

Be smart in a SNAP!



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

Developing Smart Applications & Business Intelligence Solutions

Sept. 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
DIPARTIMENTO DI
INGEGNERIA
DELL'INFORMAZIONE

DISIT
DISTRIBUTED SYSTEMS
AND INTERNET
TECHNOLOGIES LAB



SNAP4CITY



Powered by

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

<https://www.Snap4City.org>

Developing Smart Applications
& Business Intelligence Solutions



Sept. 2023, Course, Part 8

<https://www.snap4city.org/944>

<https://www.snap4city.org/577>

Paolo Nesi, paolo.nesi@unifi.it

<https://www.Km4City.org>

<https://www.disit.org>



Work with US: Open 3 Grants for PHD Course

- <https://www.snap4city.org/581>
- **2 Grants from late 2023-2027 on:**
 - Study of AI tools for the automatic generation of scenarios for the resolution of critical conditions in complex geolocated structures.
 - *Deadline: August 7, 2023, h 13:00 CET*
- **1 Grant from late 2023-2027 on:**
 - Study of artificial intelligence techniques integrated with visual analytics and business intelligence tools at support of Public Administrations decisions.
 - *Deadline: August 11, 2023, h 12:00 CET*

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

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 ...

ANY: DATA, BROKER, NETWORK AND VERTICAL

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

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

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

Native and External
Applications

Smart Parking

Smart Light

Smart Waste

Smart Energy

Social Media Analysis

...



METHODOLOGIES
LIVING LABS
COURSES AND COMMUNITY
DEVELOPMENT TOOLS



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

<https://www.snap4city.org/577>



On Line Training Material (free of charge)

<https://www.snap4city.org/944>

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

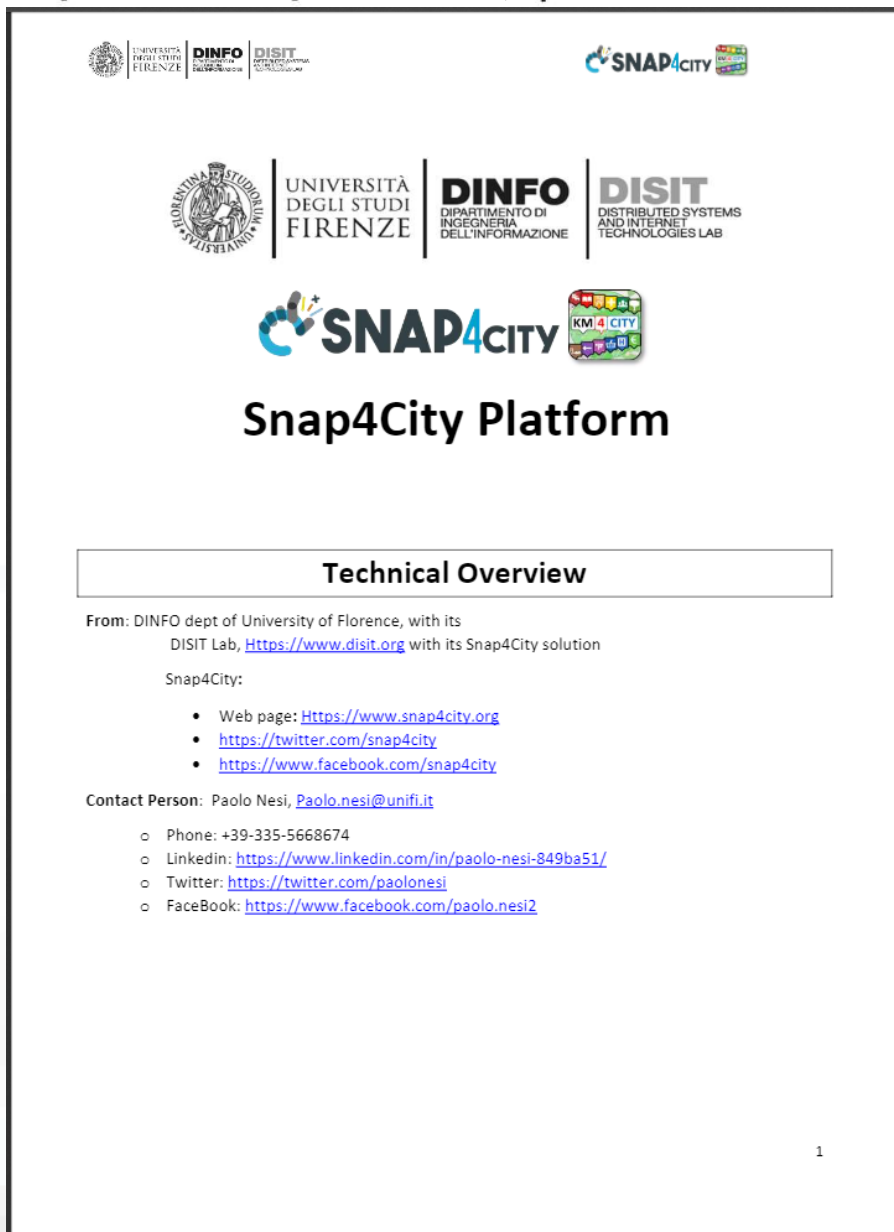
Video1								
Video2								
Video3								
Video4				none		none	none	none

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>

Tech Overview

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



The slide contains the following content:

- Logos for University of Florence, DINFO, DISIT, and SNAP4CITY at the top.
- A central logo for SNAP4CITY with the text "Snap4City Platform" below it.
- A box titled "Technical Overview" containing:
 - 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
 - o Phone: +39-335-5668674
 - o LinkedIn: <https://www.linkedin.com/in/paolo-nesi-849ba51/>
 - o Twitter: <https://twitter.com/paolonesi>
 - o FaceBook: <https://www.facebook.com/paolo.nesi2>
- A small number "1" at the bottom right corner.

Development

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



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

Client Side Business Logic

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

[Wdf](https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual)



Client-Side Business Logic Widget Manual

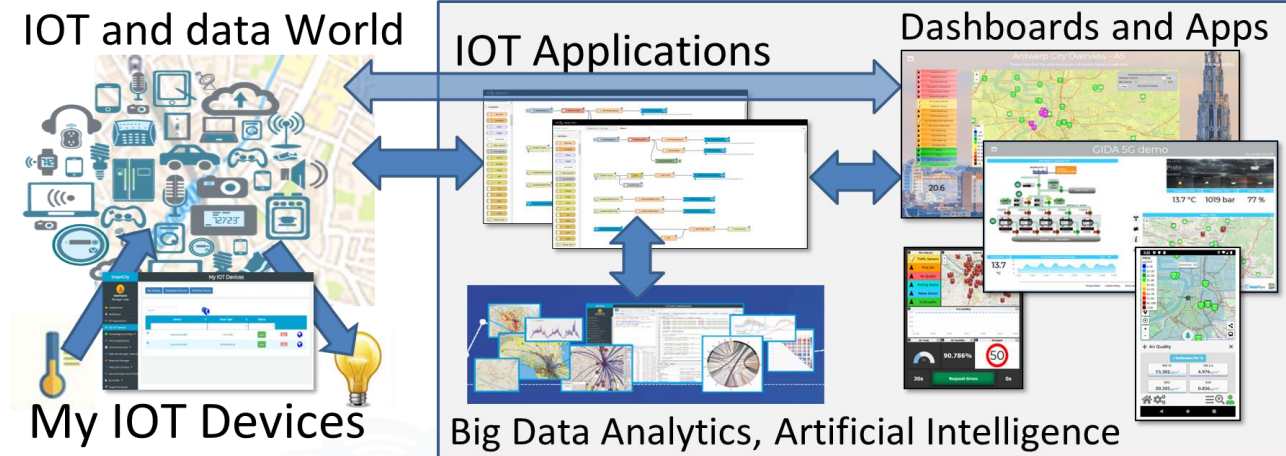
From Snap4City:

- 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/UC3TAO09EbNba8f2-u4vandu>

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

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.

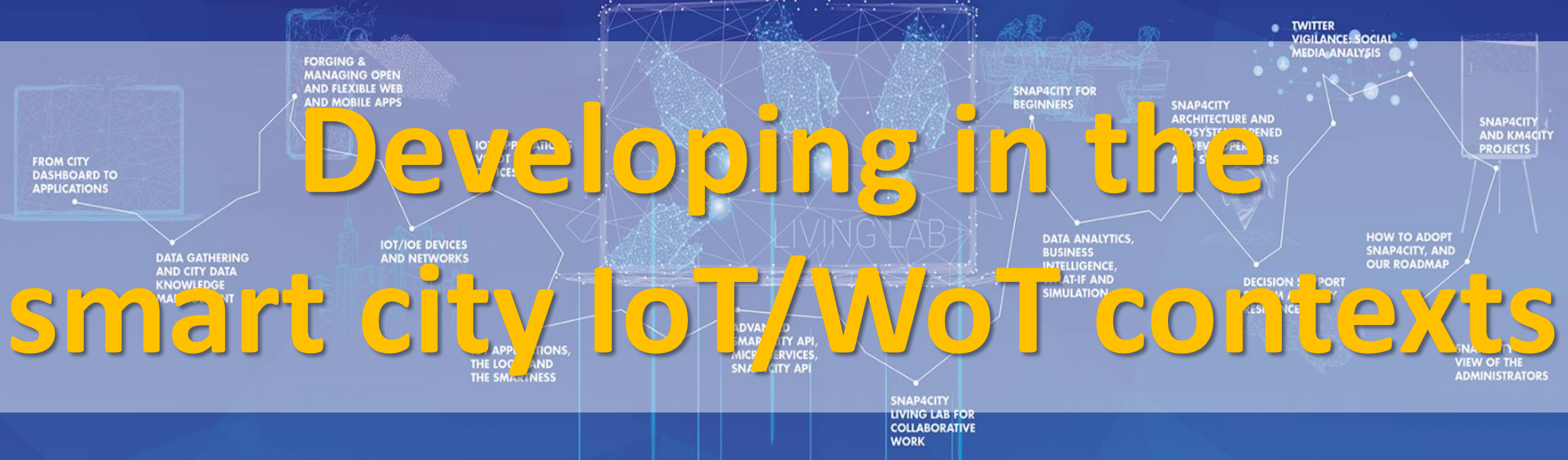
Agenda of Part 8

- **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

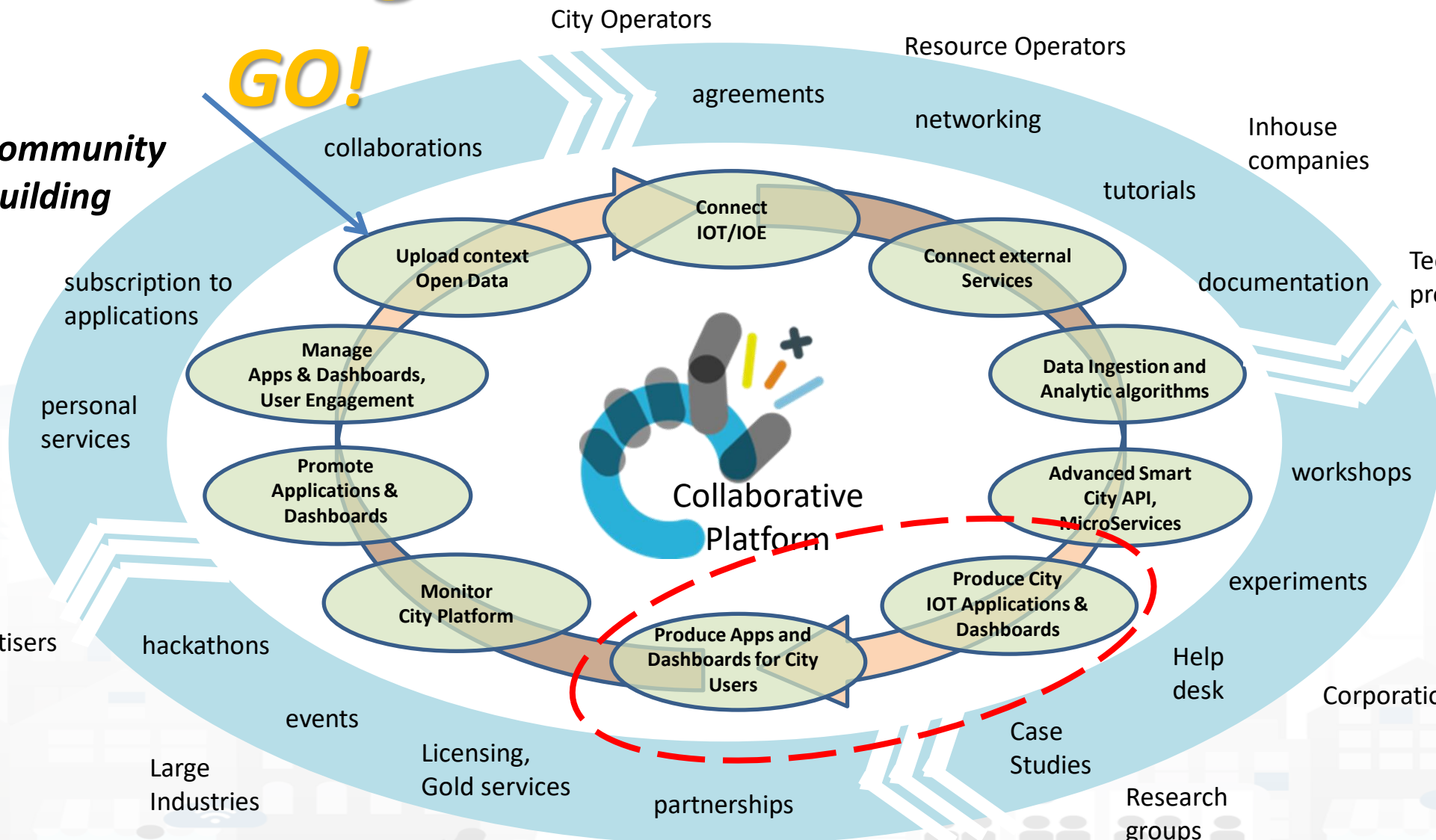


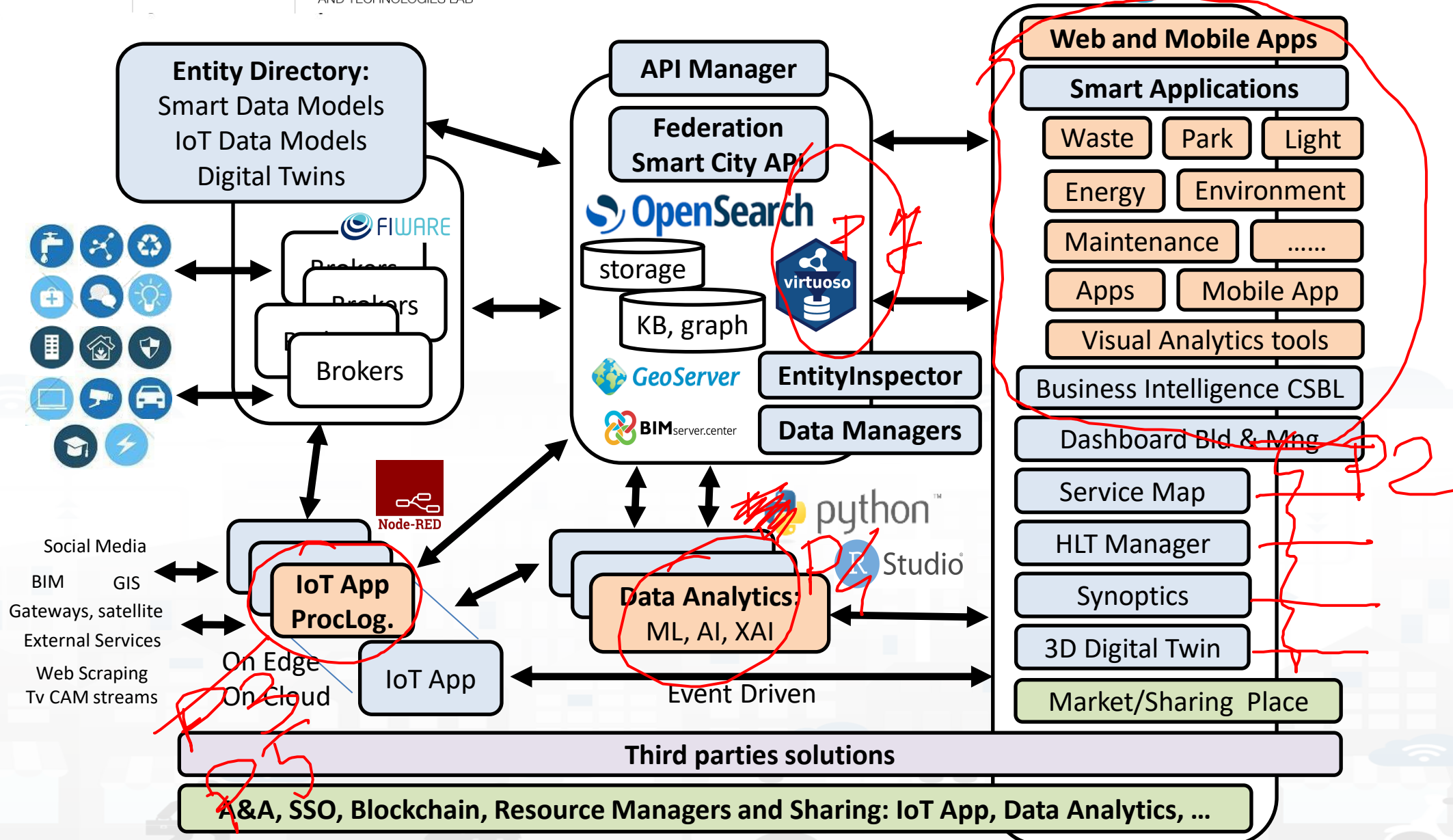
Accelerating



GO!

Community Building





Tech Arch

My IoT Sensors and Actuators

Entities/Devices Management

ID	IoT Device	Entity Type	Model	Ownership	Category	Status	API	Entity Location
ARQualityObservedMM428001	Antwerp	ARQualityObserved	custom	PRIVATE	Antwerp	visibility:antwerp	active	LOG
ARQualityObservedMM428002	Antwerp	ARQualityObserved	custom	PRIVATE	Antwerp	visibility:antwerp	active	LOG
ARQualityObservedMM428003	Antwerp	ARQualityObserved	custom	PRIVATE	Antwerp	visibility:antwerp	active	LOG
ARQualityObservedMM428004	Antwerp	ARQualityObserved	custom	PRIVATE	Antwerp	visibility:antwerp	active	LOG
ARQualityObservedMM428005	Antwerp	ARQualityObserved	custom	PRIVATE	Antwerp	visibility:antwerp	active	LOG

API Manager

Federation Smart City API

OpenSearch

storage

KB, graph

GeoServer

BIMserver.center

EntityInspector

Data Managers

virtuoso

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

Social Media

BIM GIS

Gateways, satellite

External Services

Web Scraping

Tv CAM streams

On Edge

On Cloud

IoT App ProcLog.

