SNAP4CITY

www.snap4city.org
www.snap4solutions.org

KM4CITY

www.km4city.org

Developing Smart Applications & Business Intelligence Solutions

Nov. 2023, Course, Part 8

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

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

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

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

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

SNAP4CITY

KM4CITY

SNAP4 Appliances and Dockers Installations

Be smart in a SNAP!

SMARTCITY EXPO WORLD CONGRESS
7-9 November 2023, Barcelona, Spain
Visit Snap4City in Hall 1

100% OPEN SOURCE

EU GDPR COMPLIANT

PEN Test Passed

Powered by FIWARE

SNAP4CITY on EUROPEAN OPEN SCIENCE CLOUD MARKETPLACE

Node-RED

Developing Smart Applications & Business Intelligence Solutions

Nov. 2023, Course, Part 8
https://www.snap4city.org/944
https://www.snap4city.org/577

SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

CC BY NC ND
SOME RIGHTS RESERVED

Certifications to the Course

- Certification of Participation will be provided to all the course attendees

- Certification of proficiency will be provided to everybody would like to get one. The Certification will be provided passing a short interview

Smart Solutions and Decision Support Systems

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

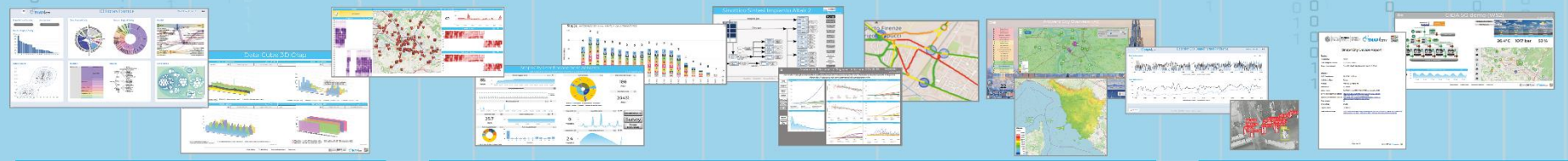
JS Foundation

E015 digital ecosystem

NVIDIA

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

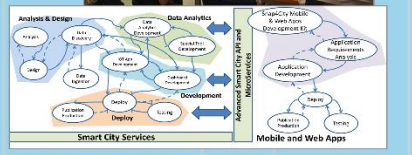
DASHBOARDS - VISUAL ANALYTICS - SYNOPTICS - DIGITAL TWIN - GRAPHICAL WIDGETS - ANALYTICS - GUI CUSTOM STYLES - VISUAL PROGRAMMING



DASHBOARDS, WIDGETS
TEMPLATES

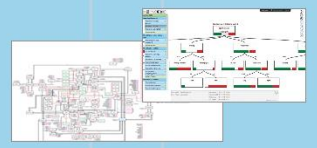
PREDICTION - ANOMALY DETECTION - CLUSTERING - ROUTING - SENTIMENT NLP - TRAFFIC FLOW
PEOPLE FLOWS - SDG - 15 MIN CITY INDEX - KPI - HEATMAPS - ORIGIN DESTINATION - ETC...

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

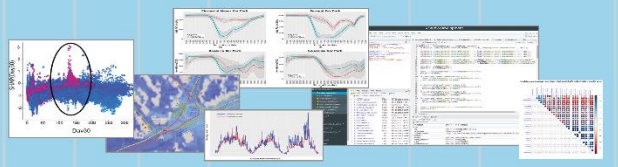


METHODOLOGIES
LIVING LABS
COURSES AND COMMUNITY
DEVELOPMENT TOOLS

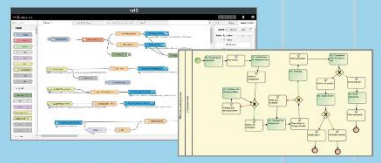
ANY: DATA, BROKER, NETWORK AND VERTICAL



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

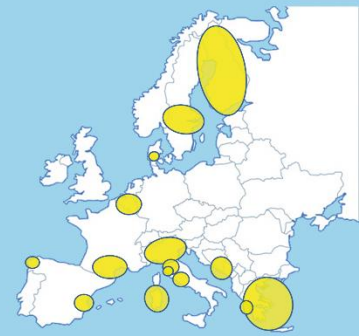


VISUAL PROGRAMMING, ADAPTERS
DATA FLOWS, WORKFLOWS
PARALLEL DISTRIBUTED PROCESSING
DATA DRIVEN

Native and External Applications

- Smart Parking
- Smart Light
- Smart Waste
- Smart Energy
- Smart Building
- Smart Tourism
- Social Media Analysis

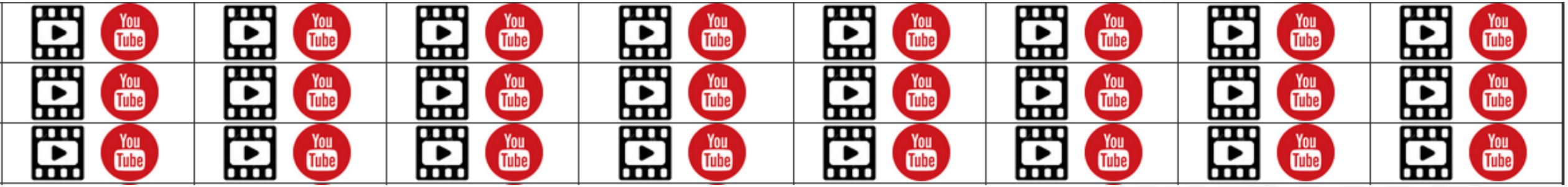
...



https://www.snap4city.org/944

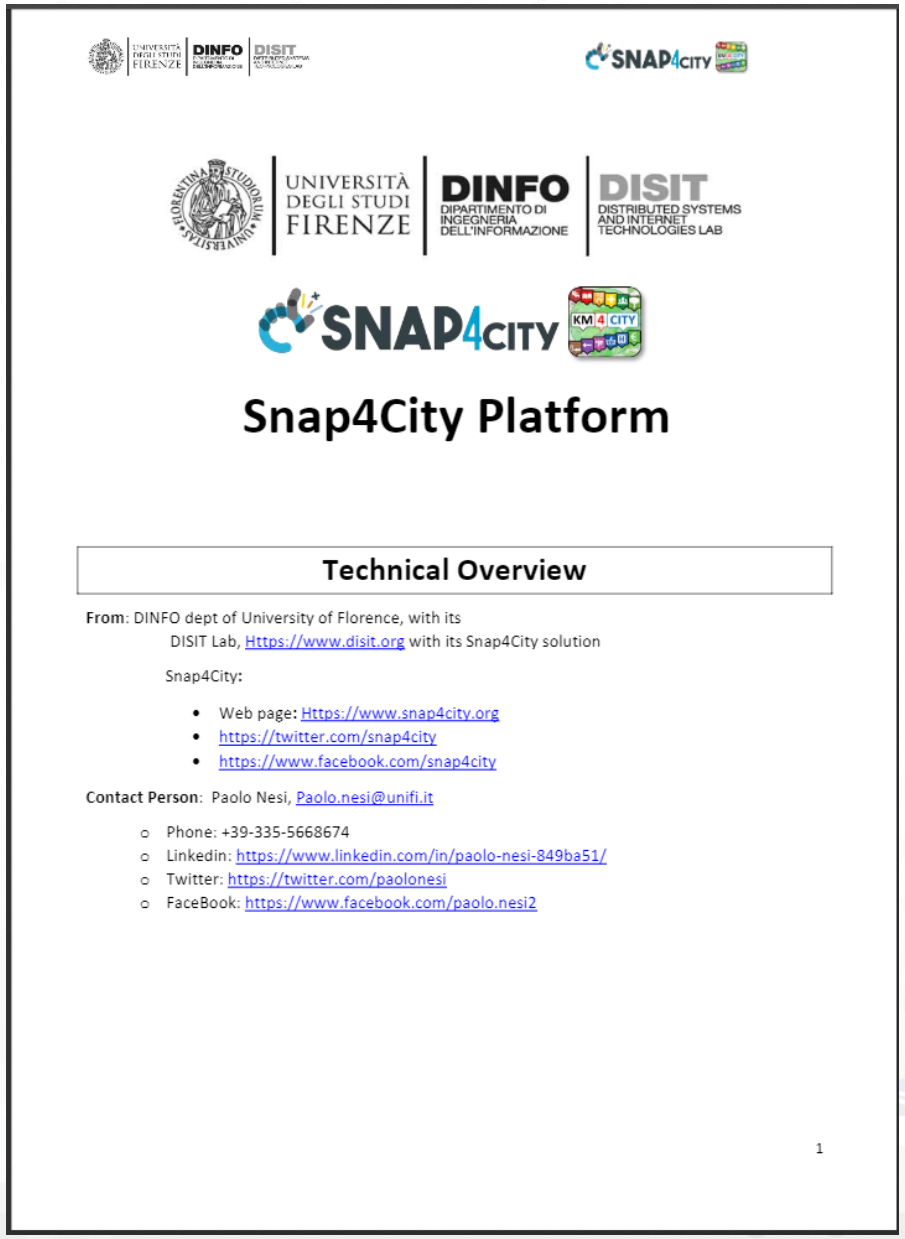
On Line Training Material (free of charge)

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



Note on Training Material

- **Course 2023**: https://www.snap4city.org/944
  - Introductionary course to Snap4City technology
- **Course** https://www.snap4city.org/577
  - Full training course with much more details on mechanisms and a wider set of cases/solutions of the Snap4City Technology
- **Documentation** includes a deeper round of details
  - Snap4City Platform Overview:
    - https://www.snap4city.org/drupal/sites/default/files/files/Snap4City-PlatformOverview.pdf
  - Development Life Cycle:
    - https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf
  - Client Side Business Logic:
    - https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf
- **On line cases and documentation**:
  - https://www.snap4city.org/108
  - https://www.snap4city.org/78
  - https://www.snap4city.org/426



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

SNAP4CITY KM4CITY

Snap4City Platform

Technical Overview

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

Snap4City:

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

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

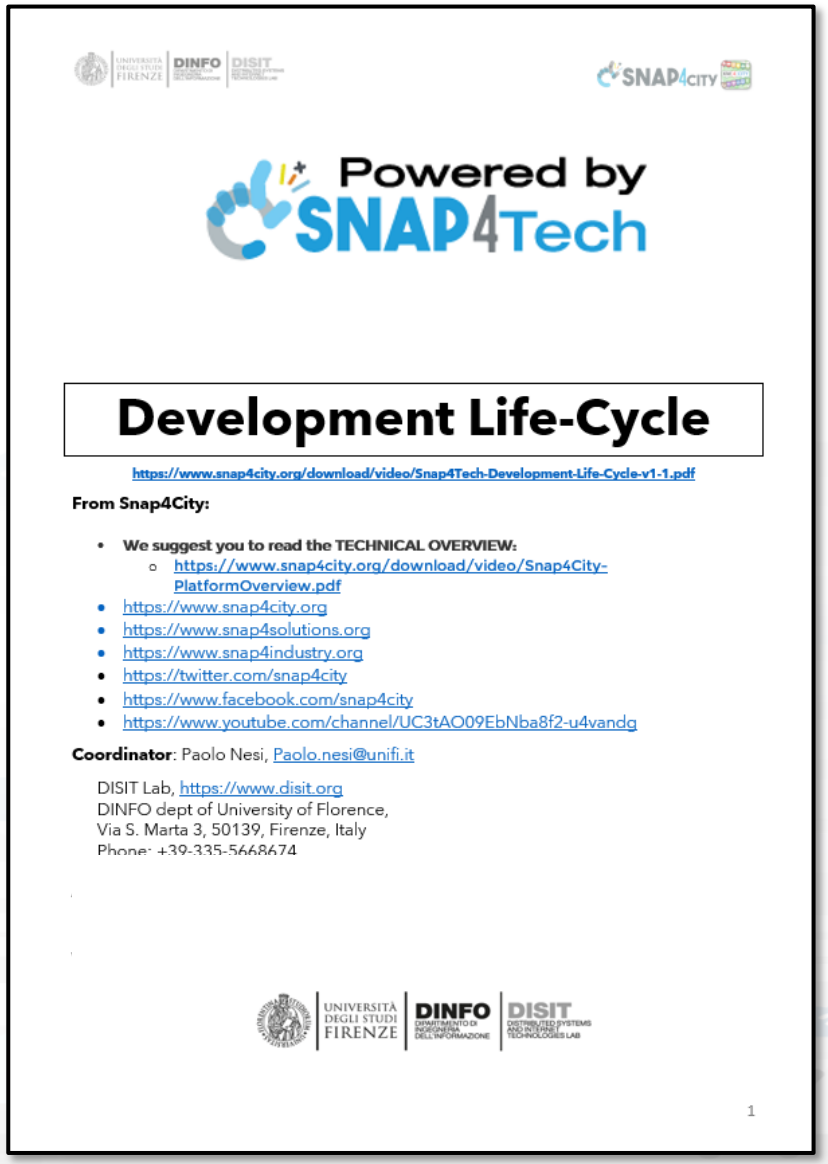
- Phone: +39-335-5668674
- Linkedin: https://www.linkedin.com/in/paolo-nesi-849ba51/
- Twitter: https://twitter.com/paolonesi
- FaceBook: https://www.facebook.com/paolo.nesi2

1

Tech Overview

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

Development



Powered by SNAP4Tech

Development Life-Cycle

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

From Snap4City:

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

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

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

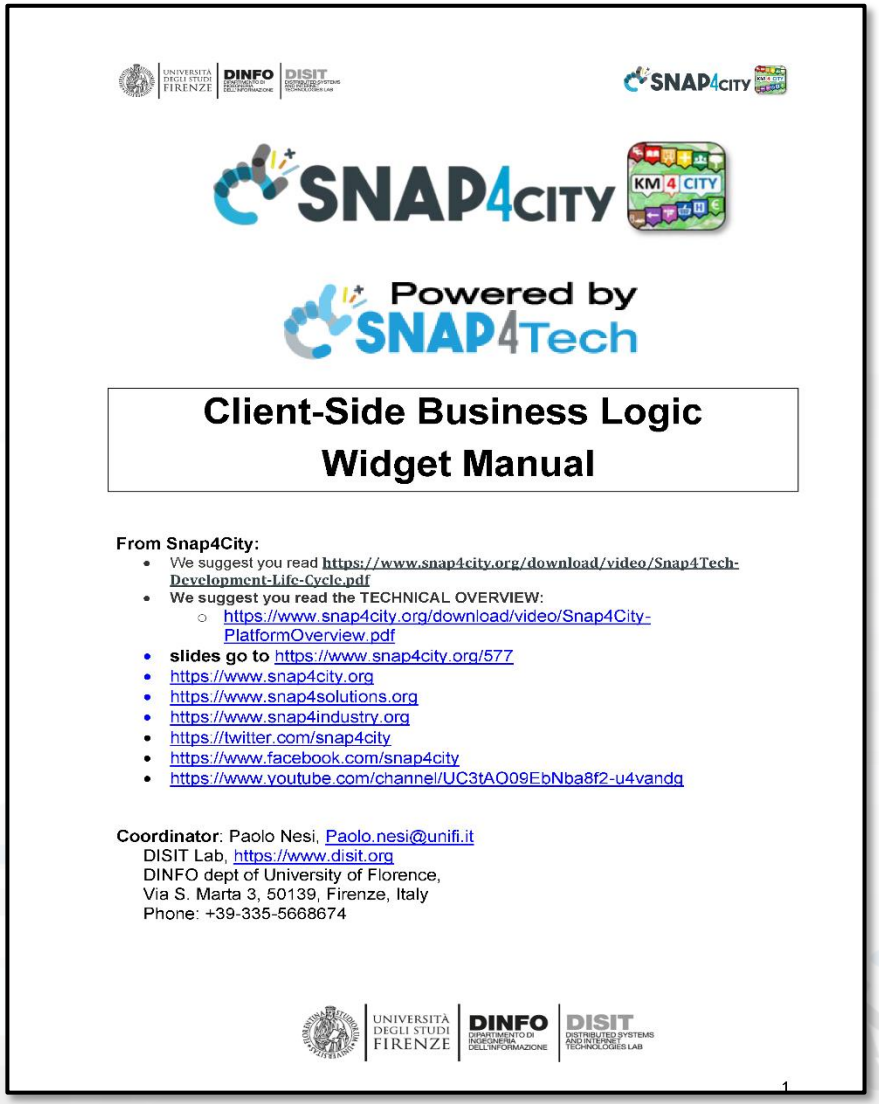
UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

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

CSBL

Client Side Business Logic



Powered by SNAP4Tech

Client-Side Business Logic Widget Manual

From Snap4City:

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

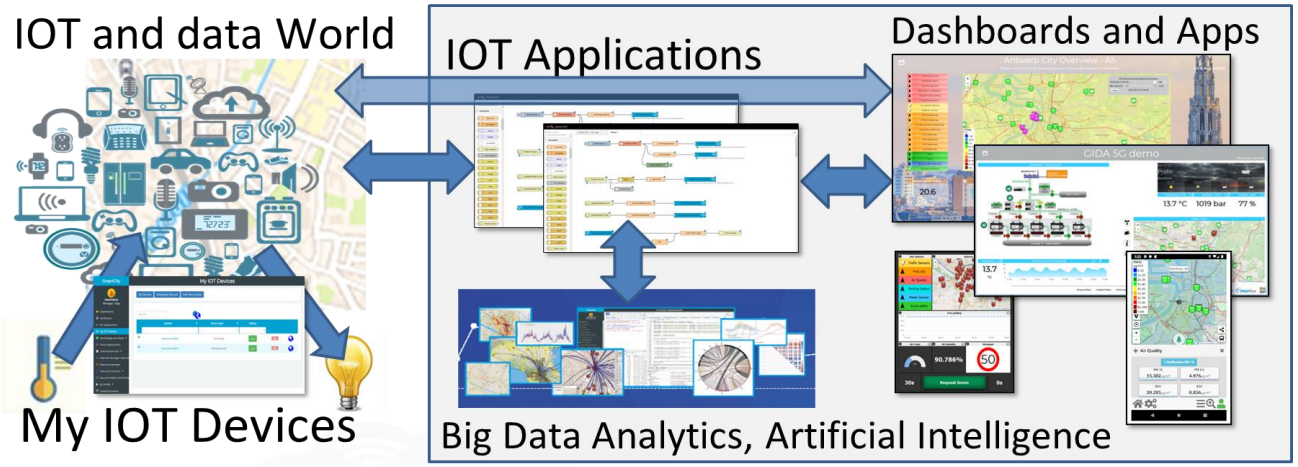
Coordinator: Paolo Nesi, Paolo.nesi@unifi.it
DISIT Lab, https://www.disit.org
DINFO dept of University of Florence,
Via S. Marta 3, 50139, Firenze, Italy
Phone: +39-335-5668674

1

https://www.snap4city.org/download/video/ClientSideBusinessLogic-W[QR]df

Free Trial

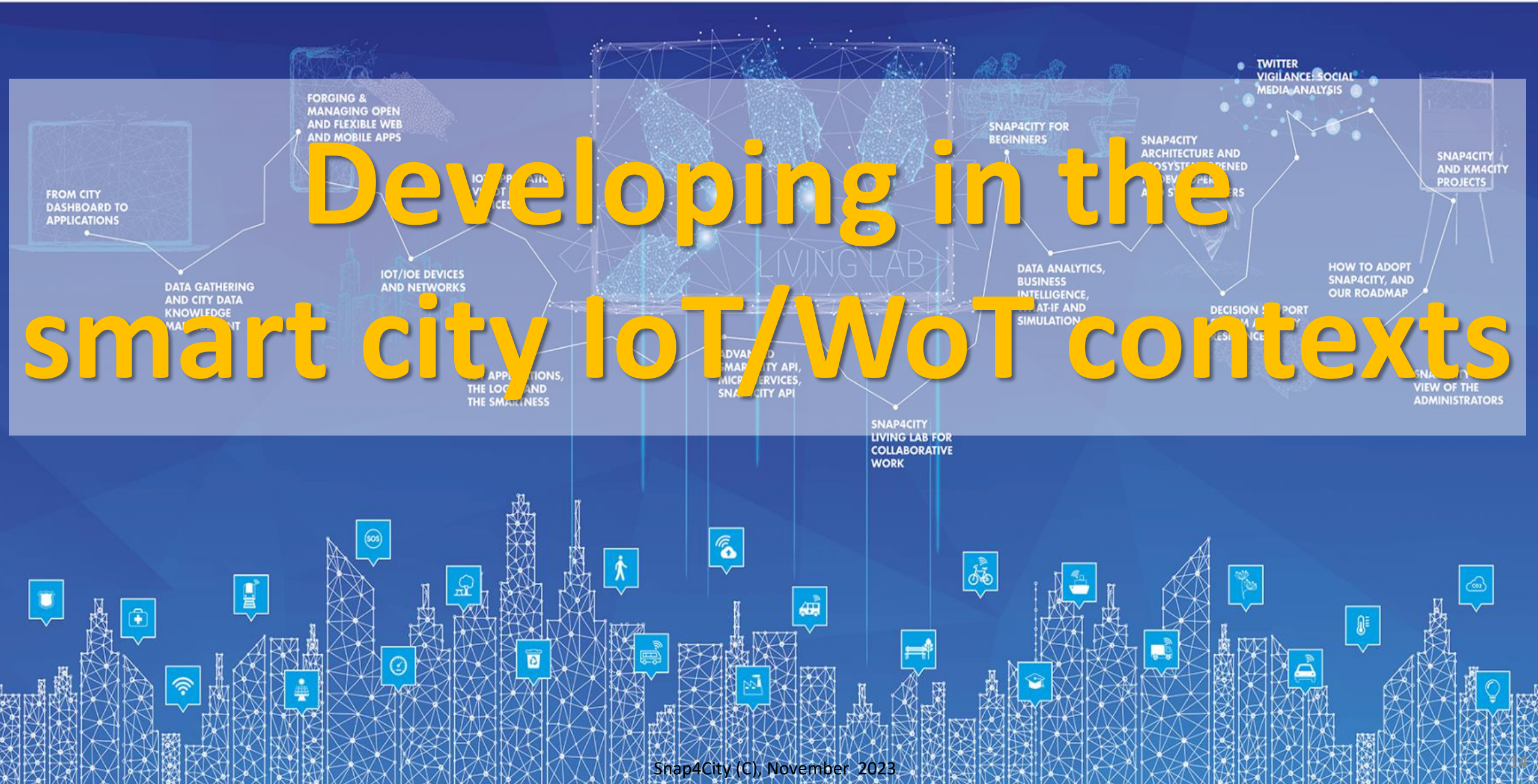
- Register on WWW.snap4city.org
  - Subscribe on **DISIT Organization**
- **You can:**
  - Access on basic Tools
  - Access to a large volume of Data
  - Create Dashboards
  - Create IOT Applications
  - Connect your IOT Devices
  - Exploit Tutorials and Demonstrations



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

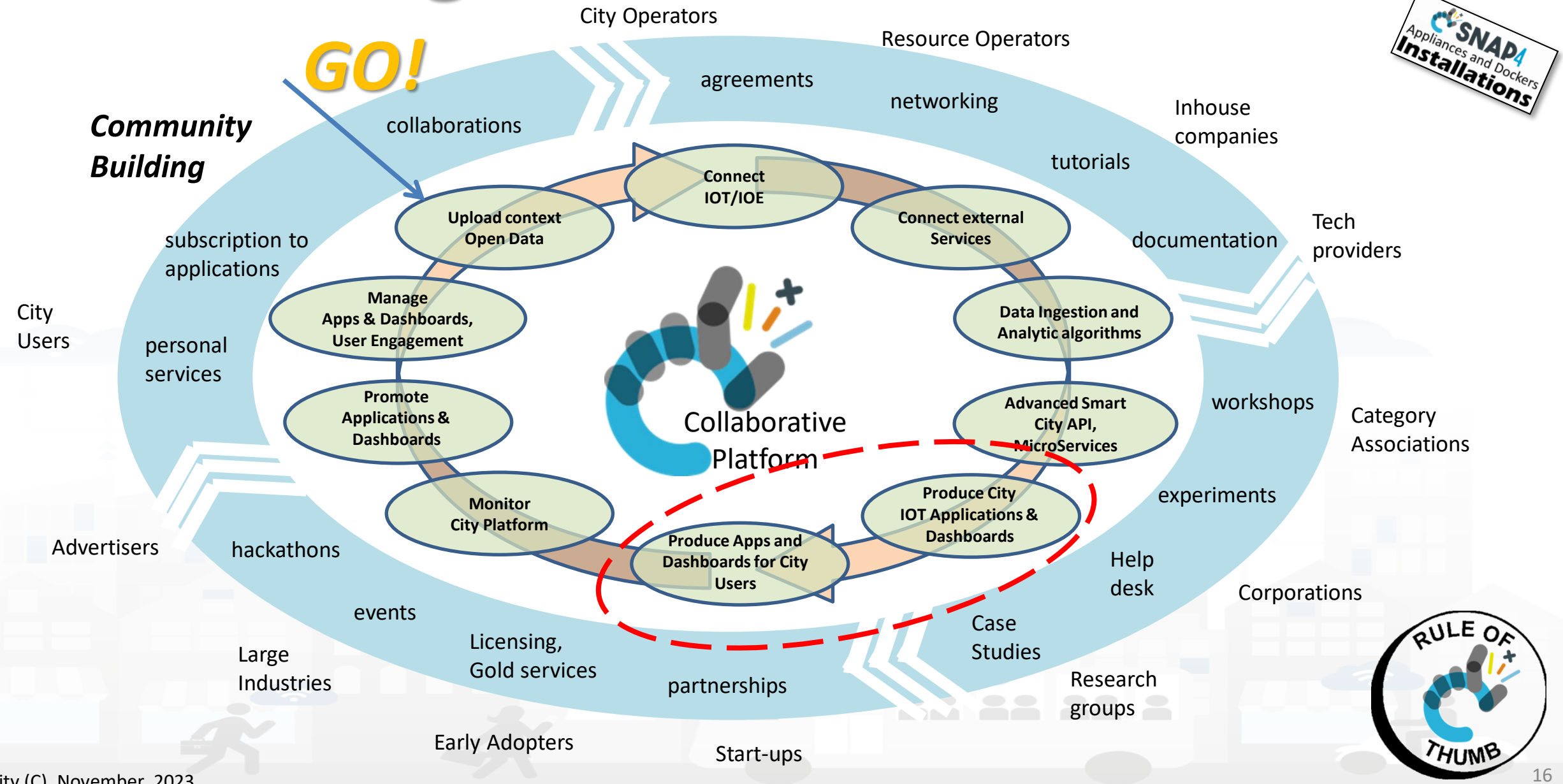
Agenda of Part 8

- **Developing** in the smart city IoT/WoT context
- Smart Solutions **Development Life Cycle**
- **Analysis** for Innovation (Co-Creation and Co-Working)
- **Design**: Data, Data Models, Data Relationships
- **Design** & **Develop**: Data Processes Proc.Logic / IoT App
- **Design** & **Develop** of Data Analytics
- **Design** & **Develop**: user interfaces, visual tools
- **Visual Analytic vs Data Analytics**: Client Side Business Logic Intelligence
- **Design and Control** of Smart Applications
- What is missing here and you can get from former course

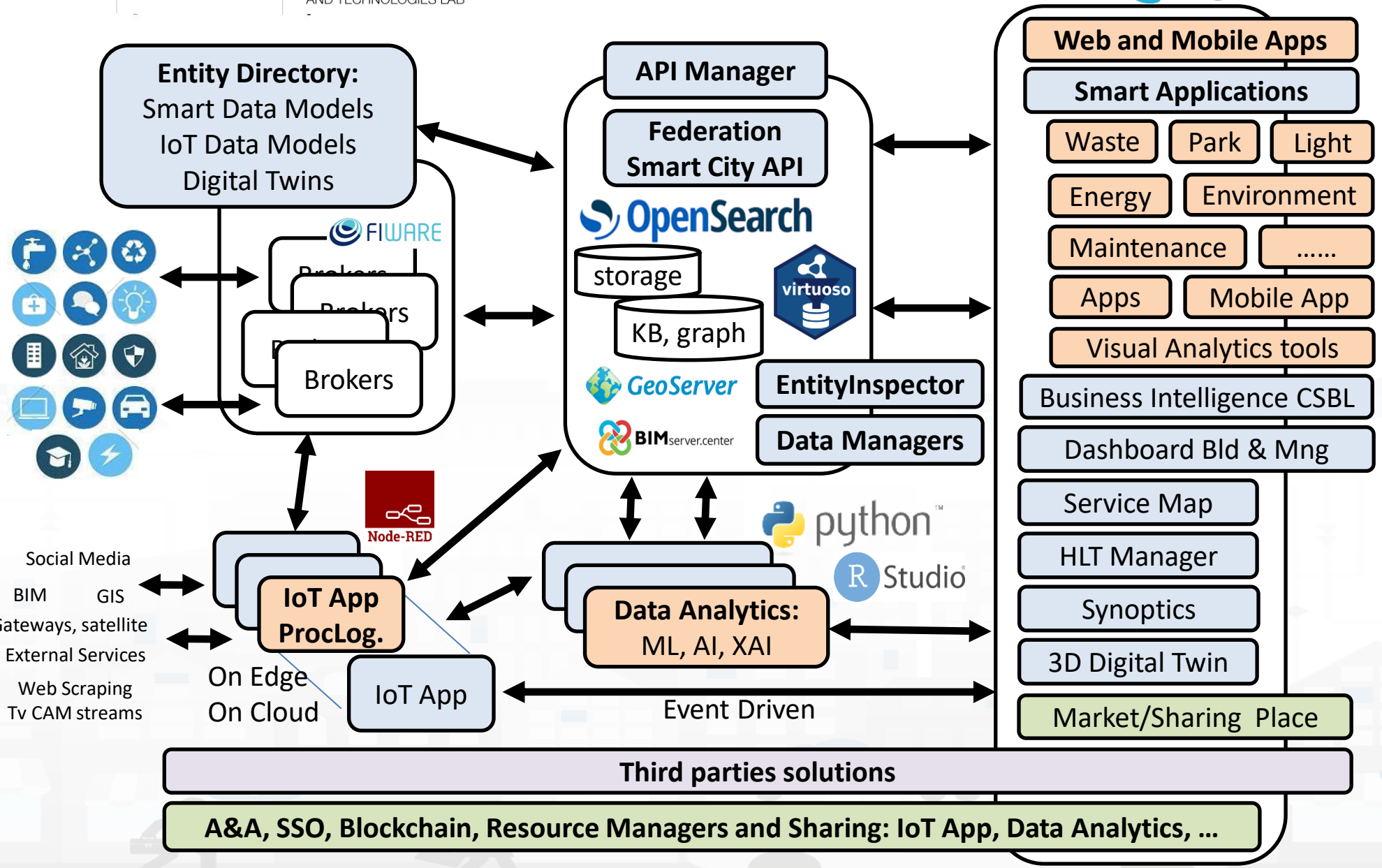


Developing in the smart city IoT/WoT contexts

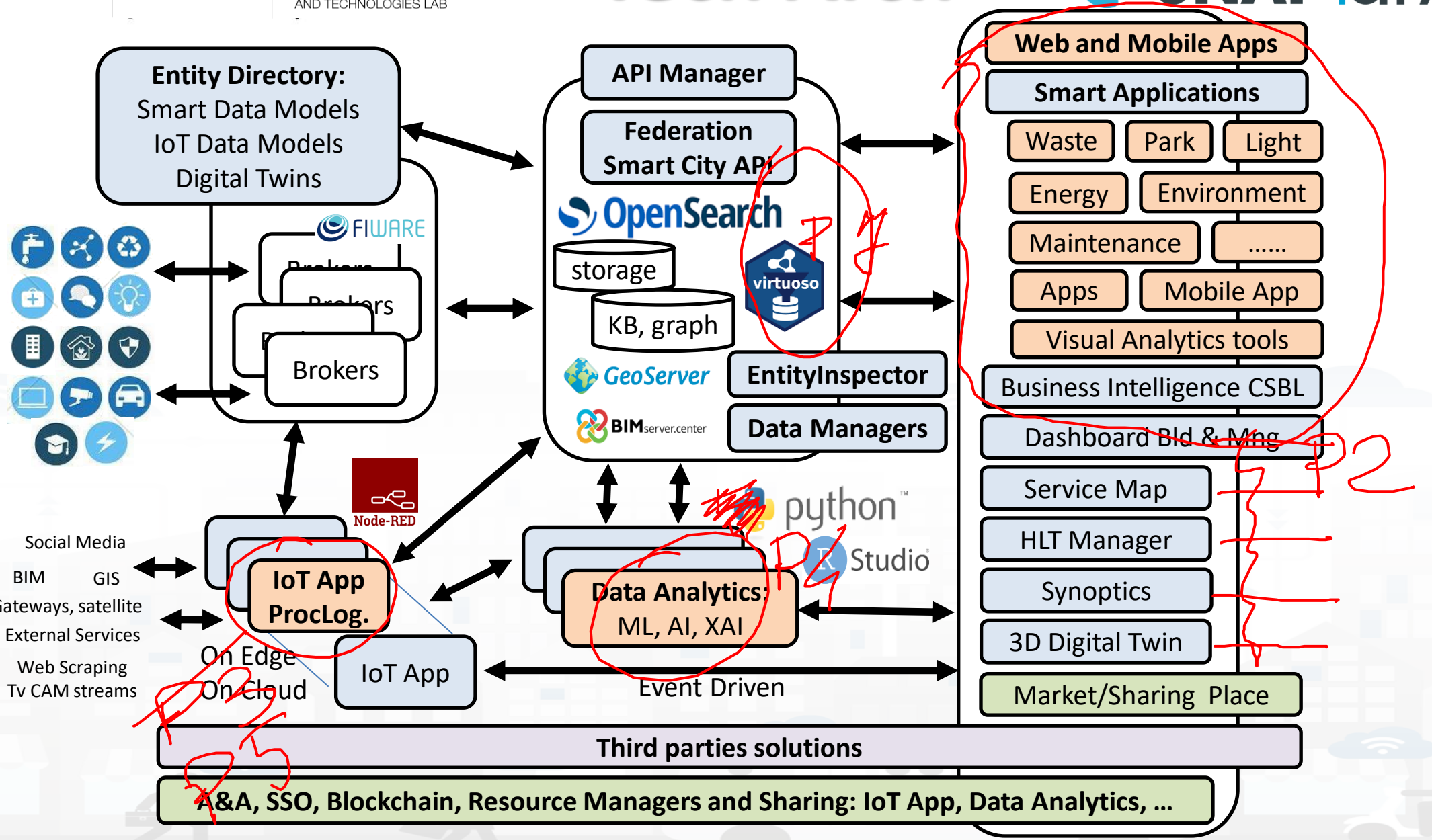
Accelerating GO!



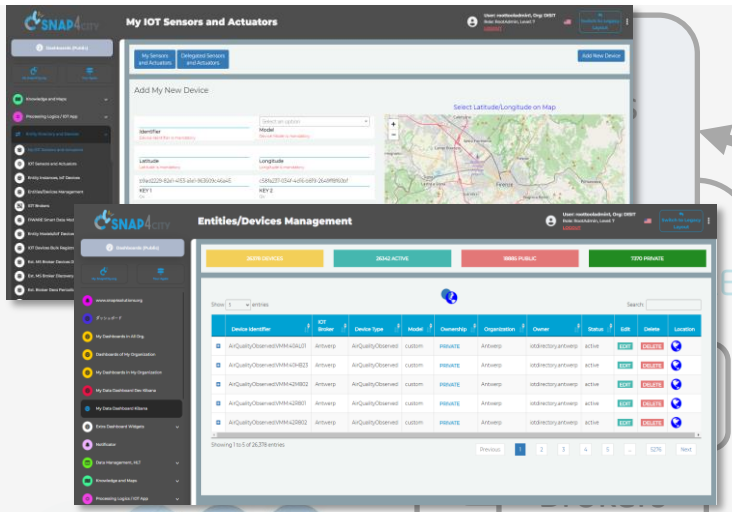
Tech Arch


Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIMserver.center
Data Managers
Node-RED
python
RStudio
Social Media
BIM
GIS
Gateways, satellite
External Services
Web Scraping
Tv CAM streams
IoT App
ProcLog.
On Edge
On Cloud
IoT App
Data Analytics:
ML, AI, XAI
Event Driven
Web and Mobile Apps
Smart Applications
Waste
Park
Light
Energy
Environment
Maintenance
……
Apps
Mobile App
Visual Analytics tools
Business Intelligence CSBL
Dashboard Bld & Mng
Service Map
HLT Manager
Synoptics
3D Digital Twin
Market/Sharing Place
Third parties solutions
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, …

Tech Arch


Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIMserver.center
Data Managers
Node-RED
python
R Studio
Social Media
BIM
GIS
Gateways, satellite
External Services
Web Scraping
Tv CAM streams
IoT App
ProcLog.
On Edge
On Cloud
IoT App
Data Analytics:
ML, AI, XAI
Event Driven
Web and Mobile Apps
Smart Applications
Waste
Park
Light
Energy
Environment
Maintenance
.....
Apps
Mobile App
Visual Analytics tools
Business Intelligence CSBL
Dashboard Bld & Mng
Service Map
HLT Manager
Synoptics
3D Digital Twin
Market/Sharing Place
Third parties solutions
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, ...

Tech Arch



API Manager
Federation Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIMserver.center
Data Managers

Web and Mobile Apps
Smart Applications
Waste
Park
Light
Energy
Environment
Maintenance
.....
Apps
Mobile App
Visual Analytics tools
Business Intelligence CSBL
Dashboard Bld & Mng
Service Map
HLT Manager
Synoptics
3D Digital Twin
Market/Sharing Place

Node-RED
python
R Studio

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

IoT App ProcLog.
On Edge
On Cloud
IoT App

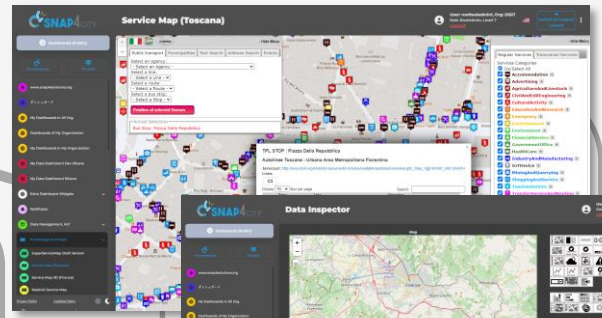
Data Analytics:
ML, AI, XAI

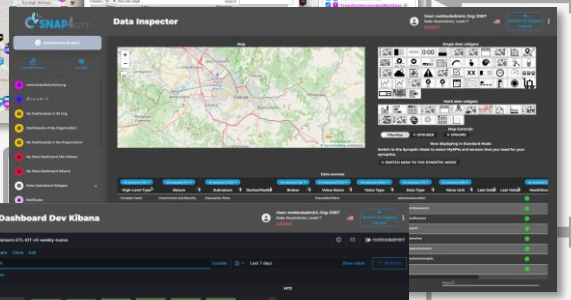
Event Driven

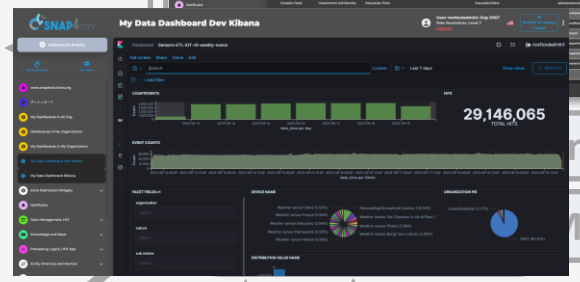
Third parties solutions

A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, ...

Tech Arch







Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins

FIWARE Brokers

Brokers

Node-RED

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

IoT App ProcLog.

On Edge
On Cloud

IoT App

Data Analytics:
ML, AI, XAI

python

R Studio

Event Driven

Web and Mobile Apps

Smart Applications

Waste | Park | Light

Energy | Environment

Maintenance | .....

Apps | Mobile App

Visual Analytics tools

Business Intelligence CSBL

Dashboard Bld & Mng

Service Map

HLT Manager

Synoptics

3D Digital Twin

Market/Sharing Place

Third parties solutions

A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, ...

Tech Arch

Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins

FIWARE Brokers

Brokers

Node-RED

API Manager

Federation Smart City API

OpenSearch

storage

KB, graph

virtuoso

GeoServer

EntityInspector

BIMserver.center

Data Managers

python

R Studio

Data Analytics:
ML, AI, XAI

Event Driven

Web and Mobile Apps

Smart Applications

Waste | Park | Light

Energy | Environment

Maintenance | ……

Apps | Mobile App

Visual Analytics tools

Business Intelligence CSBL

Dashboard Bld & Mng

Service Map

HLT Manager

Synoptics

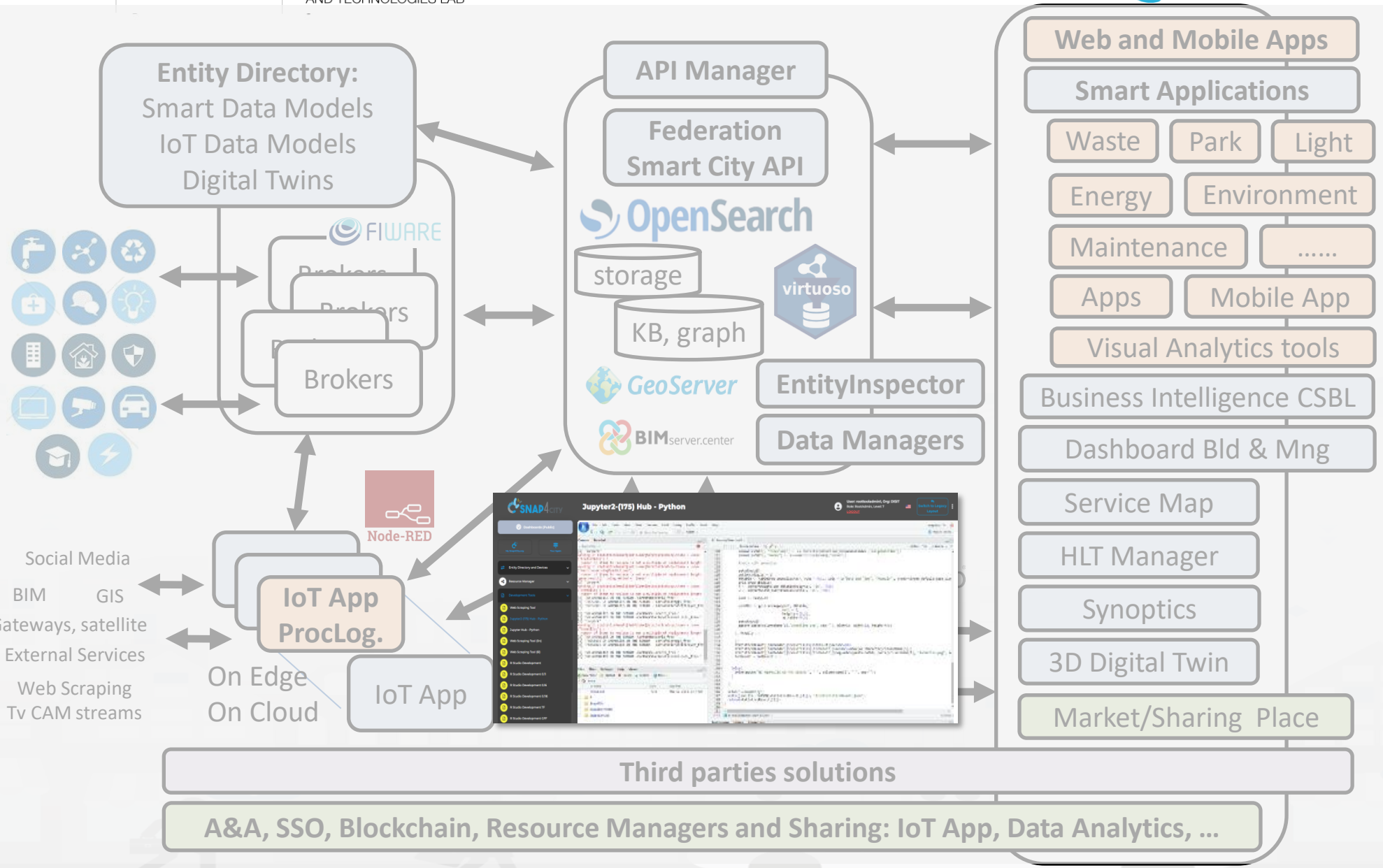
3D Digital Twin

Market/Sharing Place

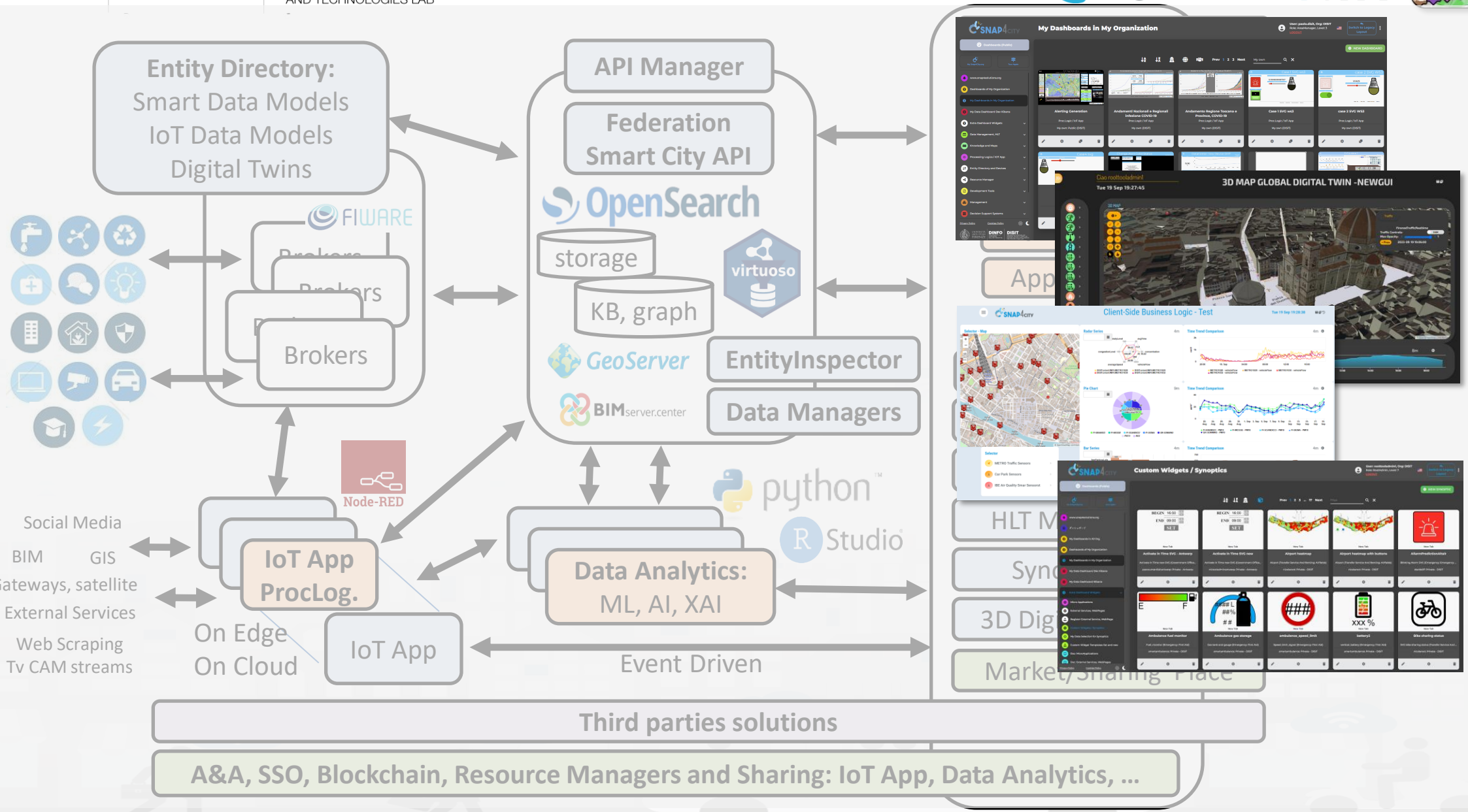
Third parties solutions

A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, …

Tech Arch


Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIMserver.center
Data Managers
Node-RED
Social Media
BIM
GIS
Gateways, satellite
External Services
Web Scraping
Tv CAM streams
IoT App
ProcLog.
On Edge
On Cloud
IoT App
Jupyter2-(175) Hub - Python
Web and Mobile Apps
Smart Applications
Waste
Park
Light
Energy
Environment
Maintenance
.....
Apps
Mobile App
Visual Analytics tools
Business Intelligence CSBL
Dashboard Bld & Mng
Service Map
HLT Manager
Synoptics
3D Digital Twin
Market/Sharing Place
Third parties solutions
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, ...

Tech Arch


Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
BIMserver.center
EntityInspector
Data Managers
Node-RED
python
RStudio
Social Media
BIM
GIS
Gateways, satellite
External Services
Web Scraping
Tv CAM streams
IoT App
ProcLog.
On Edge
On Cloud
IoT App
Data Analytics:
ML, AI, XAI
Event Driven
My Dashboards in My Organization
3D MAP GLOBAL DIGITAL TWIN -NEWGUI
Client-Side Business Logic - Test
Custom Widgets / Synoptics
App
HLT M
Sync
3D Dig
Market/Sharing Place
Third parties solutions
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, …

UNIVERSITÀ DEGLI STUDI FIRENZE

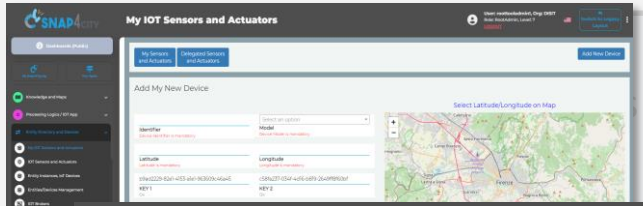
DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

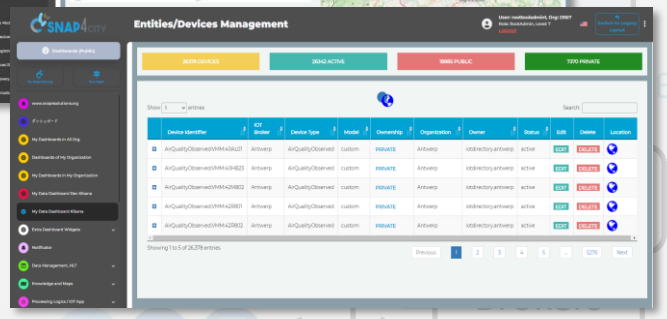
DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB
DISTRIBUTED DATA INTELLIGENCE AND TECHNOLOGIES LAB

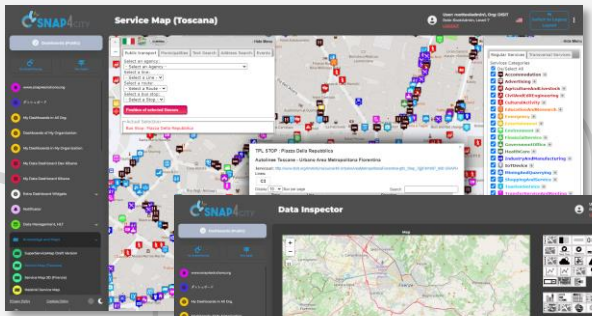
Tech Arch

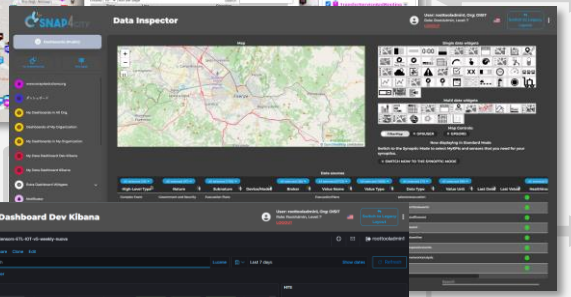
SNAP4CITY

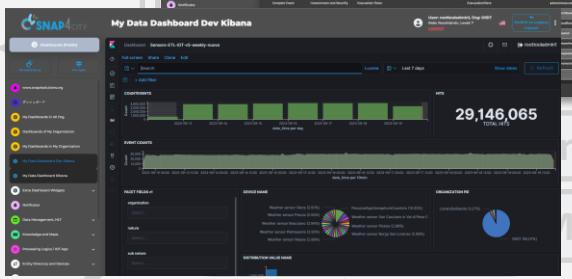
KM4CITY

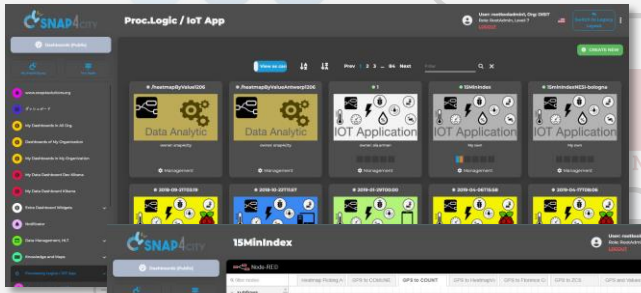


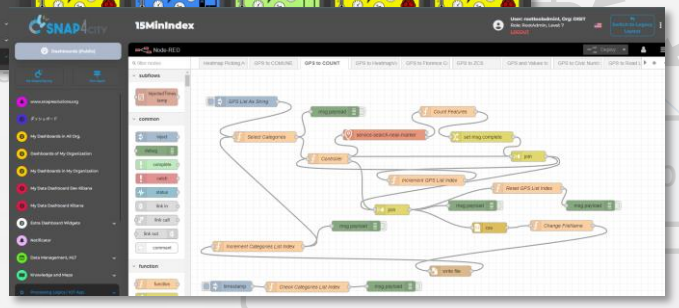


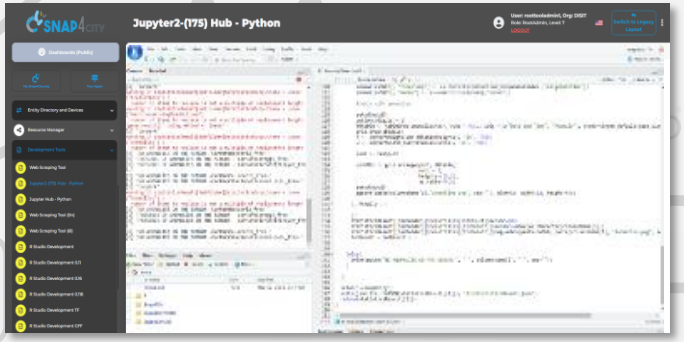


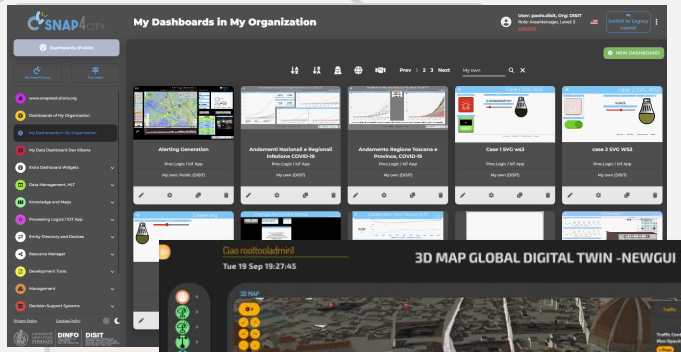


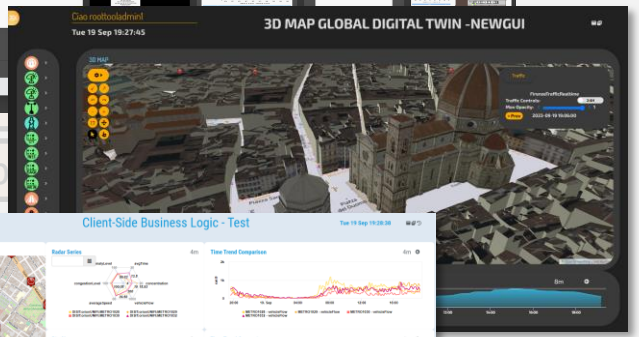


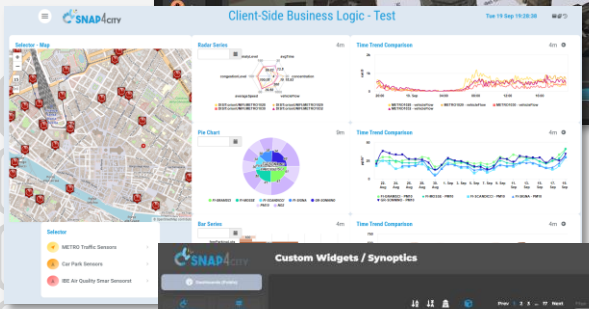












Node-RED

Third parties solutions

A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, …

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/ and internal for some…
  - **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 / IoT App** any protocols: https://www.snap4city.org/65 They can also expose some specific API, custom made

Your Applications and IPR in Snap4City

- **Data Models**: Entity Models / IoT Device Models, Smart Data Models, etc.
- **Proc.Logic / IoT App**: data ingestion, adapter, transformation, wrappers, business logic, transcoding, integration, interoperability, algorithms, etc.
- **Data Analytics:** algorithm and processing in RStudio or Python, ML, AI, XAI, etc.
- **User Interface Design**: Dashboards, client-side business logic, Synoptics, widgets, templates, styles, etc.
- **Client-Side Business Logics (if any)** realized in JavaScript on Dashboard widgets.
- **Server-Side Business Logics (if any)** realized in Processing Logic as Node-RED and JavaScript.
- and the **data instances** for the High-Level Types.

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

legenda

SNAP4CITY | KM 4 CITY

Presentation: Control Room, Widgets, Mobile Apps, Visual Analytics Applications, Telegram Bot, …

Dashboard Builder | Dashboard Mng | Synoptics | 3D Digital Twin | Dashboard Wizard | Data Inspector | Scenario Editor | CSBL

External Interoperability: Smart City API, Federated, API Accounting/Billing

Super | ASCAPI | kafka | WebSocket

Operation:

What-if | Simulation | Event Reporting | File Mng | Heatmap Mng | Traffic Flows Mng | ODM Mng | TVCam Mng | BIM Mng | .....

Studio | python | docker | .....

KPI, Indicator Analysis | WorkFlow Tickets BPM | Sentiment Analysis | SUMO | Tipyc. Time Trend | ..... | Open to any module and system

Data Processing | Predictions | Anomaly detection | Statistics | Artificial Intelligence | Routing | Internal GIS GeoServer | ckan | openMAINT | BIMserver.center | .....

Internal Interoperability: API, MicroServices

Snap4City Node-red Libraries | ASCAPI | WebSocket | FIWARE

Data Storage and Reasoning: data storage, noSQL, aggregation, semantic modeling, city entities, normalization, knowledge base

Service Map | LOGraph | SuperServiceMap | DevDash | OpenSearch | nifi | virtuoso

NGSI V1, V2, LD | FIWARE

Linked Data triples

Data Collection: data mining, harvesting, integration, transformation, data models, …

Directory | Blockchain | Harvester | Data Manager | Brokers | FIWARE | Node-RED

Any protocol and format | Any protocol and format

Connectivity: wired, wireless (Lora, 5G, 4G, 3G, Wi-Fi, etc…), IoT Edge, etc.

Node-RED

Any protocol and format | Any protocol and format

GIS | BIM | OSM | Gateways | External Services | Web Scraping | Social Media

External Third Party Services

Device Layer

Platform Management: docker | SNAP4CITY

Node-RED

Proc.Logic / IoT Applications Data Flow Logic

Authentication and Authorization: KEYCLOAK | OpenLDAP | SNAP4CITY

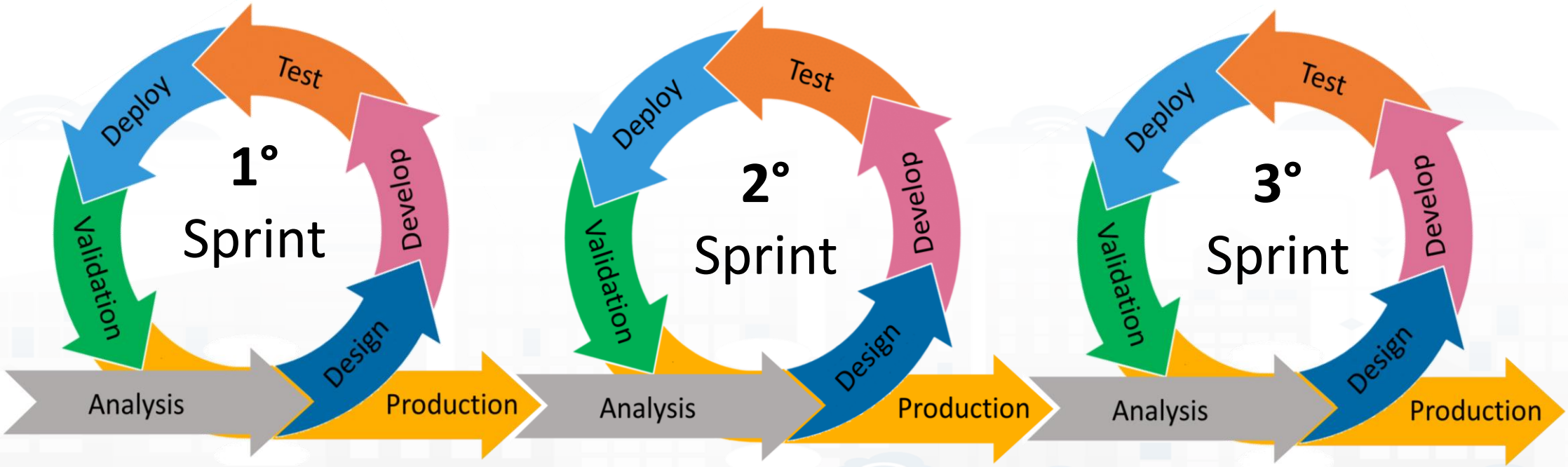
GDPR compliant

Partecipatori and Living Lab Supports

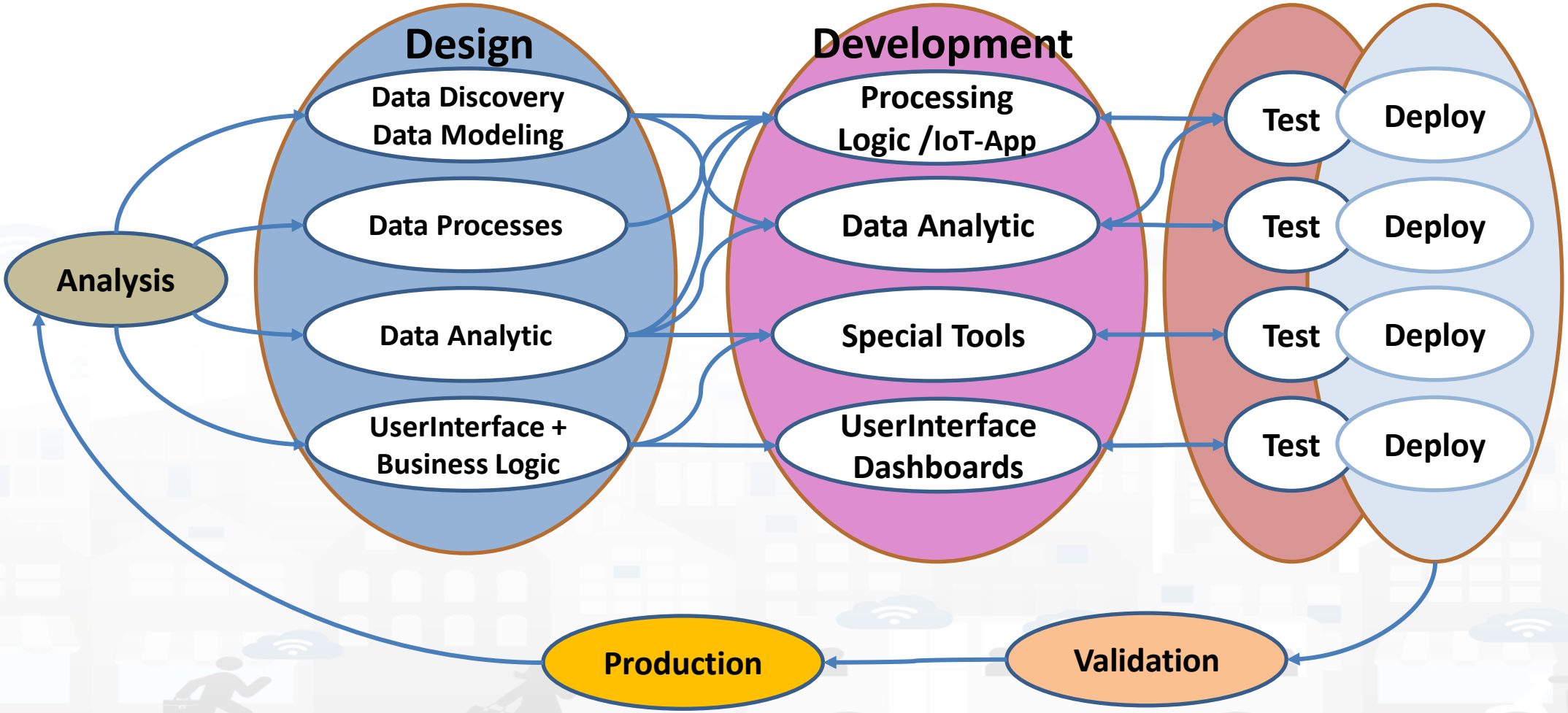
TOP

Smart Solutions Development Life Cycle

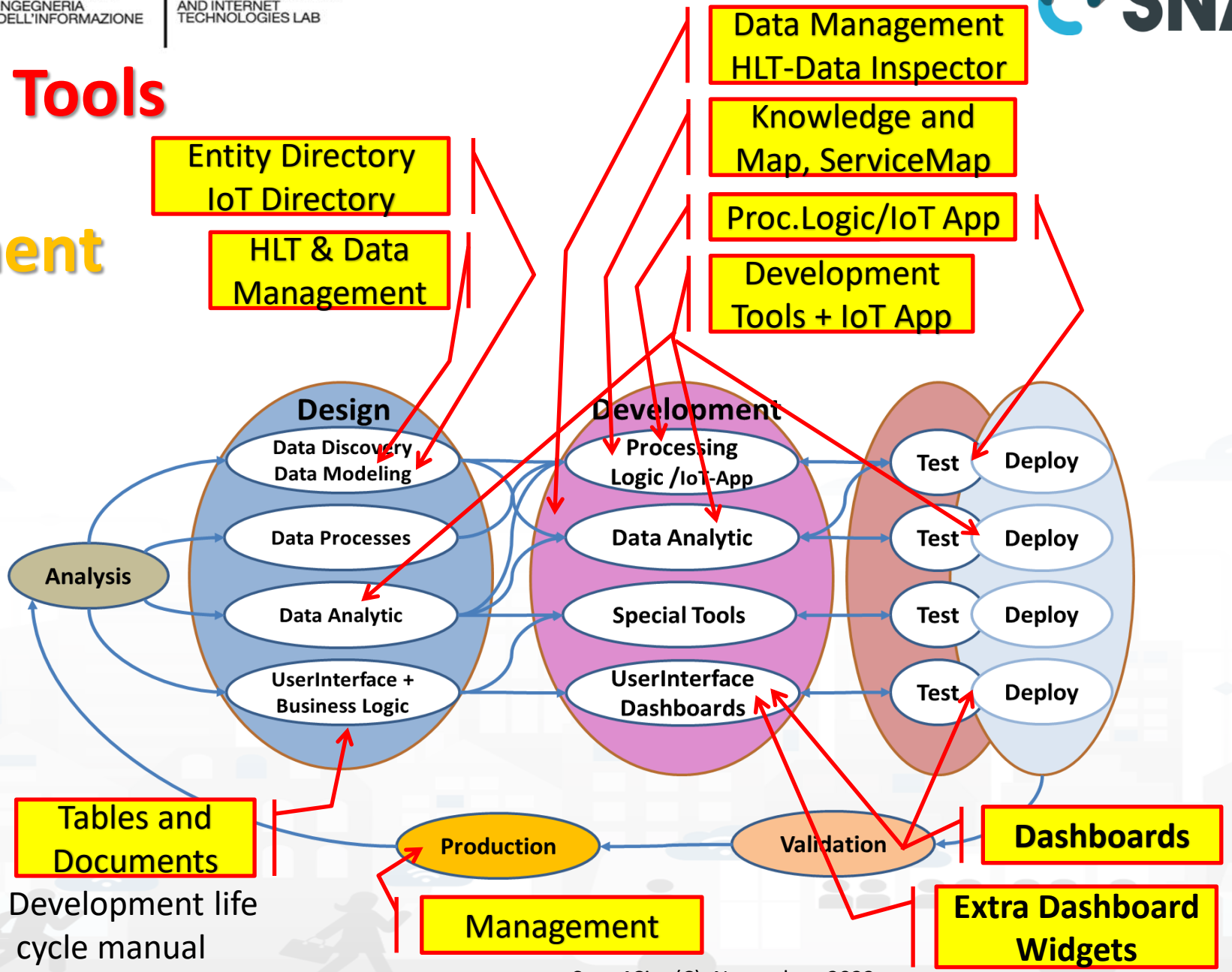
Development Life Cycle
Smart Solutions
Agile: CD-CI, Continuos Dev – Continuos Improvement



Development Life Cycle Smart Solutions



Snap4City Tools vs Development Life Cycle



Development life cycle manual

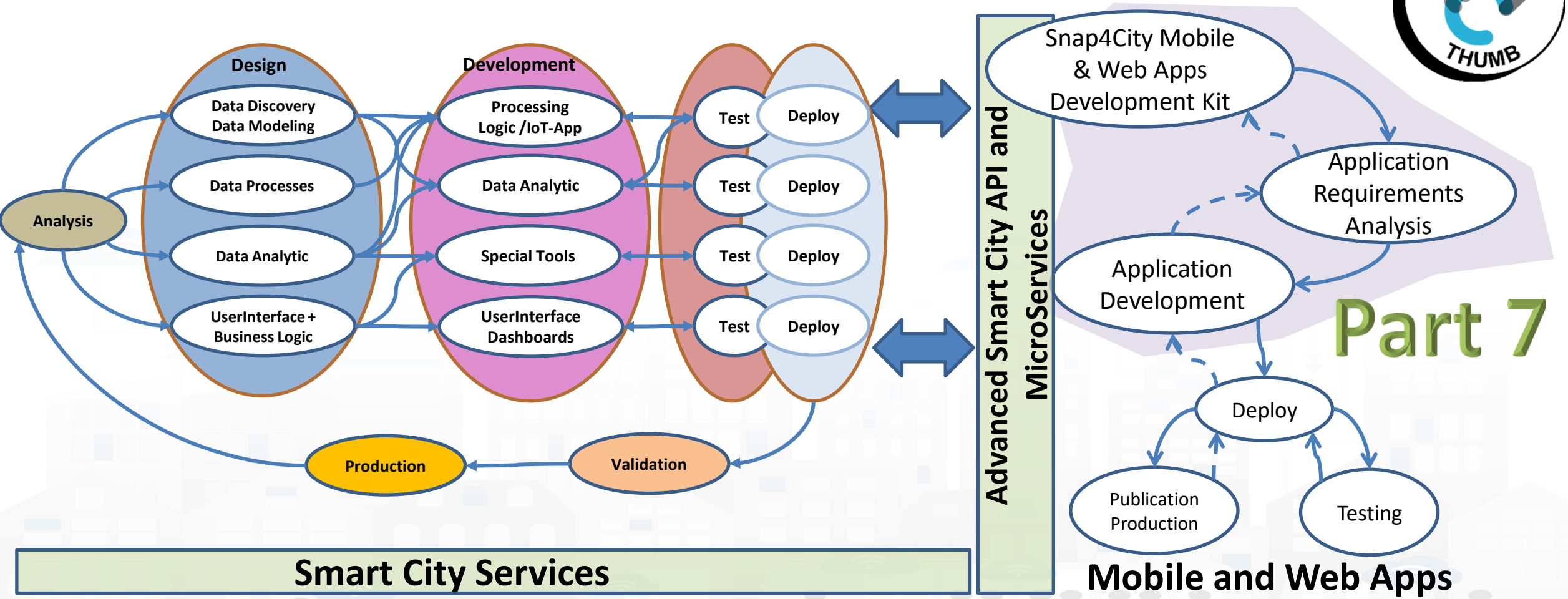
Test & Deploy

- 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

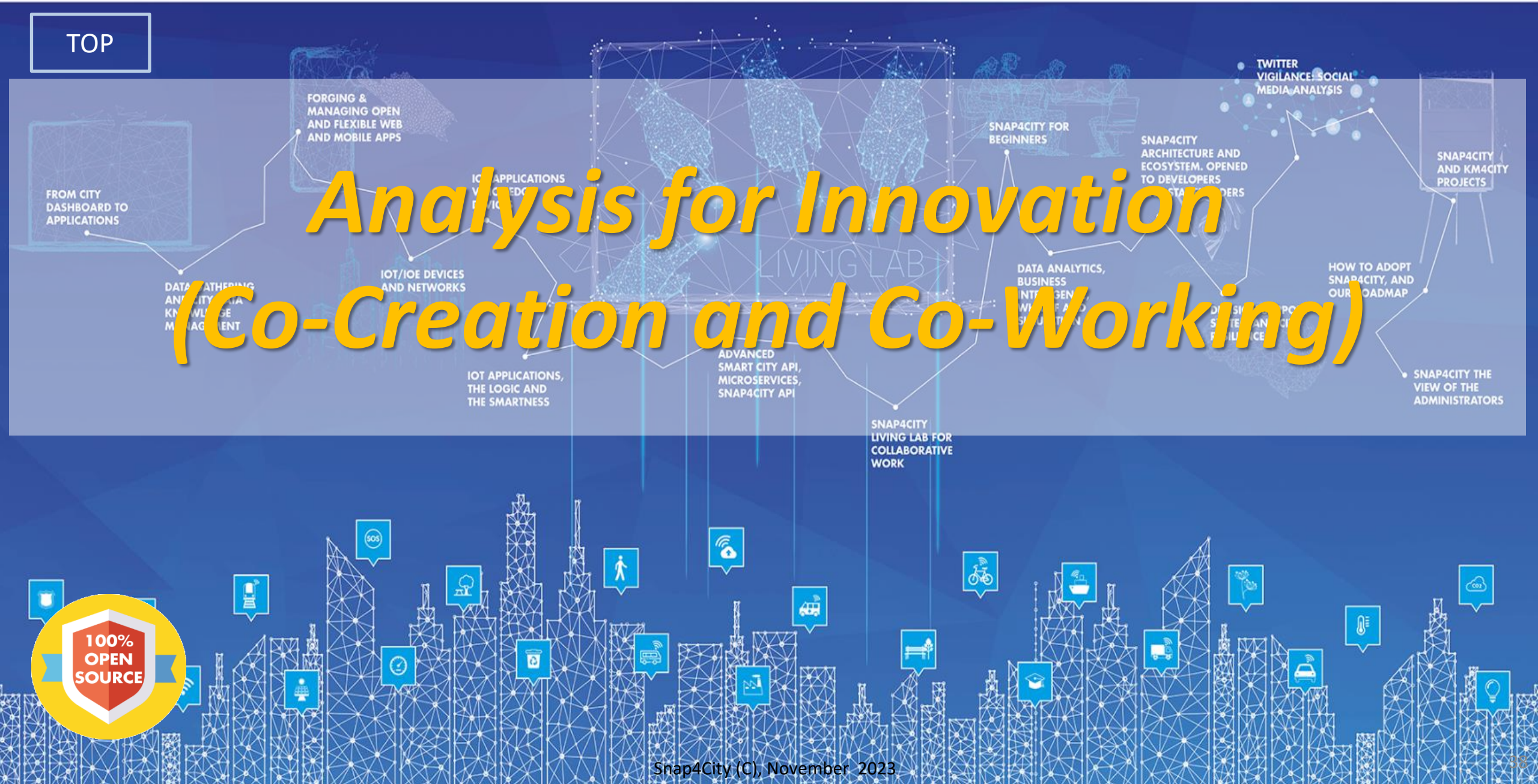
- Is the phase in which all components can be integrated and tested in their integration on the platform ready to be used in production.
- The **validation** should be performed verifying:
  - Functional Requirements
  - Non functional Requirements
- The **production** process is very easy in Snap4City 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



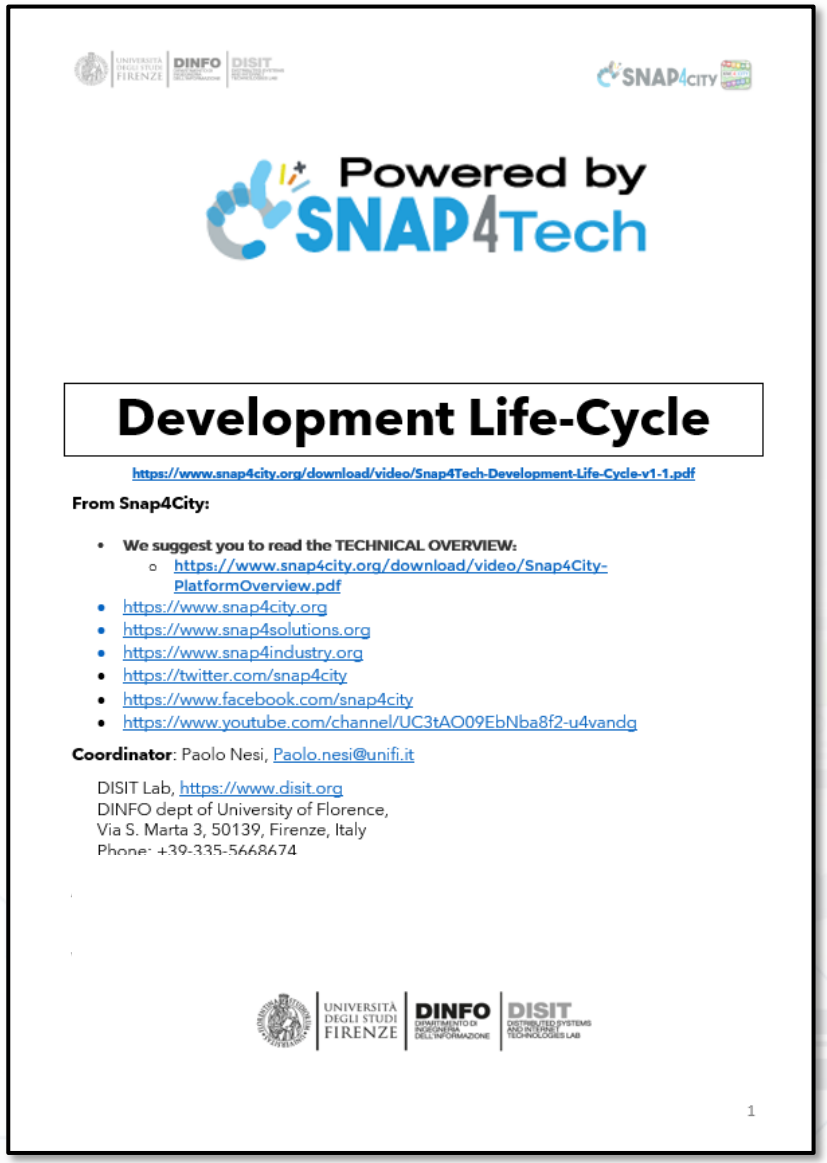
TOP

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



Development

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



Powered by SNAP4Tech

Development Life-Cycle

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

From Snap4City:

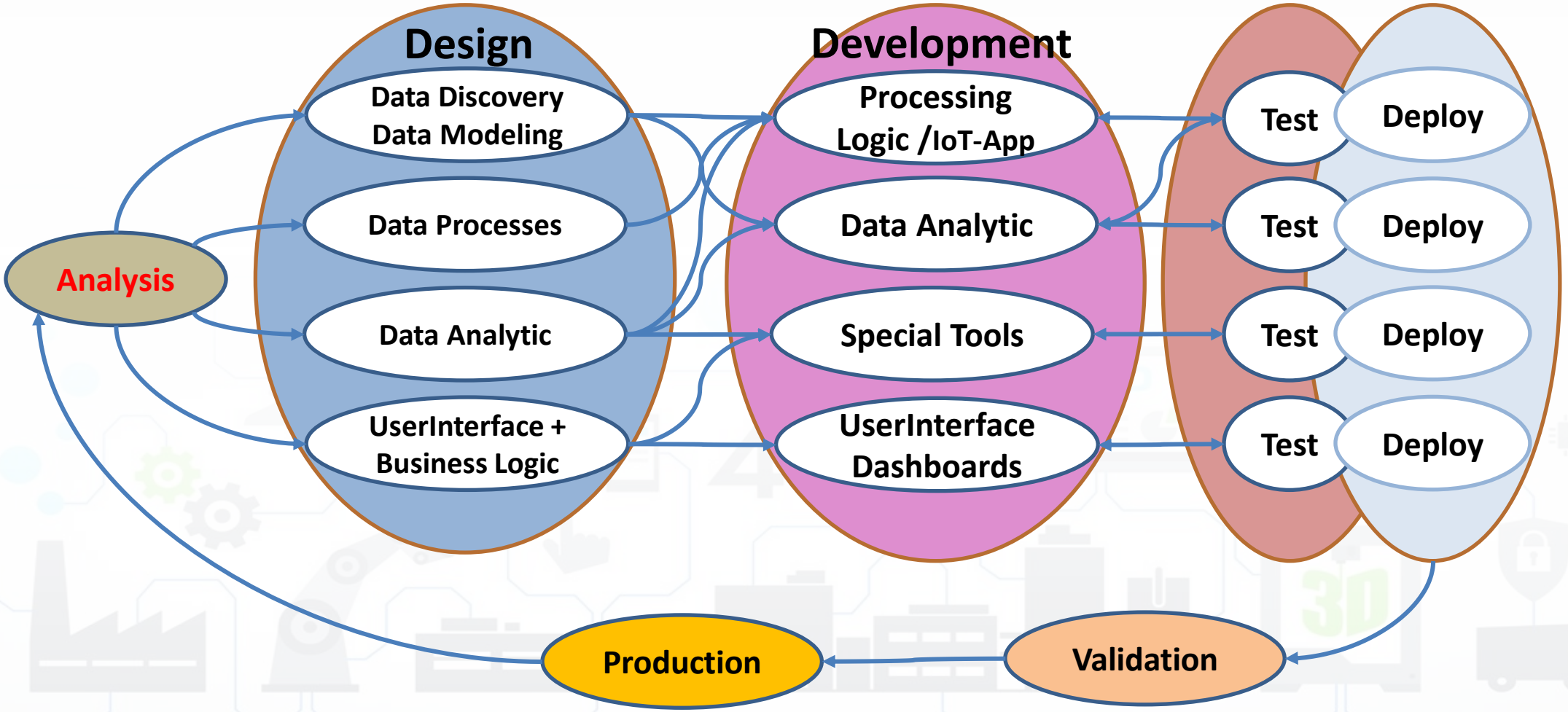
- **We suggest you to read the TECHNICAL OVERVIEW:**
  - https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- https://www.snap4city.org
- https://www.snap4solutions.org
- https://www.snap4industry.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city
- https://www.youtube.com/channel/UC3tAO09EbNba8f2-u4vandg

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

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

1

Development Life Cycle Smart Solutions

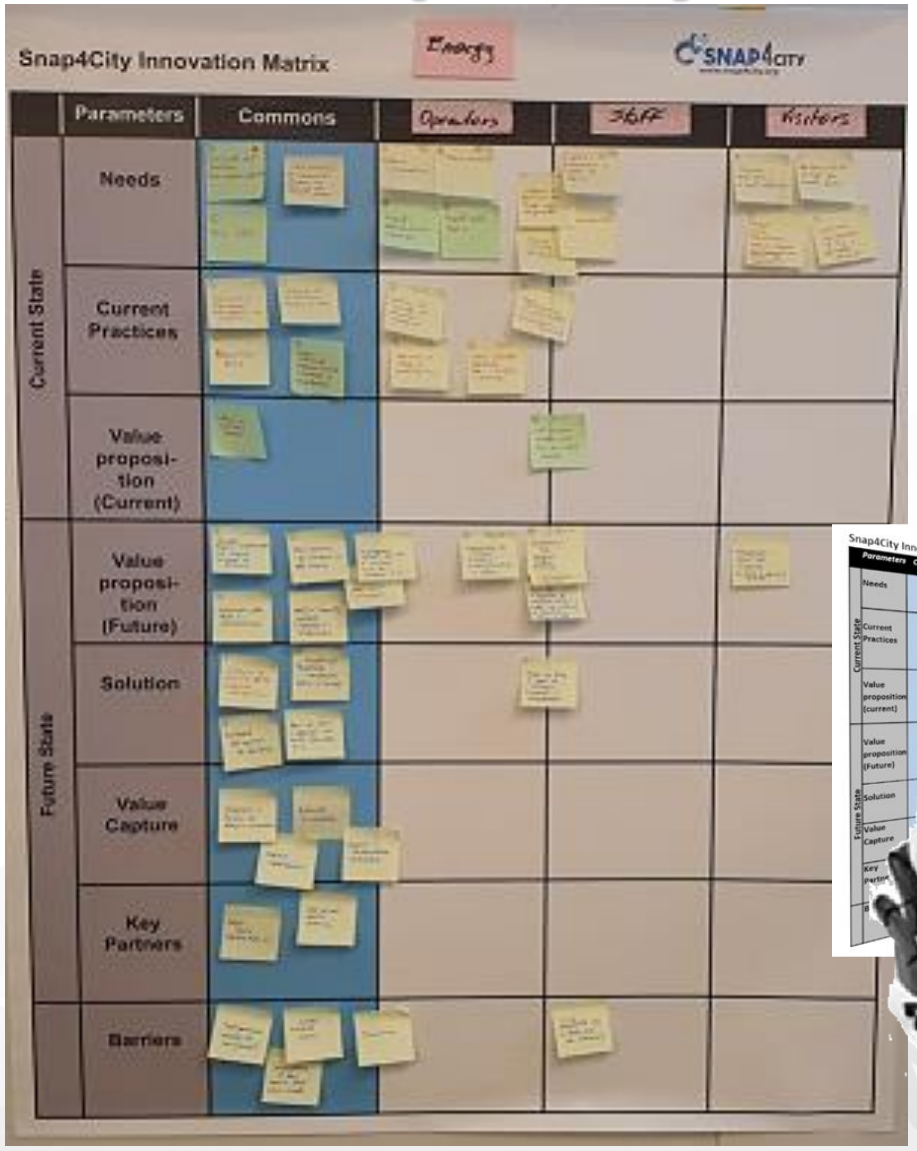


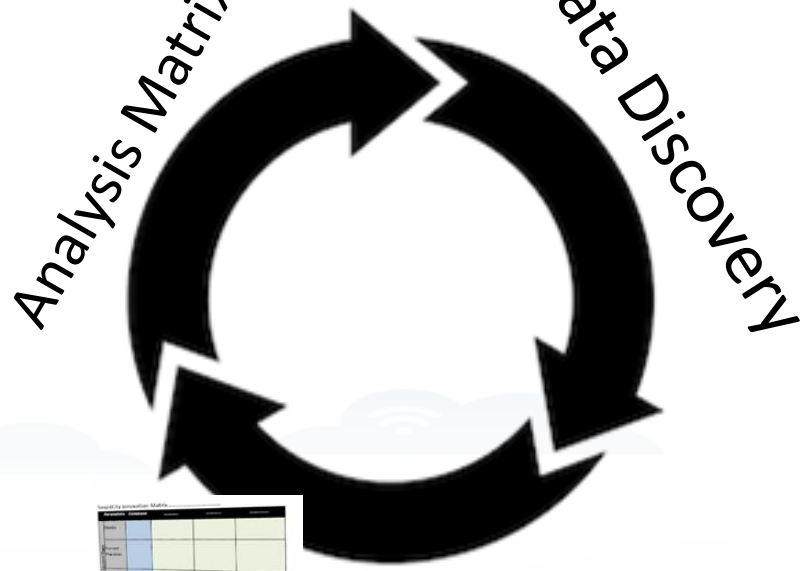
Analysis 1/2

- **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.
  - **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.
    - **External API:** to interoperate with any other application and service / servers.
    - **External Services / Web Pages:** 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.

