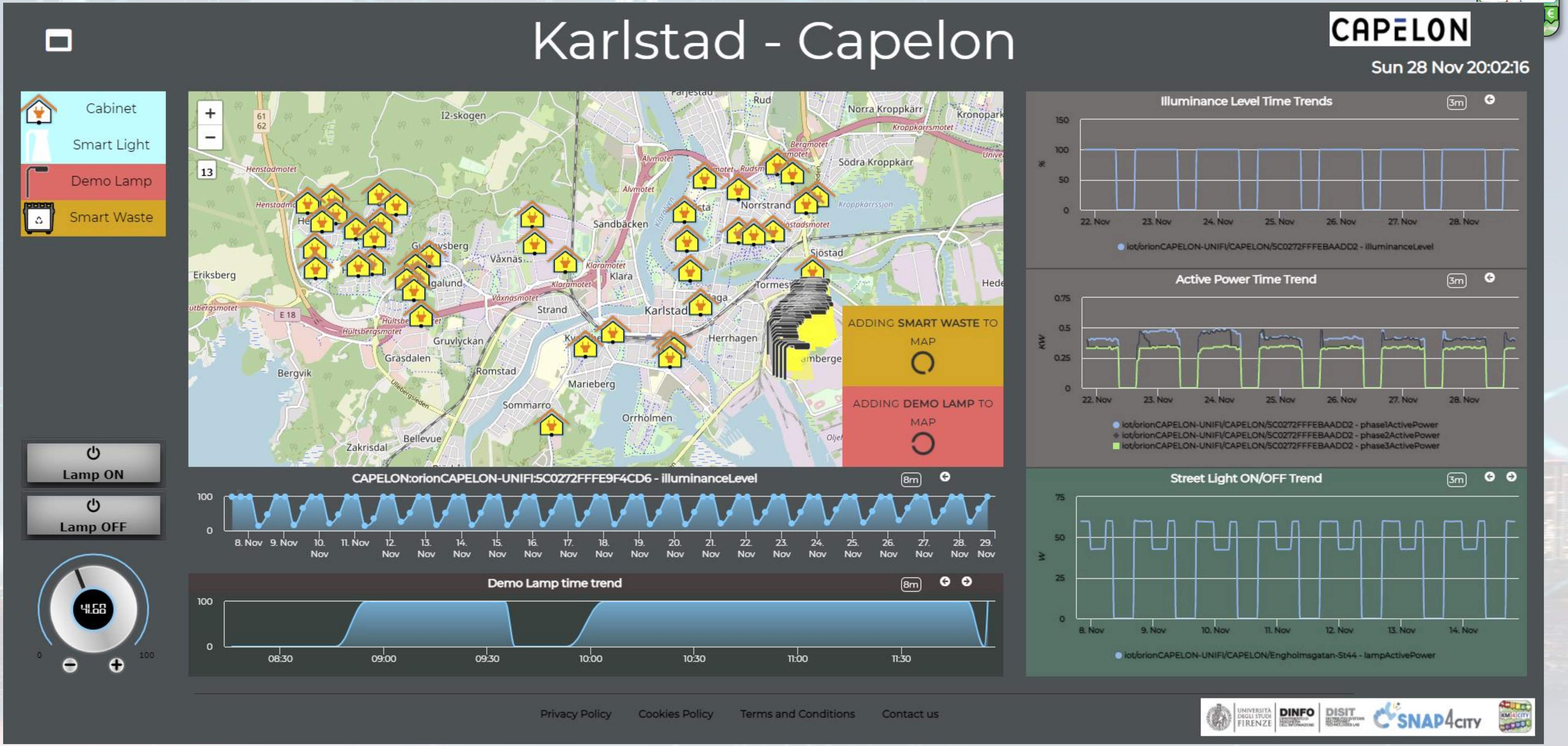
QUERY:
SELECT ?subject ?property ?object

Type of relations

Select all | Deselect all | Invert | □ Hide all inverse

☑ exposedBy
☑ hasAttribute
☑ hasSystemCapability
☑ implements
□ owl:sameAs
☑ rdfs:seeAlso
☑ usedByWidget
□ foaf:depiction
☑ hasDashboard
☑ hasWidget
☑ observes
☑ rdf:type
☑ useData

Karlstad Street Lights



Karlstad - Capelon

Sun 28 Nov 20:02:16

- Cabinet
- Smart Light
- Demo Lamp
- Smart Waste

ADDING SMART WASTE TO MAP

ADDING DEMO LAMP TO MAP

Lamp ON

Lamp OFF

Illuminance Level Time Trends

● iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFEBAADD2 - illuminanceLevel

Active Power Time Trend

● iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFEBAADD2 - phase1ActivePower
◆ iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFEBAADD2 - phase2ActivePower
■ iot/orionCAPELON-UNIFI/CAPELON/5C0272FFFEBAADD2 - phase3ActivePower