IoT App

Data Analytics: ML, AI, XAI

Event Driven

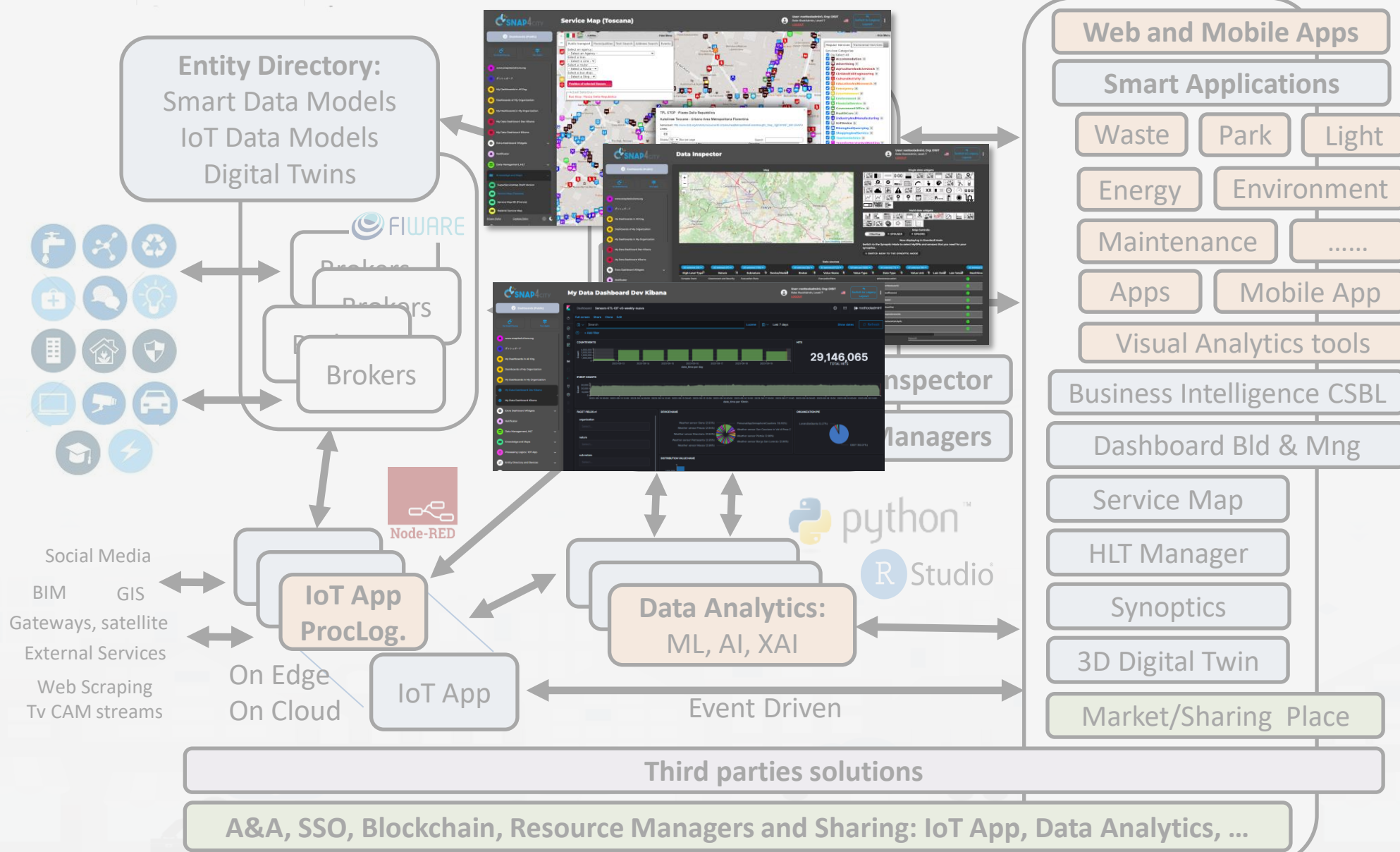
python

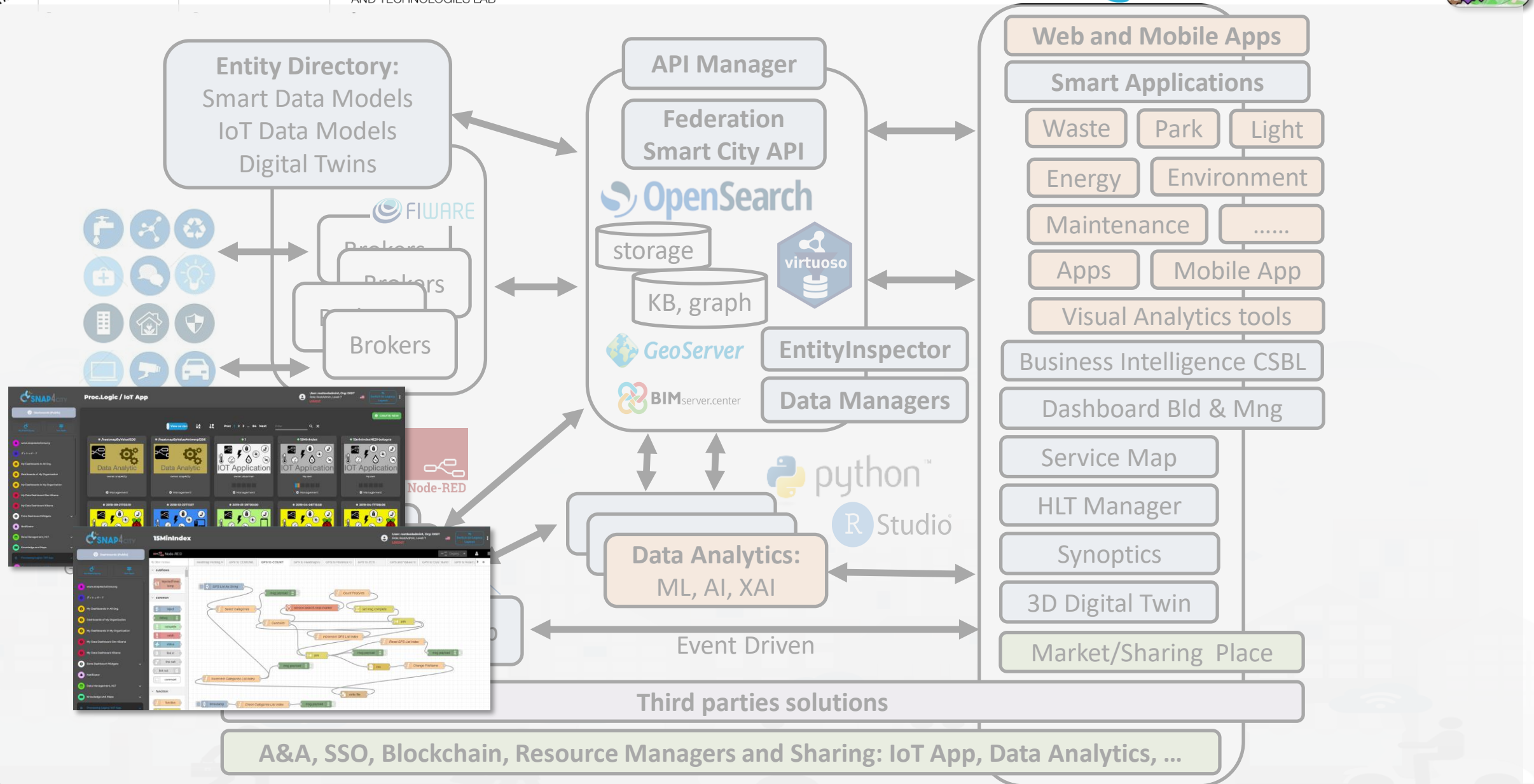
R Studio

Node-RED

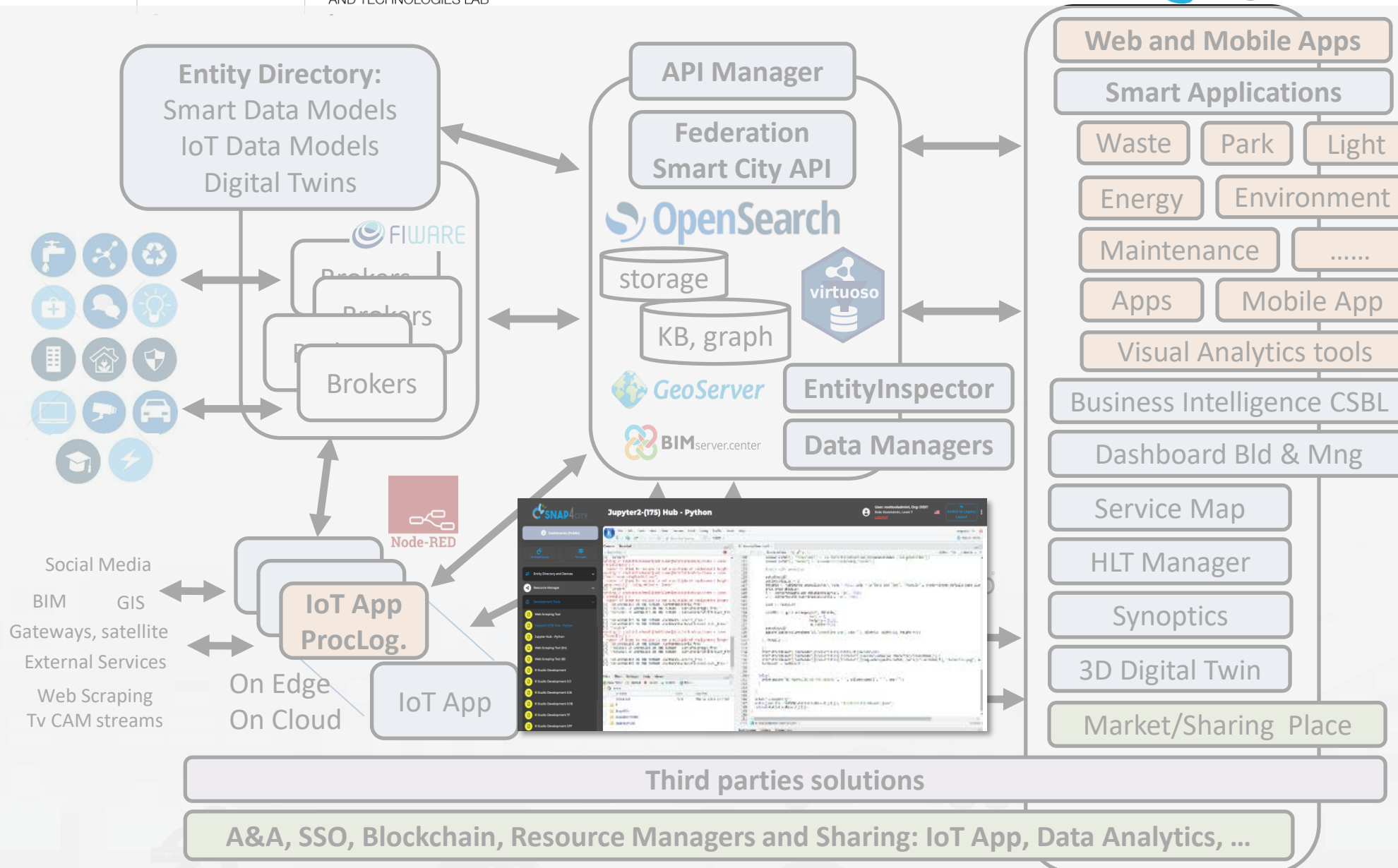
Third parties solutions

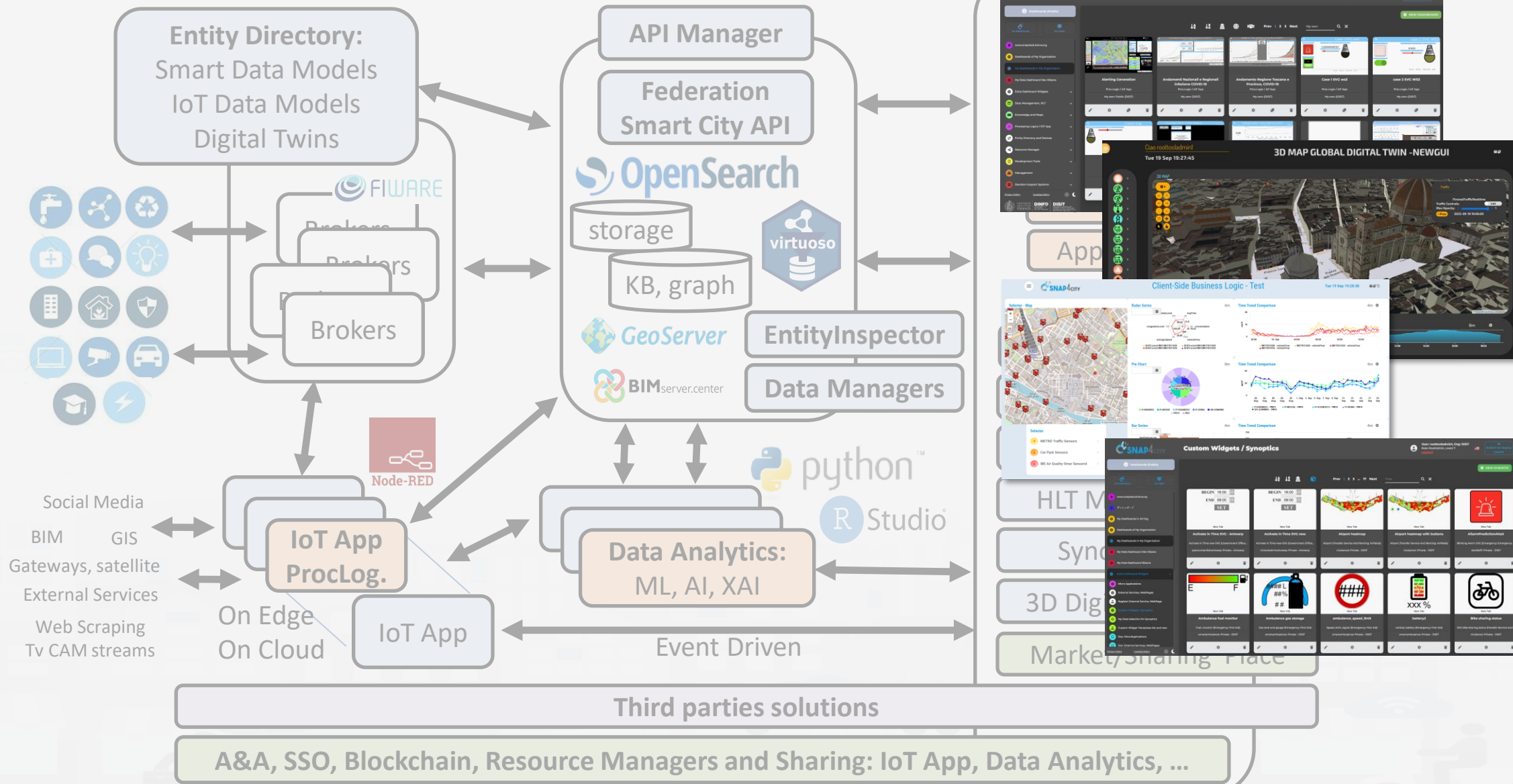
A&A, SSO, Blockchain, Resource Managers and Sharing: IoT App, Data Analytics, ...





Tech Arch





Tech Arch



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

Third parties solutions

- **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.

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

ASCAP



Operation:

What-if

Simula
tion

Event
Reporting

File
Mng

Heatmap
Mng

Traffic Flows
Mng

ODM
Mng

TVCam
Mng

BIM
Mng

....



KPI, Indicator
Analysis

WorkFlow
Tickets BPM

Sentiment
Analysis

SUMO

Tipyc. Time
Trend

....

Open to any module
and system

Data
Processing

Predic
tions

Anomaly
detection

Statis
tics

Artificial
Intelligence

Routi
ng



....

Internal Interoperability: API, MicroServices

Snap4City Node-red Libraries

ASCAP



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

Service Map

LOGraph



SuperServiceMap

DevDash



Linked Data triples

NGSI V1, V2, LD



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

Directory

Blockchain

Brokers

Brokers

Harvester

Data Manager



Any protocol and format

Any protocol and format

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

Any protocol and format

Any protocol and format



Device Layer



External Third Party Services

docker

SNAP4CITY

Platform Management:

Proc.Logic / IoT Applications
Data Flow Logic

KEYCLOAK

OpenLDAP

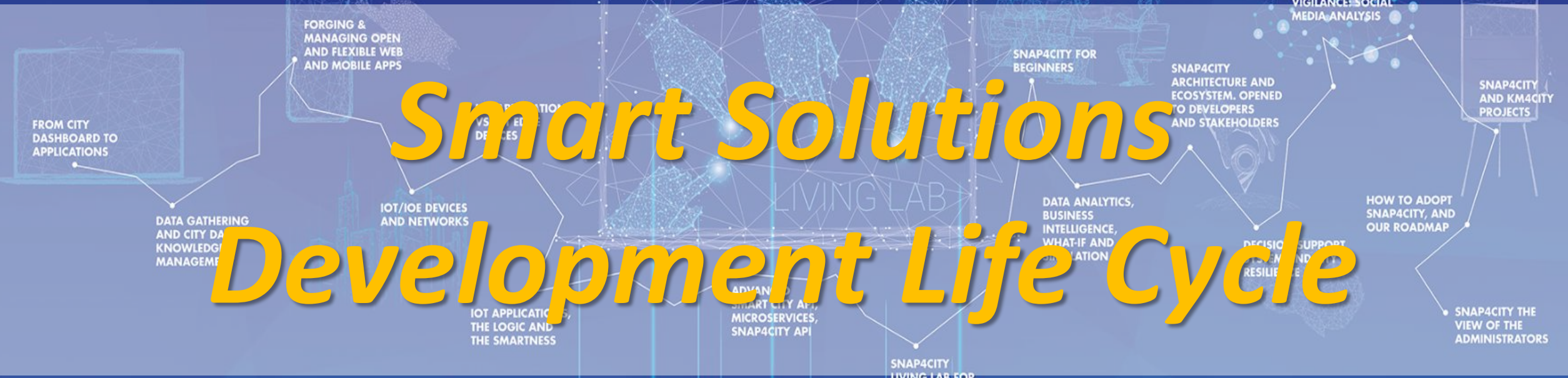
SNAP4CITY

Authentication and Authorization:
GDPR compliant

Partecipatori and Living Lab Supports

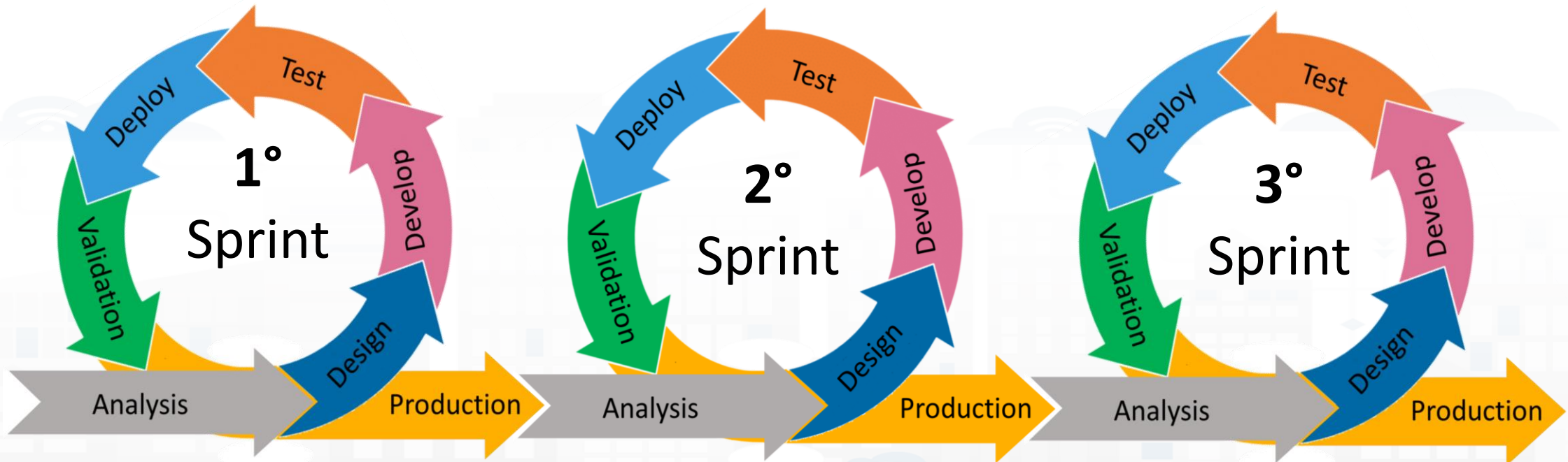
TOP

Smart Solutions Development Life Cycle

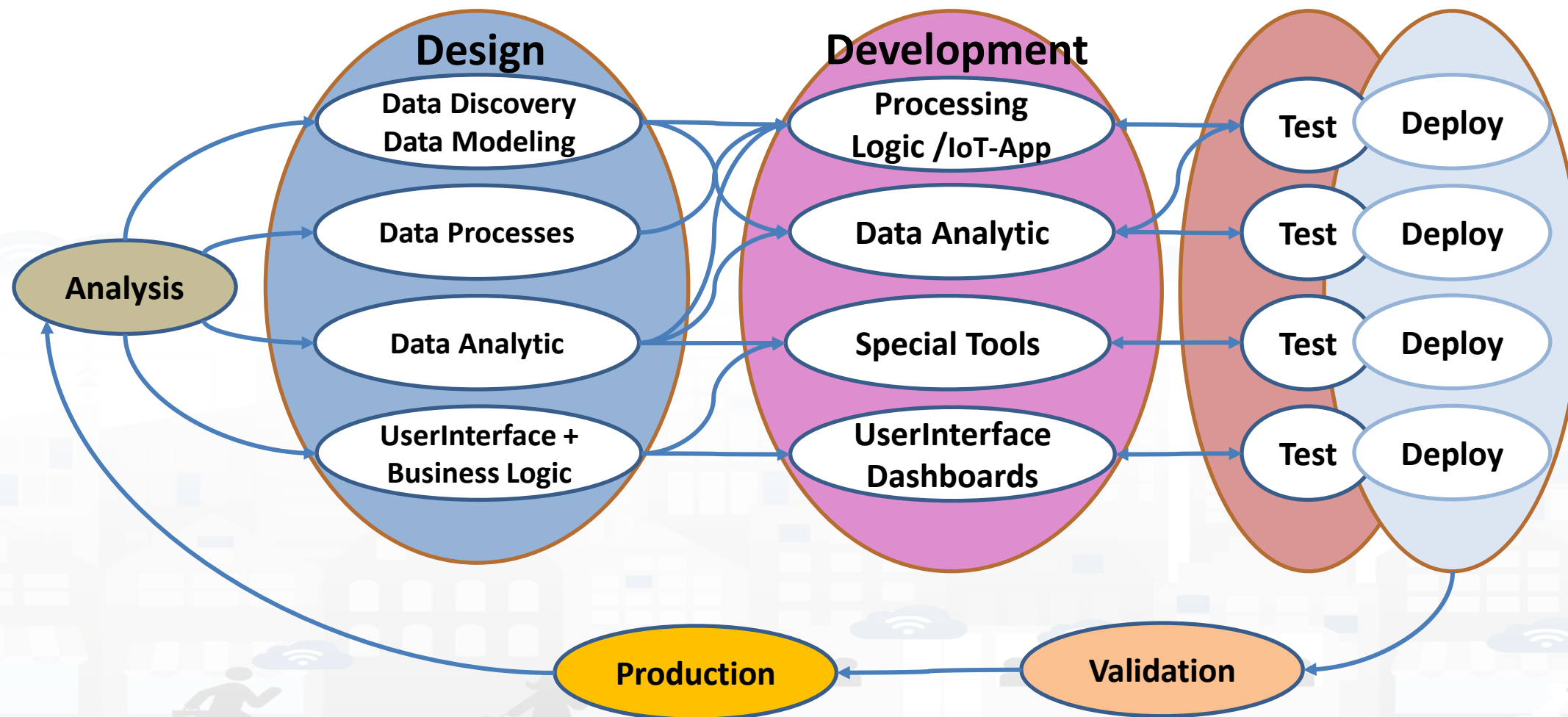




Development Life Cycle Smart Solutions Agile: CD-CI, Continuous Dev – Continuous Improvement



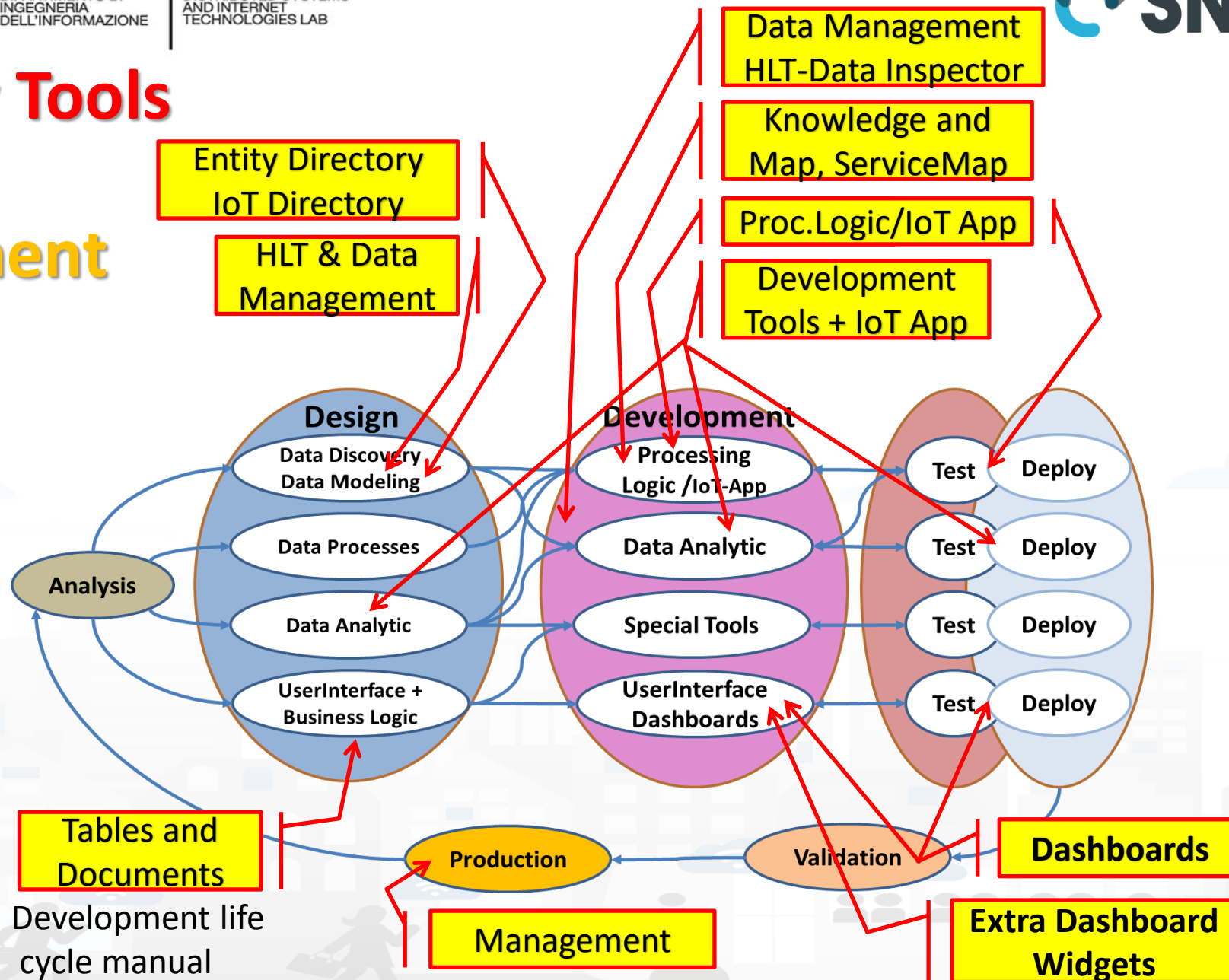
Development Life Cycle Smart Solutions



Snap4City Tools

vs

Development Life Cycle



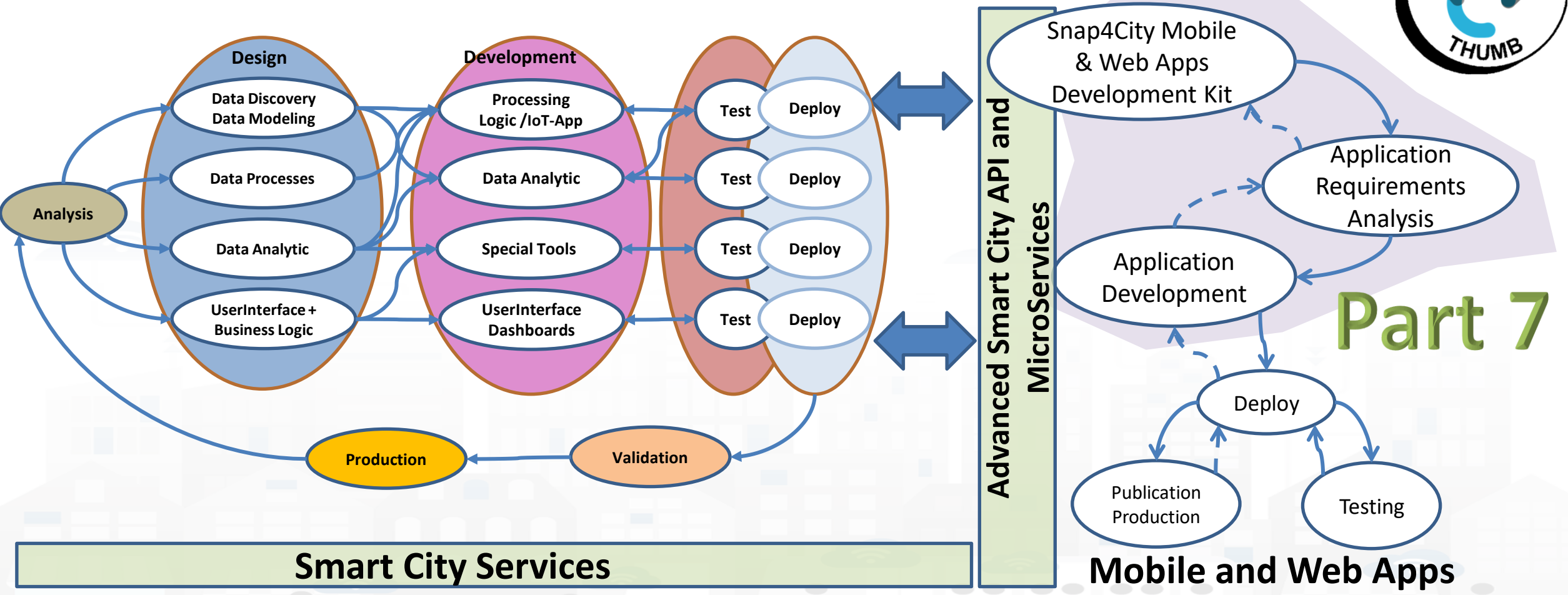
Test & Deploy

- 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



Development

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



Development Life-Cycle

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

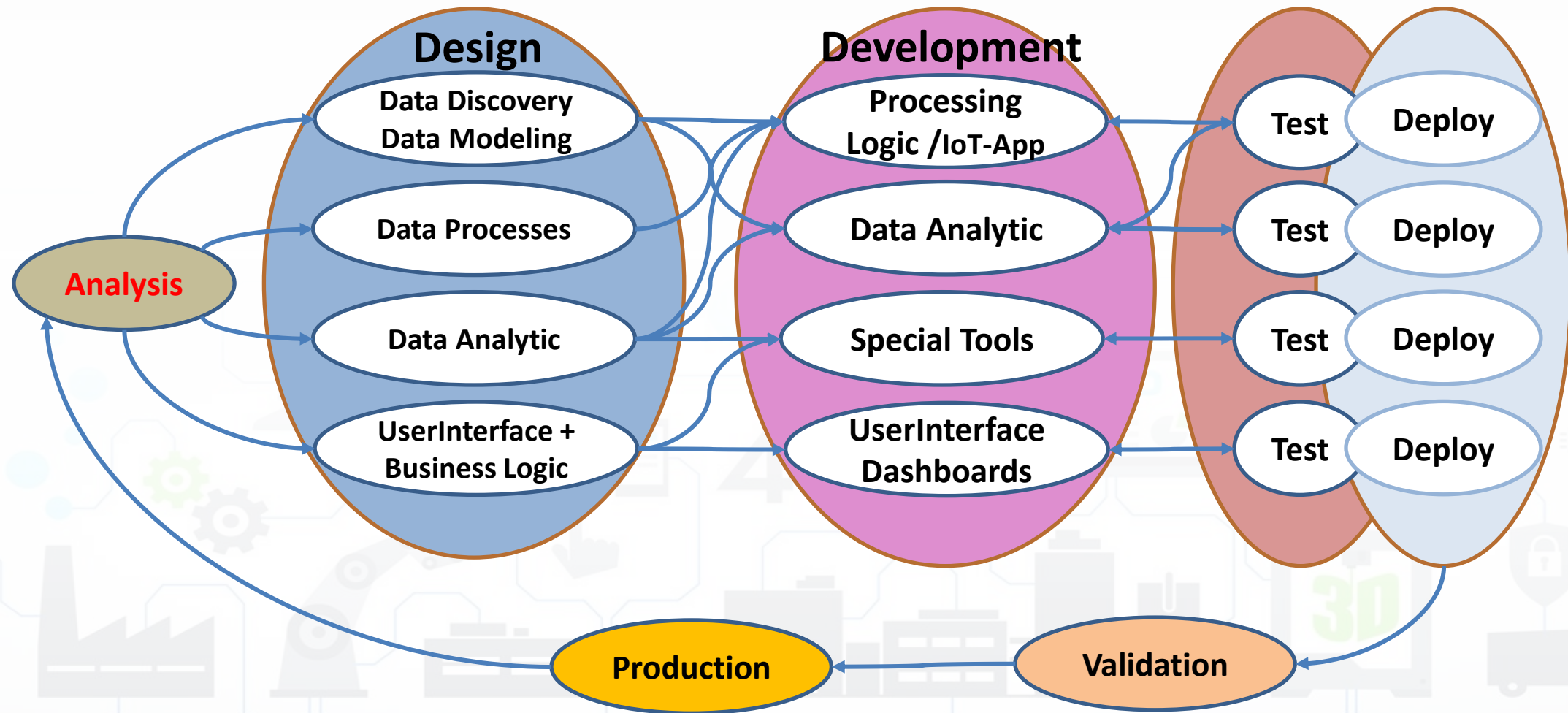
From Snap4City:

- 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-u4vandq>

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

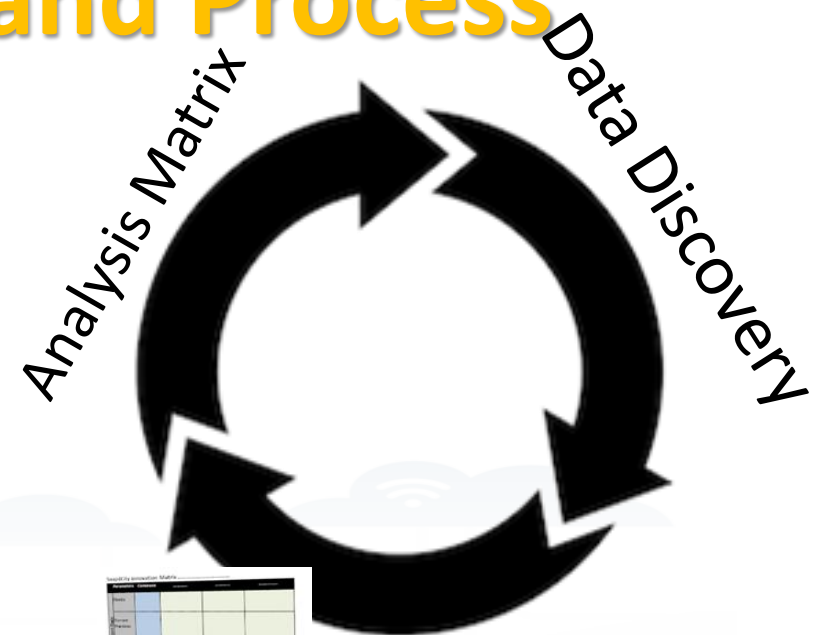
Development Life Cycle Smart Solutions



- **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



Snap4City Innovation Matrix

	Parameters	Commons	Operators	360°	Visitors
Current State	Needs	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]
	Current Practices	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]
	Value proposition (Current)	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]
Future State	Value proposition (Future)	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]
	Solution	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]
	Value Capture	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]
	Key Partners	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]
	Barriers	[Sticky notes]	[Sticky notes]	[Sticky notes]	[Sticky notes]

Snap4City Innovation Matrix

	Parameters	Commons
Needs		
Current Practices		
Value proposition (current)		
Value proposition (Future)		
Solution		
Value Capture		
Key Partners		



Design Scenarios

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 Console	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 Driver** 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

Data Model of the Driver

- Name: string
- Surname: string
- Age: number
- Weight: number
- Phone: string
- Email: string
- DriverAnalysisID: ServiceURI
-

Register to instantiate

Driver: user45

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

Write SURIs 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

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

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
-

legenda

- Entity Instance
- Entity Model
- Entity Messages with dateObserved

Register to instantiate

Driver: user45

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

Handwritten notes: "Date observed" and "time" with arrows pointing to the DriverAnalysis field and a separate diagram.

DriverAnalysis: user45driveranalysis

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

Data Model of the Driver

- Name: string
- Surname: string
- Age: number
- Weight: number
- Phone: string
- Email: string
- DriverAnalysisID: <http://.../user45driveranalysis>
- ServiceURI

Write SURJ to create cross references

DriverAnalysis: user45driveranalysis

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

New update on user45driveranalysis by sending a message

DriverAnalysis: user45driveranalysis

- DriverID: <http://.../user45>
- dateObserved: 22-03-2022T12:00:00
- Status: "good"
- Location: room45
- Doctor: [https://.....](https://...)
- Tools: null

New update on user45driveranalysis by sending a message

Register to instantiate

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

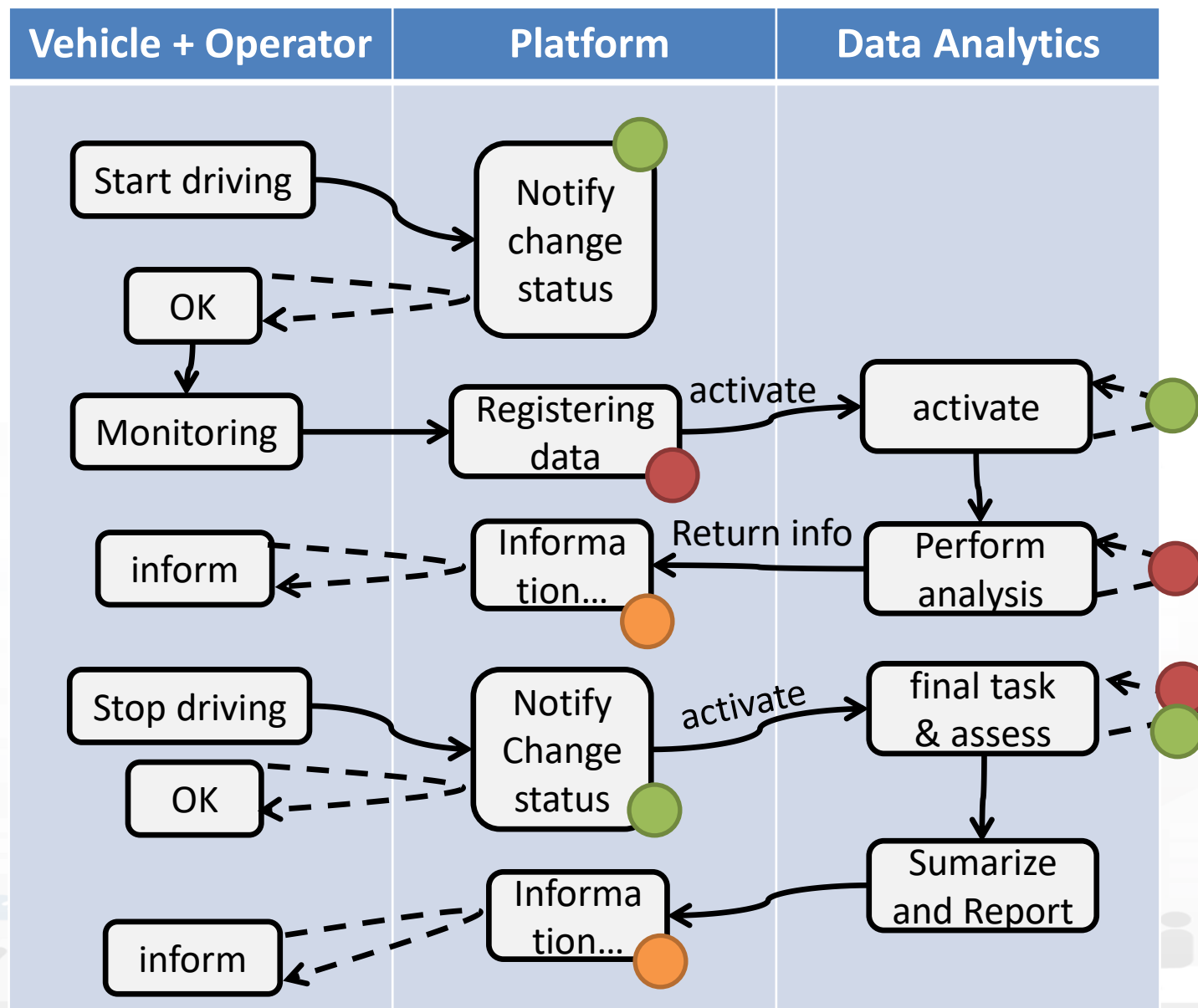
External Services				
URL Web pages	parameter	Description	Nature	Subnature

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

- **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

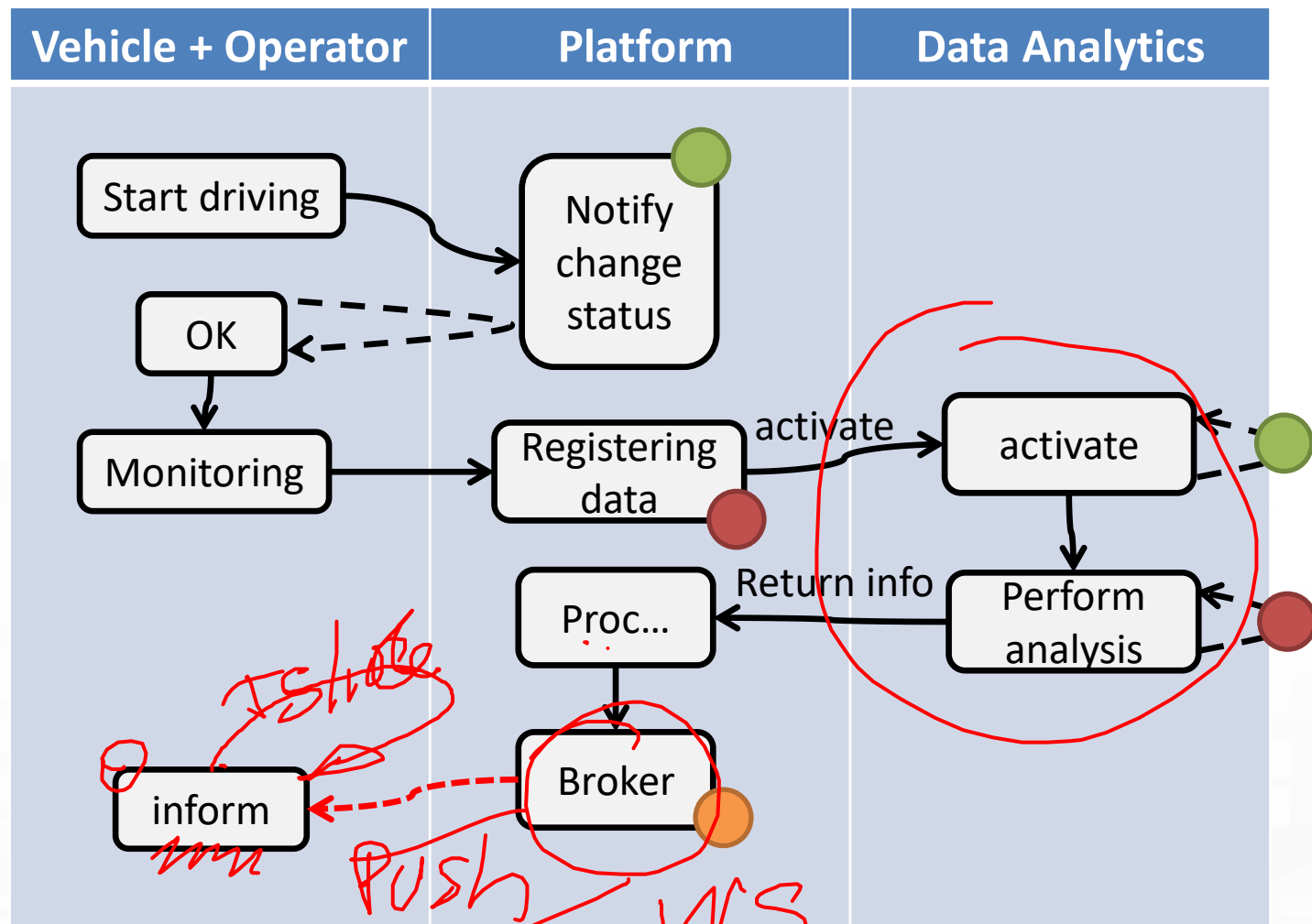
Example: Activity Diagram

- **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

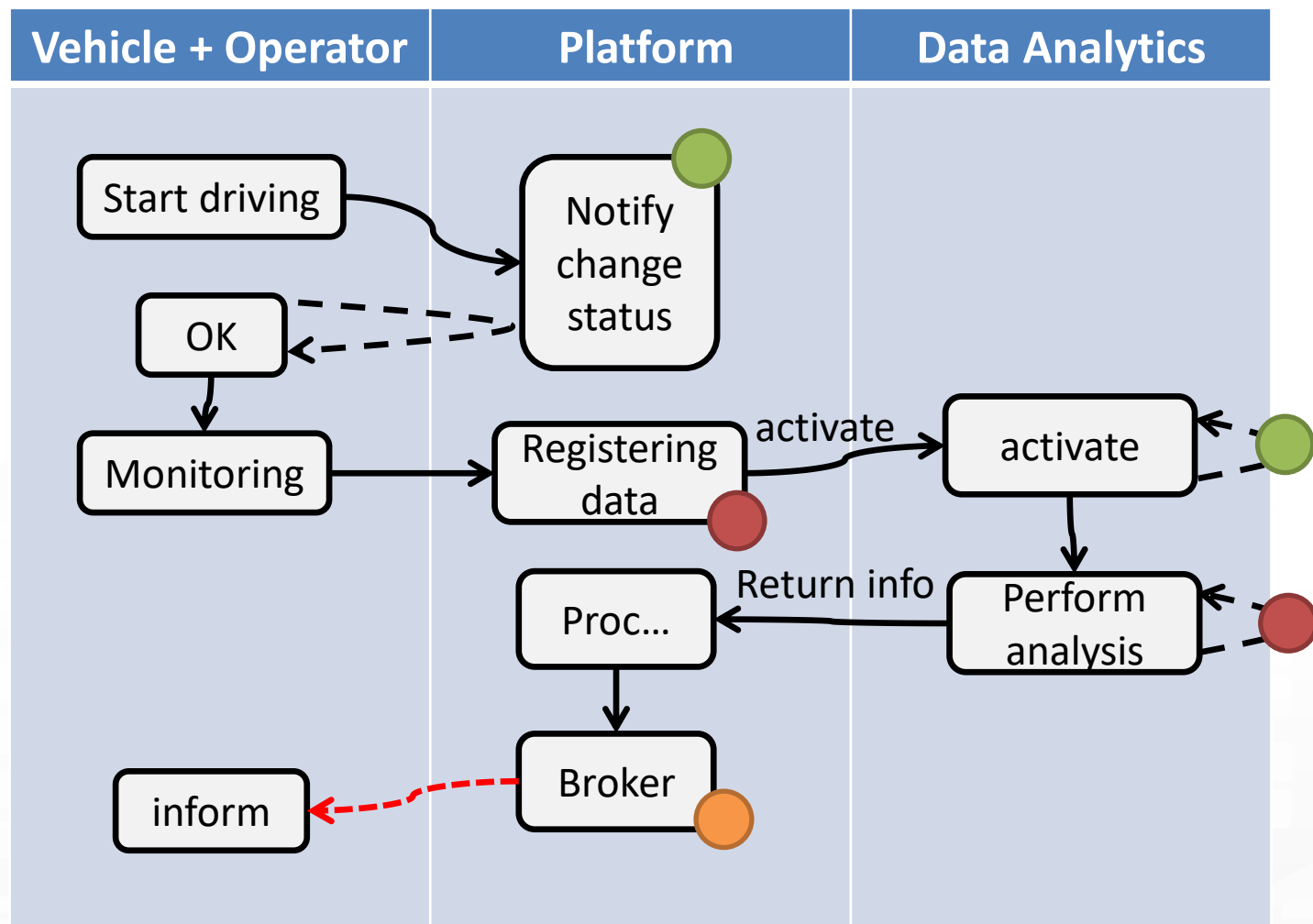


A variant

- 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)**



- 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=myspersonaldatatester-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":"myspersonaldatatester-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>



## 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:***

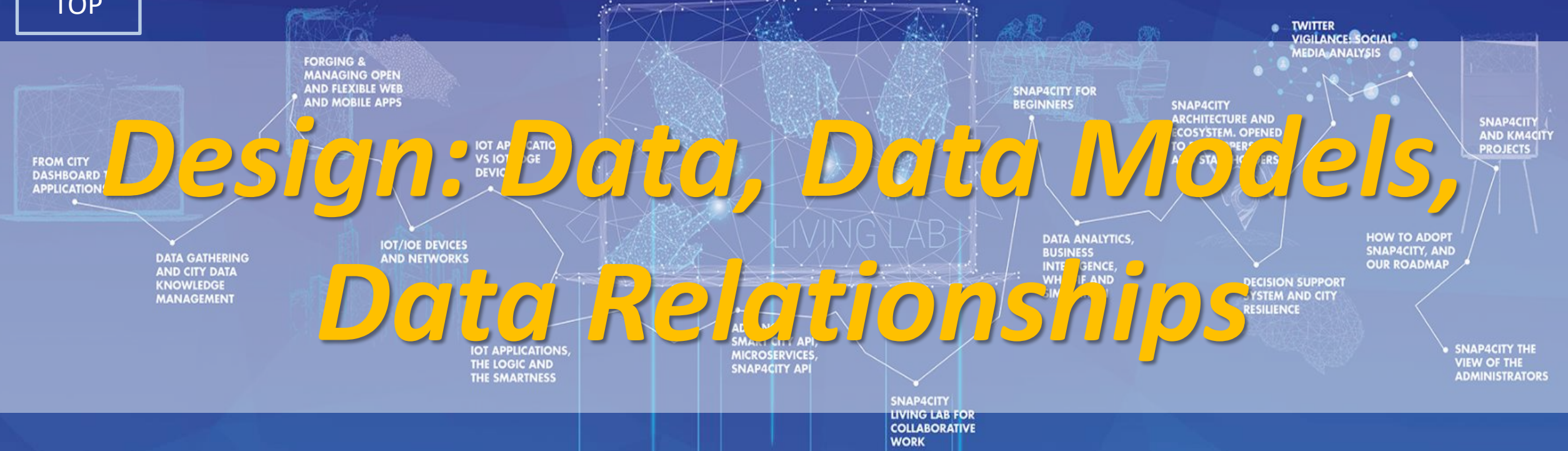
*Do not worry if at the first sprint of the above steps you forgotten to fill some details. It is quite sure that, you have also provided some details that would have to be revised/changed at the next iteration.*

*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.*

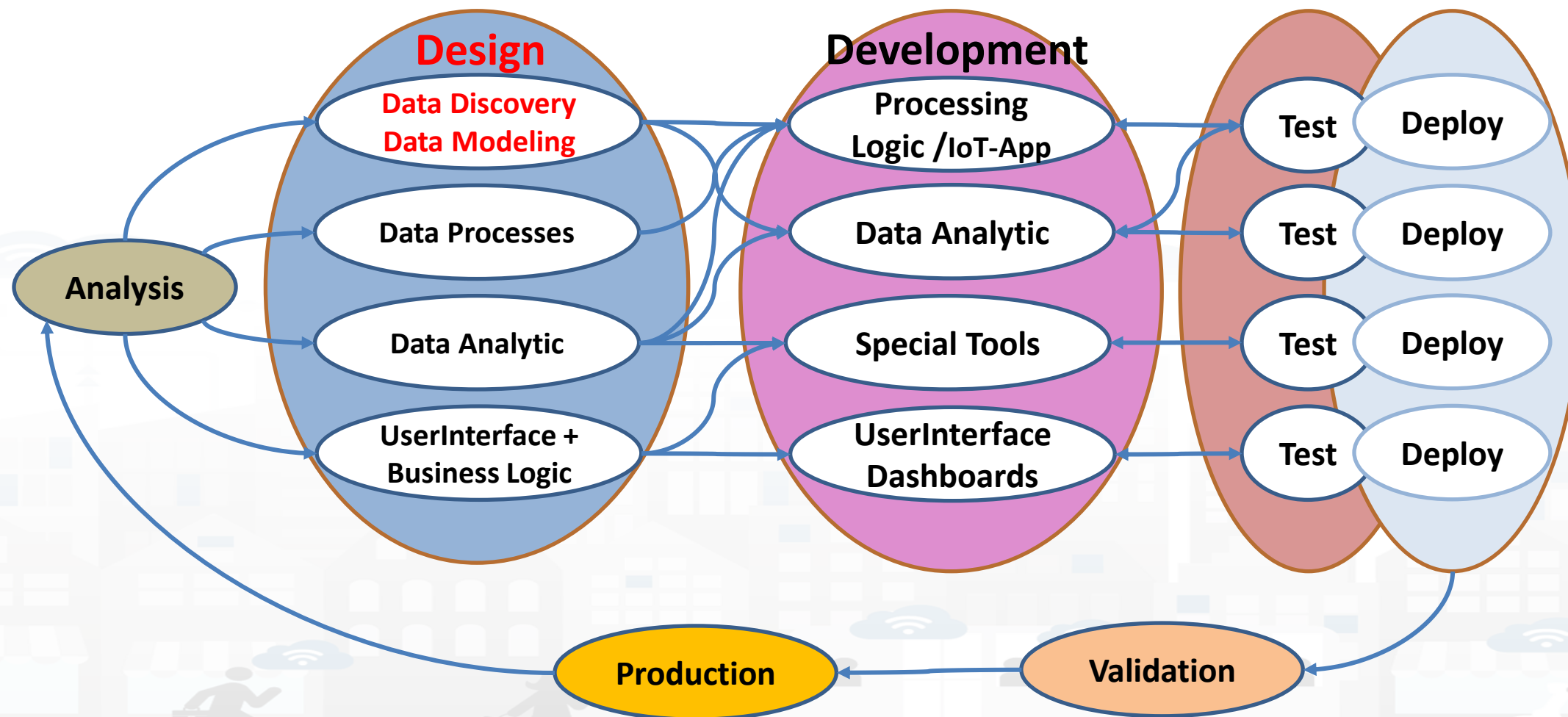
*They would leverage the smart solutions to a new level, at each sprint.*

TOP

# Design: Data, Data Models, Data Relationships



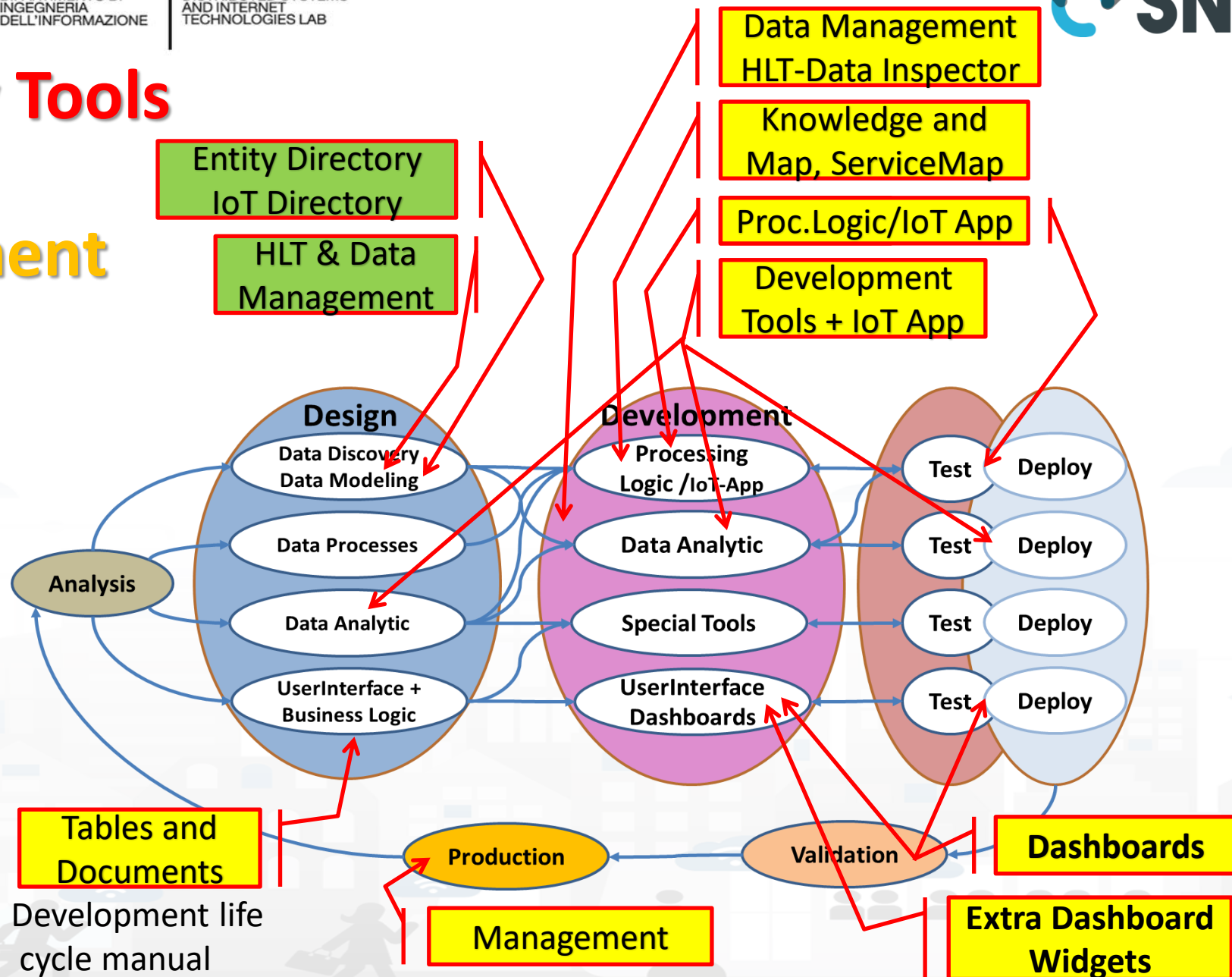
# Development Life Cycle Smart Solutions