Snap4City Innovation Matrix and Process

Part 6



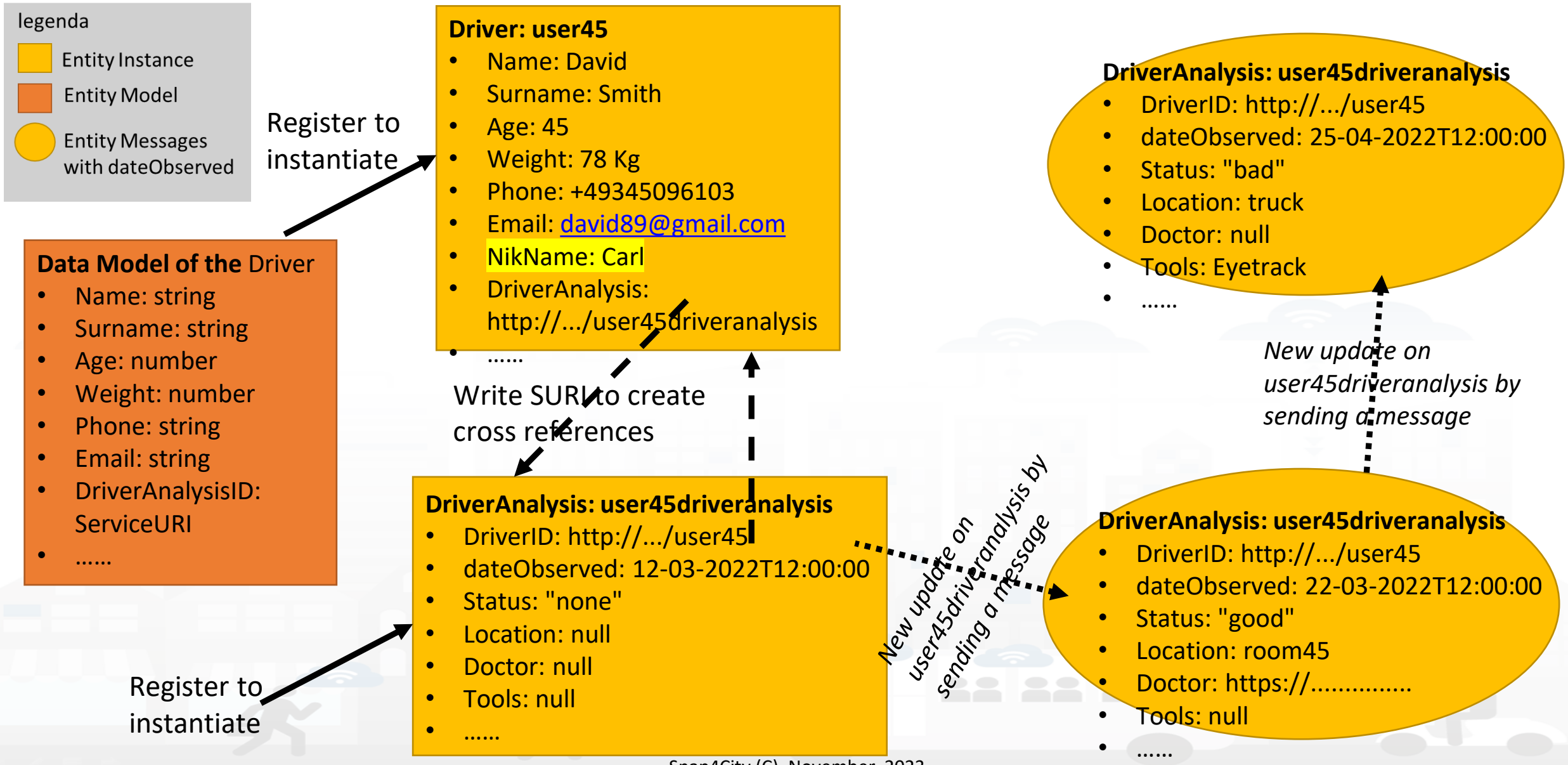


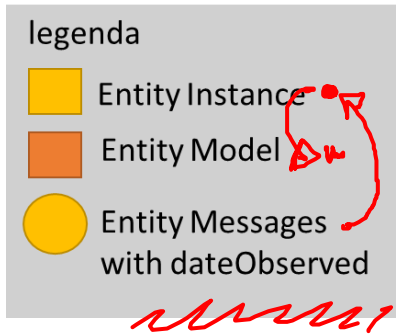
The Dictionary of Entities

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
		IoT App			
		Dashboards			

Columns in green are expected to be filled in the design phase

- **For example:** *Let us now to suppose that we have to develop a solution for monitoring Vehicles and Drivers. Each Vehicle has a profile description and can be driven by a number of Drivers over time.* ***Each Vehicle*** *can experience some maintenance and performs trips in the city area. A trip has an official start/end and over time is described by its velocity, acceleration, brakes, charging level, or thank level, etc.* ***Each Drive****r has a profile and can use a number of Vehicles to perform trips. During the trip also the Driver is monitored for its healthiness, attention, etc., and before, during and after the driving, periodically or sporadically may experience some Analysis to certify its capability to drive in that moment and for the next days. The Driver may experience some warning cases for healthiness, some tickets from policeman, some warning for high-speed velocity or generically bad driving, some problems from the vehicle's status, etc.*


legenda
Entity Instance
Entity Model
Entity Messages with dateObserved
Register to instantiate
Data Model of the Driver
Name: string
Surname: string
Age: number
Weight: number
Phone: string
Email: string
DriverAnalysisID: ServiceURI
……
Driver: user45
Name: David
Surname: Smith
Age: 45
Weight: 78 Kg
Phone: +49345096103
Email: david89@gmail.com
NikName: Carl
DriverAnalysis: http://.../user45driveranalysis
……
Write SURI to create cross references
DriverAnalysis: user45driveranalysis
DriverID: http://.../user45
dateObserved: 12-03-2022T12:00:00
Status: "none"
Location: null
Doctor: null
Tools: null
……
Register to instantiate
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
……



Register to instantiate

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

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

Write SURI to create cross references

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

Register to instantiate

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
- ……

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
- ……

API, External Services

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

Columns in green are expected to be filled in the design phase

External Services				
URL Web pages	parameter	Description	Nature	Subnature

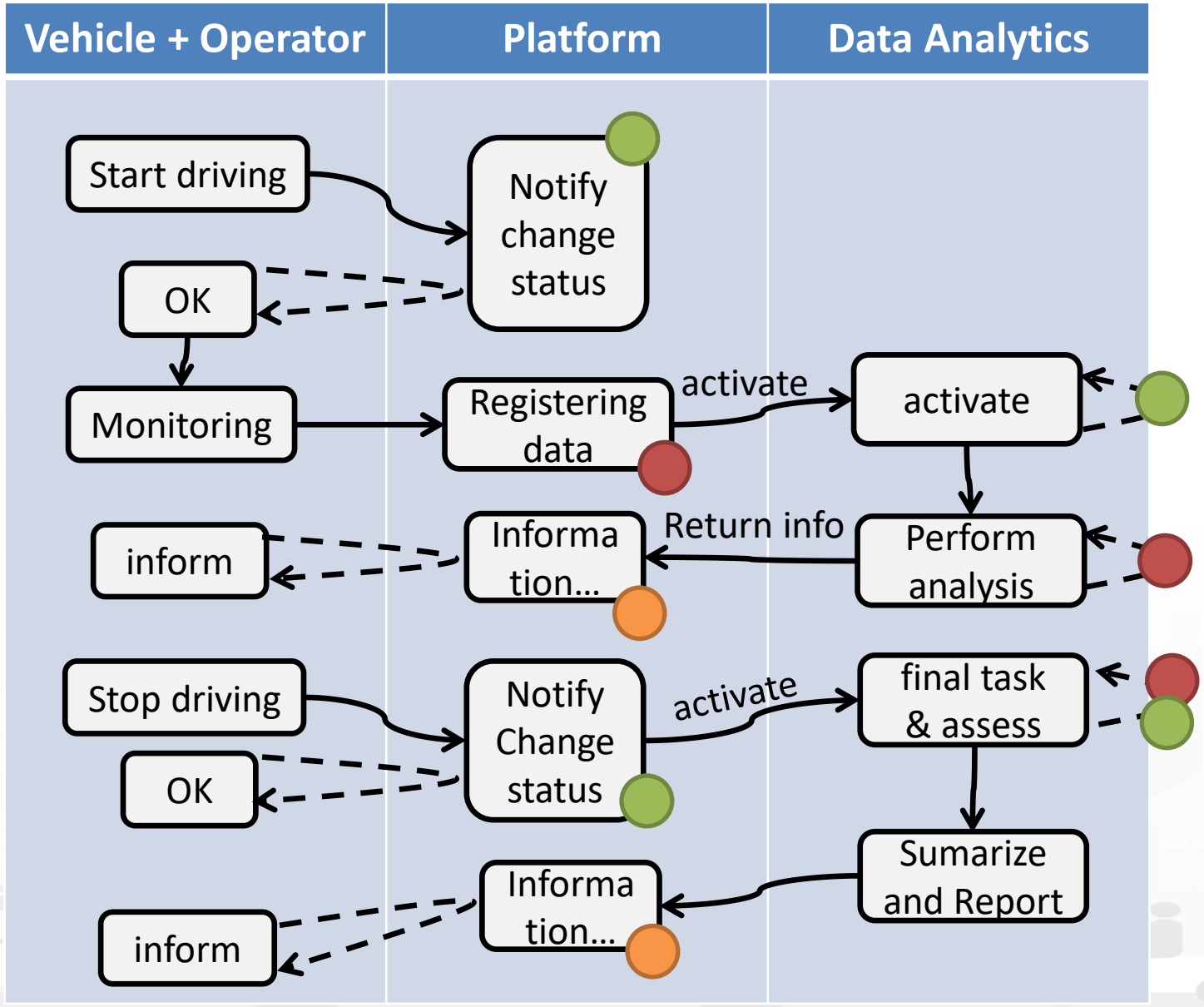
These info can be loaded on Snap4City platform to show them on dashboards easily

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

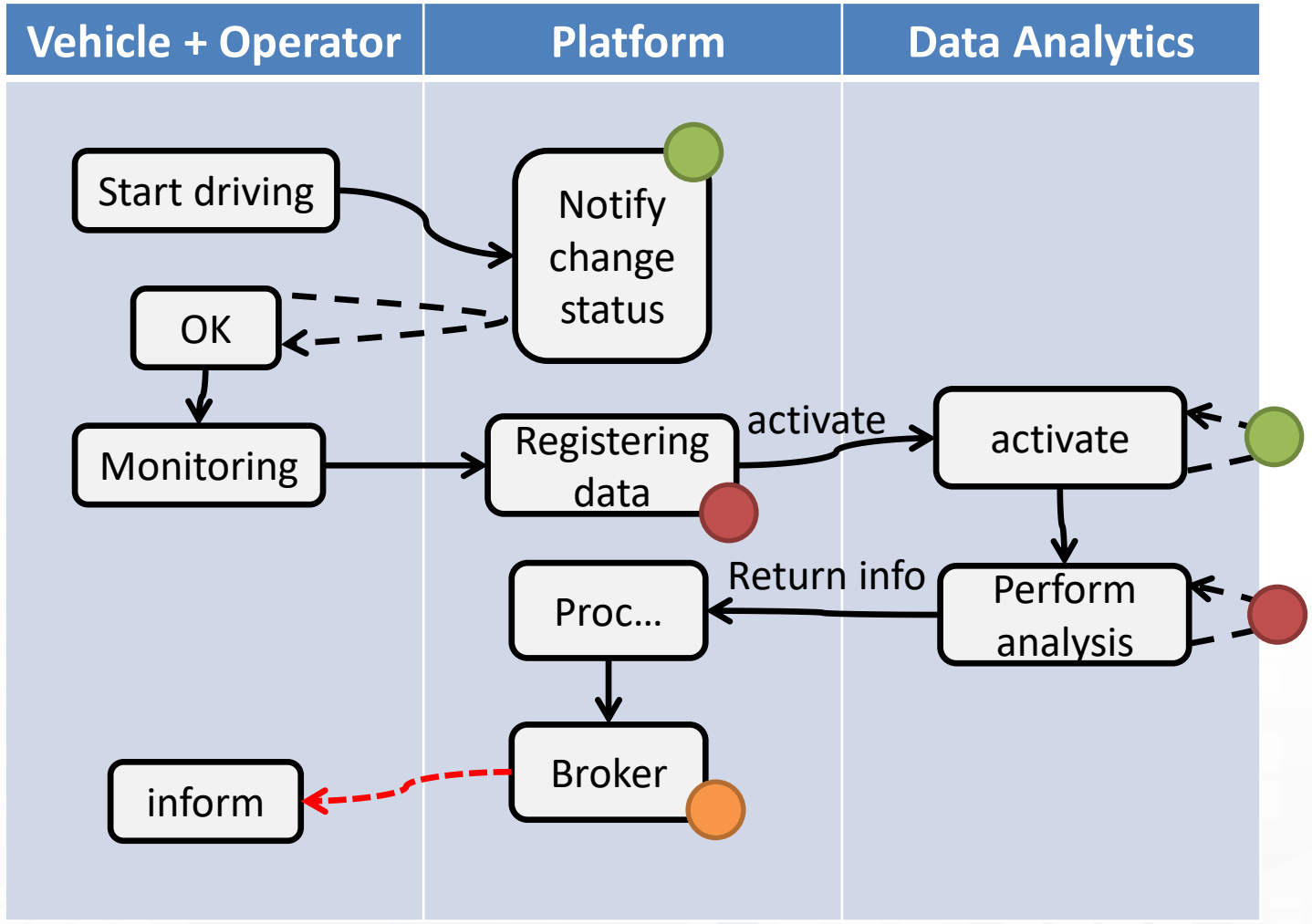
Example: Activity Diagram

- **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



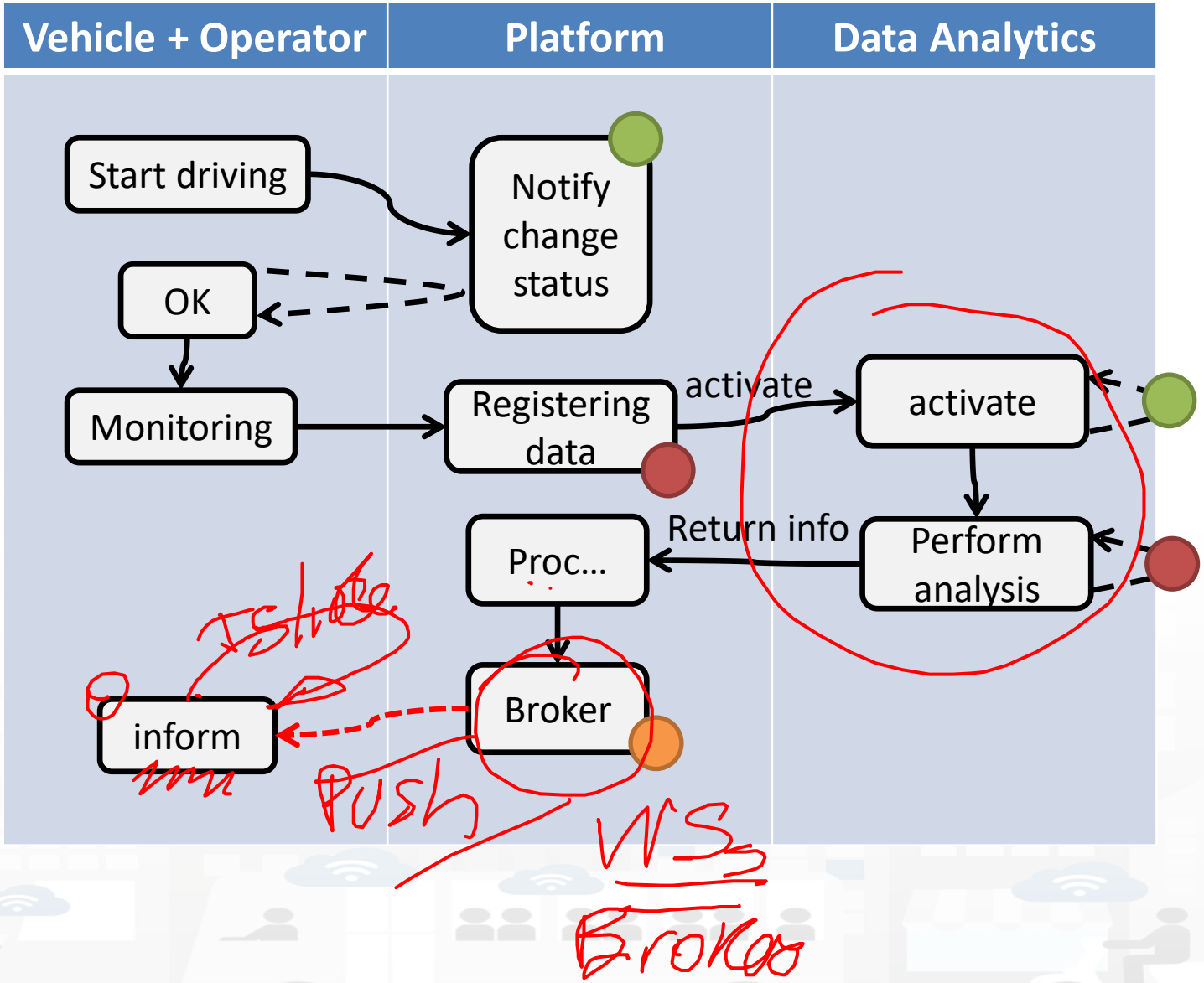
A variant

- 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)**



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**

Fun 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
	OperatorTool	Has to provide the list of pending assessment to be done				

Columns in green are expected to be filled in the design phase

Non Functional Requirements

Somehow related each other

- Protection, privacy, PENTest, GDPR compliance, …
- Scalability, performance, efficiency, cloud/edge/container compliance
- Resilience, robustness
- Modularity, flexibility, reusability, maintainability, ..
- Portability, Openness, opensource
- Interoperability, standards compliance
- Responsive, usability, ..
- Etc.

All largely covered by Snap4City platform

Be carefull

- **Despite** the Snap4City platform provides a full range of Non-Functional Requirements
  - You can with your analysis and design produce poor solutions
- **For example, it is not a good approach to:**
  - Collect user profiles and putting them public
  - Collect data every 10 second of phenomena which change only once a day
  - Couple your web/mobile applications with server-side processes by using synchronous communication in a context which is not synchronous and neither real time

- ***As a general remark:***

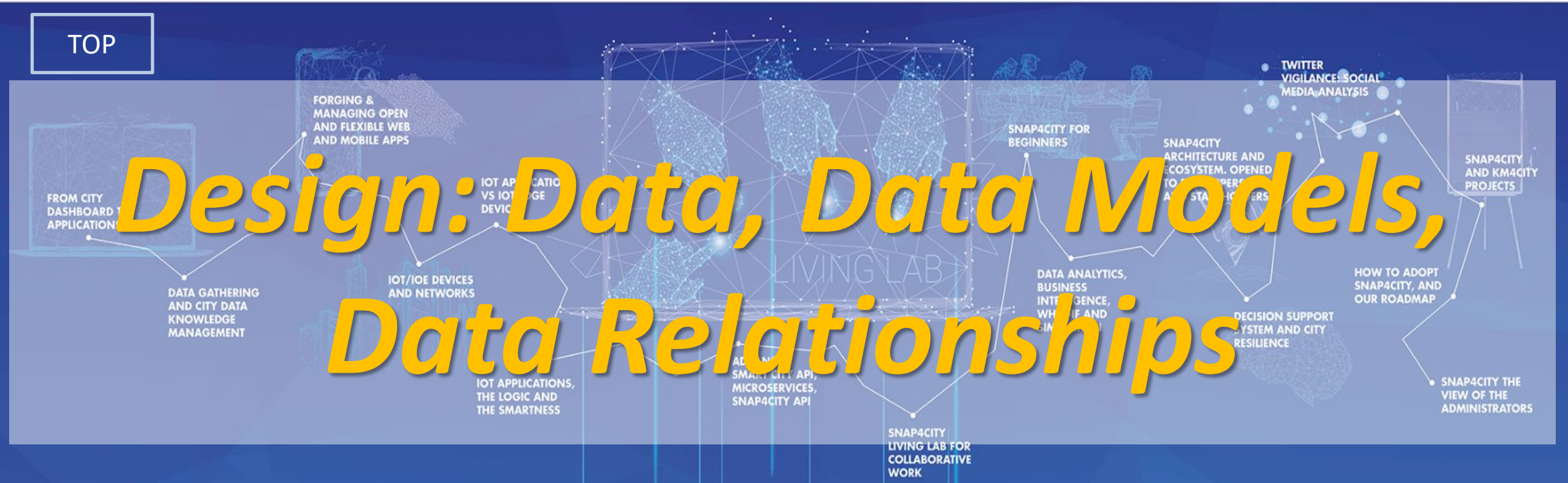
Do not worry *if at the first sprint of the above steps you forgotten to fill some details. It is quite sure that, you have also provided some details that would have to be revised/changed at the next iteration.*

The suggestion is ***start developing from the core parts****, which are the production of Entity Instances from the Entity Models, the ingestion of Entity Messages for the Entity Instances, etc., and detailing the most relevant and innovative Use Cases with respect to the state of the art.*

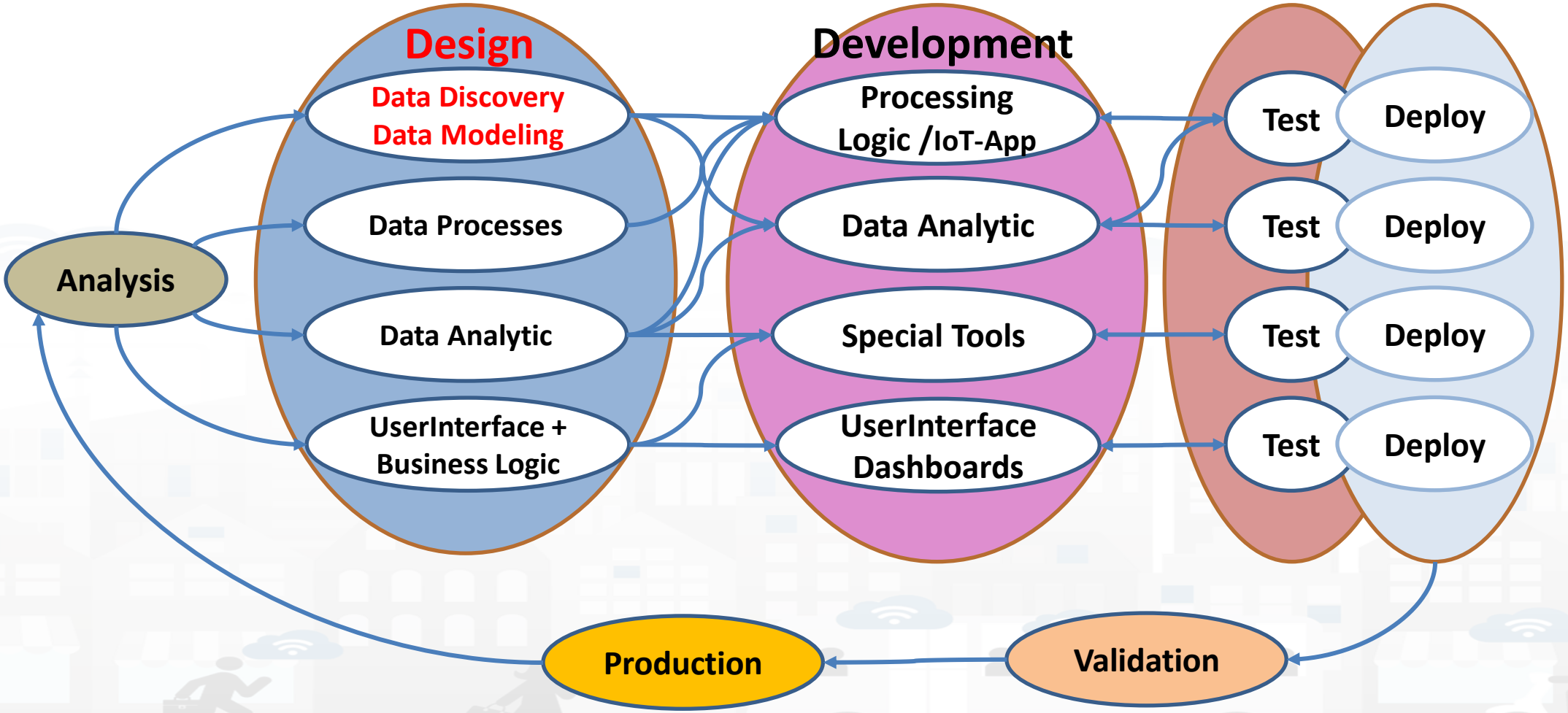
They would leverage the smart solutions to a new level, at each sprint.

TOP

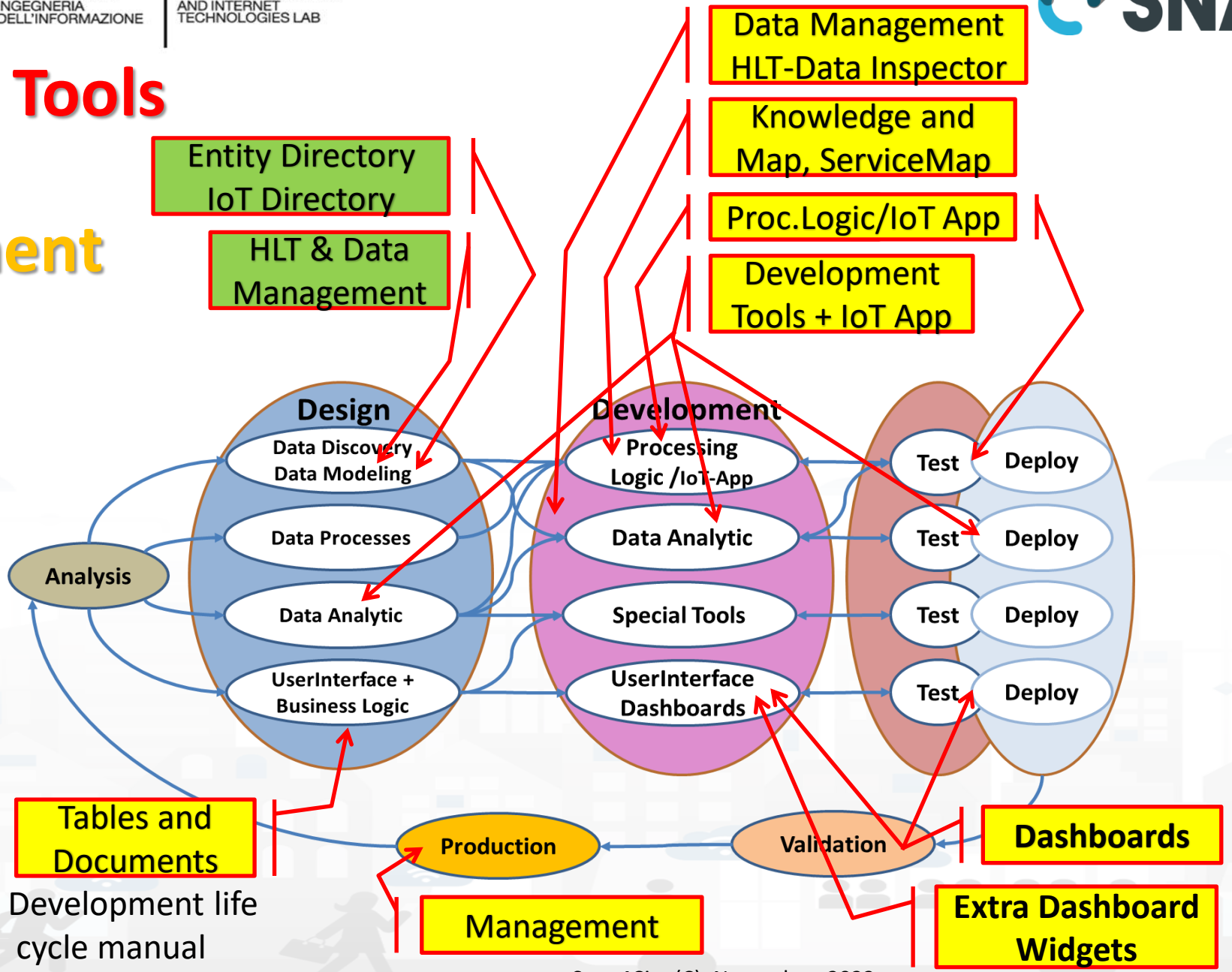
Design: Data, Data Models, Data Relationships



Development Life Cycle Smart Solutions



Snap4City Tools vs Development Life Cycle



Development life cycle manual

Design: Data Discovery

Data Discovery

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

Data Collection per Domain/Scenario

Description	domain	S/CO/RT	I/O	Type	Status	Referent	Provider	endpoint	Authentication	HL protocol	protocol	HLT	Format	Size	Volume	Rate	GPSed	foto	License / Condition of use
Graph road	Energy	Static	In	Struct	Understood	Name Surname	Stakeholder ID	url	Simple	Push	Datex	Sensor	XML	2 variables	10Byte	Every 10 minutes	Yes	URL,	Public as CC…
Parking	Graph	Real Time	Out	Non struct	Acquired	Email	Staff or not staff	Broker	Certificate	Pull	WS	Sens-Actuator	JSON	15 fields	1245 Kbyte	Sporadic, max 1000 times per day	No	IMG	Link to file
Consumption of energy	Mobility	Combined	In/out		Scheduled	Phone	Internal..		Etc.		REST	KPI	GeoJSON			Periodic	Kind		Private …
	Transport	RT stream			Tested	Etc..					Custom	Personal Data	KMZ			2 per day	Inside msg		Restricted to …
	….	RT Messages			Operative						ODBC	Ext Srv	WFS				Static …		Sensible data
					Failed						JDBC	IOT	WMS						GDPR aspects
					Not needed							Virtual Sensor	GTFS						
												GIS	db						
												Heatmap							
												Path, trajectory							
												Trend							
												….							

Examples are provided per colum.

The resulted raws may have not sense.

The status refers to the ingestion process.

Design: Data Modeling

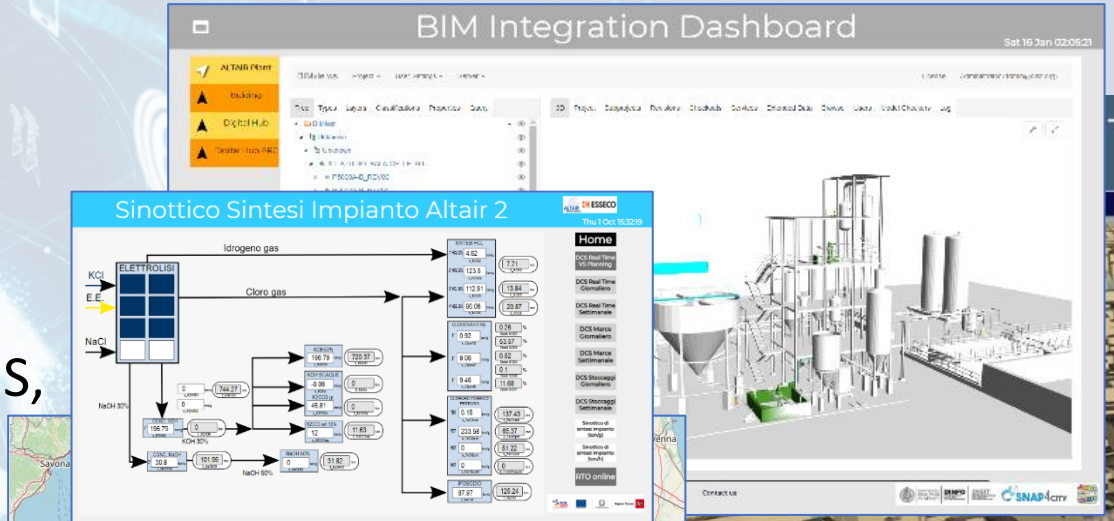
Part 4

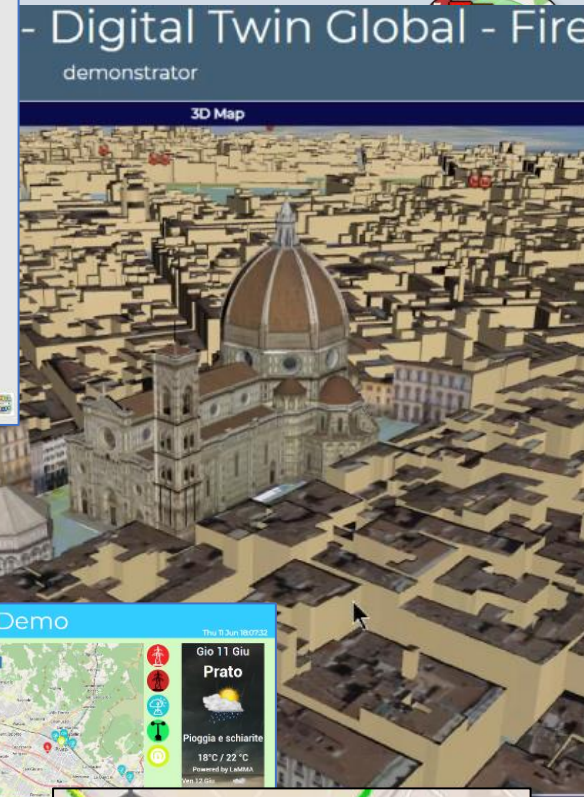
Part 5

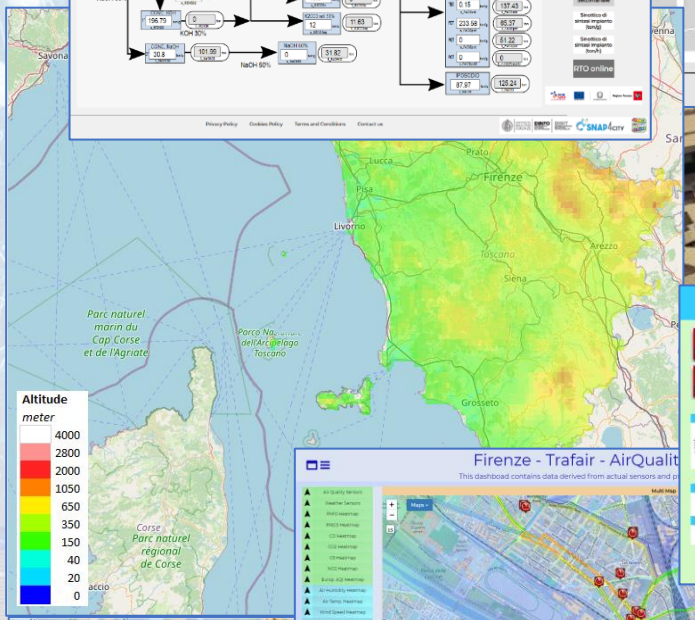
High Level Types

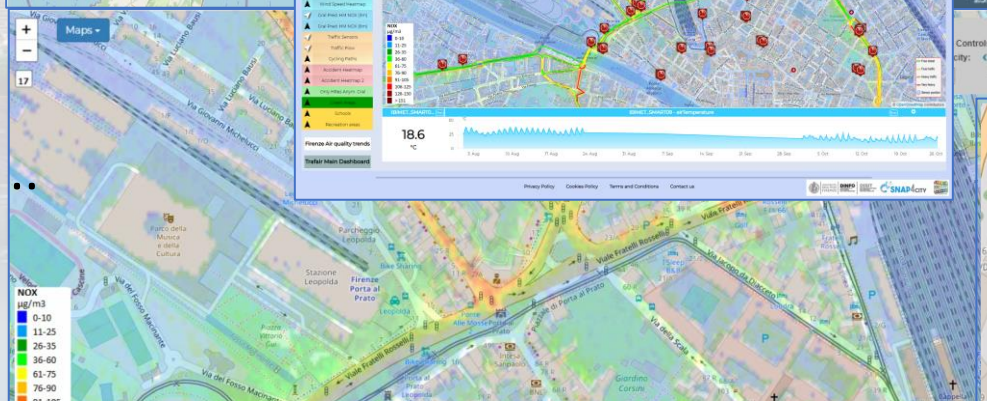
Snap4City (C), November 2023

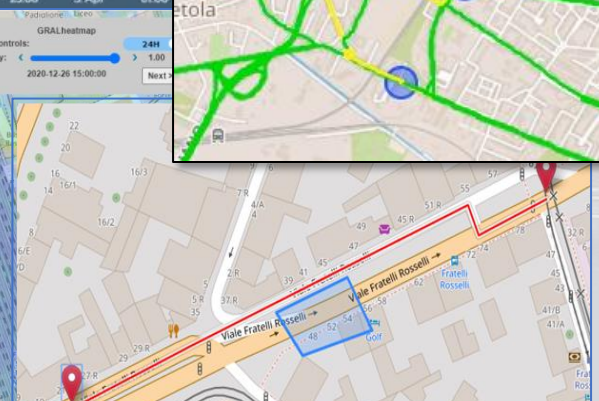
- POI, IOT Devices, shapes,..
  - FIWARE Smart Data Models,
  - IoT Device Models
- GIS, maps, orthomaps, WFS/WMS, GeoTiff, calibrated heatmaps, ..
- Satellite data, ..
- traffic flow, typical trends, ..
- trajectories, events, Workflow, ..
- 3D Models, BIM, Digital Twins, ..
- OD Matrices of several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











Standards and Interoperability (6/2023)

Compliant with:

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

https://www.snap4city.org/65

Parts 3, 5

What About Entity Instances / IoT Devices, Time Series

Entity / IOT Device

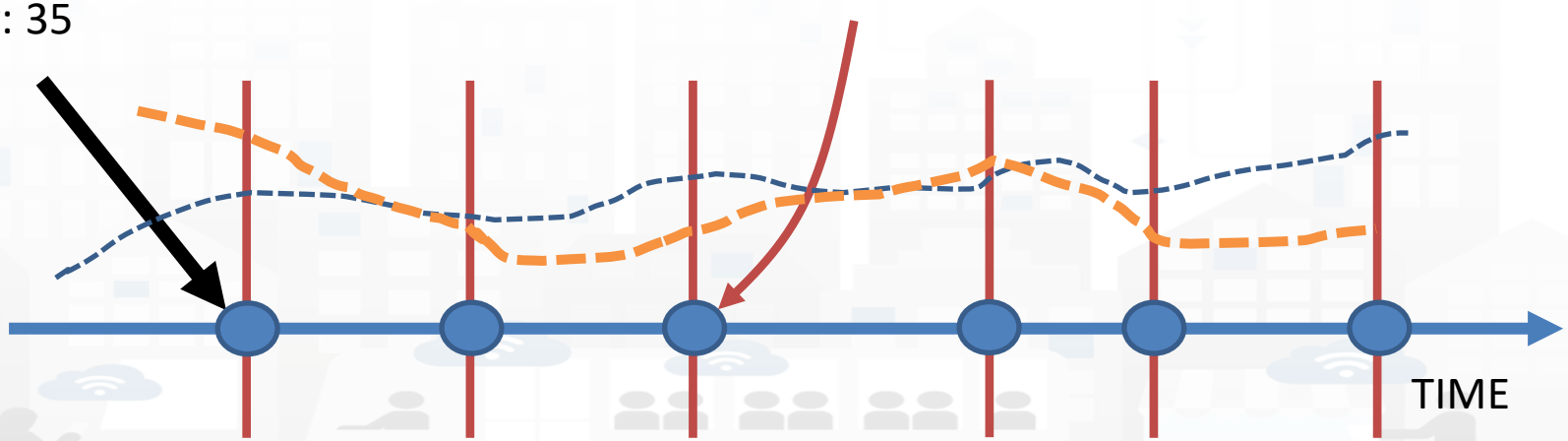
Entity: IOT Device

Sends a message

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

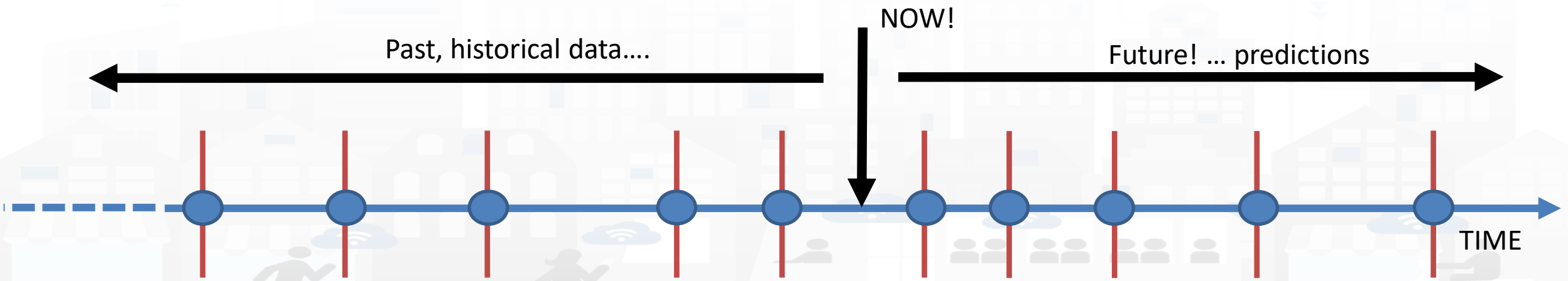
dateObserved	Temp	Humidity
02-04-2020 10:30	34.5	23
02-04-2020 10:40	36.5	24
02-04-2020 10:50	36.0	22.5

- 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



HLT: Unified Classification for Data and Services

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

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

High Level Types

Nature

SubNature

Semantic Classific.

Dev/Model name

Broker name

Technical Source

Value Name

Value Type

Data Type

Value Unit

Variables, names

Last Date/Time

Last Value

Real Time

Healthiness

Last Check

Status

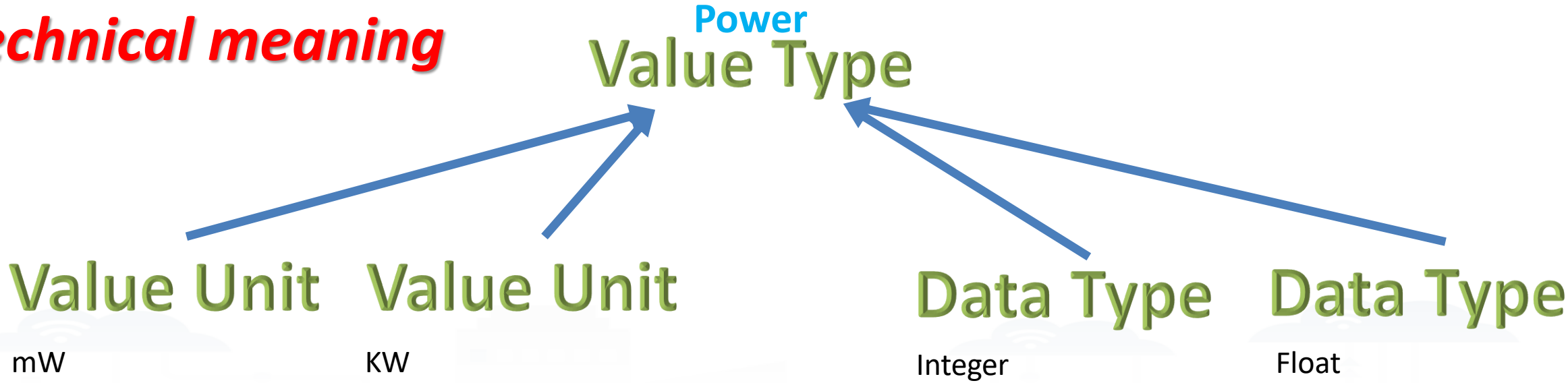
Ownership

Organization

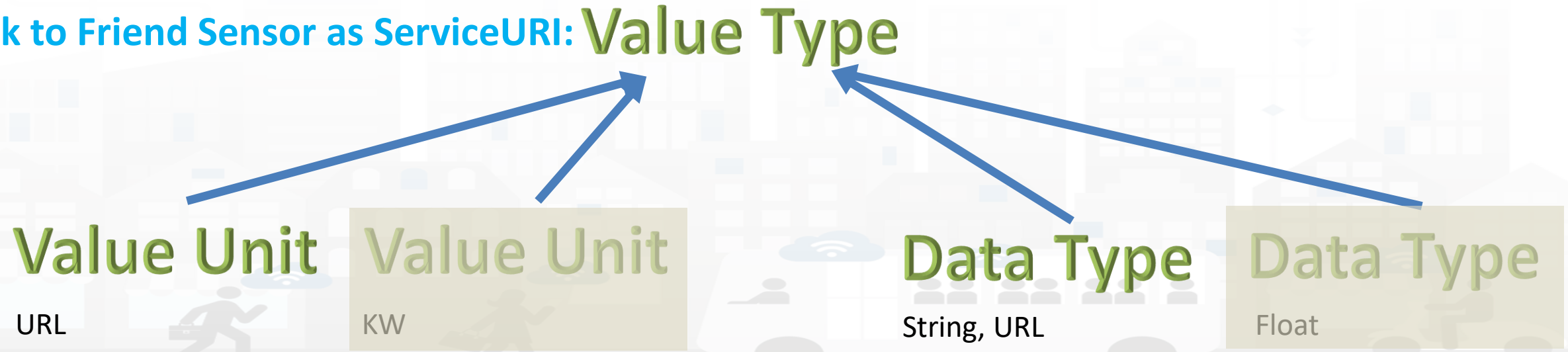
For Admin

For example

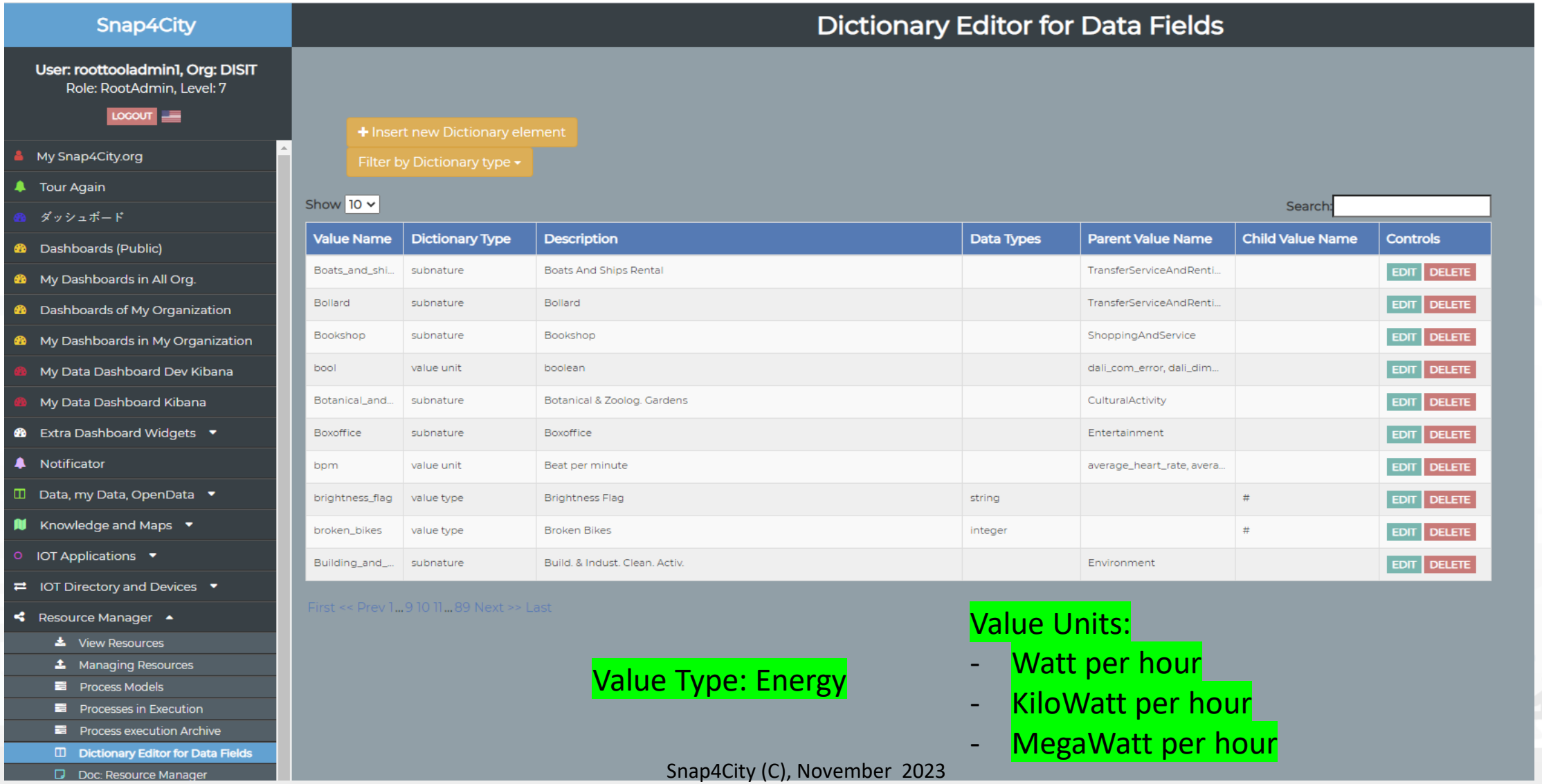
Technical meaning



Link to Friend Sensor as ServiceURI:



Example of Energy and its Value Units



Value Type: Energy

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

Please note on: Data Type

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

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**

Connections among Entities

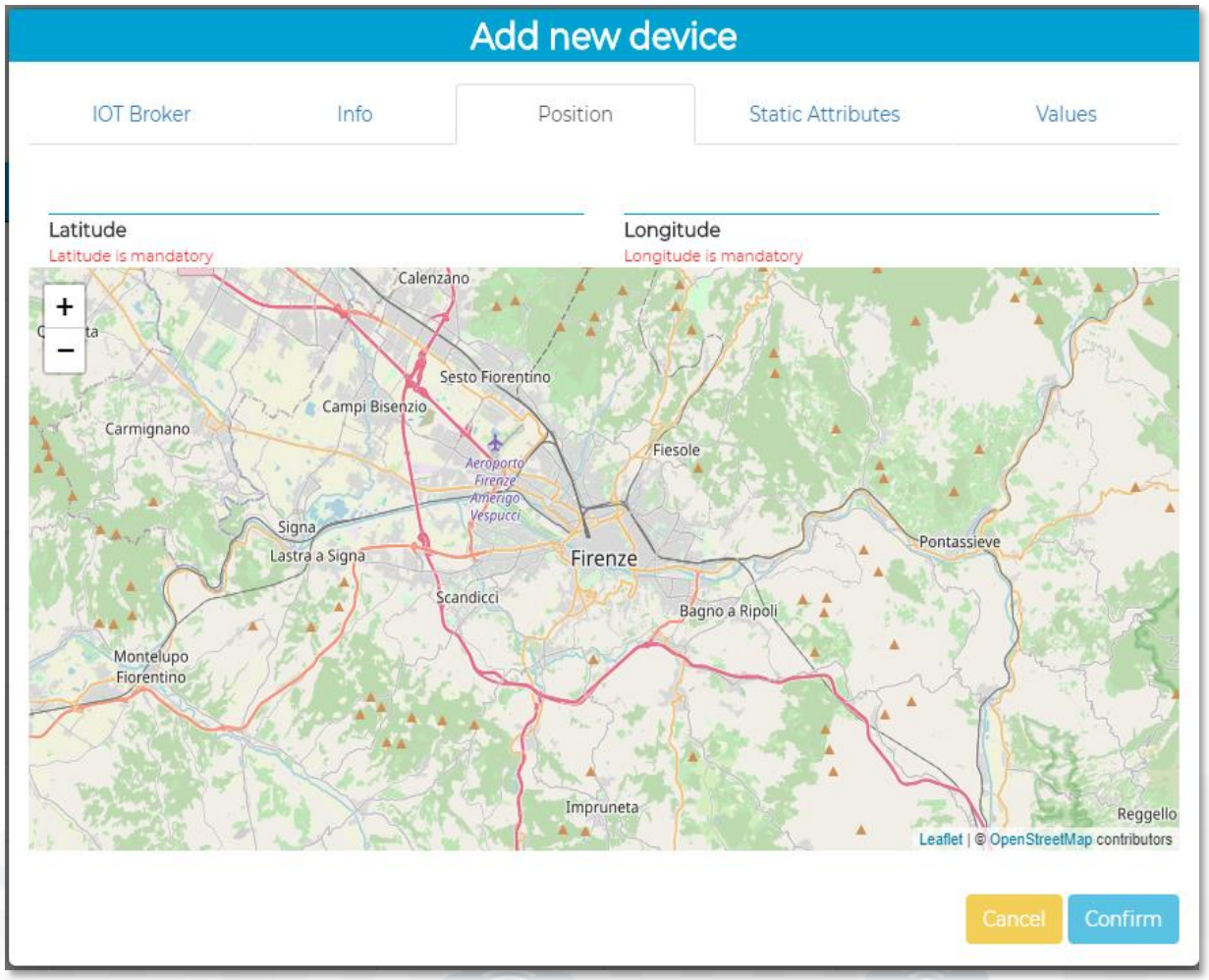
Where	Entity Model (IOT Device Model)	Entity Instance (IOT Device)	Entity Message at 23-12-2019T20:15:00	Entity Message at 23-12-2019T20:30:12
Broker	**Broker: OrionUNIFI**			
Broker	**Protocol: NGSI**			
Info	**ID: string**	ID: "park45"	**park45**	**park45**
Position	**GPS: lat, long**	GSP: 43.12, 11.34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

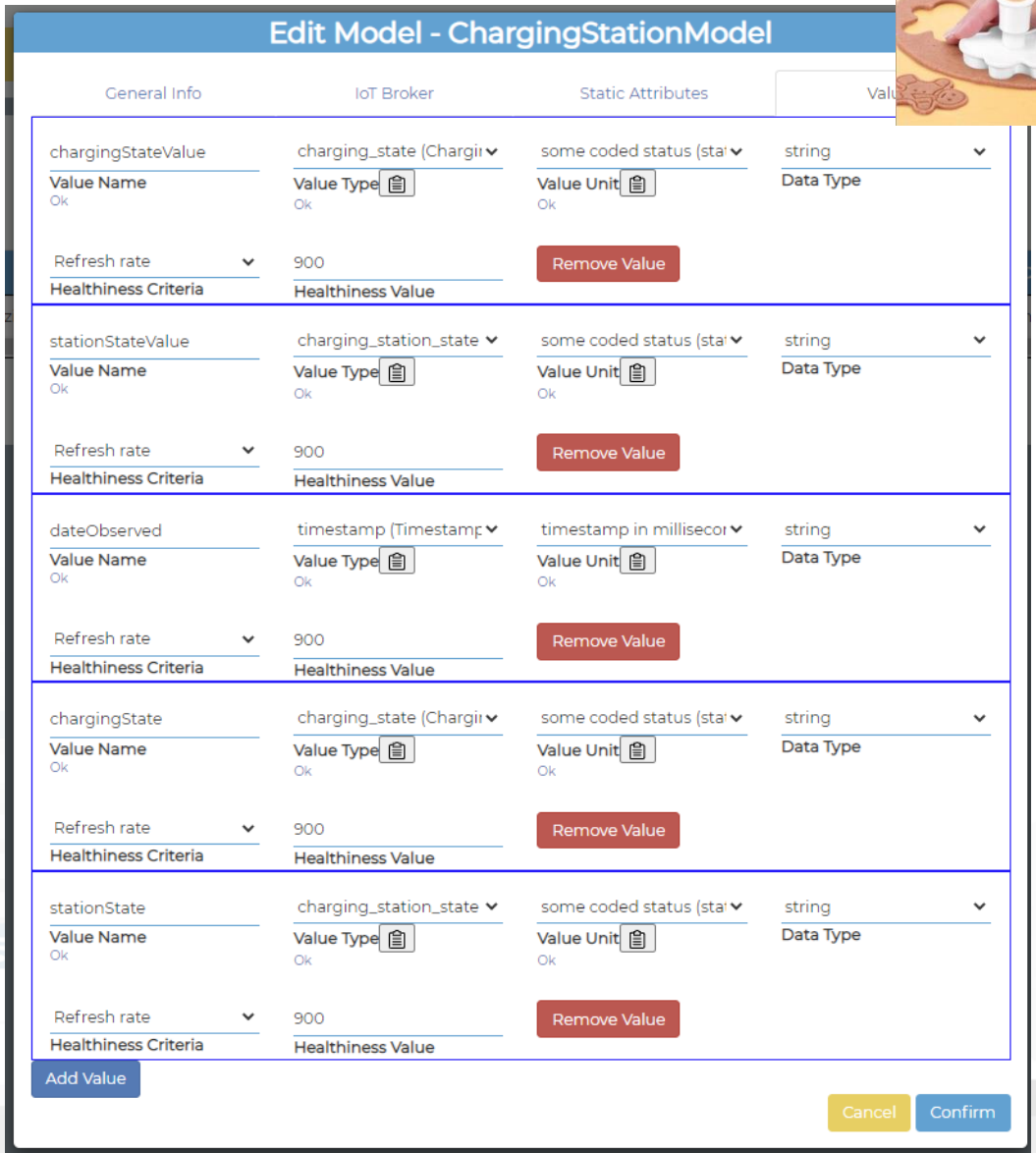
Model meaning

parts 3 and 5

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

Entity / Device Model (2)





SURI Connections

From a

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

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

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

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

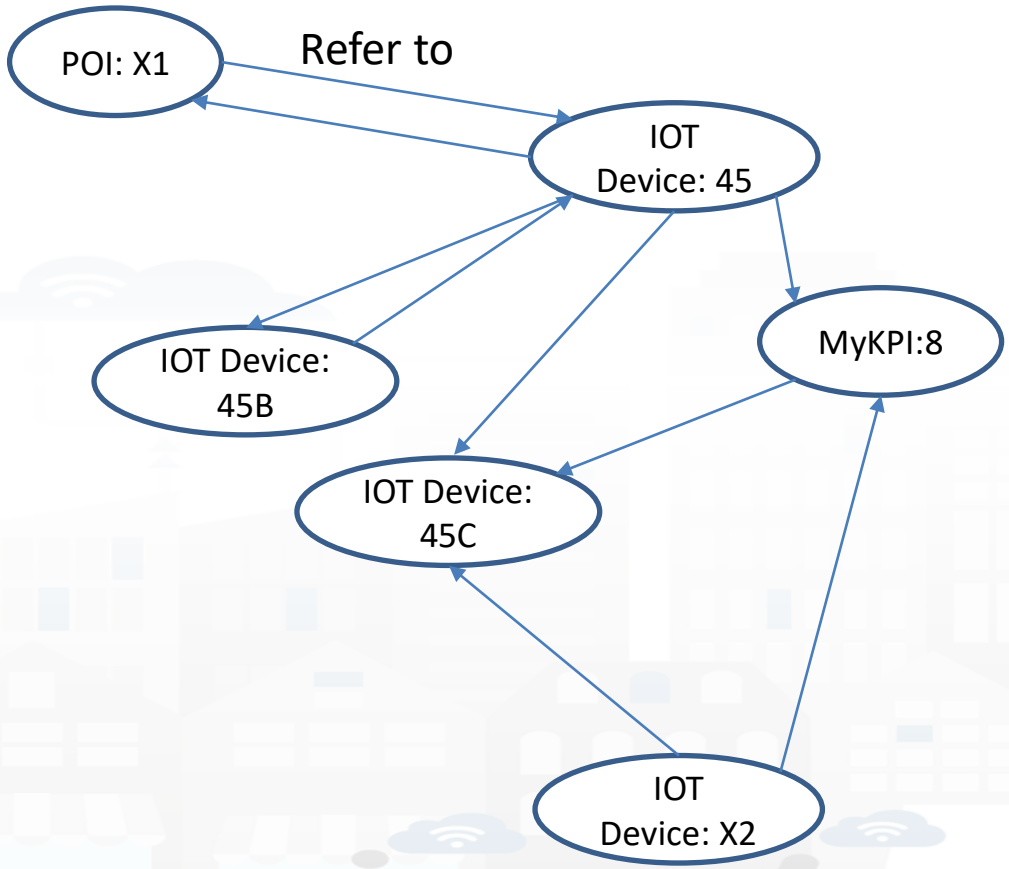
p	o
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/ns/sosa/Sensor
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.disit.org/km4city/schema#Traffic_sensor
http://www.w3.org/ns/ssn/implements	http://www.disit.org/km4city/resource/iot/traffic
http://www.disit.org/km4city/schema#hasAttribute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

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

- They are triples
- Subject-predicate-Object
  - Subject=SURI
  - Predicate=p
  - Object = o

References/Links to Entities Instances / IoT Devices

Relationships among Devices/Entities, POI and MyKPI



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

IoT device with References to other and MyKPI


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


Value Type: Identifier

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

Data Type: String

Data Modeling example1

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

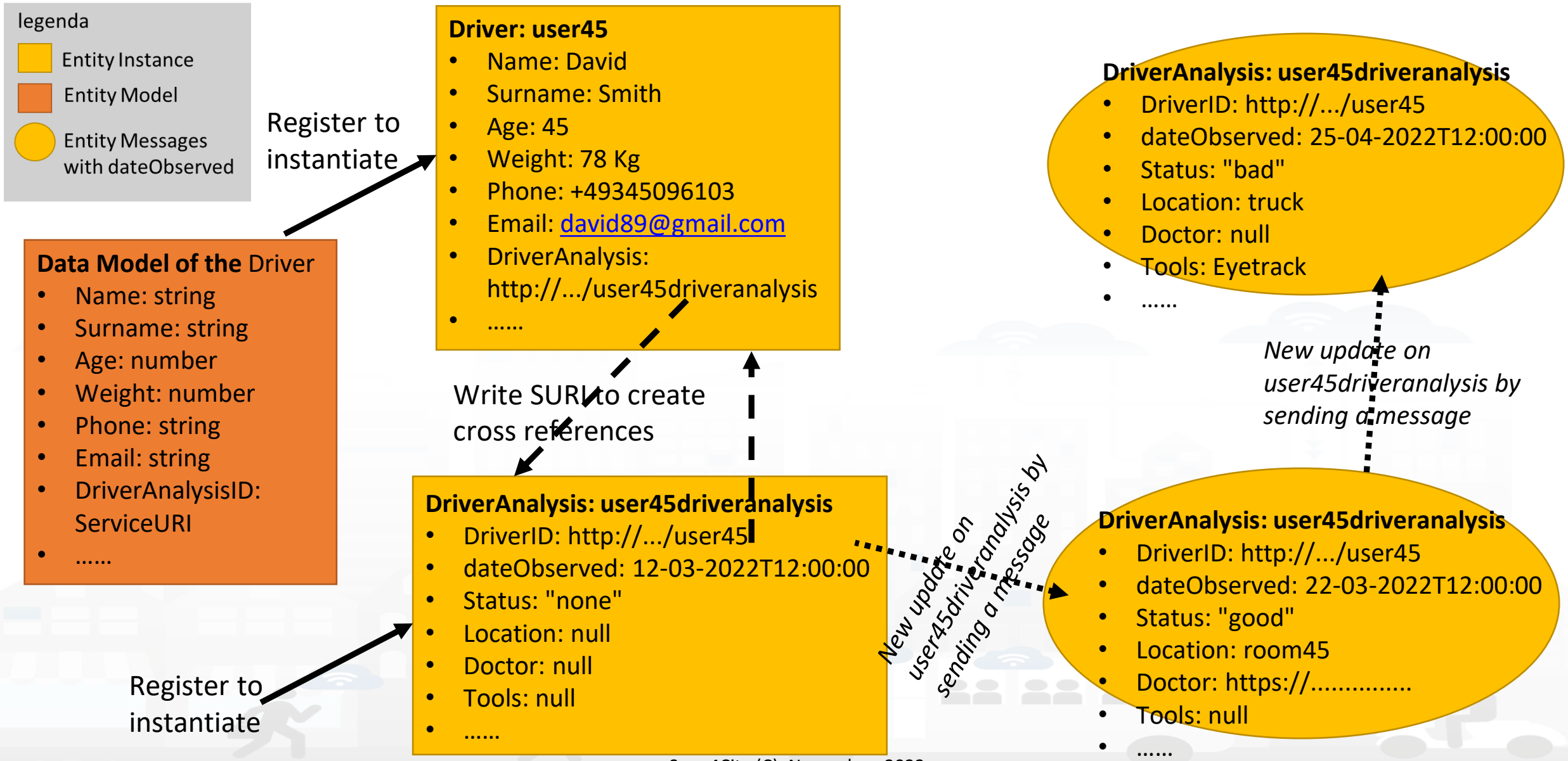
IoT Device Model: driverHelthiness Nature:…… Subnature: …… Lat,lon: …… Device in Mobility: ……			
Value_name	Value Type	Value Unit	Data Type
dateObserved	Timestamp	Timestamp in ms	String
kind			
levelAttentionFactor1			
levelAttentionFactor2			
driver	Identifier	ServiceURI	String

Example 3

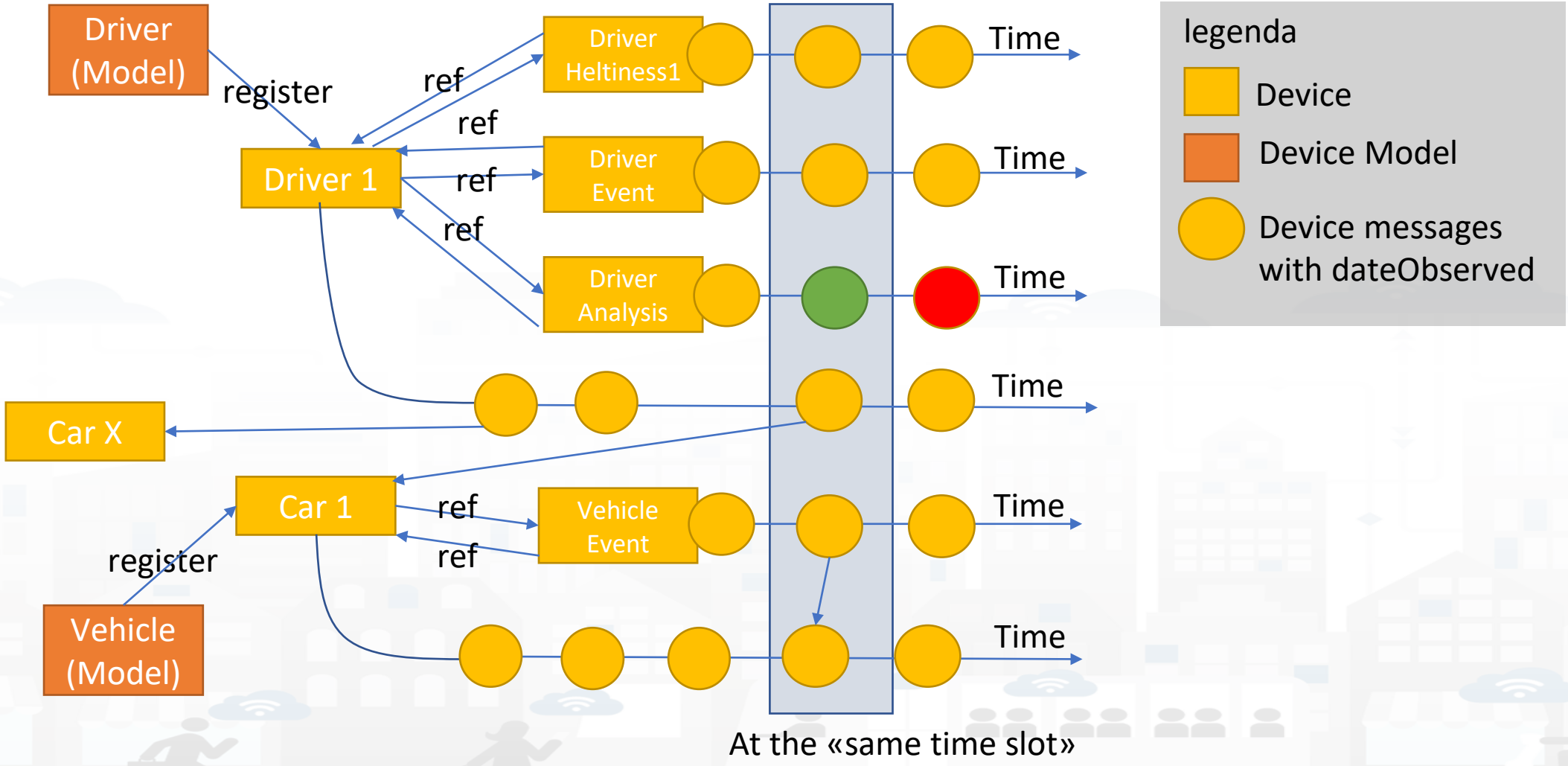
IoT Device Model: Vehicle Nature:…… Subnature: …… Lat,lon: …… Device in Mobility: ……			
Value_name	Value Type	Value Unit	Data Type
dateObserved	Timestamp	Timestamp in ms	String
producer	entity	text	String
model	entity	text	String
plate	entity	text	String
companyID	entity	text	String
velocity	velocity	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

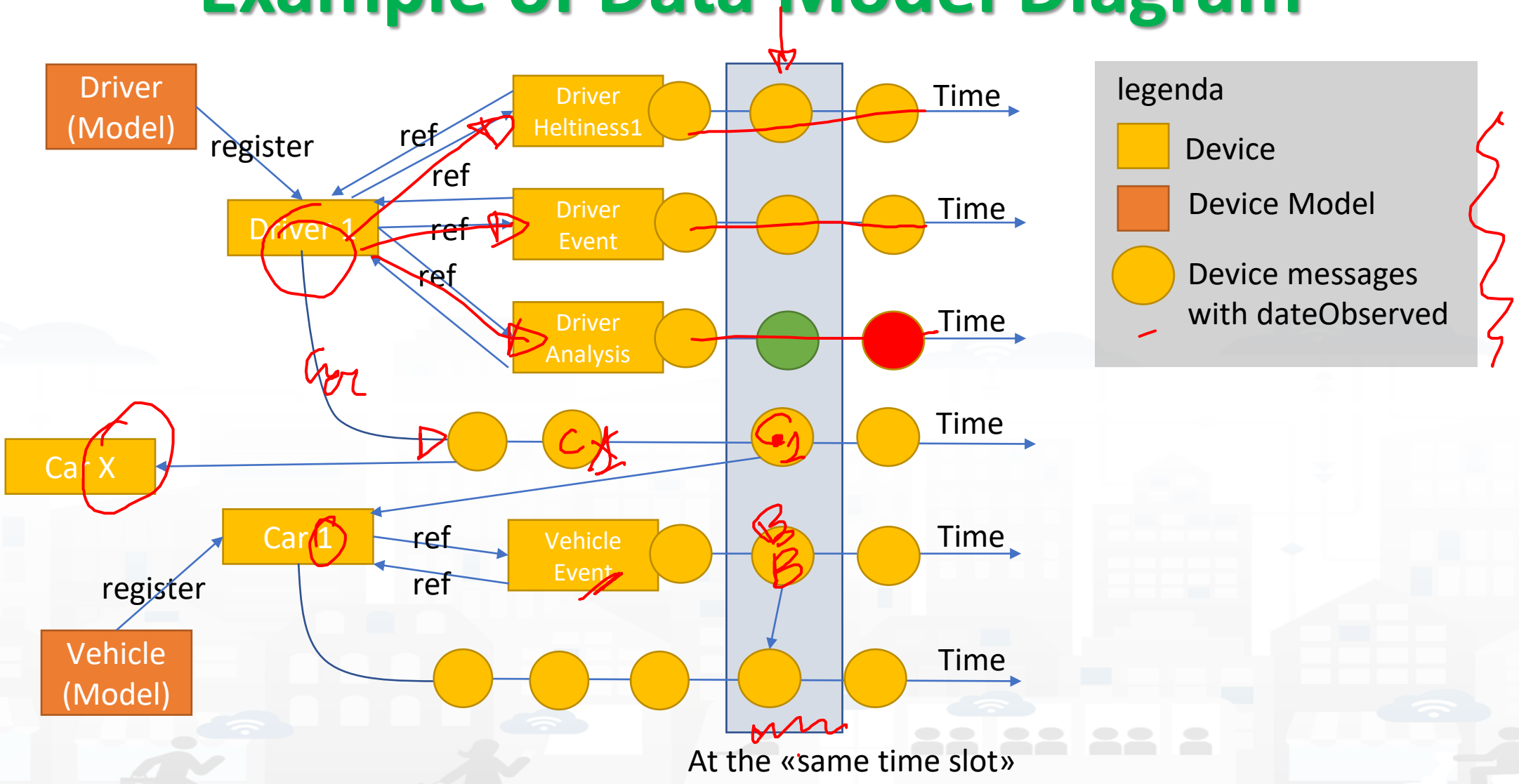
IoT Device Model: VehicleEvent Nature:…… Subnature: …… Lat,lon: …… Device in Mobility: ……			
Value_name	Value Type	Value Unit	Data Type
dateObserved	Timestamp	Timestamp in ms	String
eventID	ID	text	String
eventKind	status	some coded status	String
status	status	some coded status	String
vehicle	Identifier	ServiceURI	String

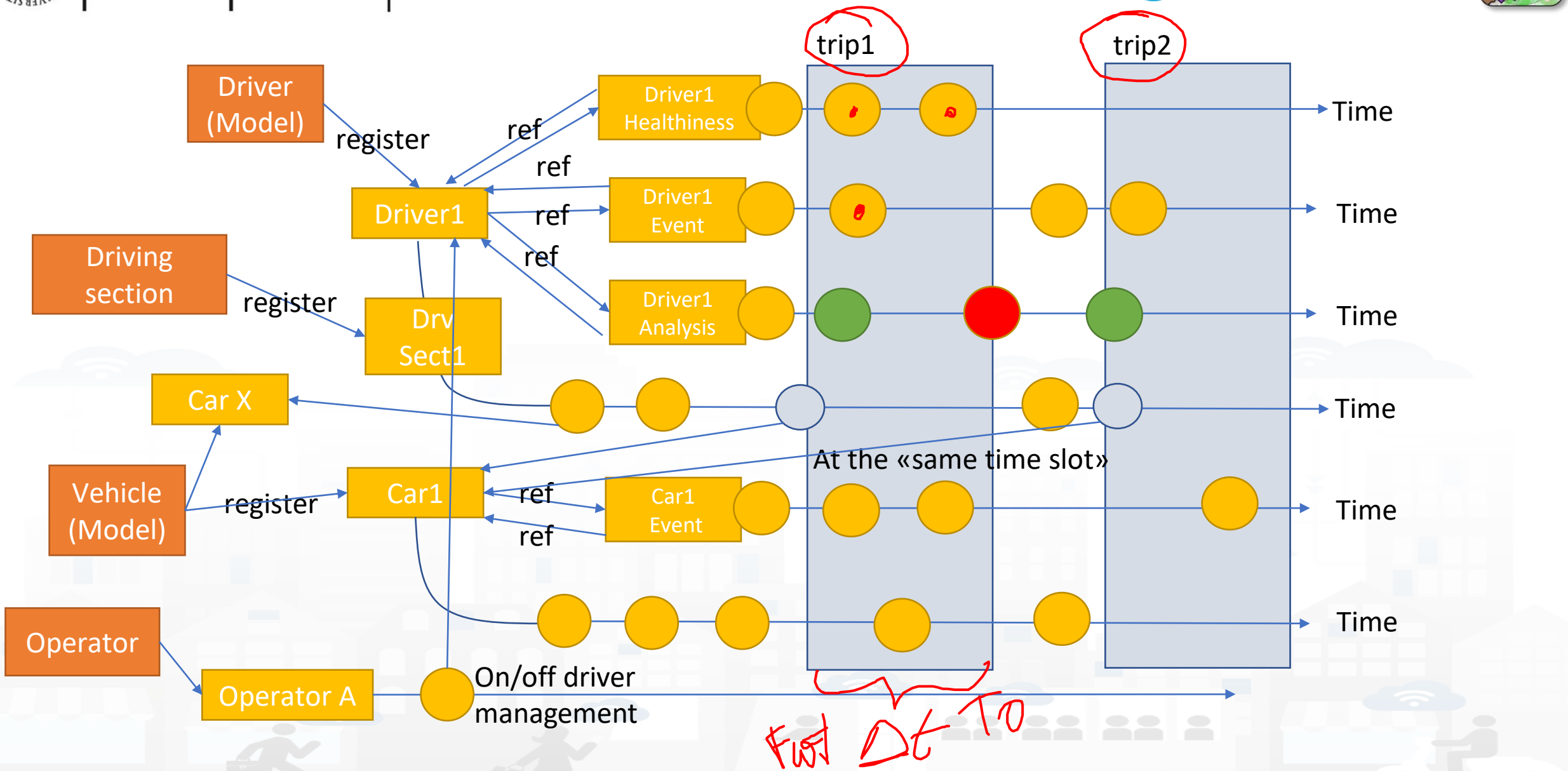

legenda
Entity Instance
Entity Model
Entity Messages with dateObserved
Register to instantiate
Data Model of the Driver
• Name: string
• Surname: string
• Age: number
• Weight: number
• Phone: string
• Email: string
• DriverAnalysisID: ServiceURI
• ……
Driver: user45
• Name: David
• Surname: Smith
• Age: 45
• Weight: 78 Kg
• Phone: +49345096103
• Email: david89@gmail.com
• DriverAnalysis: http://.../user45driveranalysis
• ……
Write SURI to create cross references
DriverAnalysis: user45driveranalysis
• DriverID: http://.../user45
• dateObserved: 12-03-2022T12:00:00
• Status: "none"
• Location: null
• Doctor: null
• Tools: null
• ……
Register to instantiate
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