CAPELON:orionCAPELON-UNIFI:5C0272FFFE9F4CD6 - illuminanceLevel

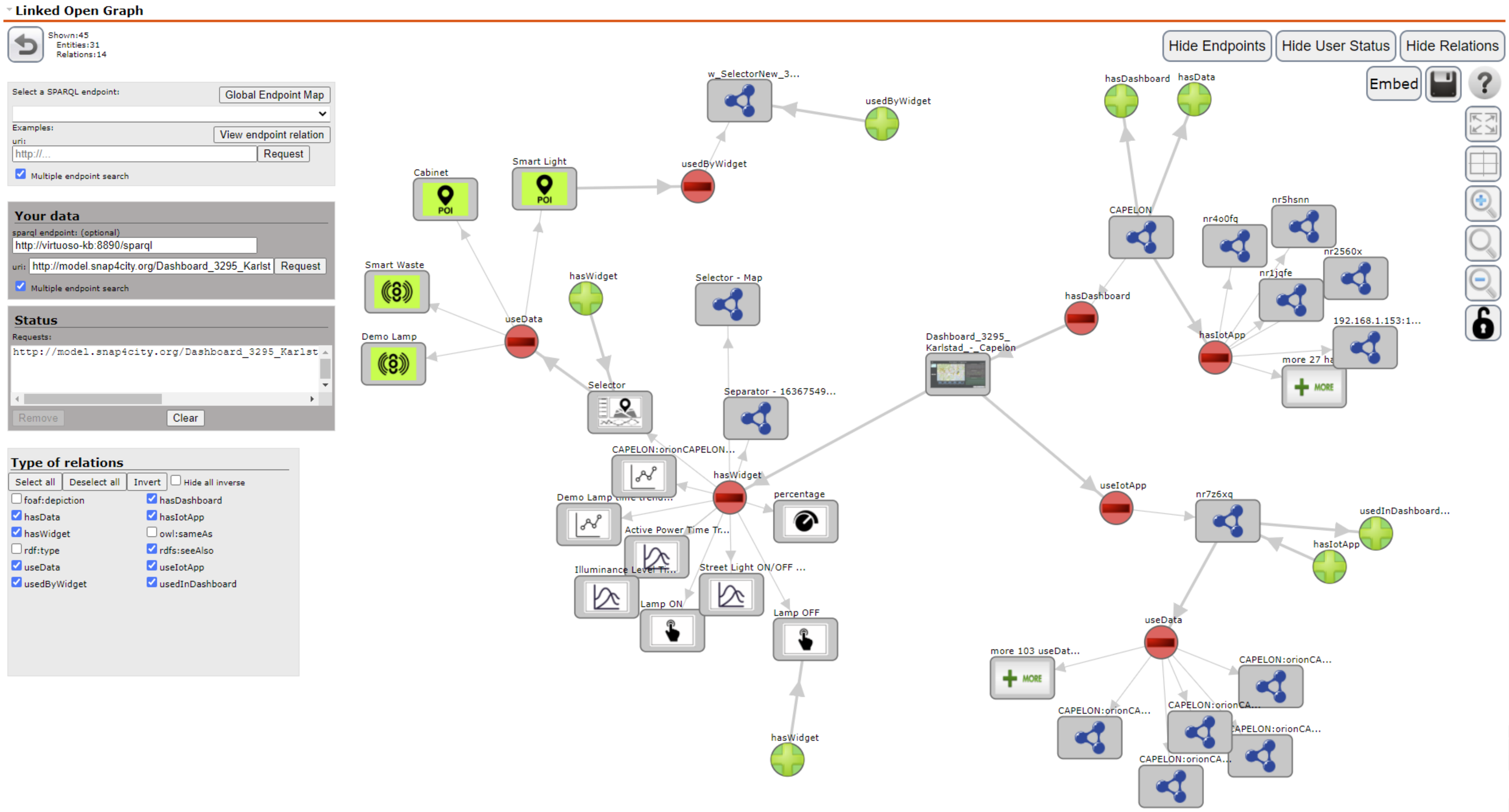
Street Light ON/OFF Trend

● iot/orionCAPELON-UNIFI/CAPELON/Engholmsgatan-St44 - lampActivePower

Demo Lamp time trend

Privacy Policy | Cookies Policy | Terms and Conditions | Contact us

https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MzI5NQ==



Linked Open Graph

Shown:45
Entities:31
Relations:14

Select a SPARQL endpoint: Global Endpoint Map

Examples: View endpoint relation

uri: http://... Request

☑ Multiple endpoint search

Your data

sparql endpoint: (optional)

http://virtuoso-kb:8890/sparql

uri: http://model.snap4city.org/Dashboard_3295_Karlst Request

☑ Multiple endpoint search

Status

Requests:

```
http://model.snap4city.org/Dashboard_3295_Karlst
```

Remove Clear

Type of relations

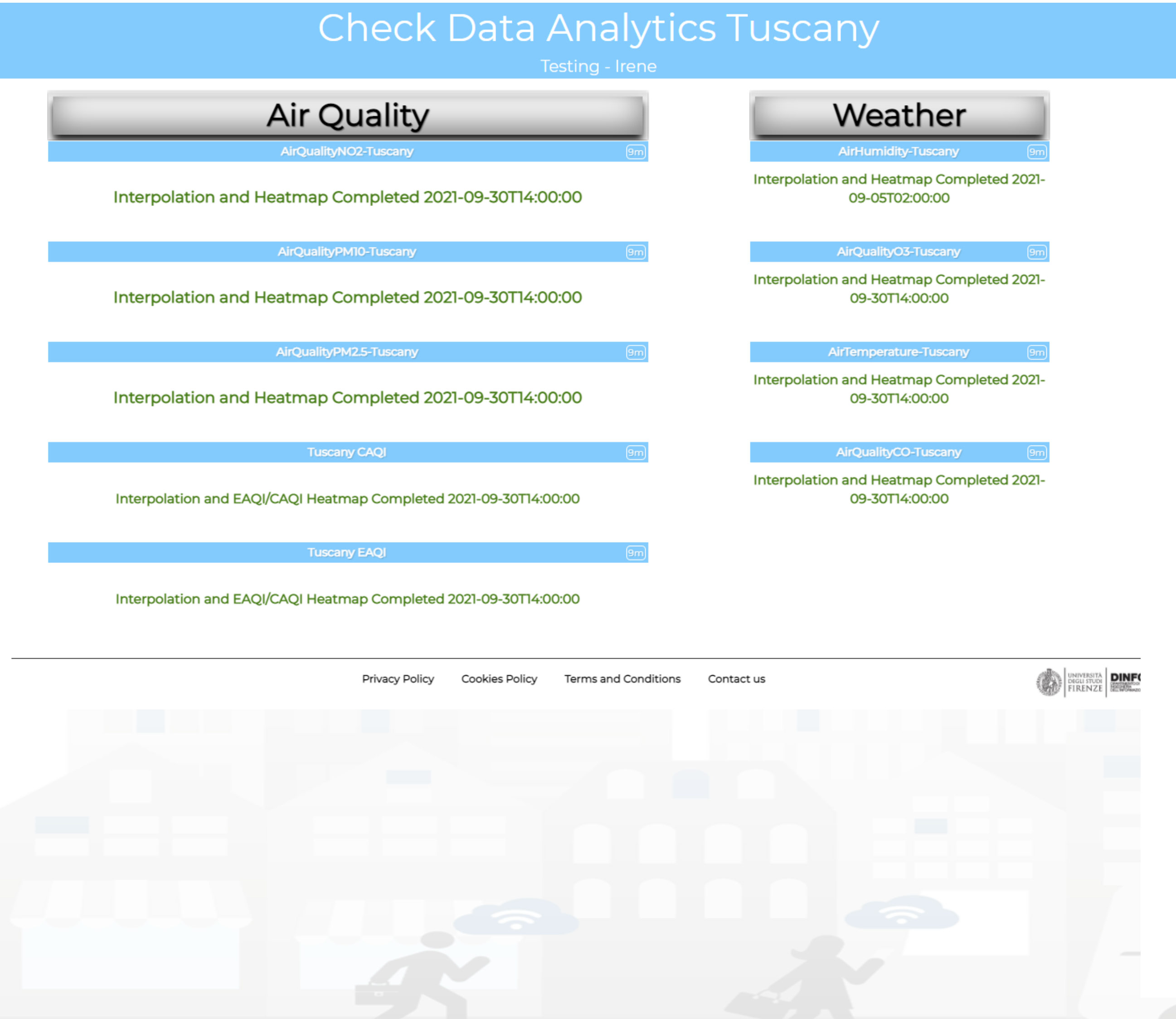
Select all | Deselect all | Invert | □ Hide all inverse