# Snap4City Tools

vs

# Development Life Cycle



# *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        | Referen<br>t    | Provid<br>er          | endpoi<br>nt | Authent<br>ication | HL<br>protocol | protoc<br>ol | HLT                 | Format  | Size               | Volum<br>e    | Rate                                   | GPS<br>ed      | foto | License /<br>Condition of<br>use |
|--------------------------|-----------|-------------|--------|---------------|---------------|-----------------|-----------------------|--------------|--------------------|----------------|--------------|---------------------|---------|--------------------|---------------|----------------------------------------|----------------|------|----------------------------------|
| Graph road               | Energy    | Static      | In     | Struct        | Understood    | Name<br>Surname | Stakehol<br>der ID    | url          | Simple             | Push           | Datex        | Sensor              | XML     | 2<br>variable<br>s | 10Byte        | Every 10<br>minutes                    | Yes            | URL, | Public as CC...                  |
| Parking                  | Graph     | Real Time   | Out    | Non<br>struct | Acquired      | Email           | Staff or<br>not staff | Broker       | Certificate        | Pull           | WS           | Sens-<br>Actuator   | JSON    | 15<br>fields       | 1245<br>Kbyte | Sporadic,<br>max 1000<br>times per day | No             | IMG  | Link to file                     |
| Consumption<br>of energy | Mobility  | Combined    | In/out |               | Scheduled     | Phone           | Internal..            |              | Etc.               |                | REST         | KPI                 | GeoJSON |                    |               | Periodic                               | Kind           |      | Private ...                      |
|                          | Transport | RT stream   |        |               | Tested        | Etc..           |                       |              |                    |                | Custom       | Personal<br>Data    | KMZ     |                    |               | 2 per day                              | Insid<br>e msg |      | Restricted to ...                |
|                          | ....      | RT Messages |        |               | Operative     |                 |                       |              |                    |                | ODBC         | Ext Srv             | WFS     |                    |               |                                        | Static<br>...  |      | Sensible data                    |
|                          |           |             |        |               | Failed        |                 |                       |              |                    |                | JDBC         | IOT                 | WMS     |                    |               |                                        |                |      | GDPR aspects                     |
|                          |           |             |        |               | Not<br>needed |                 |                       |              |                    |                |              | Virtual<br>Sensor   | GTFS    |                    |               |                                        |                |      |                                  |
|                          |           |             |        |               |               |                 |                       |              |                    |                |              | GIS                 | db      |                    |               |                                        |                |      |                                  |
|                          |           |             |        |               |               |                 |                       |              |                    |                |              | Heatmap             |         |                    |               |                                        |                |      |                                  |
|                          |           |             |        |               |               |                 |                       |              |                    |                |              | Path,<br>trajectory |         |                    |               |                                        |                |      |                                  |
|                          |           |             |        |               |               |                 |                       |              |                    |                |              | Trend               |         |                    |               |                                        |                |      |                                  |
|                          |           |             |        |               |               |                 |                       |              |                    |                |              | ....                |         |                    |               |                                        |                |      |                                  |

**Examples are provided per colum.**

The resulted raws may have not sense.

The status refers to the ingestion process.



TOP

# Design: Data Modeling

Part 4

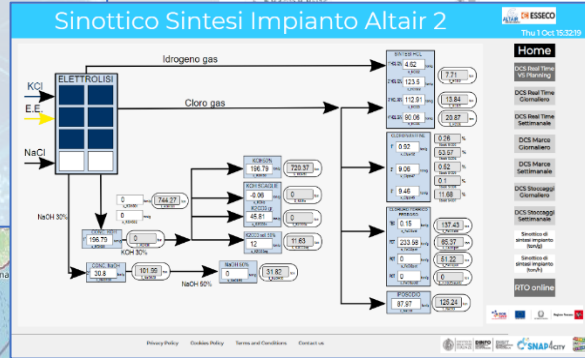
Part 5



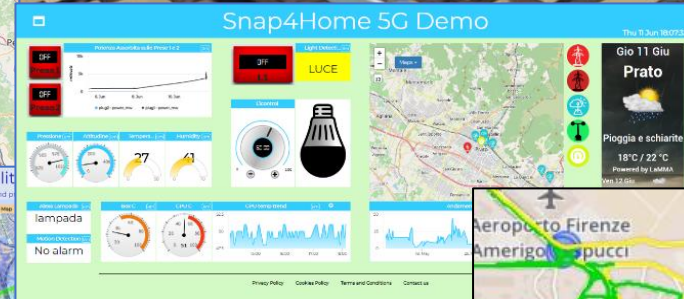
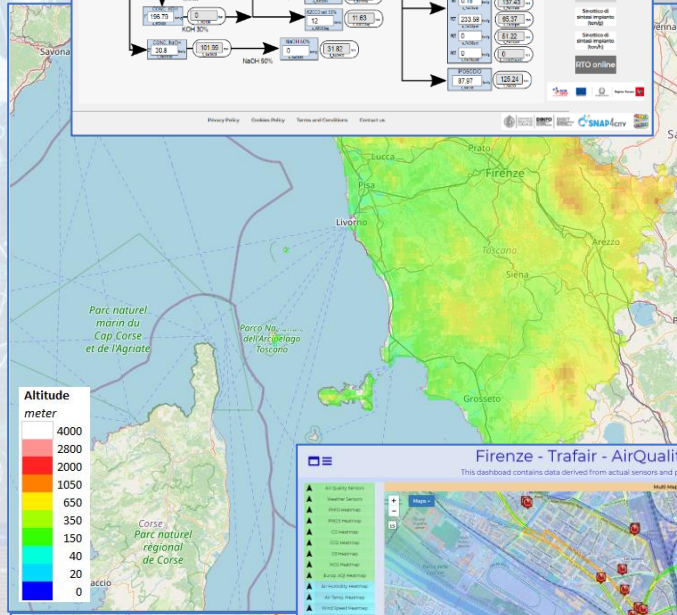
# High Level Types

Snap4City (C), September 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.



**SNAP4CITY**  
- Digital Twin Global - Fire  
demonstrator



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

**DINFO**  
DIPARTIMENTO DI  
INGEGNERIA  
DELL'INFORMAZIONE

**DISIT**  
DISTRIBUTED SYSTEMS  
AND INTERNET  
TECHNOLOGIES LAB

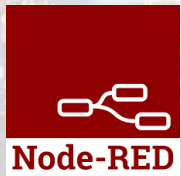
# Standards and Interoperability (6/2023)



## 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, XPD, OSM, Enfuser FMI, Lidar, gITF, 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>



## What About Entity Instances / IoT Devices, Time Series

### Entity / IOT Device



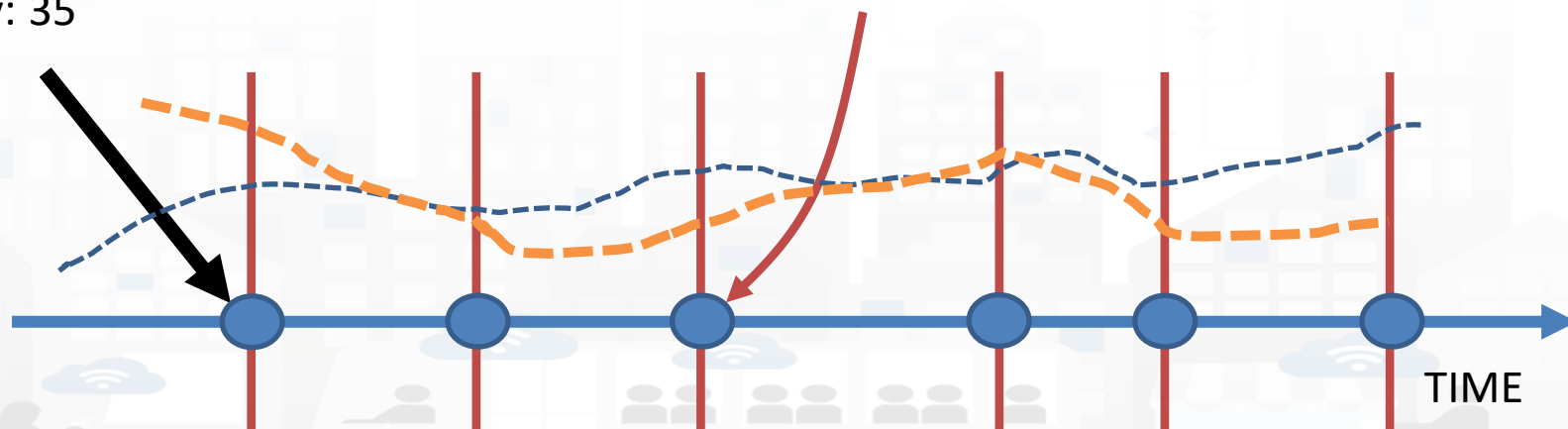
Entity: IOT Device

Sends a message

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

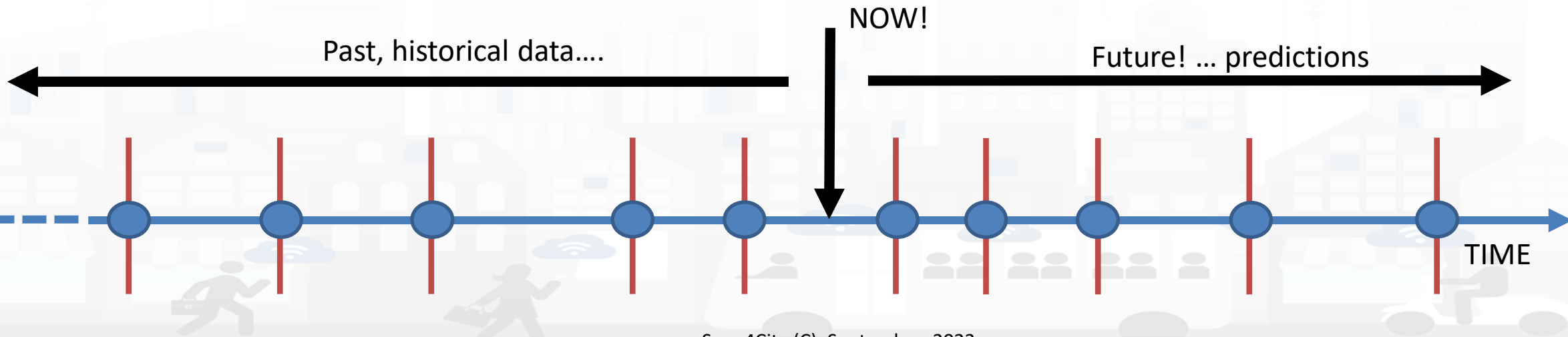
- A set of data coming from an Entity Instance / IoT Device with multiple sensor become a time series of values for devices.
  - For example: taking a new measure every 10 minutes (**Red Lines**)
  - Non regular rates can be valid data as well.
- Each new measure in Snap4City is conventionally time located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

| 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     |
|                  |      |          |



## 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) | Last Date           | Last Value | All selected (2) |                     | All selected (2) |
|------------------------------------|-------------------|-------------------|-------------------|------------|---------------------|--------------------|-------------------|-------------------|---------------------|------------|------------------|---------------------|------------------|
| High-Level Type                    | Nature            | Subnature         | Device/Model      | Broker     | Value Name          | Value Type         | Data Type         | Value Unit        |                     |            | 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          |

**High Level Types**

**Nature**

**Semantic  
Classific.**

**SubNature**

**Dev/Model name**

**Technical  
Source**

**Broker name**

**Value Name**

**Variables, names**

**Value Type**

**Data Type**

**Value Unit**

**Last Date/Time**

**Real  
Time**

**Last Value**

**Healthiness**

**Status**

**Last Check**

**Ownership  
Organization**

**For  
Admin**



## *Technical meaning*

<sup>Power</sup>  
Value Type

Value Unit

mW

Value Unit

KW

Data Type

Integer

Data Type

Float

## Link to Friend Sensor as ServiceURI: Value Type

Value Unit

URL

Value Unit

KW

Data Type

String, URL

Data Type

Float

# Example of Energy and its Value Units

### 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
- Notificator
- Data, my Data, OpenData
- Knowledge and Maps
- IOT Applications
- IOT Directory and Devices
- Resource Manager
  - View Resources
  - Managing Resources
  - Process Models
  - Processes in Execution
  - Process execution Archive
  - Dictionary Editor for Data Fields**
  - Doc: Resource Manager

### Dictionary Editor for Data Fields

+ Insert new Dictionary element

Filter by Dictionary type

Show 10

Search:

| Value Name       | Dictionary Type | Description                    | Data Types | Parent Value Name            | Child Value Name | Controls    |
|------------------|-----------------|--------------------------------|------------|------------------------------|------------------|-------------|
| Boats_and_shi... | subnature       | Boats And Ships Rental         |            | TransferServiceAndRenti...   |                  | EDIT DELETE |
| Bollard          | subnature       | Bollard                        |            | TransferServiceAndRenti...   |                  | EDIT DELETE |
| Bookshop         | subnature       | Bookshop                       |            | ShoppingAndService           |                  | EDIT DELETE |
| bool             | value unit      | boolean                        |            | dali_com_error, dali_dim...  |                  | EDIT DELETE |
| Botanical_and... | subnature       | Botanical & Zoolog. Gardens    |            | CulturalActivity             |                  | EDIT DELETE |
| Boxoffice        | subnature       | Boxoffice                      |            | Entertainment                |                  | EDIT DELETE |
| bpm              | value unit      | Beat per minute                |            | average_heart_rate, avera... |                  | EDIT DELETE |
| brightness_flag  | value type      | Brightness Flag                | string     |                              | #                | EDIT DELETE |
| broken_bikes     | value type      | Broken Bikes                   | integer    |                              | #                | EDIT DELETE |
| Building_and_... | subnature       | Build. & Indust. Clean. Activ. |            | Environment                  |                  | EDIT DELETE |

First << Prev 1...9 10 11...89 Next >> Last

**Value Type: Energy**

**Value Units:**

- Watt per hour
- KiloWatt per hour
- MegaWatt per hour

Snap4City (C), September 2023

## 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, SURJ, 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<br>(IOT Device Model) | Entity Instance<br>(IOT Device)               | Entity Message<br>at 23-12-2019T20:15:00   | Entity Message<br>at 23-12-2019T20:30:12   |
|------------------|------------------------------------|-----------------------------------------------|--------------------------------------------|--------------------------------------------|
| Broker           | Broker: <b>OrionUNIFI</b>          |                                               |                                            |                                            |
| Broker           | Protocol: <b>NGSI</b>              |                                               |                                            |                                            |
| 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:<br>"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:<br>"http://...../CarNF126GD" | TodayCarSURI:<br>"http://...../CarGF789KK" |
| Values           | Temperature: float, celsius        |                                               | 34                                         | 34                                         |

# Model meaning

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



Parts 3 and 5

# Entity / Device Model (2)

### Add new device

IOT Broker   Info   **Position**   Static Attributes   Values

**Latitude**  
Latitude is mandatory

**Longitude**  
Longitude is mandatory

Cancel   Confirm

### Edit Model - ChargingStationModel

General Info   IoT Broker   Static Attributes   Values

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

Add Value   Cancel   Confirm

# 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*



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

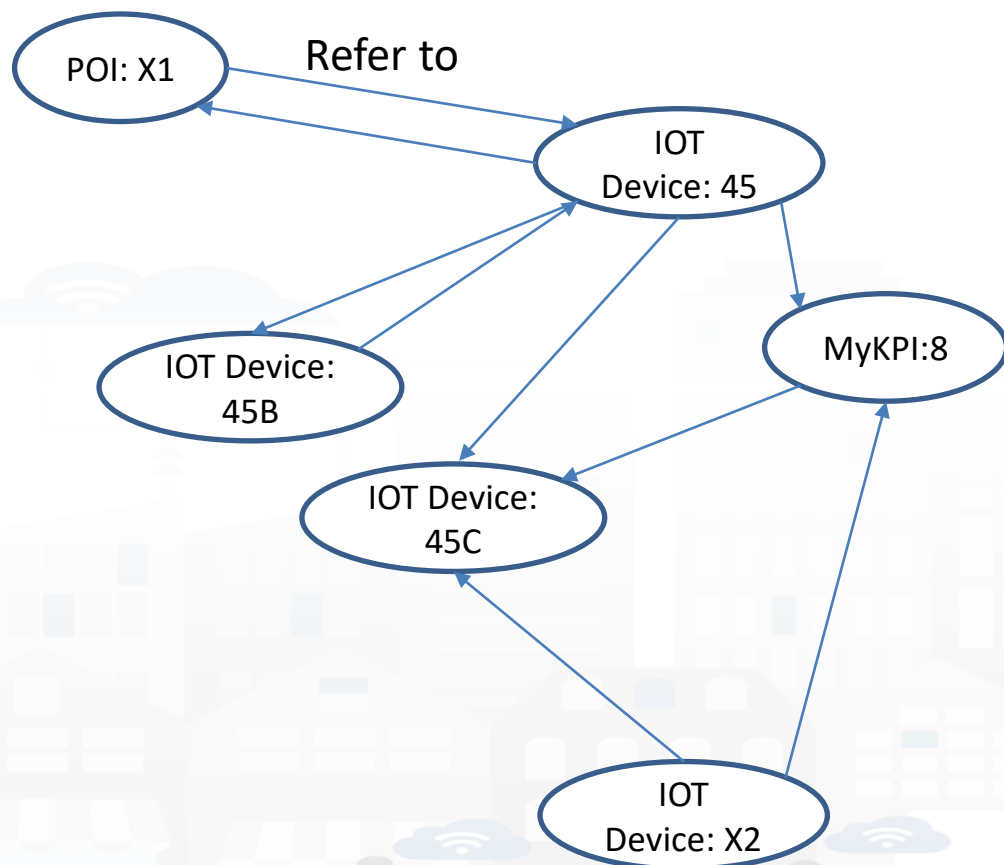
- They are triples
- Subject-predicate-Object
  - Subject=SURI
  - Predicate=p
  - Object = o

TOP

# *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/orion/Firenze2/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*

*Data Type: String*

*//any query: such as those of the Selector*

# 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     |
|---------------------|------------------|------------------------|---------------|
| <b>dateObserved</b> | <b>Timestamp</b> | <b>Timestamp in ms</b> | <b>String</b> |
| 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/s <sup>2</sup>  | float     |
| Status                    | status       | some coded status | String    |
| energyLevel               | energy level | percentage        | Float     |
| kmTotal                   | distance     | km                | Float     |
| thankLevel                | energy level | percentage        | Float     |
| vehicleEvent              | Identifier   | ServiceURI        | String    |



# Example 4

## IoT Device Model: VehicleEvent

Nature:.....

Subnature: .....

Lat,lon: .....

Device in Mobility: .....

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

**legenda**

- Entity Instance
- Entity Model
- Entity Messages with dateObserved

**Data Model of the Driver**

- Name: string
- Surname: string
- Age: number
- Weight: number
- Phone: string
- Email: string
- DriverAnalysisID: ServiceURI
- .....

Register to instantiate

**Driver: user45**

- Name: David
- Surname: Smith
- Age: 45
- Weight: 78 Kg
- Phone: +49345096103
- Email: [david89@gmail.com](mailto:david89@gmail.com)
- DriverAnalysis: <http://.../user45driveranalysis>
- .....

Write SURIs 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

**DriverAnalysis: user45driveranalysis**

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

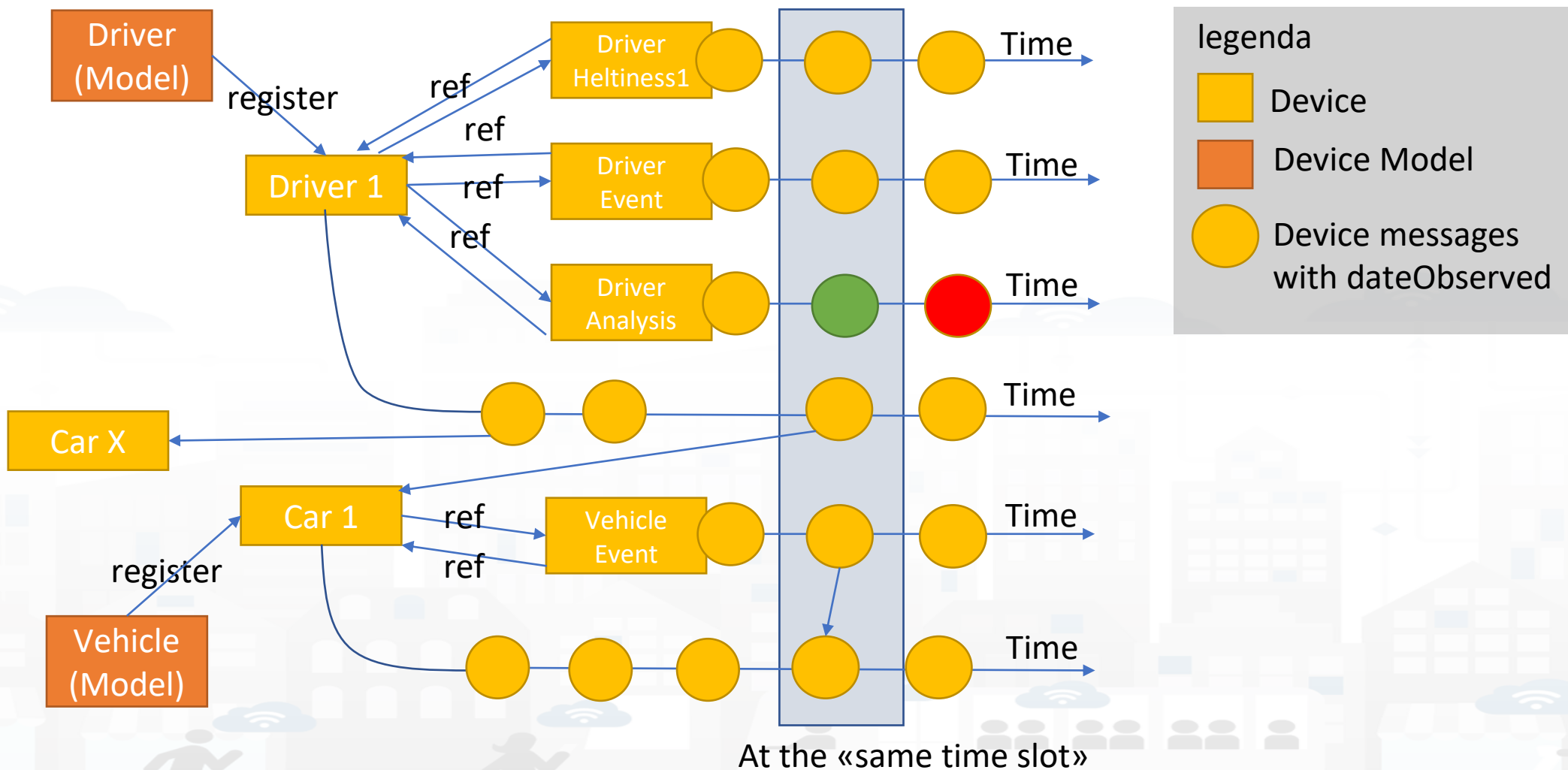
New update on user45driveranalysis by sending a message

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
- .....