Example of Data Model Diagram



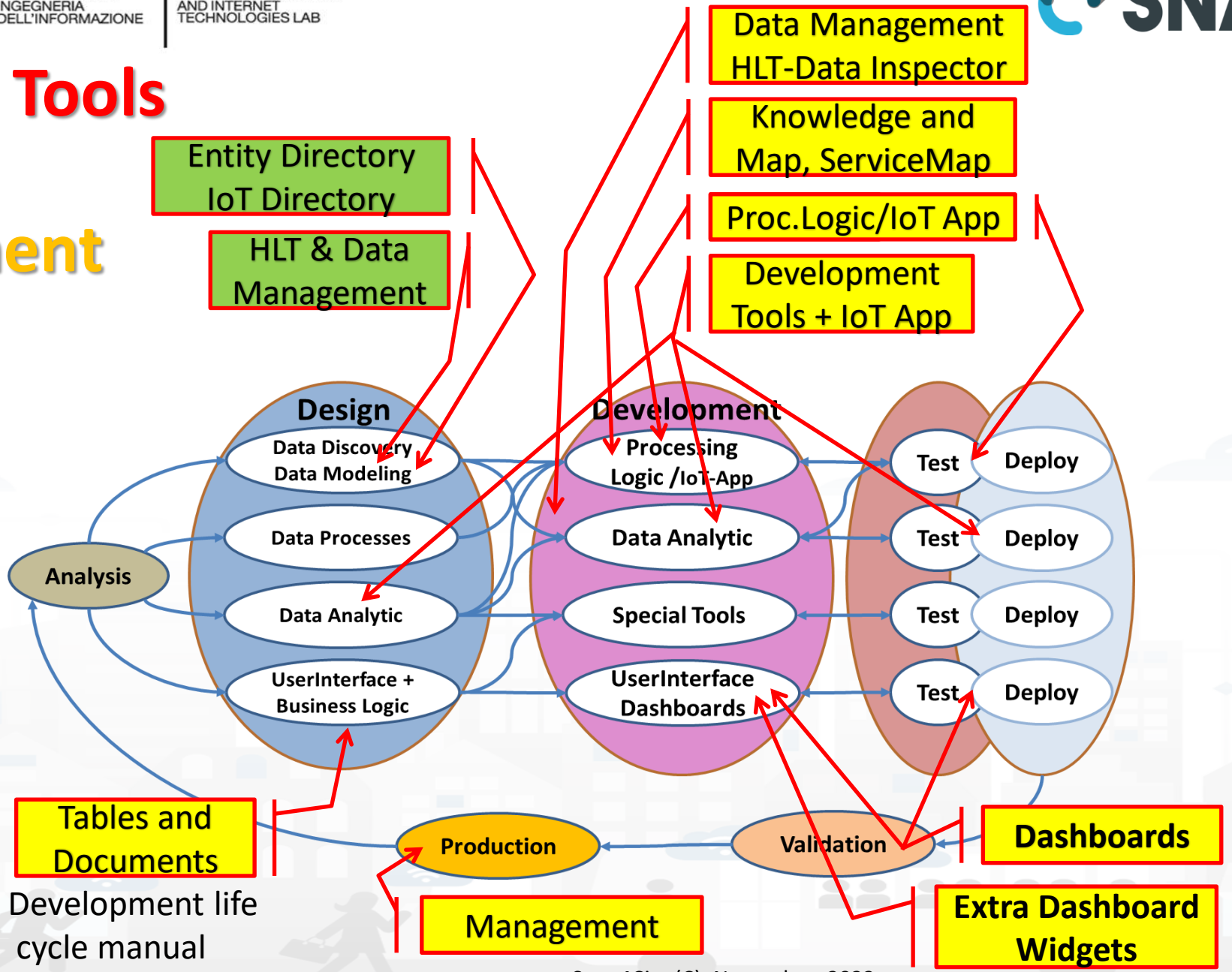

trip1
trip2
Driver (Model)
register
ref
Driver1 Healthiness
ref
Driver1
ref
Driver1 Event
ref
Driver1 Analysis
Driving section
register
Drv Sect1
Car X
Vehicle (Model)
register
Car1
ref
Car1 Event
ref
Operator
Operator A
On/off driver management
Time
Time
Time
Time
Time
Time
At the «same time slot»
Fwd Dt To

TOOLS for Data Design and HLT Exploitation

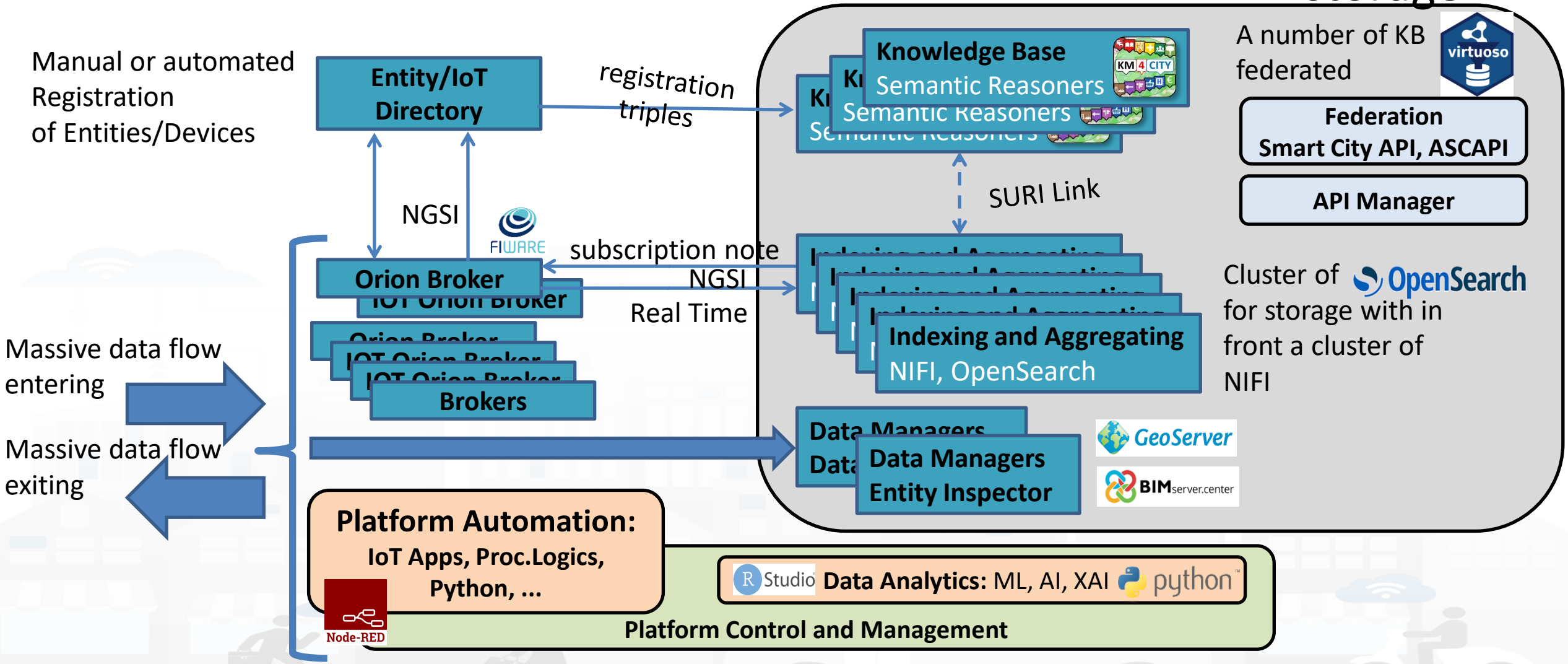
Part 2

Part 5

Snap4City Tools vs Development Life Cycle



Snap4city Data Ingestion Diagram



Any Entity has a Semantic *Classification*

Nature

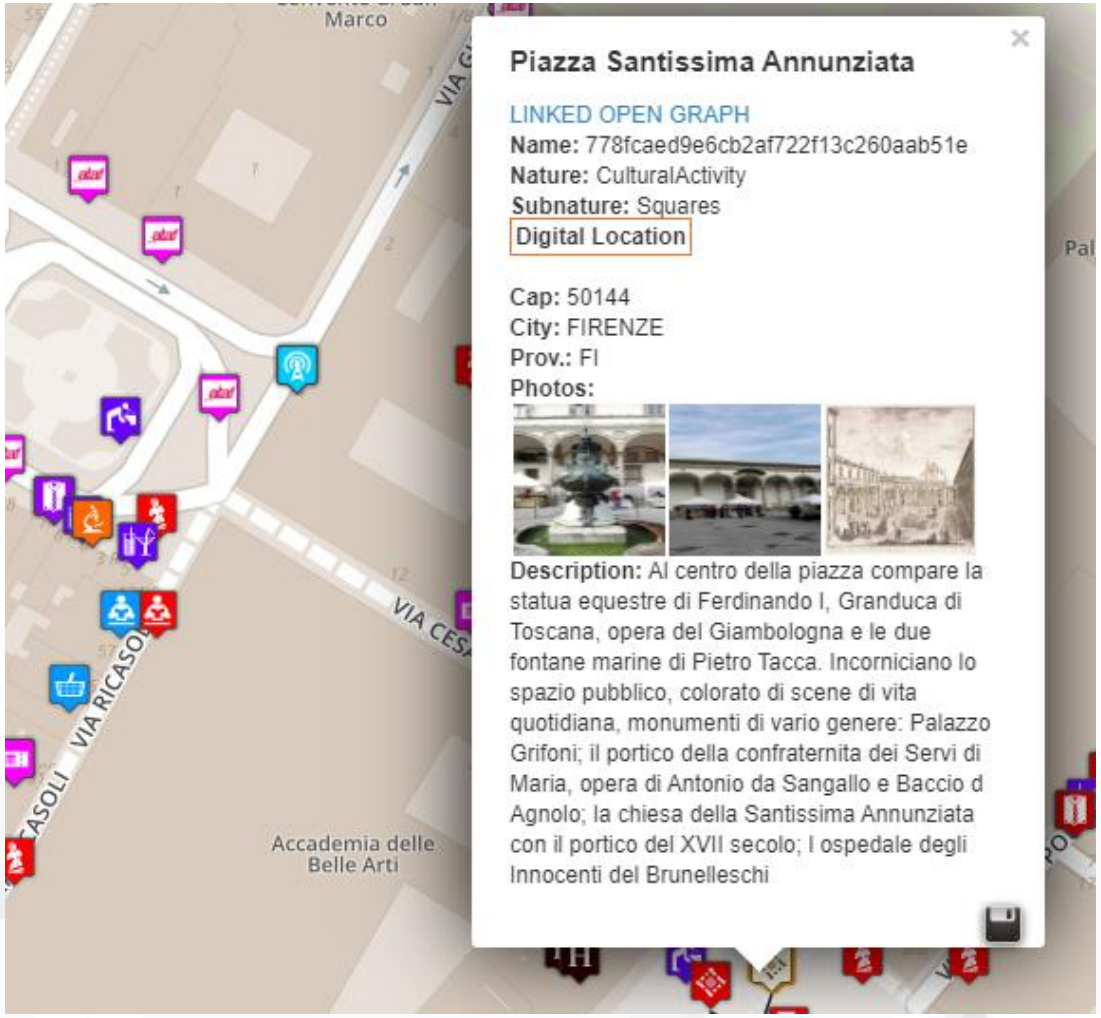
- Accommodation +
- Advertising +
- AgricultureAndLivestock +
- CivilAndEdilEngineering +
- CulturalActivity +
- EducationAndResearch +
- Emergency +
- Entertainment +
- Environment +
- FinancialService +
- GovernmentOffice +
- HealthCare +
- IndustryAndManufacturing +
- IoTDevice +
- MiningAndQuarrying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

SubNature

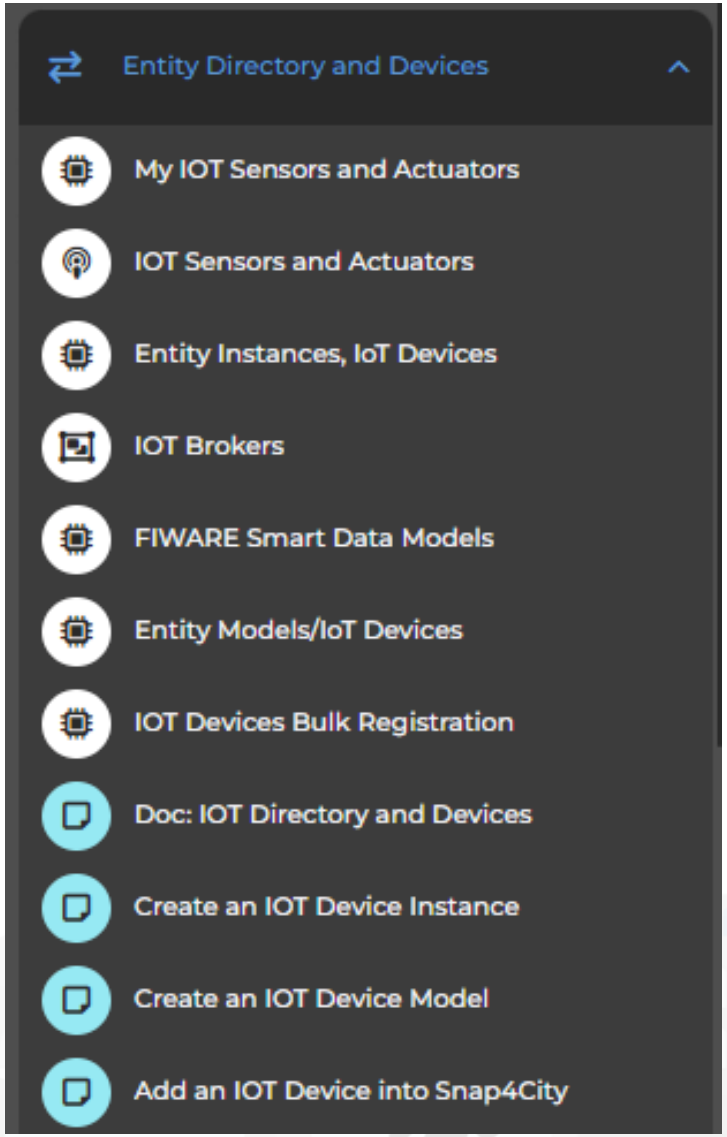
EducationAndResearch -

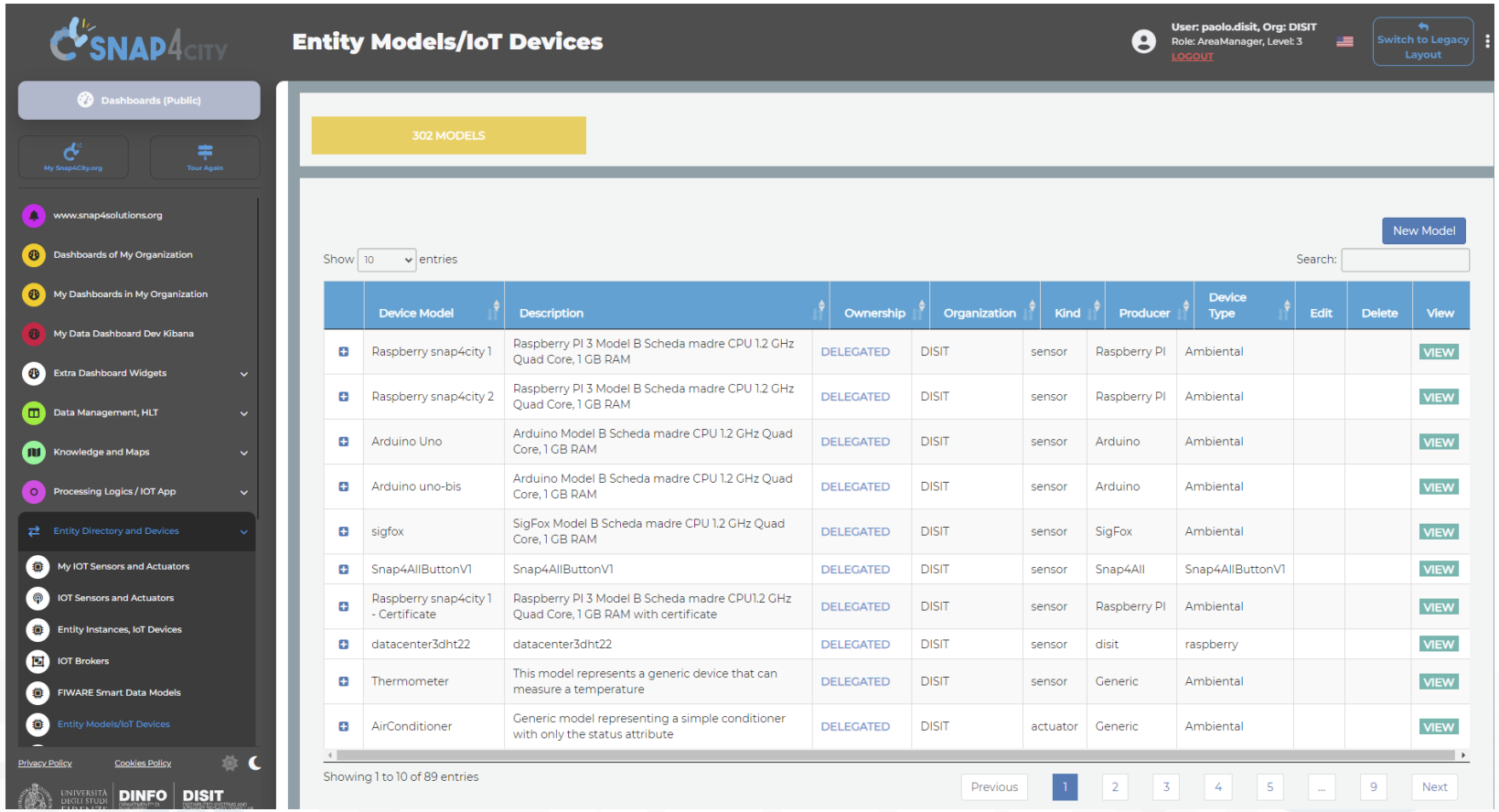
- Educational_support_activities
- Higher_education
- Language_courses
- Performing_arts_schools
- Post_secondary_education
- Pre_primary_education
- Primary_education
- Private_high_school
- Private_infant_school
- Private_junior_high_school

SKOS



Entity / Iot Directory: User Role





Entity Directory for Beginners

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

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

https://www.snap4city.org/818

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

View | Edit | Track | Access control | Convert

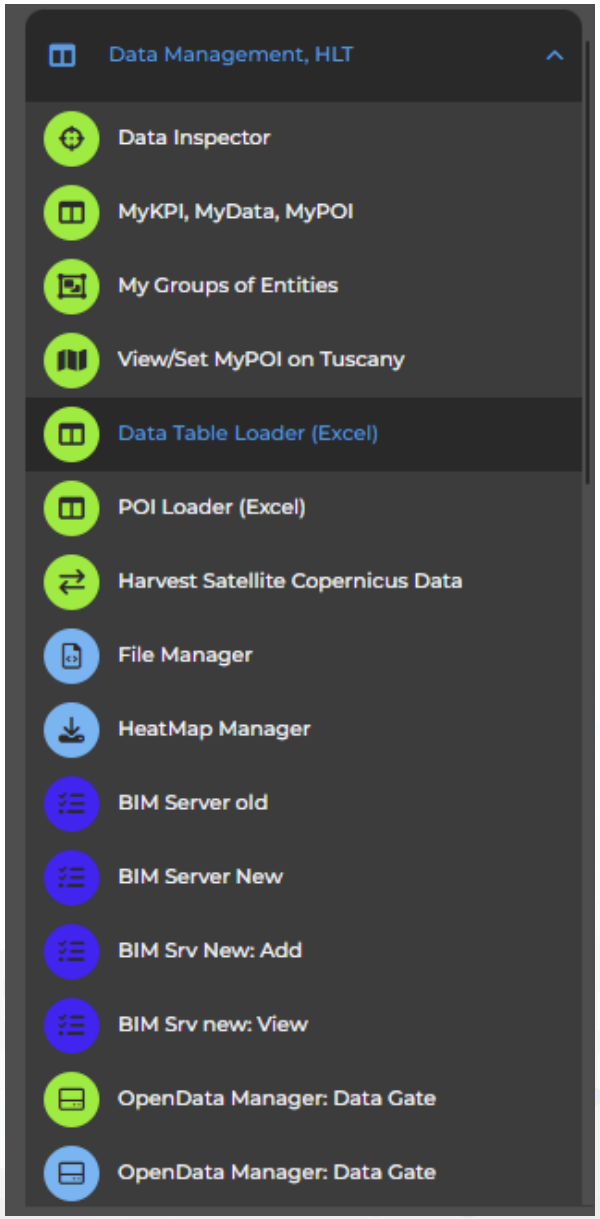
Any update and addtion to the dictionary of snap4city.org has to be requested to snap4city@disit.org

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

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

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

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



Data management, HLT

- Data Inspector
- MyKPI…..
- My Groups of Entities
- HeatMap Manager
- BIM Server…….
- Open Data…
- For user kind of users, other Managers:
  - ODM, File, TV CAM, Traffic Flow, …..

TOP

Design & Develop: Data Processes

Proc.Logic / IoT App

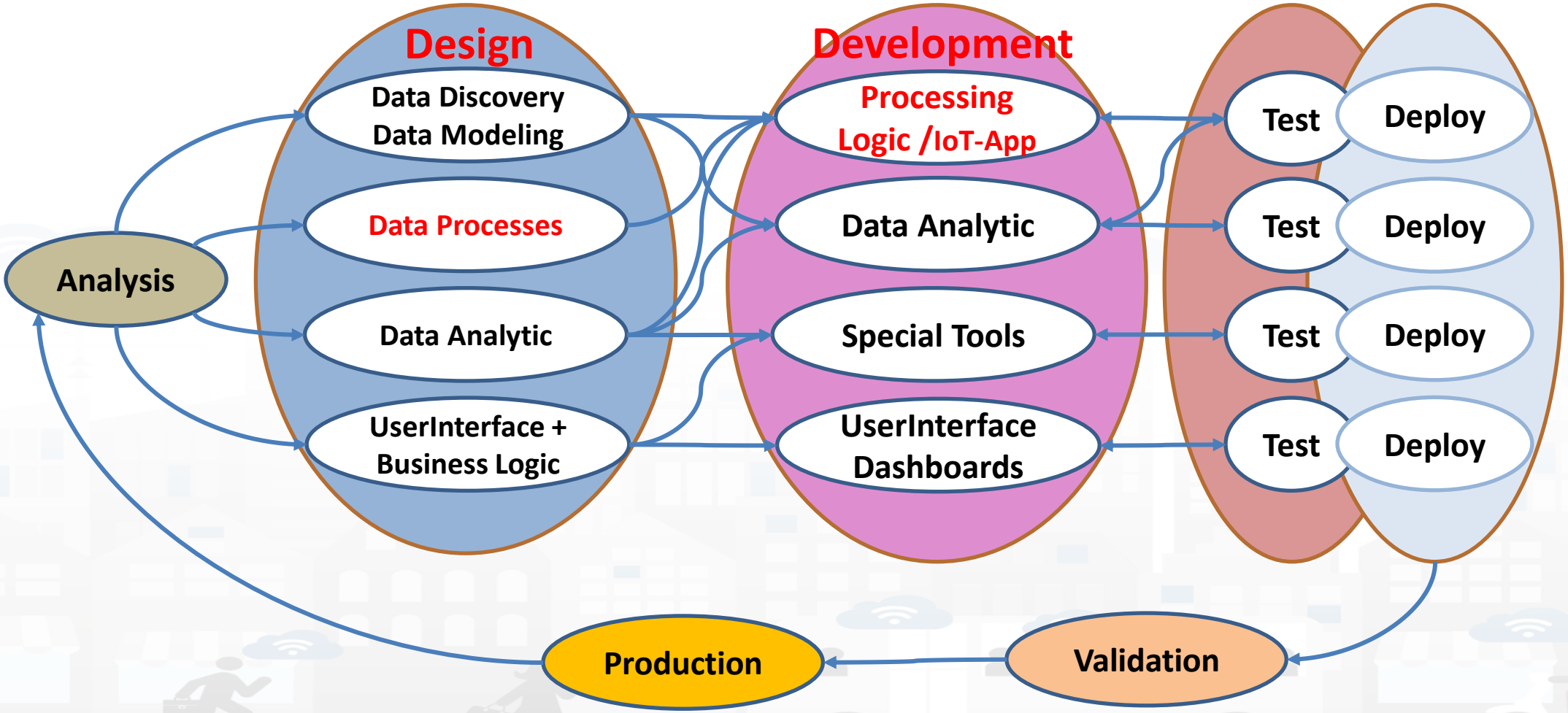
Part 3

Part 5

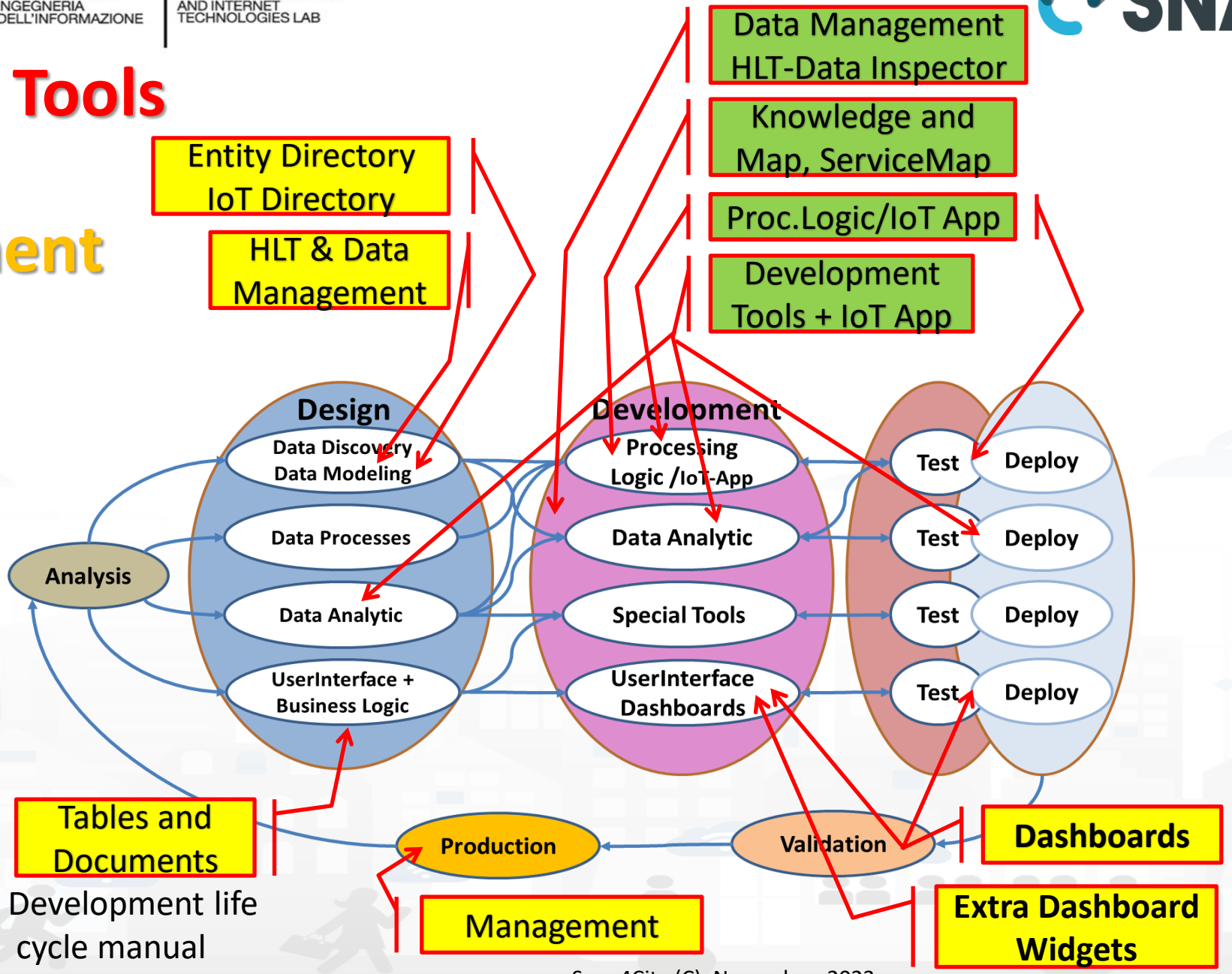
Activities for IoT App data processes

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

Development Life Cycle Smart Solutions



Snap4City Tools vs Development Life Cycle



Development life cycle manual

Design: Data Processes

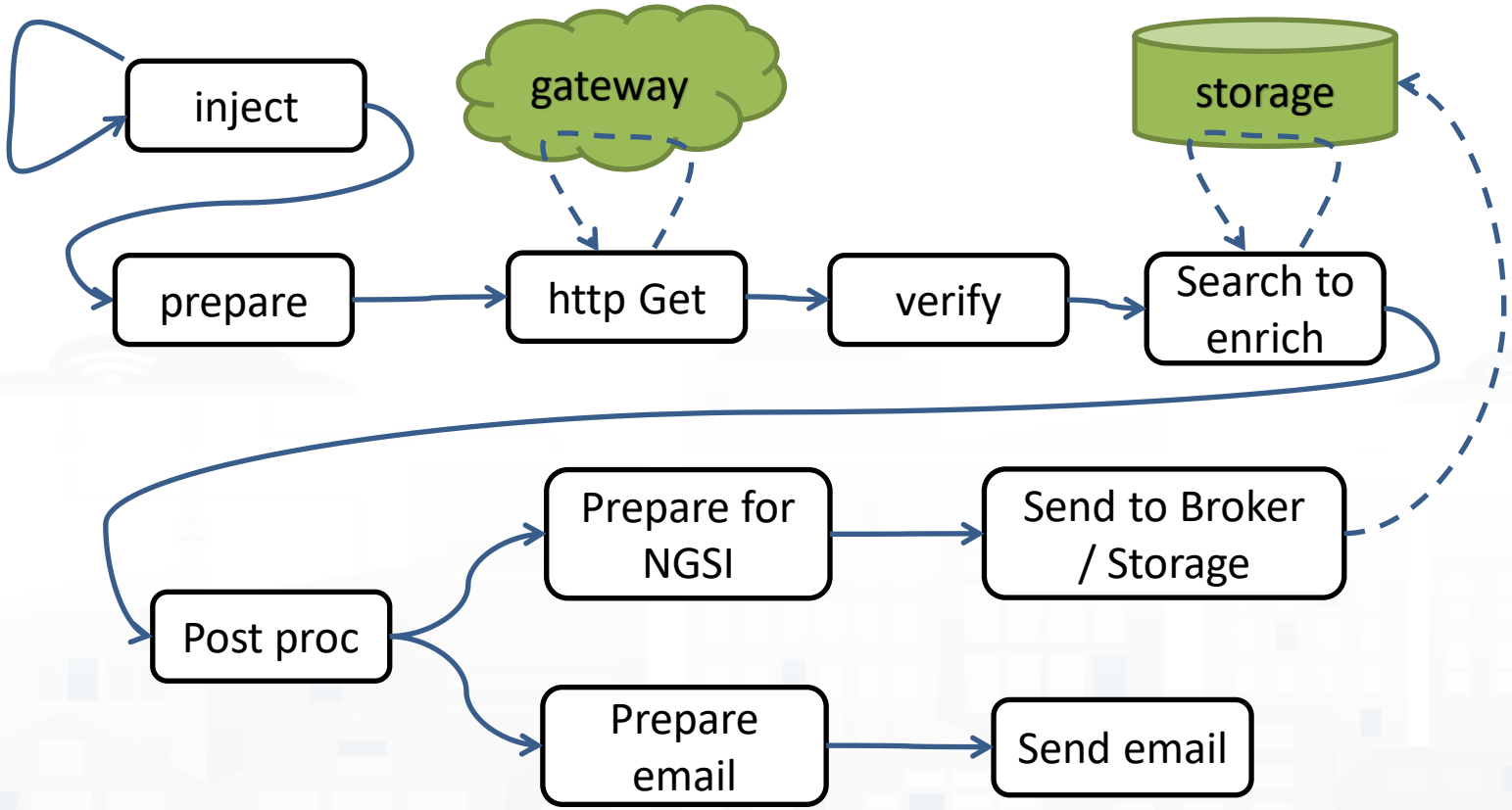
Part 3

Part 5

How to Design

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

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

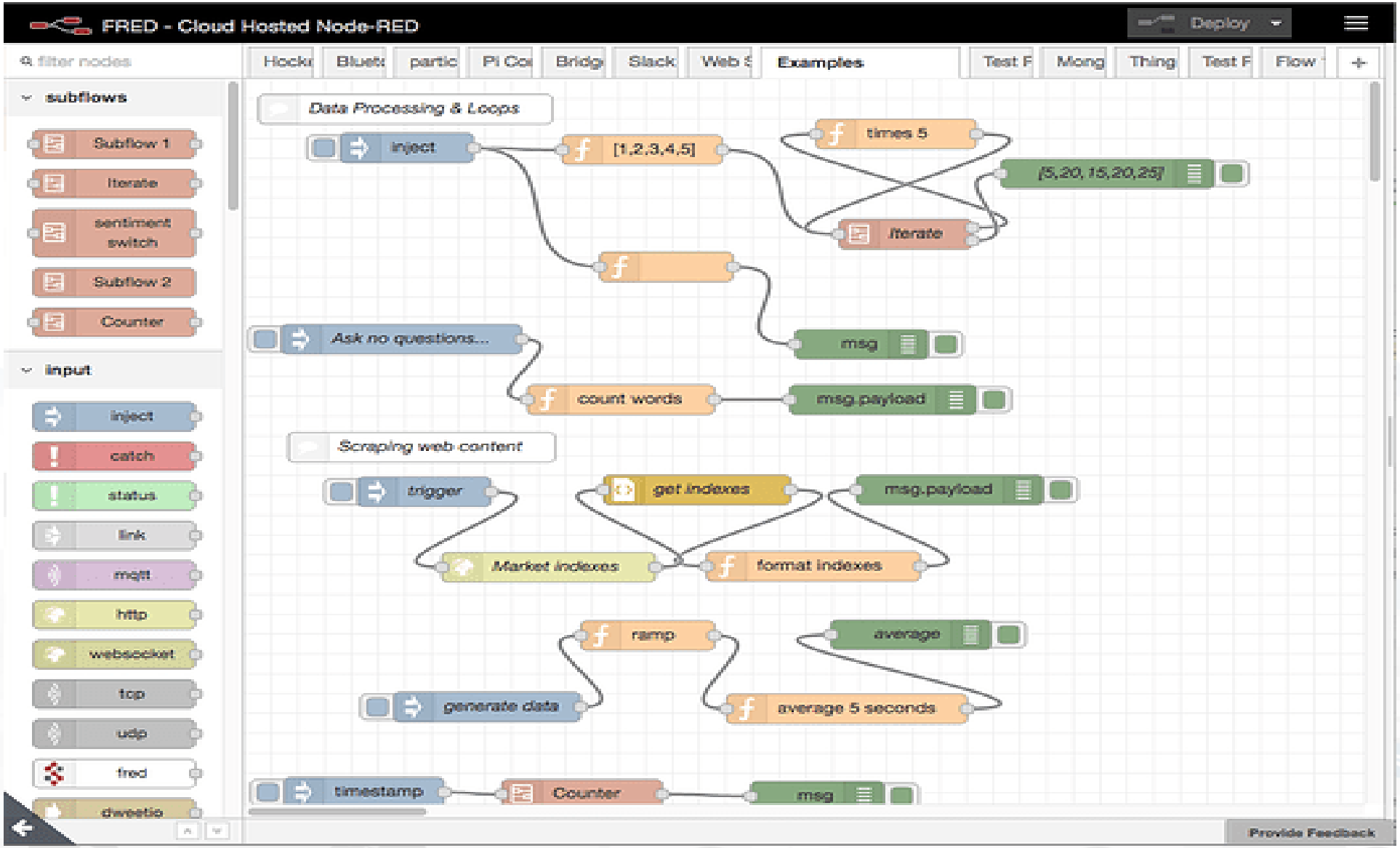


a. Periodically activate the flow
b. Call a gateway to get data
c. Verify the correctness of data
d. Enrich the data with other information coming from Cloud data into the storage
e. Transform the data in the correct forma
f. Send the data into the IoT Broker, and thus send the data in the storage on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

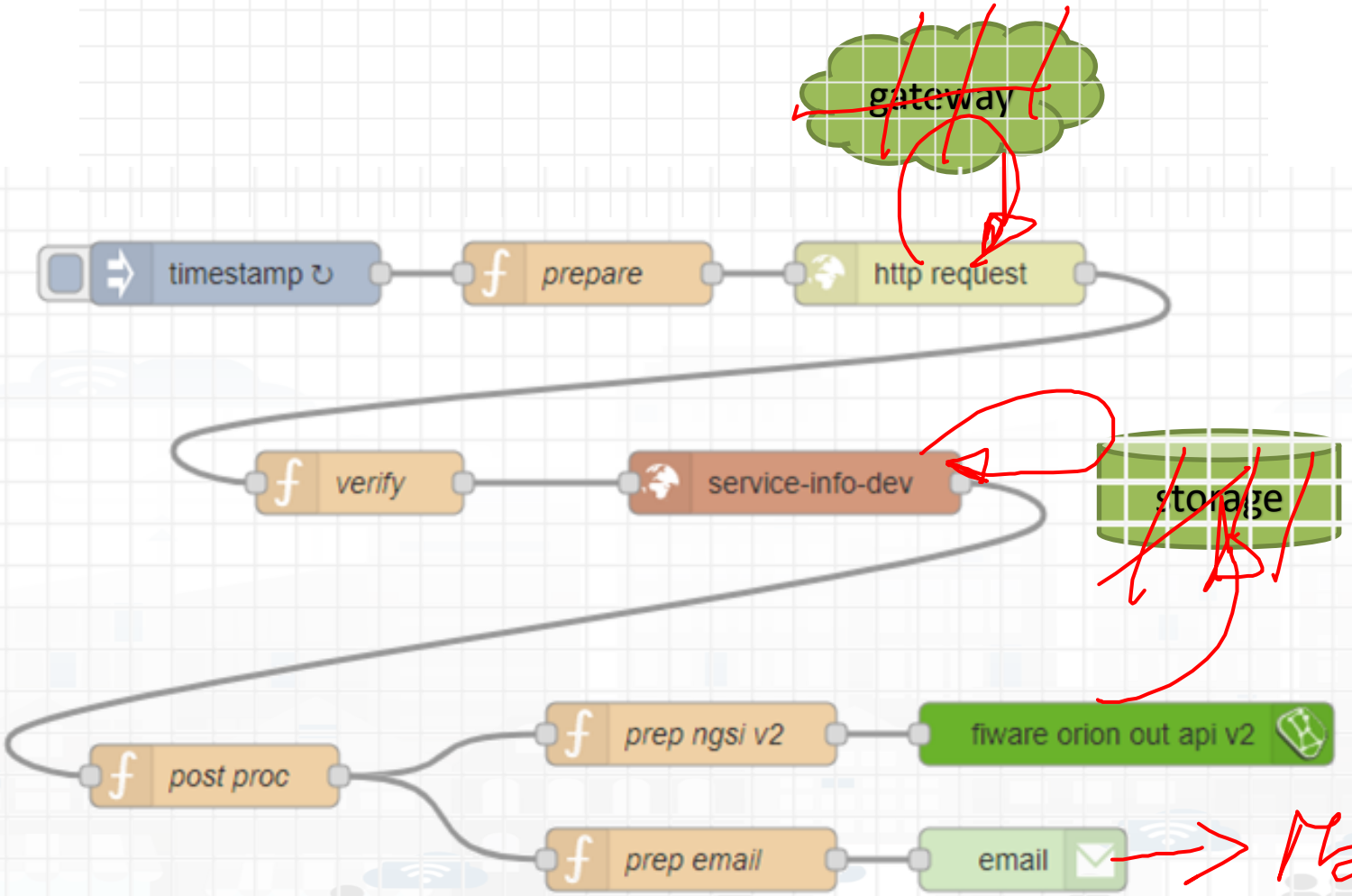
Develop: Data Processes

Part 3

Part 5


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

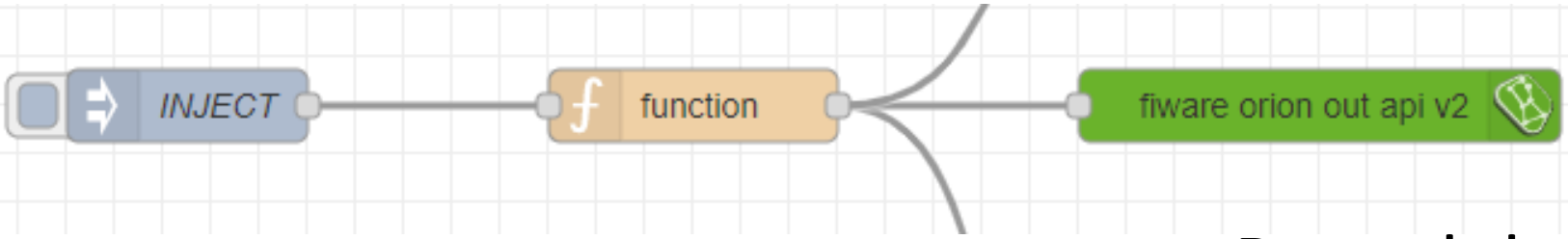
Proc.Logic (IoT App) Design, for each Data/Event Flow



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

Implicit services are not drawn

A sample of Data Ingestion

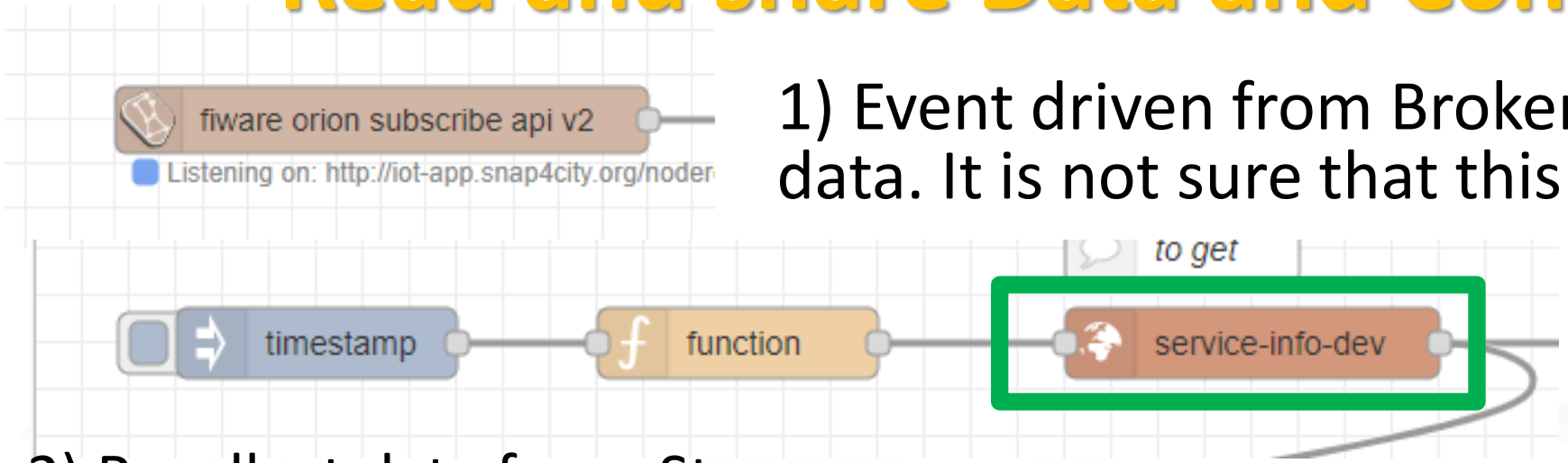


Function, example of NGSI V2 payload:

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

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

Read and share Data and Context Data



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

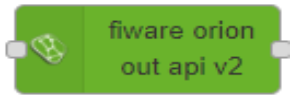
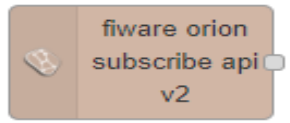
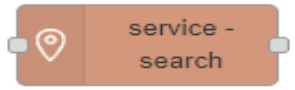
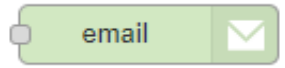
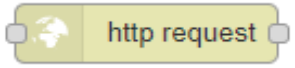
2) Recollect data from Storage
   - This node uses the Smart City API
- Any External Application can get the same data in authenticated authorized manner via Smart City API
- Smart City API is a better approach instead of producing a file outside or providing data via some local API service created from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

- 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.

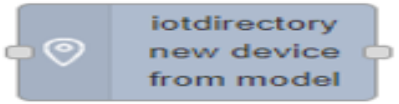
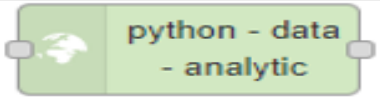
- YOU DO NOT HAVE TO ACCESS AT THE API all is provided in terms of NODEs/BLOCKS into IoT APP. Everything can be parametrized via JSON passed in input to the nodes.

- Most of the nodes can be also configured once from their user settings but the JSON is primary mode for setting parameters.

examples

Node shape	Description	Snap4City or standard
	To generate injection messages into a flow, scheduled or on manual demand by click it on left.	standard
	A java script function, from a JSON input to one or more JSON outputs, which can be produced by setting it.	standard
	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
	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
	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
	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
	Send email. With other nodes you can send Telegram, SMS, etc.	standard
	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

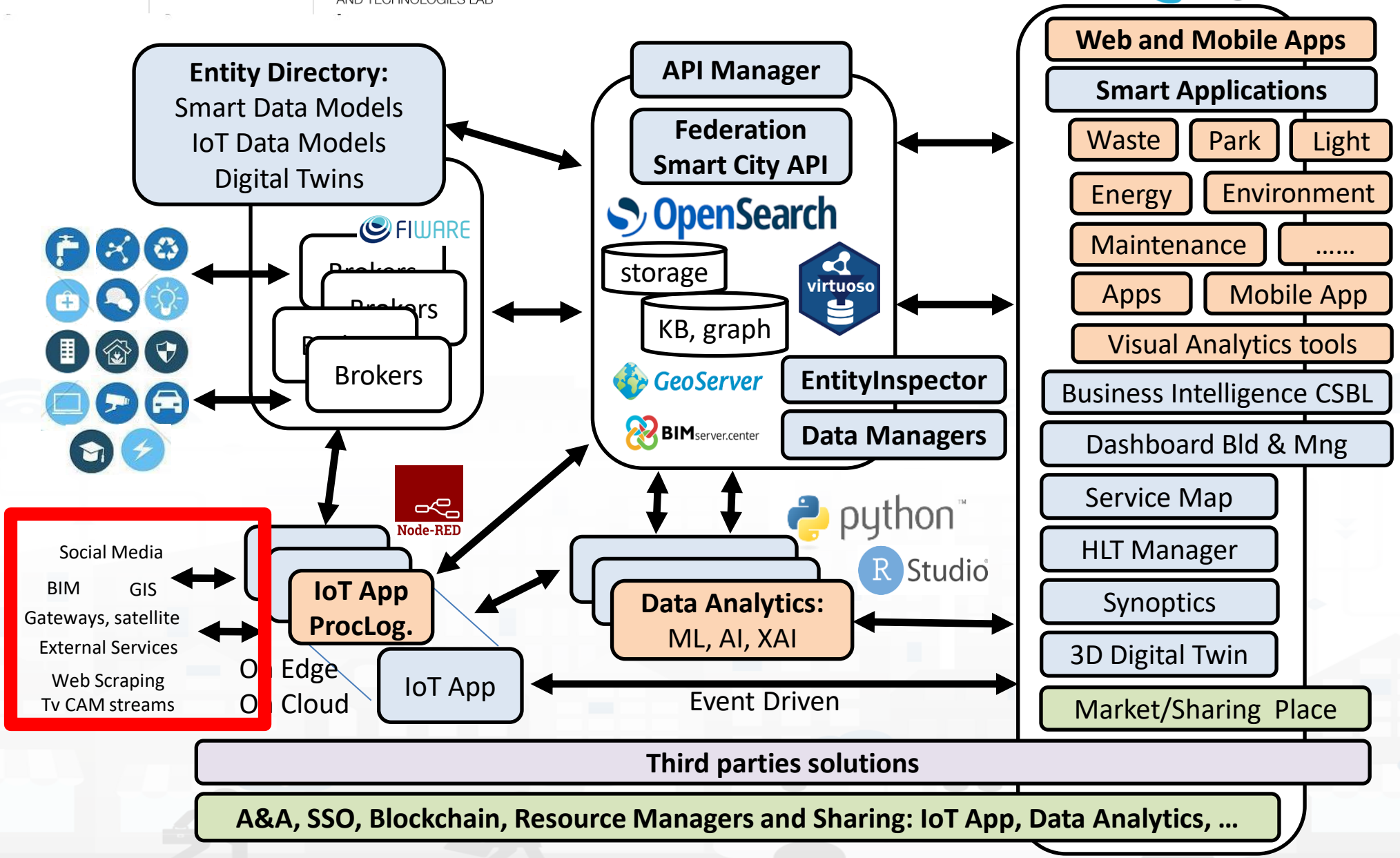
	A block which is printing on **debug** view the data JSON passed in its input. Please note that the node can be tuned to provide only msg.payload or the full JSON message, change configuration of the node.	standard
	To **create an Entity Instance** (device instance) from a model prepared on Entity Directory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

The Proc.Logic (IoT App) microservices

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

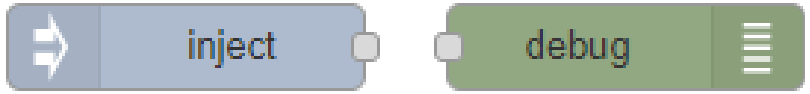
- **Data ingestion**: more than 100 protocols IOT and Industry 4.0, web Scraping, external services, any protocol database, etc.
- **Data access**: save/retrieve data, query search on expert system, georeverse solution, search on expert system Km4City ontology, call to Smart City API, etc.
- **Data Transformation/transcoding:**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

Tech Arch


Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIMserver.center
Data Managers
Node-RED
python
RStudio
Social Media
BIM
GIS
Gateways, satellite
External Services
Web Scraping
Tv CAM streams
IoT App
ProcLog.
Edge
Cloud
IoT App
Data Analytics:
ML, AI, XAI
Event Driven
Web and Mobile Apps
Smart Applications
Waste
Park
Light
Energy
Environment
Maintenance
……
Apps
Mobile App
Visual Analytics tools
Business Intelligence CSBL
Dashboard Bld & Mng
Service Map
HLT Manager
Synoptics
3D Digital Twin
Market/Sharing Place
Third parties solutions
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, …

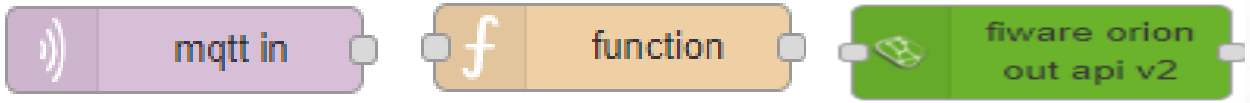
Some patterns

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



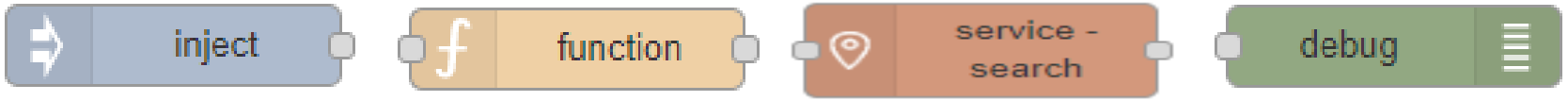
2) Hello world of node-red at two steps, the inject provides a push while a JSON is created into the function as *msg.payload = {……}* and sent/shown to/by the debug.

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

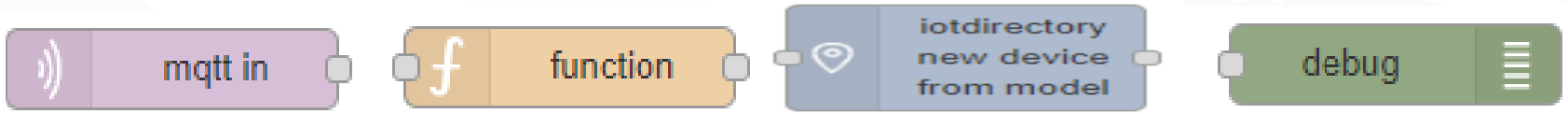


4) request on inject of a SURI to the storage to see data on debug.

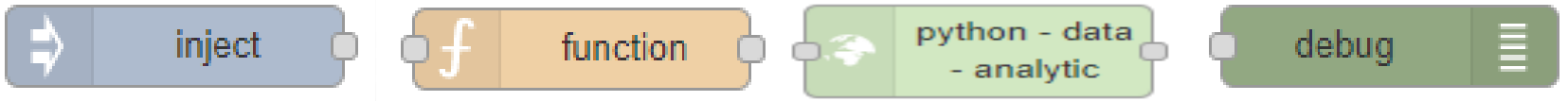
1) Preparation of data request on function, query to the storage and see data result on debug.



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):

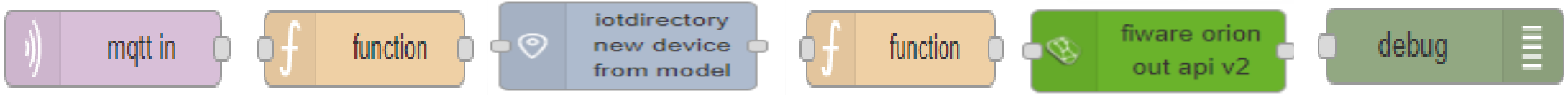


3) Preparation of data parameters on function, request computing Data Analytic, see data result on debug.



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

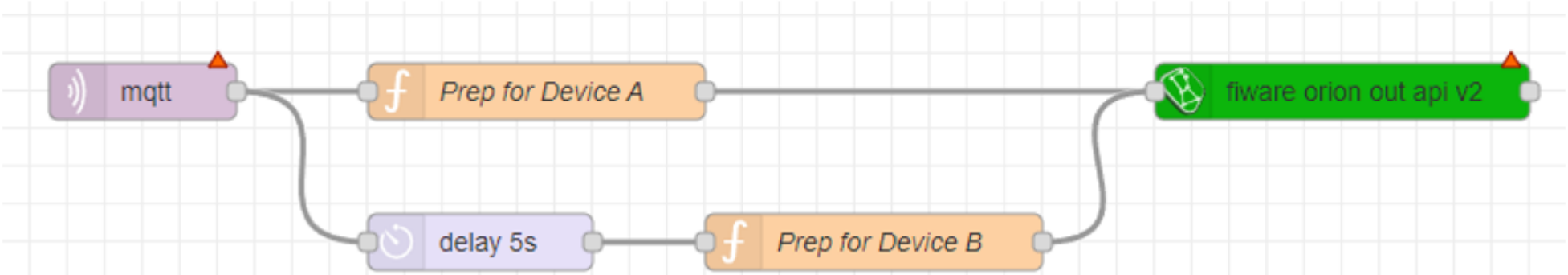
A. data reception from an MQTT broker, their transformation to create an Entity Instance from a known Entity Model, contextually to create and send an Entity Message into newly created Entity Instance, the debug to see eventual errors. This approach is typically strange since at each new message the 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.

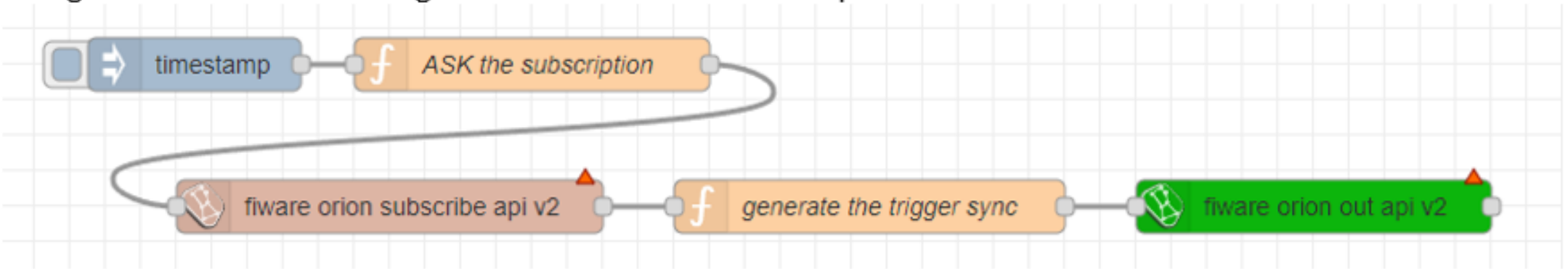
Sync data changes on Entities from an Event

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

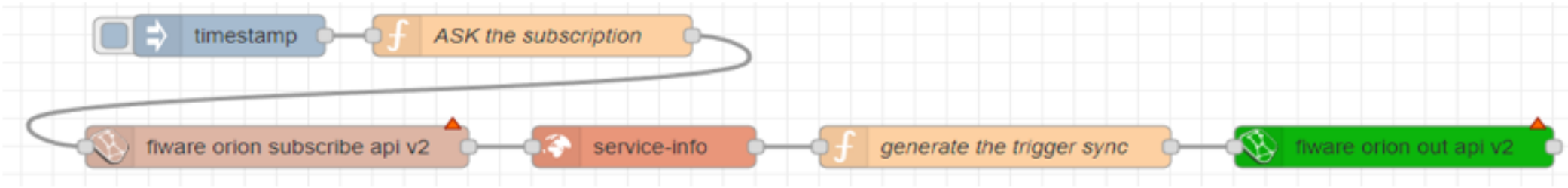


Subscribe on event from Broker, be carefull....

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



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



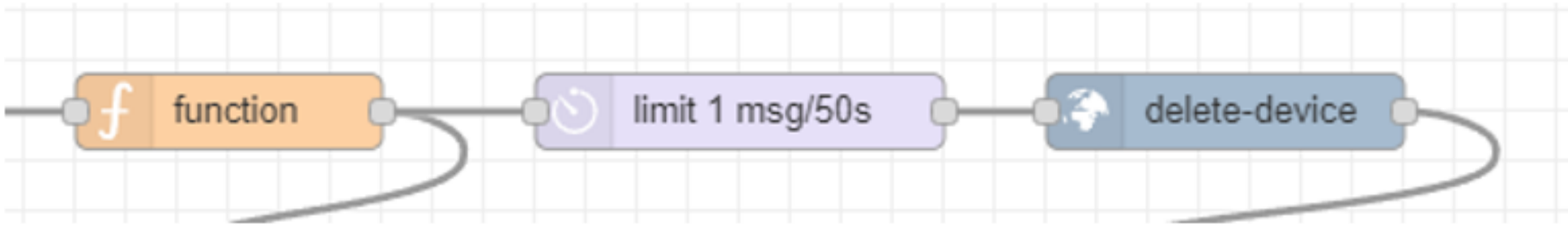
Delete Devices

IV.C.1.e- Delete Devices

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

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

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

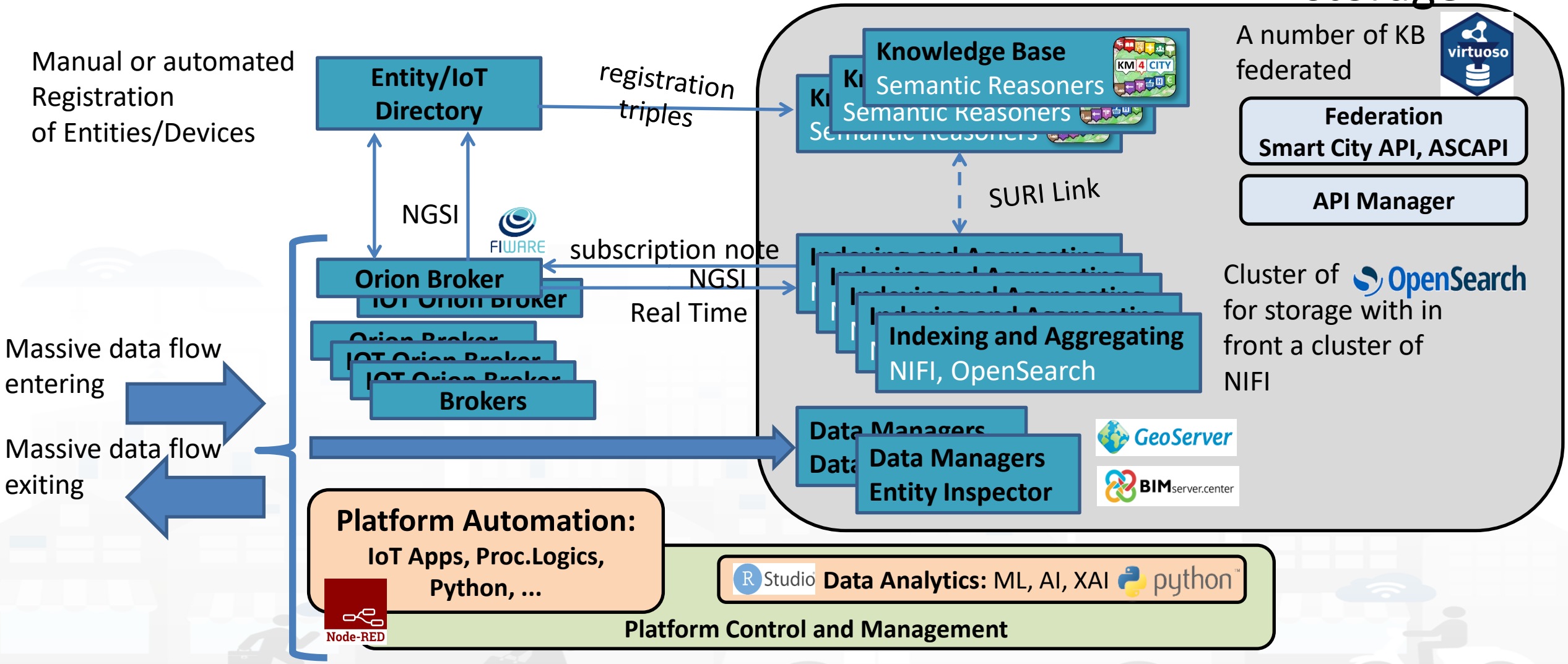


The delete device node needs in input Device ID and Broker ID. All data that you can recover from the Entity Directory.

Proc.Logic / IoT App Development

Part 3
Part 5

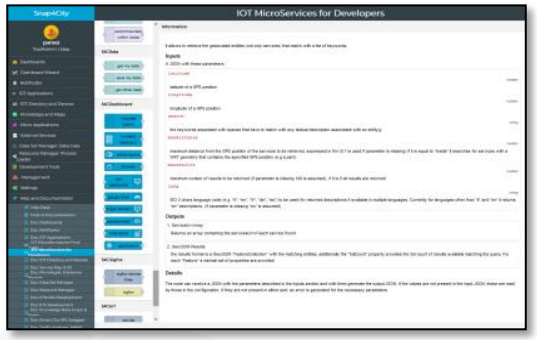
Snap4city Data Ingestion Diagram



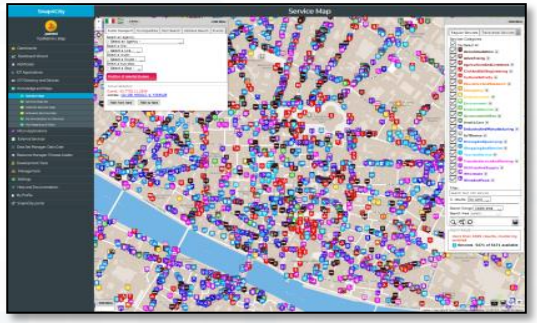
Proc.Logic / IoT App Development

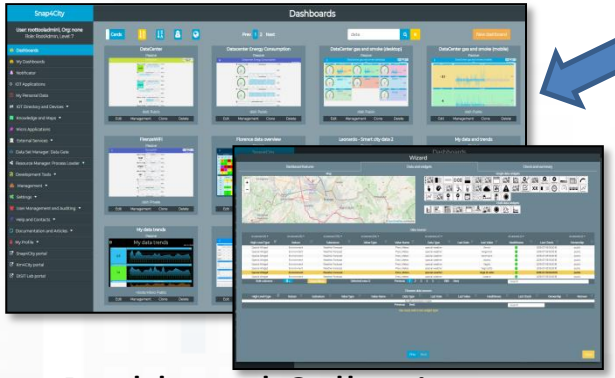
IOT Discovering

MicroServices collections



My IoT Applications





ServiceMap Discovery

Dashboard Collection, Editor and Wizard

IoT App. Editor

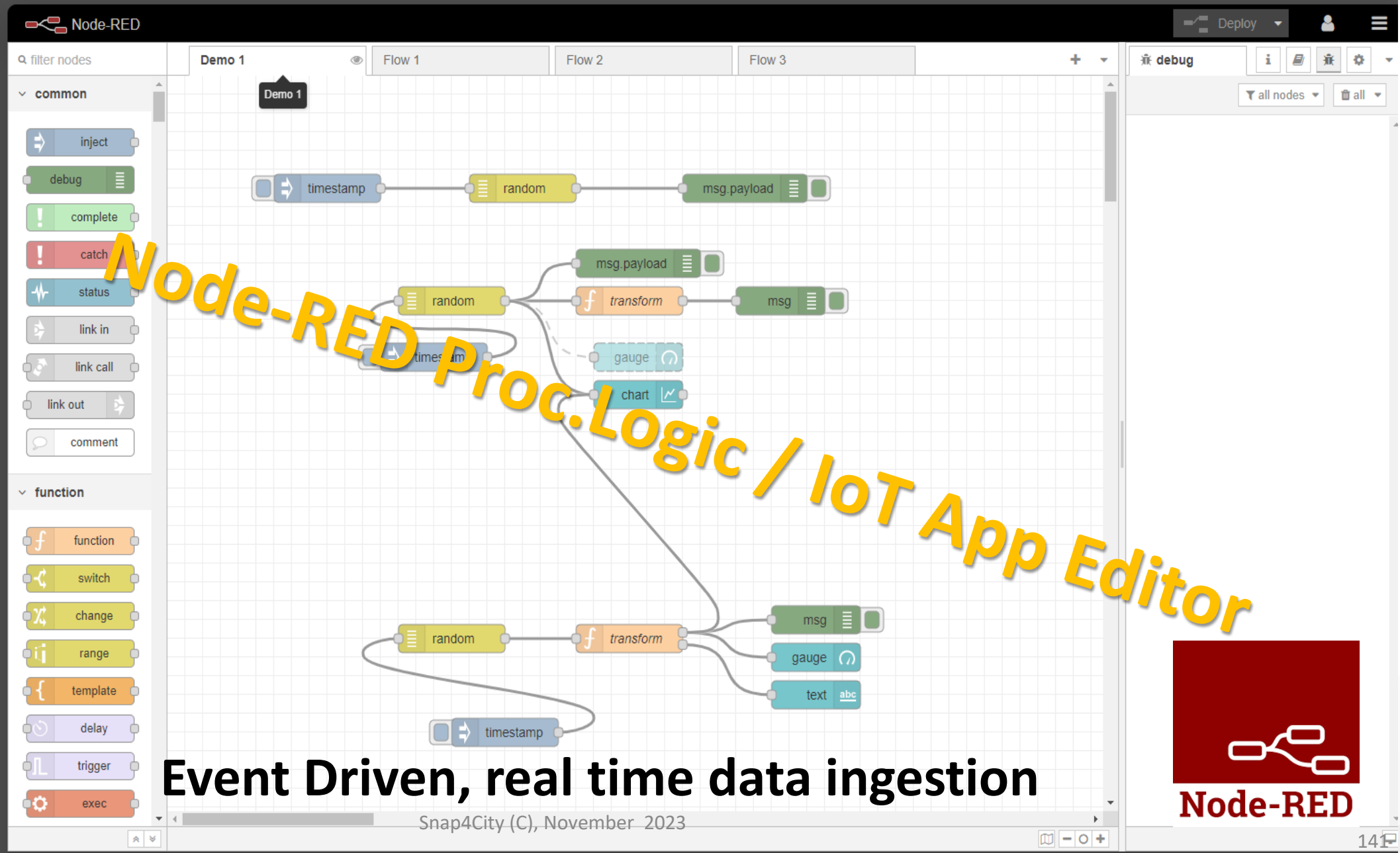
Generating IoT App With Dashboard

Sharing/saving reusing IoT App

Resource Manager

GitHub

Dashboards (Public)
My Snap4City.org
Tour Again
www.snap4solutions.org
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
Extra Dashboard Widgets
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Processing Logics / IOT App
MicroServices for Proc.Logic/IoT Apps
MicroServices from DataAnalytic
IOT MicroServices for Final Users
IOT MicroServices for Developers
DOC: Processing Logic/IOT App
How to Develop Proc.Logic / IoT Apps

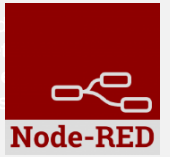


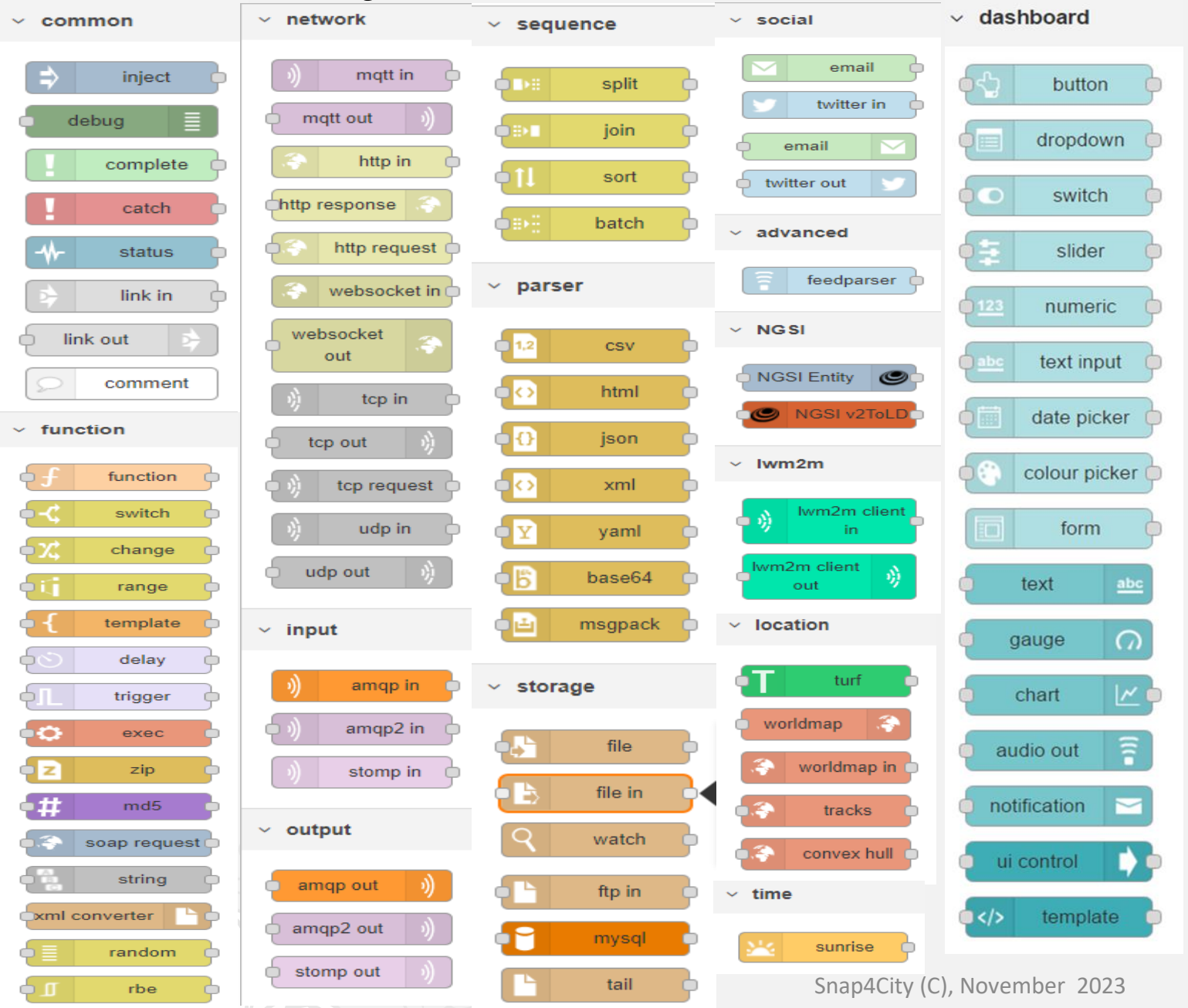
Node-RED Proc.Logic / IoT App Editor

Event Driven, real time data ingestion

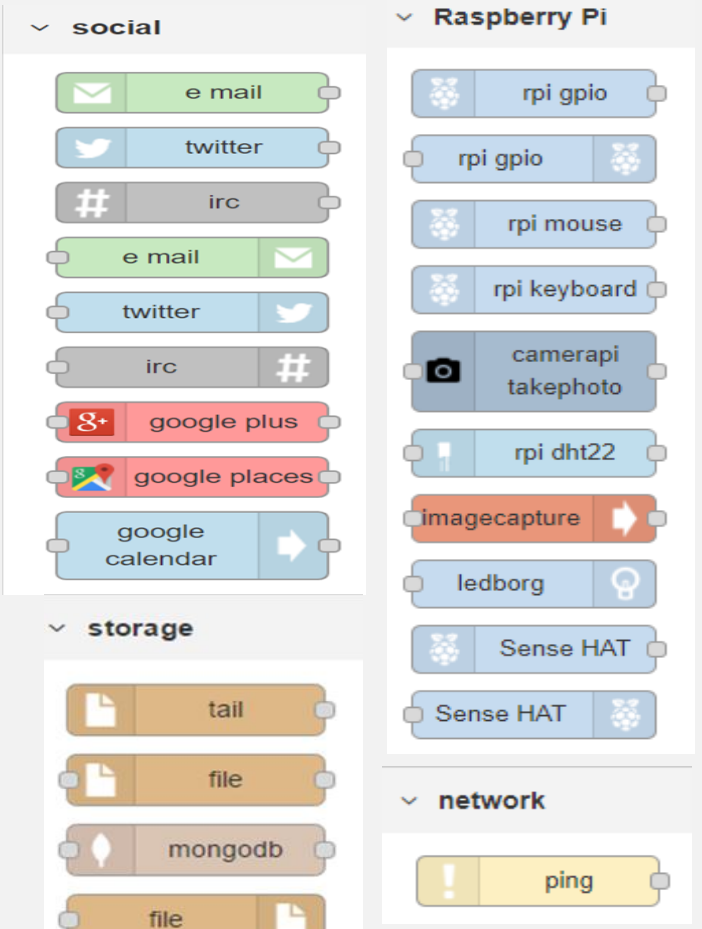
Basic Node.js Blocks on NodeRed on our Advanced IOT Apps