□ foaf:depiction
☑ hasData
☑ hasWidget
□ rdf:type
☑ useData
☑ usedByWidget
☑ hasDashboard
☑ hasIotApp
□ owl:sameAs
☑ rdfs:seeAlso
☑ useIotApp
☑ usedInDashboard

Hide Endpoints | Hide User Status | Hide Relations

Embed

https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MTk2Mw==



Check Data Analytics Tuscany

Testing - Irene

Sun 28 Nov 22:43:30

Air Quality

AirQualityNO2-Tuscany 9m

Interpolation and Heatmap Completed 2021-09-30T14:00:00

AirQualityPM10-Tuscany 9m

Interpolation and Heatmap Completed 2021-09-30T14:00:00

AirQualityPM2.5-Tuscany 9m

Interpolation and Heatmap Completed 2021-09-30T14:00:00

Tuscany CAQI 9m

Interpolation and EAQI/CAQI Heatmap Completed 2021-09-30T14:00:00

Tuscany EAQI 9m

Interpolation and EAQI/CAQI Heatmap Completed 2021-09-30T14:00:00

Weather

AirHumidity-Tuscany 9m

Interpolation and Heatmap Completed 2021-09-05T02:00:00

AirQualityO3-Tuscany 9m

Interpolation and Heatmap Completed 2021-09-30T14:00:00