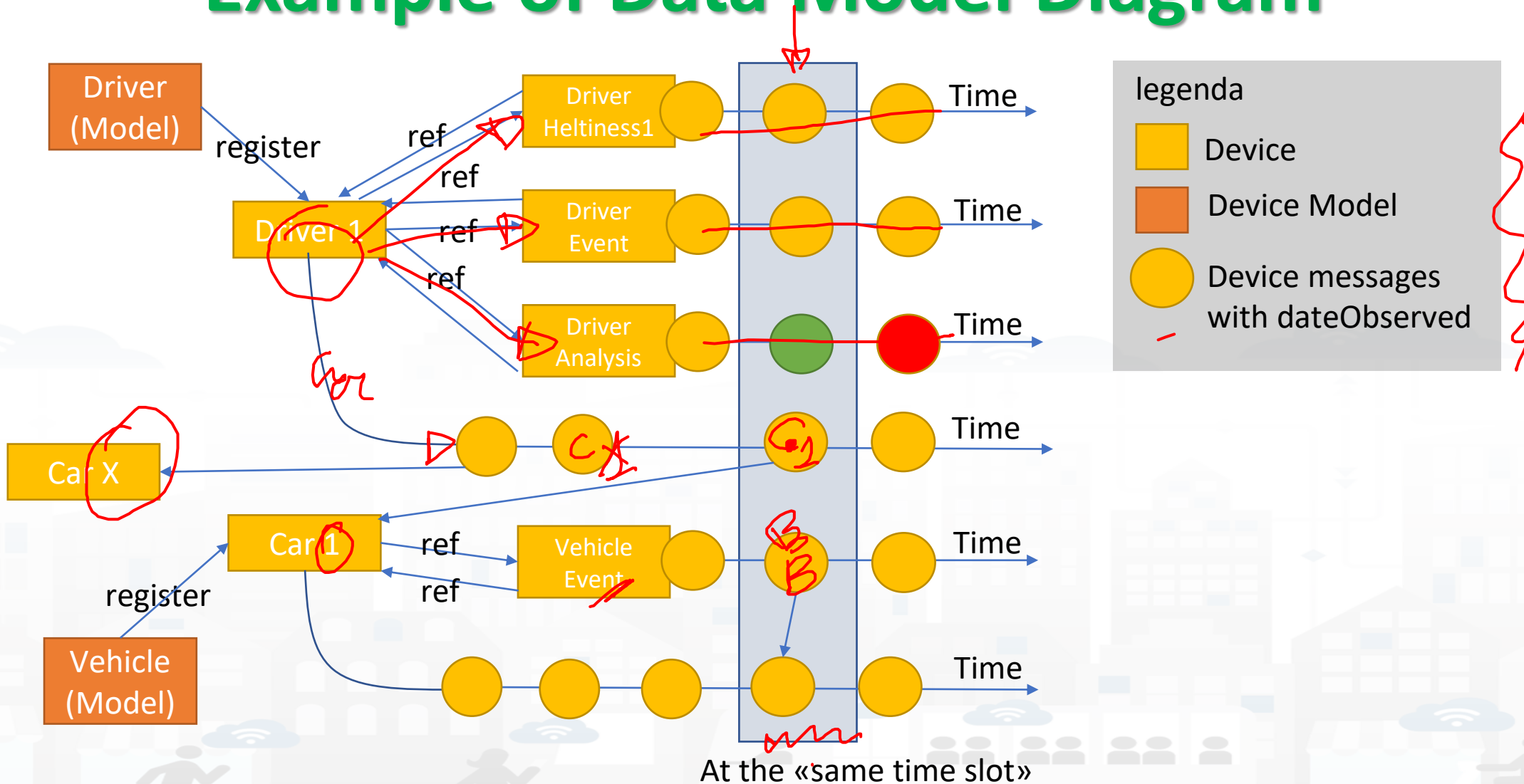
# Example of Data Model Diagram

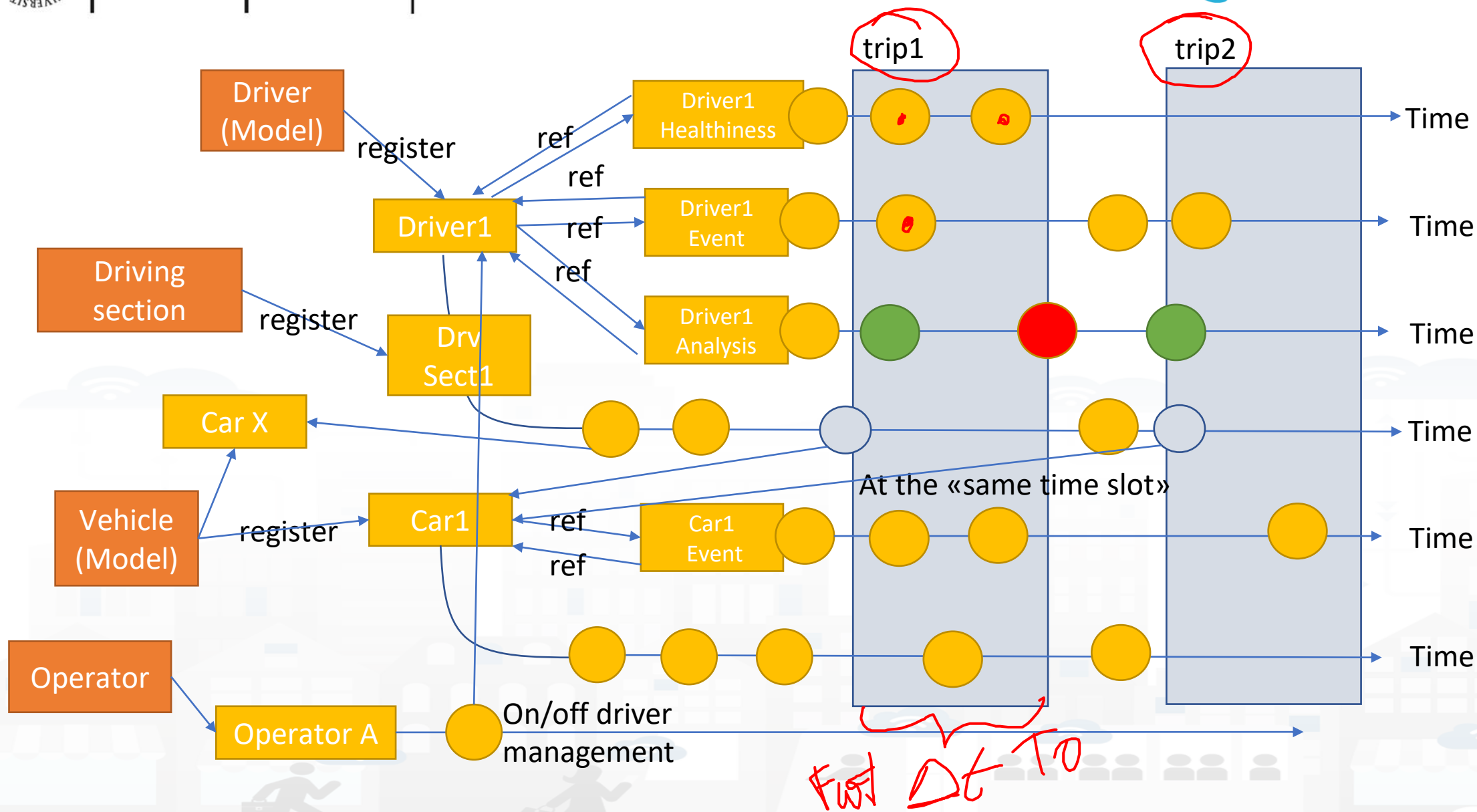


legenda

- Device
- Device Model
- Device messages with dateObserved

# Example of Data Model Diagram





TOP

# ***TOOLS for Data Design and HLT Exploitation***

Part 2

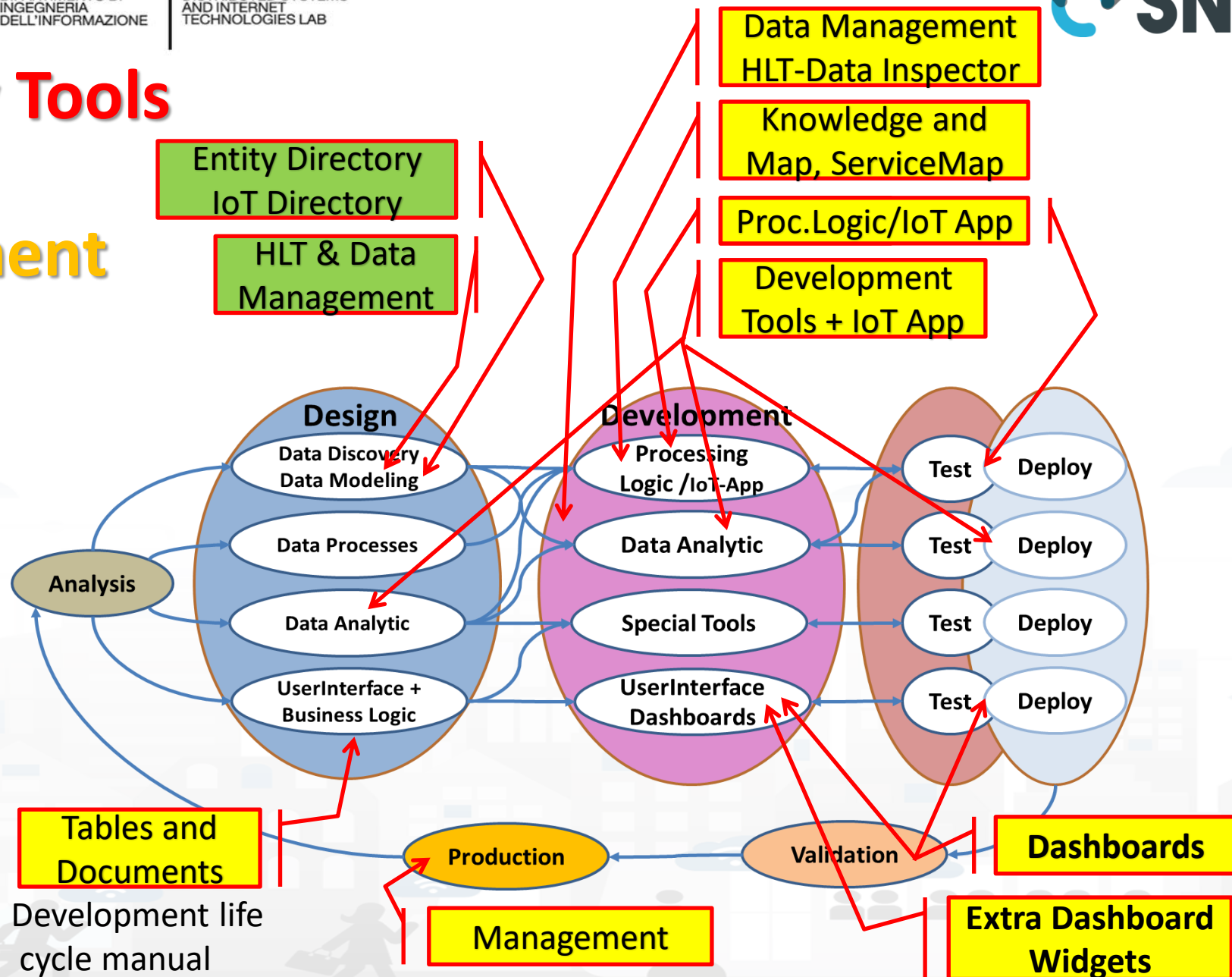
Part 5



# Snap4City Tools

vs

# Development Life Cycle



# Snap4city Data Ingestion Diagram

storage

Manual or automated  
Registration  
of Entities/Devices

**Entity/IoT  
Directory**

registration  
triples

NGSI



**Orion Broker**

IoT Orion Broker

IoT Orion Broker

IoT Orion Broker

IoT Orion Broker

**Brokers**

subscription note

NGSI

Real Time

**Knowledge Base**

Semantic Reasoners

Semantic Reasoners

Semantic Reasoners

SURI Link

**Indexing and Aggregating**

NIFI, OpenSearch

**Data Managers**

Data Managers

Entity Inspector



A number of KB  
federated



**Federation**  
Smart City API, ASCAPI

**API Manager**

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

Massive data flow  
entering

Massive data flow  
exiting

**Platform Automation:**

IoT Apps, Proc.Logics,  
Python, ...



**Data Analytics: ML, AI, XAI**



**Platform Control and Management**



# Any Entity has a Semantic Classification

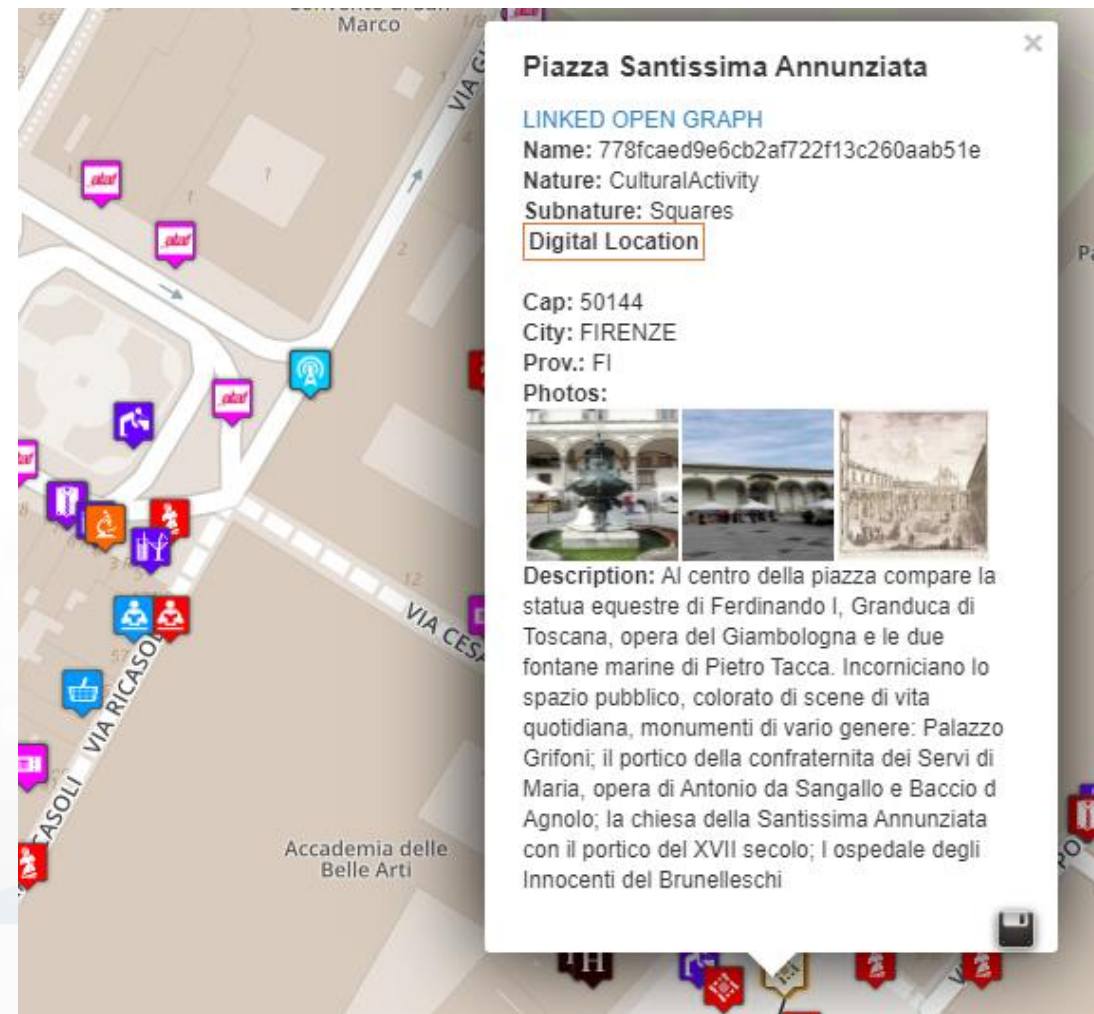
## Nature

- 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 and Devices

- My IOT Sensors and Actuators
- IOT Sensors and Actuators
- Entity Instances, IoT Devices
- IOT Brokers
- FIWARE Smart Data Models
- Entity Models/IoT Devices
- IOT Devices Bulk Registration
- Doc: IOT Directory and Devices
- Create an IOT Device Instance
- Create an IOT Device Model
- Add an IOT Device into Snap4City

Entity Models/IoT Devices

302 MODELS

Show 10 entries

Search:

New Model

|   | Device Model                        | Description                                                                          | Ownership | Organization | Kind     | Producer     | Device Type      | Edit | Delete | View |
|---|-------------------------------------|--------------------------------------------------------------------------------------|-----------|--------------|----------|--------------|------------------|------|--------|------|
| + | Raspberry snap4city 1               | Raspberry PI 3 Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM                  | DELEGATED | DISIT        | sensor   | Raspberry PI | Ambiental        |      |        | VIEW |
| + | Raspberry snap4city 2               | Raspberry PI 3 Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM                  | DELEGATED | DISIT        | sensor   | Raspberry PI | Ambiental        |      |        | VIEW |
| + | Arduino Uno                         | Arduino Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM                         | DELEGATED | DISIT        | sensor   | Arduino      | Ambiental        |      |        | VIEW |
| + | Arduino uno-bis                     | Arduino Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM                         | DELEGATED | DISIT        | sensor   | Arduino      | Ambiental        |      |        | VIEW |
| + | sigfox                              | SigFox Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM                          | DELEGATED | DISIT        | sensor   | SigFox       | Ambiental        |      |        | VIEW |
| + | Snap4AllButtonV1                    | Snap4AllButtonV1                                                                     | DELEGATED | DISIT        | sensor   | Snap4All     | Snap4AllButtonV1 |      |        | VIEW |
| + | Raspberry snap4city 1 - Certificate | Raspberry PI 3 Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM with certificate | DELEGATED | DISIT        | sensor   | Raspberry PI | Ambiental        |      |        | VIEW |
| + | datacenter3dht22                    | datacenter3dht22                                                                     | DELEGATED | DISIT        | sensor   | disit        | raspberry        |      |        | VIEW |
| + | Thermometer                         | This model represents a generic device that can measure a temperature                | DELEGATED | DISIT        | sensor   | Generic      | Ambiental        |      |        | VIEW |
| + | AirConditioner                      | Generic model representing a simple conditioner with only the status attribute       | DELEGATED | DISIT        | actuator | Generic      | Ambiental        |      |        | VIEW |

Showing 1 to 10 of 89 entries

Previous 1 2 3 4 5 ... 9 Next

# Entity Directory for Beginners

- Browse and see models and entities/devices of other users, that published 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 format 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

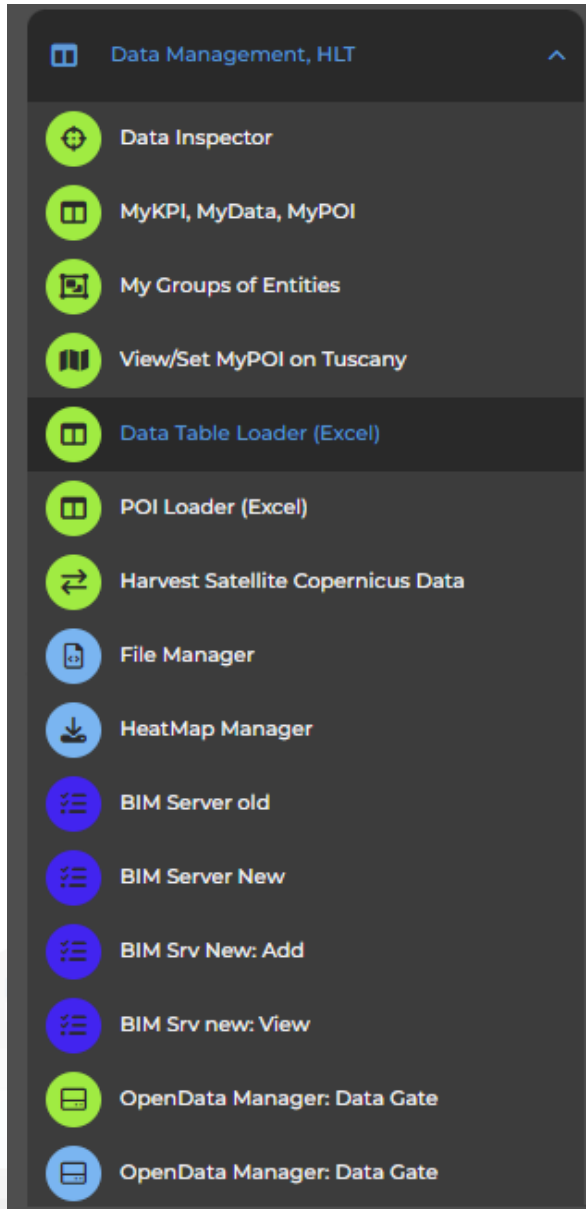
Any update and addition to the dictionary of snap4city.org has to be requested to snap4city@disit.org

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...](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/m <sup>3</sup> , µg/m <sup>3</sup> | float               |
| annual_C6H6_exceedance_count | annual_C6H6_exceedance_count | #                                          | integer, float      |
| annual_CO_average            | annual_CO_average            | ppm, mg/m <sup>3</sup> , µg/m <sup>3</sup> | float               |
| annual_CO_exceedance_count   | annual_CO_exceedance_count   | #                                          | integer, float      |
| annual_NO2_average           | annual_NO2_average           | ppm, mg/m <sup>3</sup> , µg/m <sup>3</sup> | float               |
| annual_NO2_exceedance_count  | annual_NO2_exceedance_count  | #                                          | integer, float      |
| annual_O3_average            | annual_O3_average            | ppm, mg/m <sup>3</sup> , µg/m <sup>3</sup> | 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