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





+ on IOT Edge Raspberry

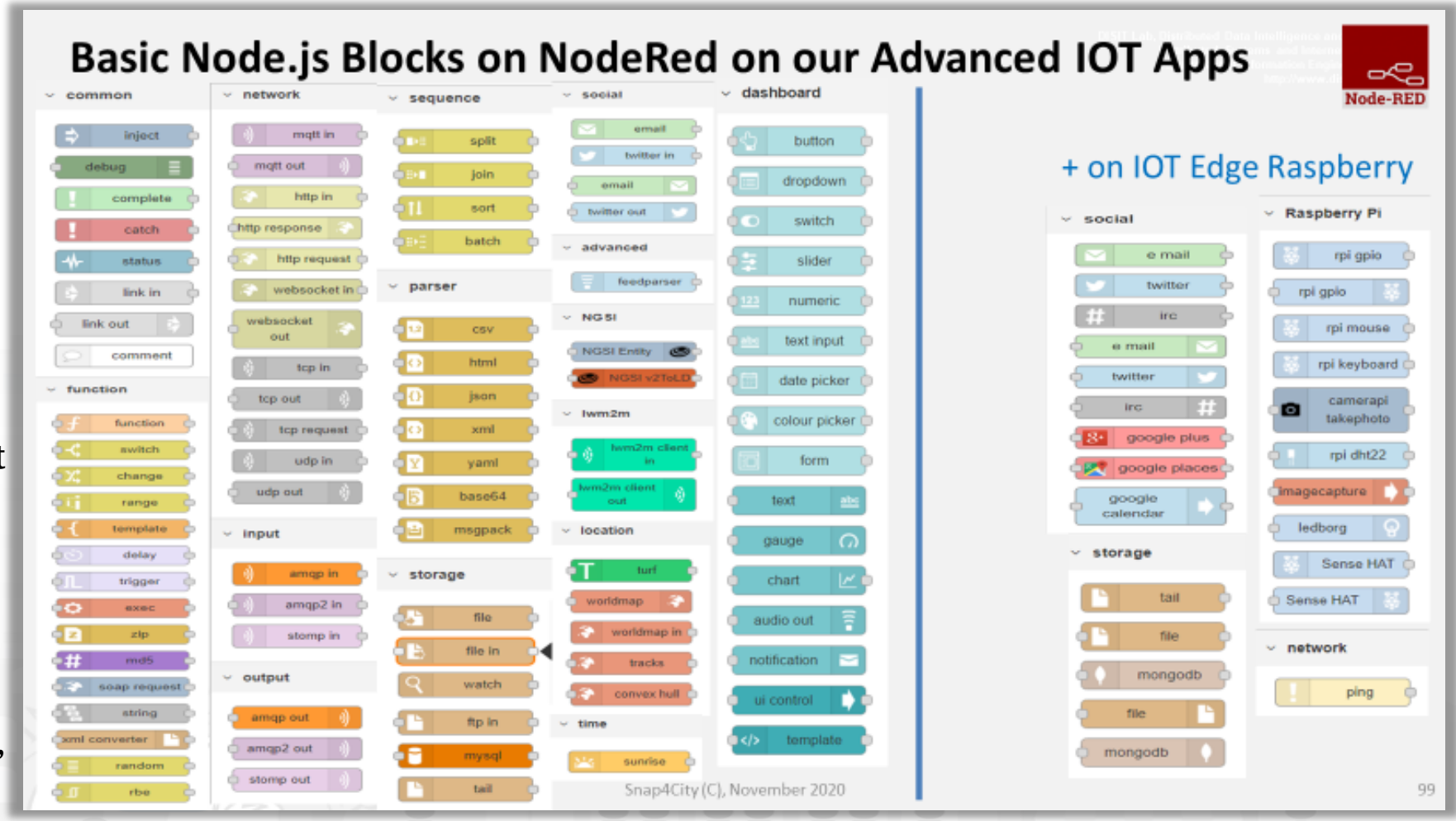


Node-RED Basic Blocks

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

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

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



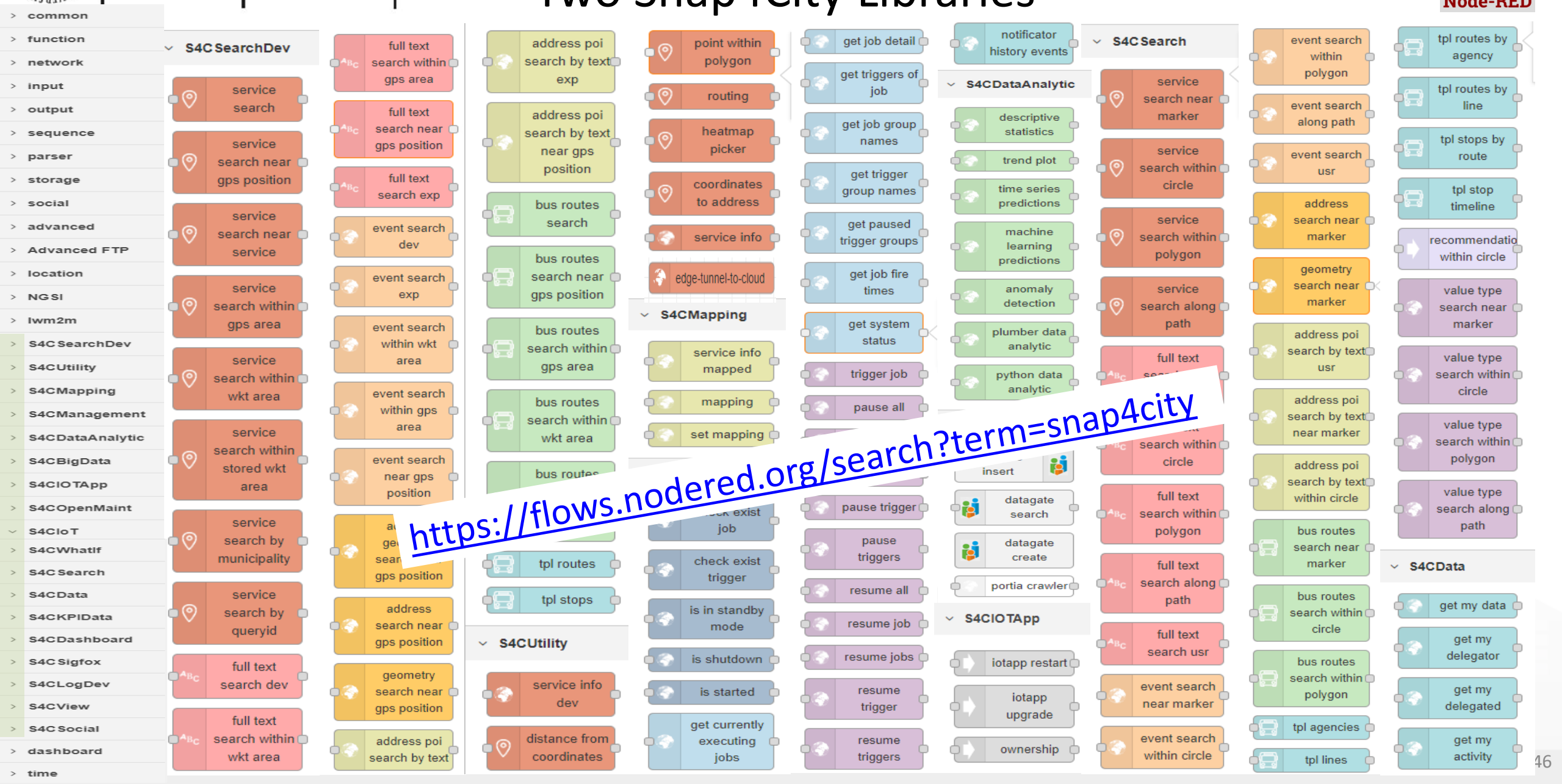
IoT Applications

- **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

2023 collection
Two Snap4City Libraries

SNAP4CITY | Node-RED



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

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

2023 collection

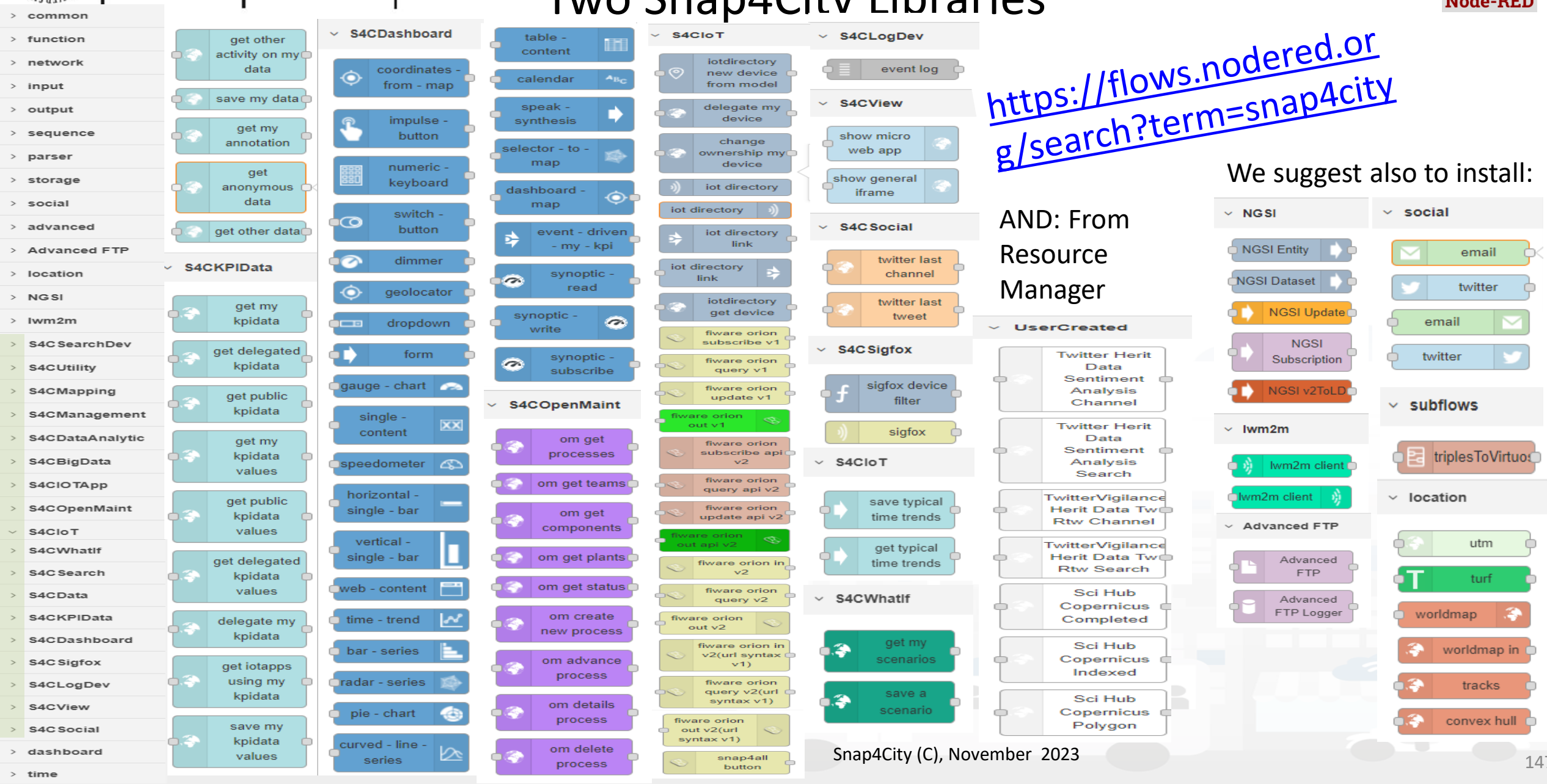
Two Snap4Citv Libraries

SNAP4CITY | Node-RED

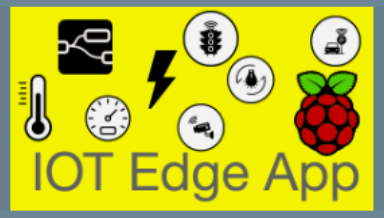
https://flows.nodered.org/search?term=snap4city

We suggest also to install:

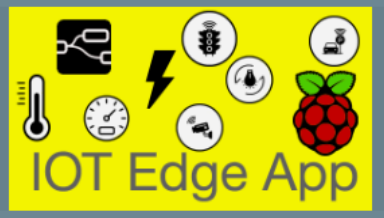
AND: From Resource Manager



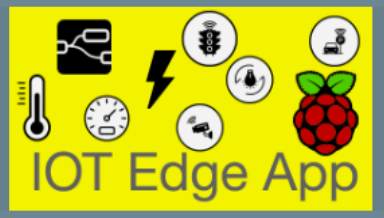
IOT Applications

2018-09-14T04:44


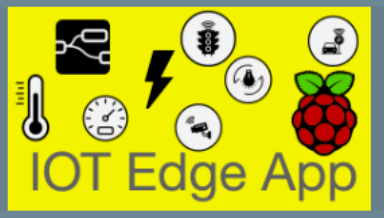
owner: badii
Management

2018-09-21T03:19


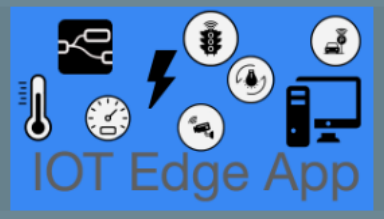
owner: panesi
Management

2018-10-19T16:07


owner: pb3
Management

2018-10-19T17:17


owner: pb3
Management

2018-10-22T11:57


owner: semolarudy
Management

application

owner: tester5
Management

Bib APP

owner: semolarudy
Management

ChargingStations

owner: comunedashres
Management

Deprecated - SiiMobilityControlRoom

owner: badii
Management

SamsungGalaxyS4BarCode

owner: badii
Management

esercitazione

owner: tester2
Management

Iot-App

owner: tester14
Management

Resource Manager: public and sharing

Snap4City

User: roottooladmin1, Org: DISIT
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
  - View Resources
  - Managing Resources
  - MicroServices for IOT Applications
  - Process Models
  - Processes in Execution
  - Process execution Archive
- Development Tools
- Management
- Settings
- User Management and Auditing
- Help and Contacts
- Documentation and Articles
- My Profile
- Snap4City portal

View Resources

Pages: Prev 1 2 3 ... 12 Next

dev

Reset | Username | Nature | Sub_nature | License | Resource_type | Format

Resource_type:
- IoTApp (118)
- ETL (53)
- MicroService (8)
- AMMA (4)
- R (3)
- DevDash (2)
- IoTBlocks (2)

Florence_Pharmacies_CSV.zip
developer1: Public
Username: developer1
Resource type: ETL
Nature: geolocated
Description: Florence Pharmacies o...
View Edit Unpublish Owner

developer1: Public
Username: developer1
Resource type: ETL
Nature: sensors
Description: Smart bench
View Edit Unpublish Owner

DeveloperDash-V3-1523555417880
Dev Dashboard
snap4city: Private
Username: snap4city
Resource type: DevDash
Nature: data category (ie: geolocat...
Description: Snap4city Developer D...
View Edit Publish Owner

node-red-contrib-snap4city-developer.rar
snap4city: Private
Username: snap4city
Resource type: IoTBlocks
Nature: data category (ie: geolocat...
Description: Snap4city NodeRed Li...
View Edit Publish Owner

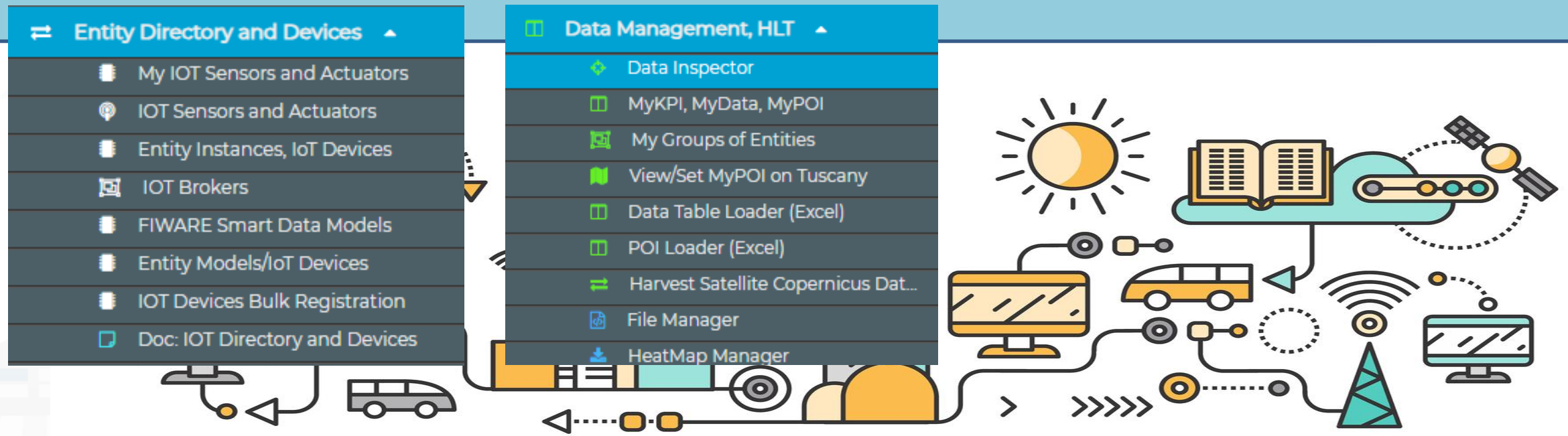
PaoloApplication.json
IOT Application
developer1: Private
Username: developer1
Resource type: IoTApp
Nature: data category (ie: geolocat...
Description: NodeRed Flow Shared ...
View Edit Publish Owner

AMMADashSnap4City-30minview-v2-152...
AMMA Tool
developer1: Private
Username: developer1
Resource type: AMMA
Nature: ToBeDefined
Description: AMMA snap4city dash...
View Edit Publish Owner

Developer Dashboard New-1526308876256
Dev Dashboard
developer1: Private
Username: developer1
Resource type: DevDash
Nature: ToBeDefined
Description: Developer Dashboard ...
View Edit Publish Owner

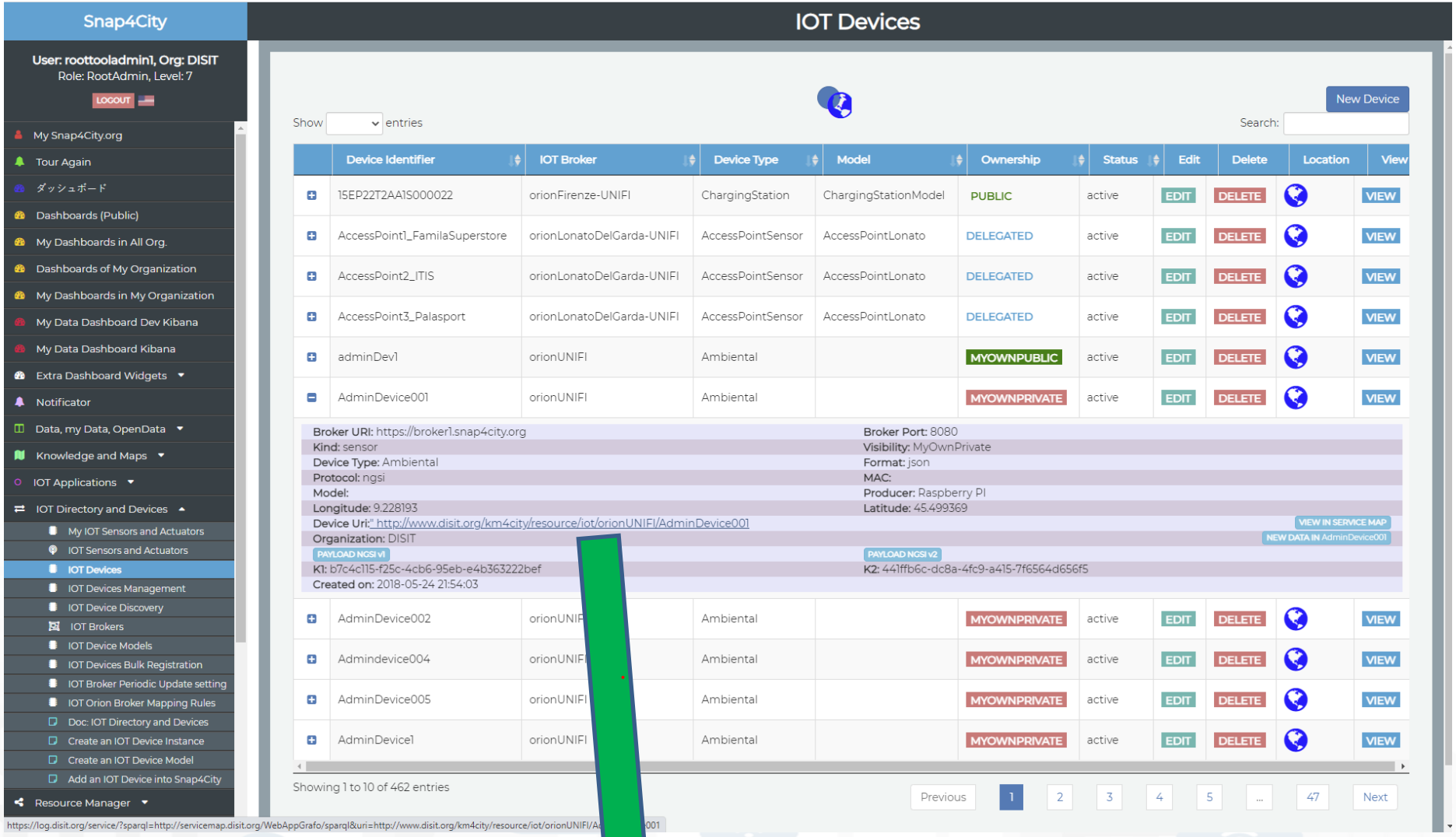
ResDash Docker-1526308998809
Res Dashboard
developer1: Private
Username: developer1
Resource type: ResDash
Nature: ToBeDefined
Description: Resource Dashboard: ...
View Edit Publish Owner

search vs services, the ServiceURI

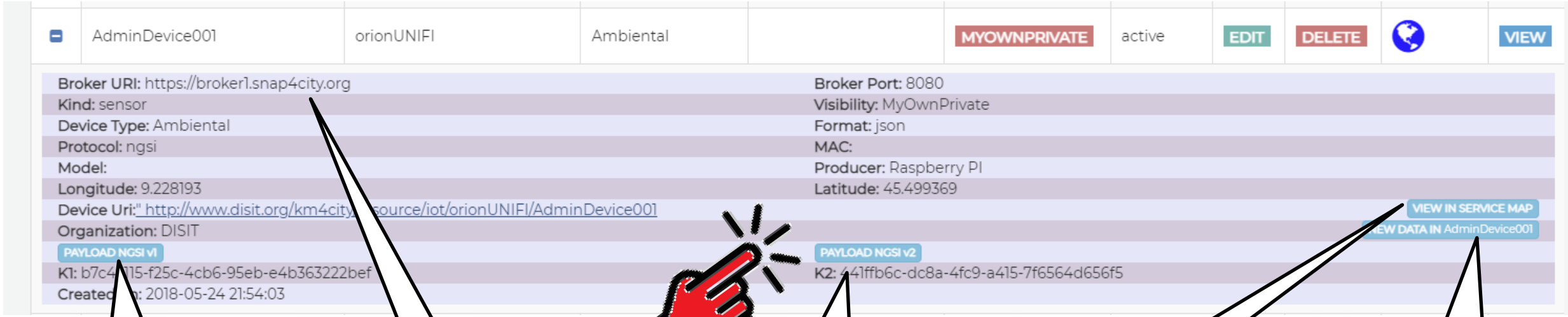


IoT Directory

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



Understanding / Testing an Entity/ IoT Device



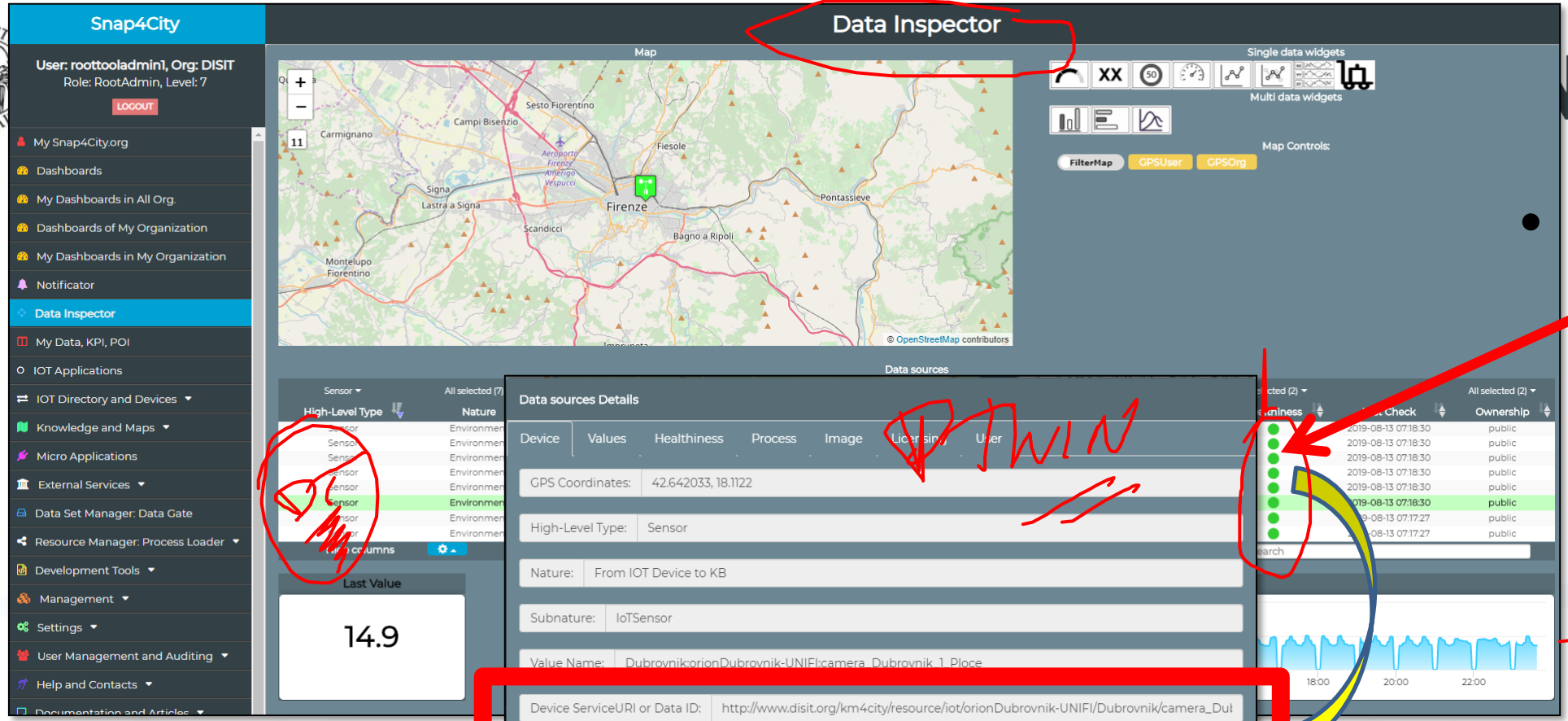
See Payload NGSI V1 in JSON directly from the Broker, Last message of the broker

The Broker

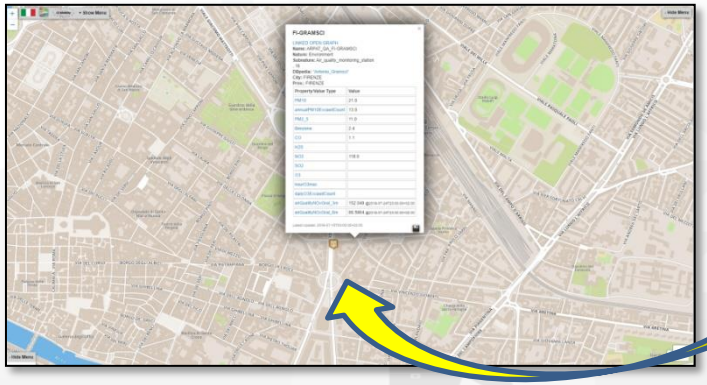
See Payload NGSI V2 in JSON directly from the Broker, Last message of the broker

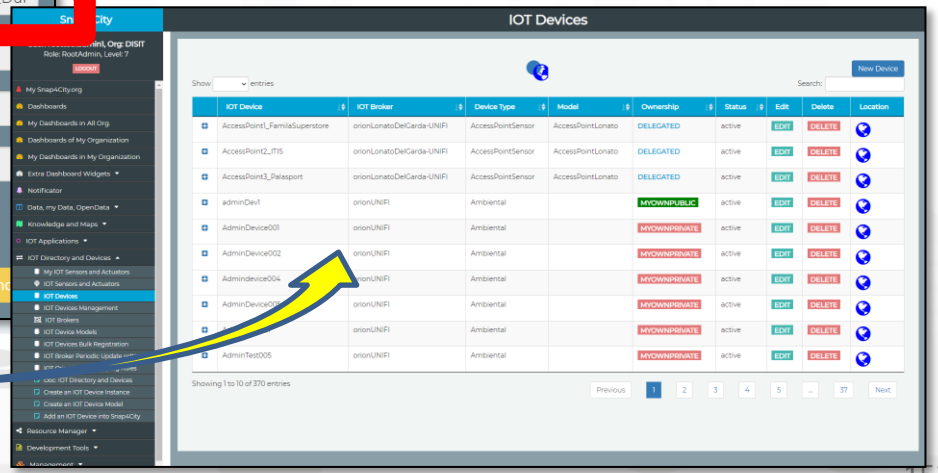
See IoT Device on ServiceMap

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



Knowledge Base view





- Click with the mouse on it

Data Inspector Wizard

Some functionalities are limited to certain roles

Notation Terminology

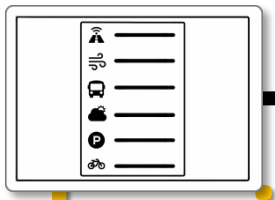
WHERE	Are synonymous at level of service which can be IoT device or entity with data and references to	Are synonymous at level of the single attribute of the entity, device, service, etc.
IoT Directory, Entity Directory	IoT Device, Entity Instance, Device URI	Sensor, Actuator, Attributes, Values (value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

ServiceURI, SURI of a sensor device:

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

ServiceURI, SURI extended with attribute/variable/value:

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



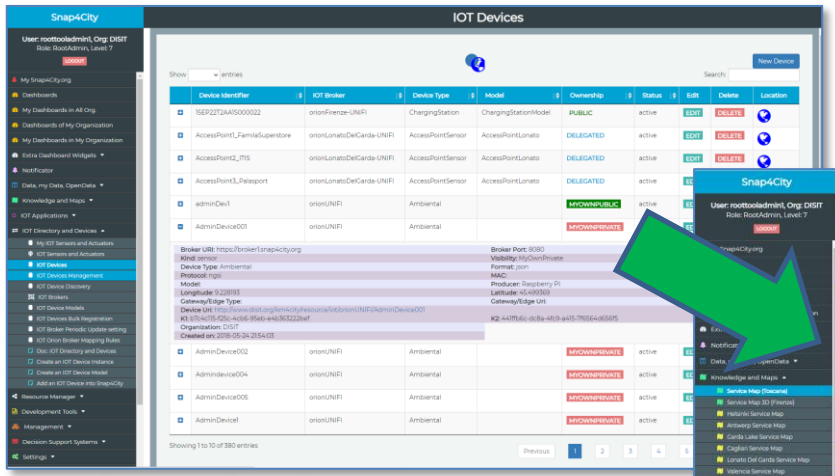
Dashboard Usage and recipe: Event map target

Parts 2, 3 and 7

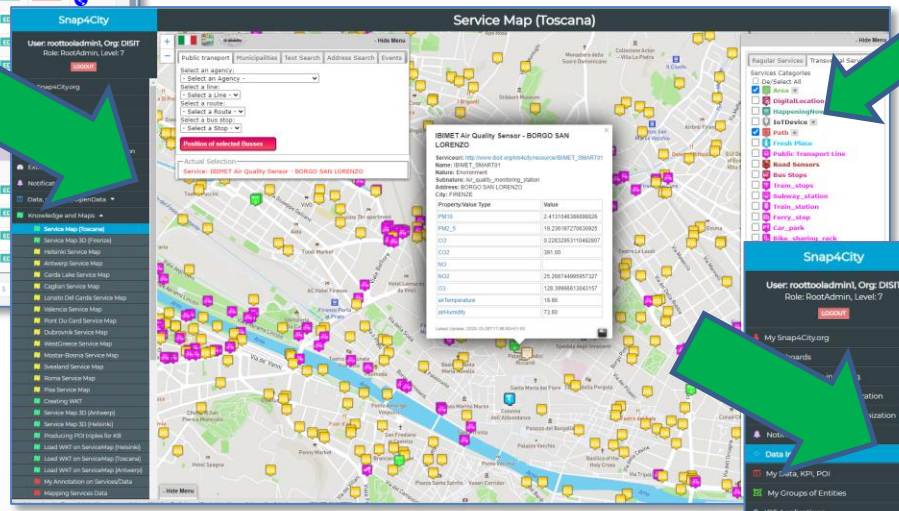
- **Selector to Show on Map a**
  - **category of Map positioned elements**
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.08694333811321;8.791809082031252;43.93568500391093;14.065246582031252&categories=Traffic_sensor&maxResults=0&maxDists=0.1&text=&model=&value_type=&format=json
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?queryId=e5f39066cd68ffe259ed8877bcee222b&format=json
  - **Entity by Model**
    - https://www.disit.org/superservicemap/api/v1?selection=59.36535064975547;13.457822799682619;59.39031474260852;13.56099435424806&model=SmartLightCapelon&format=json
  - **Single Entity**
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/iot/orionFirenze2/Firenze/SH%20Lab_new&format=json&fromTime=3-day
  - **Heatmap among many**
    - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=Florence_PM10
  - **Traffic flow**
    - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=FirenzeFIPILITrafficRealtime&trafficflowmanager=true
    - https://firenzetraffic.km4city.org/trafficRTDetails/roads/read.php
  - **Origin Destination Map**
    - https://odmm.snap4city.org/api/get?precision=communes&from_date=&organization=Toscana&inflow=True&longitude=11.255751&latitude=43.769710&od_id=mobile_Toscana_1000&perc=True
- **Events which are also PIN on map**

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

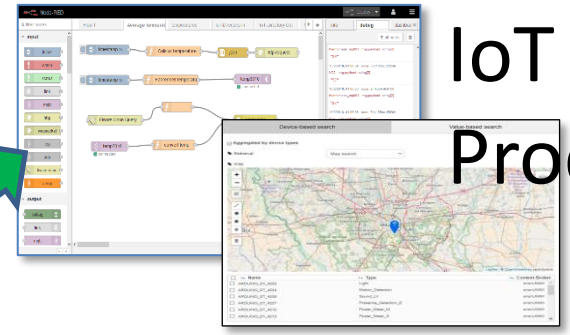
Data Registration Flow at a Glance



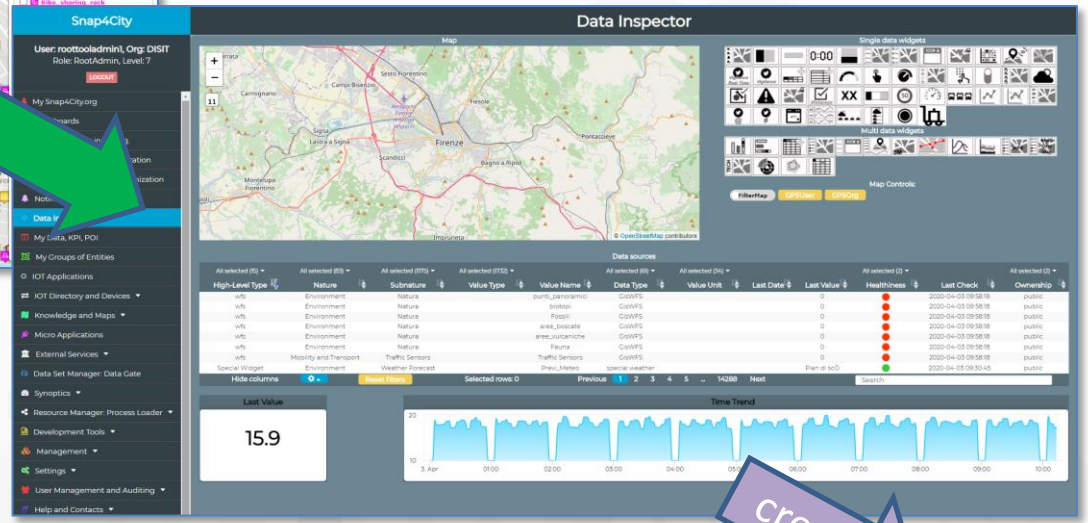
IOT Directory:
Devices…
Sensors..
Actuators…



Knowledge Base,
ServiceMap,
SuperServiceMap
SmartCity API,
ASCAPI



IoT Apps
Proc.Logic



DataInspector
Dashboard Wizard

create

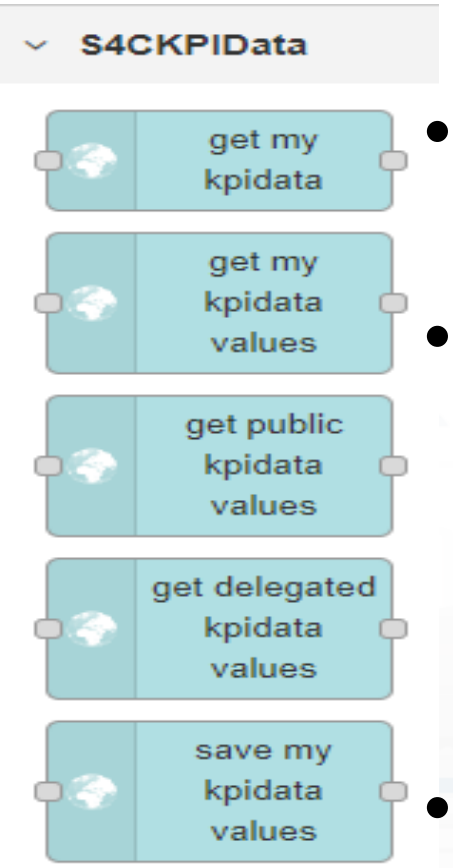
Dashboards

MyKPI Nodes

Part 3

Part 5

S4C KPI Data



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

TOOLS for Data Ingestion Verifications

Part 3

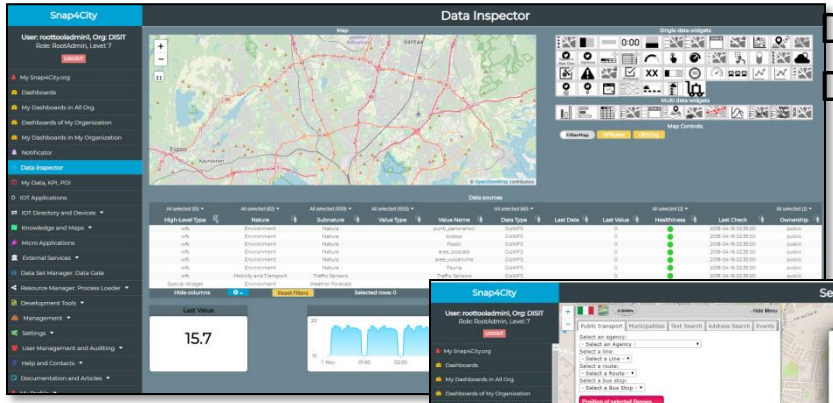
Part 5

Checking data/Entity ingestion results

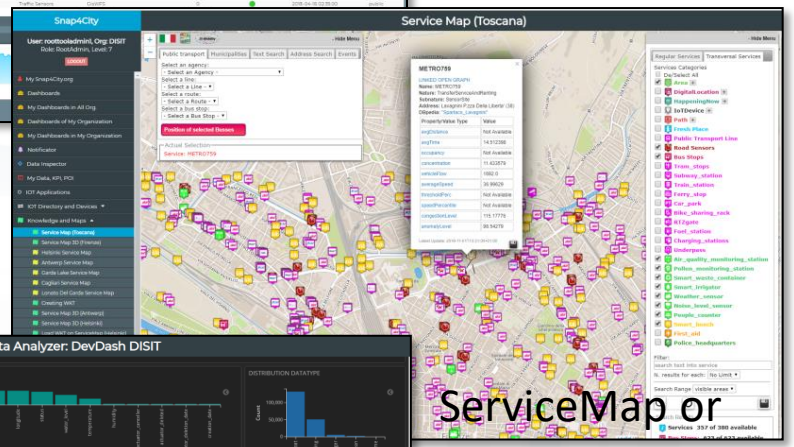
Knowledge base
Semantic reasoners

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

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



Data Inspector
Digital Twin view

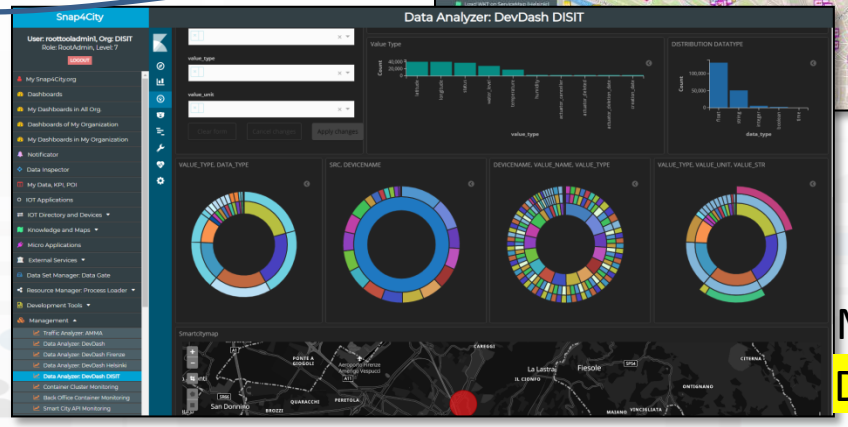


ServiceMap or
Super ServiceMap

Indexing and aggregating
NIFI, OpenSearch

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

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



My Data Dashboard
DevDash

Some functionalities are limited to certain roles

Data Inspector

Data Inspector

Snap4City

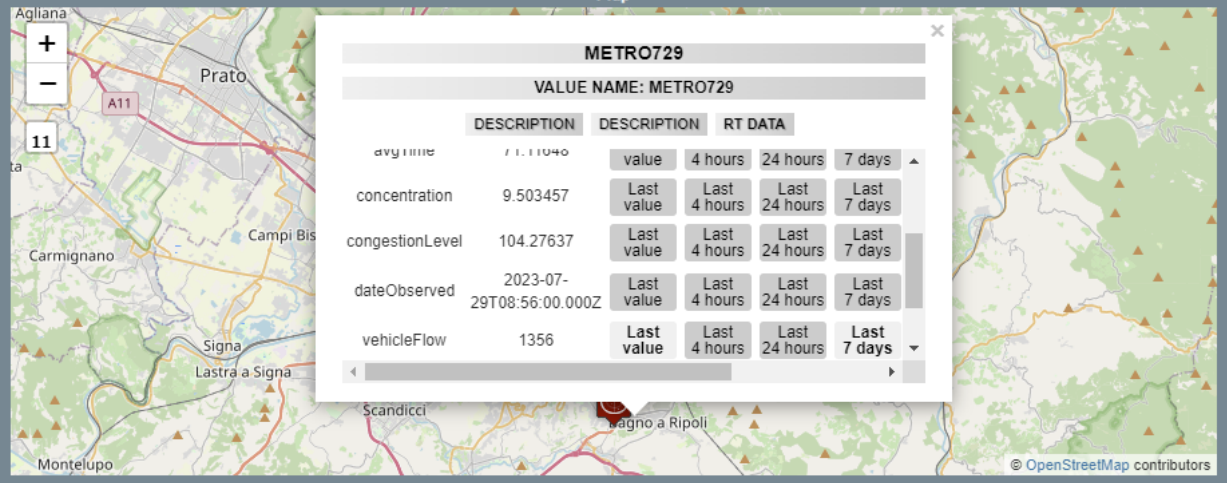
Switch To New Layout (Beta)

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

LOGOUT

- www.snap4solutions.org
- Dashboards (Public)
- Dashboards of My Organization
- My Dashboards in My Organization
- My Data Dashboard Dev Kibana
- Extra Dashboard Widgets
- Data Management, HLT
  - 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
  - File Manager
  - HeatMap Manager
  - BIM Server old
  - BIM Server New
  - BIM Srv New: Add
  - BIM Srv new: View
  - OpenData Manager: Data Gate
  - OpenData Manager: Data Gate
  - Add Data Sources into the Platform
  - Doc: Data Table Loader
  - Doc: POI Loader

Map



METRO729

VALUE NAME: METRO729

DESCRIPTION	DESCRIPTION	RT DATA			
		value	4 hours	24 hours	7 days
concentration	9.503457	Last value	Last 4 hours	Last 24 hours	Last 7 days
congestionLevel	104.27637	Last value	Last 4 hours	Last 24 hours	Last 7 days
dateObserved	2023-07-29T08:56:00.000Z	Last value	Last 4 hours	Last 24 hours	Last 7 days
vehicleFlow	1356	Last value	Last 4 hours	Last 24 hours	Last 7 days

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	Last
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO792	orionUNIFI			sensor_map		2023-07-28 13:26:00			2023-07
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO791	orionUNIFI			sensor_map		2023-07-28 13:26:00			2023-07
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO793	orionUNIFI			sensor_map		2023-07-28 13:16:00			2023-07
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO713	orionUNIFI			sensor_map		2023-07-28 13:16:00			2023-07
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO729	orionUNIFI			sensor_map		2023-07-28 13:16:00			2023-07
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO7	orionUNIFI			sensor_map		2023-07-28 13:16:00			2023-07
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO760	orionUNIFI			sensor_map		2023-07-28 13:16:00			2023-07
IoT Device	TransferServiceAndRenting	Traffic_sensor	METRO799	orionUNIFI			sensor_map		2023-07-28 13:16:00			2023-07

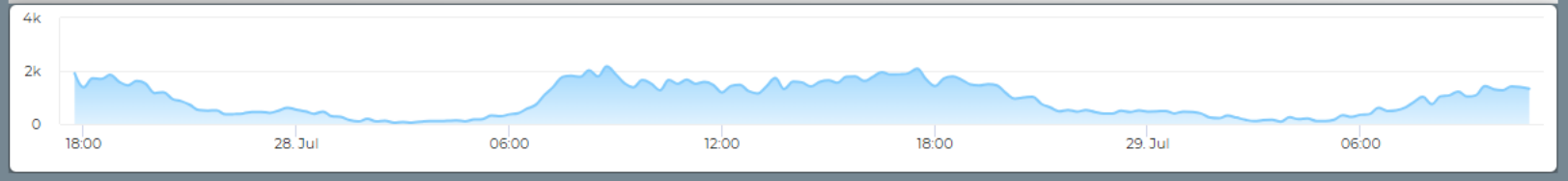
All selected (24) | All selected (48) | All selected (895) | All selected (11) | All selected (1595) | All selected (234) | All selected (48) | All selected (61) | All selected (3)

Hide columns | Reset filters | Selected rows: 1 | Previous 1 2 3 4 5 ... 262 Next | metro7

vehicleFlow

1356

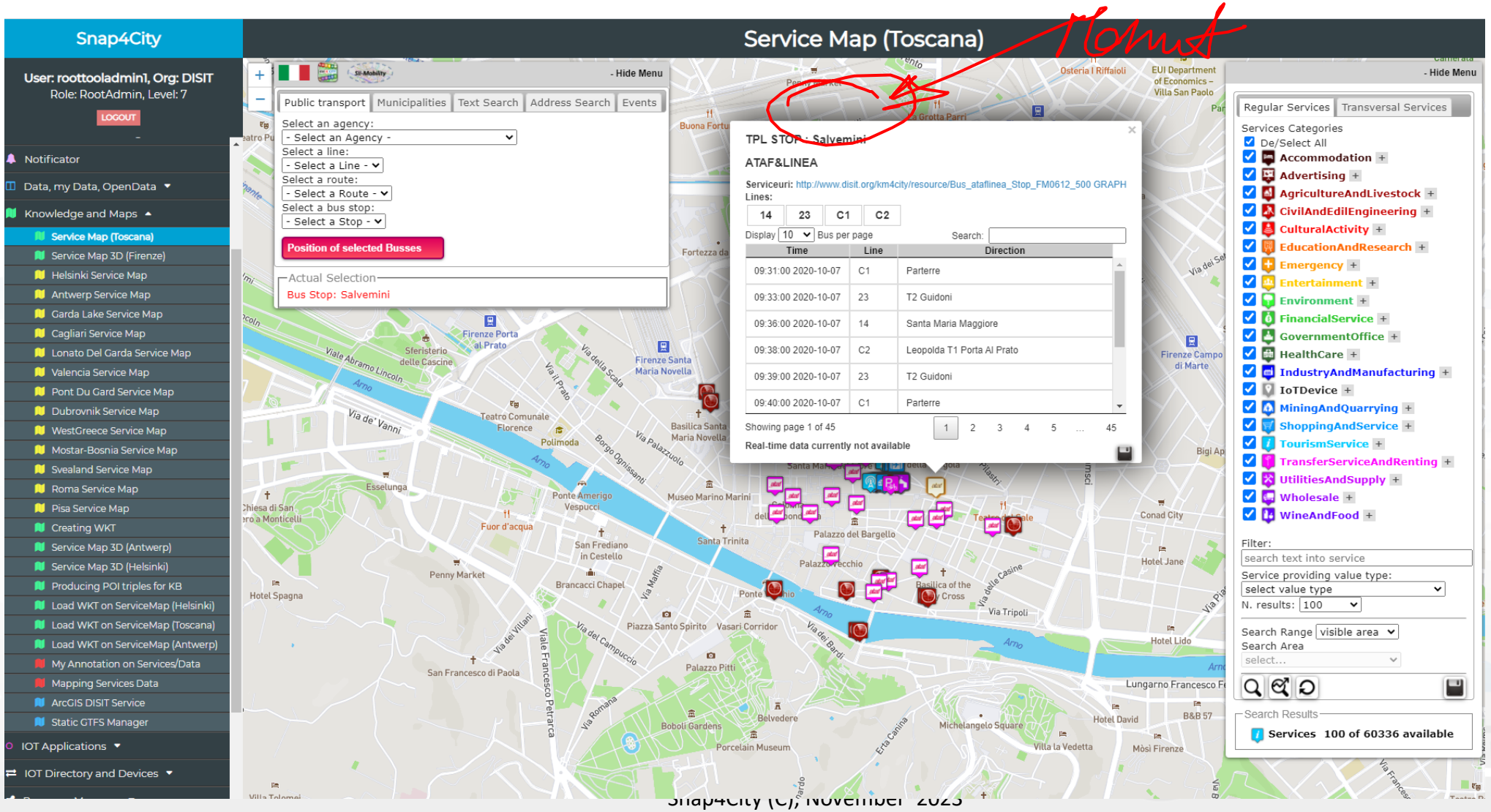
vehicleFlow - 7 days



Data Inspector for Beginner

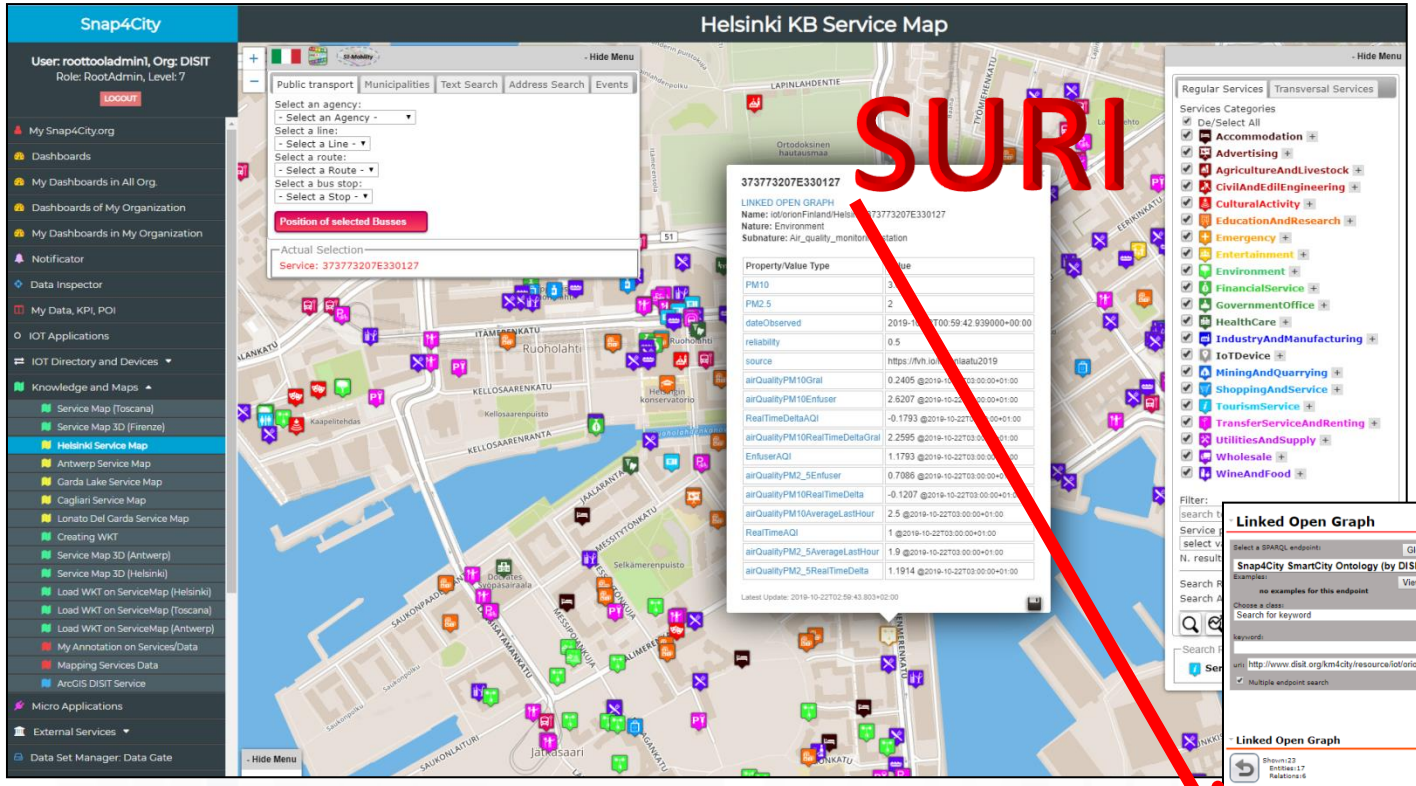
- Browse and see models data via HLT, nature and Subnature
  - All the other faceted views, search and filter, filter by map, etc.
- Identify, click them to see
  - Remaining icons representing dashboard widgets which can be used in the Dashboard Wizard
  - ICON: Click on the icon on map and on value to preview data time serie if any
- See detailed Digital Twin data on the microbutton of the healthiness
- Wider data preview is coming

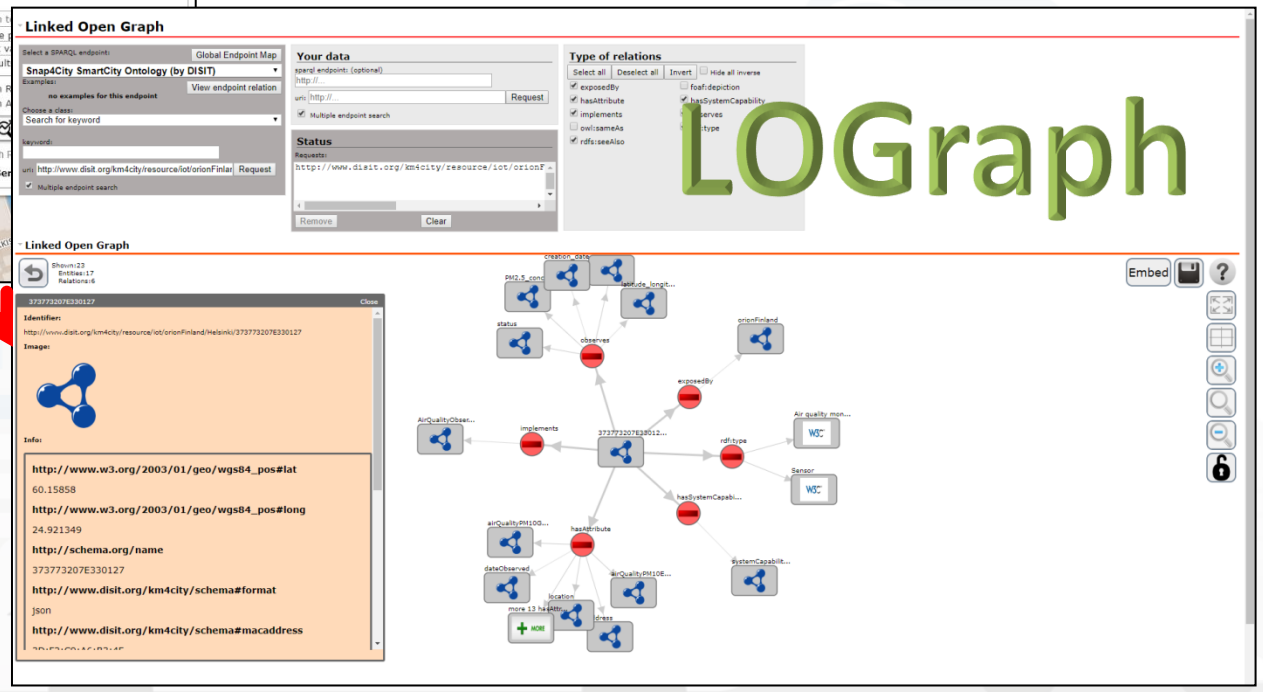
ServiceMaps/Super ServiceMap



Views of the Knowledge Base

Knowledge Base
Semantic Reasoners



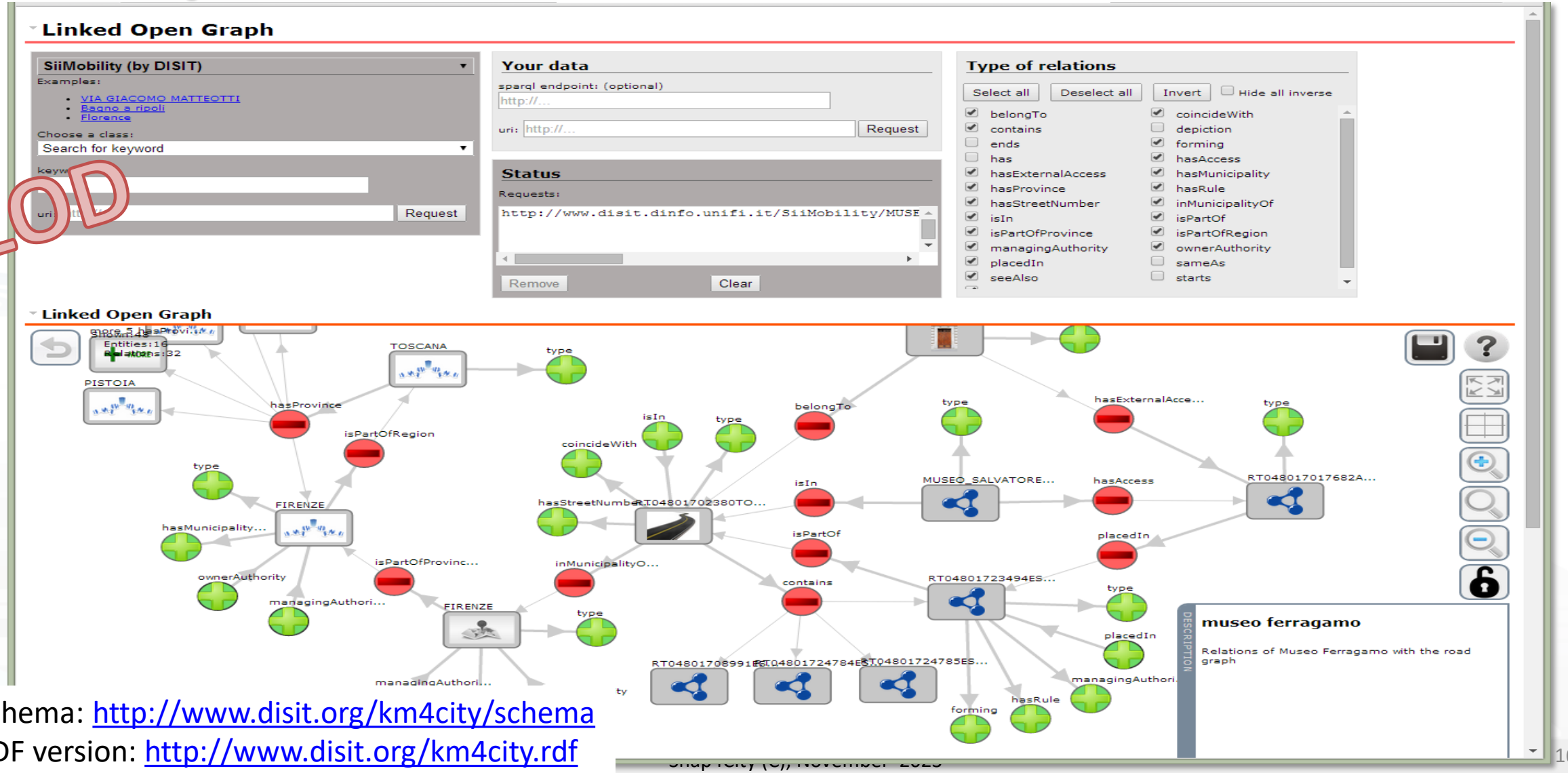


- How pass from ServiceMap to Linked Open Graph, Linked Data view tool

LOGraph

Linked Open Data

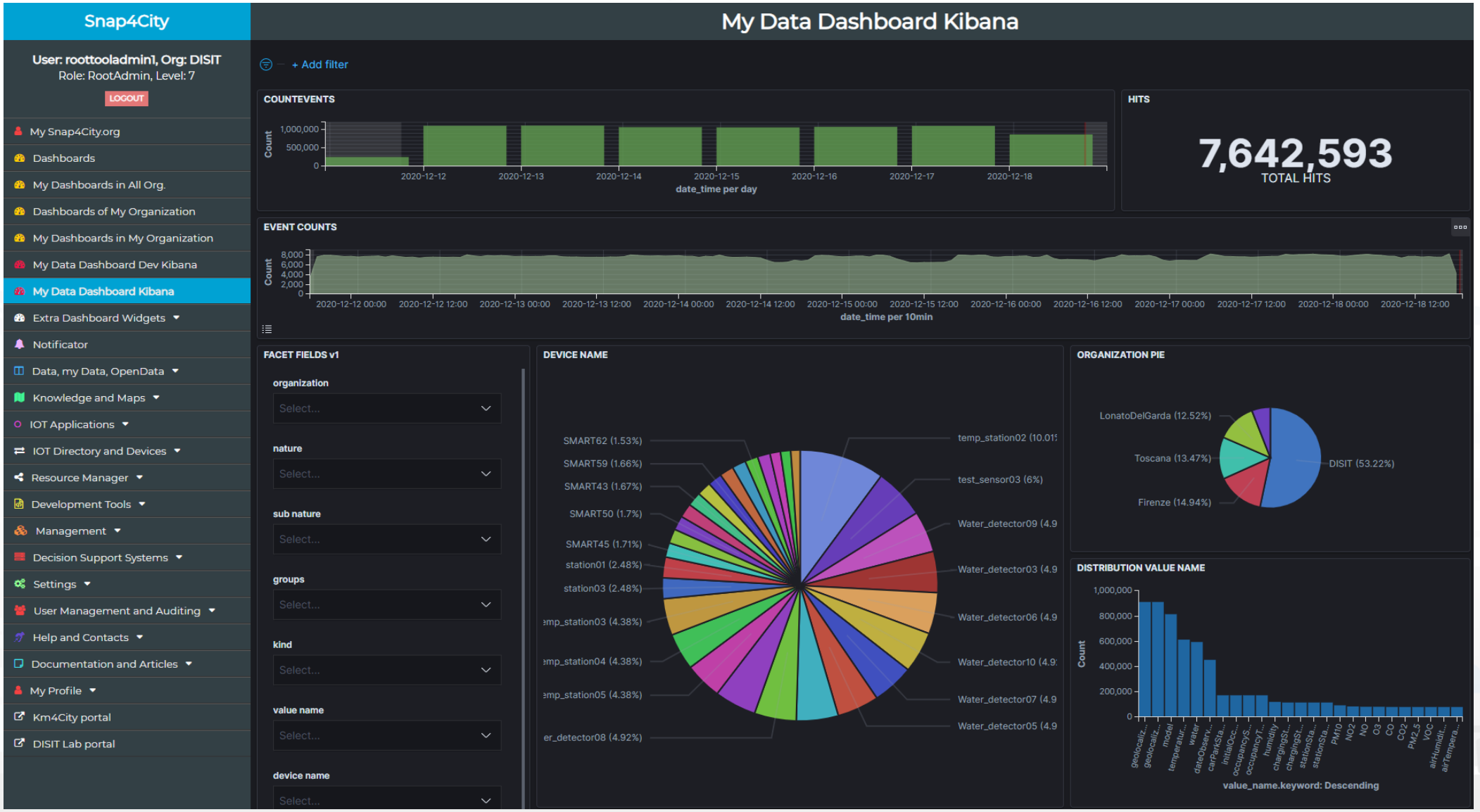
LOG: https://log.disit.org



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

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

DevDash: My Data Dashboard



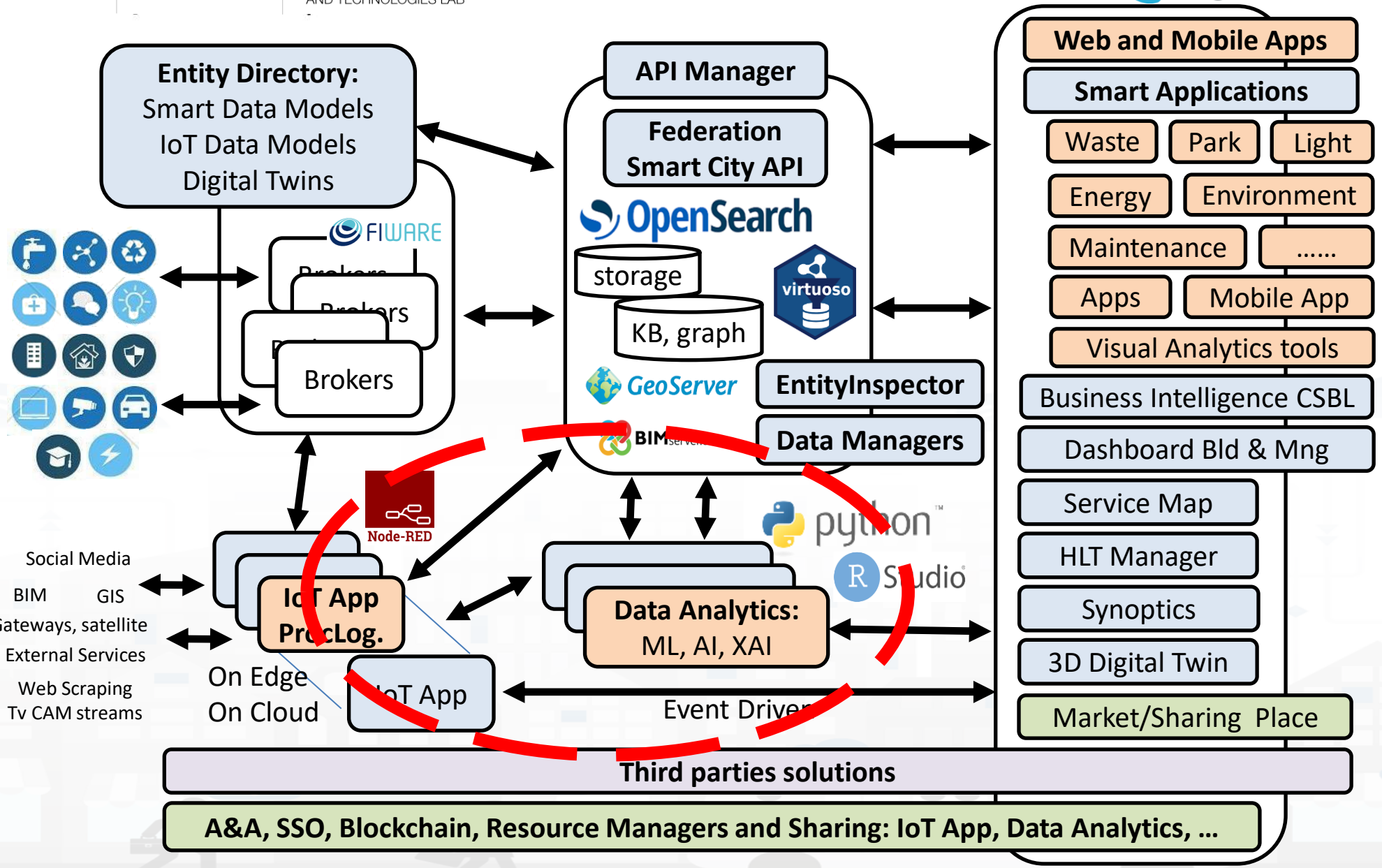
TOP

Design & Develop of Data Analytics

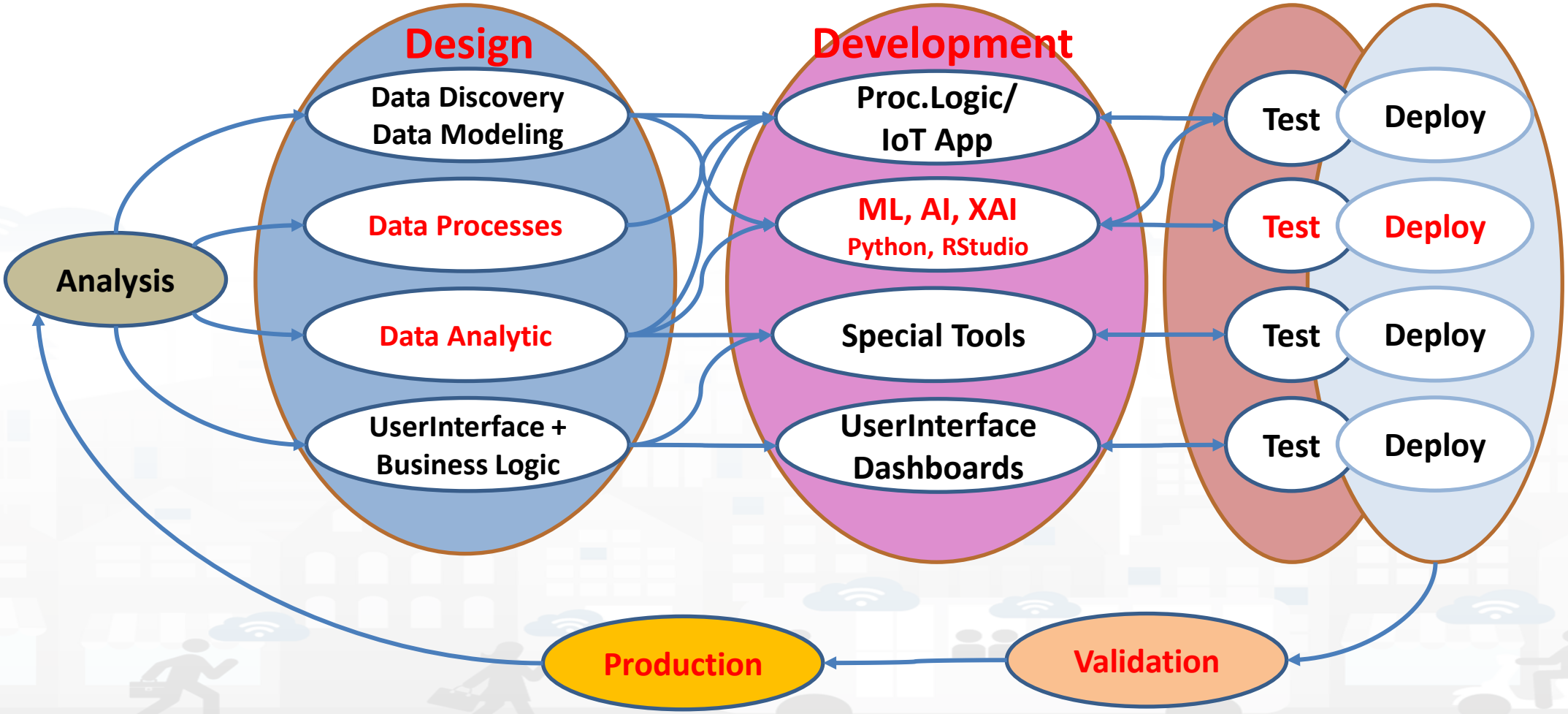
Part 4

100% OPEN SOURCE

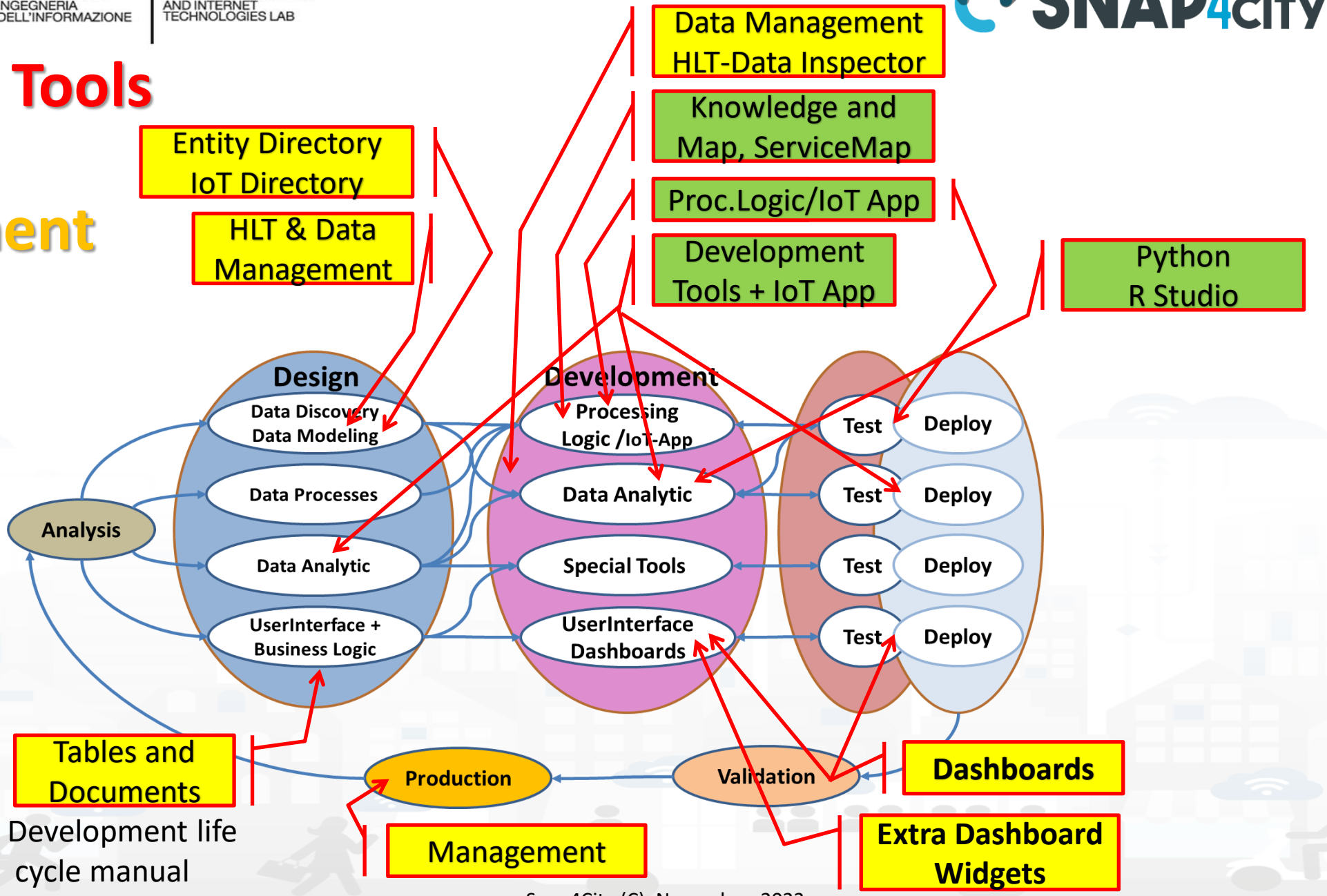
Tech Arch


Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIM
Data Managers
Node-RED
python
RStudio
Social Media
BIM
GIS
Gateways, satellite
External Services
Web Scraping
Tv CAM streams
IoT App
ProcLog.
On Edge
On Cloud
IoT App
Data Analytics:
ML, AI, XAI
Event Driven
Web and Mobile Apps
Smart Applications
Waste
Park
Light
Energy
Environment
Maintenance
……
Apps
Mobile App
Visual Analytics tools
Business Intelligence CSBL
Dashboard Bld & Mng
Service Map
HLT Manager
Synoptics
3D Digital Twin
Market/Sharing Place
Third parties solutions
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, …

Development Life Cycle Smart Solutions



Snap4City Tools vs Development Life Cycle

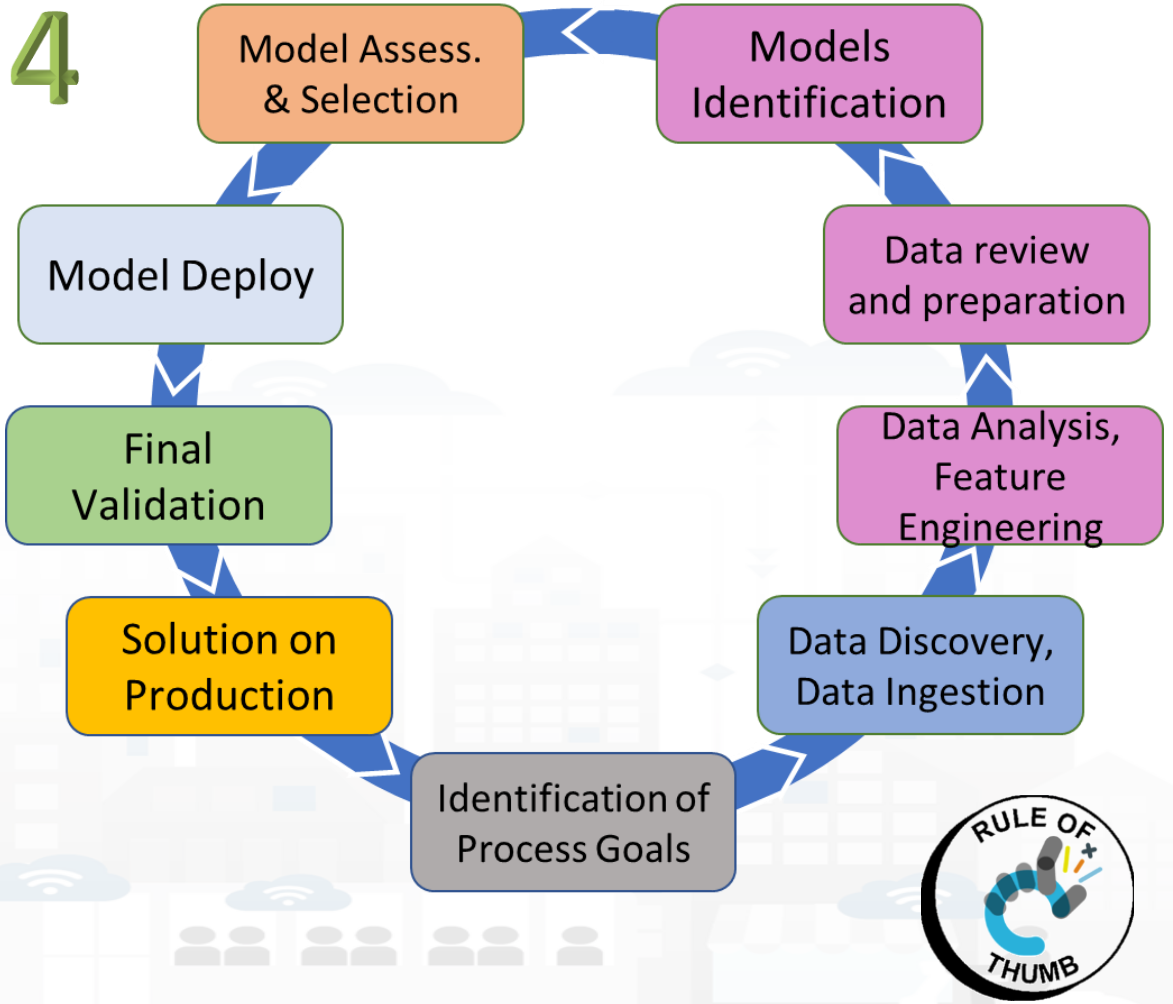


Development life cycle manual

Model/Technique Development/testing

Part 4

- **Identification of Process goals and Planning**
  - Which goals
  - How to compute, which language
  - Which environment, which libraries
- **Data Discovery and Ingestion (from the general life cycle)**
- **Data Analysis: feature engineering, feature selection**
- **Data review and preparation for the model**
- **Model Identification and building: ML, AI, etc….**
  - **Training**
  - **Tuning hyperparameters when possible**
- **Model Assessment and Selection**
  - **Validation in testing**
  - **Assessment** on a set of **metrics** depending on the goals: global relevant and feature assessment
  - **Assessing computational costs**
  - **Impact Assessment, Ethic Assessment** and incidental findings
  - **Global and Local Explanation via Explainable AI techniques**
- **Model Deploy and Final Validation**
  - **Optimisation** of computation cost for features, if needed reiterate
- **Solution on Production (security, scalability, etc.)**



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

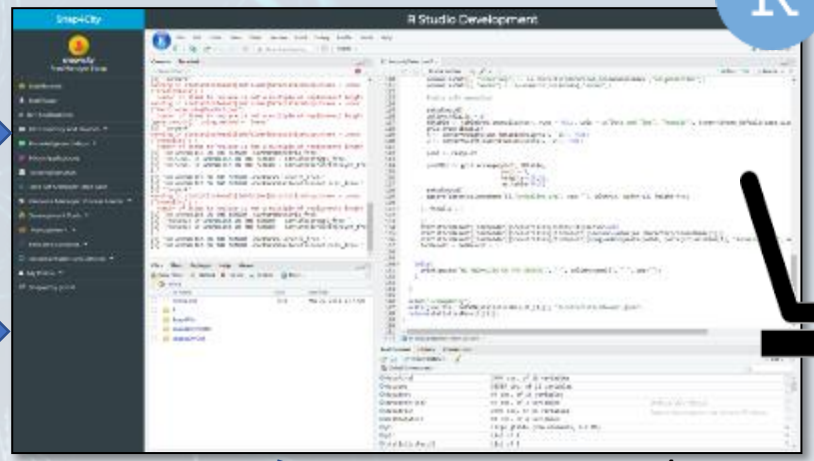
TensorFlow

NVIDIA CUDA

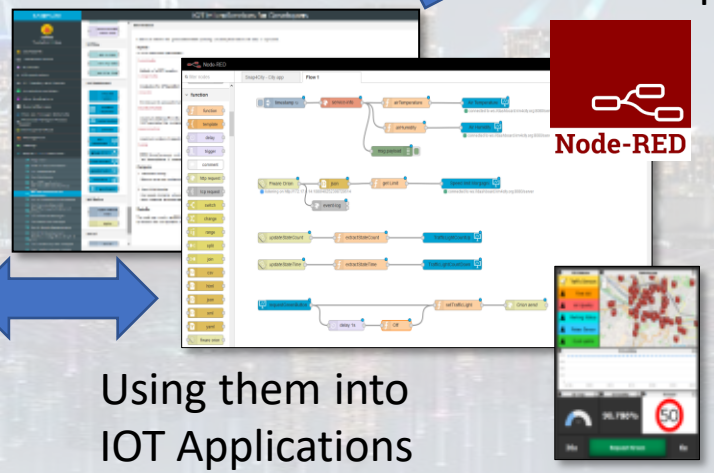
python

jupyter

R Studio



Creating MicroServices



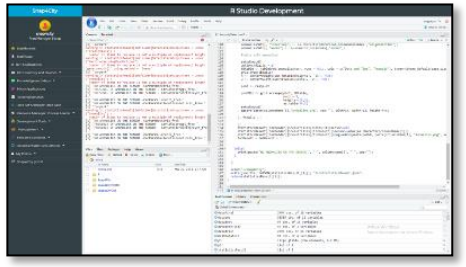
Node-RED

Using them into IOT Applications

Saving / Sharing reusing

Resource Manager

Development

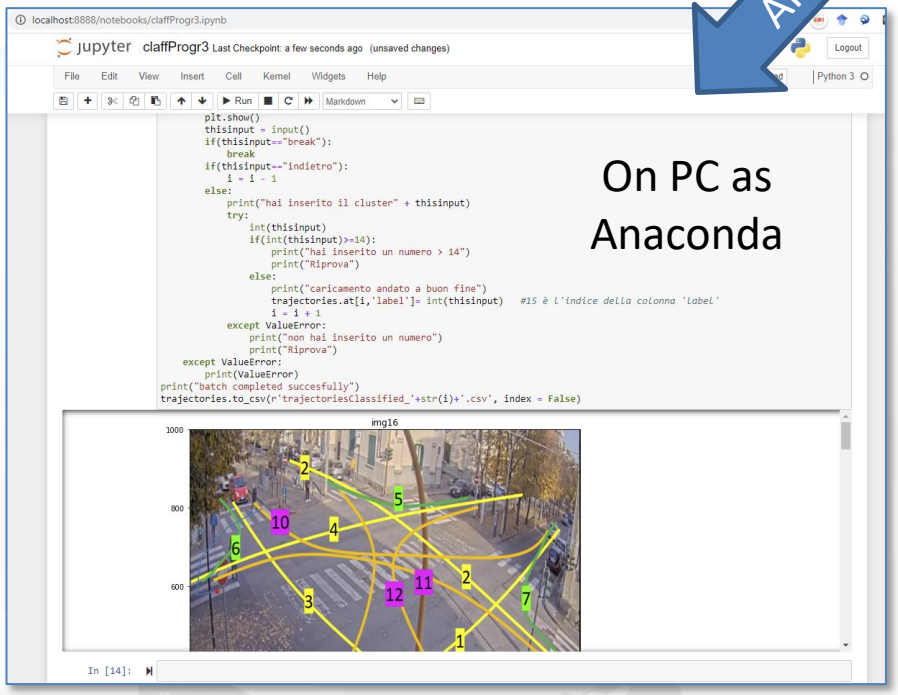


API

Big Data Store Facility

API

On Server
Or
On PC



On PC as Anaconda

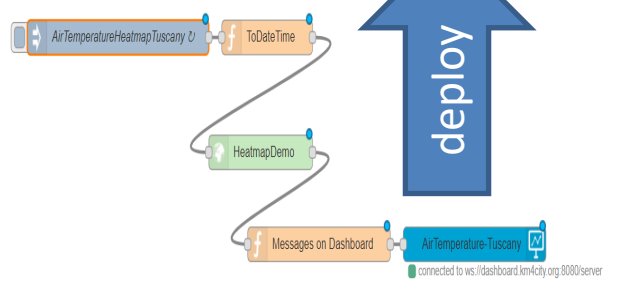
Once finalized

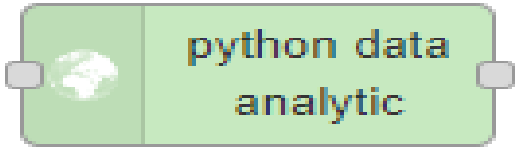
File.py
AI Model
Mapping
Data..