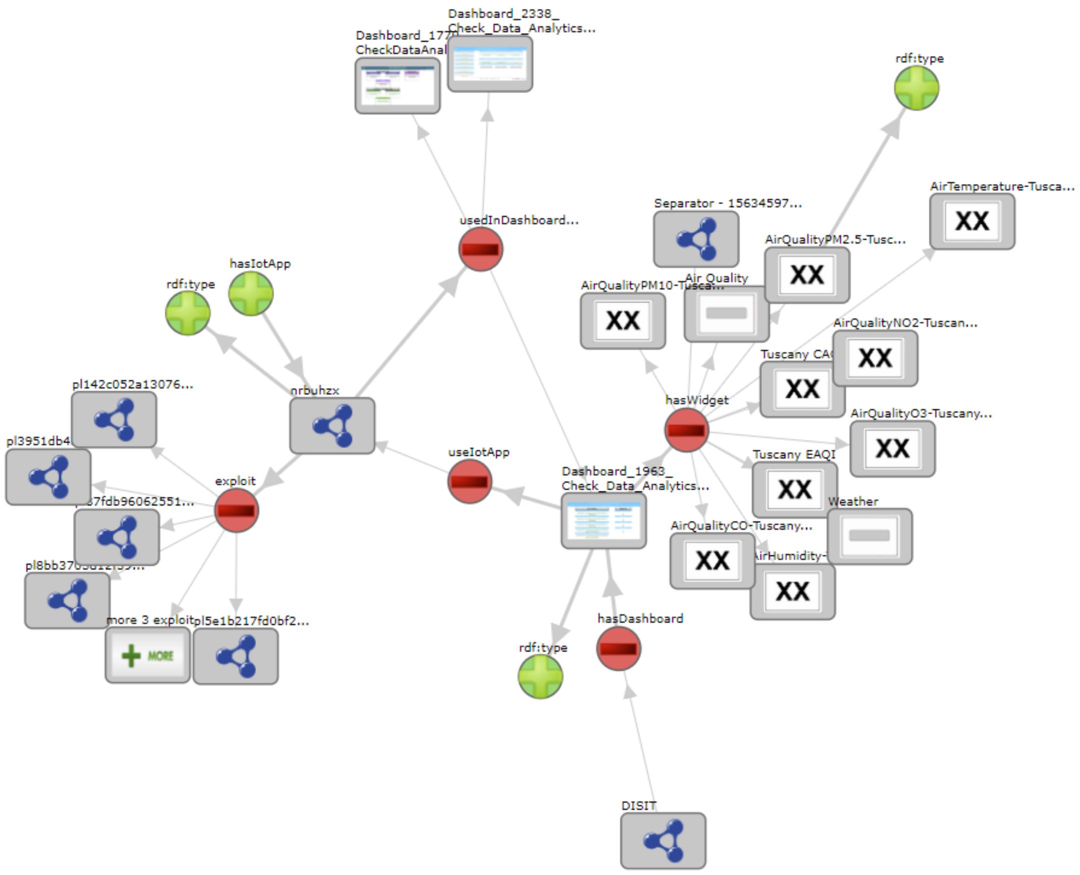
AirTemperature-Tuscany 9m

Interpolation and Heatmap Completed 2021-09-30T14:00:00

AirQualityCO-Tuscany 9m

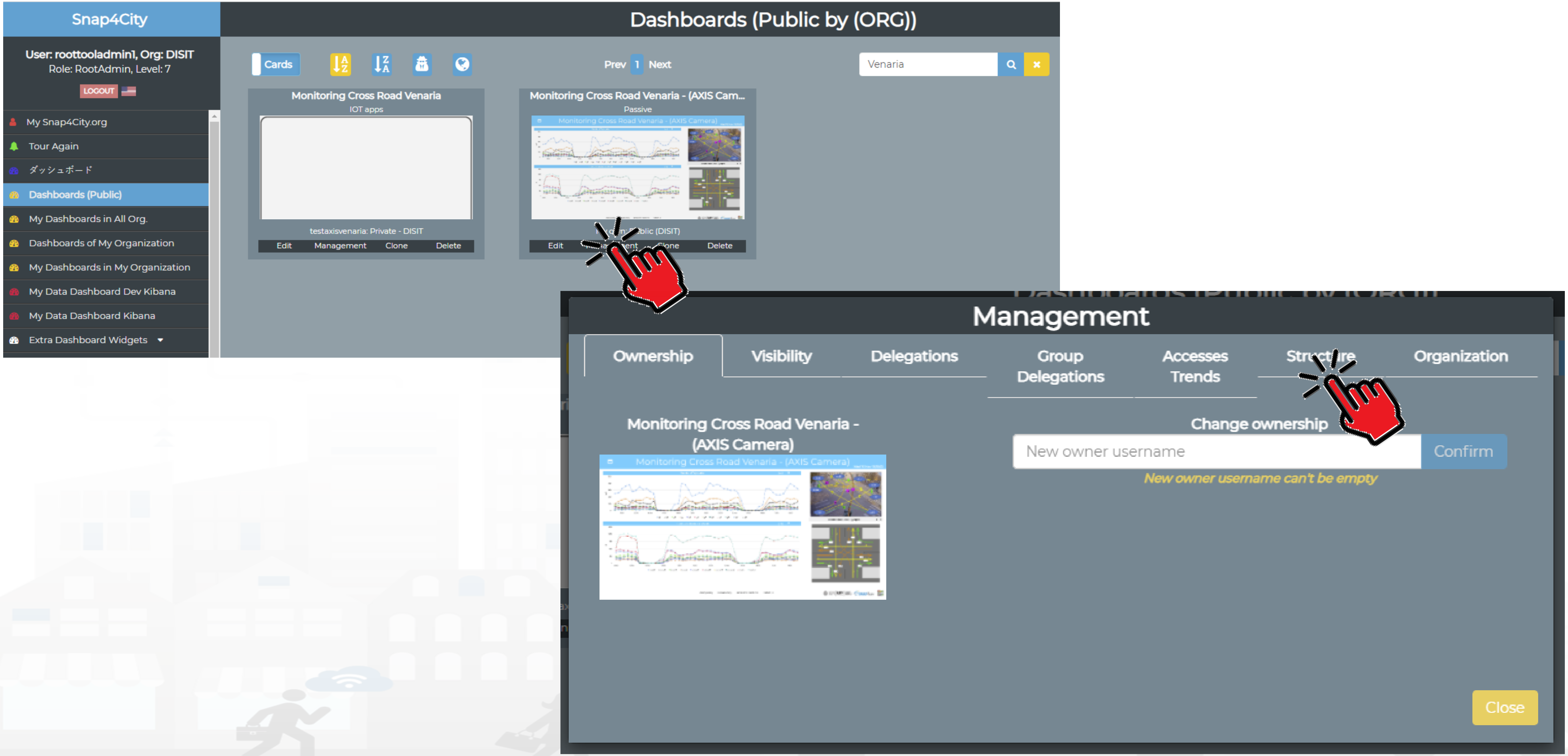
Interpolation and Heatmap Completed 2021-09-30T14:00:00

Privacy Policy | Cookies Policy | Terms and Conditions | Contact us



TOP

Dashboard Structure for all users



Snap4City

Dashboards (Public by (ORG))

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

LOGOUT

- My Snap4City.org
- Tour Again
- 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

Cards | Prev 1 Next | Venaria

Monitoring Cross Road Venaria
IOT apps
testaxisvenaria: Private - DISIT
Edit | Management | Clone | Delete

Monitoring Cross Road Venaria - (AXIS Cam...
Passive
...blic (DISIT)
Edit | Management | Clone | Delete

Management

Ownership | Visibility | Delegations | Group Delegations | Accesses Trends | Structure | Organization

Monitoring Cross Road Venaria - (AXIS Camera)