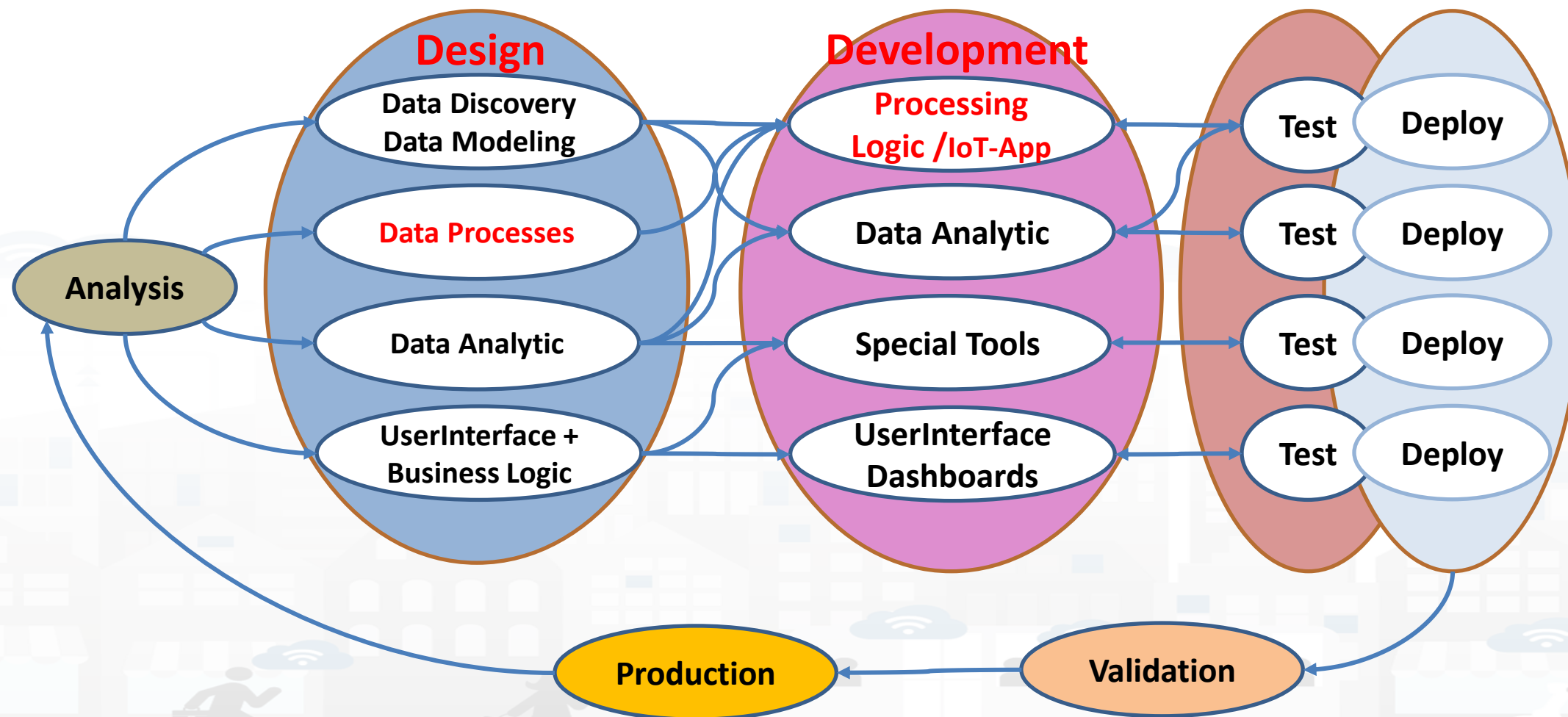


Part 3  
Part 5

# 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

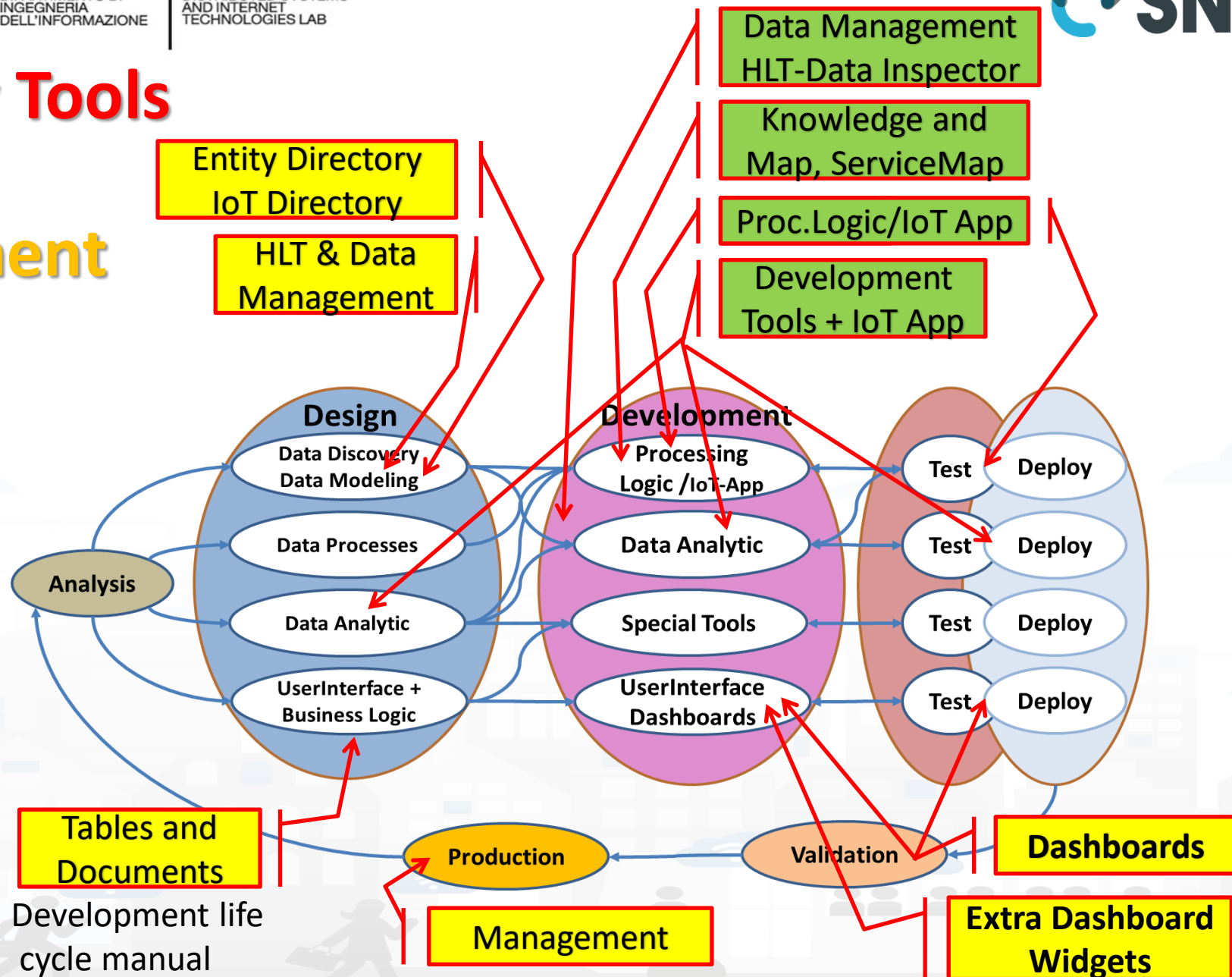




# Snap4City Tools

vs

# Development Life Cycle



Development life cycle manual

TOP

# *Design: Data Processes*

Part 3

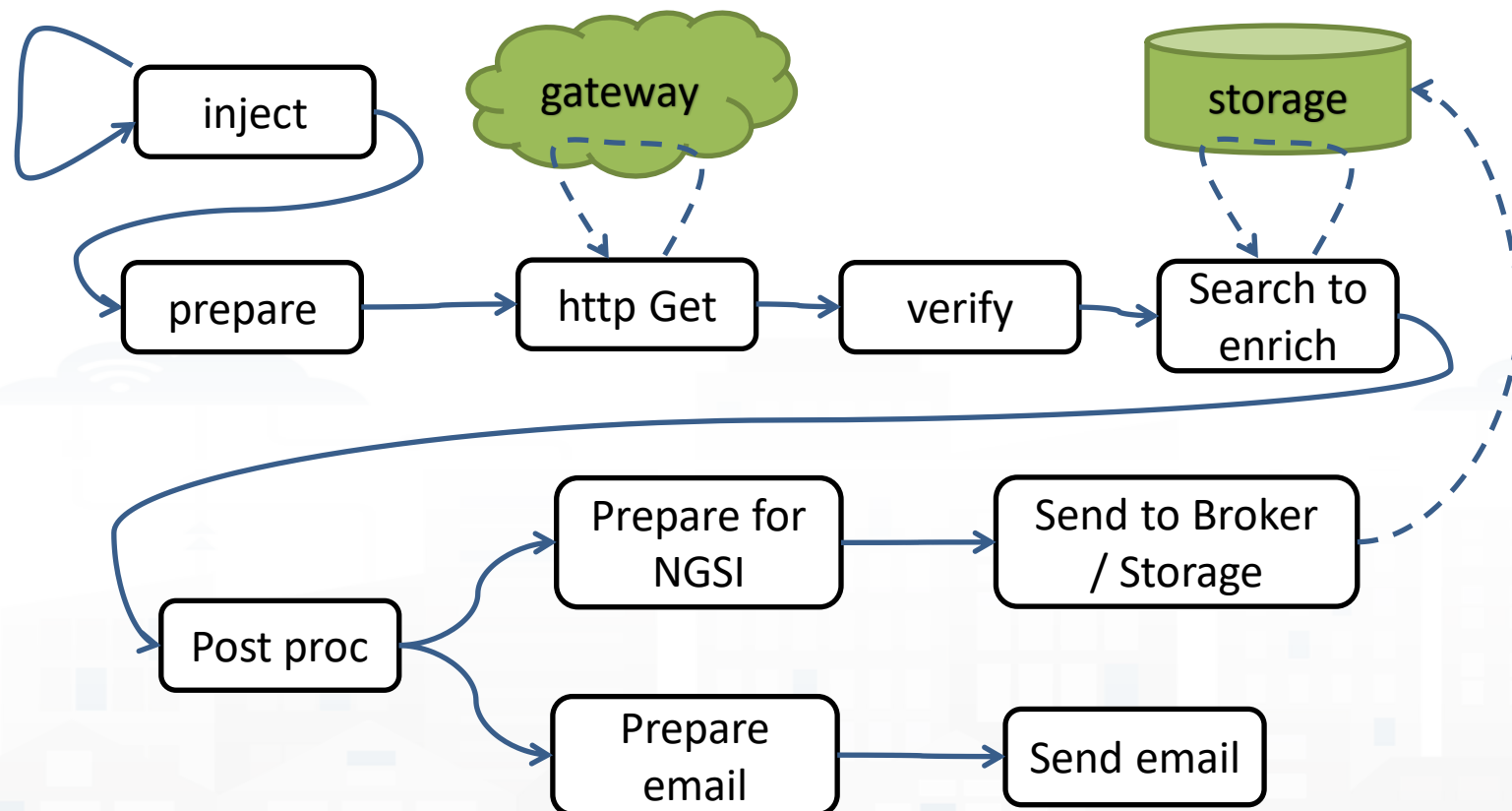
Part 5



# 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



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

TOP

# *Develop:* *Data Processes*

Part 3

Part 5



**FRED - Cloud Hosted Node-RED** Deploy

filter nodes | Hockl | Bluett | partic | Pi Co | Bridge | Slack | Web S | Examples | Test F | Mong | Thing | Test F | Flow | +

**subflows**

- Subflow 1
- Iterate
- sentiment switch
- Subflow 2
- Counter

**input**

- inject
- catch
- status
- link
- mqtt
- http
- websocket
- tcp
- udp
- fred
- directio

**Data Processing & Loops**

- inject
- f [1,2,3,4,5]
- f times 5
- iterate
- f
- msg
- msg payload

**Scraping web content**

- trigger
- get indexes
- Market indexes
- format indexes
- msg.payload

**generate data**

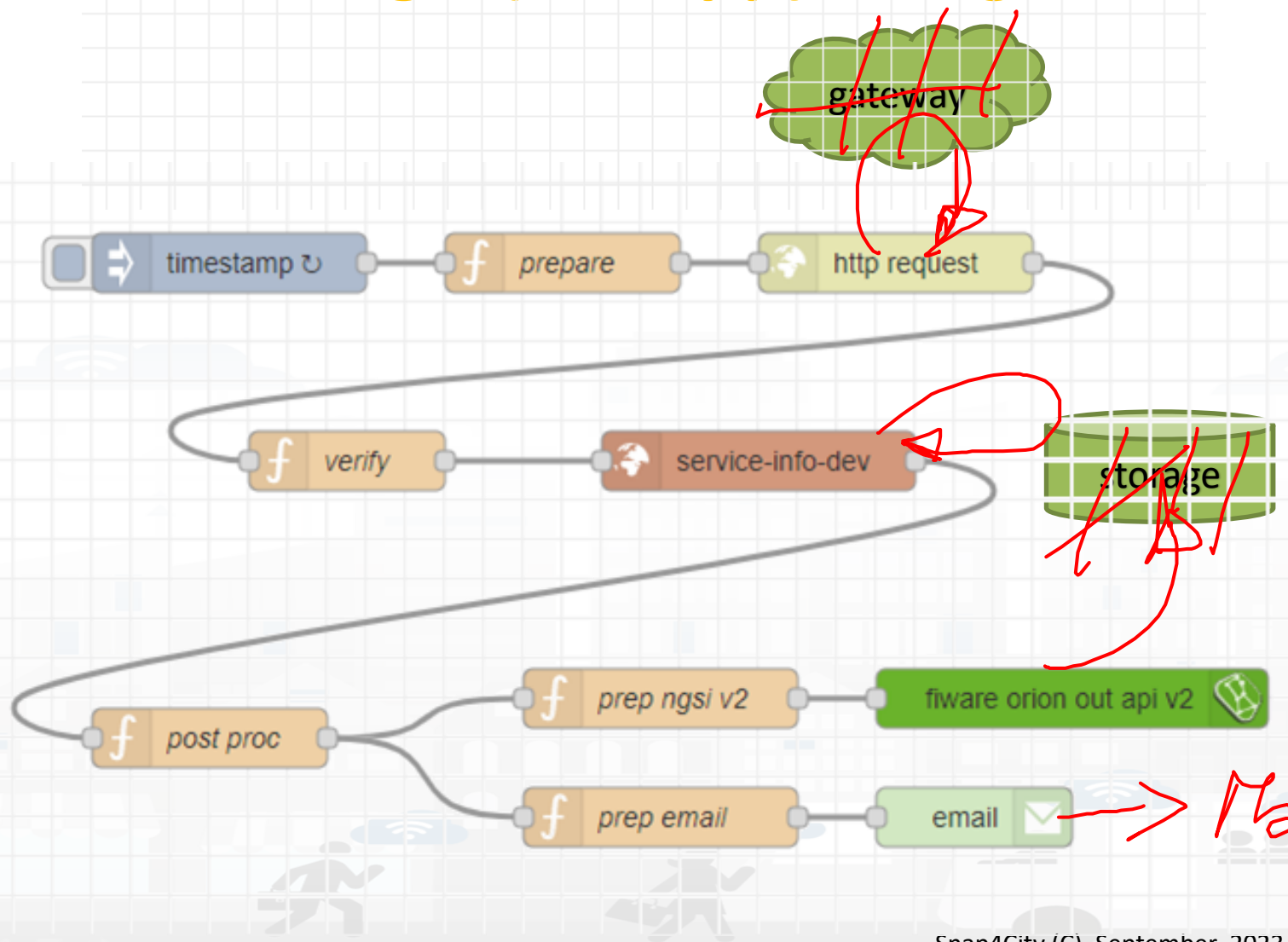
- generate data
- f ramp
- average
- f average 5 seconds

**timestamp**

- timestamp
- Counter
- msg

Provide Feedback

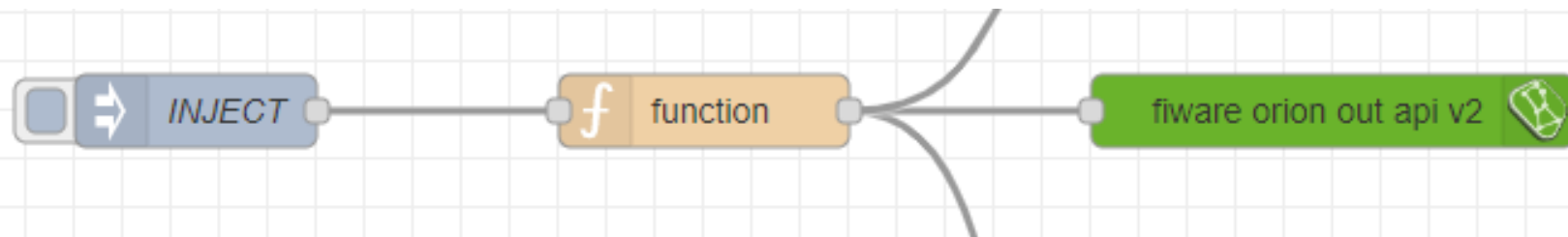
# Proc.Logic (IoT App) Design, for each Data/Event Flow