ZIP

Load
File.py
or .zip

deploy





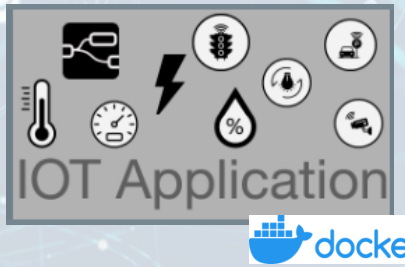
To make the .PY usable as MicroService you need to adapt it to get and send data in/out with Node-RED from a Container.
If you provide a .zip file the main .py inside has to be called doScript.py

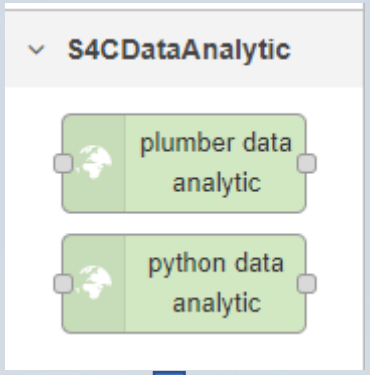
Data Analytic Container

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

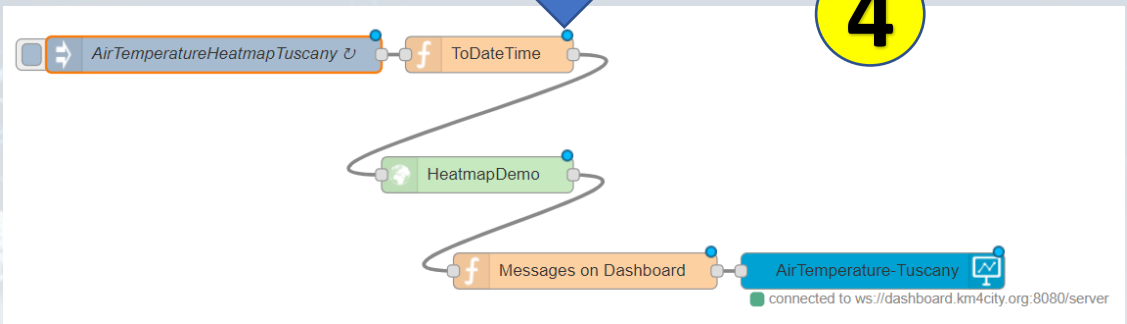




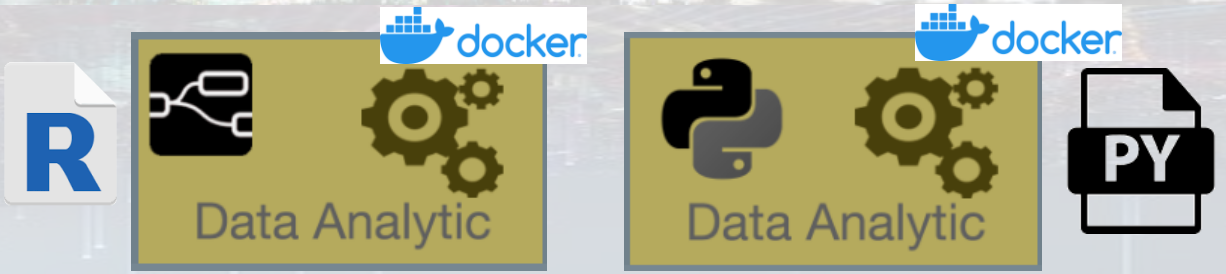
3

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

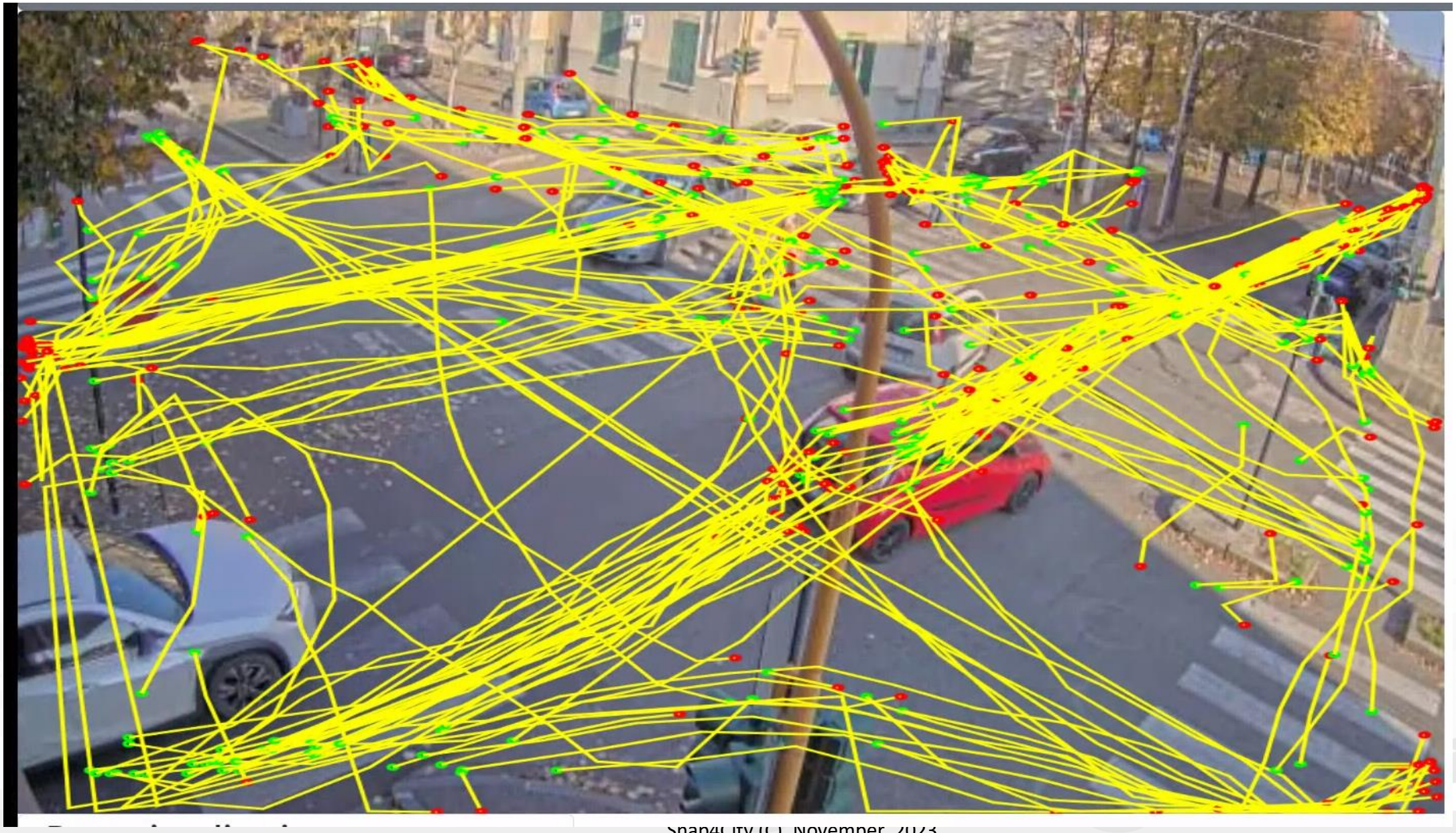
4

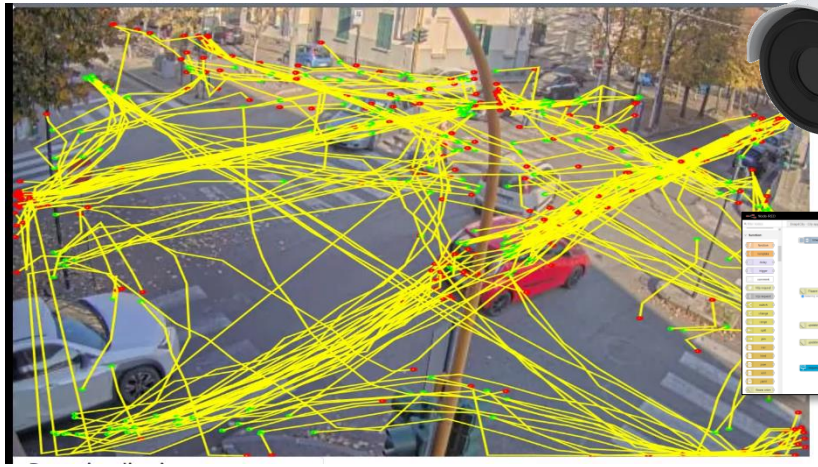


5 Deploy the IoT App → Snap4City Container Manager based on Marathon/Mesos is creating a Container for your Data Analytic code



analytics example





IoT edge on TV Camera

1

Send data to Broker

Send Trajectories

2

IOT Broker

Device: CrossVenaria2 with trajectories

Save data

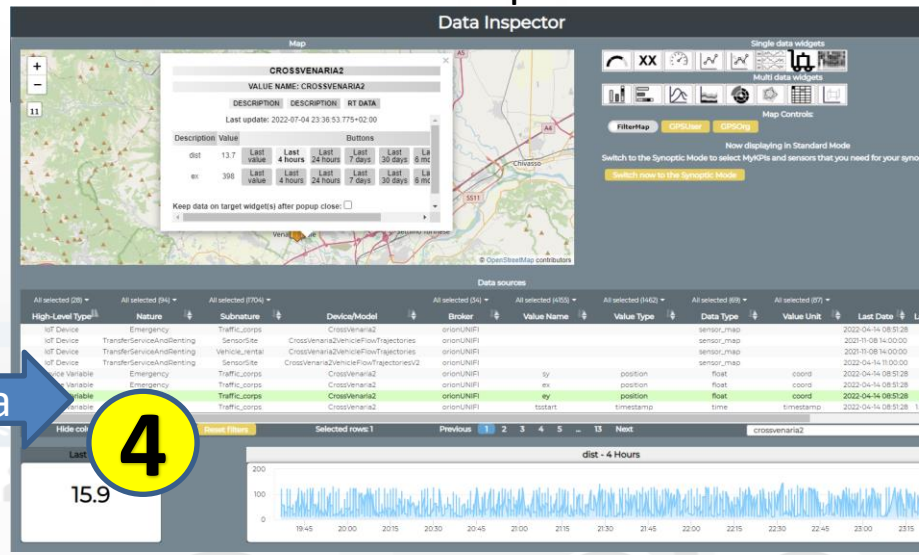
3

Big Data Store Facility

show data

4

Data Inspector



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

AXIS COMMUNICATIONS

SNAP4CITY

KM 4 CITY

IoT edge on TV Camera

Send Trajectories

Send data to Broker

IOT Broker

SNAP4CITY

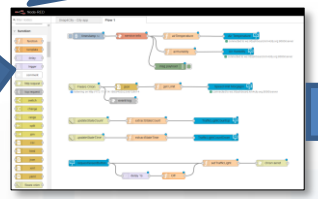
Devices:
- CrossVenaria2VehicleFlowTrajectoriesV2
- VenariaConteggio

Save data

f Save Counting per Cluster

e Send data to Broker

Periodically



b Activate

a

python data analytic

Data **d** lytic

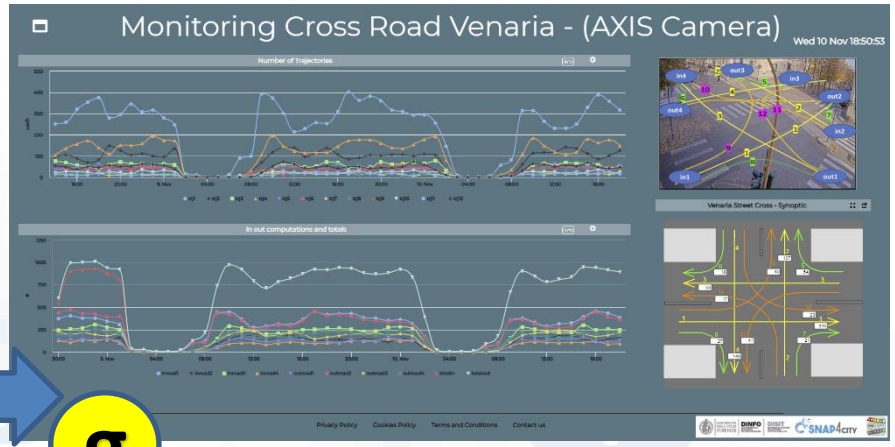
c Get data

From Trajectories to clusters. Counting in/out and flows

Device: CrossVenaria2 with trajectories

Big Data Store Facility

show data



g Create and use a Dashboard

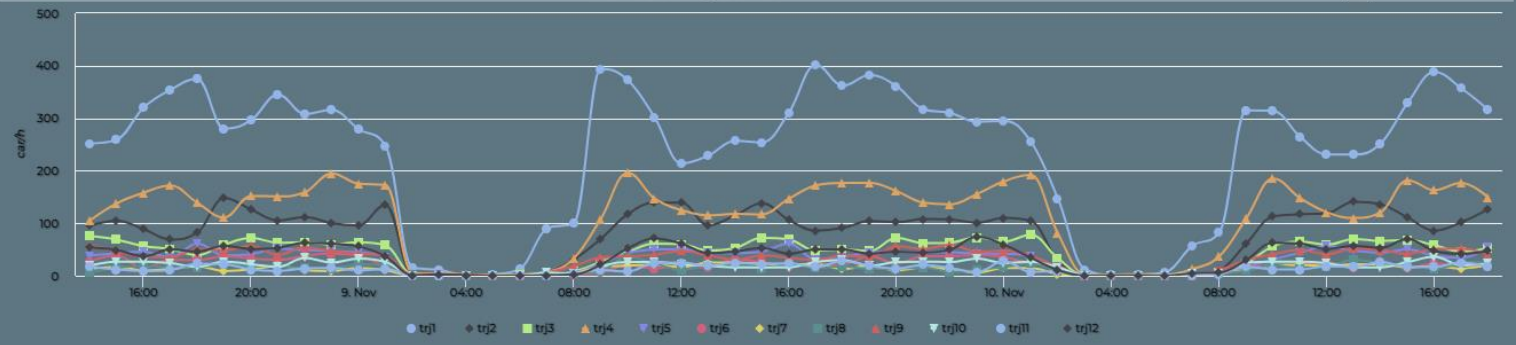
An example

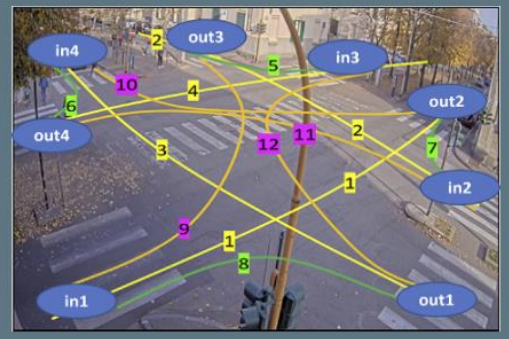
Traffic Flow Analysis via TV Camera and Clustering on cloud

Monitoring Cross Road Venaria - (AXIS Camera)

Wed 10 Nov 18:

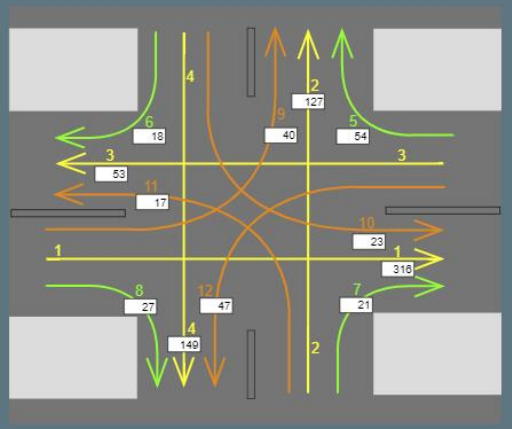
Number of Trajectories





In out computations and totals

Venaria Street Cross - Synoptic



Privacy Policy Cookies Policy Terms and Conditions Contact us

Managing TV Cam

IOT Devices Registration

IOT Directory

triples

Knowledge Base
Semantic Reasoners

SURI Link

push

IOT Orion Broker

subscription note

Real Time

Indexing and Aggregating
NIFI, OpenSearch

Processing if needed

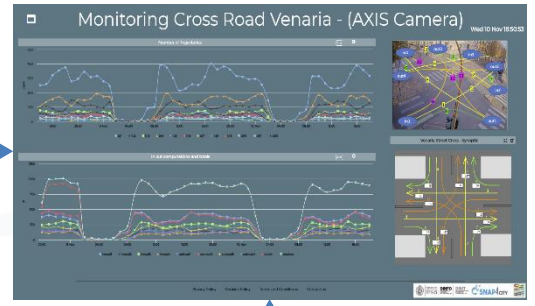
IOT App

Real Time data in Pull, Push
Any gateway, server

Data Analytics

Data Analytic

MQTT



Data Analytics on Dedicated Computer or HPC

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

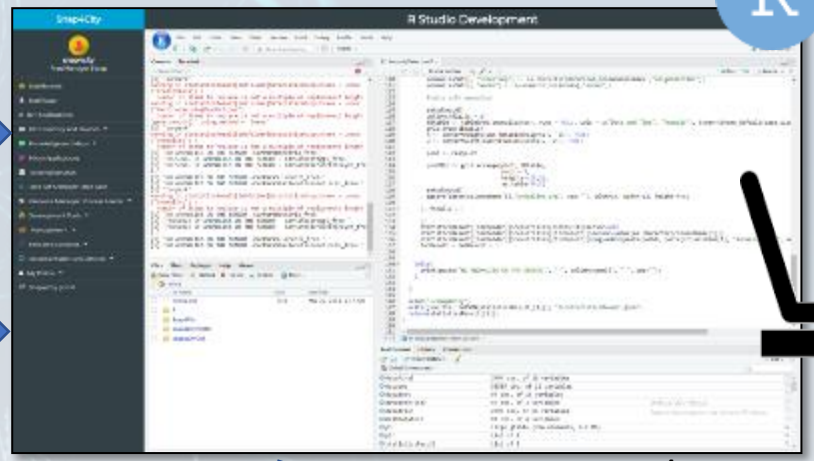
TensorFlow

NVIDIA CUDA

python

jupyter

RStudio



Saving / Sharing reusing

TensorFlow

NVIDIA CUDA

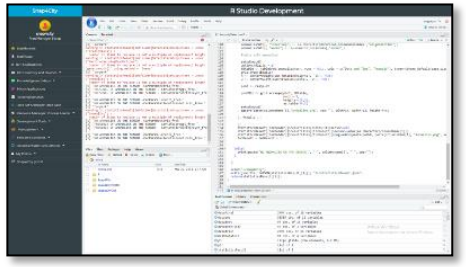
- Python *file.py* .....
- Rscript *file.r* .....

Resource Manager

SNAP4CITY

KM4CITY

Development



API

SNAP4CITY

Big Data Store Facility

Save data

Get data

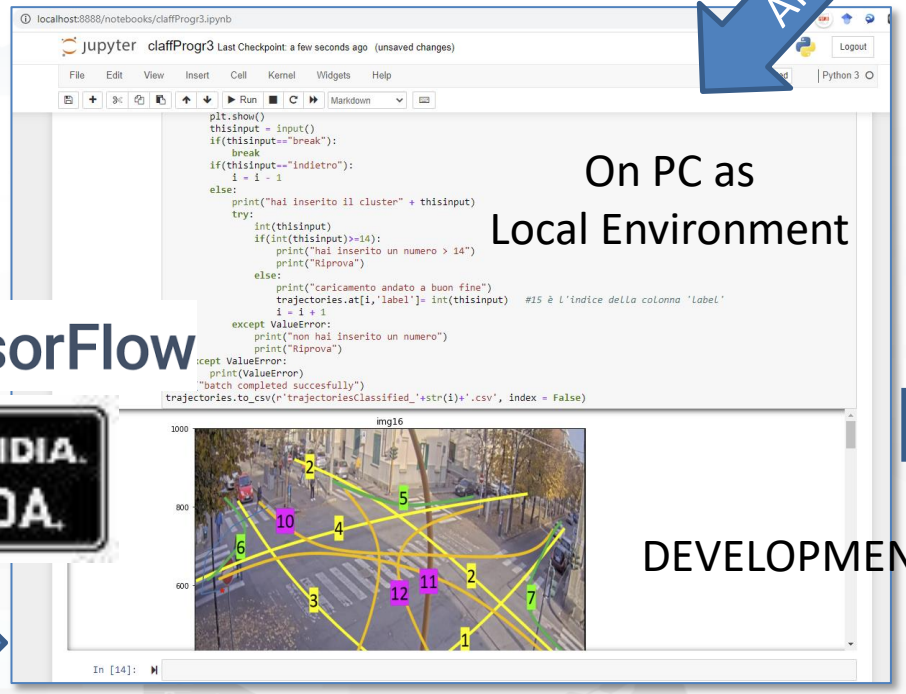
EXECUTION

TensorFlow

NVIDIA CUDA

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

On Server Or On PC



API

On PC as Local Environment

TensorFlow

NVIDIA CUDA

DEVELOPMENT

Once finalized

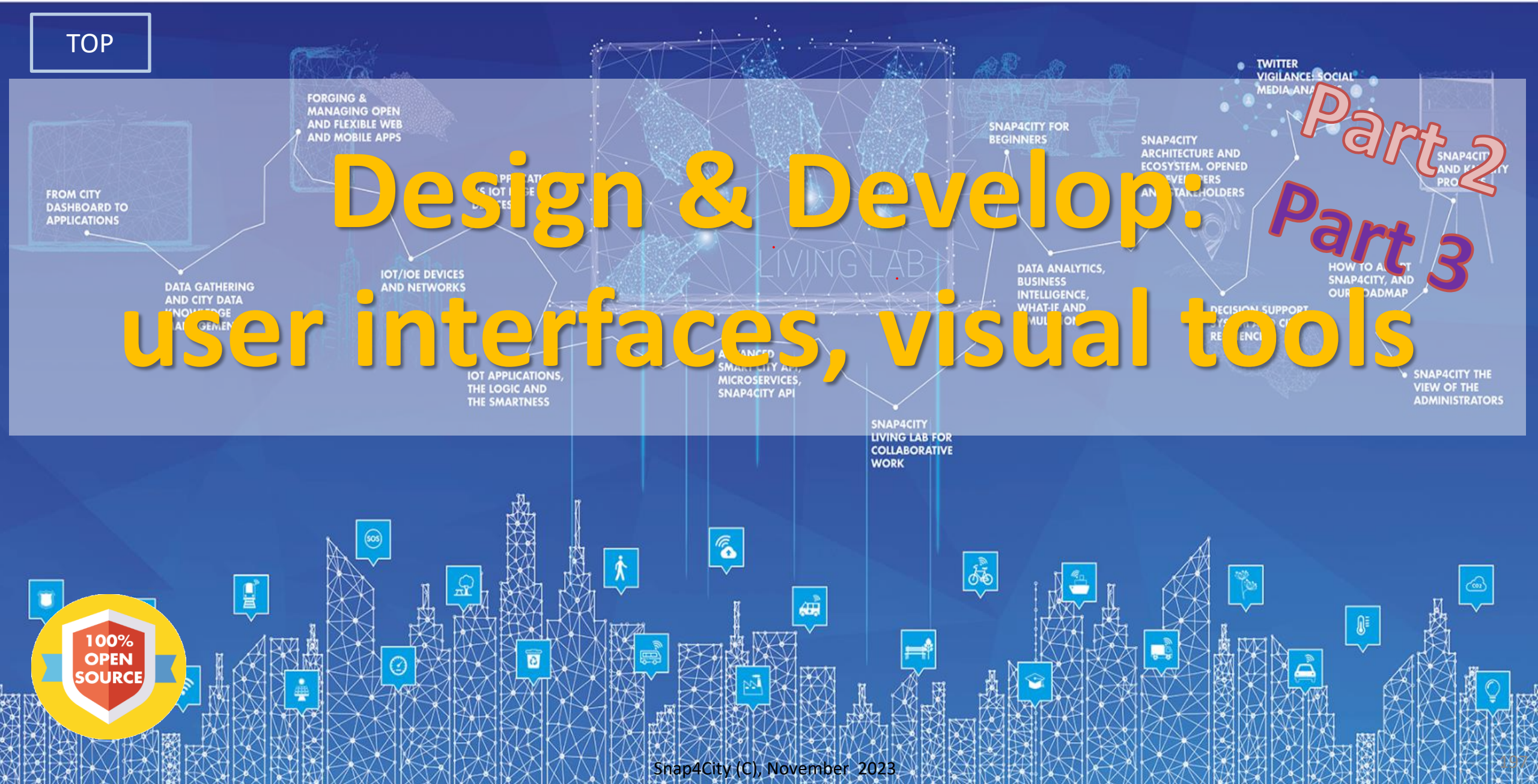
Process: file .R or .Py (+ the AI/ML model, data) can be put in execution with local scheduler or Cron

TOP

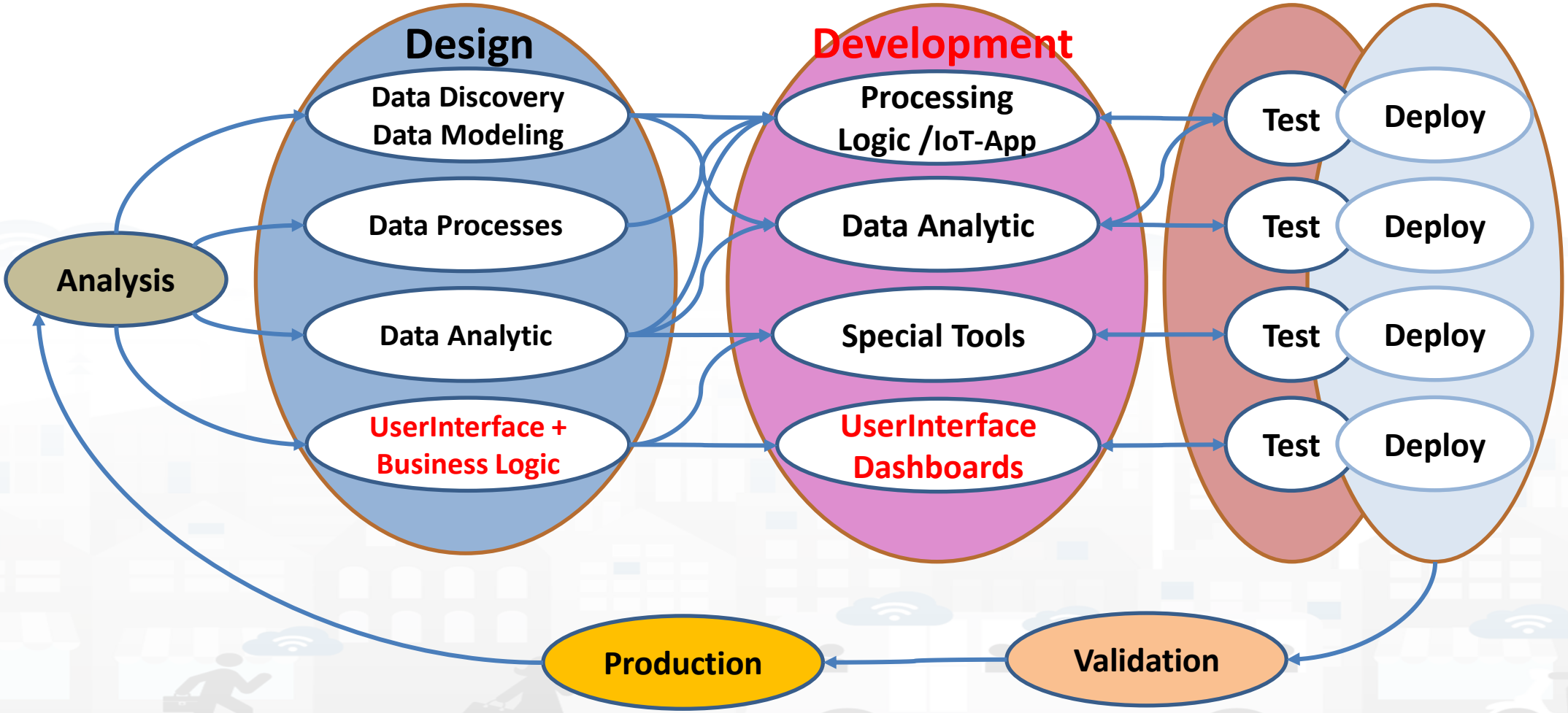
Design & Develop: user interfaces, visual tools

Part 2

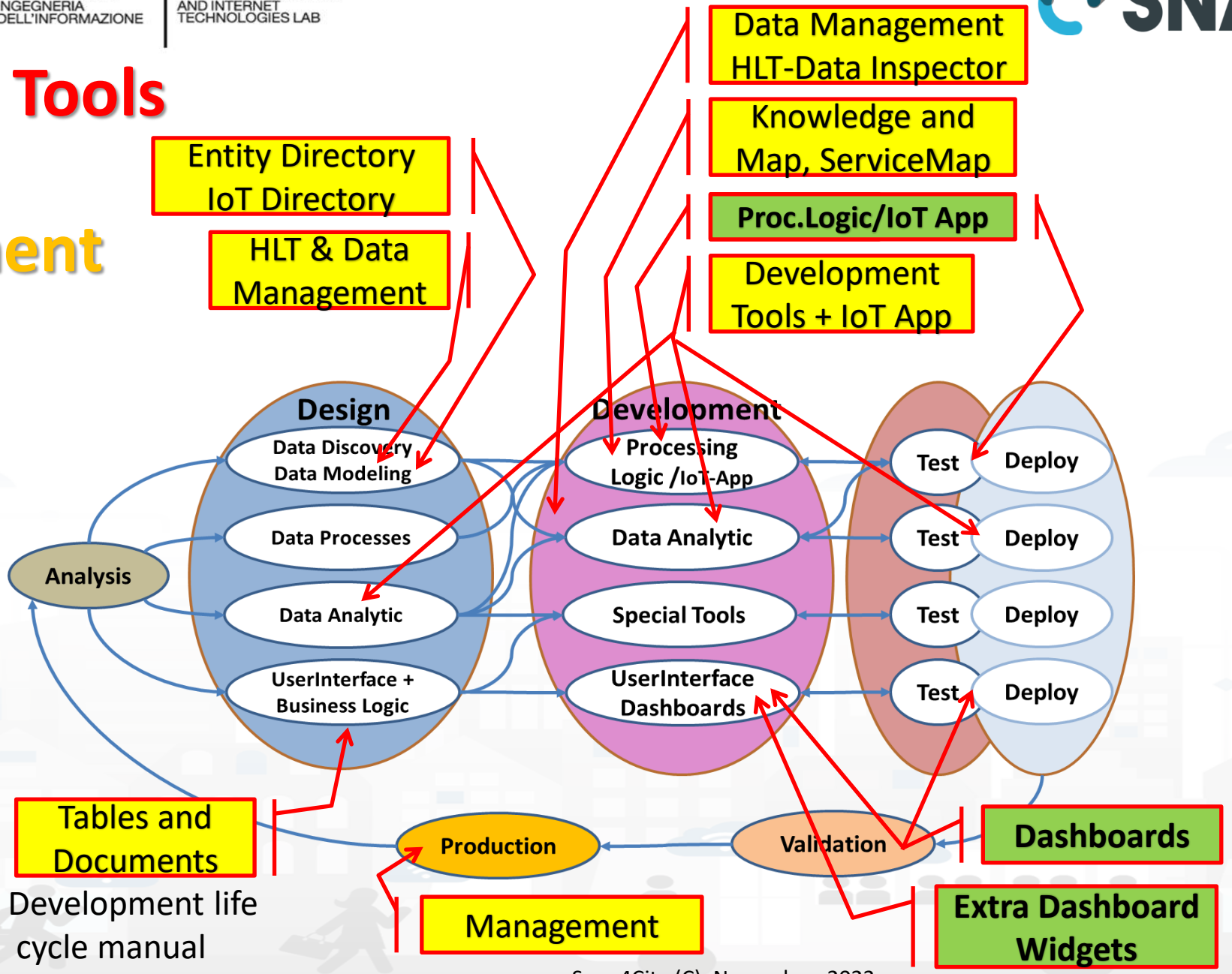
Part 3



Development Life Cycle Smart Solutions

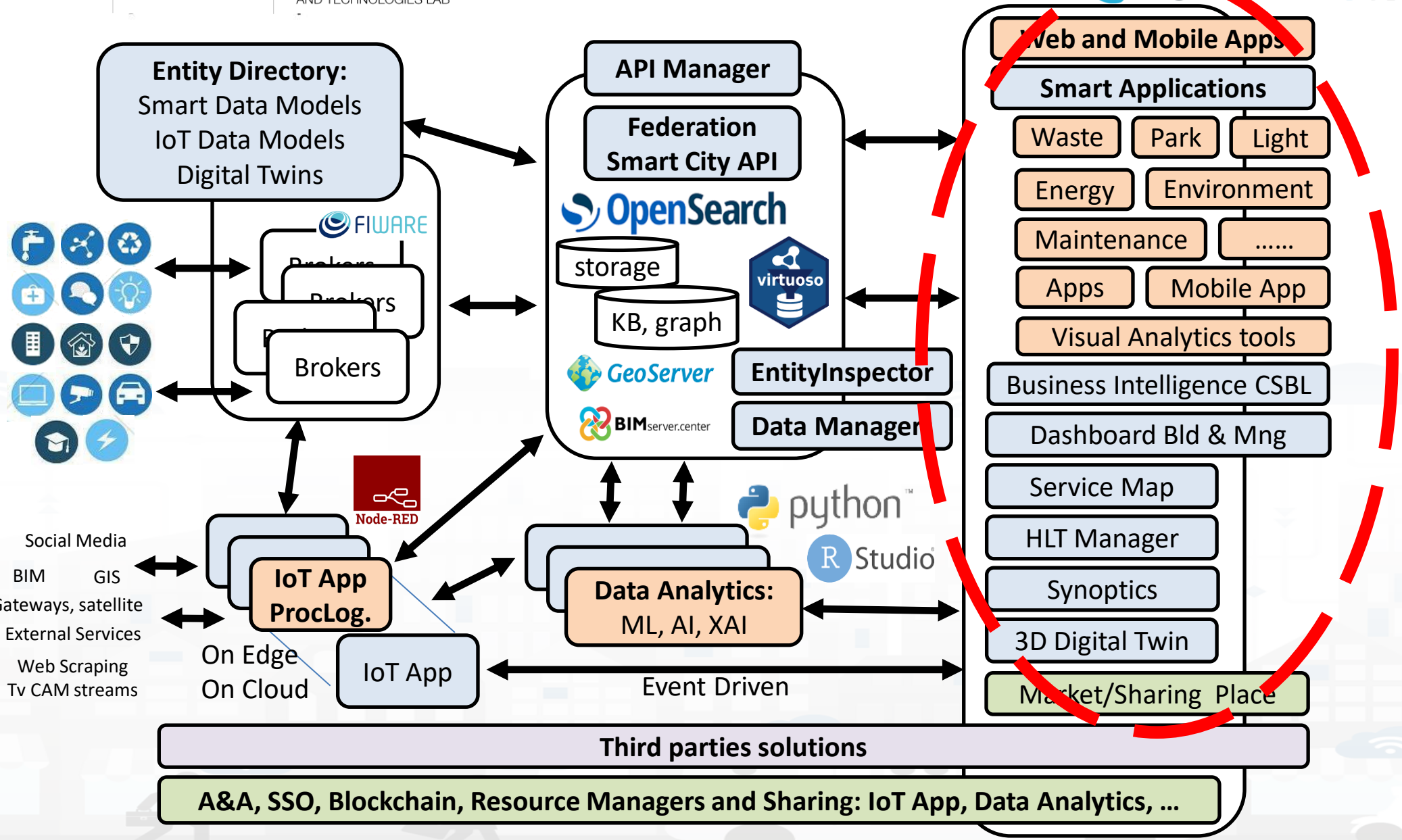


Snap4City Tools vs Development Life Cycle



Development life cycle manual

Tech Arch


Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIMserver.center
Data Manager
Node-RED
python
RStudio
Social Media
BIM
GIS
Gateways, satellite
External Services
Web Scraping
Tv CAM streams
IoT App
ProcLog.
On Edge
On Cloud
IoT App
Data Analytics:
ML, AI, XAI
Event Driven
Web and Mobile Apps
Smart Applications
Waste
Park
Light
Energy
Environment
Maintenance
……
Apps
Mobile App
Visual Analytics tools
Business Intelligence CSBL
Dashboard Bld & Mng
Service Map
HLT Manager
Synoptics
3D Digital Twin
Market/Sharing Place
Third parties solutions
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, …

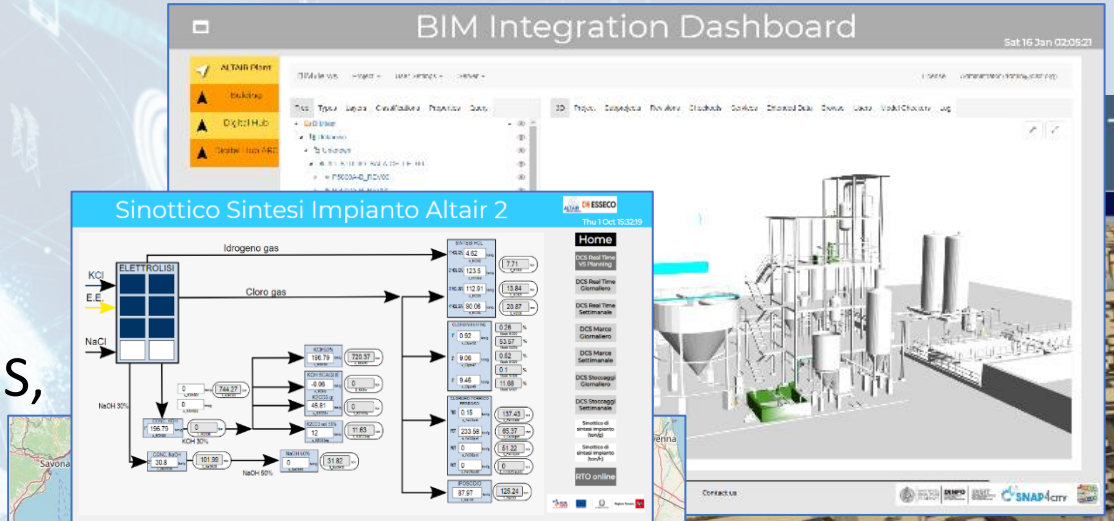
Snap4City Dashboard Builder (2023) vs Kibana/Grafana

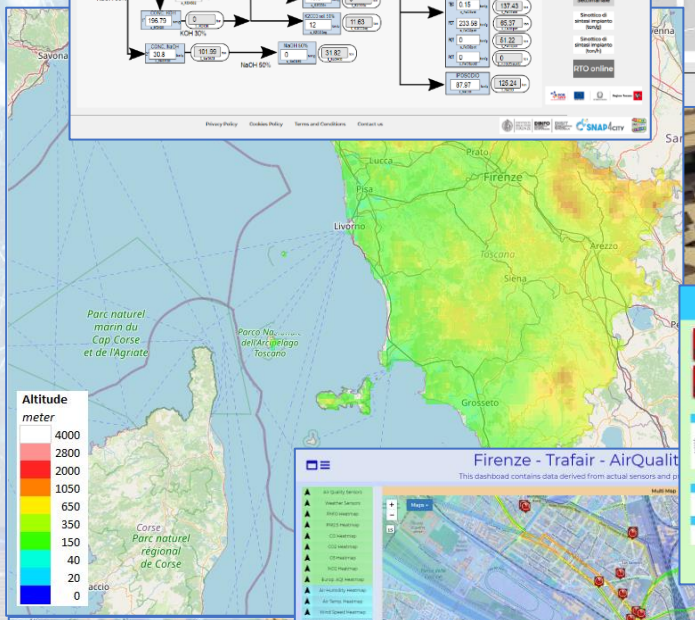
Features	Snap4City Dashboard Builder	Kibana, Grafana
Large Collection of Widgets, also from D3 library	YES	Nothing
Custom Widgets SVG of any kind, full defined process for customization	YES	Nothing
Real time event driven widgets and data	YES	Nothing
Server/Client Side Business Logic for data transformation with visual programming: Node-RED	YES: visual/coding	coding
Maps with custom PIN, bubbles, animated and moving, etc.	YES	Nothing
Maps with paths, shapes, traffic flow, scenarios, routing, heatmaps, what-if, Origin Destination Matrix, …	YES	Nothing
Maps with Orthomaps from WFS, WMS, GIS connection, etc.	YES	Nothing
TV camera integration and selection	YES	Nothing
Widgets for business logic integration on real time: buttons, selector, switch, etc.	YES	Nothing
Kiviat, Spider net, Calendar (also any other D3 Widgets)	YES	Nothing
Typical Time Trends: day hours, month week, month days, …..	YES	Nothing
Time Trend Compare: day, week, month, year	YES	Nothing
Selectors/Menus: text, icons, etc., also in connection with IOT APP, Node-RED	YES	Nothing
Full control of graphic layout, font, colours, refresh per widget, etc.	YES	Nothing
Iframe integration of third party widgets and web pages, nesting dashboards, embedding Kibana	YES	Nothing
Connection among multiple Dashboards and Widgets	YES	Nothing
Synchronization with Video Wall, and Operators Views	YES	Nothing
Multiseries, bar lines, charts, pie, donut, simple selectors, trends, etc., also from business logic	YES	Limited
Single content, string, html, any data, etc.	YES	Limited
Special widgets: Weather forecast, civil protection, road plates, Twitter, SVG, etc…	YES	Nothing
Digital Twin Local (BIM) and Global (3D city representation) with 3D traffic, Heatmaps, Devices, …	YES	Nothing
Faceted search	YES: selectors, forms, buttons	YES

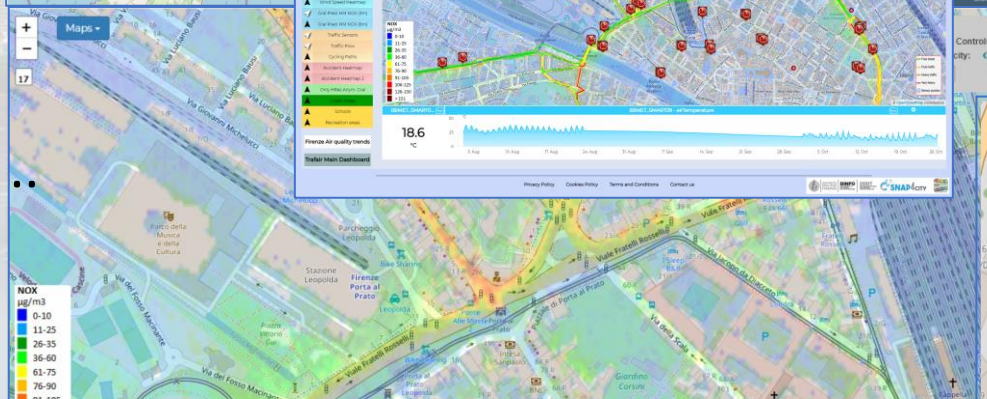
High Level Types

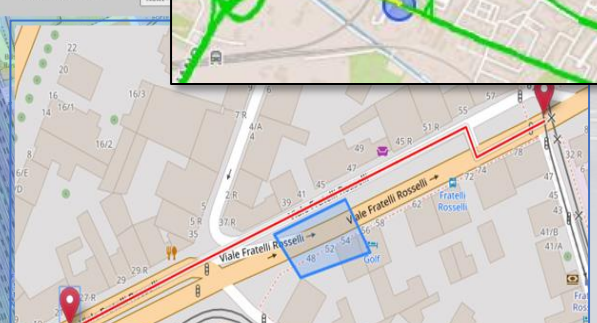
Snap4City (C), November 2023

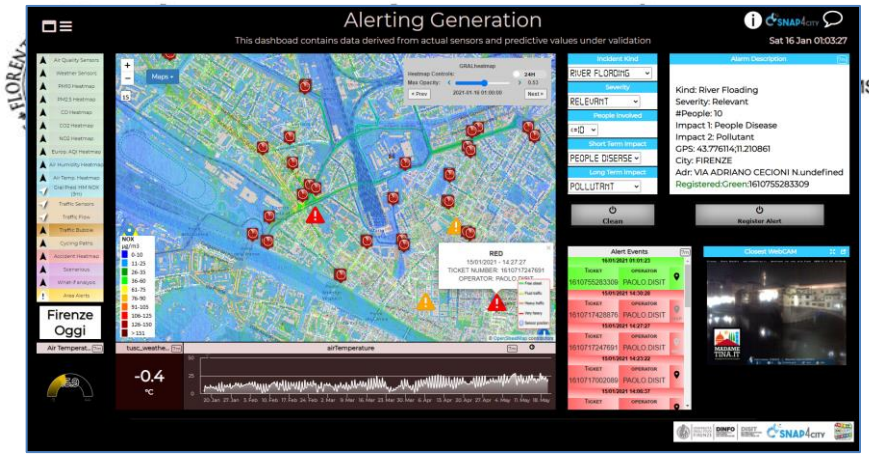
- POI, IOT Devices, shapes,..
  - FIWARE Smart Data Models,
  - IoT Device Models
- GIS, maps, orthomaps, WFS/WMS, GeoTiff, calibrated heatmaps, ..
- Satellite data, ..
- traffic flow, typical trends, ..
- trajectories, events, Workflow, ..
- 3D Models, BIM, Digital Twins, ..
- OD Matrices of several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

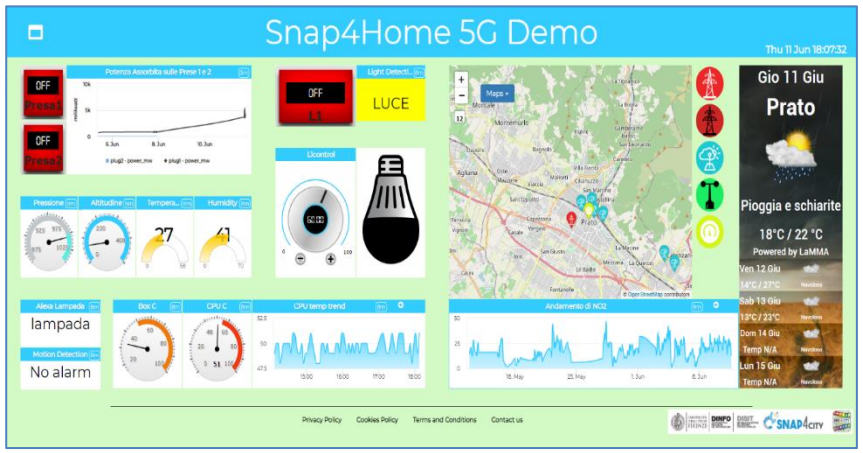


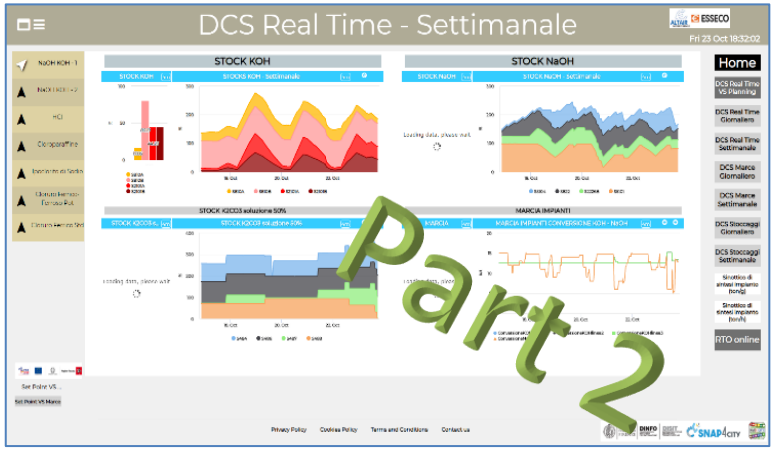


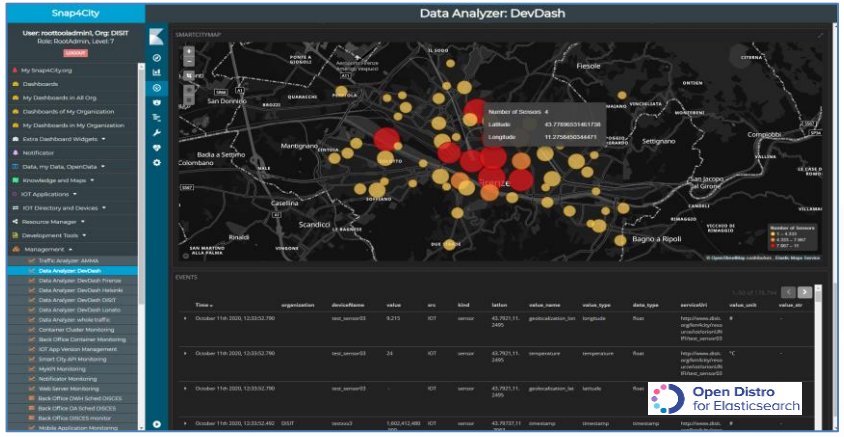


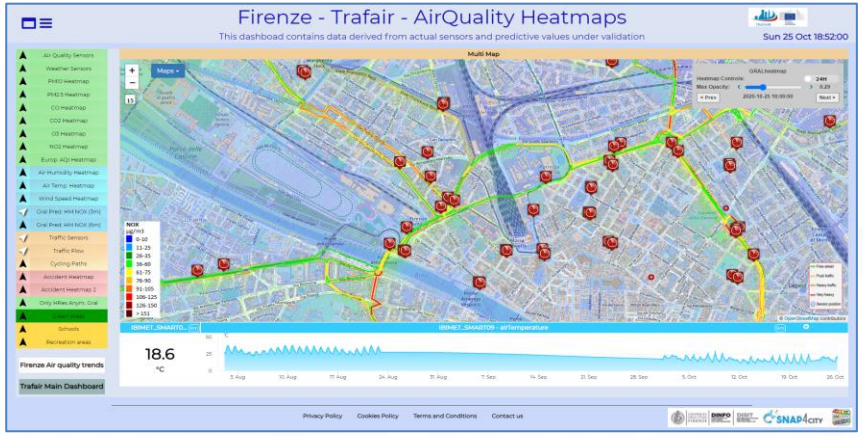


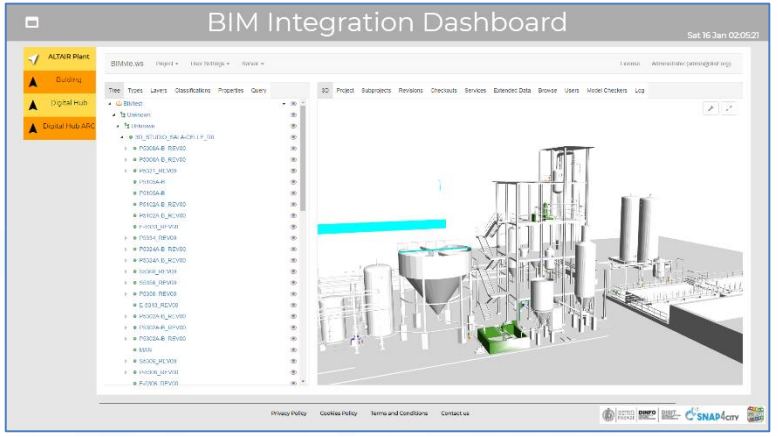

Alerting Generation
This dashboad contains data derived from actual sensors and predictive values under validation
Sat 16 Jan 01:03:27
Firenze Oggi
-0.4


Snap4Home 5G Demo
Thu 11 Jun 18:07:32
LUCE
lampada
No alarm
Gio 11 Giu
Prato
Pioggia e schiarite
18°C / 22 °C
Powered by LaMMA

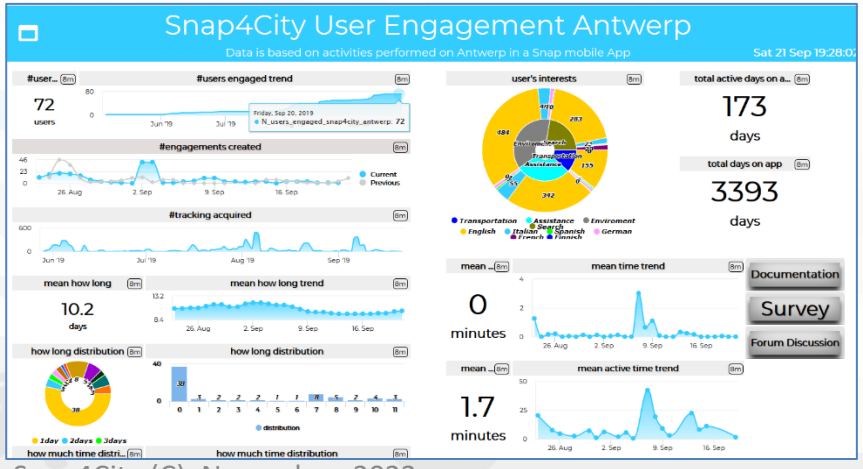

DCS Real Time - Settimanale
STOCK KOH
STOCK NaOH
Home
Part 2

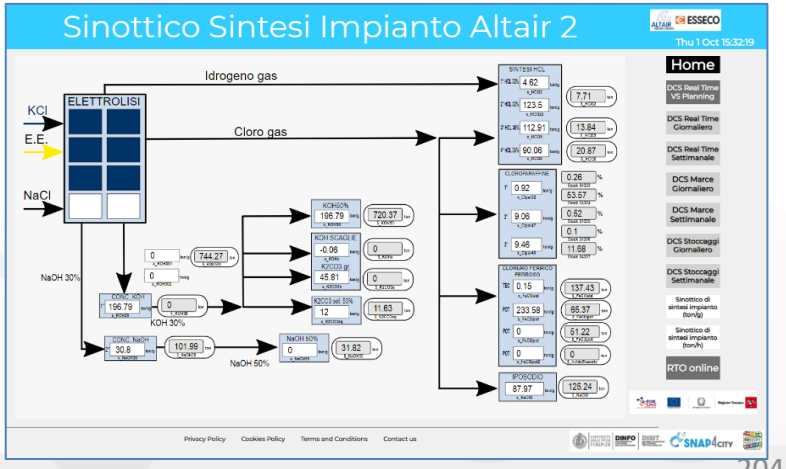

Data Analyzer: DevDash
Open Distro for Elasticsearch


Firenze - Trafair - AirQuality Heatmaps
This dashboad contains data derived from actual sensors and predictive values under validation
Sun 25 Oct 18:52:00
18.6

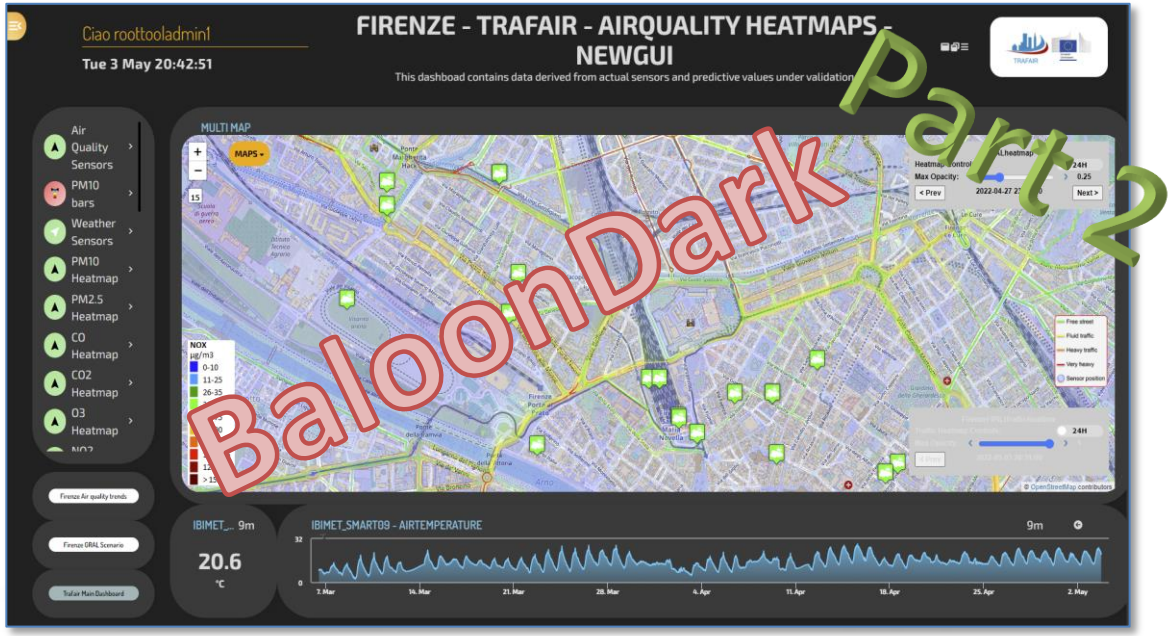

BIM Integration Dashboard
Sat 16 Jan 02:05:21

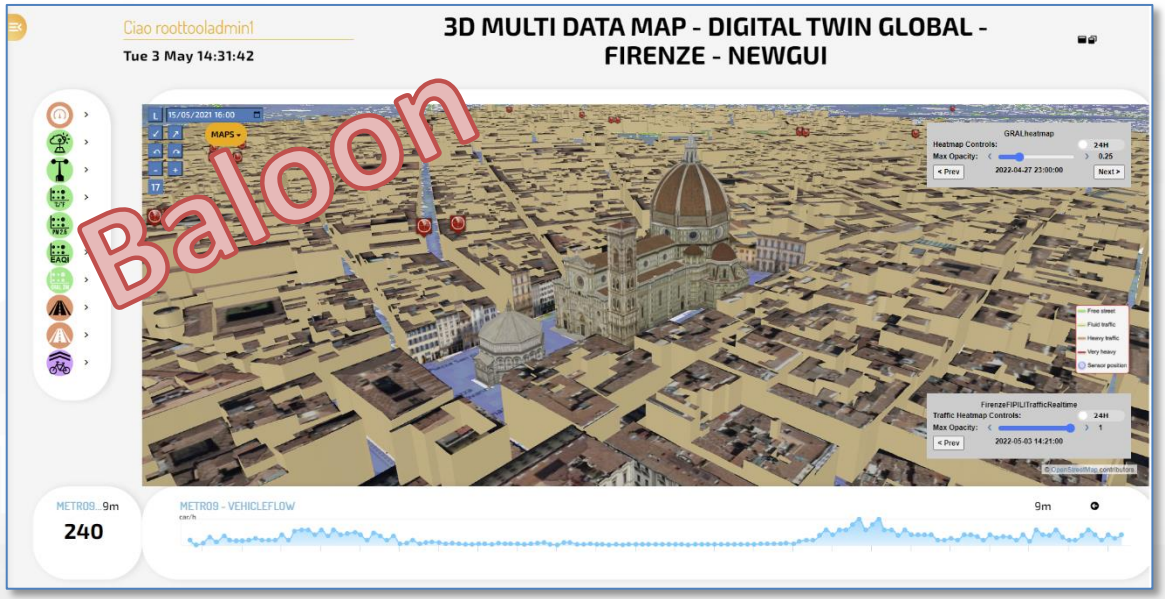
3D Multi Data Map - Digital Twin Global - Firenze
demonstrator
Sun 3 Apr 16:03:52
192

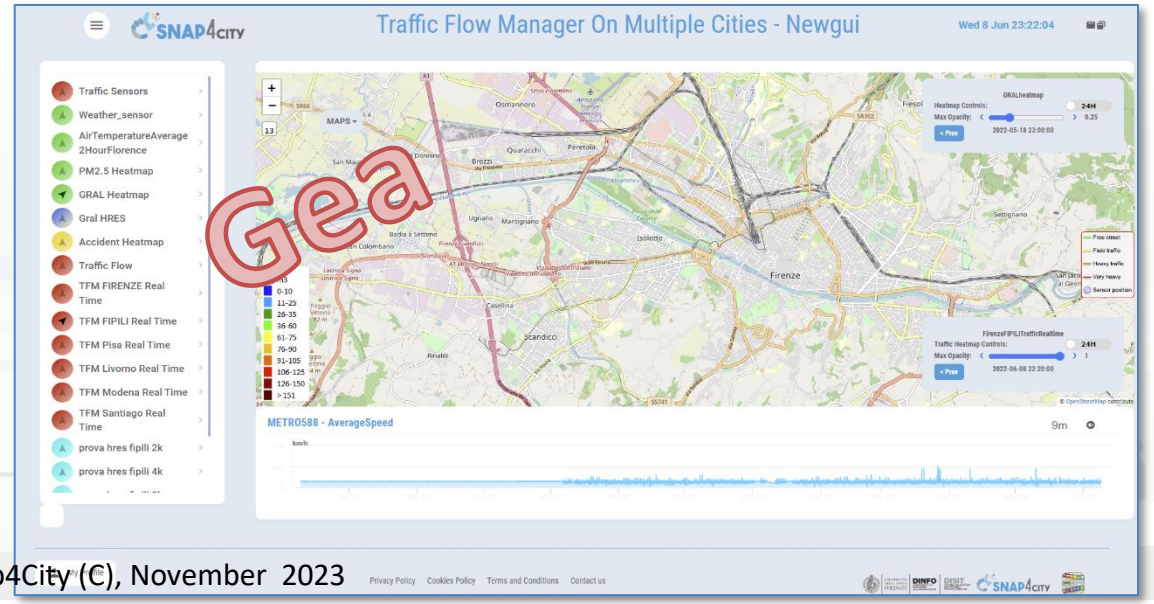

Snap4City User Engagement Antwerp
Data is based on activities performed on Antwerp in a Snap mobile App
72
users
173
days
3393
days
10.2
days
0
minutes
1.7
minutes
Documentation
Survey
Forum Discussion


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

Different Themes







Dashboards (Public by (ORG))

LOGIN

Switch to Legacy Layout

Part 2

- Dashboards (Public)
- www.snap4solutions.org
- Extra Dashboard Widgets
- Data Management, HLT
- Knowledge and Maps
- Processing Logics / IOT App
- Entity Directory and Devices
- Resource Manager
- Development Tools
- Management
- Decision Support Systems
- Deploy and Installation
- Help and Contacts
- Documentation and Articles
- Km4City portal

Prev 1 ... 34 35 36 37 38 Next

Filter by dashboard

Traffic Flow Manager test

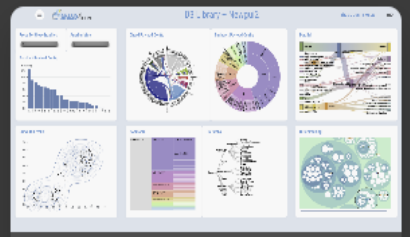
Passive

Public (DISIT)

Traffic Flow Monitoring - Firenze - Cloned2

Passive

Public (Firenze)



D3 library -- newgui2

Proc.Logic / IoT App

Public (DISIT)

Traffic Flow Reconstruction - Sii-Mobility

Passive

Public (Other)

Traffic Flow Reconstruction for the cities

Passive

Public (Other)

3D Map Global Digital Twin -newgui2

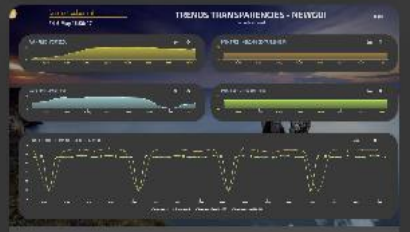
Passive

Public (DISIT)

3D Multi Data Map - Digital Twin Global - Firenze

Passive

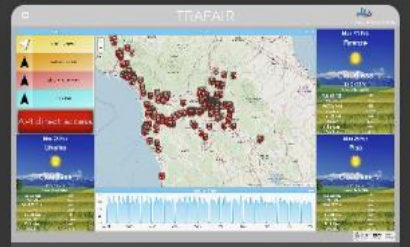
Public (DISIT)



Trends transparencies - newgui

Passive

Public (DISIT)



Tuscany TRAFAIR Data Dashboard

Passive

Public (DISIT)

Tuscany weather dashboard 1

Passive

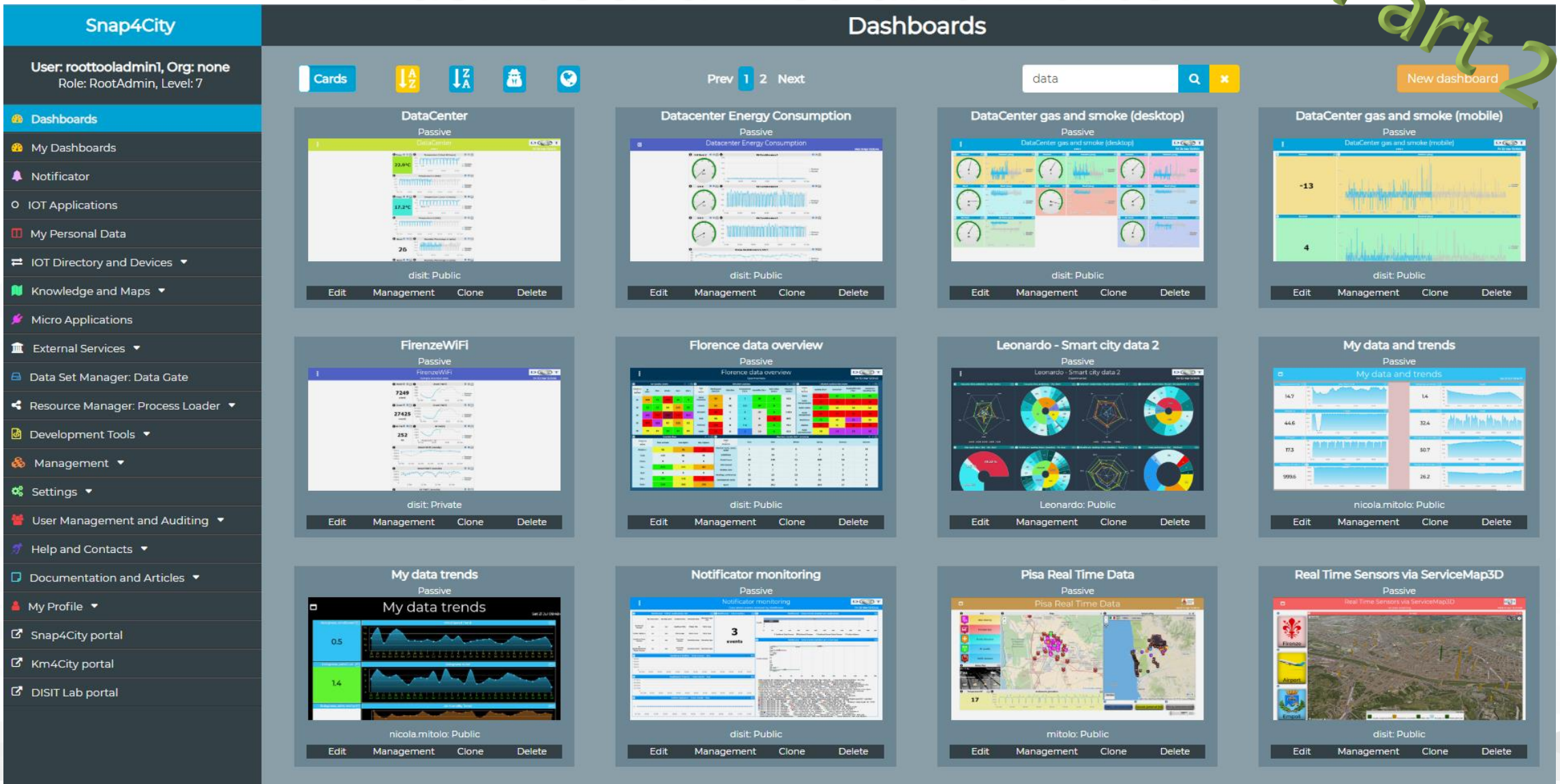
Public (DISIT)

Privacy Policy Cookies Policy

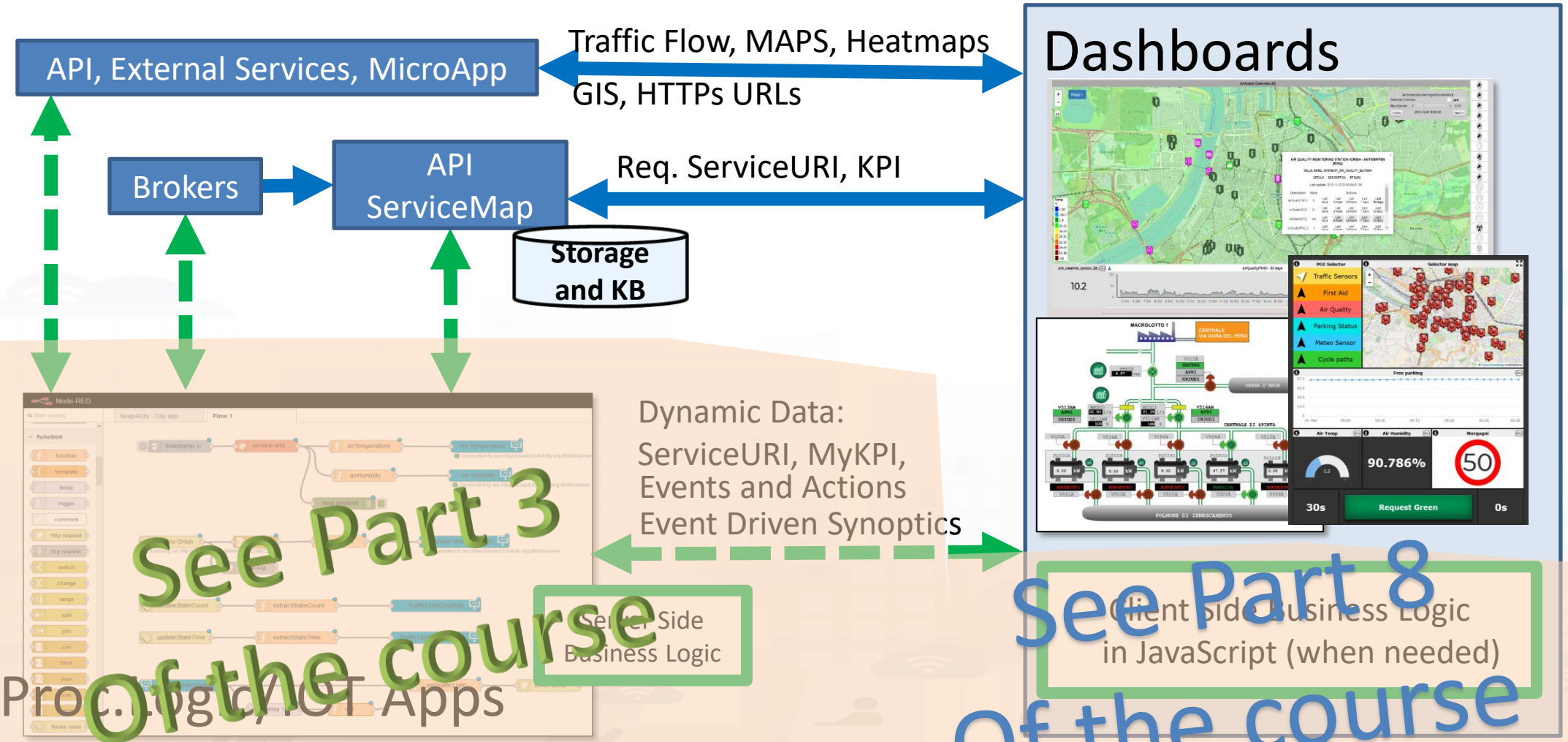
UNIVERSITÀ DEGLI STUDI FIRENZE DINFO DISIT

Dashboard List and Editor

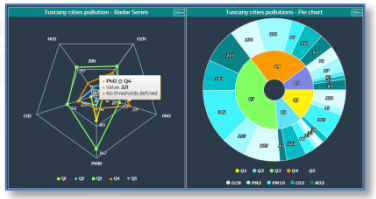
Part 2

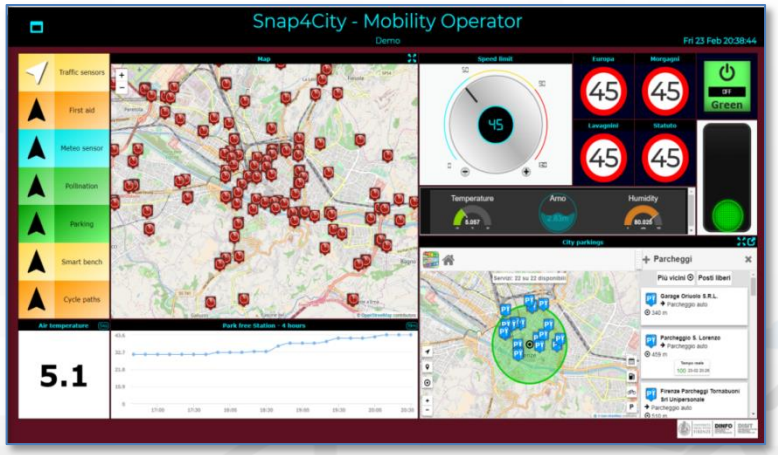


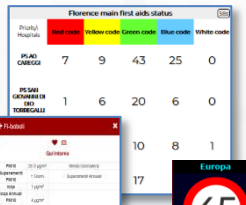
How the Dashboards exchange data

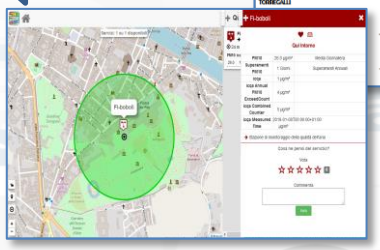


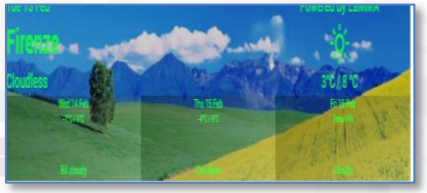
Dashboard List and Editor

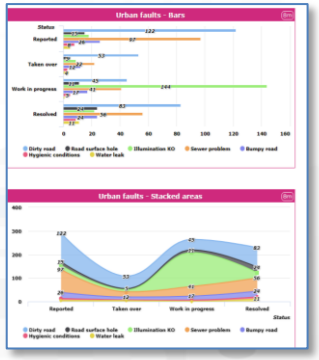




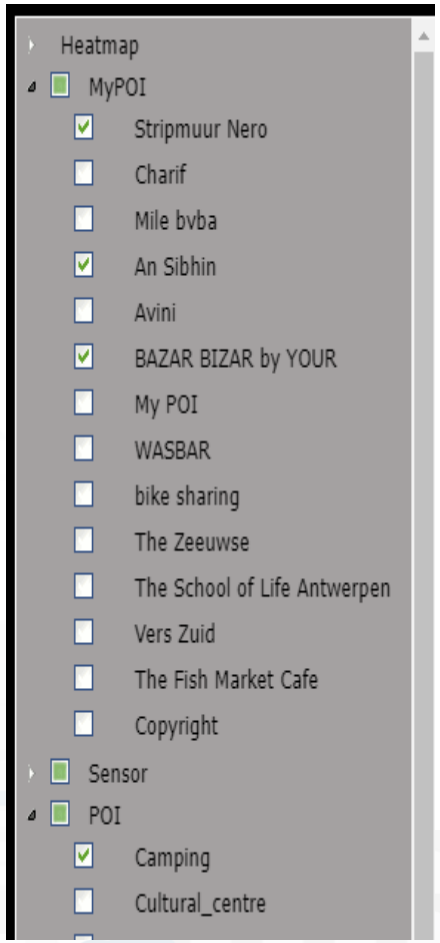


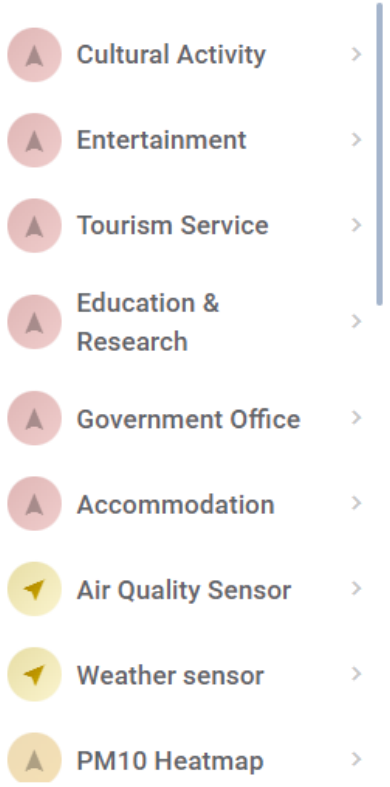






The Selector for Multi Data Maps

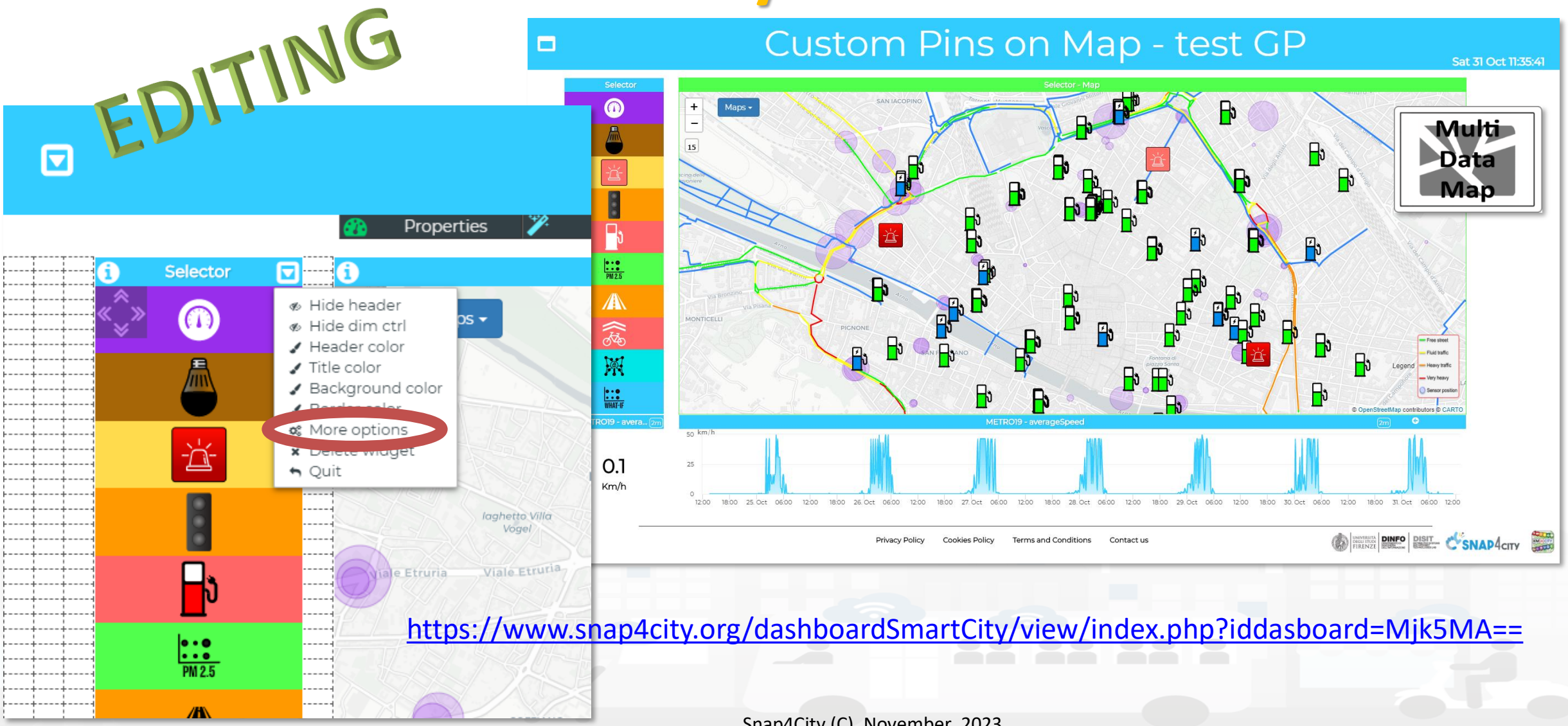




- **Different styles**
  - Icon and Text menu
  - Custom Menu Icon
  - Icon Menu buttons
  - Etc.
- **Features**
  - Removable header
  - Colours custom
  - Transparencies
  - Mixed modalities
- **Note:**
  - Manus can be realized also with a set of Buttons

The Selector is the Map Controller

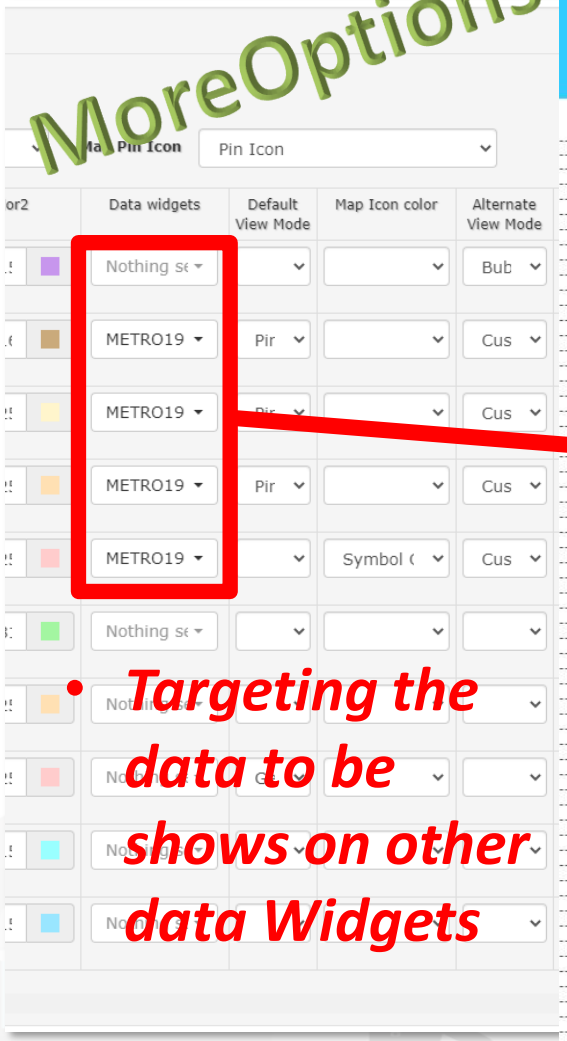
Custom Dynamic Pins



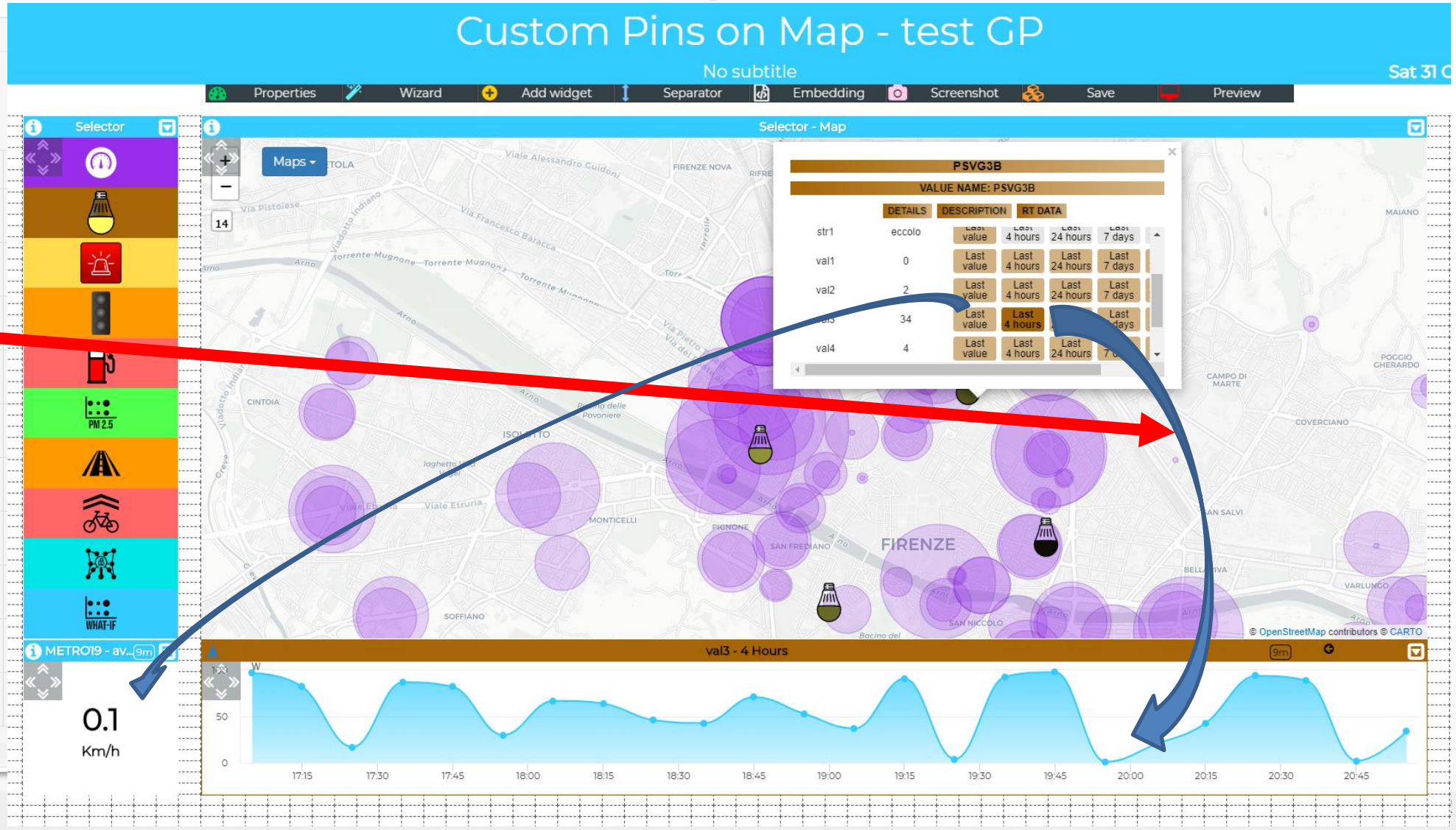
https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=Mjk5MA==

The Selector is the Map Controller

MoreOptions



- *Targeting the data to be shows on other data Widgets*



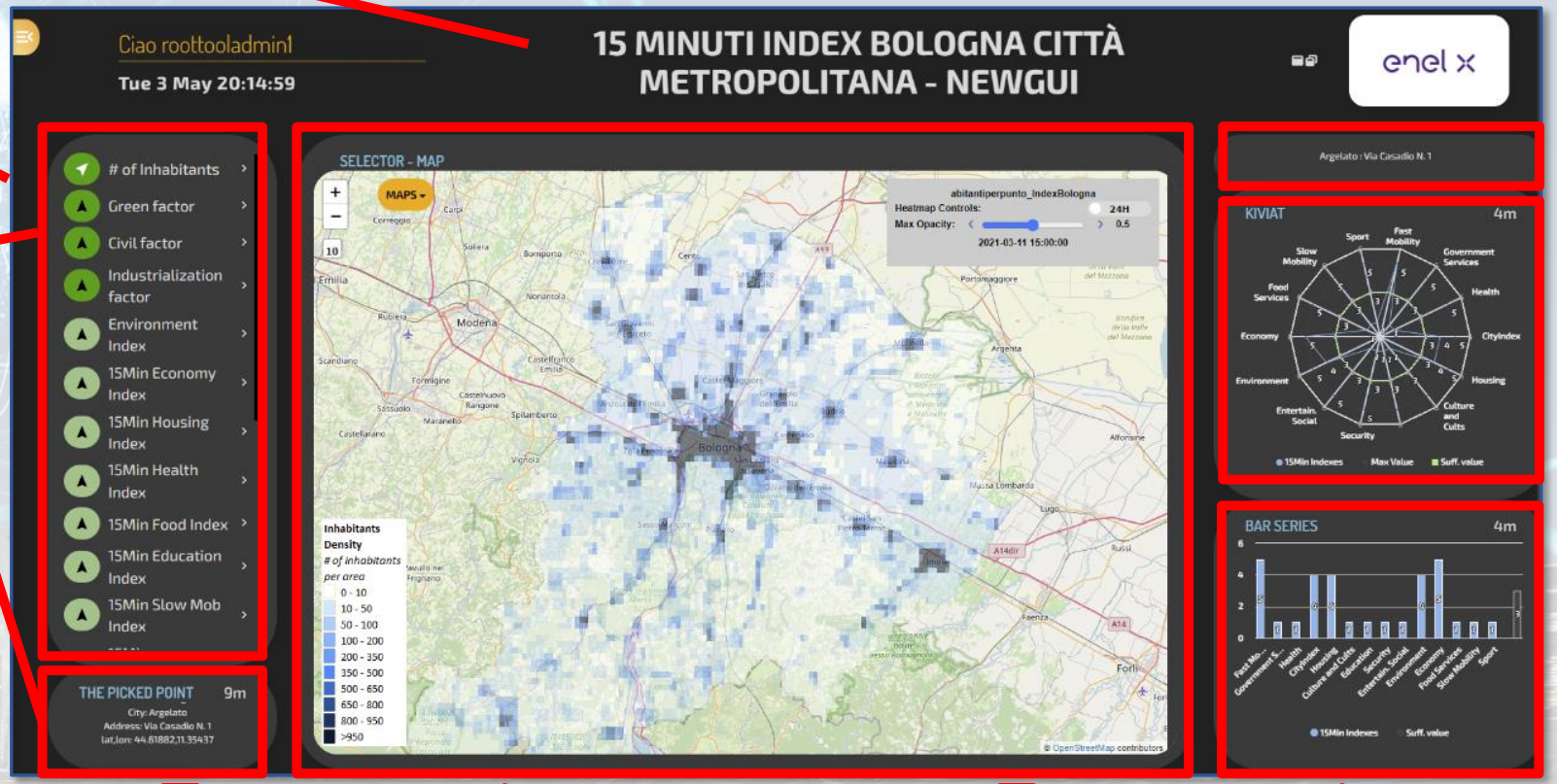
Develop Dashboard
Main Concepts

Snap4City Dashboards main concepts

Header

Dashboard

Interactive Widgets



Server Communication

- Real Time data requests/send
- Event Driven
- Server Side Business Logic
  - See Part 3 of the course

Inter Widget Communication:

Client Side Business Logic

See part 8 of the Course

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
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** • Description: 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: .....