Change ownership

New owner username | Confirm

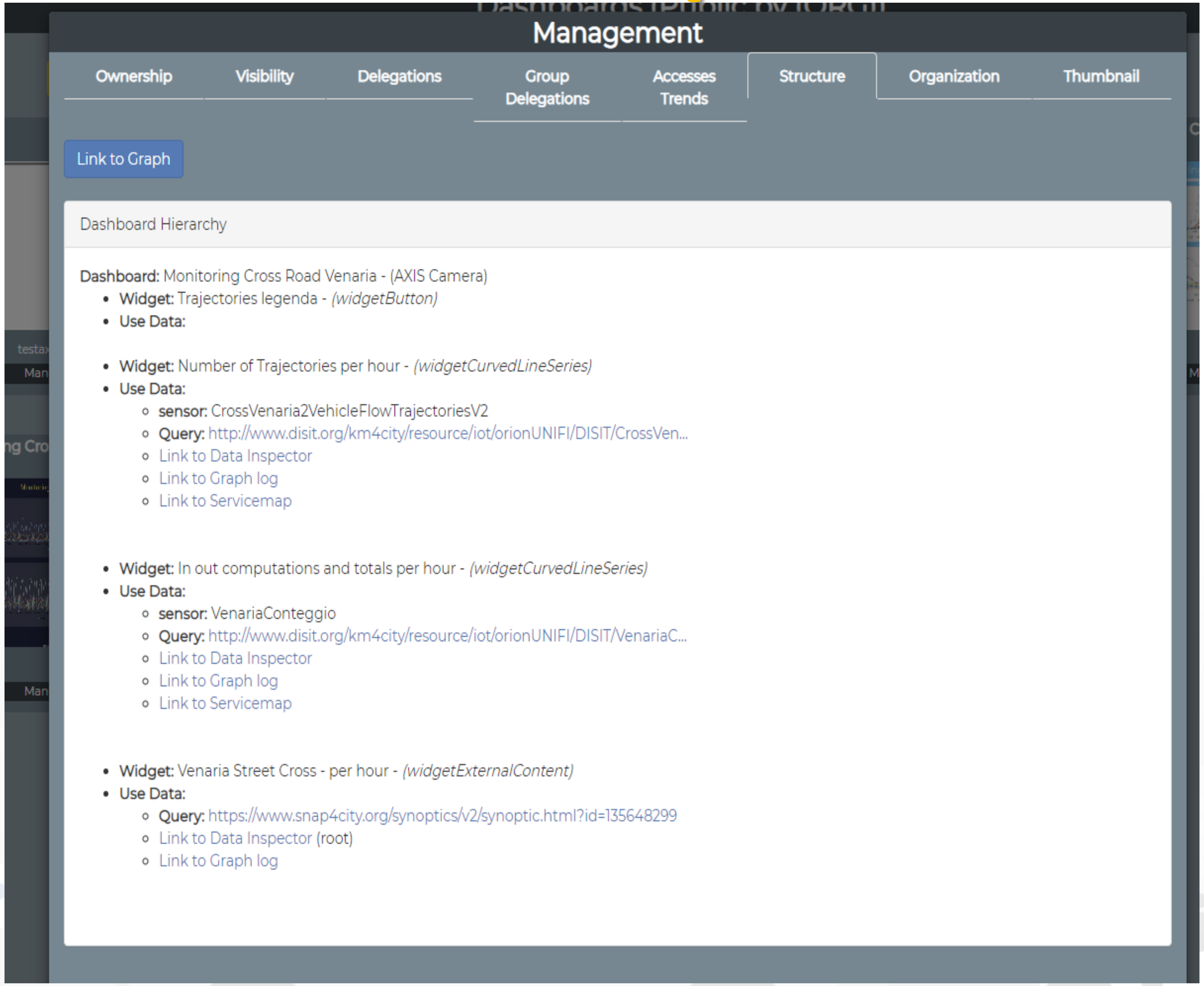
New owner username can't be empty

Close

Dashboard Structure and Components

4 Widgets

- Button
  - It is the image
- Curved LineSeries
  - …. Set of data….
- Curved LineSeries
  - ….set of data…
- External Content
  - With synoptic



Dashboard Hierarchy

Dashboard: Monitoring Cross Road Venaria - (AXIS Camera)

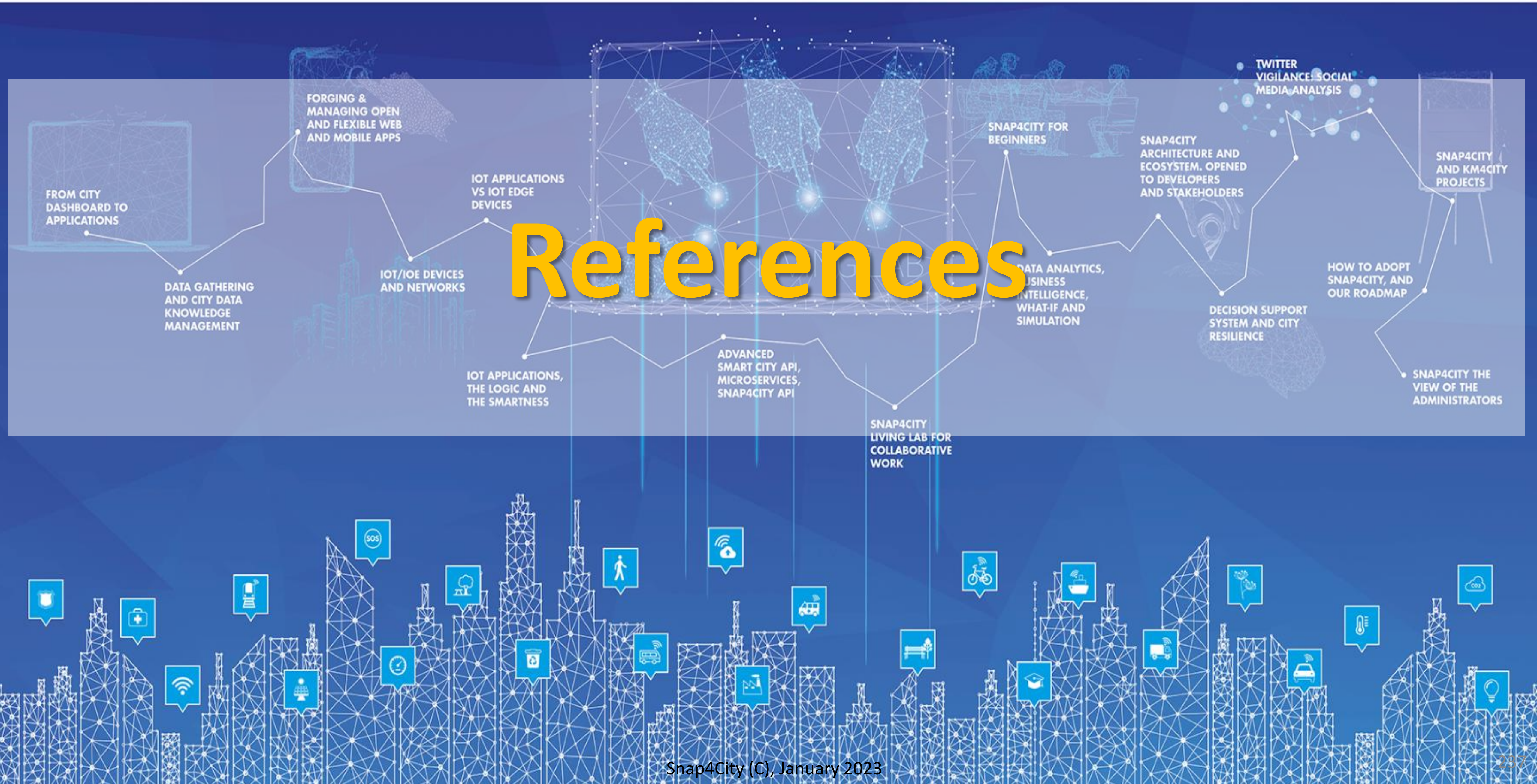
- **Widget:** Trajectories legenda - *(widgetButton)*
- **Use Data:**

- **Widget:** Number of Trajectories per hour - *(widgetCurvedLineSeries)*
- **Use Data:**
  - **sensor:** CrossVenaria2VehicleFlowTrajectoriesV2
  - **Query:** http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/CrossVen...
  - Link to Data Inspector
  - Link to Graph log
  - Link to Servicemap