- Periodically activate the flow
- Call a gateway to get data
- Verify the correctness of data
- Enrich the data with other information coming from Cloud data into the storage
- Transform the data in the correct forma
- Send the data into the Broker, and thus send the data in the storage on a specific Entity Instance
- 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},
 "latitude":{"type":"float","value":"28.61810"},
 "longitude":{"type":"float","value":"11.34300"},
 "status":{"type":"integer","value":34}
 }
```

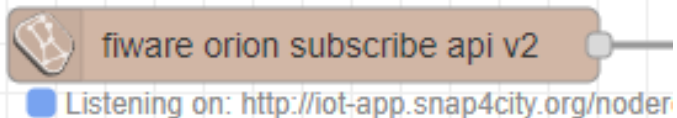
```
return msg;
```

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

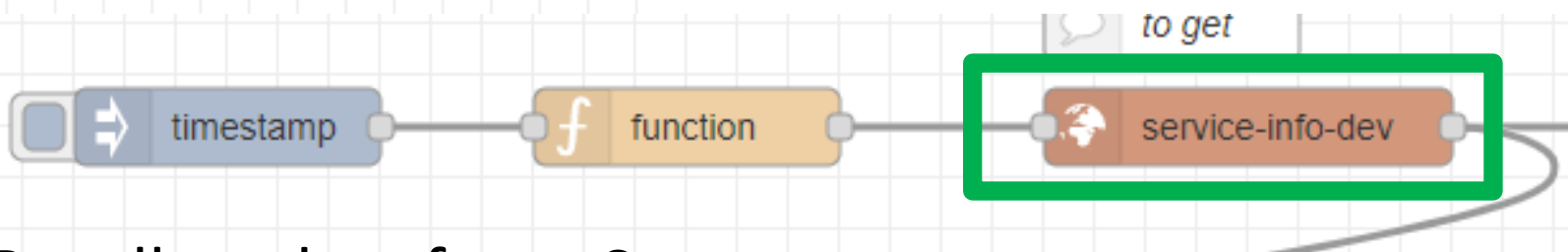
```
// it is a time serie
// it may move over time
// it may move over time
```



# 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



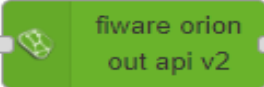
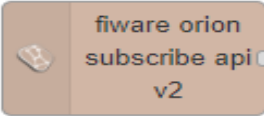
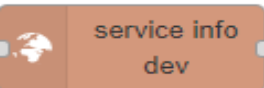
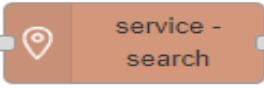

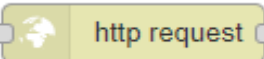


2) Recollect data from Storage

- 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              |

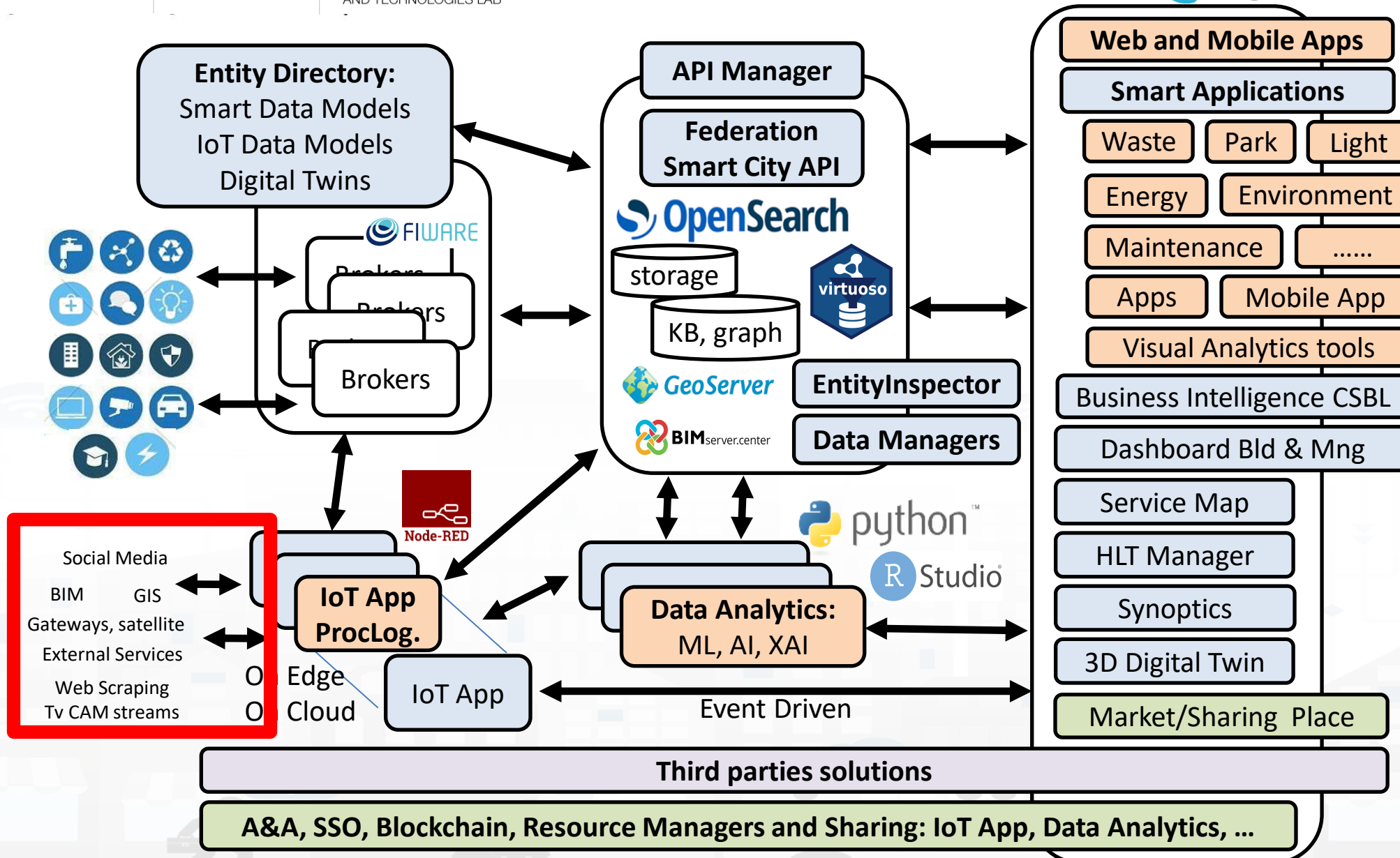
|  |                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                  |
|--|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
|  | <p>A block which is printing on <b>debug</b> view the data JSON passed in its input. Please note that the node can be tuned to provide only <u>msg.payload</u> or the full JSON message, change configuration of the node.</p>                                                                                                                                                                                                                        | <p>standard</p>  |
|  | <p>To <b>create an Entity Instance</b> (device instance) from a model prepared on Entity Directory (IoT Directory).</p>                                                                                                                                                                                                                                                                                                                               | <p>Snap4city</p> |
|  | <p>To <b>change the ownership</b> of an Entity Instance (IoT Device).</p>                                                                                                                                                                                                                                                                                                                                                                             | <p>Snap4city</p> |
|  | <p>To <b>delegate a certain Entity Instance</b> (IoT Device) to some other user for which you <u>have to</u> know the Nickname. Delegations can be: <u>Read access</u>, <u>Read write</u>, <u>Modify</u> (to modify the Entity Instance structure).</p>                                                                                                                                                                                               | <p>Snap4city</p> |
|  | <p>To <b>show something on Snap4City dashboard</b> 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.<br/>See in the following section for the Full list of Nodes for Snap4City Dashboard Widgets.</p> | <p>Snap4city</p> |
|  | <p><b>MQTT broker listener</b>, 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.</p>                                                                                                                                                                                                                         | <p>standard</p>  |
|  | <p><b>DATA ANALYTICS</b><br/>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.</p>                                                                                                                                                                                | <p>Snap4city</p> |
|  | <p><b>SPLIT:</b> 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.</p>                                                                                                                                                                                                                     | <p>standard</p>  |
|  | <p><b>JOIN:</b> This block takes in input a set of messages and join/merge them into a single message (string, buffer, <u>array</u> or object, etc.), on the basis of specific criteria.</p>                                                                                                                                                                                                                                                          | <p>standard</p>  |

# 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:

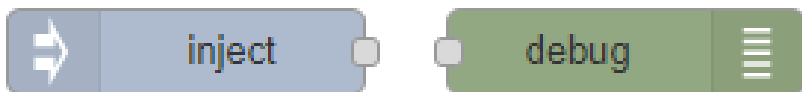
- **Data ingestion:** more than 100 protocols IOT and Industry 4.0, web Scraping, external services, any protocol database, etc.
- **Data access:** save/retrieve data, query search on expert system, georeverse solution, search on expert system Km4City ontology, call to Smart City API, etc.
- **Data Transformation/transcoding:** 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

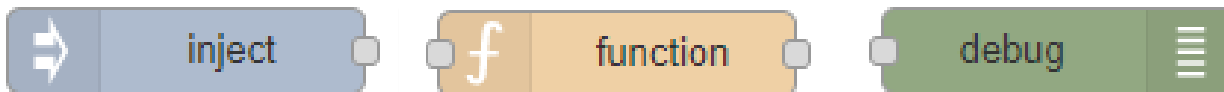


# Some patterns

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



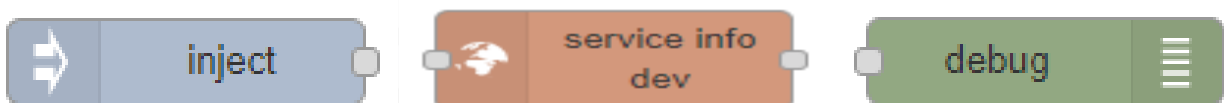
- 2) Hello world of node-red at two steps, the inject provides a push while a JSON is created into the function as `msg.payload = {.....}` and sent/shown to/by the debug.



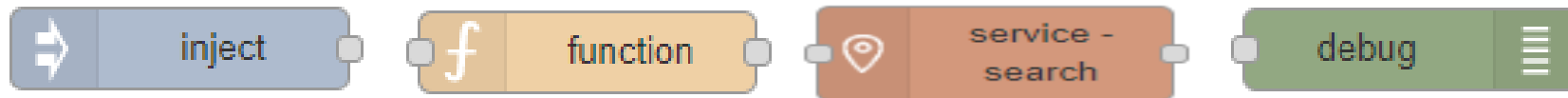
- 3) Event data reception from an MQTT broker, transformation and send it to the storage pushing data into the Orion Broker V2.



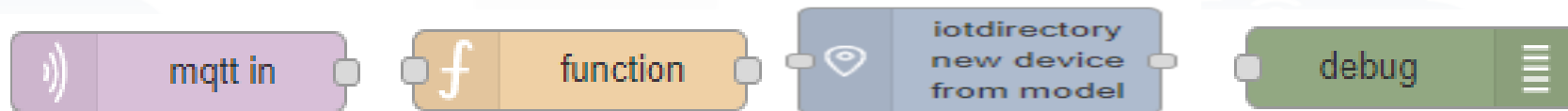
- 4) request on inject of a SURJ to the storage to see data on debug.



- 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):



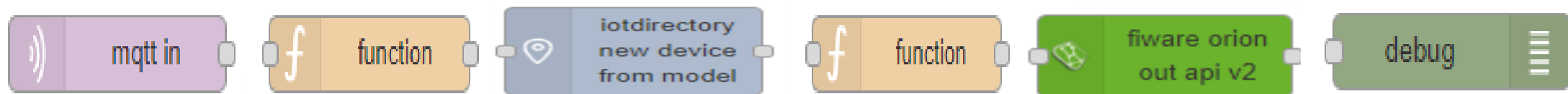
- 3) Preparation of data parameters on function, request computing Data Analytic, see data result on debug.



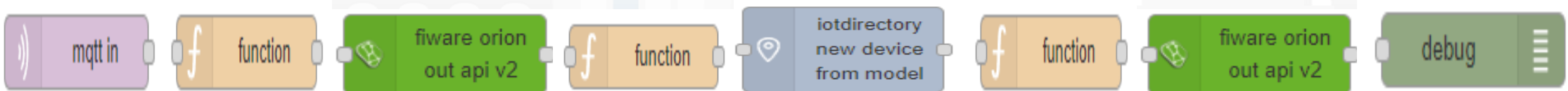


## Typical strange patterns that may be not efficient in most cases:

- 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).

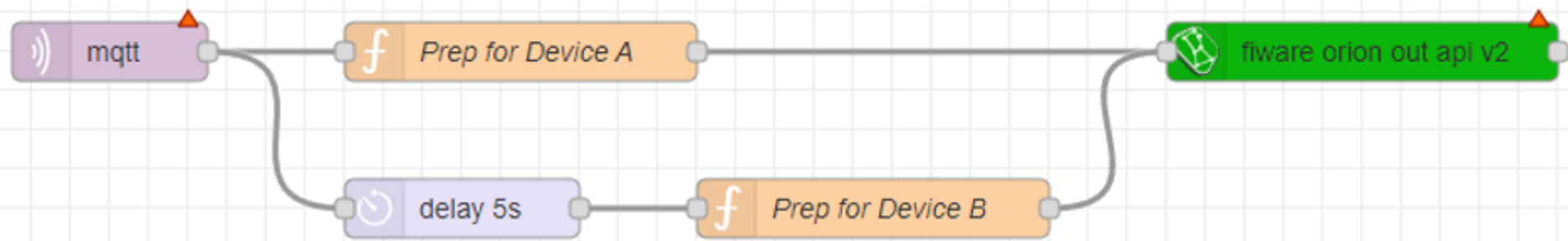


In most cases, it should be done the opposite: try to send the Entity Message, if it fails than create a new Entity Instance by known model, and if successful send again the Entity Message, or just wait for the new message to save it the first.



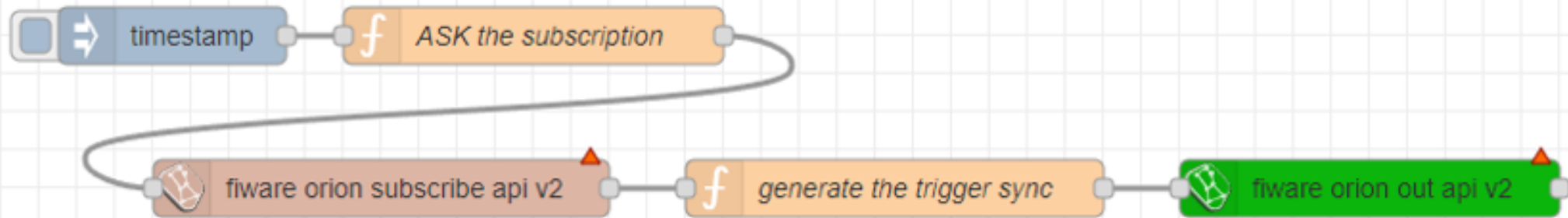
# Sync data changes on Entities from an Event

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:

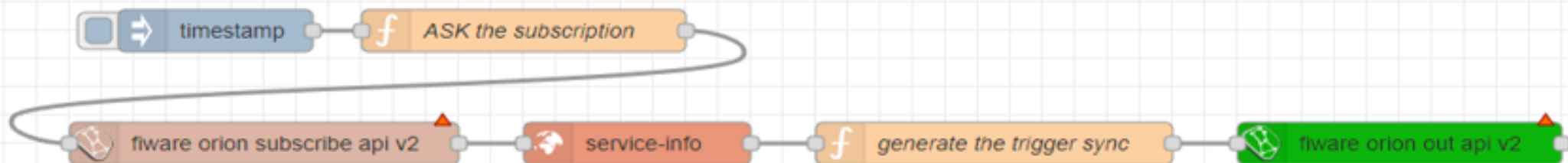


# Subscribe on event from Broker, be carefull....

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:



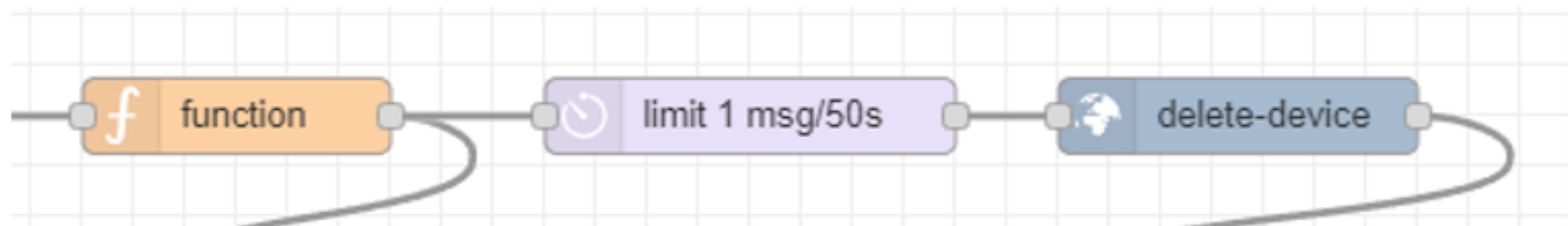
# Delete Devices

## 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.

TOP

# *Proc.Logic / IoT App*

## *Development*

Part 3

Part 5



# Snap4city Data Ingestion Diagram

storage

Manual or automated  
Registration  
of Entities/Devices

**Entity/IoT  
Directory**

registration  
triples

NGSI



**Orion Broker**

IoT Orion Broker

IoT Orion Broker

IoT Orion Broker

IoT Orion Broker

**Brokers**

subscription note

NGSI

Real Time

**Knowledge Base**

**Semantic Reasoners**

**Semantic Reasoners**

SURI Link

**Indexing and Aggregating**  
NIFI, OpenSearch

**Data Managers**

**Data Managers**

**Entity Inspector**



A number of KB  
federated



**Federation**  
Smart City API, ASCAPI

**API Manager**

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

**Platform Automation:**  
IoT Apps, Proc.Logics,  
Python, ...



**R Studio** **Data Analytics: ML, AI, XAI** **python**

**Platform Control and Management**

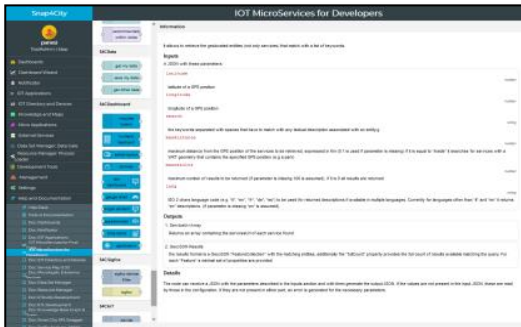
Massive data flow  
entering

Massive data flow  
exiting

# Proc.Logic / IoT App Development

IOT Discovering

MicroServices collections



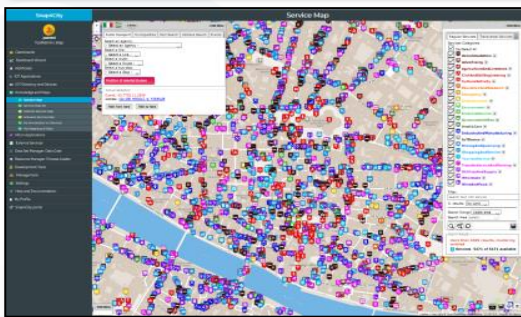
My IoT Applications



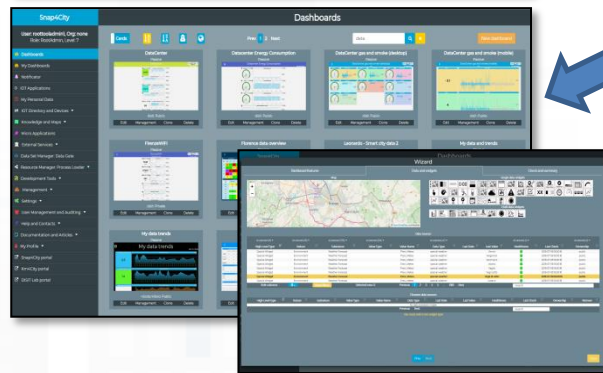
IoT App. Editor



Generating IoT App With Dashboard



ServiceMap Discovery  
Knowledge Base, Km4CITY

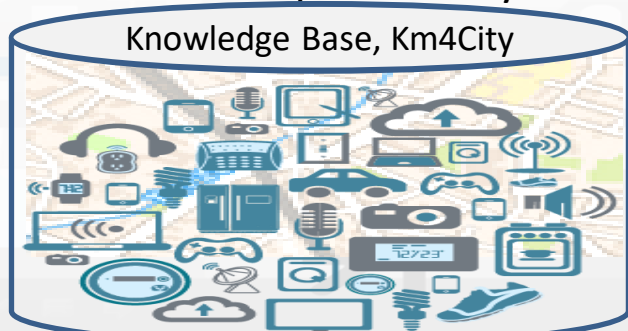


Dashboard Collection,  
Editor and Wizard

Sharing/saving  
reusing IoT App



Resource Manager  
 **GitHub**



Dashboards (Public)



My Snap4City.org



Tour Again

[www.snap4solutions.org](http://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

filter nodes

Demo 1

Flow 1

Flow 2

Flow 3

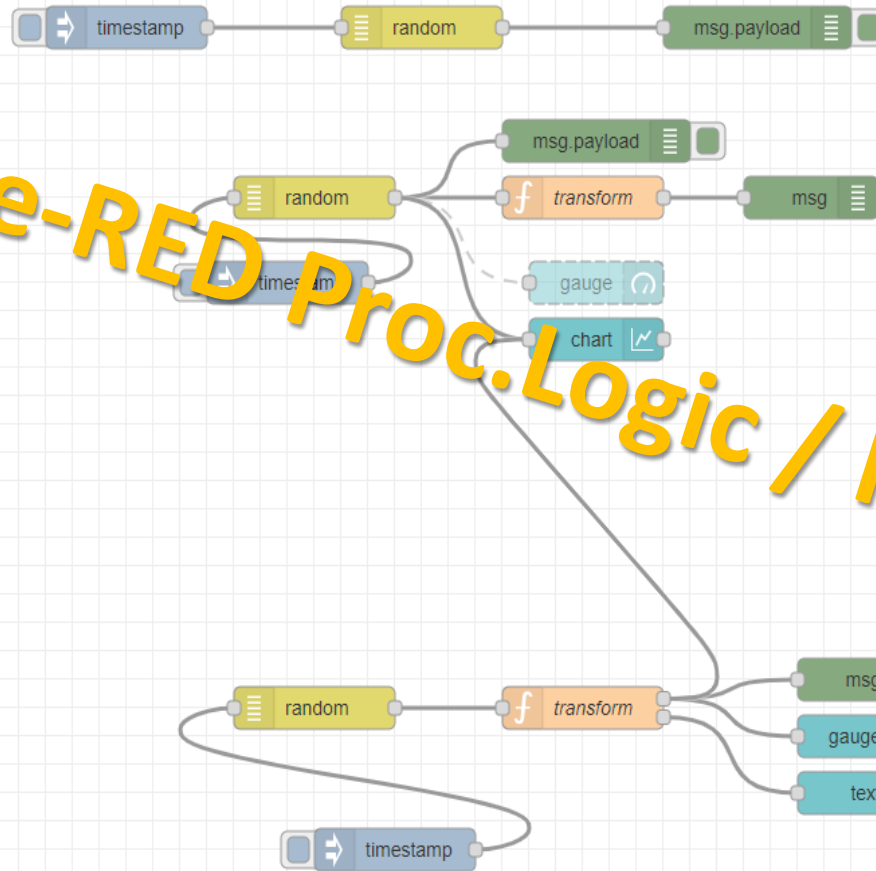
debug

common

- inject
- debug
- complete
- catch
- status
- link in
- link call
- link out
- comment

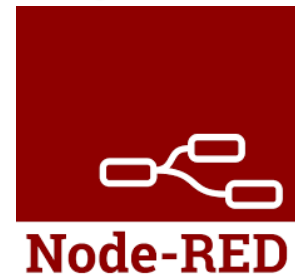
function

- function
- switch
- change
- range
- template
- delay
- trigger
- exec



Node-RED Proc.Logic / IoT App Editor

Event Driven, real time data ingestion





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



The screenshot shows the Node-RED block palette with the following categories and blocks:

- common**: inject, debug, complete, catch, status, link in, link out, comment
- function**: function, switch, change, range, template, delay, trigger, exec, zip, md5, soap request, string, xml converter, random, rbe
- network**: mqtt in, mqtt out, http in, http response, http request, websocket in, websocket out, tcp in, tcp out, tcp request, udp in, udp out
- input**: amqp in, amqp2 in, stomp in
- output**: amqp out, amqp2 out, stomp out
- sequence**: split, join, sort, batch
- parser**: csv, html, json, xml, yaml, base64, msgpack
- storage**: file, file in, watch, ftp in, mysql, tail
- social**: email, twitter in, email, twitter out, advanced, feedparser, NGSI, NGSI Entity, NGSI v2ToLD, lwm2m, lwm2m client in, lwm2m client out, location, turf, worldmap, worldmap in, tracks, convex hull, time, sunrise
- dashboard**: button, dropdown, switch, slider, numeric, text input, date picker, colour picker, form, text, gauge, chart, audio out, notification, ui control, template

+ on IOT Edge Raspberry

The screenshot shows the Node-RED block palette with the following categories and blocks:

- social**: e mail, twitter, irc, e mail, twitter, irc, google plus, google places, google calendar
- storage**: tail, file, mongodb, file, mongodb
- Raspberry Pi**: rpi gpio, rpi gpio, rpi mouse, rpi keyboard, rpi keyboard, camerapi takephoto, rpi dht22, imagecapture, ledborg, Sense HAT, Sense HAT
- network**: ping

# Node-RED Basic Blocks

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

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

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

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

Categories shown in the screenshot:

- common**: inject, debug, complete, catch, status, link in, link out, comment
- function**: function, switch, change, range, template, delay, trigger, exec, zip, md5, soap request, string, xml converter, random, rbe
- network**: mqtt in, mqtt out, http in, http response, http request, websocket in, websocket out, tcp in, tcp out, tcp request, udp in, udp out
- input**: amqp in, amqp2 in, stomp in
- output**: amqp out, amqp2 out, stomp out
- sequence**: split, join, sort, batch
- parser**: csv, html, json, yaml, base64, msgpack
- storage**: file, file in, watch, ftp in, mysql, tail
- social**: email, twitter in, email, twitter out, advanced, feedparser, NGSI, NGSI Entity, NGSI v2toLD, Iwm2m, Iwm2m client in, Iwm2m client out, location, turf, worldmap, worldmap in, tracks, convex hull, time, sunrise
- dashboard**: button, dropdown, switch, slider, numeric, text input, date picker, colour picker, form, text, gauge, chart, audio out, notification, ui control, template
- social (on IOT Edge Raspberry)**: e mail, twitter, irc, #, #, #, #, google plus, google places, google calendar
- Raspberry Pi**: rpi gpio, rpi gpio, rpi mouse, rpi keyboard, camerapi takephoto, rpi dht22, imagecapture, ledborg, Sense HAT, Sense HAT, network, ping
- storage**: tail, file, mongodb, file, mongodb

99

# IoT Applications

- **Data ingestion:** more than 70 protocols IOT and Industry 4.0, web Scraping, external services, any protocol database, etc.
- **Data access:** save/retrieve data, query search on expert system, georeverse solution, search on expert system Km4City ontology, etc.
- **Data Transformation/transcoding:** binary, hexadecimal, XML, JSON, String, any format
- **Integration:** CKAN, Web Scraping, FTP, Copernicus satellite, Twitter Vigilance, Workflow OpenMaint, Digital Twin BIMServer, any external service REST Call, etc.
- **Manipulation of complex data:** heatmaps, scenarios, typical time trend, multi series, calendar, maps, etc.
- **Access to Smart City Entities and exploitation of Smart City Services:** transport, parking, POI, KPI, personal data, scenarios, etc.
- **Data Analytic:** managing Python native, calling and scheduling Python/Rstudio containers as snap4city microservices (predictions, anomaly detection, statistics, etc.)
- **User interaction on Dashboard:** get data and message from the user interface, providing messages to the user (form, buttons, switches, animations, selector, maps, etc. )
- **Custom Widgets:** SVG, synoptics, animations, dynamic pins on maps, etc
- **Event management:** Telegram, Twitter, Facebook, SMS, WhatsApp, CAP, etc.
- **Hardware Specific Devices:** Raspberry Pi, Android, Philips, video wall management, etc.



# 2023 collection

## Two Snap4City Libraries



- > common
- > function
- > network
- > input
- > output
- > sequence
- > parser
- > storage
- > social
- > advanced
- > Advanced FTP
- > location
- > NGSI
- > Iwm2m
- > S4C SearchDev
- > S4C Utility
- > S4C Mapping
- > S4C Management
- > S4C DataAnalytic
- > S4C BigData
- > S4C IoTApp
- > S4C OpenMaint
- > S4C IoT
- > S4C Whatif
- > S4C Search
- > S4C Data
- > S4C KPIData
- > S4C Dashboard
- > S4C Sigfox
- > S4C LogDev
- > S4C View
- > S4C Social
- > dashboard
- > time

**S4C SearchDev**

- service search
- service search near gps position
- service search near service
- service search within gps area
- service search within wkt area
- service search within stored wkt area
- service search by municipality
- service search by queryid
- full text search dev
- full text search within wkt area

**full text search within gps area**

**full text search near gps position**

**full text search exp**

**event search dev**

**event search exp**

**event search within wkt area**

**event search within gps area**

**event search near gps position**

**address search near gps position**

**geometry search near gps position**

**address poi search by text**

**address poi search by text exp**

**address poi search by text near gps position**

**bus routes search**

**bus routes search near gps position**

**bus routes search within gps area**

**bus routes search within wkt area**

**bus routes**

**tpl routes**

**tpl stops**

**S4C Utility**

- service info dev
- distance from coordinates

**point within polygon**

**routing**

**heatmap picker**

**coordinates to address**

**service info**

**edge-tunnel-to-cloud**

**S4C Mapping**

- service info mapped
- mapping
- set mapping

**get job detail**

**get triggers of job**

**get job group names**

**get trigger group names**

**get paused trigger groups**

**get job fire times**

**get system status**

**trigger job**

**pause all**

**pause trigger**

**pause triggers**

**resume all**

**resume job**

**resume jobs**

**resume trigger**

**resume triggers**

**notifier history events**

**S4C DataAnalytic**

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

**S4C Search**

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

**event search within polygon**

**event search along path**

**event search usr**

**address search near marker**

**geometry search near marker**

**address poi search by text usr**

**address poi search by text near marker**

**address poi search by text within circle**

**bus routes search near marker**

**bus routes search within circle**

**bus routes search within polygon**

**tpl agencies**

**tpl lines**

**tpl routes by agency**

**tpl routes by line**

**tpl stops by route**

**tpl stop timeline**

**recommendatio within circle**

**value type search near marker**

**value type search within circle**

**value type search within polygon**

**value type search along path**

**S4C Data**

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

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



# 2023 collection

## Two Snap4City Libraries



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

- get other activity on my data
- save my data
- get my annotation
- get anonymous data
- get other data

- S4CKPIData**
  - get my kpdata
  - get delegated kpdata
  - get public kpdata
  - get my kpdata values
  - get public kpdata values
  - get delegated kpdata values
  - delegate my kpdata
  - get iotapps using my kpdata
  - save my kpdata values

- S4CDashboard**
  - coordinates - from - map
  - impulse - button
  - numeric - keyboard
  - switch - button
  - dimmer
  - geocator
  - dropdown
  - form
  - gauge - chart
  - single - content
  - speedometer
  - horizontal - single - bar
  - vertical - single - bar
  - web - content
  - time - trend
  - bar - series
  - radar - series
  - pie - chart
  - curved - line - series

- table - content
- calendar
- speak - synthesis
- selector - to - map
- dashboard - map
- event - driven - my - kpi
- synoptic - read
- synoptic - write
- synoptic - subscribe
- S4COpenMaint**
  - om get processes
  - om get teams
  - om get components
  - om get plants
  - om get status
  - om create new process
  - om advance process
  - om details process
  - om delete process

- S4CIoT**
  - iotdirectory new device from model
  - delegate my device
  - change ownership my device
  - iot directory
  - iot directory link
  - iot directory link
  - iotdirectory get device
  - fiware orion subscribe v1
  - fiware orion query v1
  - fiware orion update v1
  - fiware orion out v1
  - fiware orion subscribe api v2
  - fiware orion query api v2
  - fiware orion update api v2
  - fiware orion out api v2
  - fiware orion in v2
  - fiware orion query v2
  - fiware orion out v2
  - fiware orion in v2(url syntax v1)
  - fiware orion query v2(url syntax v1)
  - fiware orion out v2(url syntax v1)
  - fiware orion in v2(url syntax v1)
  - fiware orion query v1
  - fiware orion out v1
  - snaph4all button

- S4CLogDev**
  - event log
- S4CView**
  - show micro web app
  - show general iframe
- S4CSocial**
  - twitter last channel
  - twitter last tweet
- S4CSigfox**
  - sigfox device filter
  - sigfox
- S4CIoT**
  - save typical time trends
  - get typical time trends
- S4CWhatif**
  - get my scenarios
  - save a scenario

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

AND: From Resource Manager

- UserCreated**
  - Twitter Herit Data Sentiment Analysis Channel
  - Twitter Herit Data Sentiment Analysis Search
  - TwitterVigilance Herit Data Tw Rtw Channel
  - TwitterVigilance Herit Data Tw Rtw Search
  - Sci Hub Copernicus Completed
  - Sci Hub Copernicus Indexed
  - Sci Hub Copernicus Polygon

We suggest also to install:

- NGSI**
  - NGSI Entity
  - NGSI Dataset
  - NGSI Update
  - NGSI Subscription
  - NGSI v2ToLD
- social**
  - email
  - twitter
  - email
  - twitter
- subflows**
  - triplesToVirtual
- location**
  - utm
  - turf
  - worldmap
  - worldmap in
  - tracks
  - convex hull
- Advanced FTP**
  - Advanced FTP
  - Advanced FTP Logger

User: rootooladmin1, Org: DISIT  
Role: RootAdmin, Level: 7



Prev 1 2 3 ... 9 Next

Filter

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

● 2018-09-14T04:44

IOT Edge App

owner: badii

● 2018-09-21T03:19

IOT Edge App

owner: panesi

● 2018-10-19T16:07

IOT Edge App

owner: pb3

● 2018-10-19T17:17

IOT Edge App

owner: pb3

● 2018-10-22T11:57

IOT Edge App

owner: semolarudy

● application

IOT Application

owner: tester5

● Bib APP

IOT Application

owner: semolarudy

● ChargingStations

IOT Application

owner: comunedashres

● Deprecated - SiiMobilityControlRoom

IOT Application

owner: badii

● SamsungGalaxyS4Barcode

IOT Edge App

owner: badii

● esercitazione

IOT Application

owner: tester2

● Iot-App

IOT Application

owner: tester14

# Resource Manager: public and sharing

## Snap4City

User: rootooladmin1, Org: DISIT  
Role: RootAdmin, Level: 7

- Dashboards
- My Dashboards
- Notifier
- 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

- 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

### AMMADashSnap4City-30minview-v2-152...

developer1: Private  
Username: developer1  
Resource type: AMMA  
Nature: ToBeDefined  
Description: AMMA snap4city dash...  
★★★★★  
View Edit Publish Owner

### DeveloperDash-V3-1523555417880

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

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

### Developer Dashboard New-1526308876256

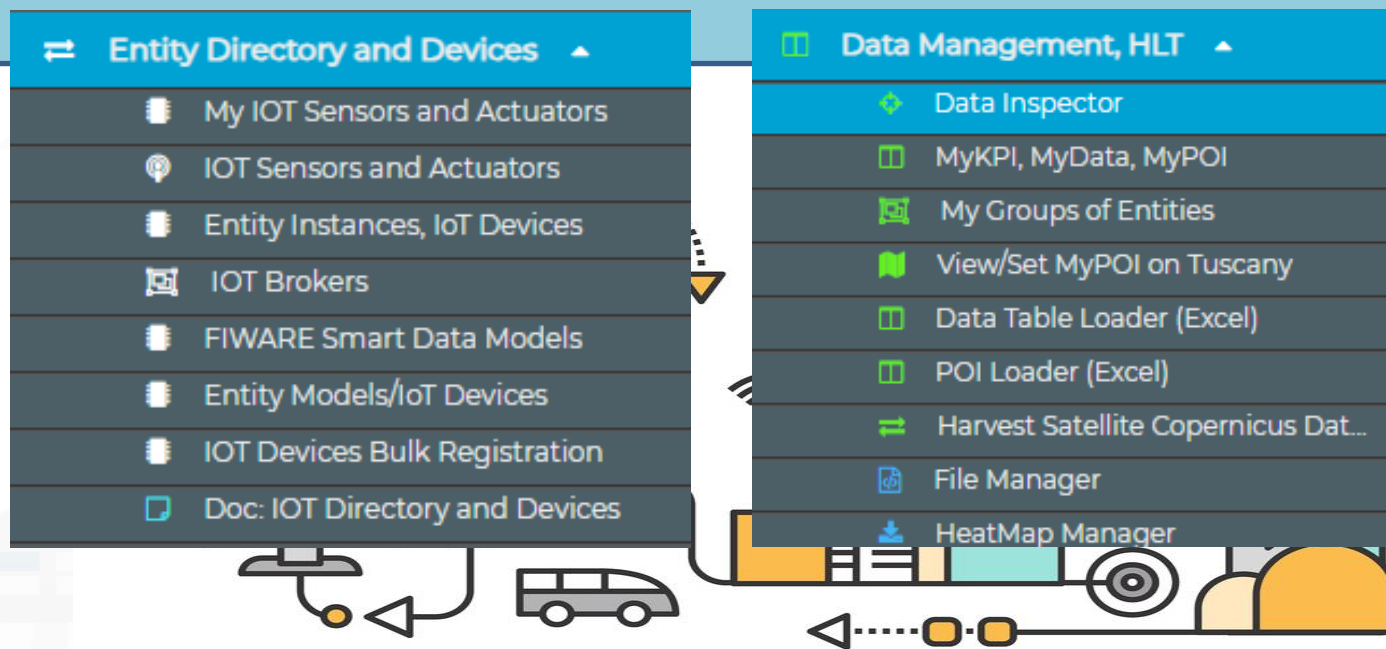
developer1: Private  
Username: developer1  
Resource type: DevDash  
Nature: ToBeDefined  
Description: Developer Dashboard ...  
★★★★★  
View Edit Publish Owner

### ResDash Docker-1526308998809

developer1: Private  
Username: developer1  
Resource type: ResDash  
Nature: ToBeDefined  
Description: Resource Dashboard: ...  
★★★★★  
View Edit Publish Owner

TOP

# search vs services, the **ServiceURI**





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

The screenshot shows the Snap4City IoT Directory interface. On the left is a navigation sidebar with the 'IOT Devices' section selected. The main area displays a table of IoT devices with columns for Device Identifier, IOT Broker, Device Type, Model, Ownership, Status, Edit, Delete, Location, and View. A detailed view for 'AdminDevice001' is shown below the table, including metadata like Broker URI, Kind, Device Type, Protocol, Model, Longitude, Device URI, Organization, and Payloads. A green arrow points from the 'Device URI' field in the detailed view to a highlighted box at the bottom of the slide.

| Device Identifier             | IOT Broker                | Device Type       | Model                | Ownership    | Status | Edit | Delete | Location | View |