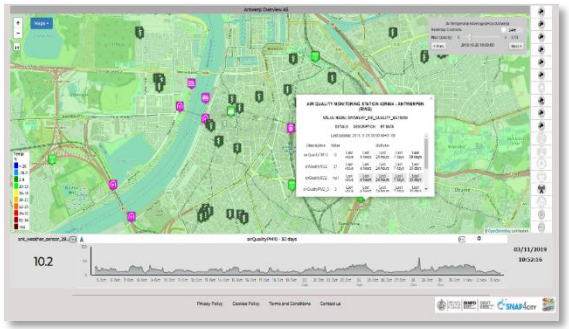
Server Side Business Logic	• **To be developed in IoT App with S4C Dashboard Nodes**
	• IoT Application →
	  ○ Event driven: .....
	• IoT Application →
	  ○ Event driven: .....

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

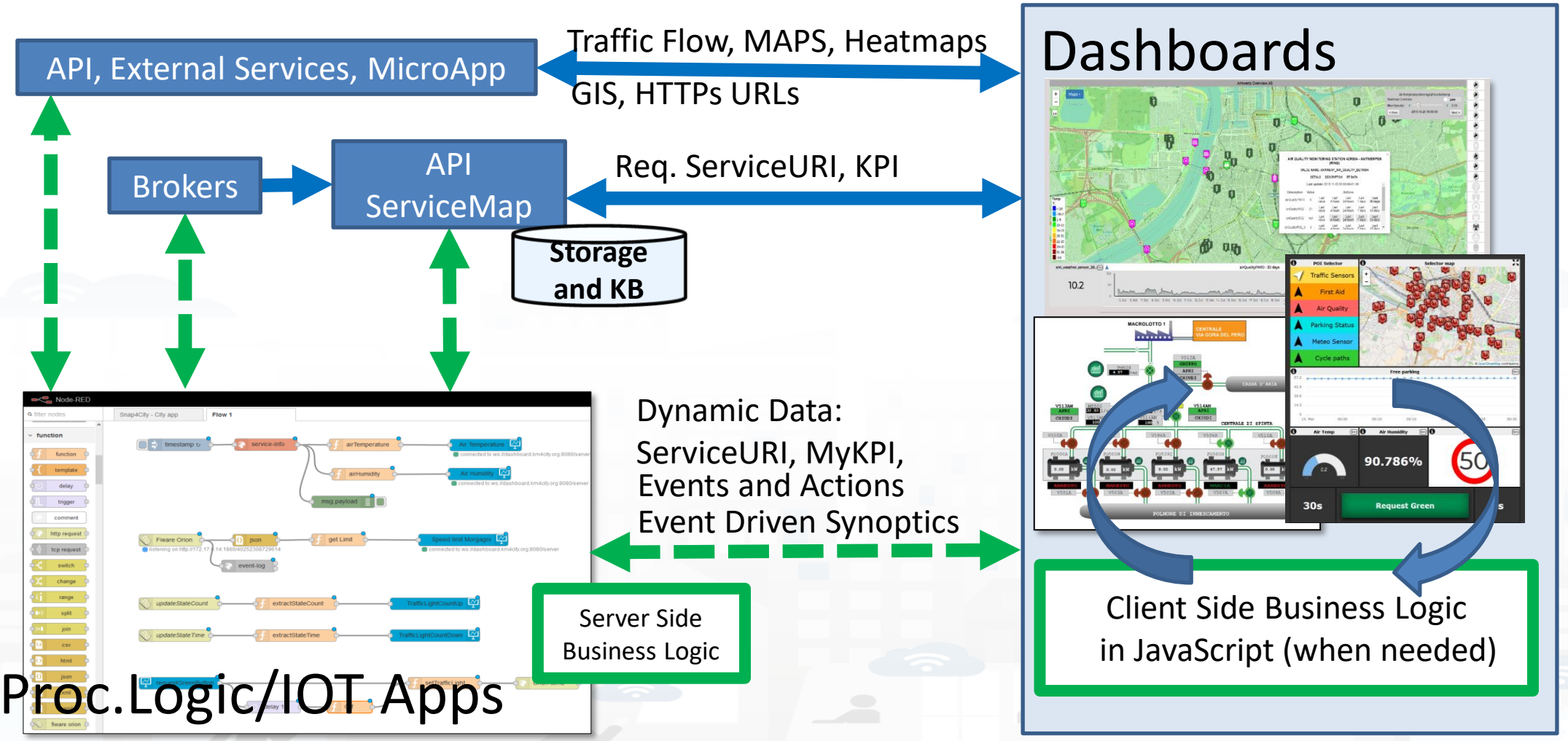
Dashboard Kind

- **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.

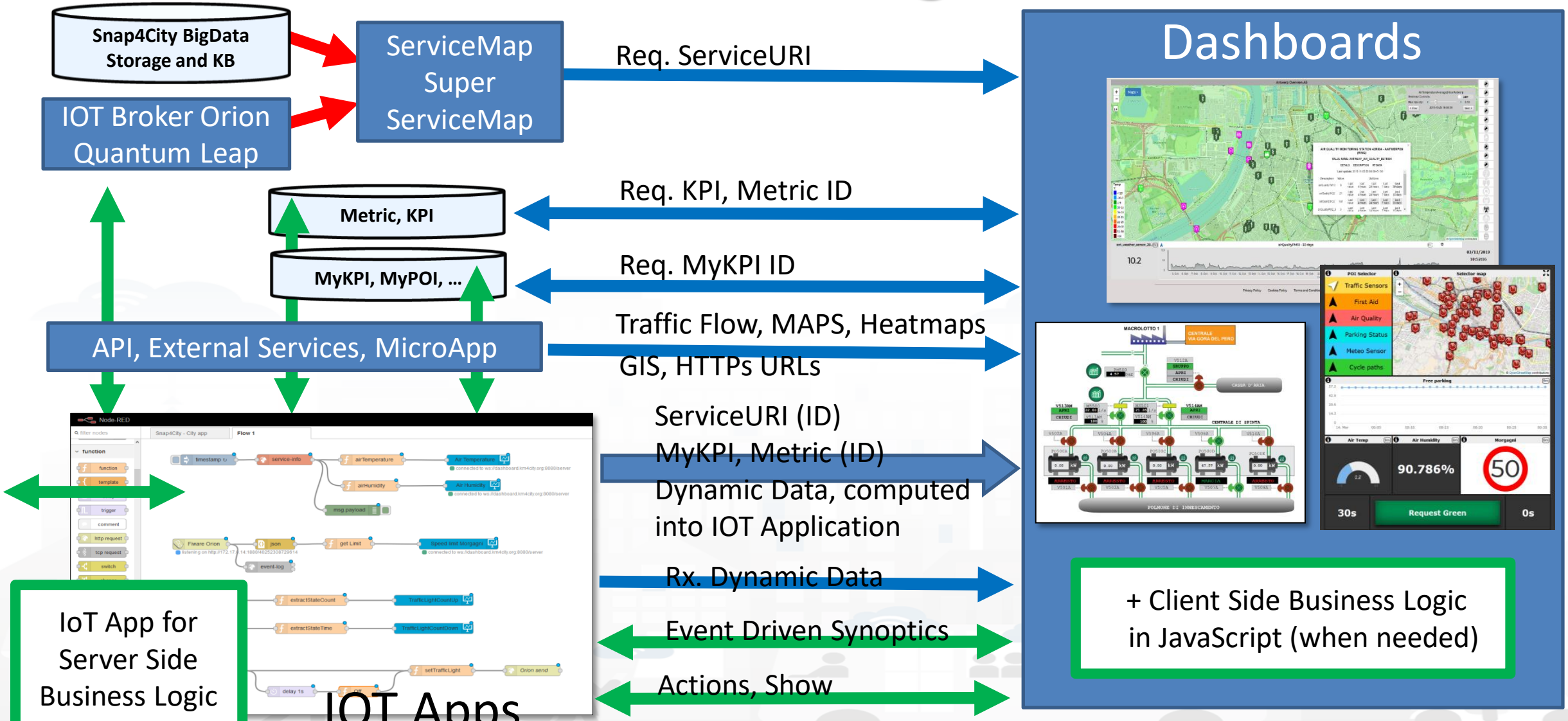


- **Custom Business Logic**
- **Active Dashboards, which can be those sending or receiving commands to/from the logic coded somehow and in particular for**
  - **Server Side Business Logic** → logic on IoT Apps with Snap4City Dashboard Nodes, which is easier to be programmed begin based on Node-RED visual 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



How the Dashboards exchange data



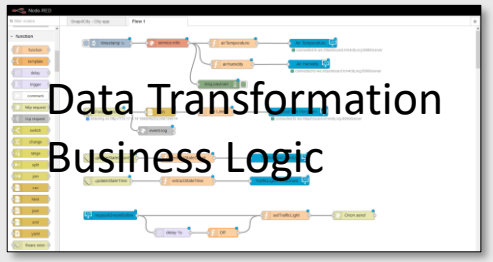
Dashboard Widgets' Capabilities for Business Logics

Widget name / description	Event Driven	Some Local Interaction	Server-Side Business Logic Node-RED	Client-Side Business Logic (IN = JavaScript)
Single Content	Yes	--	OUT	OUT
Speedometer	Yes	--	OUT	OUT
Gauge	Yes	--	OUT	OUT
MultiSeries, Time Series, Curved Line, time compare	Yes	Yes	OUT	IN/OUT
Time Trend, Time Series	Yes	Yes	OUT	IN/OUT
Spidernet, Radar	Yes	Yes	OUT	IN/OUT
BarSeries	Yes	Yes	OUT	IN/OUT
Donut, Pie	Yes	Yes	OUT	IN/OUT
Device Table	Yes	Yes	IN/OUT	IN/OUT
Multi Data Map (dashboard Map)	Yes	Yes	IN/OUT	IN/OUT
Selector	Yes	Yes	--	--
Button, Impulse button	Yes	Yes	IN	IN
Switch, on/off but.	Yes	Yes	IN/OUT	IN/OUT
Knob, Dimer	Yes	Yes	IN/OUT	IN/OUT
Keypad, Num Pad	Yes	Yes	IN	IN
External Content	Yes	Yes	IN/OUT	IN/OUT
Event Driven MyKPI	Yes	--	IN	--
Synoptics (see External Content) (read, write, subscribe)	Yes	Yes	IN/OUT	--
Dashboard Form	Yes	Yes	IN/OUT	Possible on Ext.Content
Speak Synthesis	Yes	--	OUT	Possible on Ext.Content
D3 charts	Yes	Yes	OUT	--

See for this column CSBL PDF

Develop: via Dashboard Wizard

Dashboard Builder: Development



Data Transformation
Business Logic

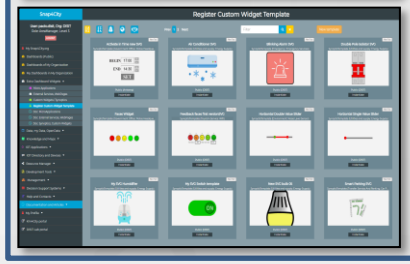
IOT Applications

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

Widget Collection

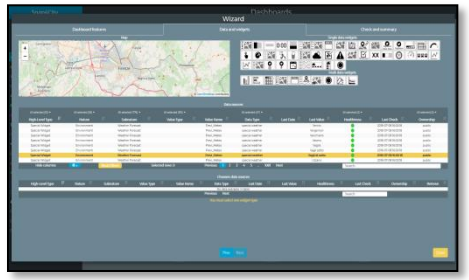
Micro
Applicat
ions

External
Services

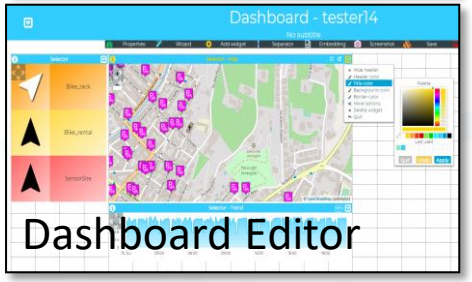


Custom
Widgets/
Synoptics

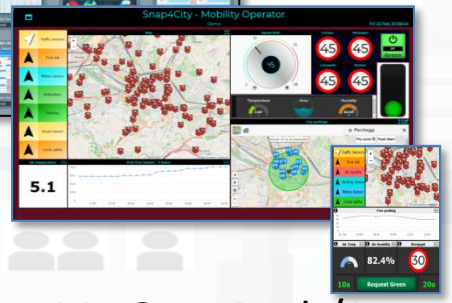
Dashboard Wizard



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



Dashboard Editor



Public
Dashboard
Collection

My Own Dash/App

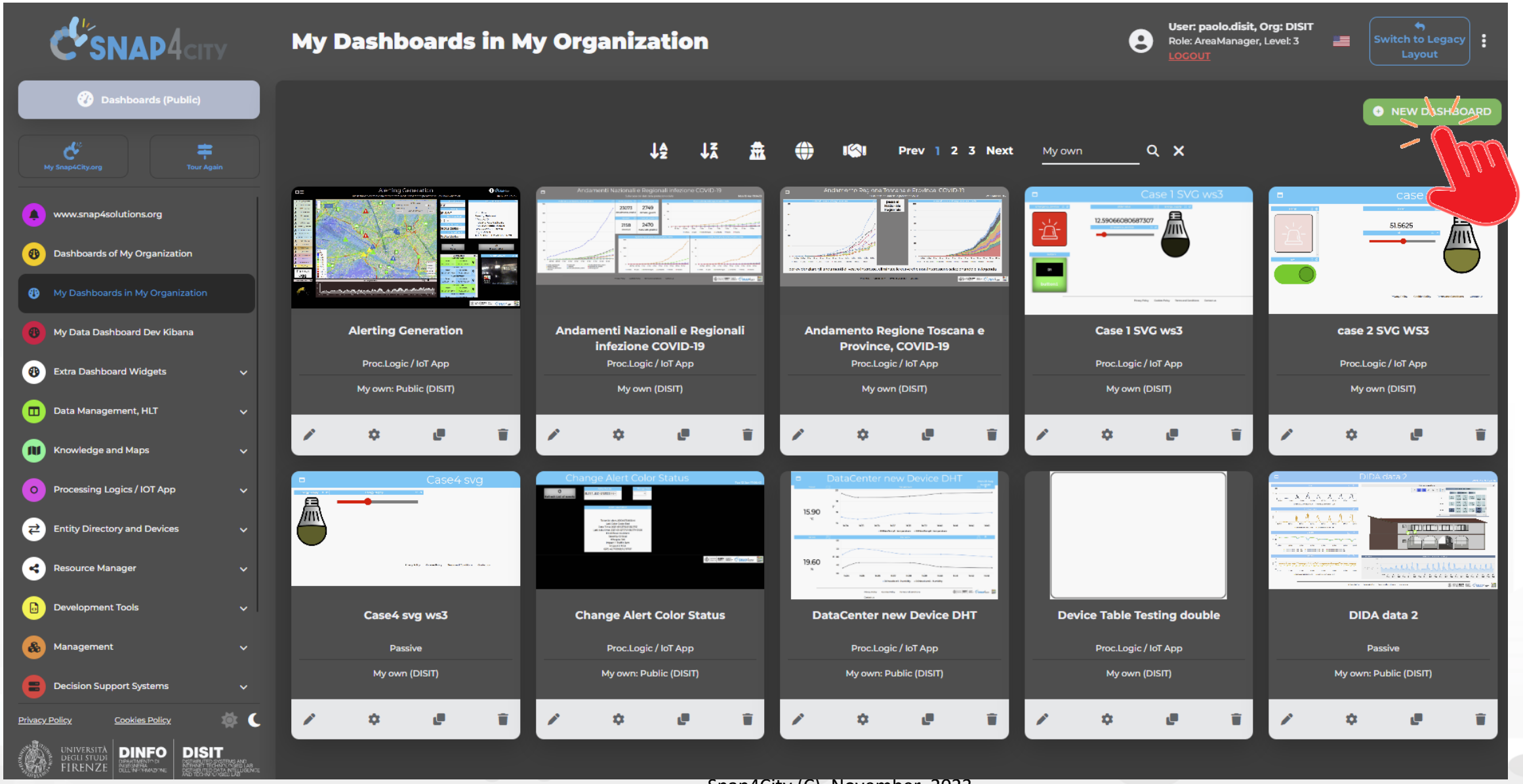
UNIVERSITÀ DEGLI STUDI FIRENZE

DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB DISTRIBUTED DATA INTELLIGENCE AND TECHNOLOGIES LAB

New Dash ?

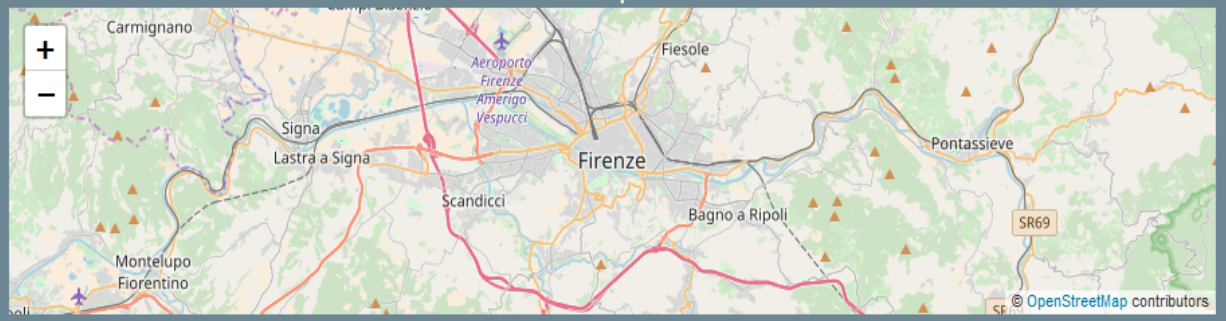
SNAP4CITY KM4CITY



Wizard

Dashboard features | Data and widgets | [illegible]

Map



Single data widgets

Multi data widgets

Data sources

All selected (10)	All selected (55)	All selected (776)	All selected (315)		All selected (47)				All selected (2)
High-Level Type	Nature	Subnature	Value Type	Value Name	Data Type	Last Dat		Last Check	Ownership
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather			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		chiano	2018-07-08 16:00:18	public
Special Widget	Environment	Weather Forecast		Previ_Meteo	special weather		aiano	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 | 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

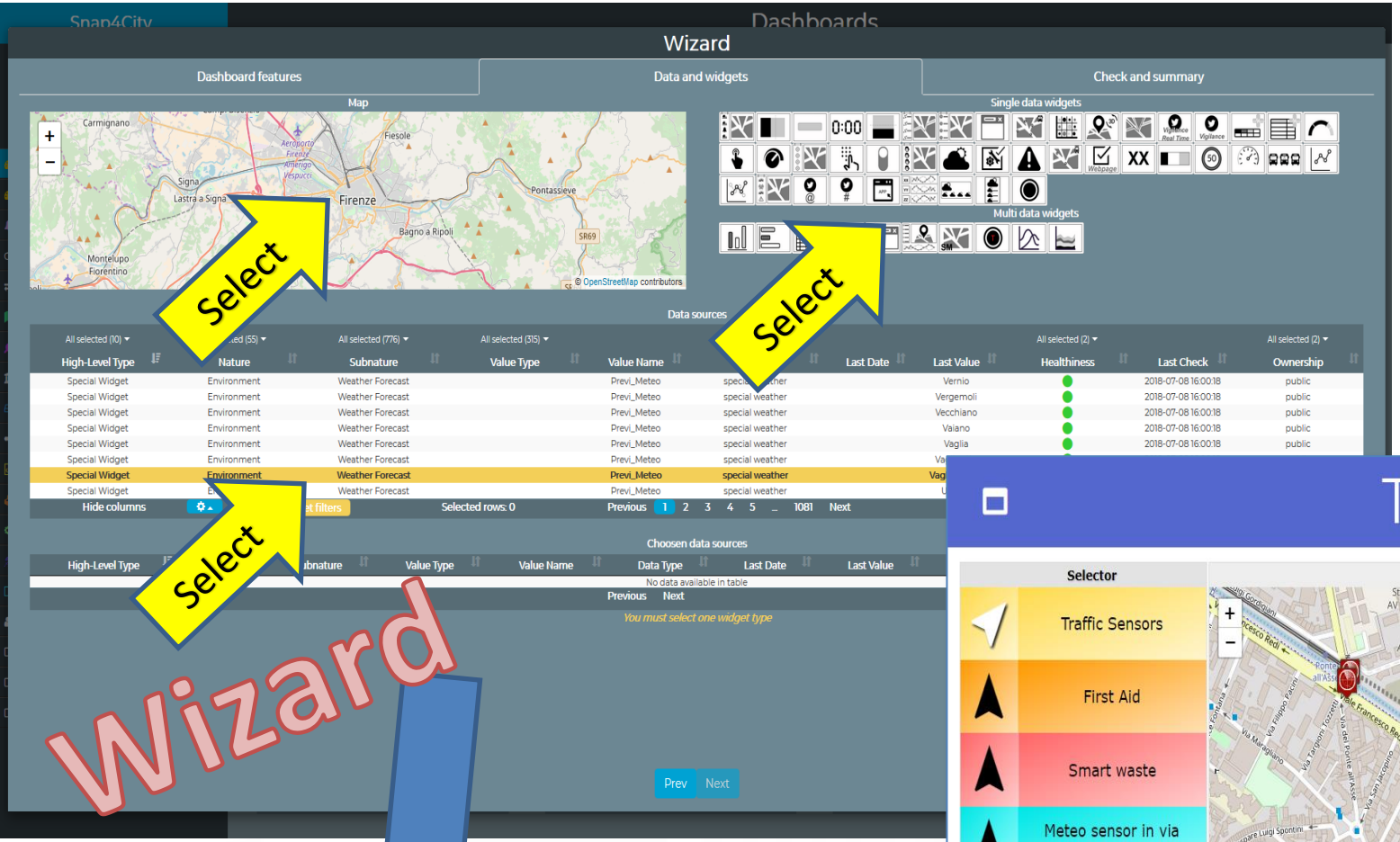
Wizard

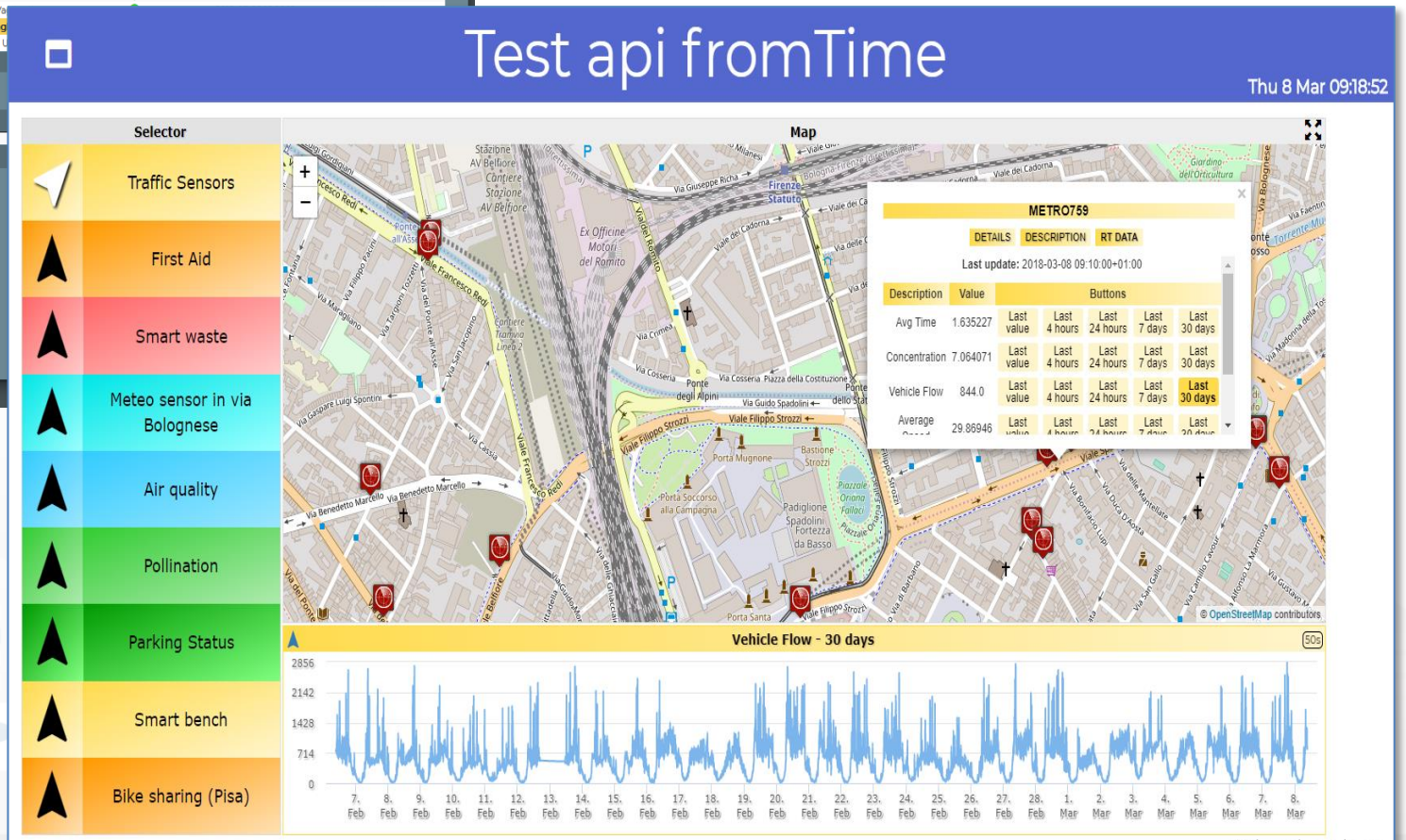
- Select the area of your interest: panning and zooming
- Select the
  - graphic aspect of your interest, or
  - High Level Type of your interest, or
  - Make a search if you a have a precise idea or
  - Act on filters: nature, subnature, type, name, value, date, health, owner, …
  - Combine them as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

RULE OF THUMB

Close

Dashboard Wizard



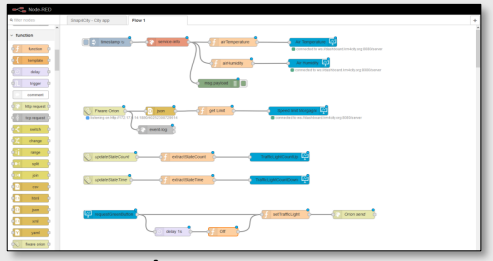


The Wizard help you in selecting only possible combination of data vs graphic representation

Develop:
Dashboard Synoptics

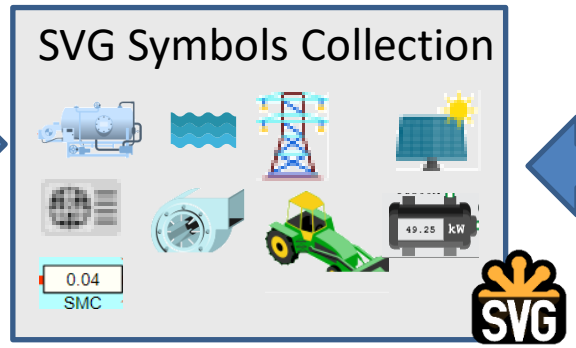
Part 3

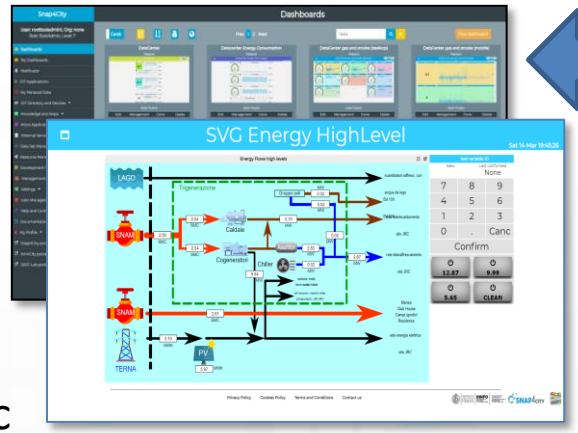
Custom Widget / Synoptic / PIN Development



IOT Applications

Knowledge and Storage Data from the Field and City

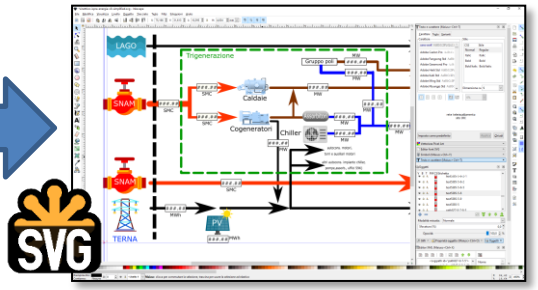




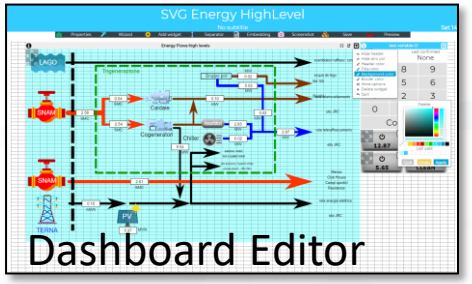
Public Dashboard Collection

My Own Dash/App

Inkscape editor on your computer



Create, save a Custom Widget in SVG



Create, save, load, delegate, grant access

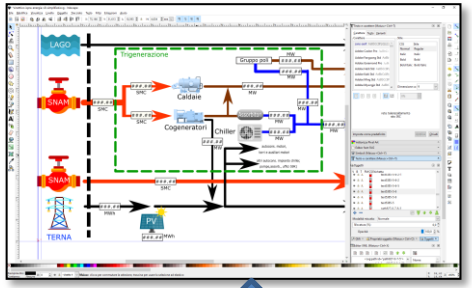
1. Create and Load a Custom SVG
2. Select/Reuse an SVG
3. Make and Instance of Synoptic by Associate Variables with MyKPI
4. Create on Dashboard a Widget based on Synoptic HLT such as Ext. Srv.:
   - https://www.snap4city.org/synoptic/v2/synoptic.html?id=xxxx

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

CW with a single READ Variable are automatically usable as PINS

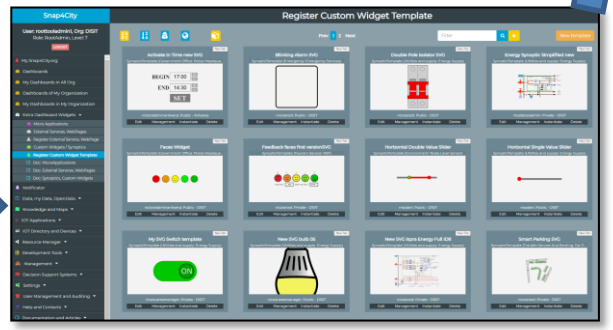
SNAP4CITY | KM 4 CITY

Create, save a Custom Widget in SVG

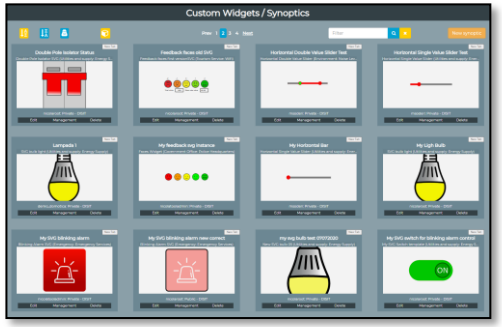


SVG

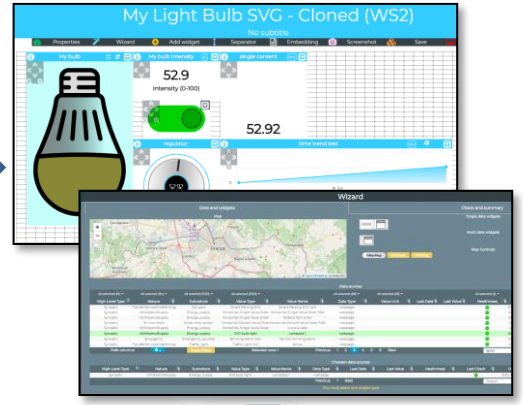
Upload as Custom Widget Template



List of Custom Widgets / Synoptics



Dashboard Editing/wizard

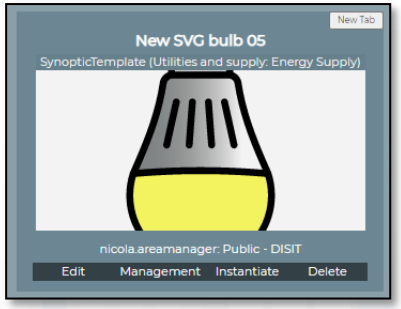


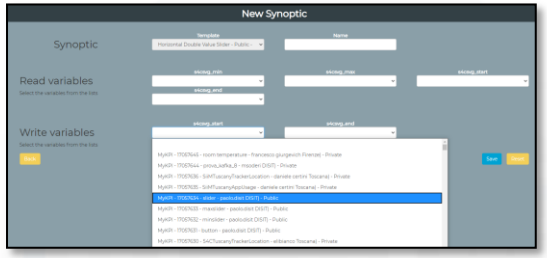
SVG Symbols Collection

From any open library

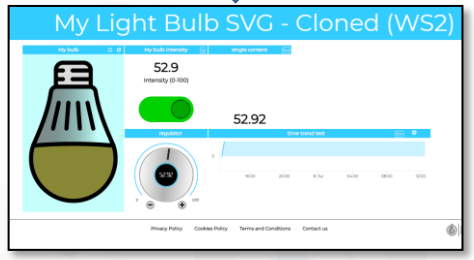
SVG

select



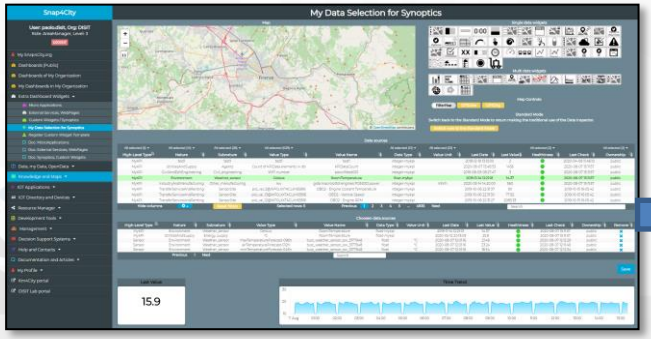


Instantiate as Custom Widgets / Synoptics Connect with WebSockets



Final Dashboard

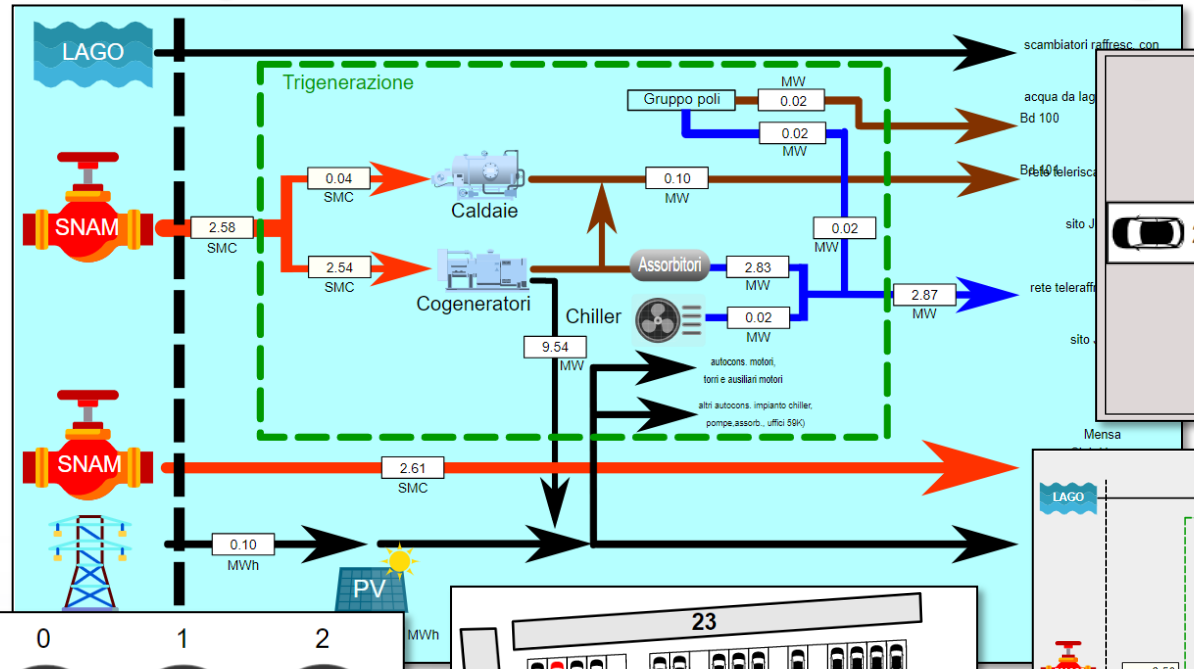
Select MyKPI and Sensor Data for Synoptics cases

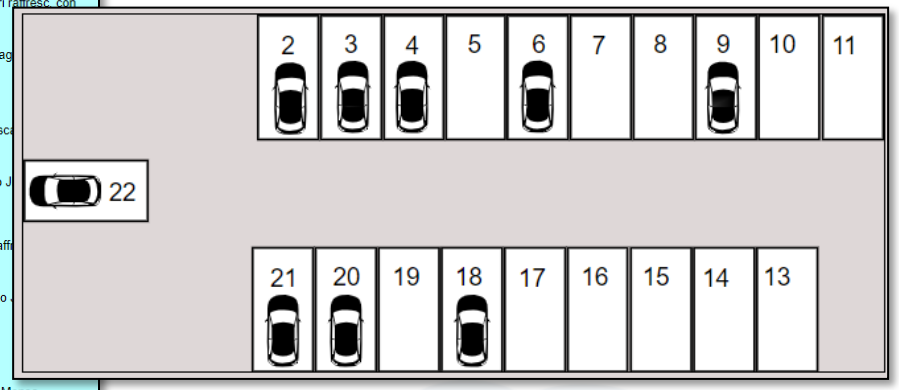


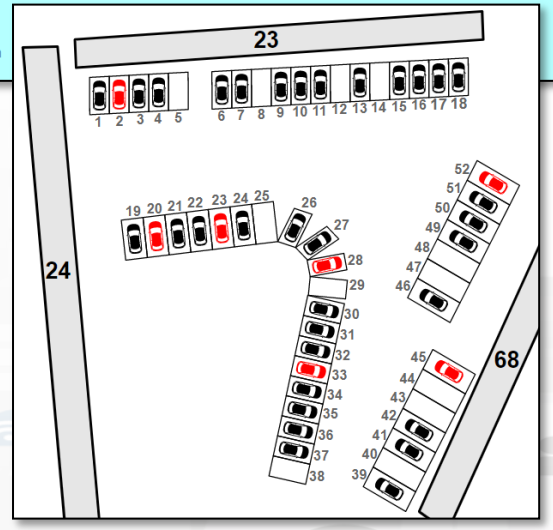
Part 3

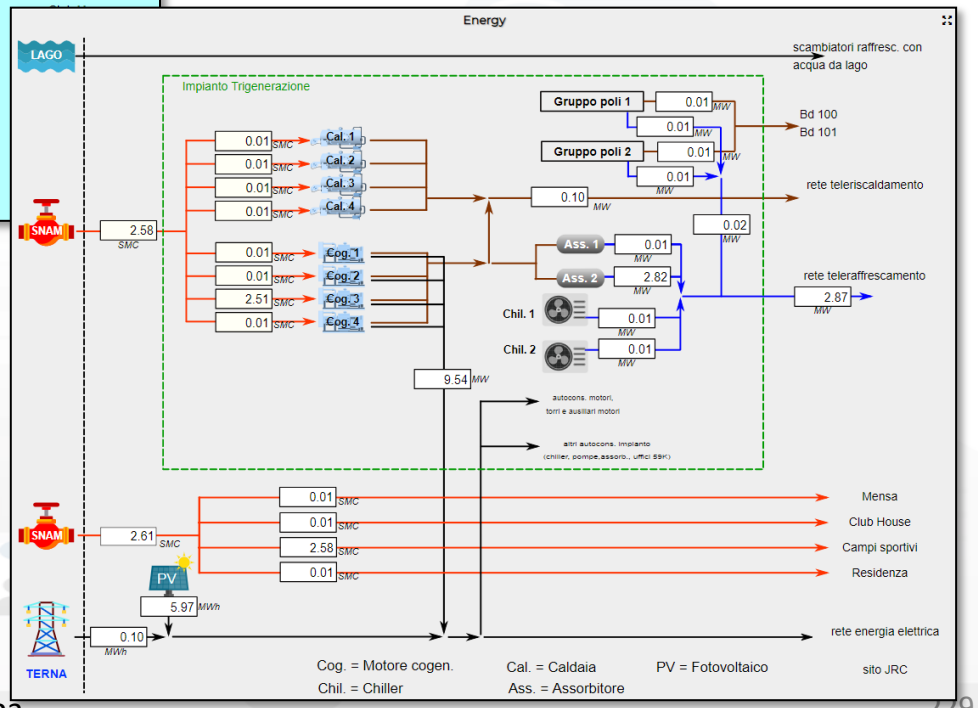
Part 3
Special Custom Widgets

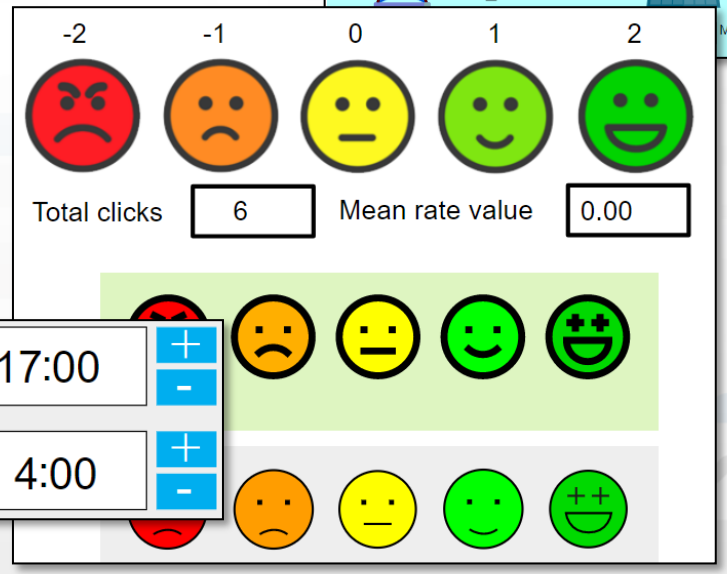
- Smart parking
- Smart Energy
- Smart Light
- Smart ….
- Energy View
- Custom Controls



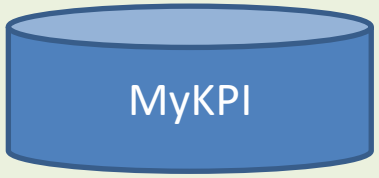




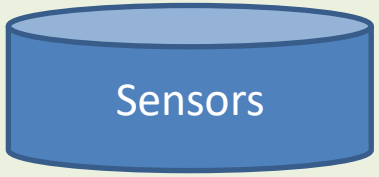




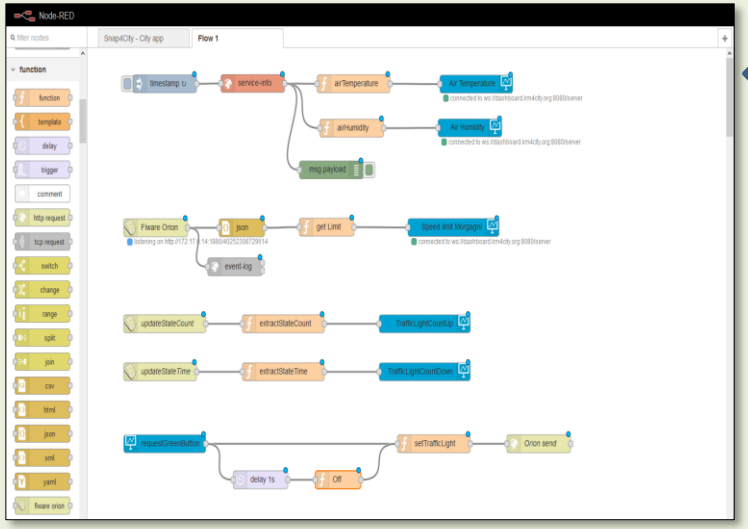
From-To Custom Widgets / Synoptics to Storage in WS



MyKPI



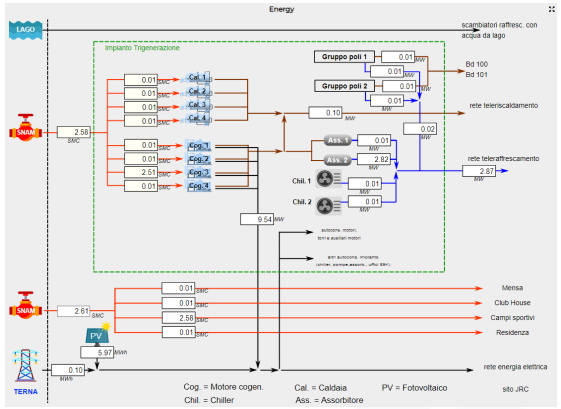
Sensor

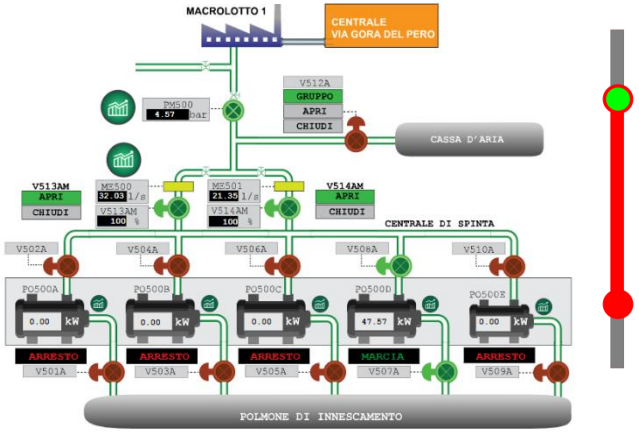


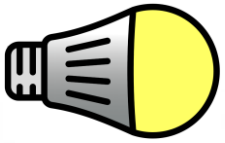
[ ]

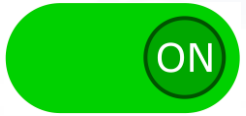
New Shared Variables

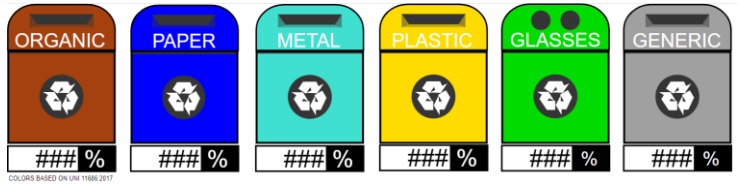
Constant Values

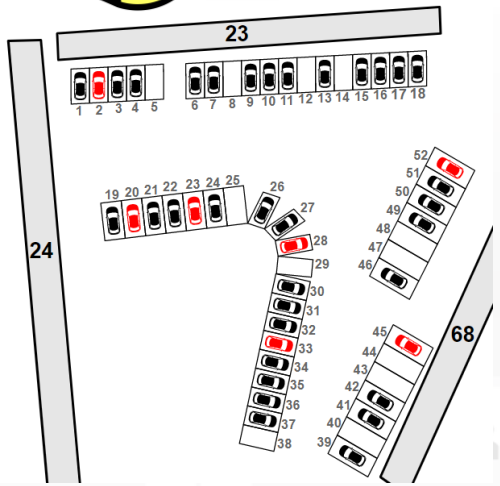


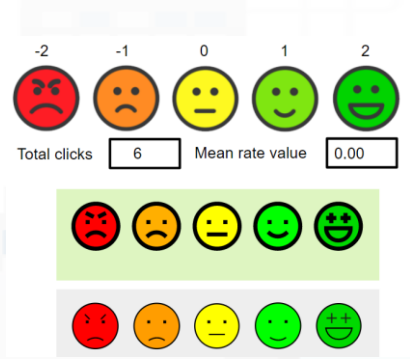


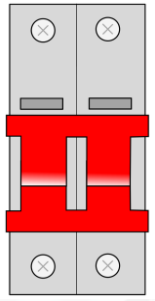


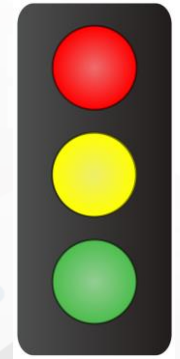








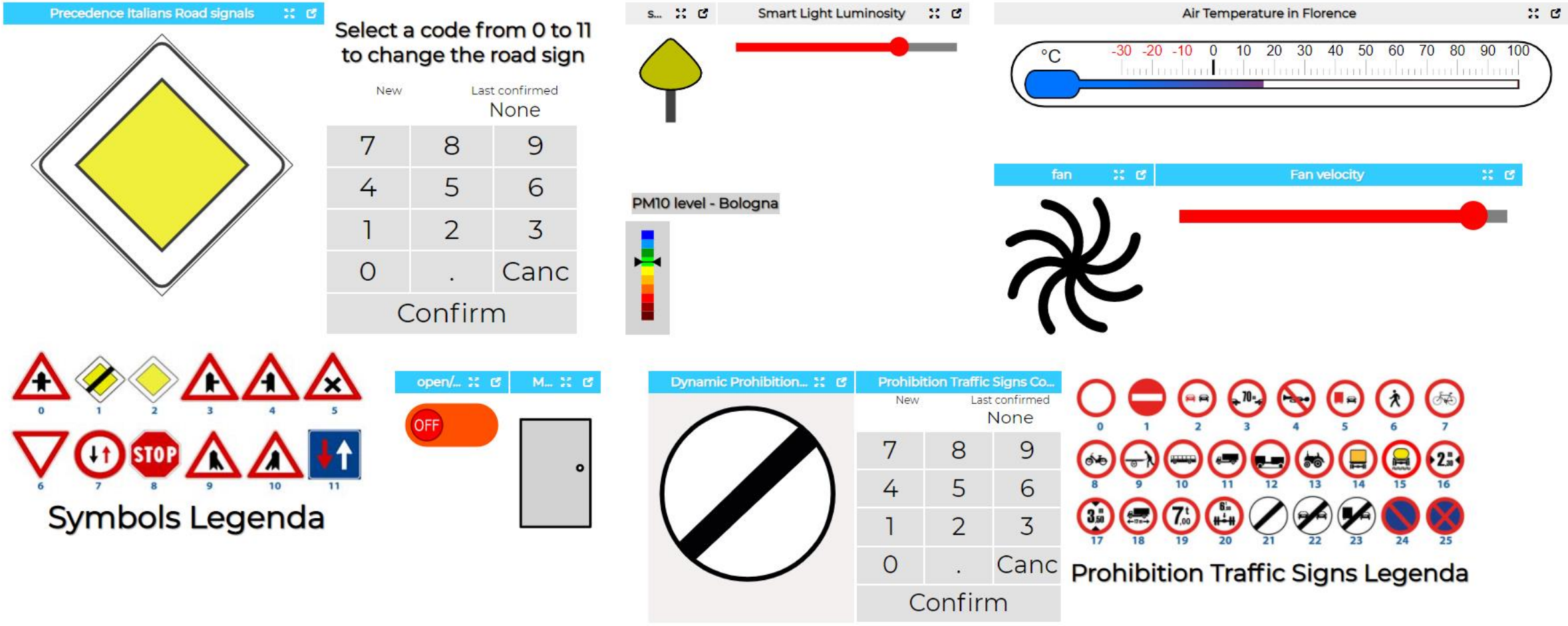




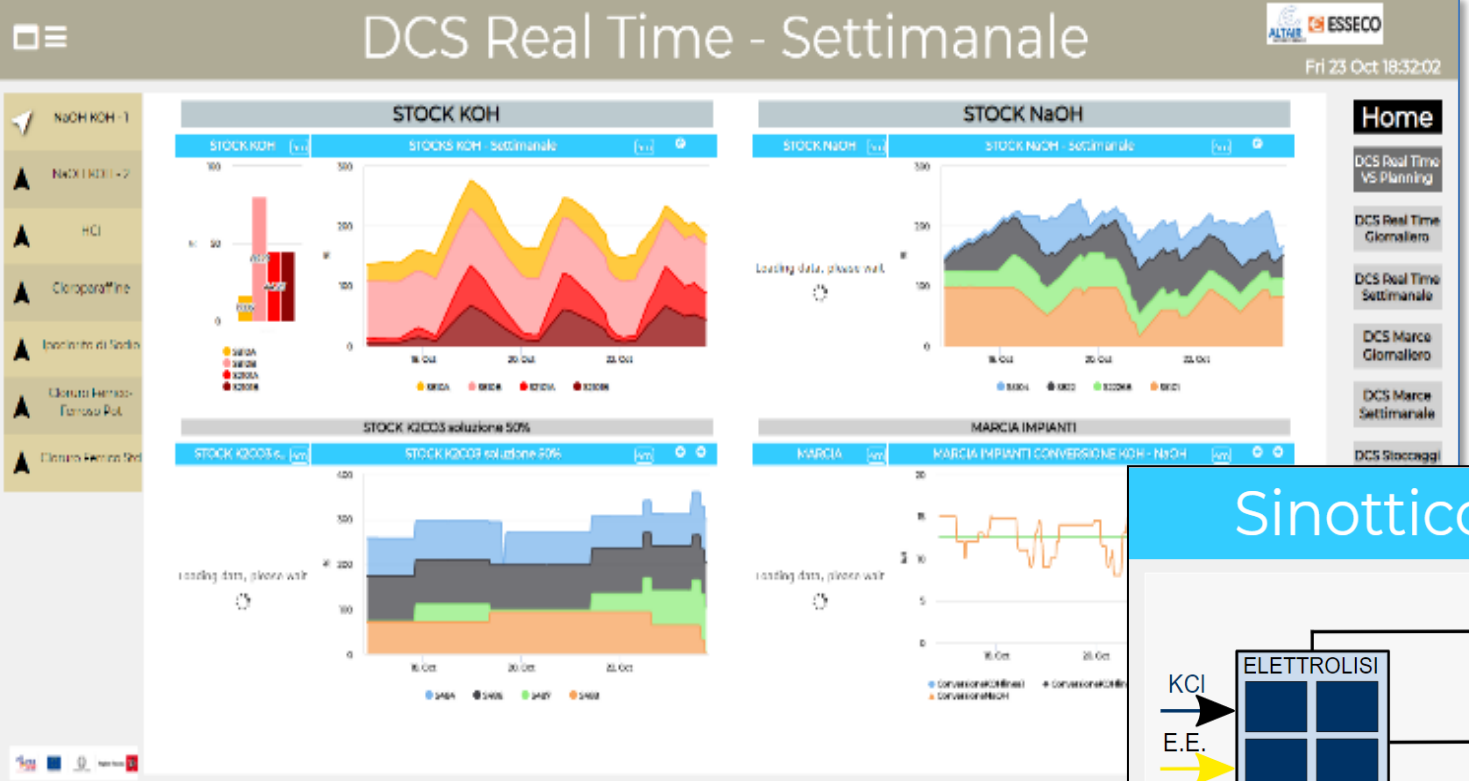
Web Socket Secure

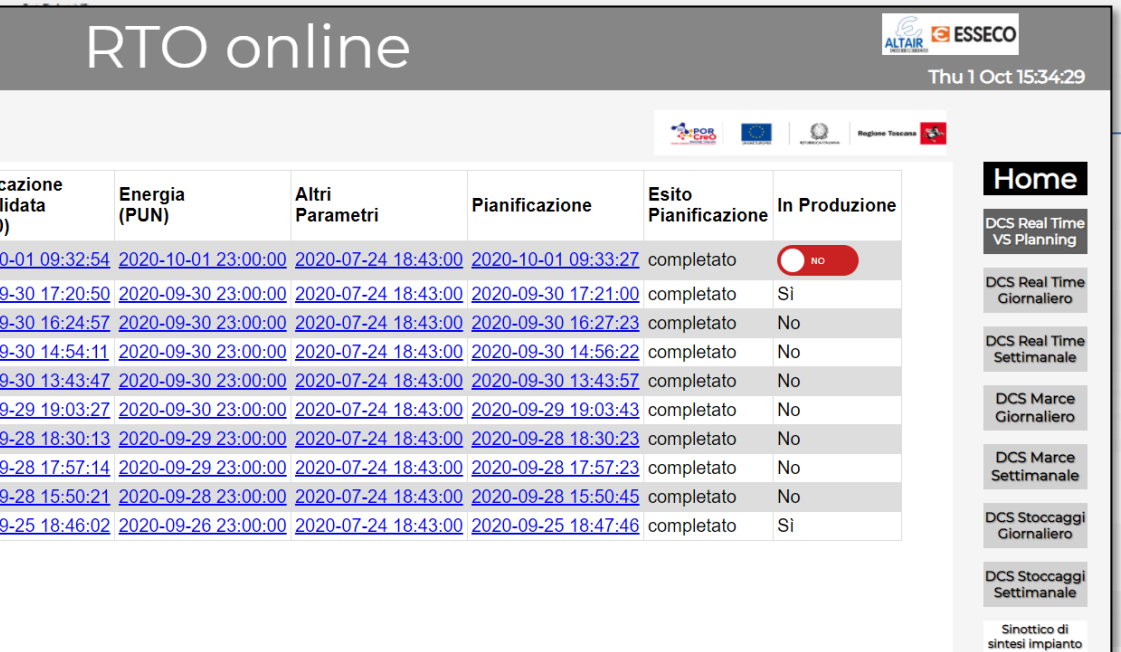
SVG Custom Widgets Examples

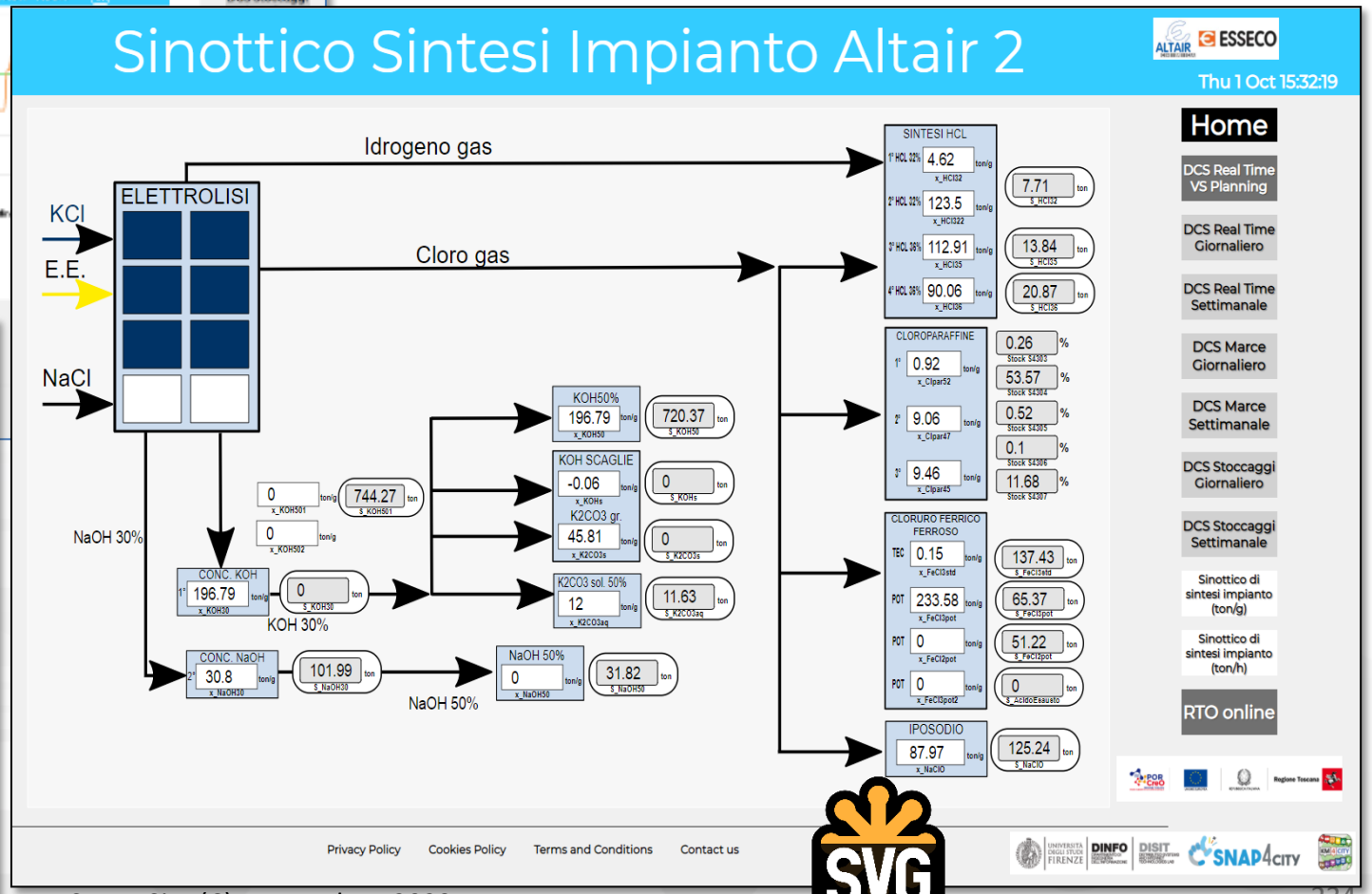
Sat 16 Jan 01:07:39



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


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


RTO online
Thu 1 Oct 15:34:29
Energia (PUN)
Altri Parametri
Pianificazione
Esito Pianificazione
In Produzione
2020-10-01 23:00:00
2020-07-24 18:43:00
completato
2020-09-30 23:00:00
2020-09-29 23:00:00
2020-09-28 23:00:00
2020-09-26 23:00:00
Home
DCS Real Time VS Planning
DCS Real Time Giornaliero
DCS Real Time Settimanale
DCS Marce Giornaliero
DCS Marce Settimanale
DCS Stoccaggi Giornaliero
DCS Stoccaggi Settimanale

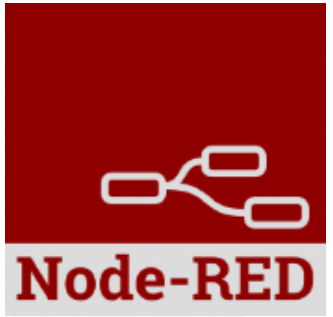

Sinottico Sintesi Impianto Altair 2
Thu 1 Oct 15:32:19
Idrogeno gas
Cloro gas
KCl
E.E.
NaCl
ELETTROLISI
NaOH 30%
KOH 30%
NaOH 50%
SINTESI HCL
CLOROPARAFFINE
CLORURO FERRICO FERROSO
IPOSODIO
Home
DCS Real Time VS Planning
DCS Real Time Giornaliero
DCS Real Time Settimanale
DCS Marce Giornaliero
DCS Marce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

Develop: Dashboards with Server-Side Business Logic, Part 3

IoT App / Proc.Logic

- Storage → IoT App / Proc.Logic
- External Service ←→ IoT App / Proc.Logic
- Dashboards ←→ IoT App / Proc.Logic

Part 3



- Data Analytics ←→ IoT App / Proc.Logic
- Broker → Storage
- IoT App / Proc.Logic → Broker
- Broker → IoT App / Proc.Logic
- IoT App / Proc.Logic → Storage

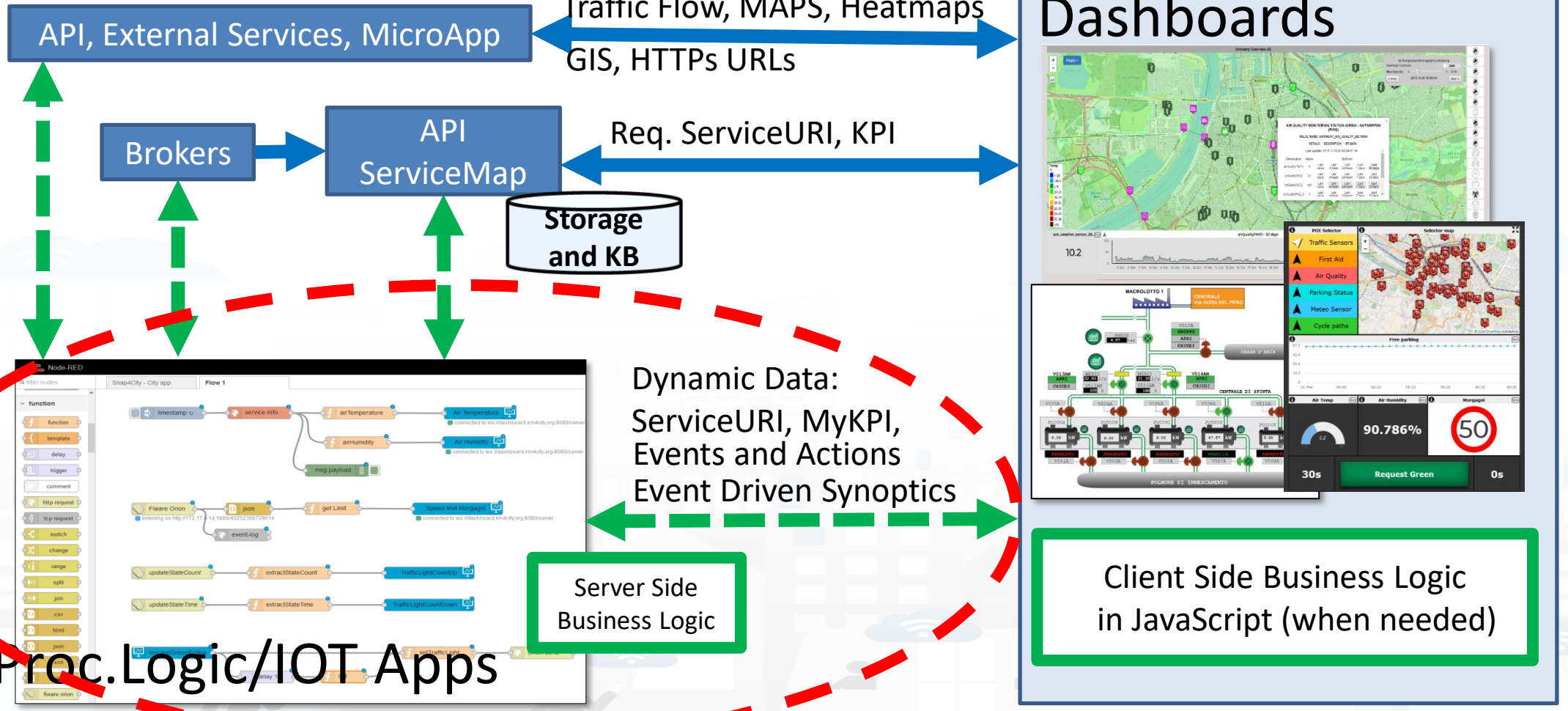
Part 4

Part 5

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

How the Dashboards exchange data



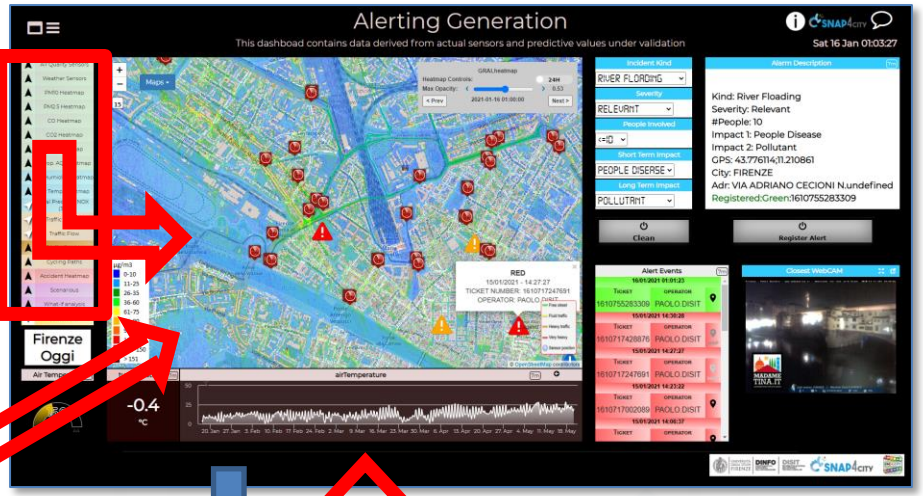
Maps Server Side Business Logic vs IOT Apps

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

data

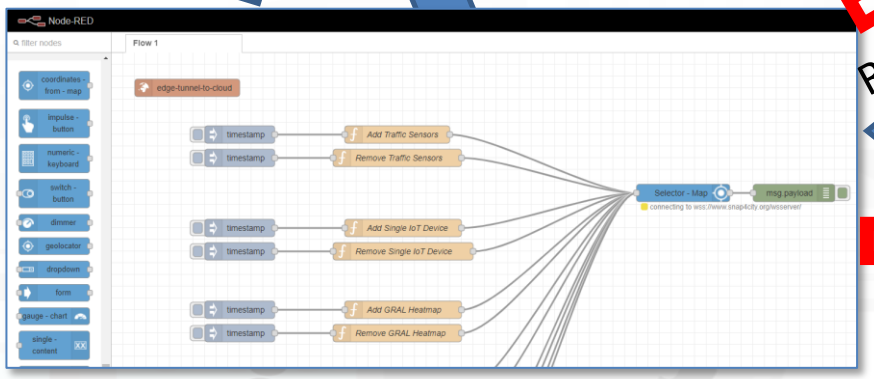
data

Selector



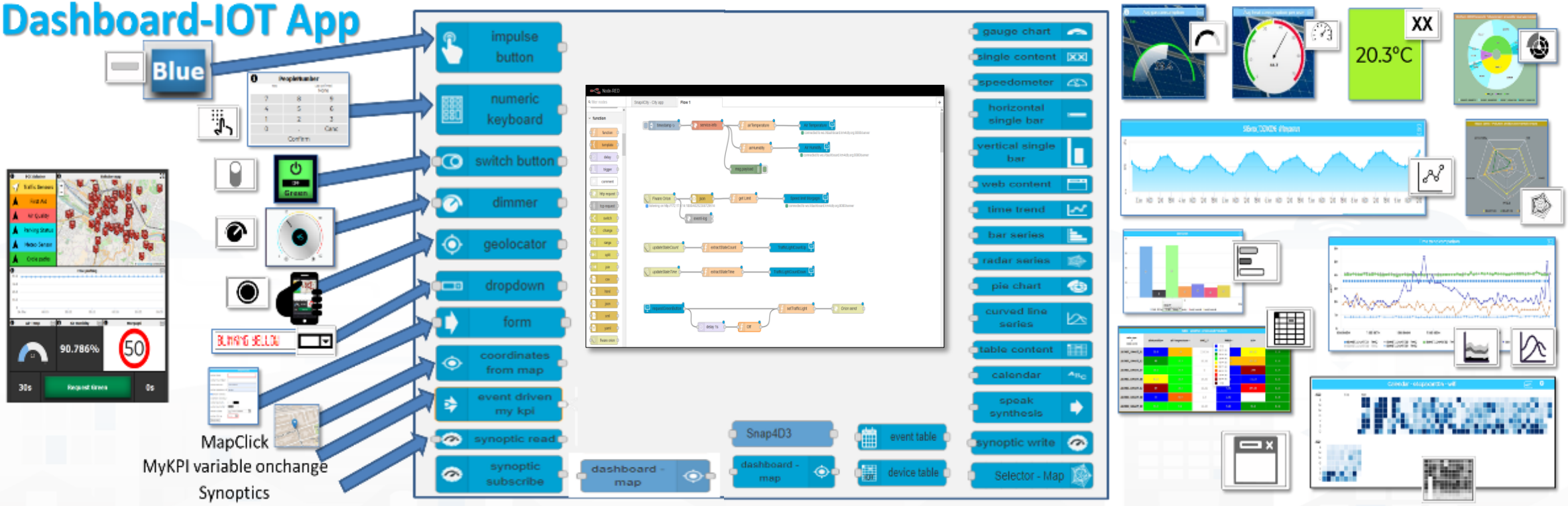
Data, changed data

References, commands, directives, selections



selections, positions, ServiceURI

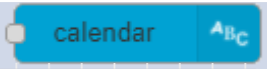
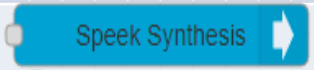
Dashboard-IOT App



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

Dynamic (6/23)

SNAP4CITY | Node-RED

Widgets ICONS	Widget Name, Description	IOT App	Dashboard-IOT App	KPI (metric)	MyPersonalData	MyData	My KPI	Sensor
XX	Single Content	X (cs)	X (ED)	X	X	X	X	X
50	Custom widgets in SVG are data driven	X (cs)	X (ED)				X	X
	Speedometer, Gauge	X (cs)	X (ED)	X	X	X	X	X
	Device Table	X (cs)	X (ED)				X	X
	Single Bar, V/H	X	X (ED)	X				
	Single and Multiple Bars, stacked or not, ordered	X (cs)	X (ED)	X	X	X	X	X
	MultiSeries, shaded, staked and non staked, TTT	X (cs)	X (ED)	X	X	X	X	X
	Time Trend (single)	X	X (ED)	X	X	X	X	X
XX YY ZZ	Time Trend Compare			X			X	X
	SpiderNet, radar, Kiviat	X (cs)	X (ED)	X	X	X	X	X
	Pie, Donut, 2 layers Donut	X (cs)	X (ED)	X	X	X	X	X
	Table	X (cs)	X (ED)	X	X	X	X	X
	Calendar 	X (cs)	X (ED)				X	X
	Speak Synthesis 	X (cs)	X (ED)				string	string
	Maps	X (cs)	X (ED)	Many High Level Types			X	X

DD: Event Driven

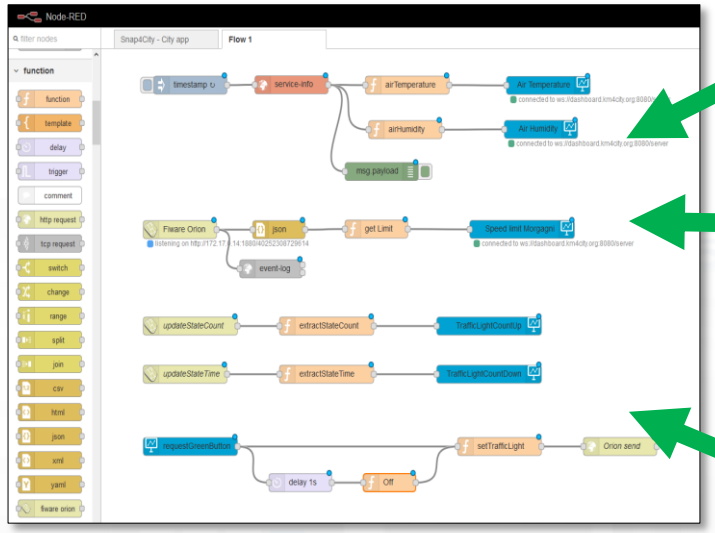
Widgets and their counterpart Nodes

- **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**).