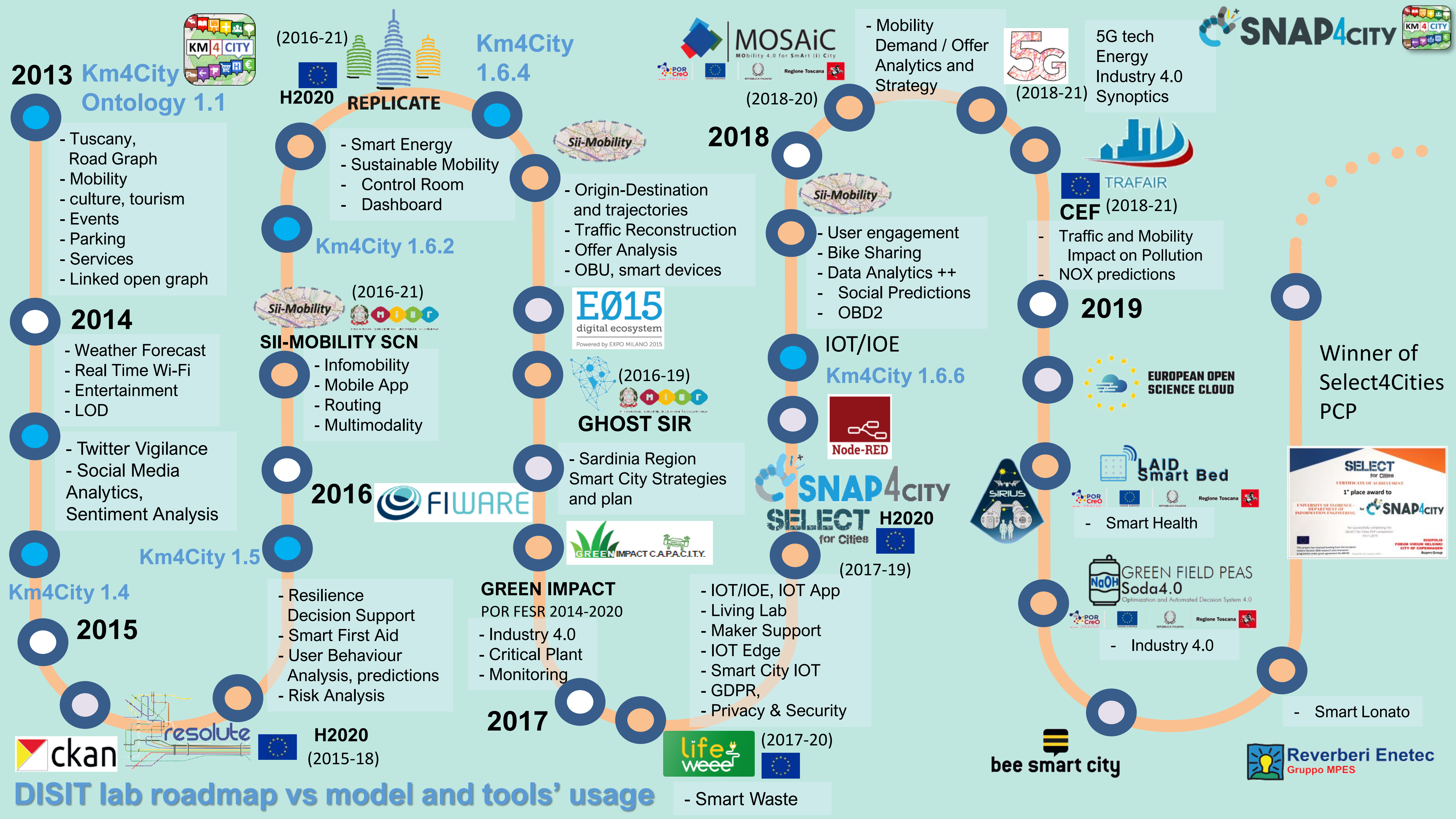
- **Widget:** In out computations and totals per hour - *(widgetCurvedLineSeries)*
- **Use Data:**
  - **sensor:** VenariaConteggio
  - **Query:** http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/VenariaC...
  - Link to Data Inspector
  - Link to Graph log
  - Link to Servicemap

- **Widget:** Venaria Street Cross - per hour - *(widgetExternalContent)*
- **Use Data:**
  - **Query:** https://www.snap4city.org/synoptics/v2/synoptic.html?id=135648299
  - Link to Data Inspector (root)
  - Link to Graph log

References


FROM CITY DASHBOARD TO APPLICATIONS
DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT
FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS
IOT/IOE DEVICES AND NETWORKS
IOT APPLICATIONS VS IOT EDGE DEVICES
IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS
ADVANCED SMART CITY API, MICROSERVICES, SNAP4CITY API
SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK
SNAP4CITY FOR BEGINNERS
DATA ANALYTICS, BUSINESS INTELLIGENCE, WHAT-IF AND SIMULATION
SNAP4CITY ARCHITECTURE AND ECOSYSTEM. OPENED TO DEVELOPERS AND STAKEHOLDERS
DECISION SUPPORT SYSTEM AND CITY RESILIENCE
TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS
HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP
SNAP4CITY AND KM4CITY PROJECTS
SNAP4CITY THE VIEW OF THE ADMINISTRATORS
SOS
CO2

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

ckan

resolute H2020 (2015-18)

Km4City 1.5

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

2016 FIWARE

SII-MOBILITY SCN (2016-21)

- Infomobility
- Mobile App
- Routing
- Multimodality

Km4City 1.6.2

- Smart Energy
- Sustainable Mobility
- Control Room
- Dashboard

(2016-21) H2020 REPLICATE

Km4City 1.6.4

Sii-Mobility

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

EØ15 digital ecosystem

Powered by EXPO MILANO 2015

(2016-19) 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

2017

life weee (2017-20)

- Smart Waste

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

SNAP4CITY SELECT for Cities H2020 (2017-19)

Node-RED

Km4City 1.6.6

IOT/IOE

Sii-Mobility

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

2018

MOSAiC MObility 4.0 for SmArt (i) City

POR CreO · Regione Toscana

(2018-20)

- Mobility Demand / Offer Analytics and Strategy

5G (2018-21)

5G tech
Energy
Industry 4.0
Synoptics

SNAP4CITY

TRAFAIR

CEF (2018-21)

- Traffic and Mobility Impact on Pollution
- NOX predictions

2019

EUROPEAN OPEN SCIENCE CLOUD

SIRIUS

LAID Smart Bed

POR CreO · Regione Toscana

- Smart Health

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

POR CreO · Regione Toscana

- Industry 4.0

bee smart city

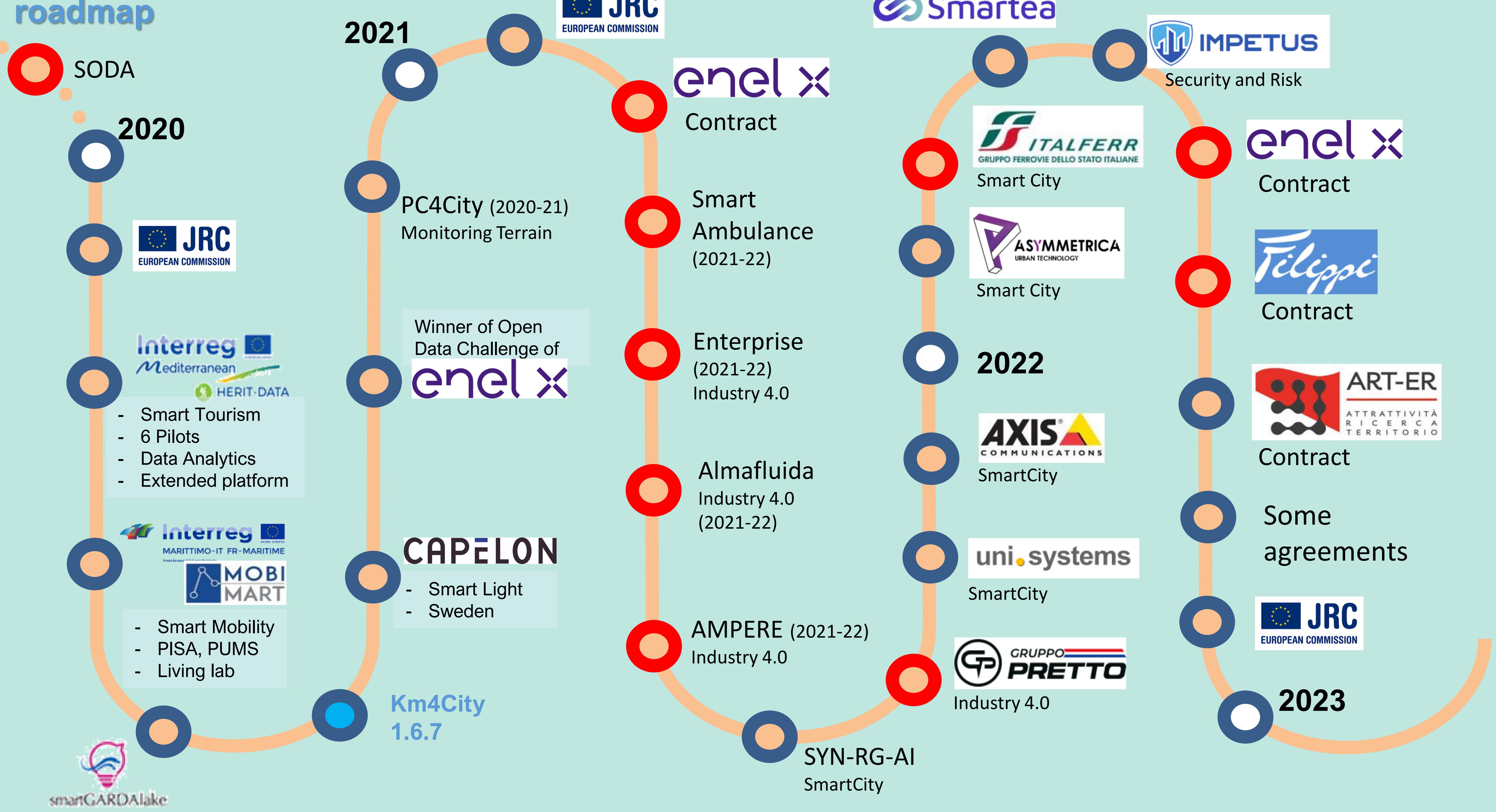
- Smart Lonato

Reverberi Enetec Gruppo MPES

Winner of Select4Cities PCP

SELECT for Cities
CERTIFICATE OF ACHIEVEMENT
1° place award to
UNIVERSITY OF FLORENCE - DEPARTMENT OF INFORMATION ENGINEERING
for SNAP4CITY

SNAP4CITY roadmap



SODA

2020

JRC EUROPEAN COMMISSION

Interreg Mediterranean HERIT-DATA
- Smart Tourism
- 6 Pilots
- Data Analytics
- Extended platform

Interreg MARITTIMO-IT FR-MARITIME MOBIMART
- Smart Mobility
- PISA, PUMS
- Living lab

smartGARDAlake

Km4City 1.6.7

CAPELON
- Smart Light
- Sweden

Winner of Open Data Challenge of enel x

PC4City (2020-21)
Monitoring Terrain

2021

JRC EUROPEAN COMMISSION

enel x
Contract

Smart Ambulance (2021-22)

Enterprise (2021-22)
Industry 4.0

Almafluida
Industry 4.0
(2021-22)

AMPERE (2021-22)
Industry 4.0

SYN-RG-AI
SmartCity

GRUPPO PRETTO
Industry 4.0

uni.systems
SmartCity

AXIS COMMUNICATIONS
SmartCity

2022

ASYMMETRICA URBAN TECHNOLOGY
Smart City

ITALFERR GRUPPO FERROVIE DELLO STATO ITALIANE
Smart City

Smartea

IMPETUS
Security and Risk

enel x
Contract

Filippi
Contract

ART-ER ATTRATTIVITÀ RICERCA TERRITORIO
Contract

Some agreements

JRC EUROPEAN COMMISSION

2023

2022 booklets

- Snap4City

SNAP4CITY
LIVING LAB
Flyer
SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES
2022 EDITION

https://www.snap4city.org/download/video/DPL_SNAP4CITY_2022-v02.pdf

- Snap4Industry

SNAP4INDUSTRY
Flyer
SIMPLE APPROACH TO SETUP SMART IoT HUBS, APPLICATIONS AND SERVICES
2022 EDITION

https://www.snap4city.org/download/video/DPL_SNAP4INDUSTRY_2022-v03.pdf

- Solutions
- Data Analytics

SNAP4SOLUTIONS
DATA ANALYTICS
ARTIFICIAL INTELIGENCE
SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES
2022 EDITION