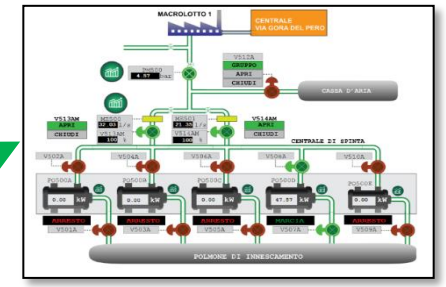
On Server-Side (into Proc.Logic) the developer can even create some HTML pages and provide them into a Dashboard Widget. And a mixt of Widgets in, out, in/out

Advanced IOT Applications

- **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 IOT Applications
  - Collect files on web and system
  - produce files on web ad system
  - have CSS and AJAX control

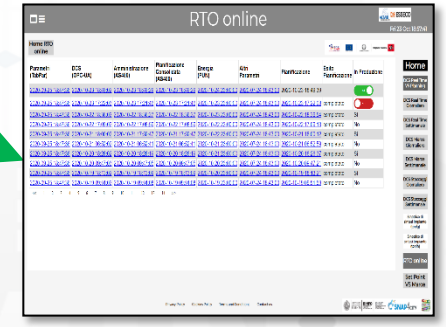


Synoptics
Custom
Widgets



Widgets
Maps
Buttons
Keypads
Controls
..

HTML pages
HTML Forms
Tables
..

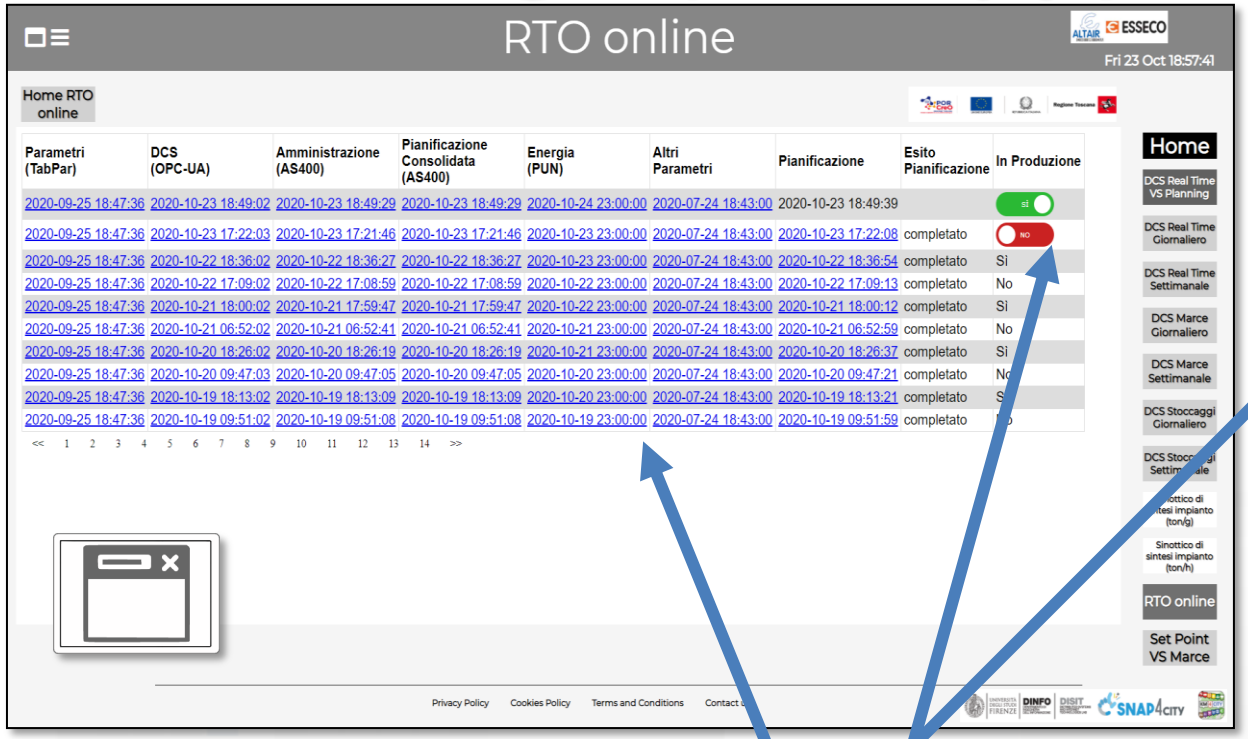


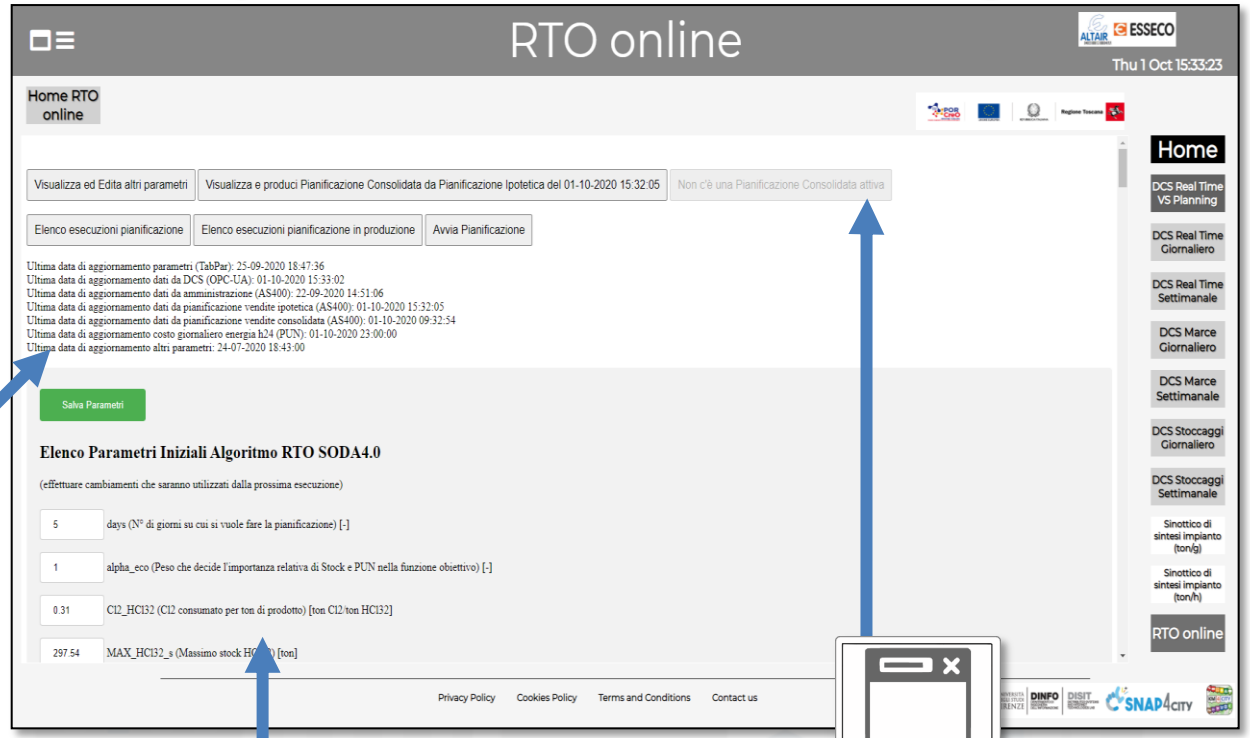
HTML

https://www.snap4city.org/394

https://www.snap4city.org/596

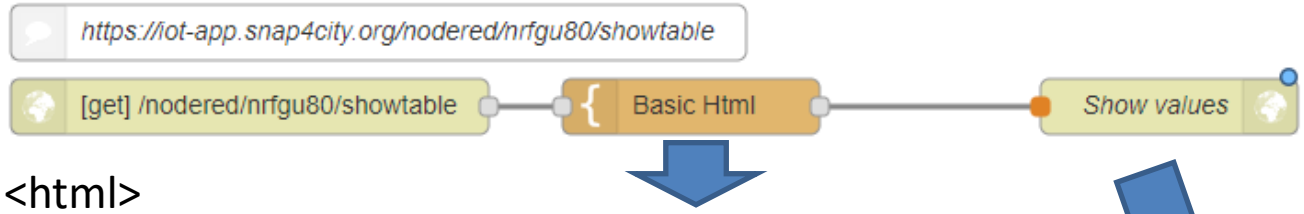
Proc.Logic (IOT App) with Dynamic Web Pages



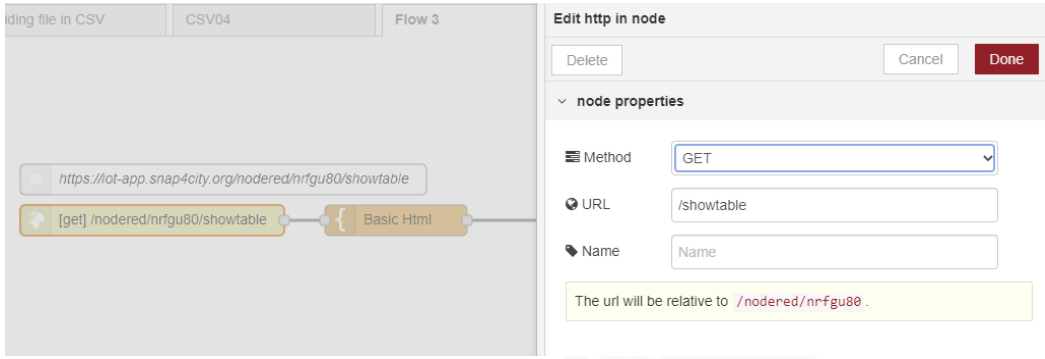


- **HTML pages** can
  - be dynamically generated from the Proc.Logic / IoT App
  - provide forms to produce data to the Proc.Logic / IoT App, also including interactive elements
  - collect file from users, and produce files to web and to the system
  - have CSS and AJAX controls

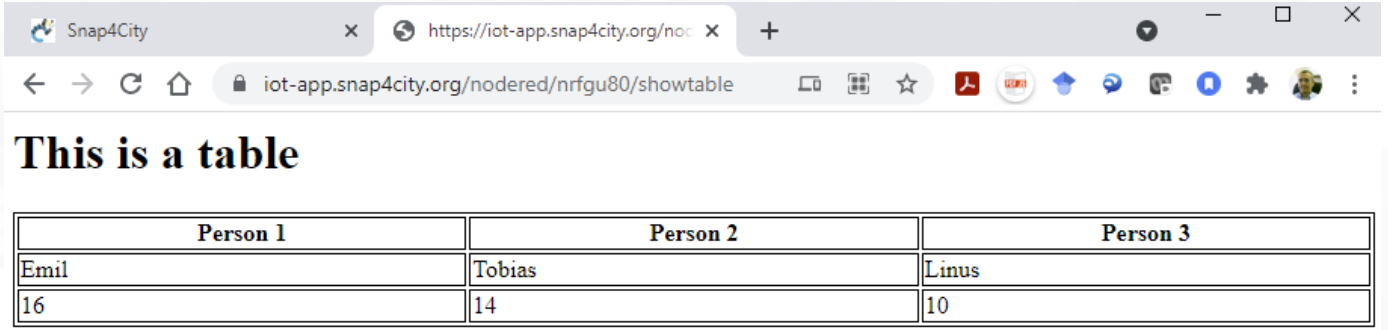
From IoT App to generate HTML pages, forms




```
<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>
```



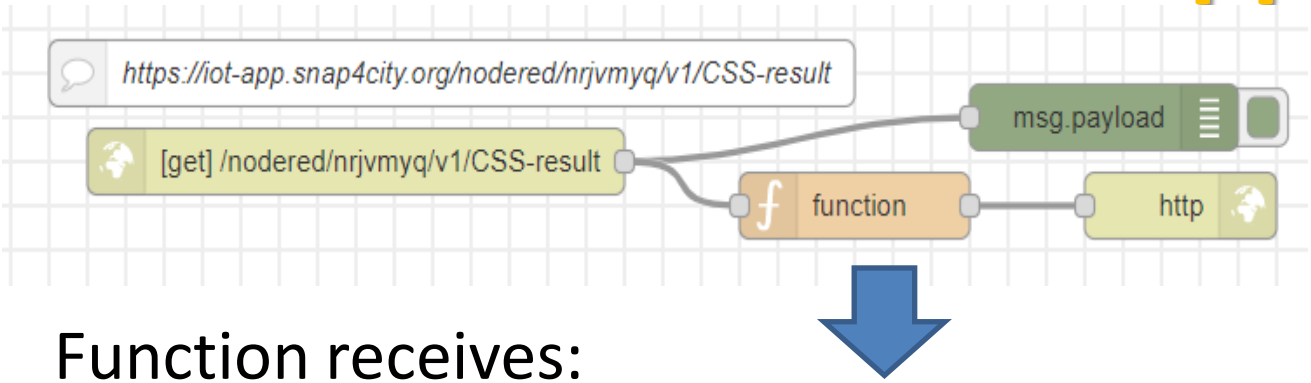
Edit “HTTP IN NODE” as above, also to get the ID of your IoT APP.
The ID is used in the URL of the generated page



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



Function receives:

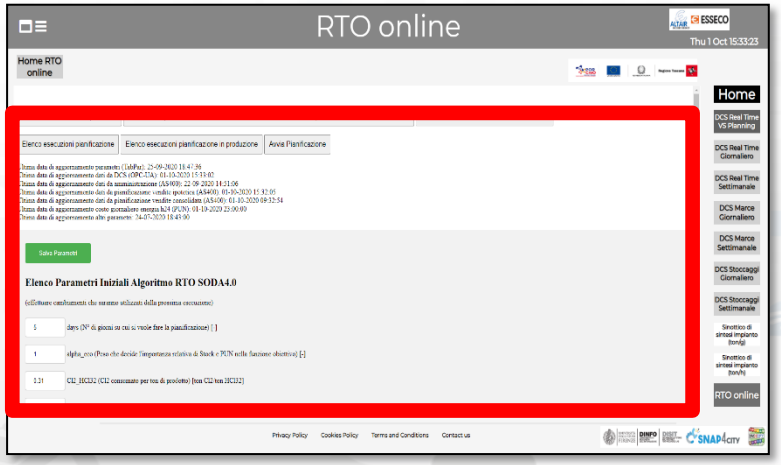
{"prova":"1","test":"mio"}

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

The HTML page can be emebedded into External Content widget of a dashboard

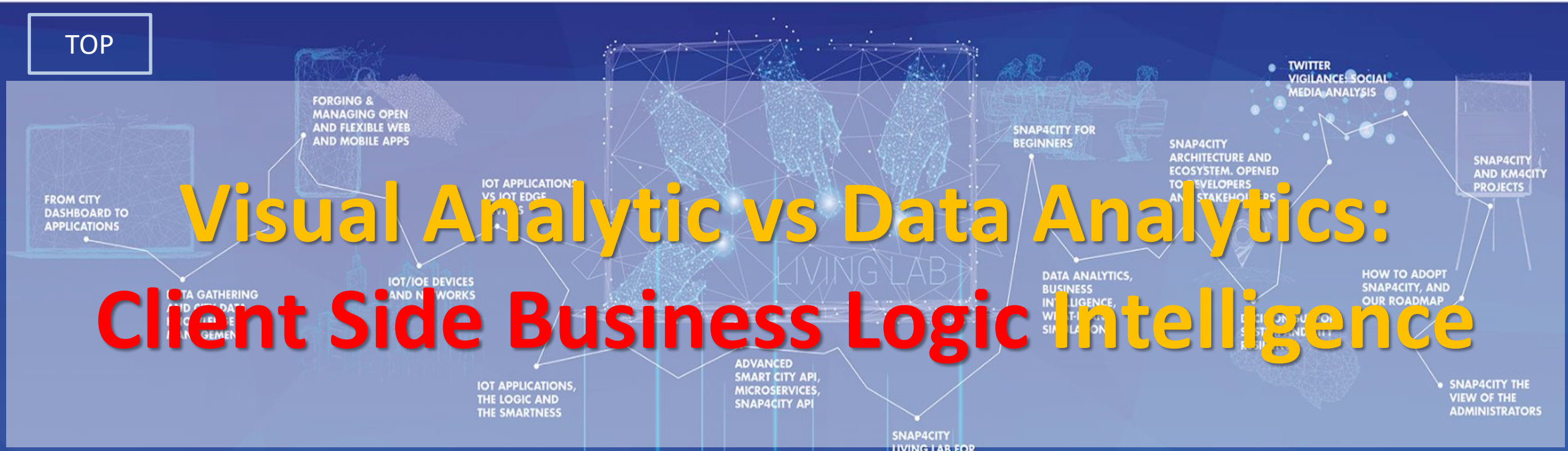


Domain Prefix

IoT App ID

Your custom

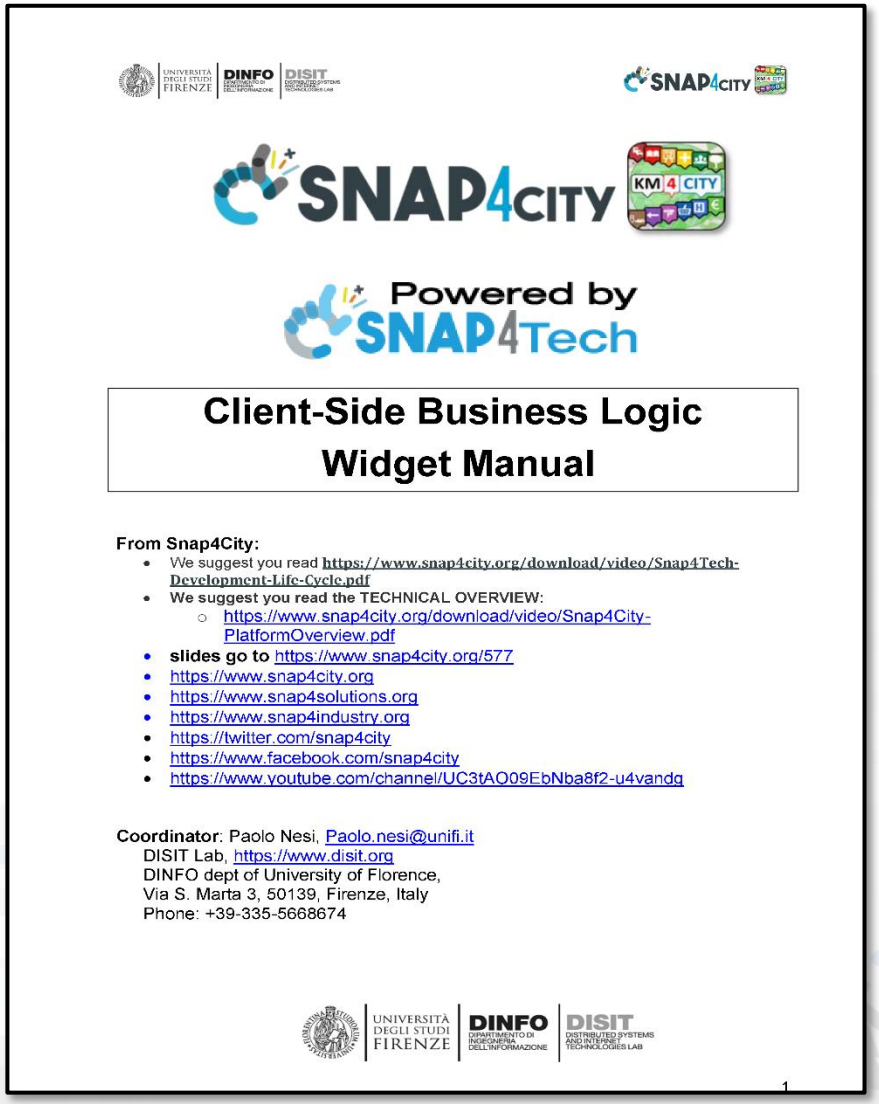
TOP



Visual Analytic vs Data Analytics: Client Side Business Logic Intelligence

CSBL

Client Side Business Logic



Powered by SNAP4Tech

Client-Side Business Logic Widget Manual

From Snap4City:

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

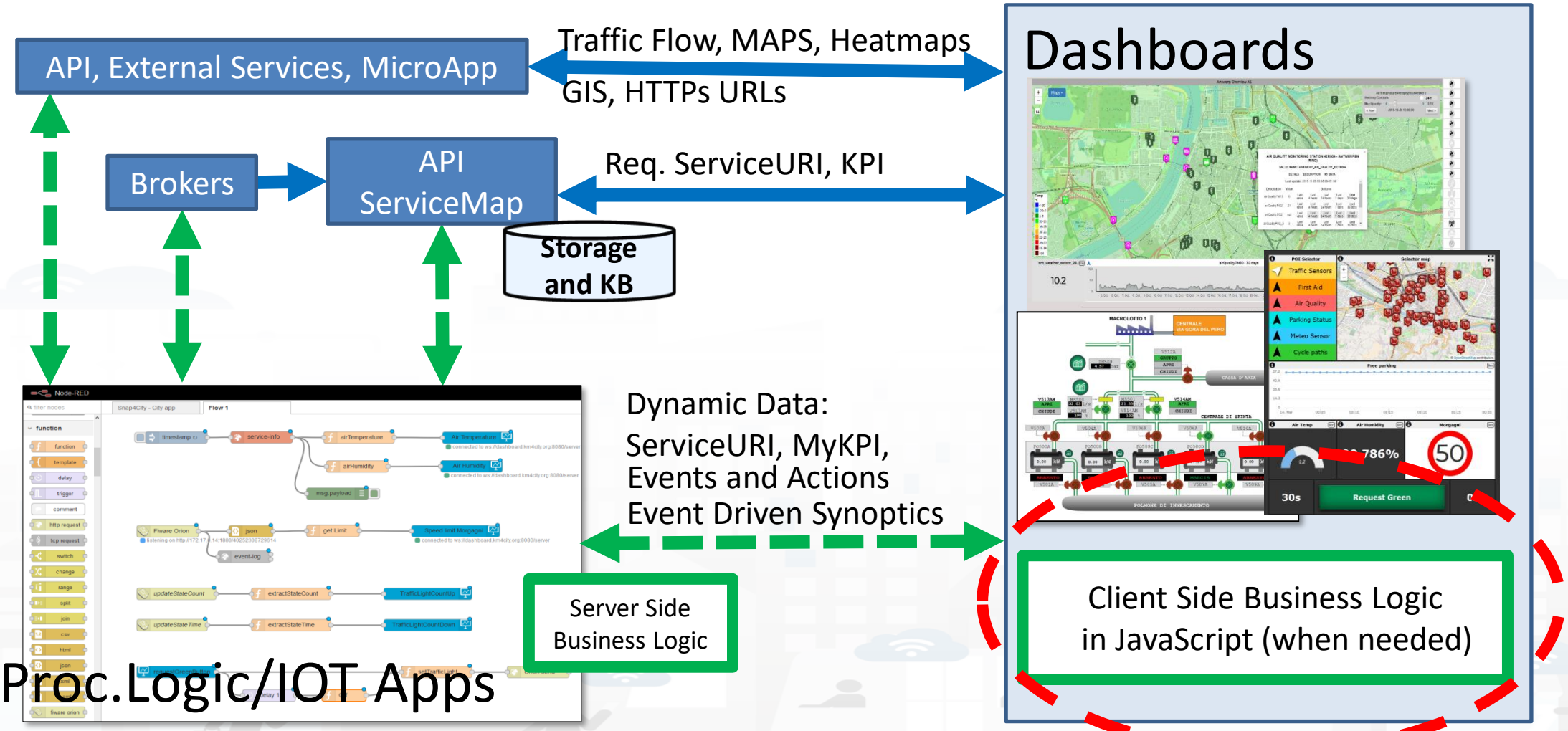
Coordinator: Paolo Nesi, Paolo.nesi@unifi.it
DISIT Lab, https://www.disit.org
DINFO dept of University of Florence,
Via S. Marta 3, 50139, Firenze, Italy
Phone: +39-335-5668674

1

https://www.snap4city.org/download/video/ClientSideBusinessLogic-W

df

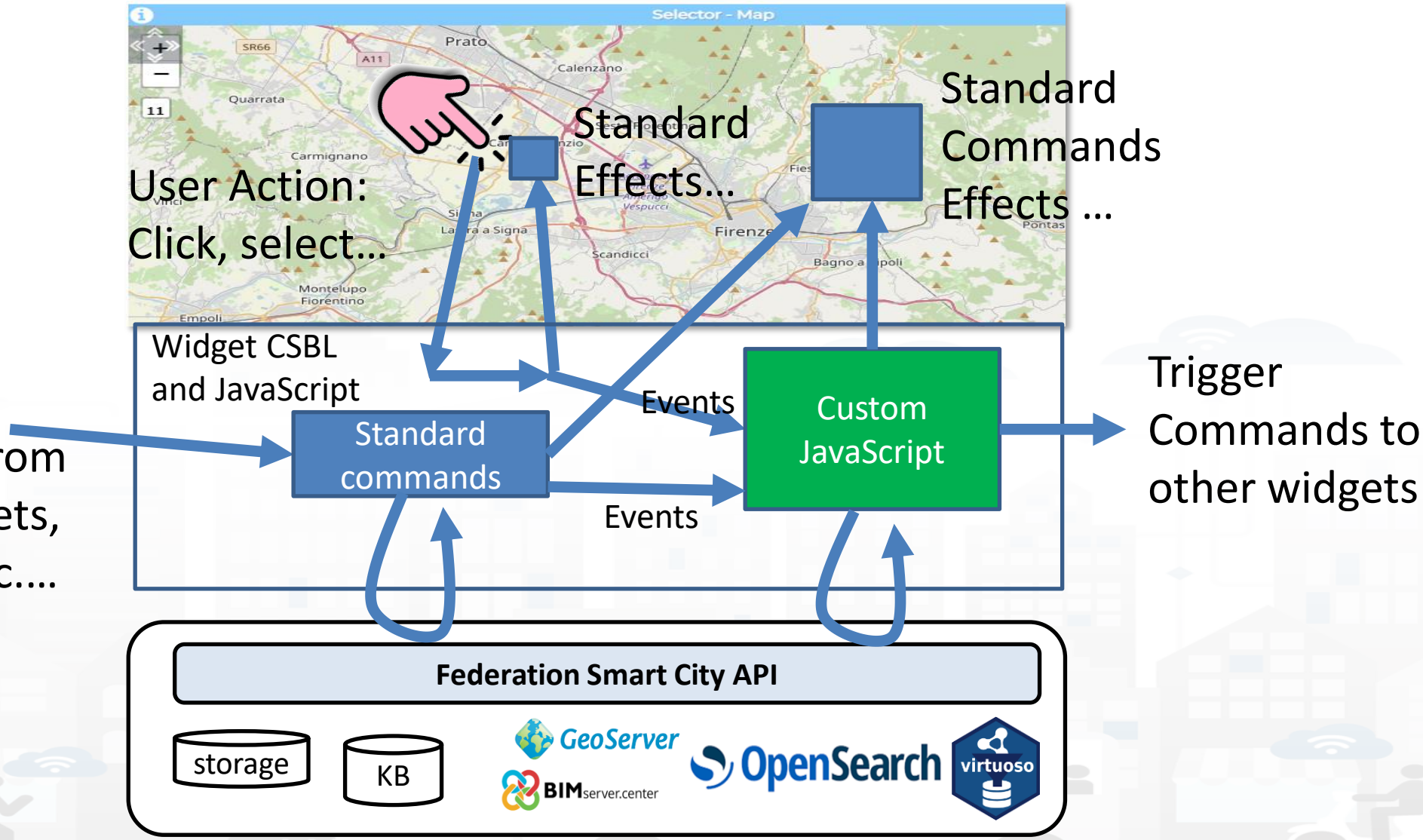
How the Dashboards exchange data



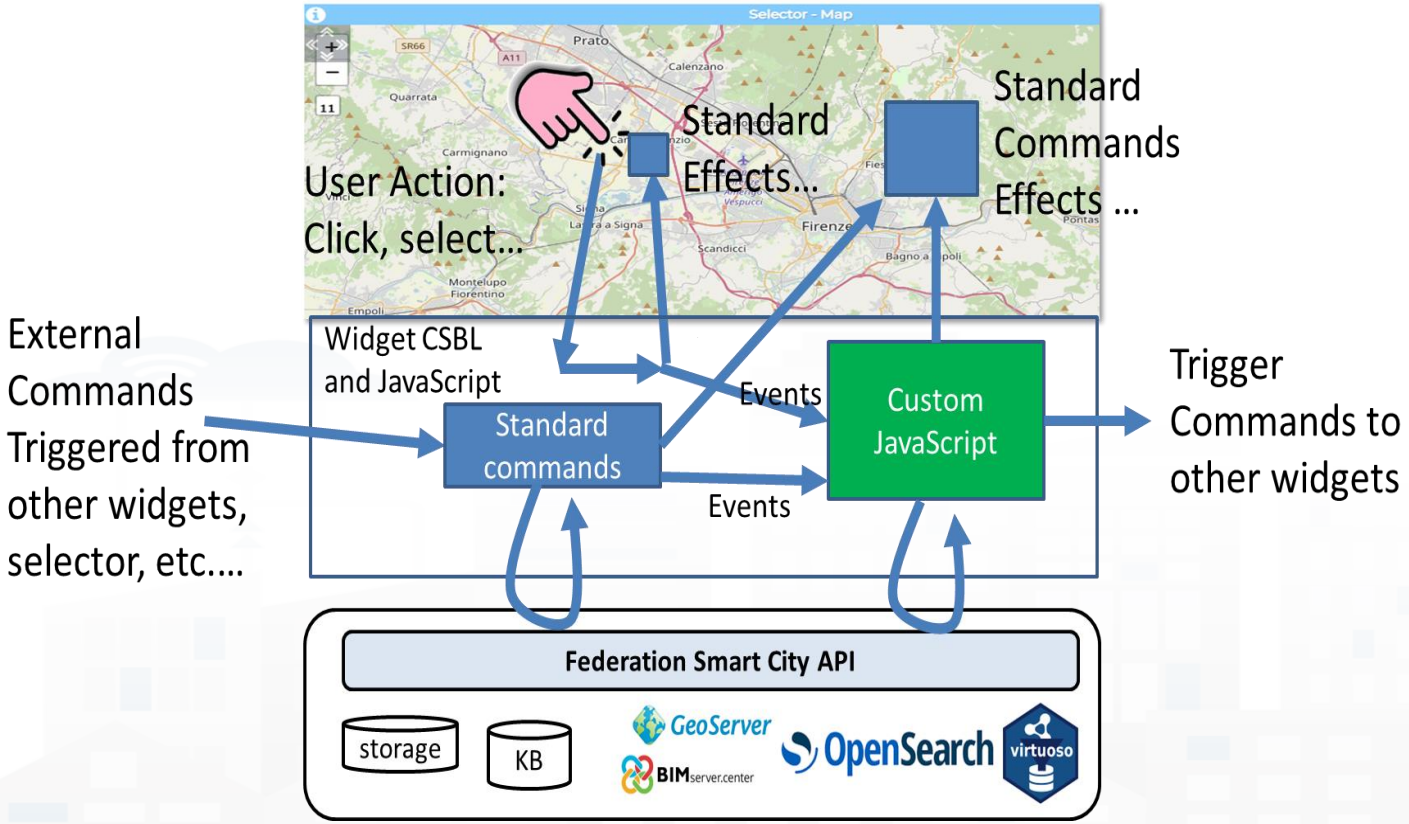
Client Side Business Logic, CSBL

- 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 authorized, please ask to snap4city@disit.org
- When working in SSBL, widgets can be created
  - and edited from Node-Red Processing Logic.
  - also through the Dashboard Wizard

Concepts of CSBL



Custom JavaScript on CK editor of Widget

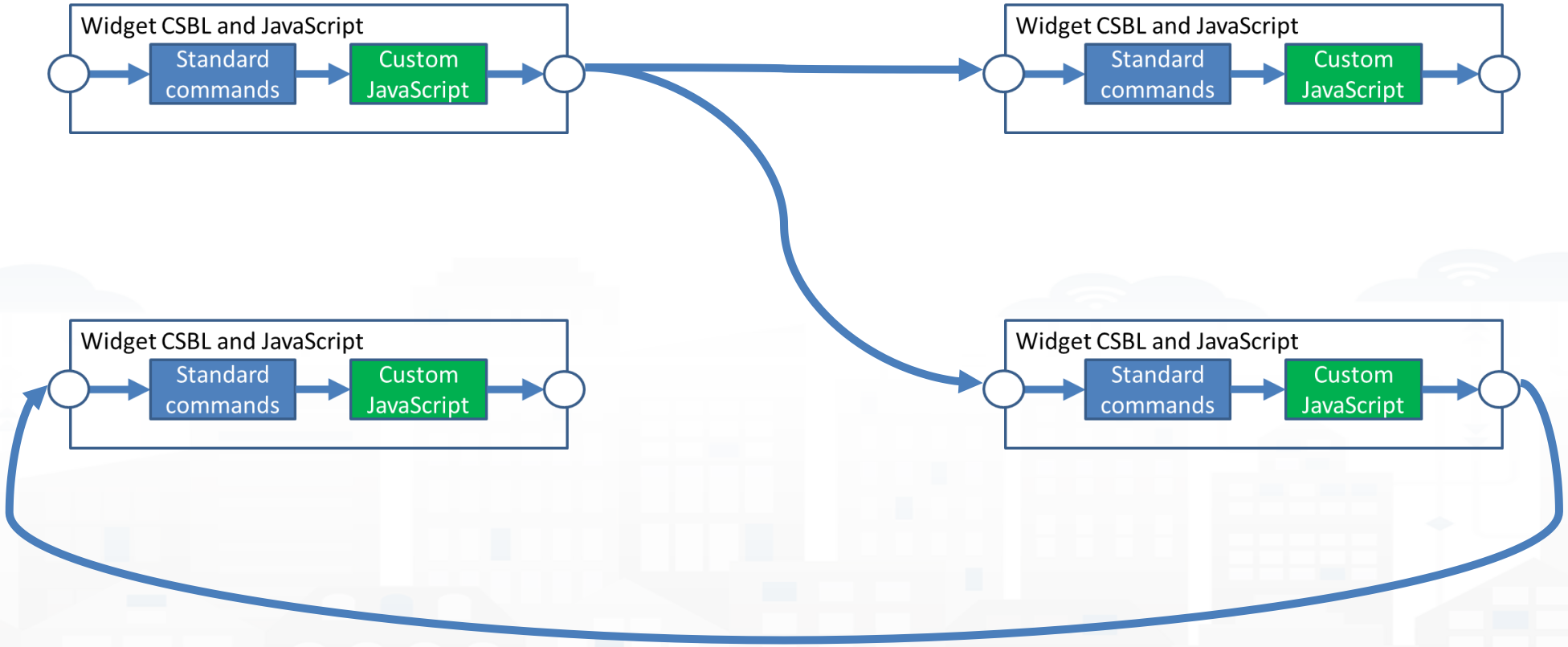


Custom JavaScript

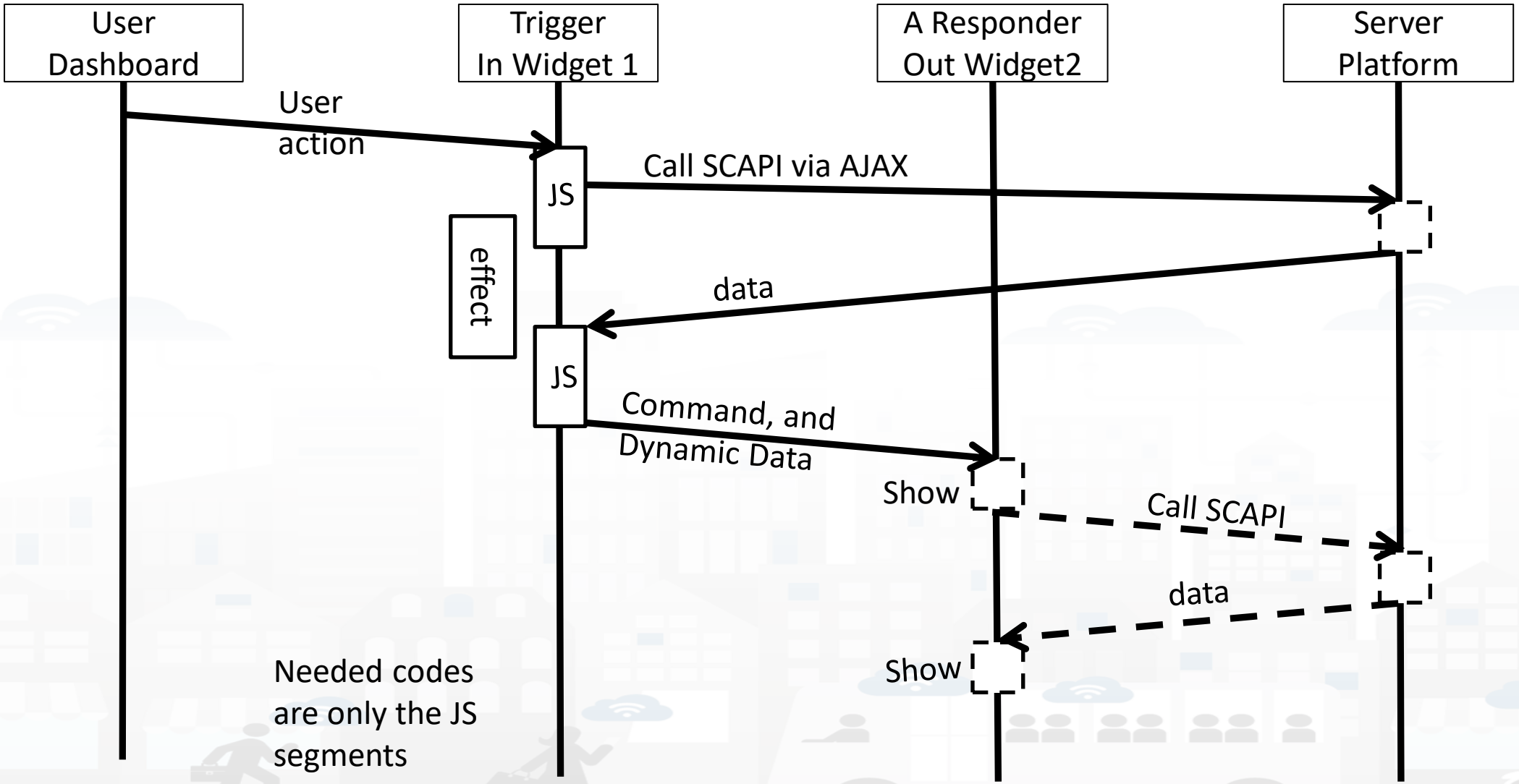
```
function execute ()
{
Var e = JSON.parse(param)
if (e.event == ‘’…….’’) {
        ……….
        }
else if (e.event == ‘’…….’’) {
        ……….
        }
else {
        ……….
        $(‘body’).trigger ({……..});
        }

}
```

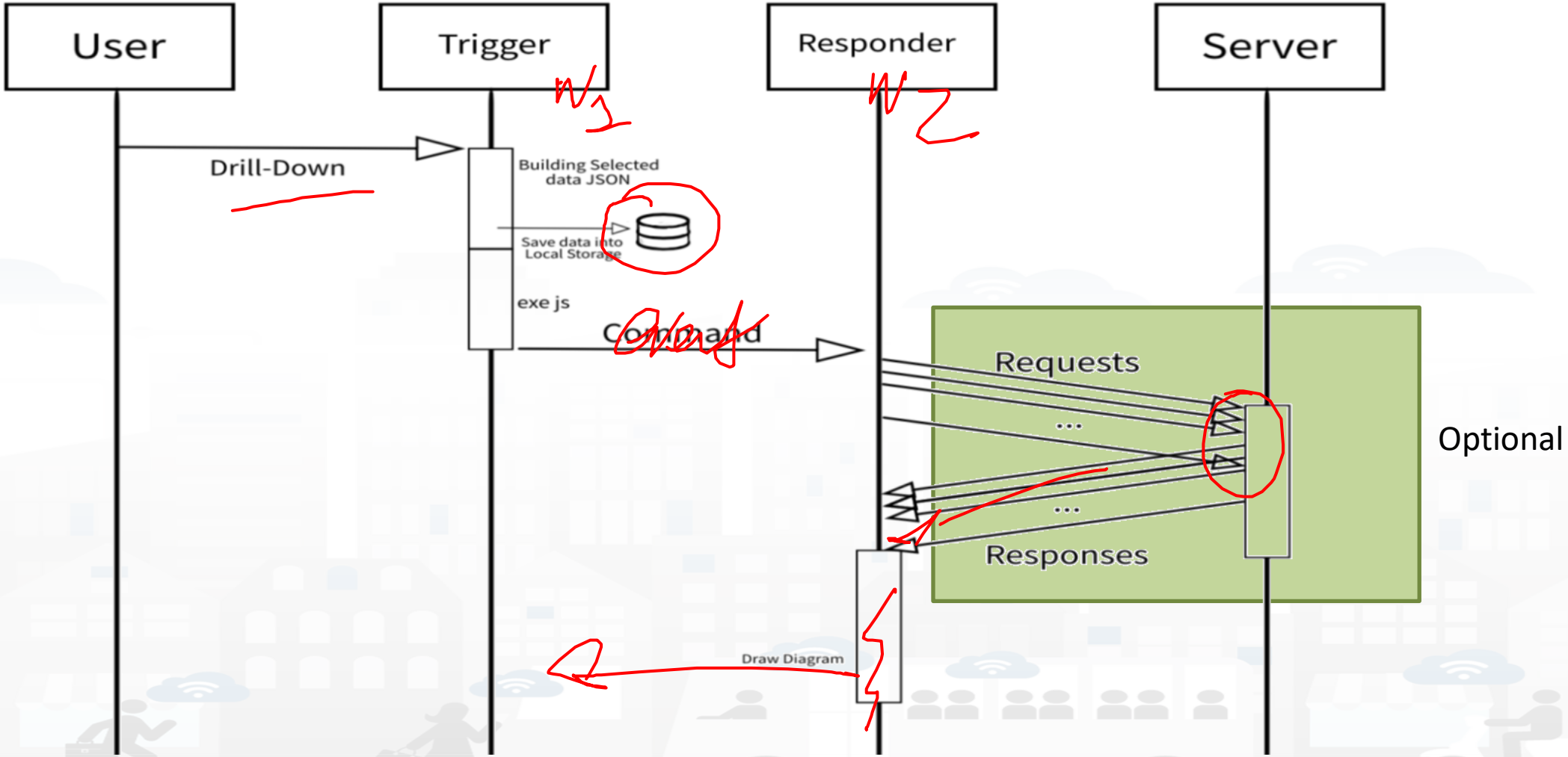
Composition

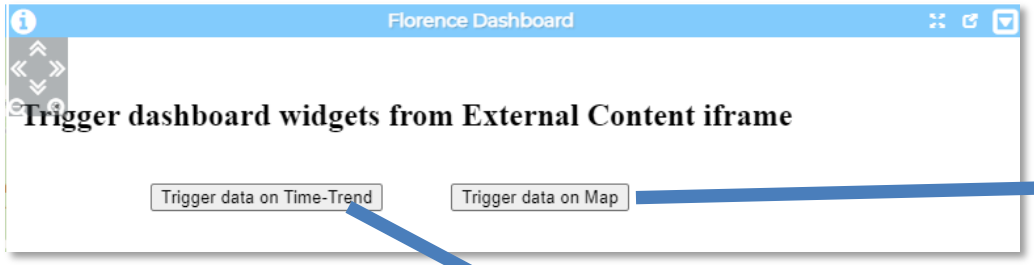


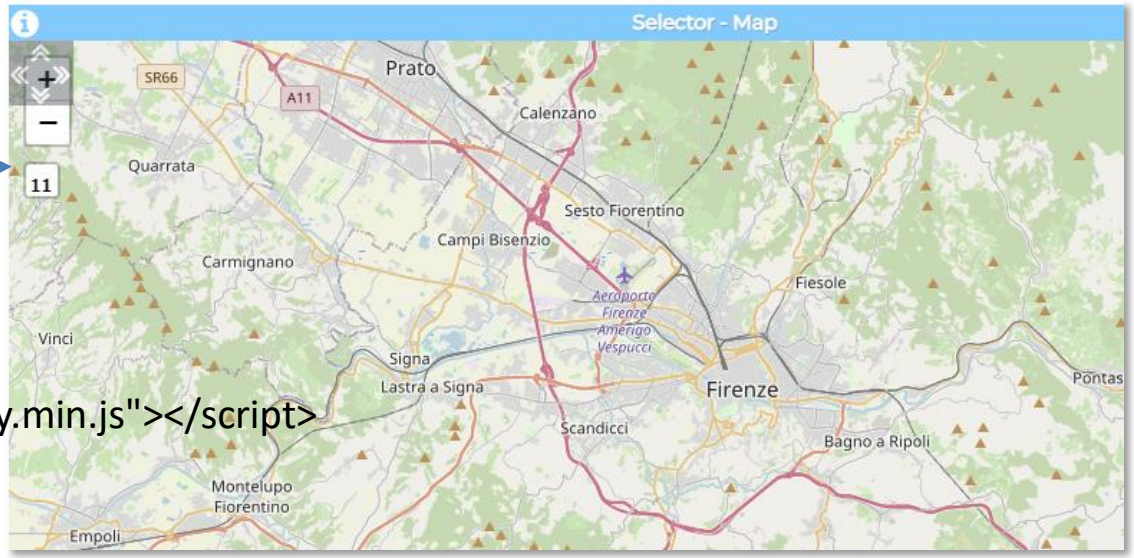
User Actions, Triggers on Risponder

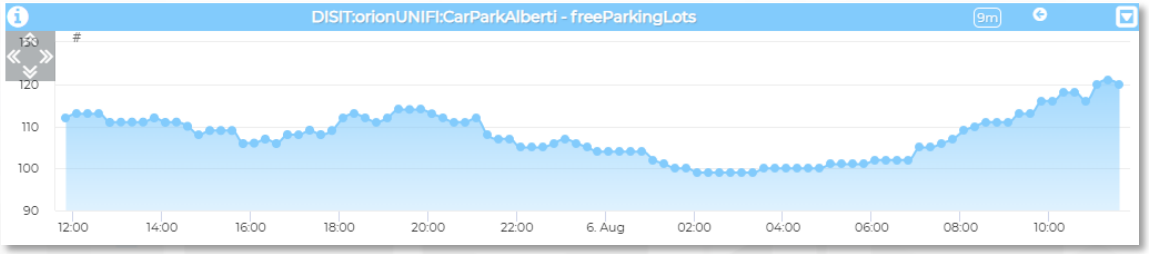


User Actions, Triggers on Risponder







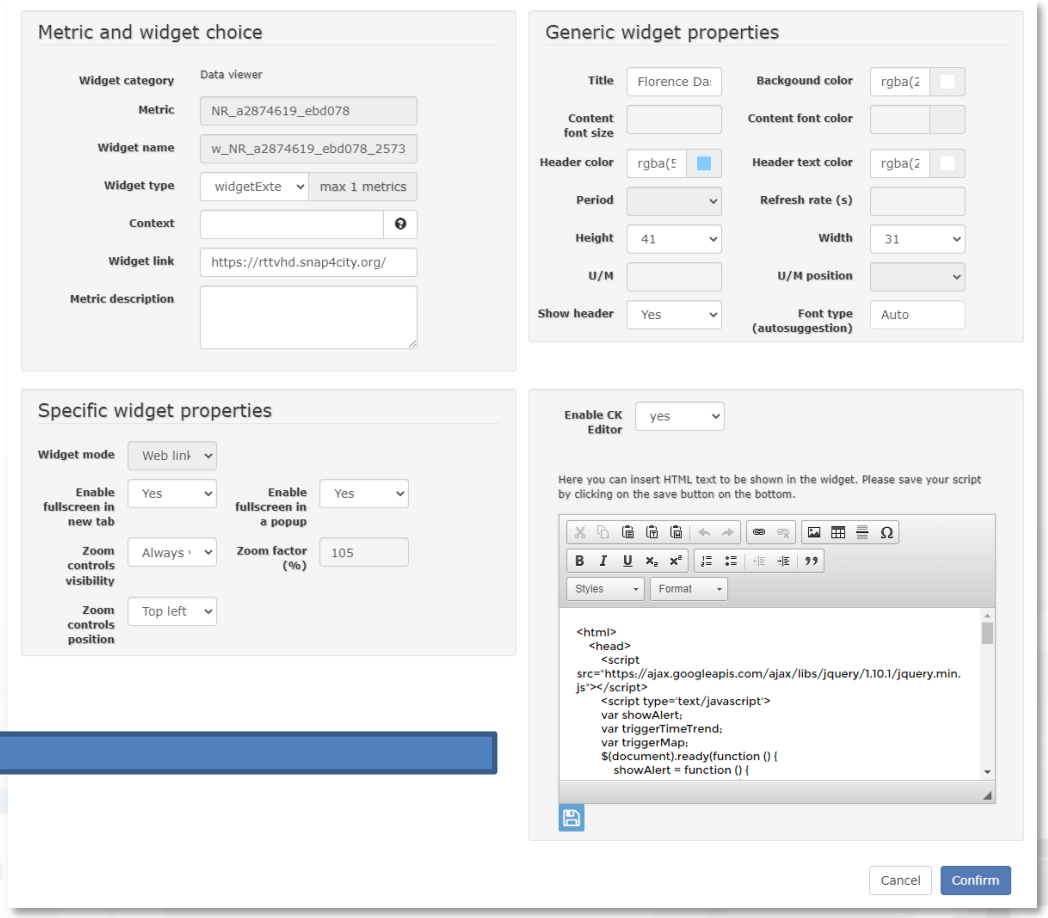



```
<html>
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js"></script>
    <script type='text/javascript'>
            …….
    </script>
  </head>
  <body>
    <h2>Trigger dashboard widgets from External Content iframe</h2>
    <div>
      <!-- <button onclick="showAlert()">Alert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

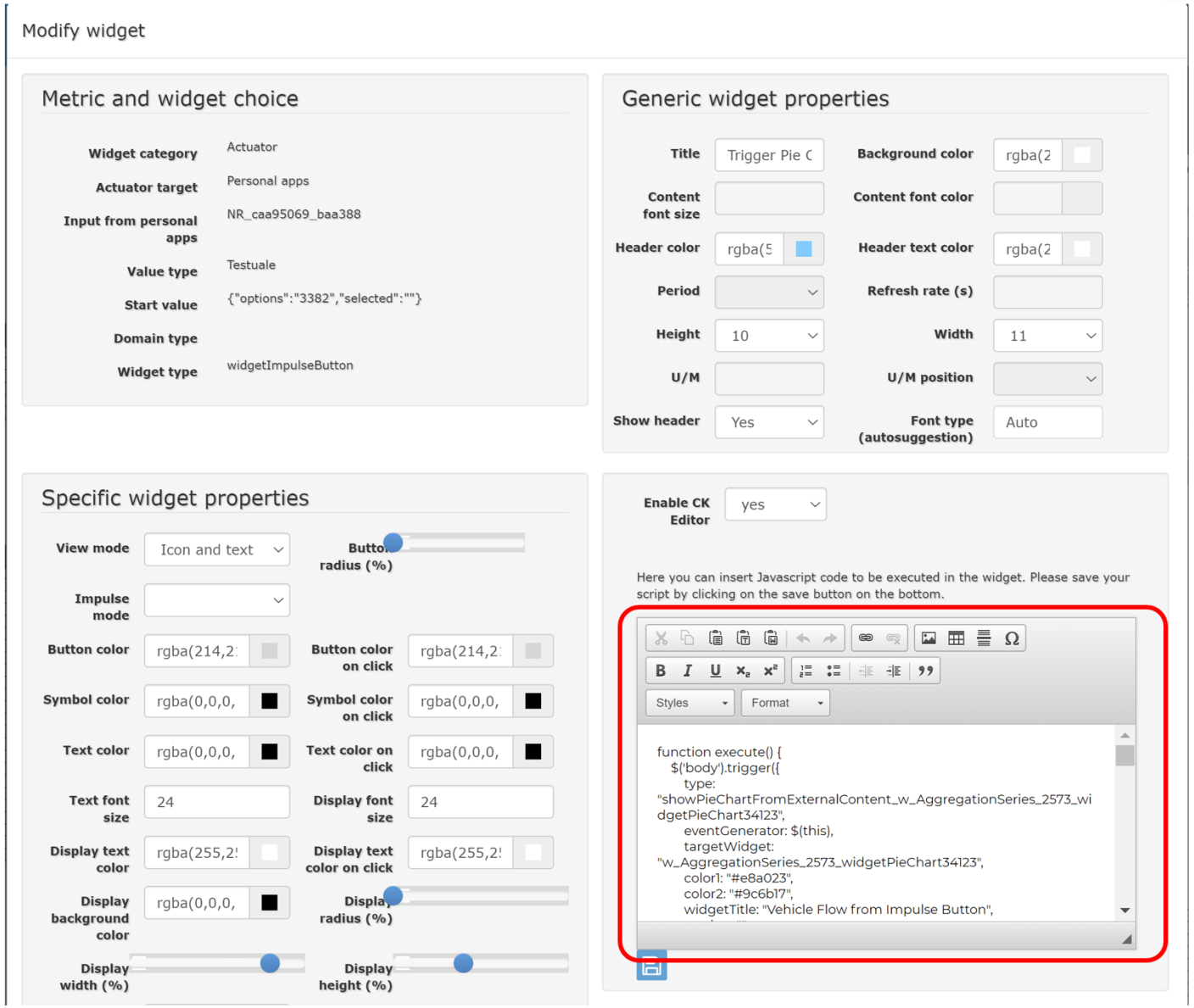
```
<script type='text/javascript'>
var showAlert;
var triggerTimeTrend;
var triggerMap;
$(document).ready(function () {
    showAlert = function () {
        var myText = "Test alert";
        alert (myText);
    }
    $('#triggerTTrend').click(function (event) {
        .................
        parent.$('body').trigger({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

Enforcing HTML and JavaScript on MoreOptions of the External Content Widget



Formalization of SSBL on In Widget More Options

- CK EDITOR



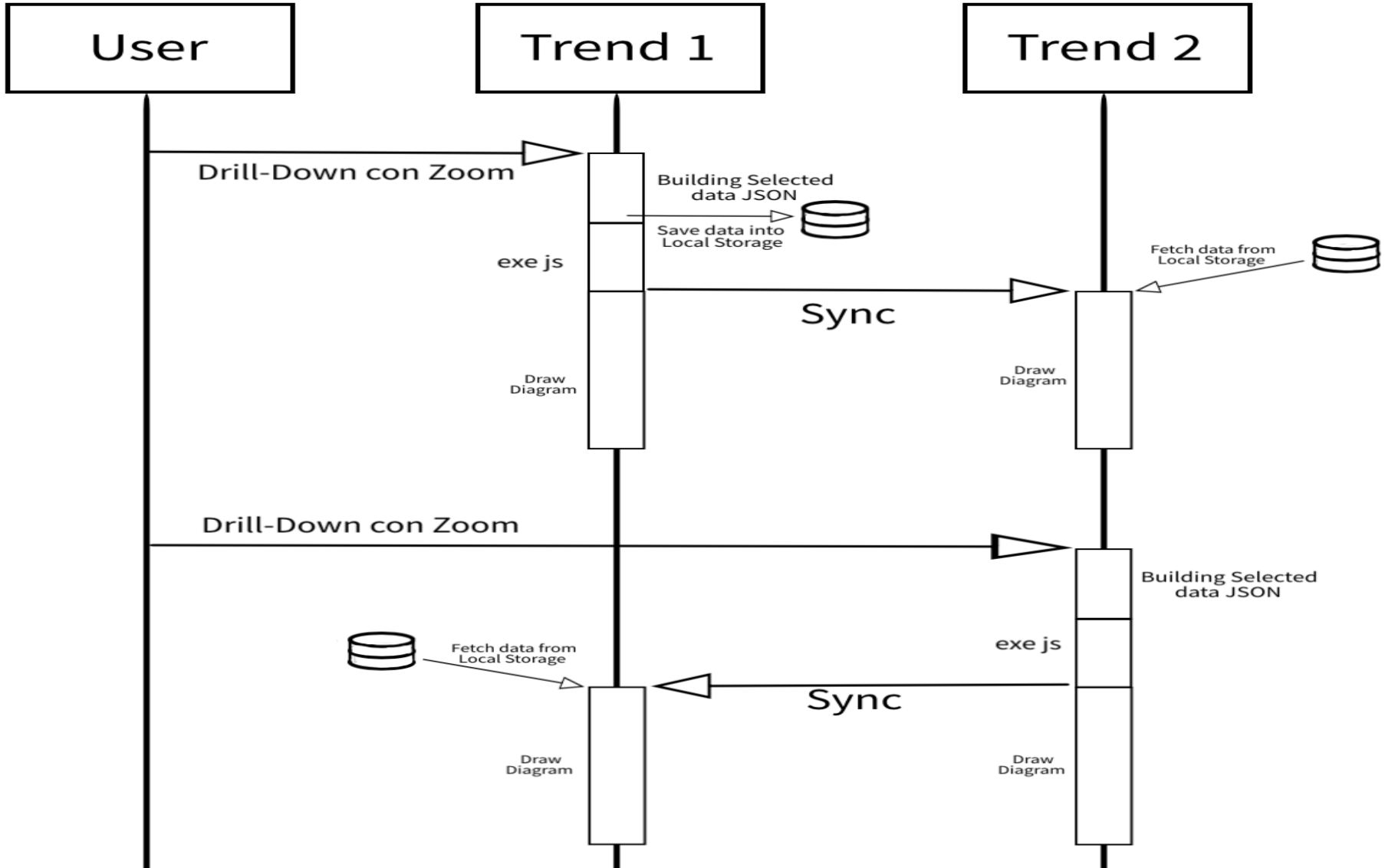
Trigger map

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

Synchronization Multiple Time Trends



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.

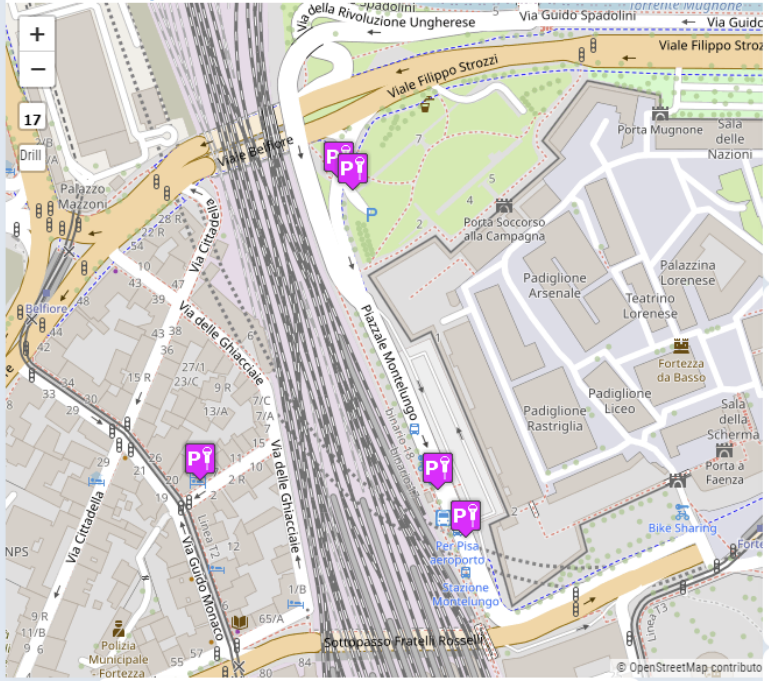
Client Side Business Logic Example

example

First BI Example

Mon 10 Apr 12:00:40

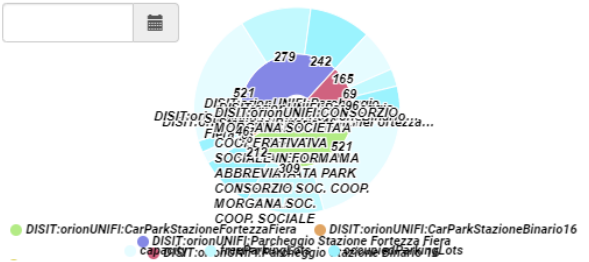
Selector - Map



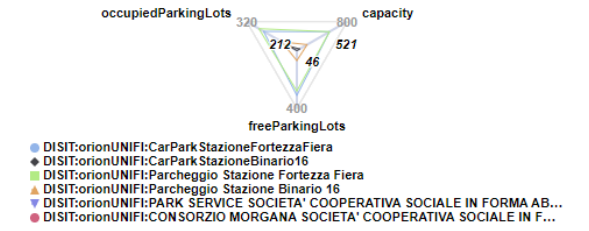
Selector

- Car_park
- metrotrafficsensor
- Air_quality_monitoring_station
- Weather_sensor

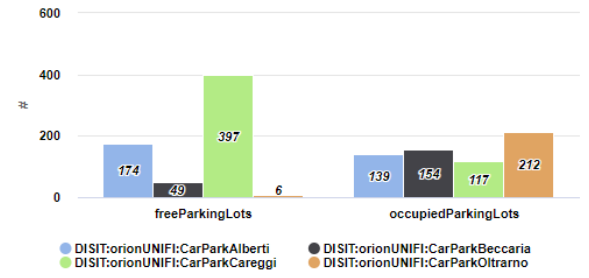
Pie Chart 9m



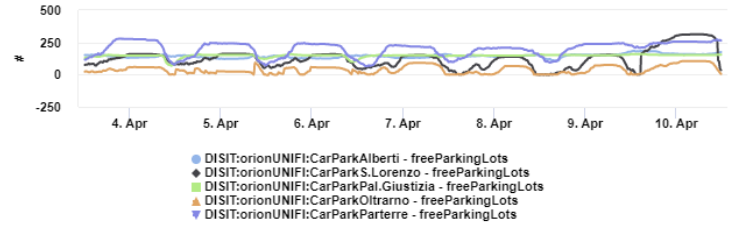
Radar Series 4m



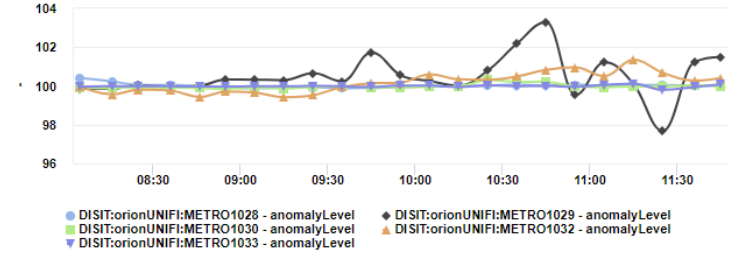
Bar Series 4m



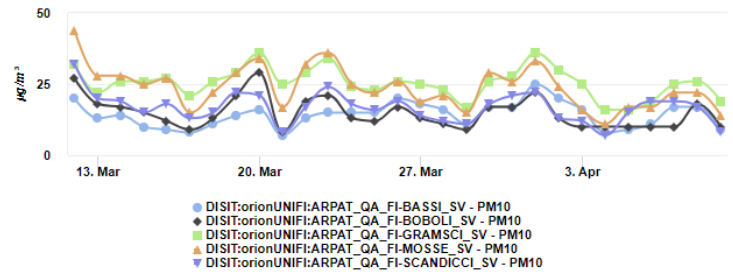
Time Trend Comparison 4m



Time Trend Comparison 4m



Time Trend Comparison 4m

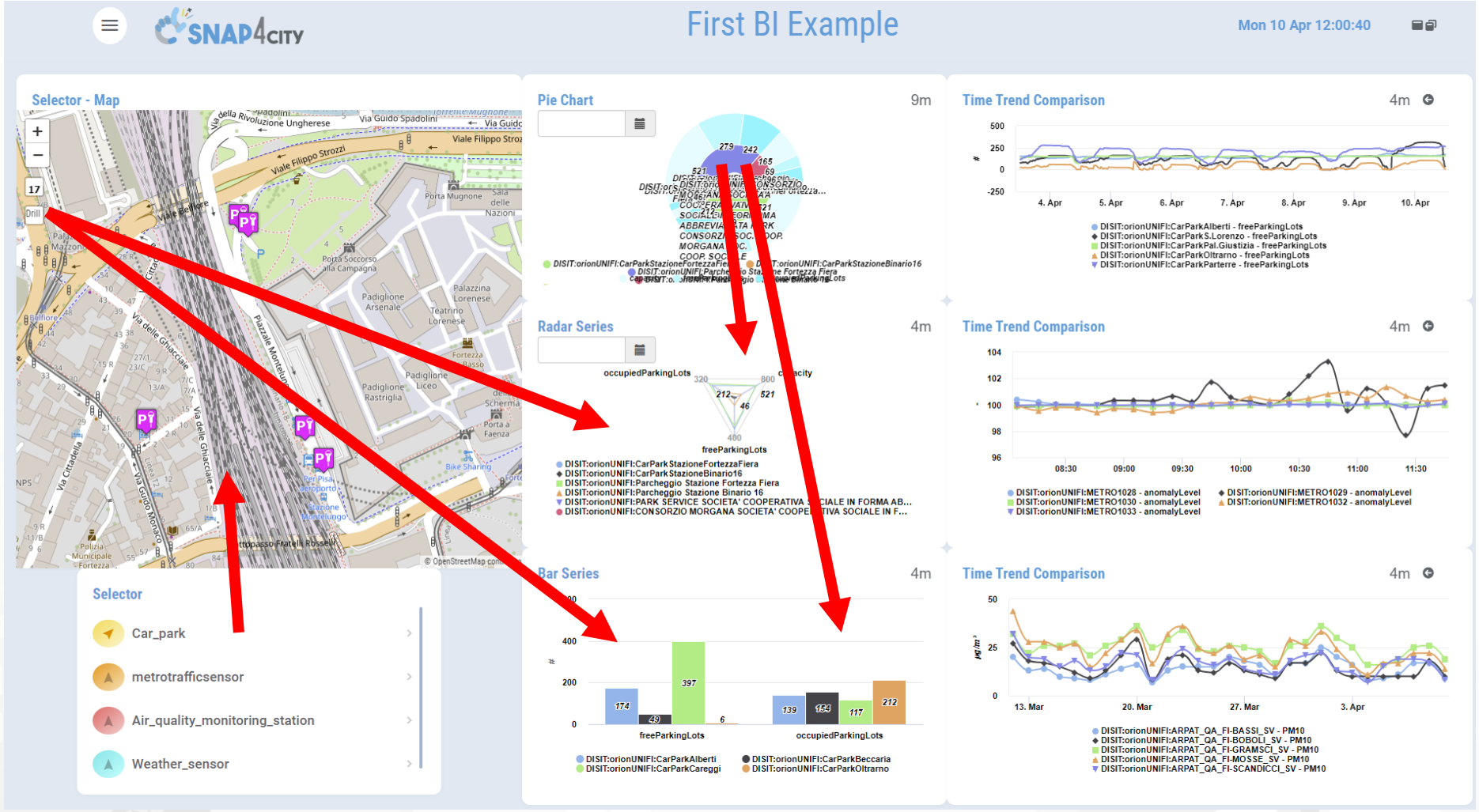


https://www.snap4city.org/dashboardSmartCity/view/Gea.php?iddasboard=MzcyNA==

Example: From Map to Graphs (spatial drill down)

1) Select the area of interest on map
2) Select the sensors kind of interest
3) Drill down on map

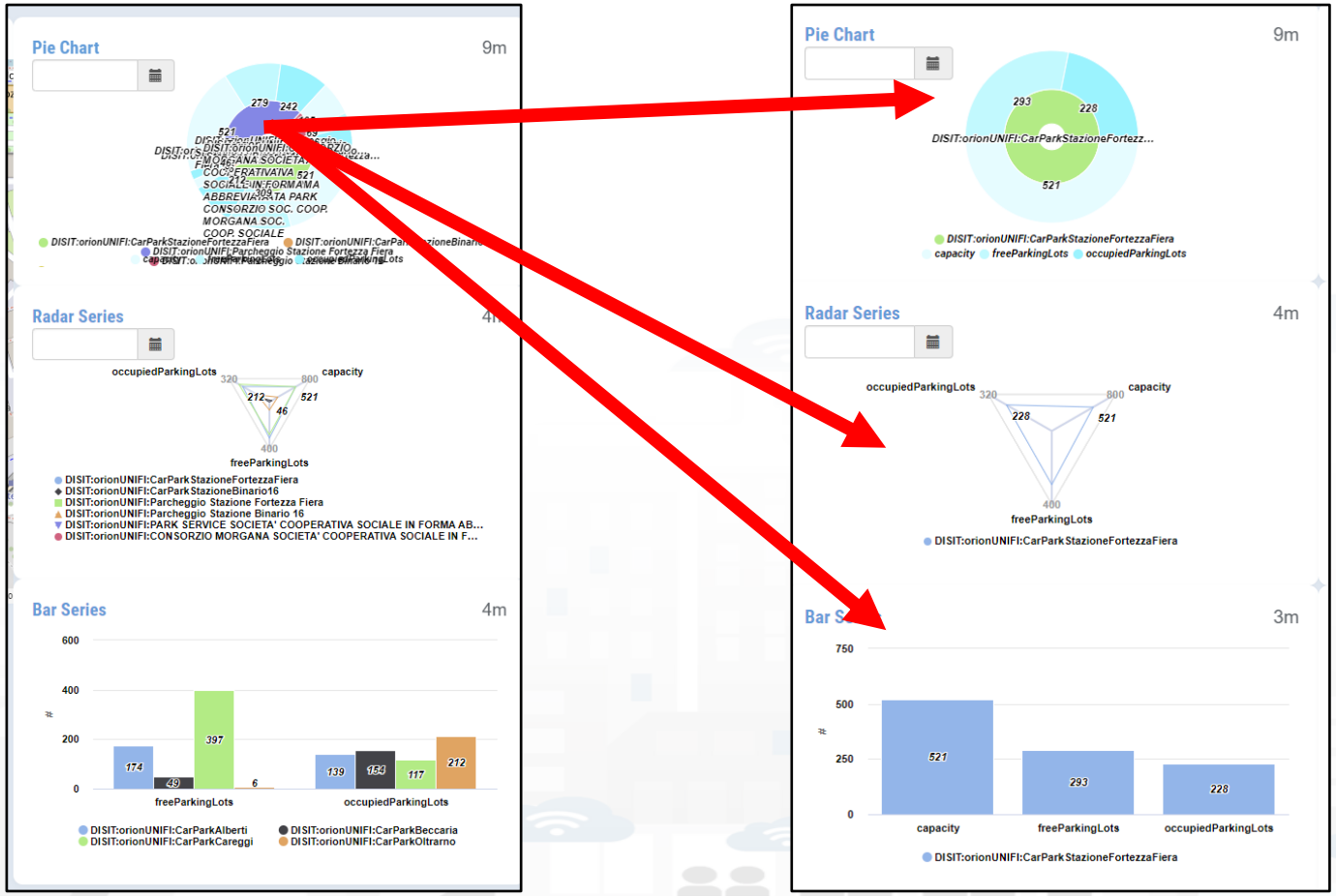
4) The JavaScript CSBL on Map will send data to the programmed Widgets. In this case, arrowed in RED



Example: From Data Graphs to Graphs (drill down)

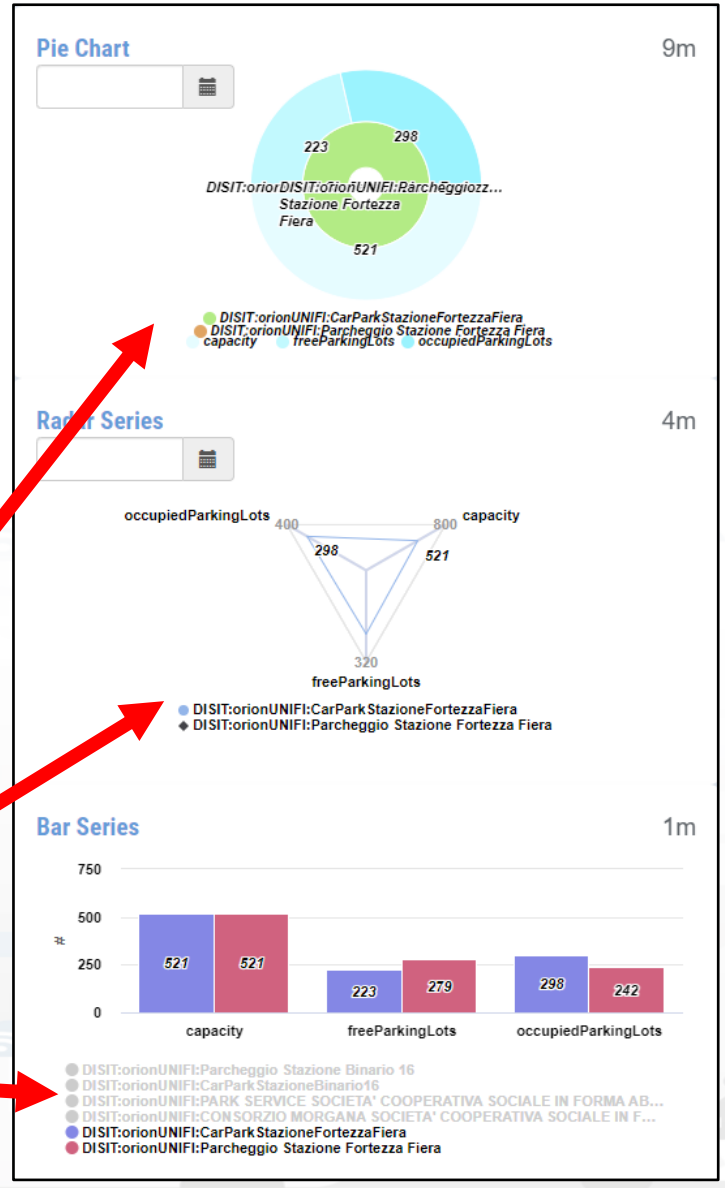
1) Click on the Donut element

2) The JavaScript CSBL on the Donut Widget will send commands to the programmed Widgets to focus on selection, as highlighted by the red arrows

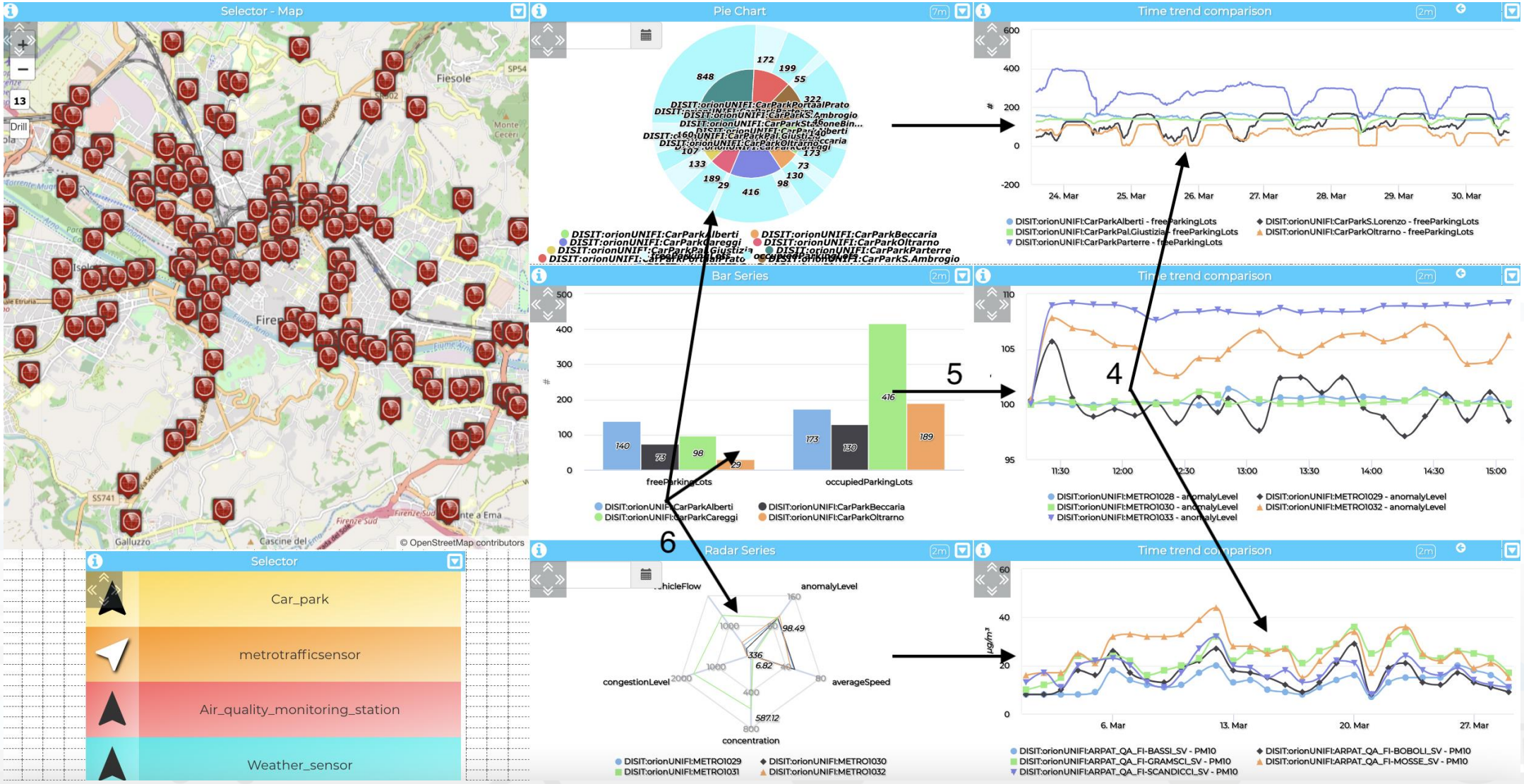


1) Click on the Legenda of Bar Series

2) The JavaScript CSBL on the Bar Series will send commands to the programmed Widgets to remove the unselected devices, as highlighted by the red arrows



Other Actions



Example Triggers

IN and 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
	Send Reset Drill-Down
	Click on a single time instant, providing time stamp, SURI and value name
widgetMap (multidatamap)	Click on a generic point on the map, providing coordinates (under development, currently it only works for SSBL)
	Click on a PIN, providing coordinates and ServiceURI of the clicked PIN
	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, entity name (from which the SURI can be reconstructed) value name, value type and value unit
	click on a sector that identifies a device ID or MyKPI ID, providing: value, timestamp, entity name (from which the SURI can be reconstructed) value name, value type and value unit
	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, entity name (from which the SURI can be reconstructed) value name, value type and value unit
	Click on legend, providing the visibility status of each metric/SURI
widgetRadarSeries	Click on a radar axis related to a specific metric of a specific device, providing: value, timestamp, entity name (from which the SURI can be reconstructed) value name, value type and value unit
	Click on legend, providing the visibility status of each metric/SURI
widgetCurvedLineSeries (multi series)	Drill-Down on time interval selection (zoom), providing: start and end time stamp, and list of SURI. It is also possible to program the synchronization of multiple widgetCurvedLineSeries widgets.
	Click on a single time instant, providing: time stamp and list of objects including SURIs and related entity names and value names
	Click on legend, providing the visibility status of each metric/SURI
	Send Reset Drill-Down
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
	Click on minus and plus action
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
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.

Typical Triggered Events

- SURI, List of SURI
  - Variable: Value Name (or metric)
- DateTime: date and time instant
- DateTime Interval: fromdatetime, todatetime
- ResetCommand
- GPS Coordinates, Bounding Box: a couple of coordinates
- Action (status | value)
- Etc.

Examp. Responders

OUT and IN/OUT Widgets	Commands which are ready to be executed from Widgets according to JavaScript in some IN Widget
widgetPieChart	Receive a JSON containing a list of SURI, metric names and/or values, and show their corresponding values on a Pie Chart graph.
widgetRadarSeries	Receive a JSON containing a list of SURI, metric names and/or values, and show their corresponding values on a Radar/Kiviat 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.
	Receive reset zoom
widgetTable	Receive a JSON containing a list SURI, metric names and/or values, and show the corresponding time-series on a HTML static table.
	Receive start datetime, end datetime without change sources IDs
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.
	Receive start datetime, end datetime without change sources IDs
	Receive reset zoom
widgetDeviceTable	Receive a JSON containing a list of SURI representing 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
widgetKnob	Receive and show a value
widgetNumericKeyboard	Receive and show a value

Typical commands received

- SURI, List of SURI
  - Variable: Value Name (or metric)
- DateTime: date and time instant
- DateTime Interval: fromdatetime, todatetime
- ResetCommand
- MyKPI, List of MyKPI
- GPS Coordinates, Bounding Box: a couple of coordinates
- Action (status | value)
- Etc.

CSBL: Useful Functions

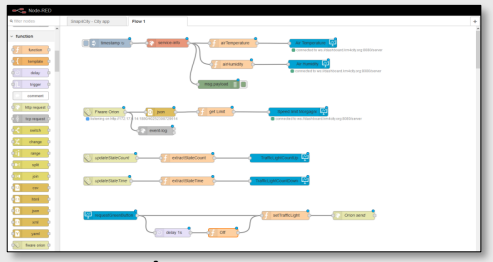
functions on Actions JavaScript segments:

- 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.

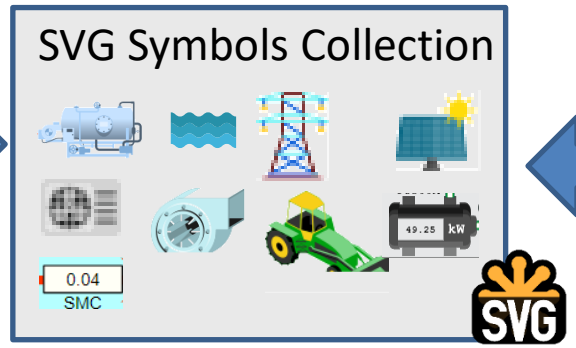
Develop: Client Side Business Logic Dashboards with Synoptics

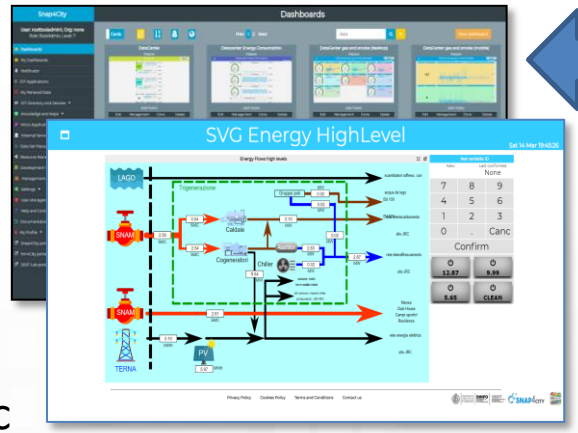
Custom Widget / Synoptic / PIN Development



IOT Applications

Knowledge and Storage Data from the Field and City

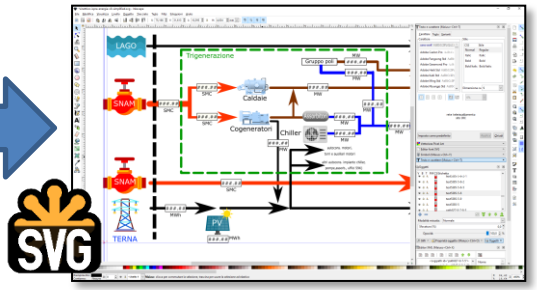




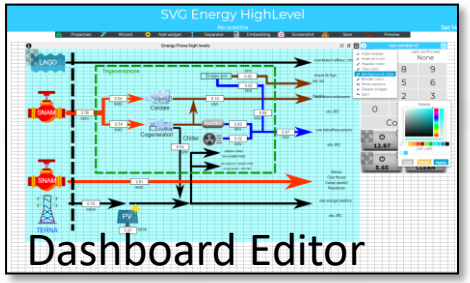
Public Dashboard Collection

My Own Dash/App

Inkscape editor on your computer



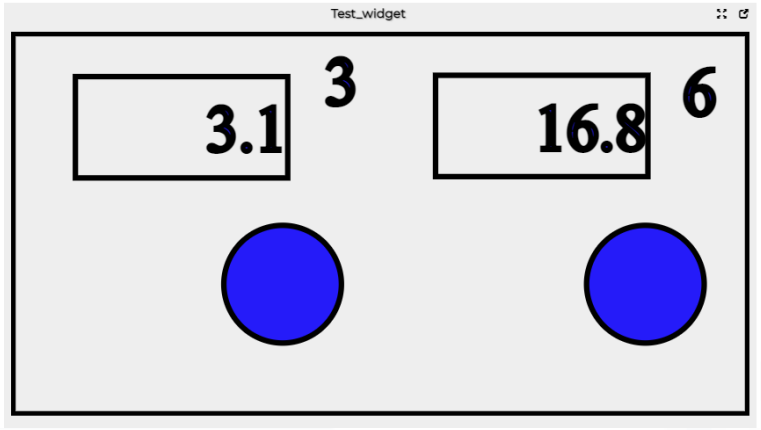
Create, save a Custom Widget in SVG



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 or sensors
4. Create on Dashboard a Widget based on Synoptic HLT such as Ext. Srv.:
   - https://www.snap4city.org/synoptic/v2/synoptic.html?id=xxxx

Synoptic for Client Side Business Logic



In the SVG puts some code:

-- at the button for example OnClick(), triggering an event

-- at the text box to write the value in the box, for example:

TextBoxWrite3(……)

TextBoxWrite6(…..)

In the JavaScript of the External Content Widget hosting the SVG html code

The programmer can:

-- exploit the functions

TextBoxWrite3(……)

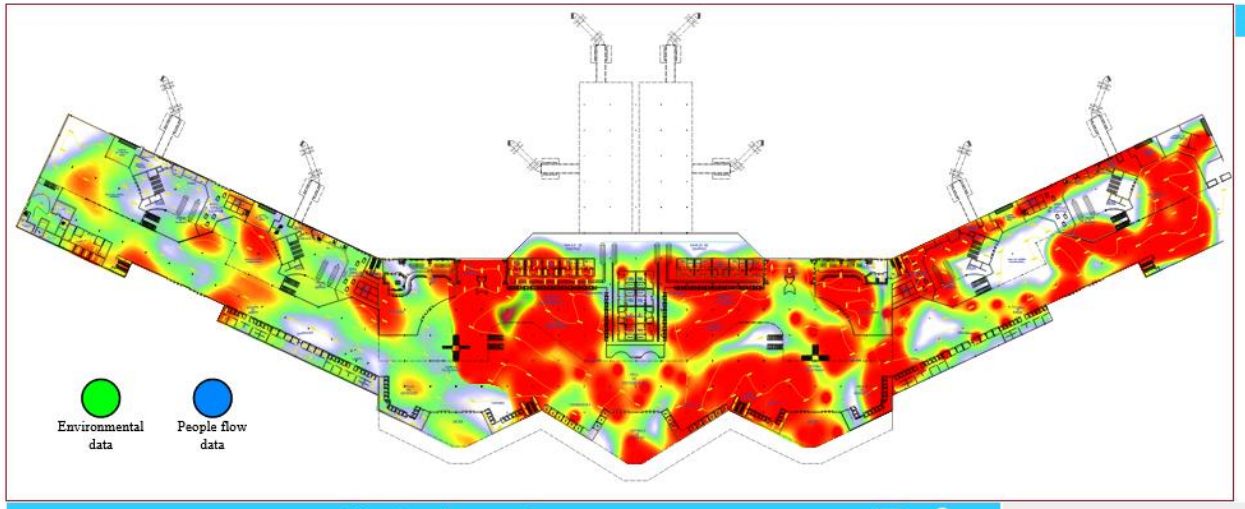
TextBoxWrite4(……)

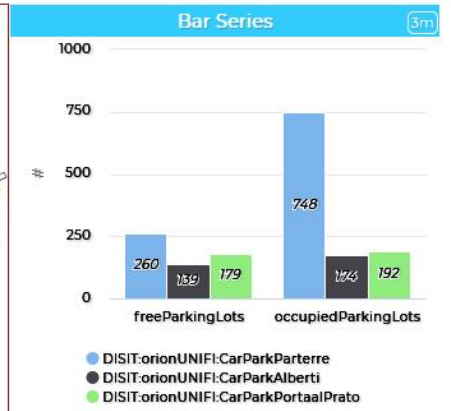
-- receive the Triggered event and

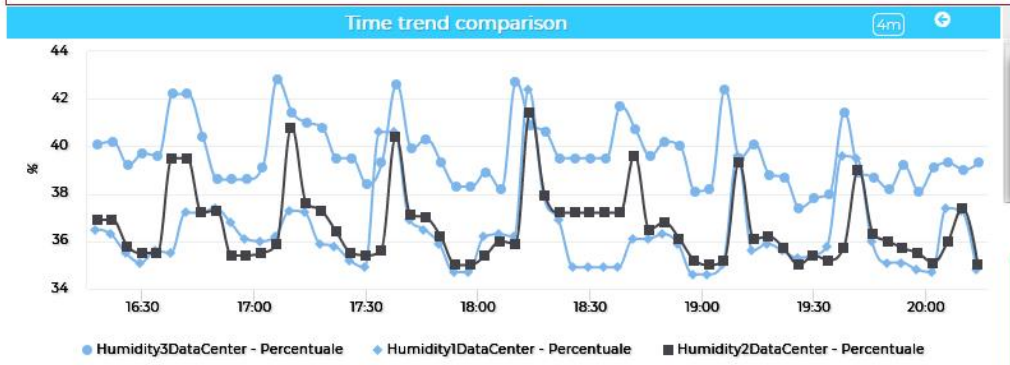
write a JavaScript with a corresponding action

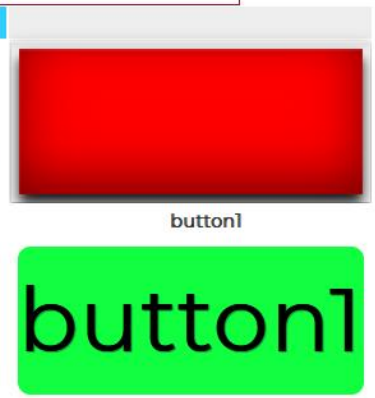
The same SVG may have some elements working with respect to Server, IoT App, etc., and also with Client Side Business Logic

svg_embed1









Privacy Policy Cookies Policy Terms and Conditions Contact us

https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=Mzg0OA==

TOP

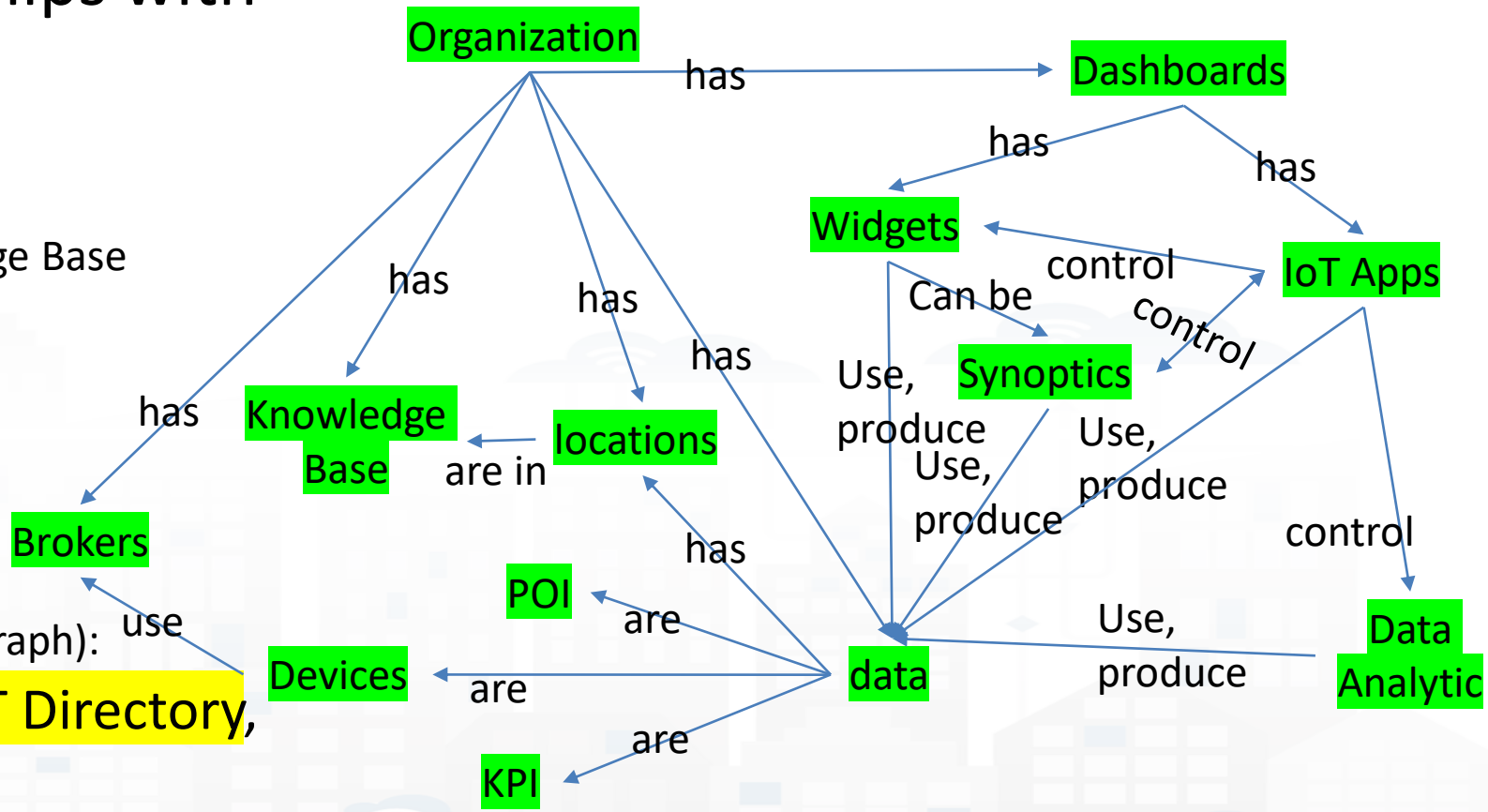
Design and Control of Smart Applications

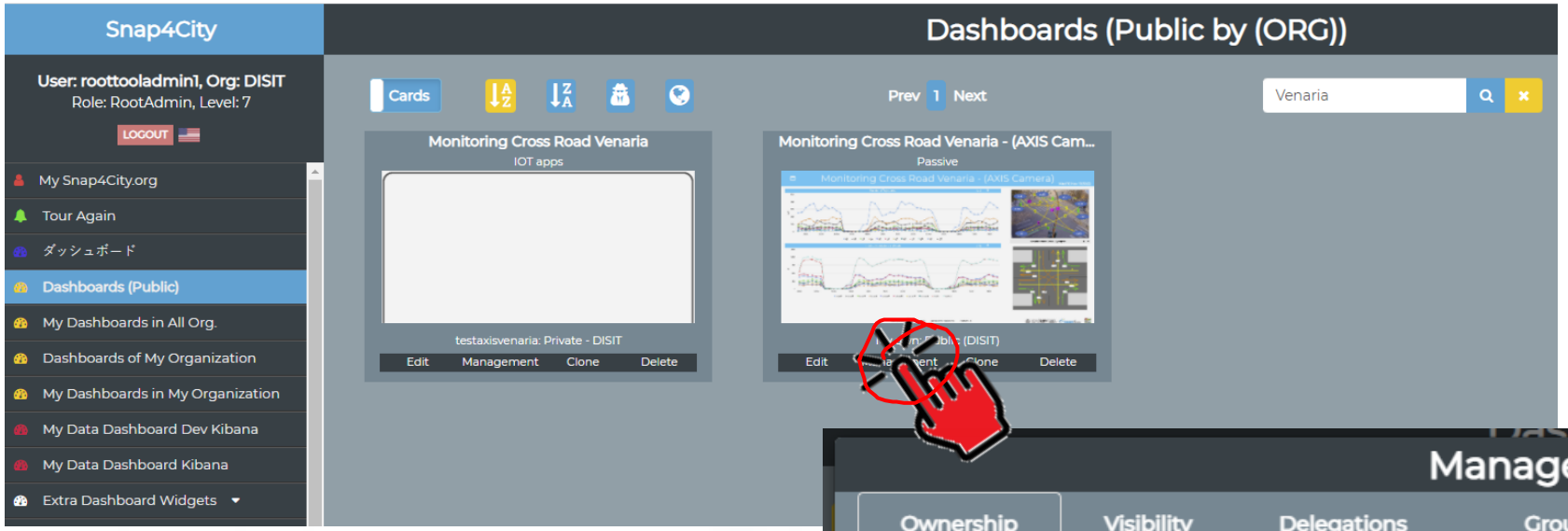
only for user with RootAdmin role

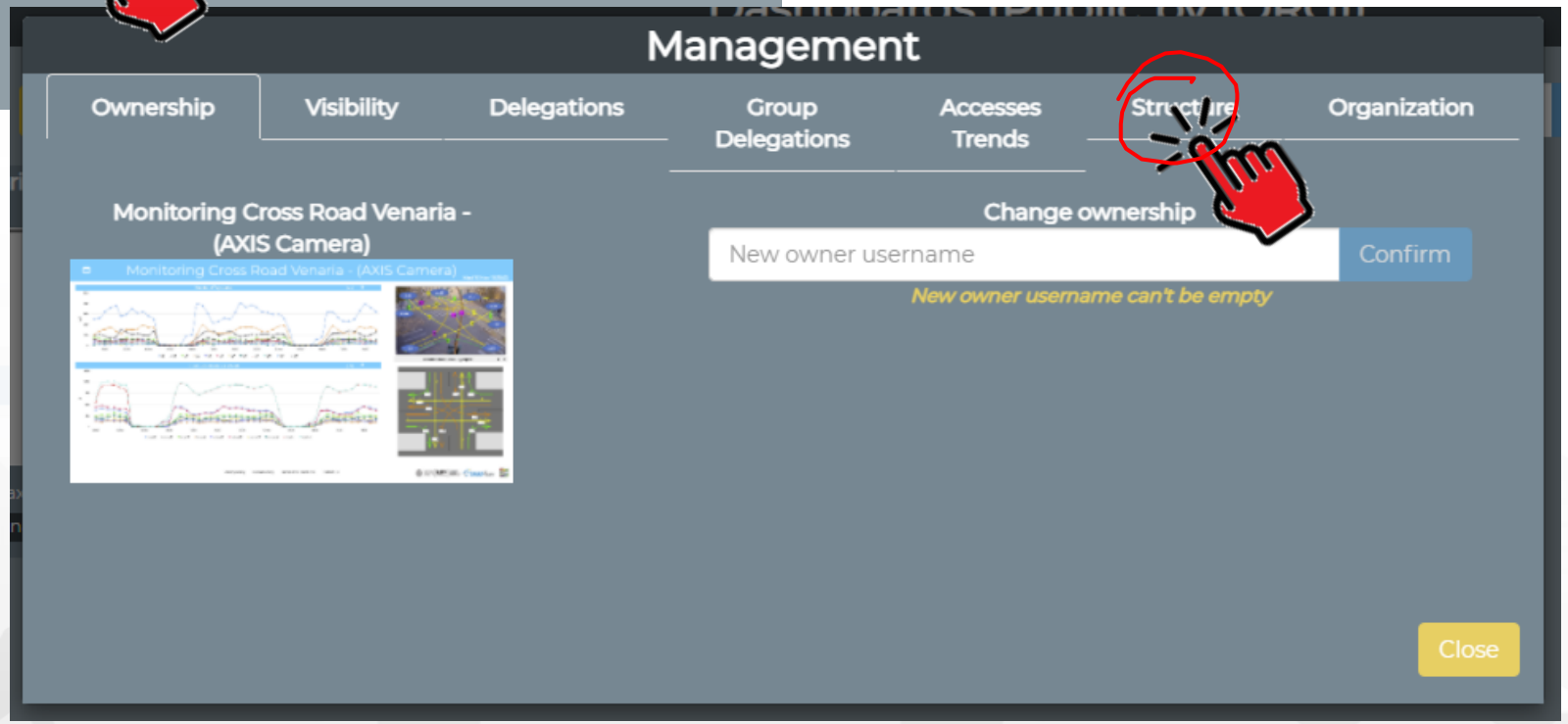
partially accessible also for all Dashboard owners

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

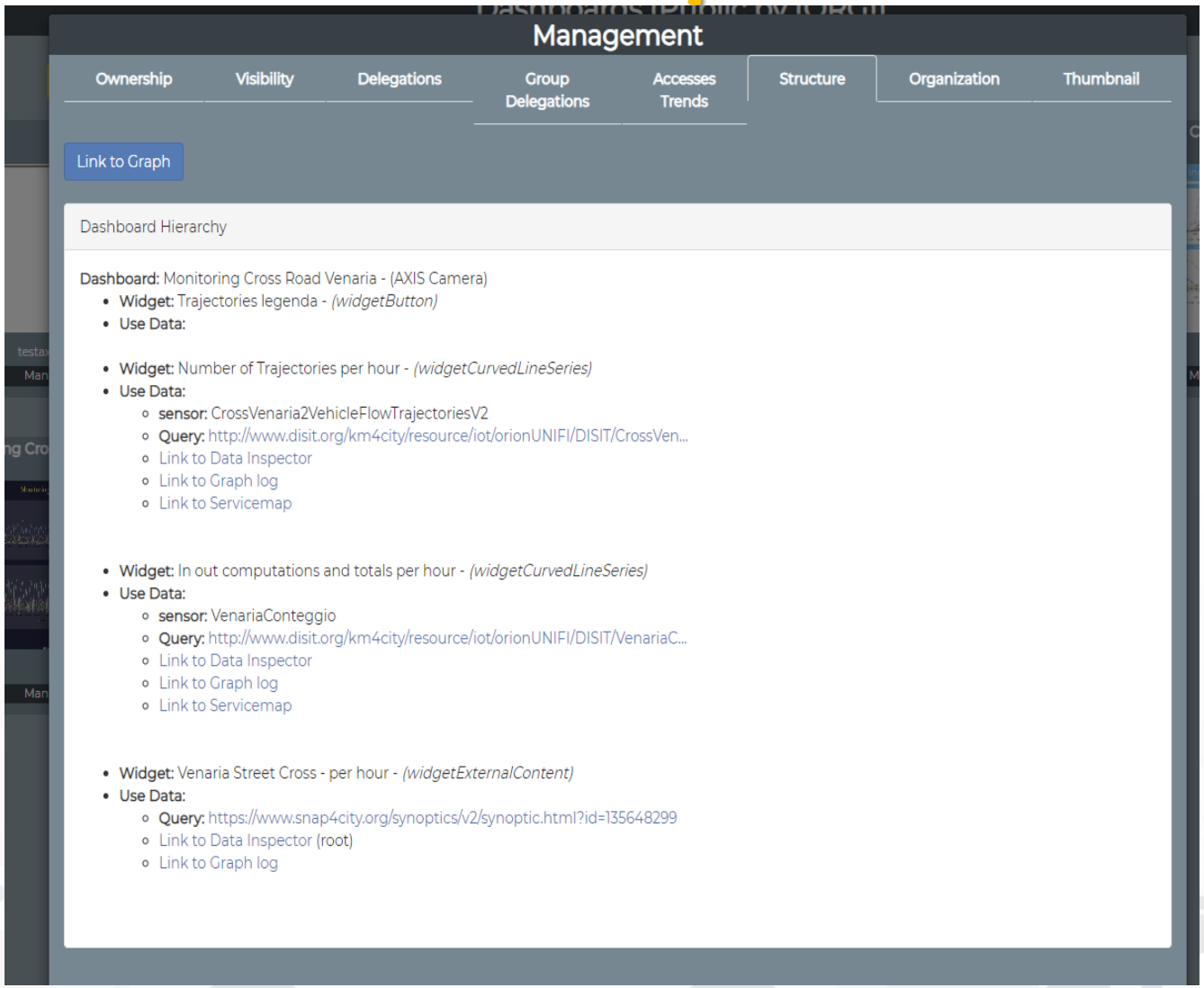



Snap4City
User: roottooladmin1, Org: DISIT
Role: RootAdmin, Level: 7
LOGOUT
My Snap4City.org
Tour Again
ダッシュボード
Dashboards (Public)
My Dashboards in All Org.
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
My Data Dashboard Kibana
Extra Dashboard Widgets
Dashboards (Public by (ORG))
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
Edit Clone Delete


Management
Ownership
Visibility
Delegations
Group Delegations
Accesses Trends
Organization
Monitoring Cross Road Venaria - (AXIS Camera)
Change ownership
New owner username
Confirm
New owner username can't be empty
Close

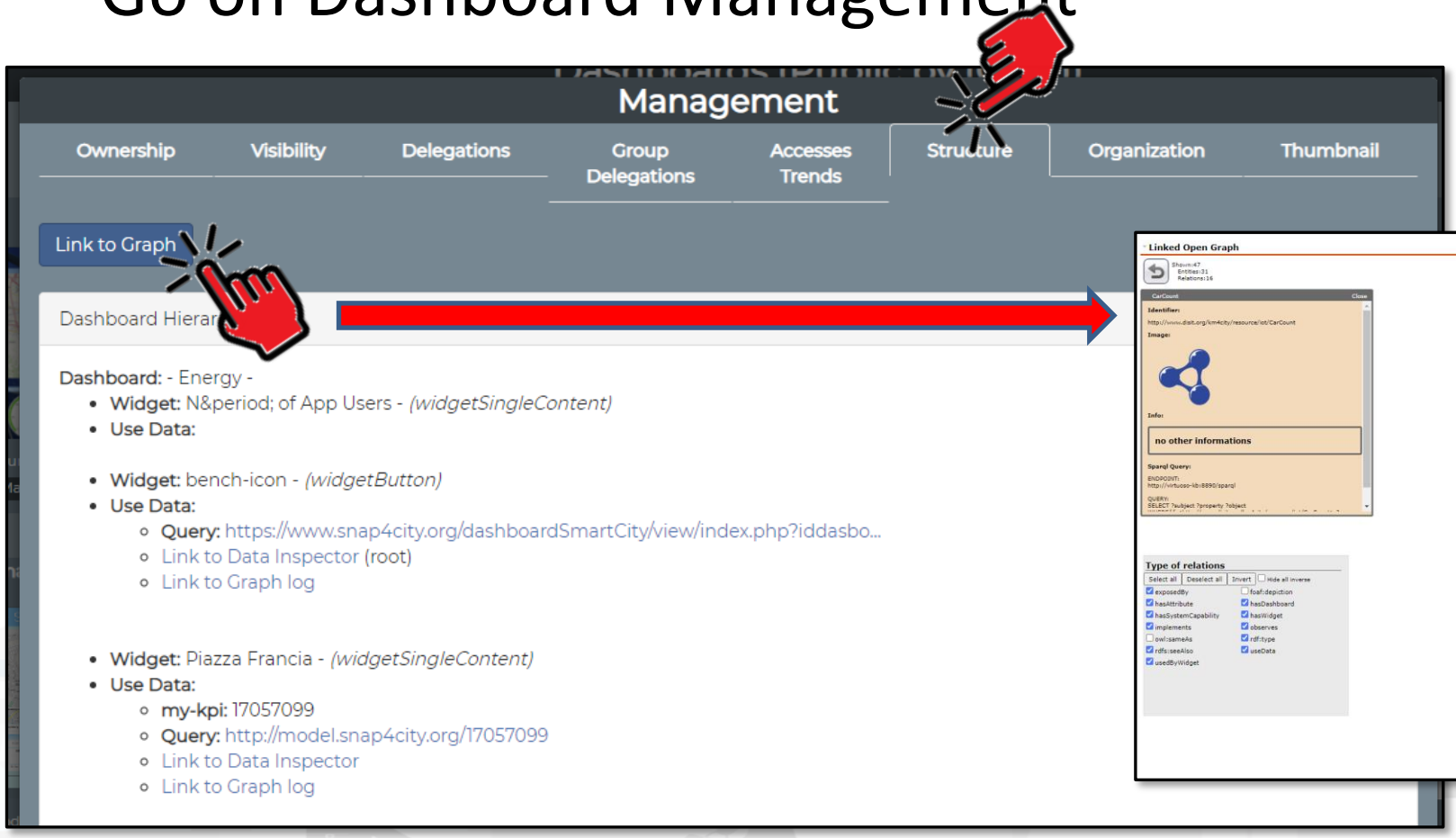
Dashboard Structure and Components

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

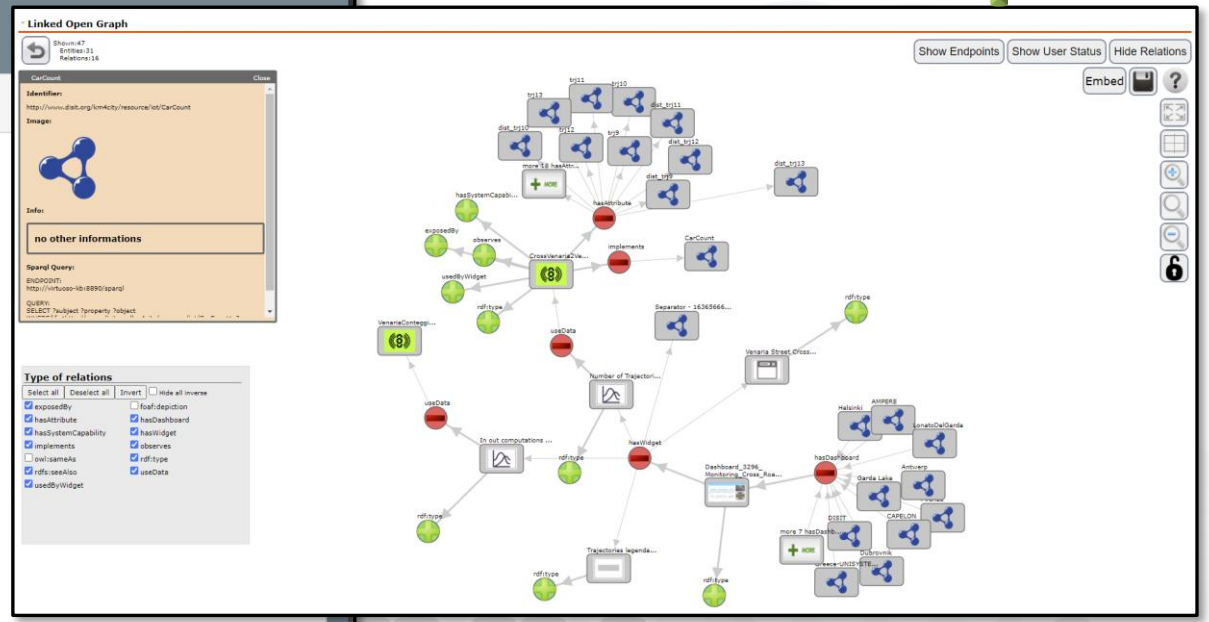


For All Dashboard owners: Graph and Structure

- Go on Dashboard Management



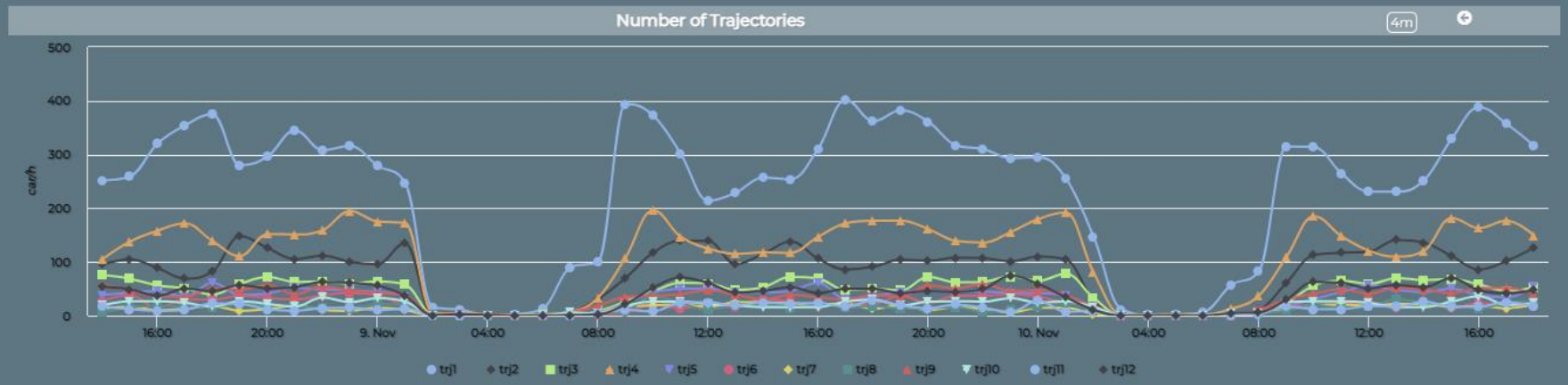
LOGraph

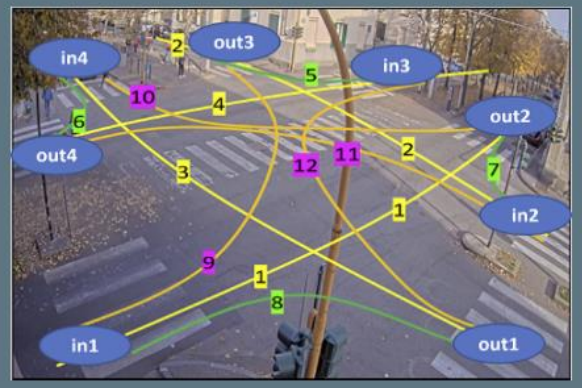


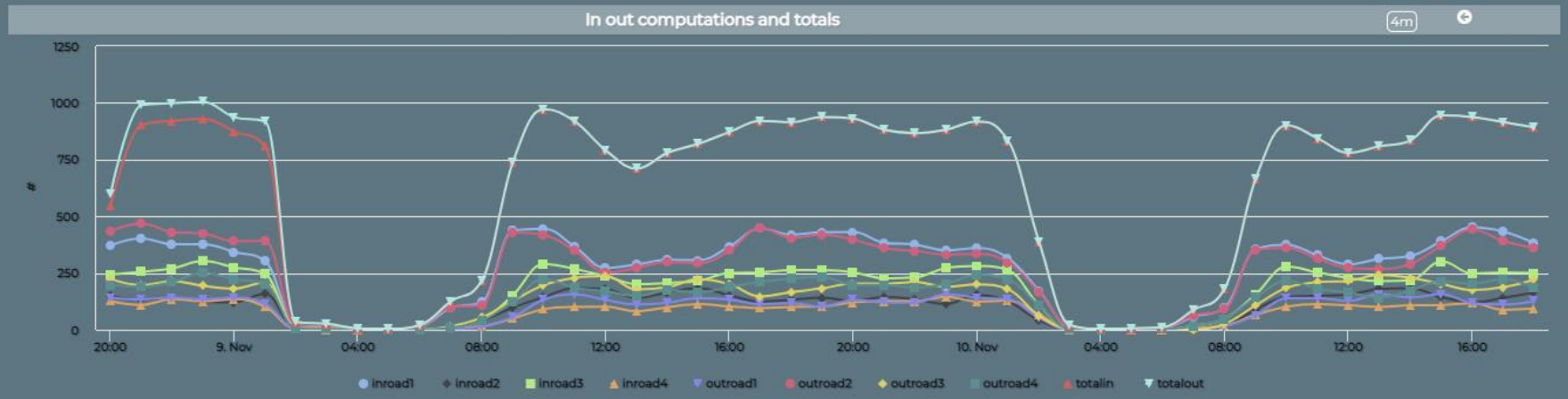
Venaria Reale

Monitoring Cross Road Venaria - (AXIS Camera)

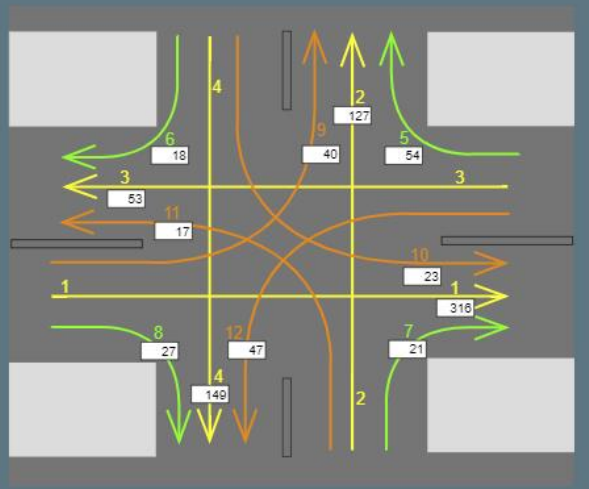
Wed 10 Nov 18:50:53





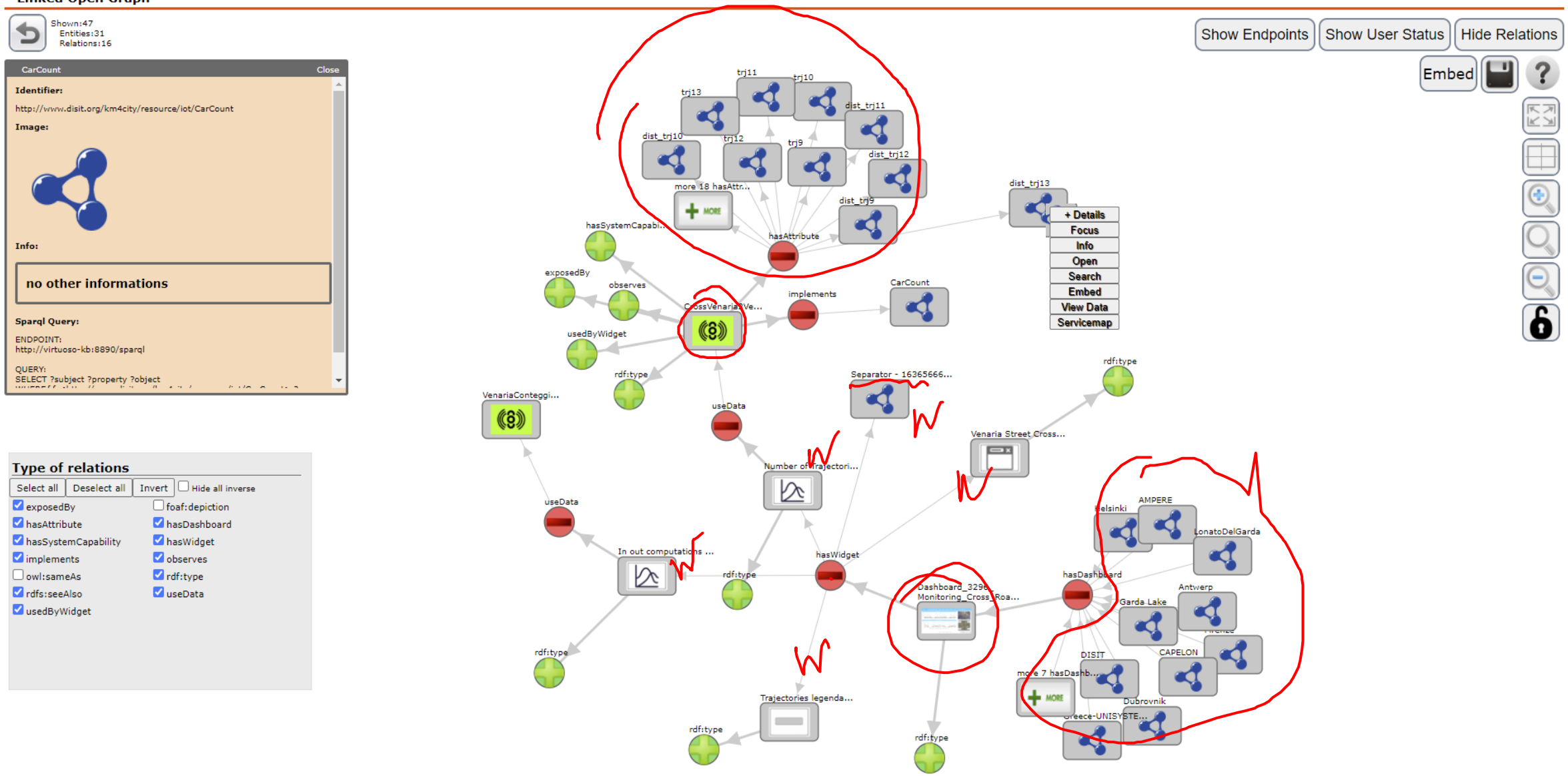


Venaria Street Cross - Synoptic

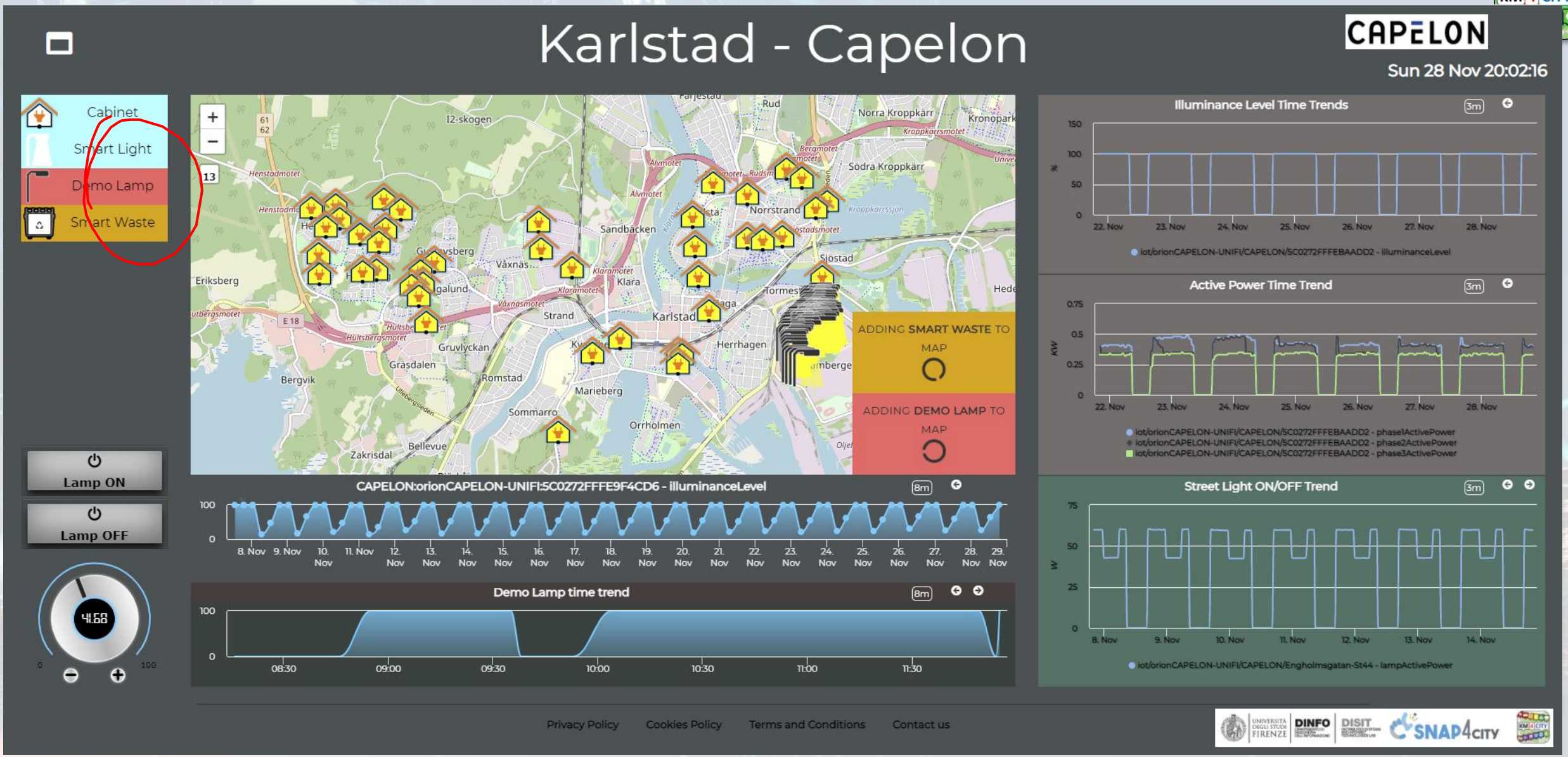


https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MzI5Ng==

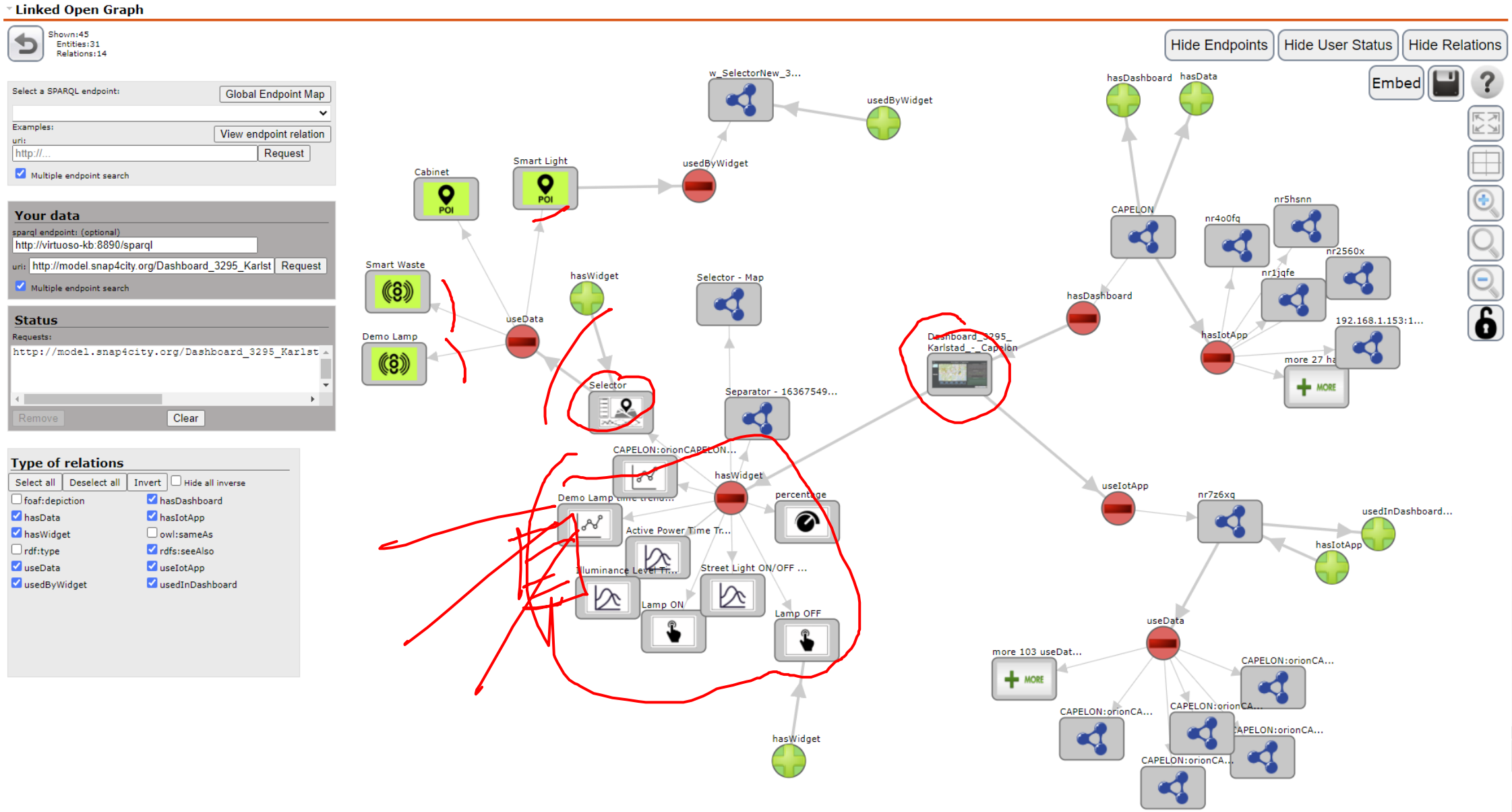
Linked Open Graph



Karlstad Street Lights



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:
uri:
View endpoint relation
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
w_SelectorNew_3...
usedByWidget
Cabinet
POI
Smart Light
POI
usedByWidget
Smart Waste
hasWidget
Selector - Map
Demo Lamp
useData
Selector
Separator - 16367549...
CAPELON:orionCAPELON...
hasWidget
Demo Lamp time trend...
percentage
Active Power Time Tr...
Illuminance Level Tr...
Street Light ON/OFF ...
Lamp ON
Lamp OFF
hasWidget
hasDashboard
hasData
CAPELON
nr4o0fq
nr5hsnn
nr2560x
nr1jqfe
hasDashboard
hasIotApp
192.168.1.153:1...
more 27 ha...
MORE
Dashboard_3295_Karlstad_-_Capelon
useIotApp
nr7z6xq
usedInDashboard...
hasIotApp
useData
more 103 useDat...
MORE
CAPELON:orionCA...
CAPELON:orionCA...
CAPELON:orionCA...
CAPELON:orionCA...
CAPELON:orionCA...

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

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

AirQualityPM10-Tuscany

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

AirQualityPM2.5-Tuscany

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

Tuscany CAQI

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

Tuscany EAQI

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

Weather

AirHumidity-Tuscany

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

AirQualityO3-Tuscany

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

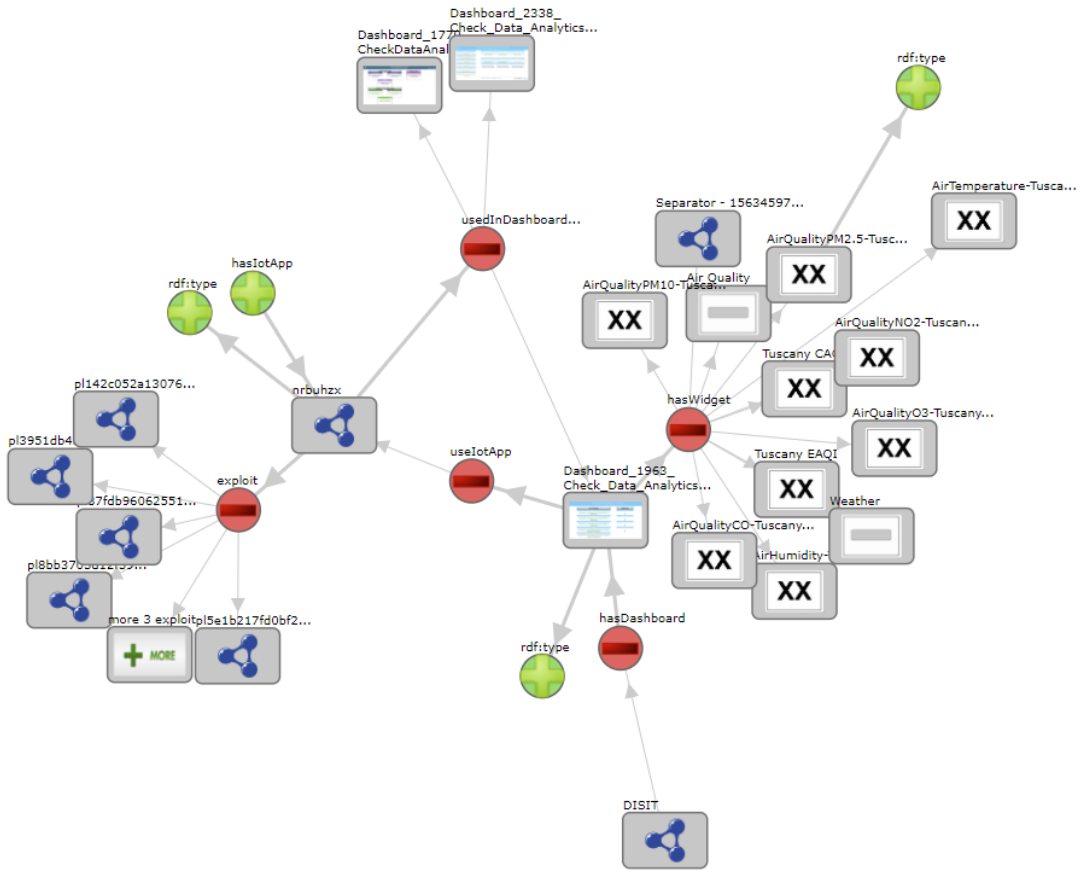
AirTemperature-Tuscany

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

AirQualityCO-Tuscany

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

Privacy Policy Cookies Policy Terms and Conditions Contact us



What is missing here and you can get from former course

What is missing here and you can find in the former course

https://www.snap4city.org/577

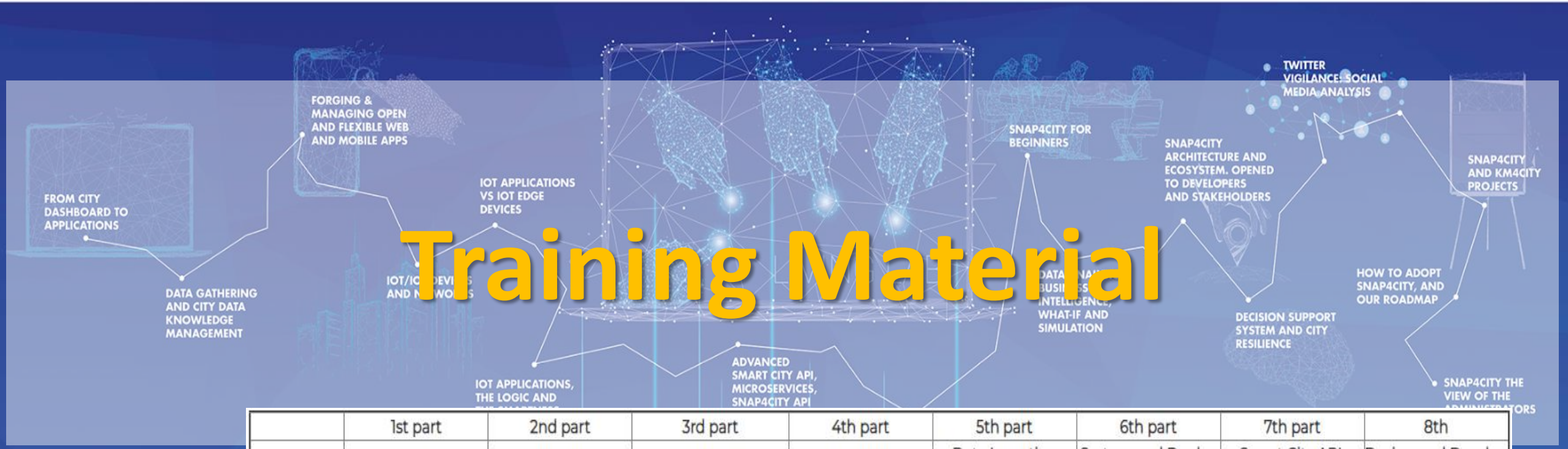
- Data Streams from partecipatori, Mobile App
- Data streams from Mobile vehicles and smart phones Devices
- Data Ingestion via Web Scraping
- Data stream from TV Cameras, TV Cam Manager
- From external API to Node-RED node/block automatically
- Social Media interoperability

- Another Complete Example
- BlockChain models and devices in Snap4City (new feature)
- Orion Broker:
  - Services/SrvPath and Multitenant
- External and Internal Brokers,
  - External Broker harvesting
- Managing Node-RED on edge from cloud
- More on: Security of Snap4City Stack from device to dashboards
- VM based installation of Snap4City
- ETL: Penthao Kettle interoperability

https://www.snap4city.org/944

On Line Training Material (free of charge)

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



Training Material

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

Note on Training Material

- **Course 2023**: https://www.snap4city.org/944
  - Introductionary course to Snap4City technology
- **Course** https://www.snap4city.org/577
  - Full training course with much more details on mechanisms and a wider set of cases/solutions of the Snap4City Technology
- **Documentation** includes a deeper round of details
  - Snap4City Platform Overview:
    - https://www.snap4city.org/drupal/sites/default/files/files/Snap4City-PlatformOverview.pdf
  - Development Life Cycle:
    - https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf
  - Client Side Business Logic:
    - https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf
- **On line cases and documentation**:
  - https://www.snap4city.org/108
  - https://www.snap4city.org/78
  - https://www.snap4city.org/426

Welcome: how to start using Snap4City for beginners

We suggest you:

Congratulations! You have really contributed to Snap4City and successfully passed all first levels!

You have reached a level in which you can contribute with competence to the city improvement and smartness. We hope you interested in helping other users in conquering higher levels on the city smartness ranking, and provising of smart services to all city users!
So that we could be interested in engaging and elevating your role in the Snap4City community as coordinator of thematic groups, for example on **Mobile APP development**, **Dashboard** on Mobility, **IOT Application** Development, etc., according to your preferences.

Please contact **paonesi@gmail.com** !

Share / Save ...

Add to your favorites

- TECHNICAL OVERVIEW: https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf
- Development Life Cycle: https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf
- Client-Side Business Logic Widget Manual: https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf
- Booklet Data Analytics, Snap4Solutions: https://www.snap4city.org/download/video/DPL_SNAP4SOLU.pdf

Please start a fully guided training cases:

- HOW TO: create a Dashboard in Snap4City
- HOW TO: add a device to the Snap4City Platform
- HOW TO: add data sources to the Snap4City Platform

Powered by www.km4city.org

FIWARE Node-RED

Sii-Mobility

Organization Groups

DISIT
- Developer
- Operativo

Updates on Tools

Training Course Snap4City - 2023 Edition new
drupaladmin

Snap4City Newsletter of April 2023 new
roottooladmin1

Dashboards (Public)

My Snap4City.org | Tour Again

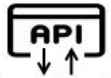
www.snap4solutions.org
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
Extra Dashboard Widgets
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Entity Directory and Devices
Resource Manager
Development Tools
Management
Decision Support Systems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

Privacy Policy | Cookies Policy

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

Snap4City: Smart aNalytic APp builder for sentient Cities and IOT

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



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

Username: paolo.disit

Search

Search

-Any-

Organization Groups

DISIT
- Developer
- Operativo

Updates on

2023 booklets

- Smart City

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

- Industry

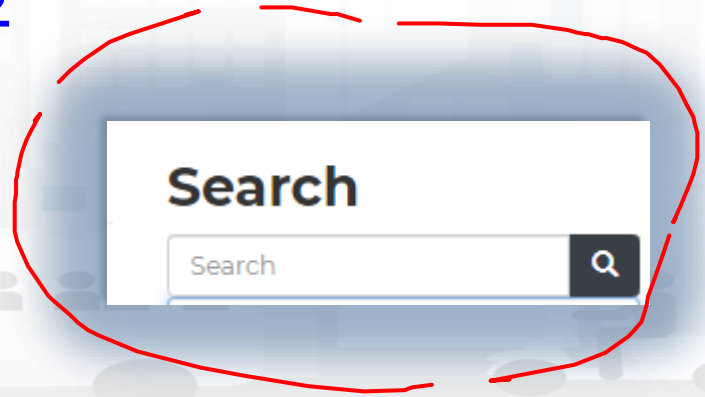
https://www.snap4city.org/download/video/DPL_SNAP4INDUSTRY.pdf

- Artificial Intelligence

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

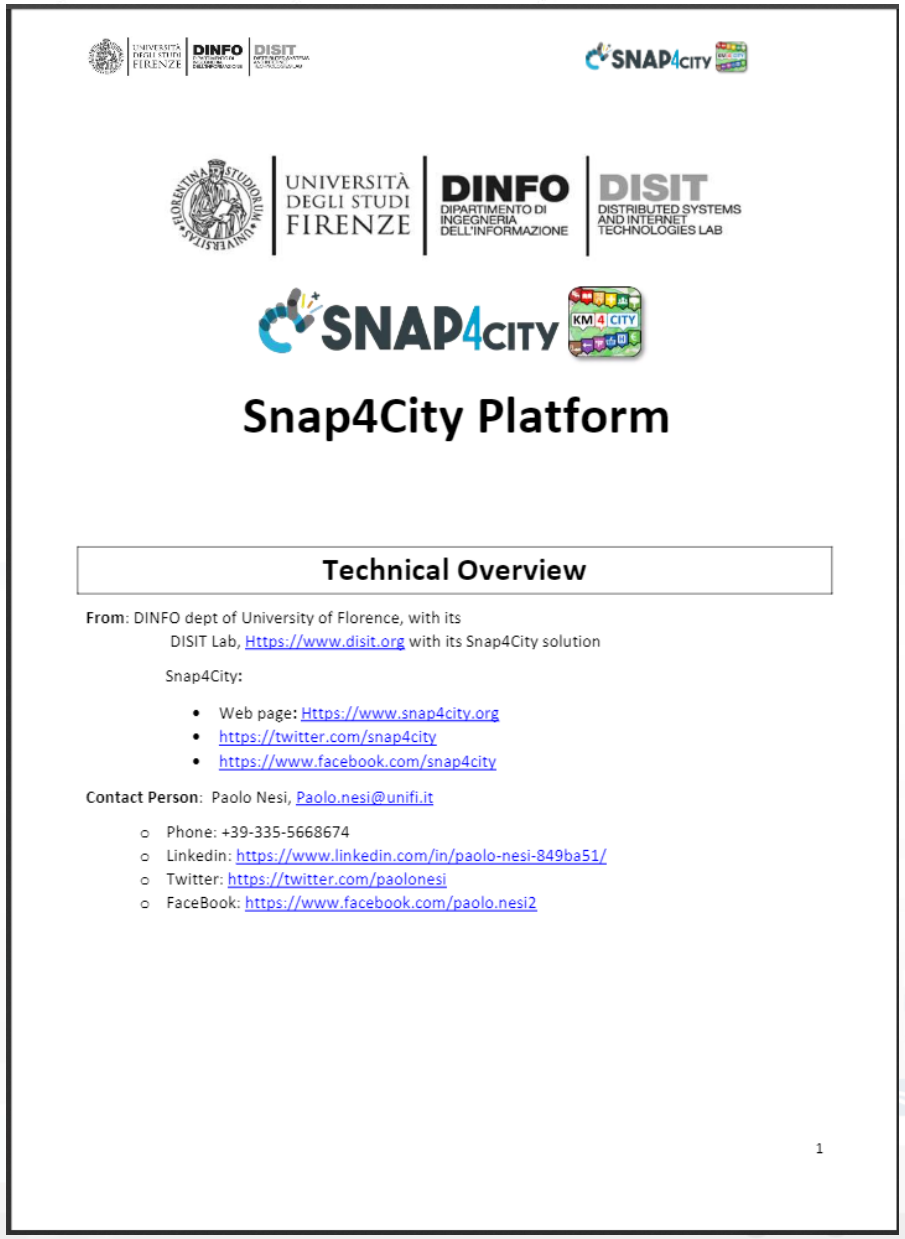
The Platform

- **Free Registration on Snap4City.org**
  - Please select DISIT ORG to be sure to access at the examples
  - Most of the cities / tenant are private and they do not left much visible
- **What you get** is probably the 10% of what is on the platform ☺
- **Training**: https://www.snap4city.org/577
- **Scenarious**: https://www.snap4city.org/4
- **Publications**: https://www.snap4city.org/426
- **WEB pages**: https://www.snap4city.org/78
- ***SEARCH on the right side***



Tech. Overview

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



Snap4City Platform

Technical Overview

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

Snap4City:

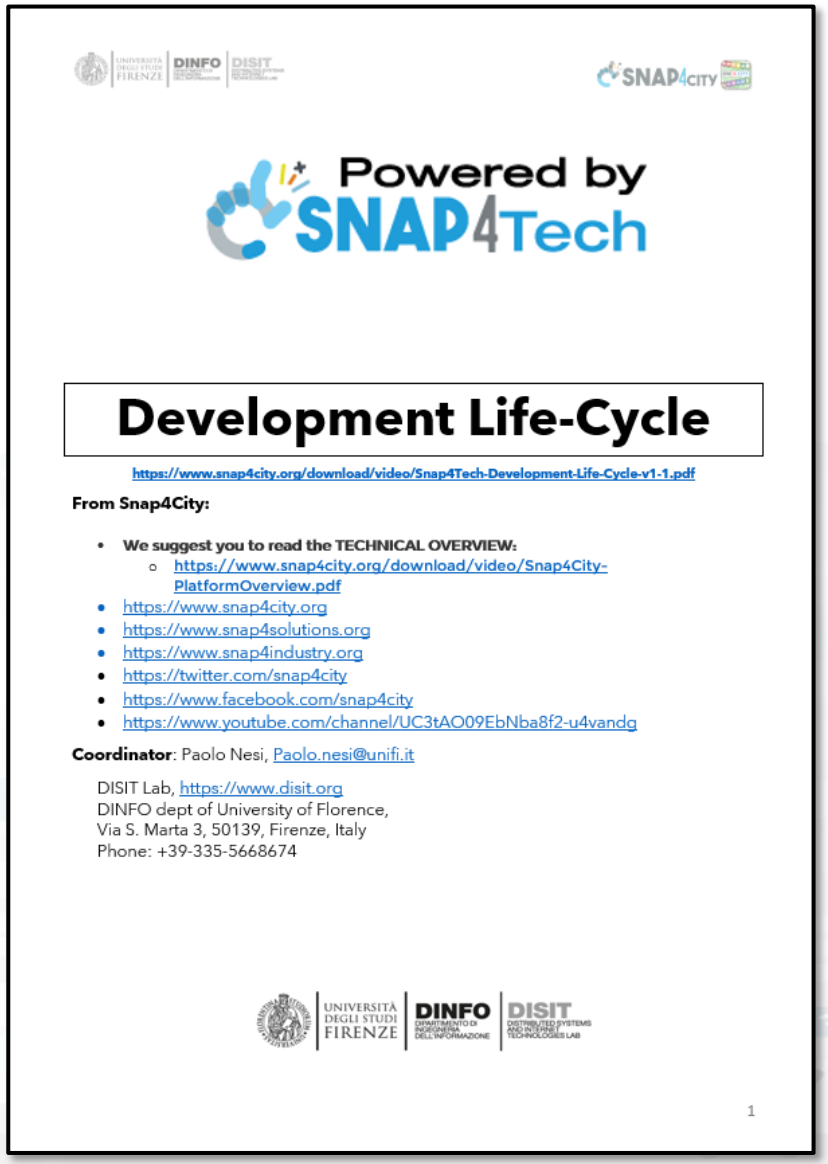
- Web page: Https://www.snap4city.org
- https://twitter.com/snap4city
- https://www.facebook.com/snap4city

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

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

1

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

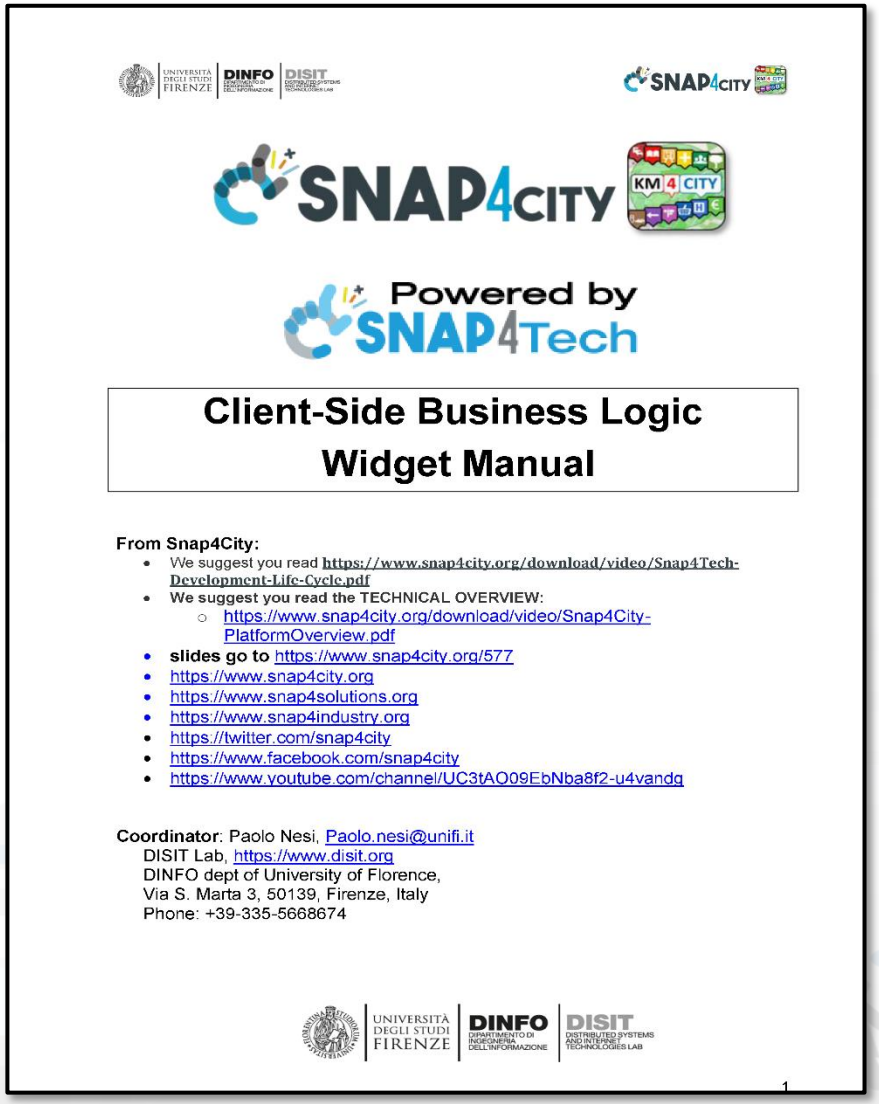
UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

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

CSBL

Client Side Business Logic



Powered by SNAP4Tech

Client-Side Business Logic Widget Manual

From Snap4City:

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

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

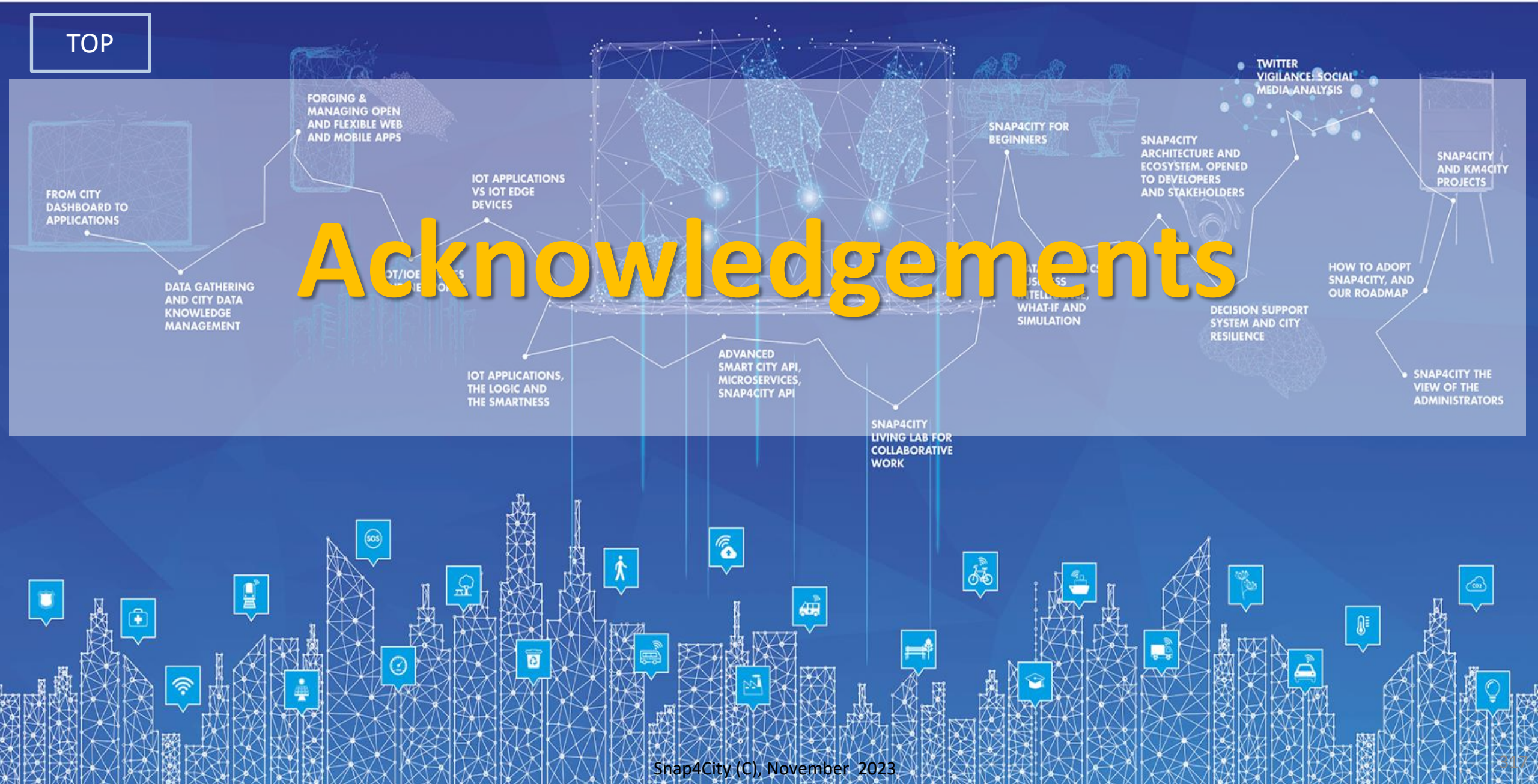
1

https://www.snap4city.org/download/video/ClientSideBusinessLogic-W[QR]df

Commercial Overview

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

TOP



Acknowledgements

DISIT lab roadmap vs model and tools' usage

2013
Km4City Ontology 1.1

- Tuscany, Road Graph
- Mobility
- culture, tourism
- Events
- Parking
- Services
- Linked open graph

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

- Twitter Vigilance
- Social Media Analytics, Sentiment Analysis

Km4City 1.4

2015
Km4City 1.5

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

ckan, resolute

H2020 (2015-18)

2016
(2016-21) H2020 REPLICATE

Km4City 1.6.4

- Smart Energy
- Sustainable Mobility
- Control Room
- Dashboard

Km4City 1.6.2

Sii-Mobility (2016-21) MIUR

SII-MOBILITY SCN

- Infomobility
- Mobile App
- Routing
- Multimodality

FIWARE

- 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
Sii-Mobility

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

E015 digital ecosystem (Powered by EXPO MILANO 2015)

(2016-19) MIUR

GHOST SIR

IOT/IOE

Km4City 1.6.6

Node-RED

SNAP4CITY SELECT for Cities H2020 (2017-19)

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

life weee (2017-20)

- Smart Waste

MOSAiC Mobility 4.0 for SmArt (i) City (POR CreO, Regione Toscana) (2018-20)

- Mobility Demand / Offer Analytics and Strategy

2018
Sii-Mobility

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

5G (2018-21)

5G tech
Energy
Industry 4.0
Synoptics

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

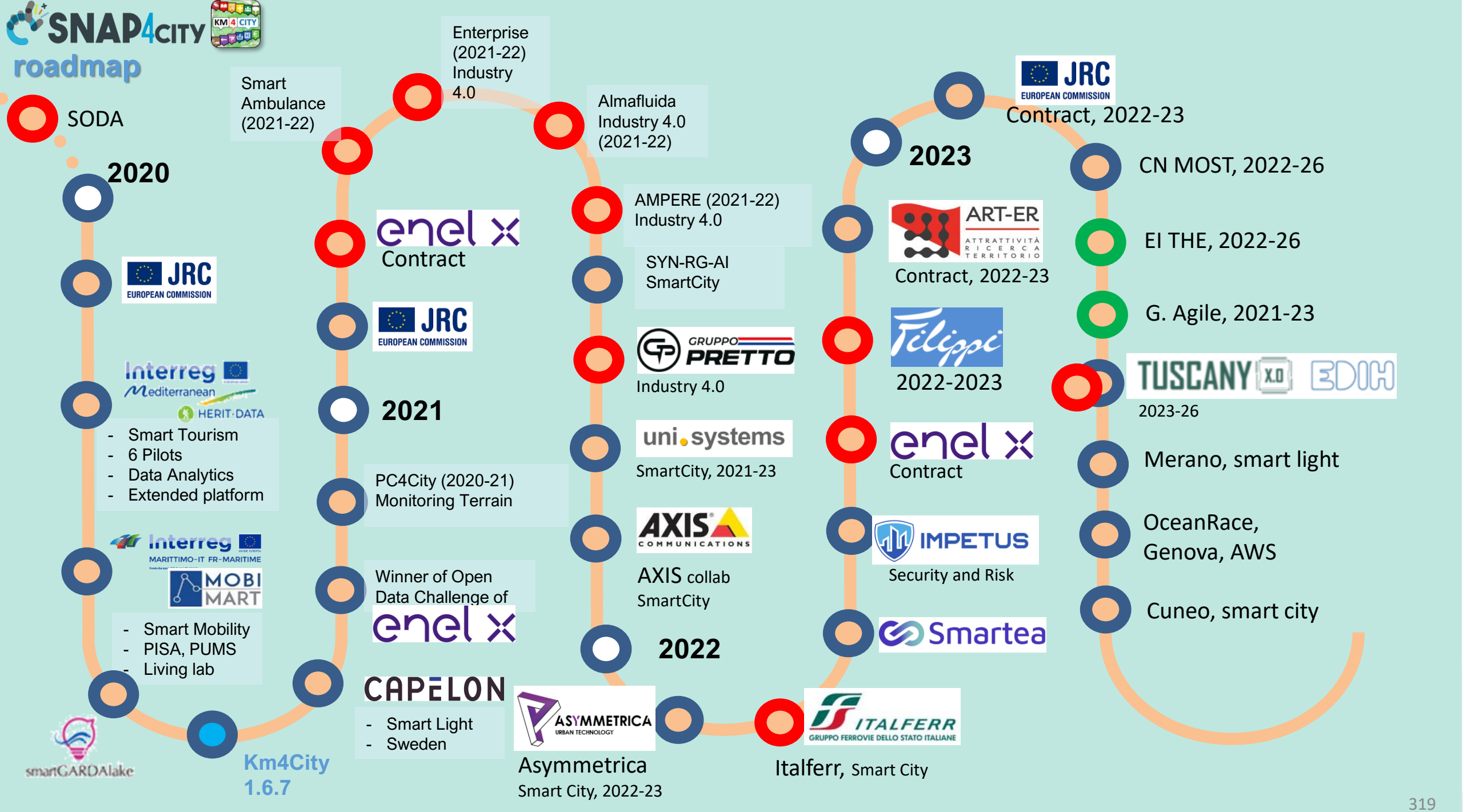
bee smart city

- Smart Lonato

Reverberi Enetec Gruppo MPES

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

Winner of Select4Cities PCP


SNAP4CITY
KM4CITY
roadmap
SODA
2020
JRC EUROPEAN COMMISSION
Interreg Mediterranean
HERIT-DATA
- Smart Tourism
- 6 Pilots
- Data Analytics
- Extended platform
Interreg MARITTIMO-IT FR-MARITIME
MOBI MART
- Smart Mobility
- PISA, PUMS
- Living lab
smartGARDAlake
Km4City 1.6.7
Smart Ambulance (2021-22)
Enterprise (2021-22) Industry 4.0
enel x
Contract
JRC EUROPEAN COMMISSION
2021
PC4City (2020-21) Monitoring Terrain
Winner of Open Data Challenge of
enel x
CAPELON
- Smart Light
- Sweden
Almafluida Industry 4.0 (2021-22)
AMPERE (2021-22) Industry 4.0
SYN-RG-AI SmartCity
GRUPPO PRETTO
Industry 4.0
uni.systems
SmartCity, 2021-23
AXIS COMMUNICATIONS
AXIS collab SmartCity
2022
ASYMMETRICA URBAN TECHNOLOGY
Asymmetrica
Smart City, 2022-23
ITALFERR GRUPPO FERROVIE DELLO STATO ITALIANE
Italferr, Smart City
2023
JRC EUROPEAN COMMISSION
Contract, 2022-23
ART-ER ATTRATTIVITÀ RICERCA TERRITORIO
Contract, 2022-23
Filippi
2022-2023
enel x
Contract
IMPETUS
Security and Risk
Smartea
CN MOST, 2022-26
EI THE, 2022-26
G. Agile, 2021-23
TUSCANY X.0 EDIH
2023-26
Merano, smart light
OceanRace, Genova, AWS
Cuneo, smart city

TOP

SNAP4CITY

KM 4 CITY

100% OPEN SOURCE

Be smart in a SNAP!

SMARTCITY EXPO WORLD CONGRESS

7-9 November 2023, Barcelona, Spain

Visit Snap4City in Hall 1

SNAP4 Appliances and Dockers **Installations**

CONTACT

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

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

www.snap4city.org

Email: snap4city@disit.org

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

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