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

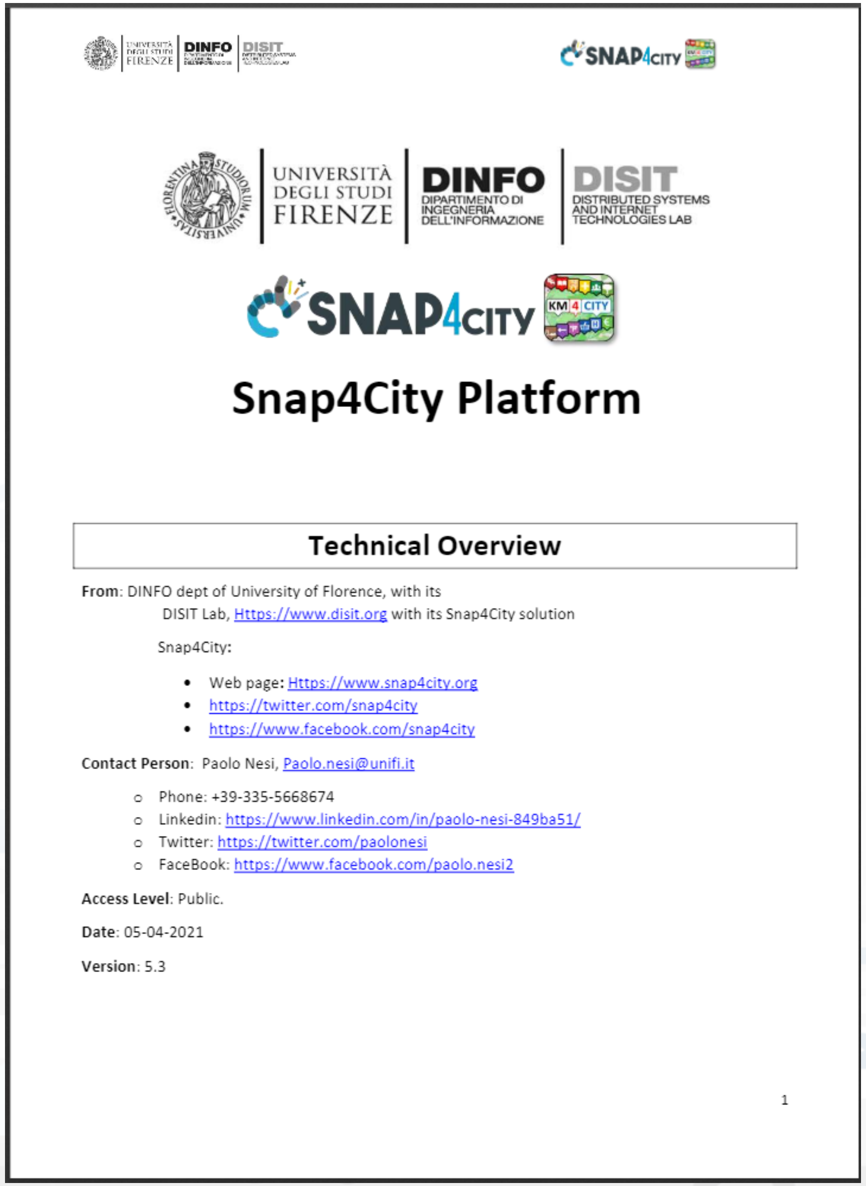
https://www.snap4city.org/577

On Line Training Material (free of charge)

	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								

Video1	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube
Video2	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube
Video3	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube	▶ YouTube
Video4	▶ YouTube	▶ YouTube	▶ YouTube	none	▶ YouTube	none	none

Overview



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

Access Level: Public.

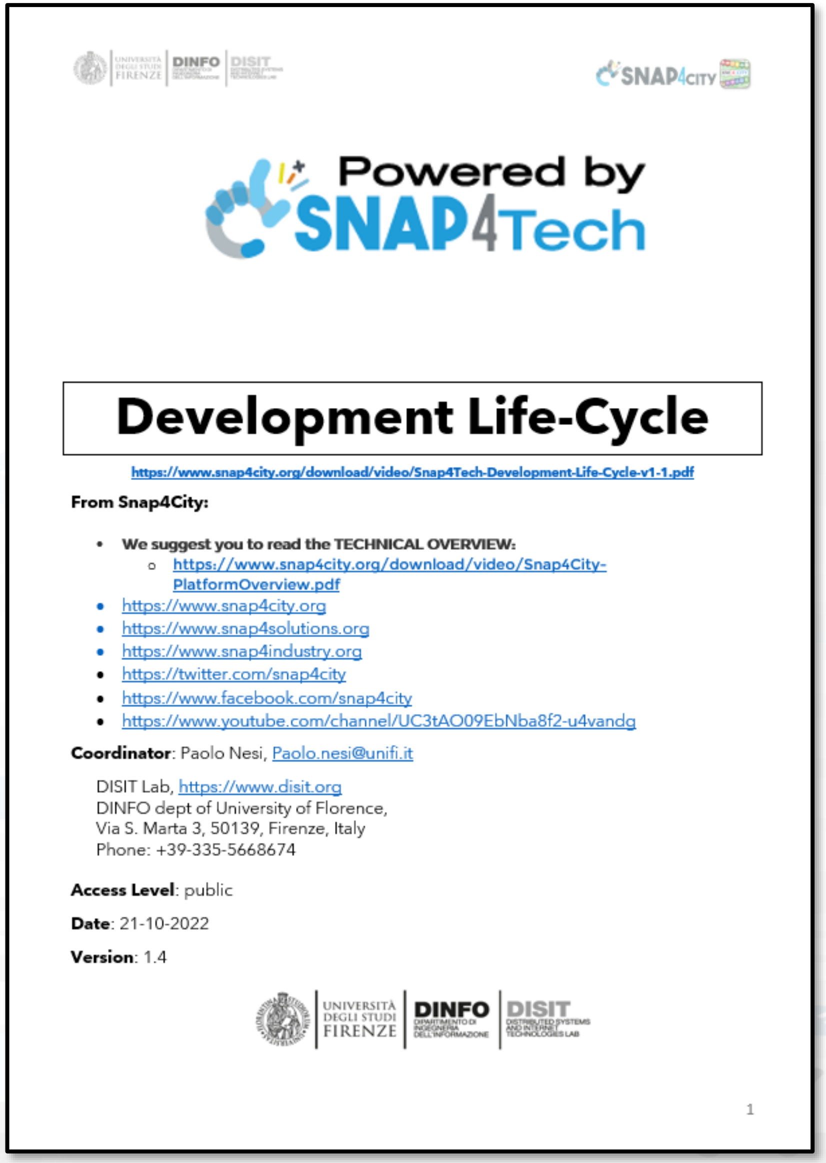
Date: 05-04-2021

Version: 5.3

1

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

Development



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

Access Level: public

Date: 21-10-2022

Version: 1.4

1

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

TOP

SNAP4CITY

KM4CITY

Be smart in a SNAP!

100% OPEN SOURCE

SNAP4
Appliances and Dockers
Installations

CONTACT

www.snap4city.org

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

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