|-------------------------------|---------------------------|-------------------|----------------------|--------------|--------|------|--------|----------|------|
| 15EP22T2AA1S000022            | orionFirenze-UNIFI        | ChargingStation   | ChargingStationModel | PUBLIC       | active | EDIT | DELETE |          | VIEW |
| AccessPoint1_FamilaSuperstore | orionLonatoDelGarda-UNIFI | AccessPointSensor | AccessPointLonato    | DELEGATED    | active | EDIT | DELETE |          | VIEW |
| AccessPoint2_ITIS             | orionLonatoDelGarda-UNIFI | AccessPointSensor | AccessPointLonato    | DELEGATED    | active | EDIT | DELETE |          | VIEW |
| AccessPoint3_Palaspport       | orionLonatoDelGarda-UNIFI | AccessPointSensor | AccessPointLonato    | DELEGATED    | active | EDIT | DELETE |          | VIEW |
| adminDev1                     | orionUNIFI                | Ambiental         |                      | MYOWNPUBLIC  | active | EDIT | DELETE |          | VIEW |
| AdminDevice001                | orionUNIFI                | Ambiental         |                      | MYOWNPRIVATE | active | EDIT | DELETE |          | VIEW |

**AdminDevice001 Details:**

- Broker URI: https://broker1.snap4city.org
- Kind: sensor
- Device Type: Ambiental
- Protocol: ngsi
- Model:
- Longitude: 9.228193
- Device URI: http://www.disit.org/km4city/resource/iot/orionUNIFI/AdminDevice001
- Organization: DISIT
- PAYLOAD NGSi v1: K1: b7c4c115-f25c-4cb6-95eb-e4b363222bef
- PAYLOAD NGSi v2: K2: 441ffb6c-dc8a-4fc9-a415-7f6564d656f5
- Created on: 2018-05-24 21:54:03

Device Uri: <http://www.disit.org/km4city/resource/iot/orionUNIFI/AdminDevice001>

# Understanding / Testing an Entity/ IoT Device

|                                                                                                                                                               |                |                                         |           |                            |        |                                    |               |  |             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|-----------------------------------------|-----------|----------------------------|--------|------------------------------------|---------------|--|-------------|
|                                                                                                                                                               | AdminDevice001 | orionUNIFI                              | Ambiental | <b>MYOWNPRIVATE</b>        | active | <b>EDIT</b>                        | <b>DELETE</b> |  | <b>VIEW</b> |
| Broker URI: <a href="https://broker1.snap4city.org">https://broker1.snap4city.org</a>                                                                         |                | Kind: sensor                            |           | Broker Port: 8080          |        | Visibility: MyOwnPrivate           |               |  |             |
| Device Type: Ambiental                                                                                                                                        |                | Protocol: ngsi                          |           | Format: json               |        | MAC:                               |               |  |             |
| Model:                                                                                                                                                        |                | Longitude: 9.228193                     |           | Producer: Raspberry PI     |        | Latitude: 45.499369                |               |  |             |
| Device Uri: <a href="http://www.disit.org/km4city/source/iot/orionUNIFI/AdminDevice001">http://www.disit.org/km4city/source/iot/orionUNIFI/AdminDevice001</a> |                | Organization: DISIT                     |           | <b>VIEW IN SERVICE MAP</b> |        | <b>VIEW DATA IN AdminDevice001</b> |               |  |             |
| <b>PAYLOAD NGSI v1</b>                                                                                                                                        |                | <b>PAYLOAD NGSI v2</b>                  |           |                            |        |                                    |               |  |             |
| K1: b7c4115-f25c-4cb6-95eb-e4b363222bef                                                                                                                       |                | K2: 41ffb6c-dc8a-4fc9-a415-7f6564d656f5 |           |                            |        |                                    |               |  |             |
| Created: 2018-05-24 21:54:03                                                                                                                                  |                |                                         |           |                            |        |                                    |               |  |             |



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.



**Snap4City**

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

[LOGOUT](#)

- My Snap4City.org
- Dashboards
- My Dashboards in All Org.
- Dashboards of My Organization
- My Dashboards in My Organization
- Notifier
- Data Inspector**
- My Data, KPI, POI
- IOT Applications
- IOT Directory and Devices
- Knowledge and Maps
- Micro Applications
- External Services
- Data Set Manager: Data Gate
- Resource Manager: Process Loader
- Development Tools
- Management
- Settings
- User Management and Auditing
- Help and Contacts
- Documentation and Articles

**Data Inspector**

Single data widgets  
Multi data widgets

Map Controls:  
FilterMap GPSUser GPSOrg

**Data sources**

| Sensor | High-Level Type | Nature      |
|--------|-----------------|-------------|
| Sensor | Environment     | Environment |
| Sensor | Environment     | Environment |
| Sensor | Environment     | Environment |
| Sensor | Environment     | Environment |
| Sensor | Environment     | Environment |
| Sensor | Environment     | Environment |
| Sensor | Environment     | Environment |
| Sensor | Environment     | Environment |

**Data sources Details**

| Device                        | Values                                                                             | Healthiness | Process | Image | Licensing | User |
|-------------------------------|------------------------------------------------------------------------------------|-------------|---------|-------|-----------|------|
| GPS Coordinates:              | 42.642033, 18.1122                                                                 |             |         |       |           |      |
| High-Level Type:              | Sensor                                                                             |             |         |       |           |      |
| Nature:                       | From IOT Device to KB                                                              |             |         |       |           |      |
| Subnature:                    | IoTSensor                                                                          |             |         |       |           |      |
| Value Name:                   | Dubrovnik:orionDubrovnik-UNIFI/camera_Dubrovnik_1_Ploc                             |             |         |       |           |      |
| Device ServiceURI or Data ID: | http://www.disit.org/km4city/resource/iot/orionDubrovnik-UNIFI/Dubrovnik/camera_Du |             |         |       |           |      |
| Sensor ServiceURI or Data ID: | http://www.disit.org/km4city/resource/iot/orionDubrovnik-UNIFI/Dubrovnik/camera_Du |             |         |       |           |      |

Ownership: private

Organizations: Dubrovnik

[Link to Service Map](#) [Link to IoT Device](#)

**MAP4CITY**

- Click with the mouse on it

**Data Inspector Wizard**

Knowledge Base view

Device ServiceURI or Data ID: http://www.disit.org/km4city/resource/iot/orionDubrovnik-UNIFI/Dubrovnik/camera\_Du

Sensor ServiceURI or Data ID: http://www.disit.org/km4city/resource/iot/orionDubrovnik-UNIFI/Dubrovnik/camera\_Du

Datasource: IoT

Ownership: private

Organizations: Dubrovnik

[Link to Service Map](#) [Link to IoT Device](#)

**IOT Devices**

| IOT Device                  | IOT Boiler               | Device Type       | Model             | Ownership    | Status | Soft | Delete | Location |
|-----------------------------|--------------------------|-------------------|-------------------|--------------|--------|------|--------|----------|
| AccessPoint1_FamiasSupstore | orionLanatoDeCarde-UNIFI | AccessPointSensor | AccessPointLanato | DELEGATED    | active | EDIT | DELETE |          |
| AccessPoint2_ITIS           | orionLanatoDeCarde-UNIFI | AccessPointSensor | AccessPointLanato | DELEGATED    | active | EDIT | DELETE |          |
| AccessPoint3_DataSport      | orionLanatoDeCarde-UNIFI | AccessPointSensor | AccessPointLanato | DELEGATED    | active | EDIT | DELETE |          |
| adminDev1                   | orionUNIFI               | Ambiental         |                   | MYOWNPRIVATE | active | EDIT | DELETE |          |
| AdminDevice001              | orionUNIFI               | Ambiental         |                   | MYOWNPRIVATE | active | EDIT | DELETE |          |
| AdminDevice002              | orionUNIFI               | Ambiental         |                   | MYOWNPRIVATE | active | EDIT | DELETE |          |
| AdminDevice004              | orionUNIFI               | Ambiental         |                   | MYOWNPRIVATE | active | EDIT | DELETE |          |
| AdminDevice005              | orionUNIFI               | Ambiental         |                   | MYOWNPRIVATE | active | EDIT | DELETE |          |
| AdminTest005                | orionUNIFI               | Ambiental         |                   | MYOWNPRIVATE | active | EDIT | DELETE |          |

Showing 1 to 10 of 370 entries

Some functionalities are limited to certain roles

# Notation Terminology

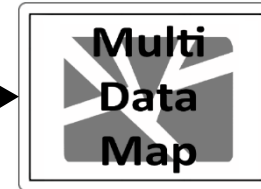
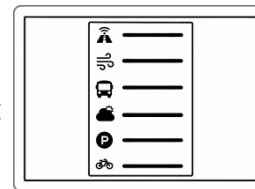
| WHERE                                             | Are synonymous at level of service which can be <b>IoT device or entity</b> with data and references to | Are synonymous at level of the single <b>attribute of the entity</b> , device, service, etc. |
|---------------------------------------------------|--------------------------------------------------------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

## ServiceURI, SURI of a sensor device:

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

## ServiceURI, SURI extended with attribute/variable/value:

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



# Dashboard Usage and recipe: Event map target

- Selector to Show on Map a

- category of Map positioned elements

- [https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.08694333811321;8.791809082031252;44.93168500391093;14.065246582031252&categories=Traffic\\_sensor&maxResults=0&maxDists=0.1&text=&model=&value\\_type=&format=json](https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.08694333811321;8.791809082031252;44.93168500391093;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;13.560999435424806&model=SmartLightCapelon&format=json>

- Single Entity

- [https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/iot/orionFirenze2/Firenze/SH\\_20\\_1a01new&format=json&fromTime=3-day](https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/iot/orionFirenze2/Firenze/SH_20_1a01new&format=json&fromTime=3-day)

- Heatmap among many

- [https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=Florence\\_PM10](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](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>

Parts 2, 3 and 7

# Data Registration Flow at a Glance

**IOT Devices**

| Device Identifier           | IOT Broker               | Device Type       | Model                | Ownership     | Status | Location |
|-----------------------------|--------------------------|-------------------|----------------------|---------------|--------|----------|
| SEP22ZPARSO002              | orionfirenze-UNFI        | ChargingStation   | ChargingStationModel | PUBLIC        | active |          |
| AccessPoint_FamilaSupertore | orionlanatoDetCarda-UNFI | AccessPointSensor | AccessPointLanato    | DELEGATED     | active |          |
| AccessPoint_ITS             | orionlanatoDetCarda-UNFI | AccessPointSensor | AccessPointLanato    | DELEGATED     | active |          |
| AccessPoint_Thesport        | orionlanatoDetCarda-UNFI | AccessPointSensor | AccessPointLanato    | DELEGATED     | active |          |
| adminDev1                   | orionUNFI                | Ambiental         |                      | MOWNER/PUBLIC | active |          |
| AdminDevice01               | orionUNFI                | Ambiental         |                      | MOWNER/PUBLIC | active |          |

IOT Directory:  
Devices...  
Sensors...  
Actuators...

Knowledge Base,  
ServiceMap,  
SuperServiceMap  
SmartCity API,  
ASCAPI

**Service Map (Toscana)**

Actual Selection:  
Service: BIMETAIR Quality Sensor - BORGO SAN LORENZO

| Property    | Value             |
|-------------|-------------------|
| PM10_5      | 2.413104836888828 |
| PM2_5       | 19.23619727063823 |
| CO2         | 9.282635931149007 |
| NO2         | 381.80            |
| NO3         | 25.28674469587327 |
| SO2         | 128.399961394157  |
| Temperature | 18.10             |
| Humidity    | 73.50             |

**IoT Apps Proc.Logic**

IoT Apps  
Proc.Logic

**Data Inspector**

| High-Level Type | Nature                 | Subnature       | Value Type      | Data Type | Value Unit | Last Date           | Last Value | Healthiness | Last Check          | Ownership |
|-----------------|------------------------|-----------------|-----------------|-----------|------------|---------------------|------------|-------------|---------------------|-----------|
| url             | Environment            | Natura          | url             | GeoWFS    |            | 2020-04-03 09:58:59 | 0          |             | 2020-04-03 09:58:59 | public    |
| url             | Environment            | Natura          | Prosp           | GeoWFS    |            | 2020-04-03 09:58:59 | 0          |             | 2020-04-03 09:58:59 | public    |
| url             | Environment            | Natura          | www_bioscote    | GeoWFS    |            | 2020-04-03 09:58:59 | 0          |             | 2020-04-03 09:58:59 | public    |
| url             | Environment            | Natura          | www_campochiese | GeoWFS    |            | 2020-04-03 09:58:59 | 0          |             | 2020-04-03 09:58:59 | public    |
| url             | Environment            | Natura          | fauna           | GeoWFS    |            | 2020-04-03 09:58:59 | 0          |             | 2020-04-03 09:58:59 | public    |
| url             | Mobility and Transport | Traffic Sensors | Traffic Sensors | GeoWFS    |            | 2020-04-03 09:58:59 | 0          |             | 2020-04-03 09:58:59 | public    |

Last Value: 15.9

Time Thread: 3 Apr 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00

DataInspector  
Dashboard Wizard

create  
Dashboards

TOP

# MyKPI Nodes

Part 3

Part 5

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



- 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 change 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

▼ S4CKPIData

get my kpdata

get my kpdata values

get public kpdata values

get delegated kpdata values

save my kpdata values



TOP

# ***TOOLS for Data Ingestion Verifications***

Part 3

Part 5



# Checking data/Entity ingestion results

## Knowledge base Semantic reasoners

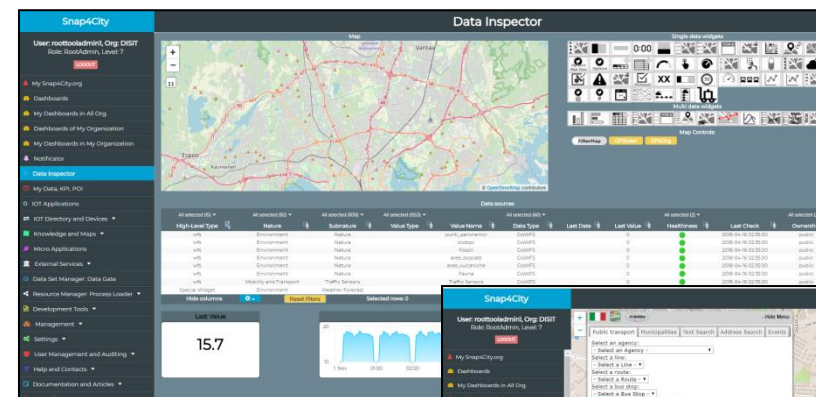


- All searches
- Metata
- Structure
- Last values of IoT Dev
- GTFS
- Only public IoT Dev

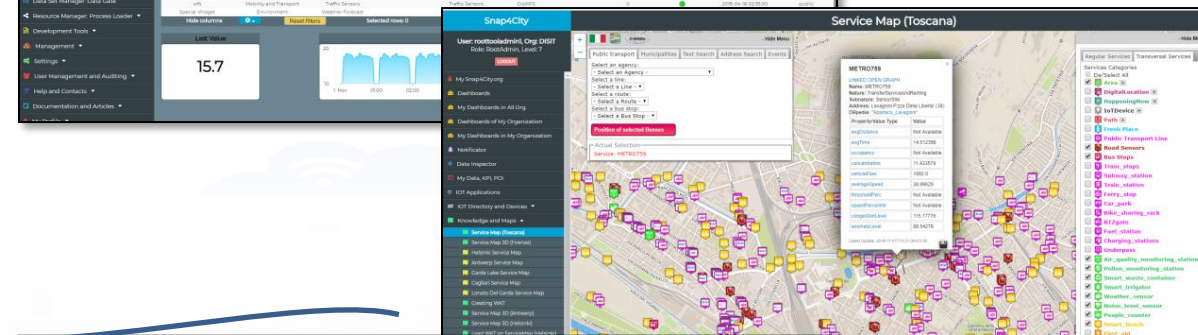
## Indexing and aggregating NIFI, OpenSearch

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

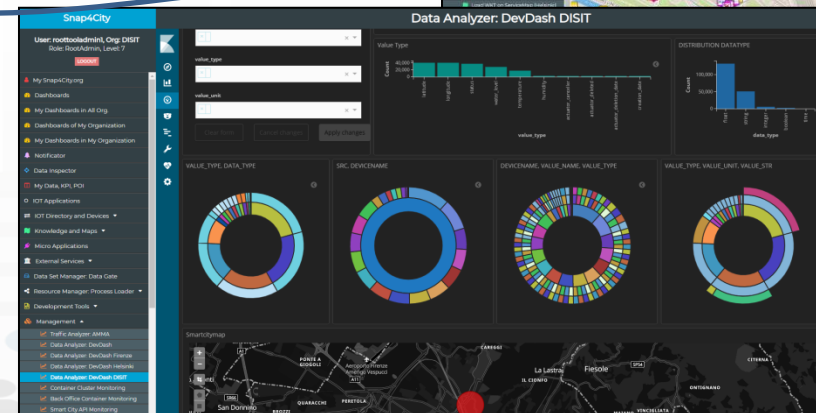
- **ServiceMap, SCAPI**
  - LOG / LOD viewer
  - Super Service Map
  - SCAPI: Swagger
  - Last data
- **Data Inspector (last data)**
- **IoT/Entity Directory**
  - IoT Brokers
- **ServiceMap, SCAPI (last data)**
- **My Data Dashboard, OpenSearchDash**
- **Data Inspector (last data)**



Data Inspector  
Digital Twin view



ServiceMap or  
Super ServiceMap



My Data Dashboard  
DevDash



Snap4City

Switch To New Layout (Beta)

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

LOGOUT

[www.snap4solutions.org](http://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

## Data Inspector

Map

**METRO729**

VALUE NAME: METRO729

|                 | DESCRIPTION              | DESCRIPTION | RT DATA    |              |               |             |
|-----------------|--------------------------|-------------|------------|--------------|---------------|-------------|
| avgTime         | 11.11040                 |             | 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.

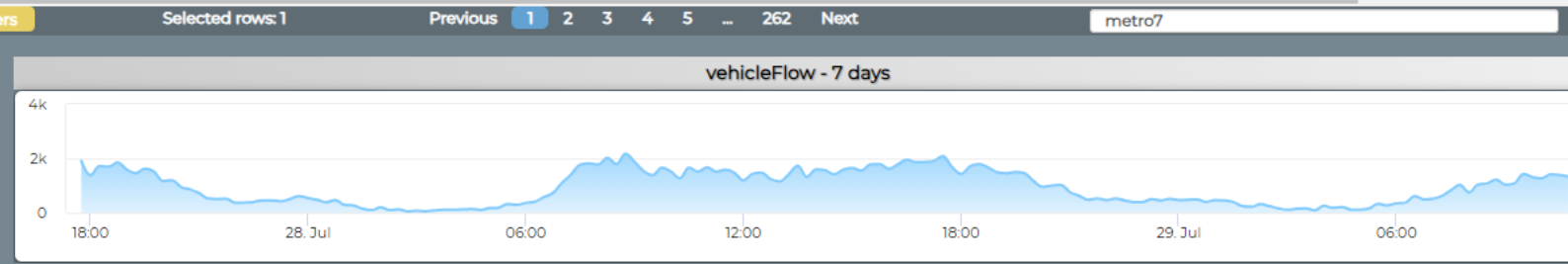
Switch now to the Synoptic Mode

Data sources

| High-Level Type | Nature                    | Subnature      | Device/Mode | 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 |            | <span style="color: green;">●</span> | 2023-07 |
| IoT Device      | TransferServiceAndRenting | Traffic_sensor | METRO791    | orionUNIFI |            |            | sensor_map |            | 2023-07-28 13:26:00 |            | <span style="color: green;">●</span> | 2023-07 |
| IoT Device      | TransferServiceAndRenting | Traffic_sensor | METRO793    | orionUNIFI |            |            | sensor_map |            | 2023-07-28 13:16:00 |            | <span style="color: green;">●</span> | 2023-07 |
| IoT Device      | TransferServiceAndRenting | Traffic_sensor | METRO713    | orionUNIFI |            |            | sensor_map |            | 2023-07-28 13:16:00 |            | <span style="color: green;">●</span> | 2023-07 |
| IoT Device      | TransferServiceAndRenting | Traffic_sensor | METRO729    | orionUNIFI |            |            | sensor_map |            | 2023-07-28 13:16:00 |            | <span style="color: green;">●</span> | 2023-07 |
| IoT Device      | TransferServiceAndRenting | Traffic_sensor | METRO7      | orionUNIFI |            |            | sensor_map |            | 2023-07-28 13:16:00 |            | <span style="color: green;">●</span> | 2023-07 |
| IoT Device      | TransferServiceAndRenting | Traffic_sensor | METRO760    | orionUNIFI |            |            | sensor_map |            | 2023-07-28 13:16:00 |            | <span style="color: green;">●</span> | 2023-07 |
| IoT Device      | TransferServiceAndRenting | Traffic_sensor | METRO799    | orionUNIFI |            |            | sensor_map |            | 2023-07-28 13:16:00 |            | <span style="color: green;">●</span> | 2023-07 |

vehicleFlow

1356



# 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

**Service Map (Toscana)**

**Public transport** Municipalities Text Search Address Search Events

Select an agency:  
- Select an Agency -

Select a line:  
- Select a Line -

Select a route:  
- Select a Route -

Select a bus stop:  
- Select a Stop -

**Position of selected Busses**

Actual Selection:  
**Bus Stop: Salvemini**

**TPL STOP - Salvemini**

ATAF&LINEA

Serviceuri: [http://www.disit.org/km4city/resource/Bus\\_ataflinea\\_Stop\\_FM0612\\_500\\_GRAPH](http://www.disit.org/km4city/resource/Bus_ataflinea_Stop_FM0612_500_GRAPH)

Lines:  
14 23 C1 C2

Display 10 Bus per page Search:

| Time                | Line | Direction                  |
|---------------------|------|----------------------------|
| 09:31:00 2020-10-07 | C1   | Parterre                   |
| 09:33:00 2020-10-07 | 23   | T2 Guidoni                 |
| 09:36:00 2020-10-07 | 14   | Santa Maria Maggiore       |
| 09:38:00 2020-10-07 | C2   | Leopolda T1 Porta Al Prato |
| 09:39:00 2020-10-07 | 23   | T2 Guidoni                 |
| 09:40:00 2020-10-07 | C1   | Parterre                   |

Showing page 1 of 45

Real-time data currently not available

Regular Services Transversal Services

Services Categories

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

Filter:  
search text into service

Service providing value type:  
select value type

N. results: 100

Search Range: visible area

Search Area: select...

Search Results:  
**Services 100 of 60336 available**

# Views of the Knowledge Base

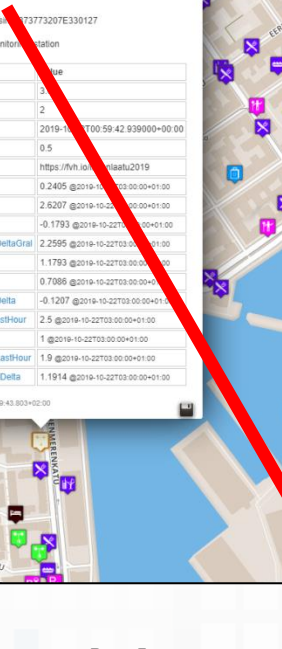
Knowledge Base  
Semantic Reasoners



# LOGraph

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

# SURI



**Helsinki KB Service Map**

Public transport Municipalities Text Search Address Search Events

Select an agency:  
- Select an Agency -  
Select a line:  
- Select a Line -  
Select a route:  
- Select a Route -  
Select a bus stop:  
- Select a Stop -

Position of selected Buses

Actual Selection  
Service: 373773207E330127

**LINKED OPEN GRAPH**  
Name: Iot/orion/Finland/Helsinki/373773207E330127  
Nature: Environment  
Subnature: Air\_quality\_monitoring\_station

| Property                        | Type     | Value                                  |
|---------------------------------|----------|----------------------------------------|
| PM10                            | float    | 3                                      |
| PM2.5                           | float    | 2                                      |
| dateObserved                    | dateTime | 2019-10-22T09:42:939000+00:00          |
| reliability                     | float    | 0.5                                    |
| source                          | string   | https://hki.io/iot/orion/aaahu2019     |
| airQualityPM10Gral              | float    | 0.2405 @2019-10-22T09:42:939000+00:00  |
| airQualityPM10Enfuser           | float    | 2.6207 @2019-10-22T09:42:939000+00:00  |
| RealTimeDeltaAQI                | float    | -0.1793 @2019-10-22T09:42:939000+00:00 |
| airQualityPM10RealTimeDeltaGral | float    | 2.2595 @2019-10-22T09:42:939000+00:00  |
| EnfuserAQI                      | float    | 1.1793 @2019-10-22T09:42:939000+00:00  |
| airQualityPM2_5Enfuser          | float    | 0.7086 @2019-10-22T09:42:939000+00:00  |
| airQualityPM10RealTimeDelta     | float    | -0.1207 @2019-10-22T09:42:939000+00:00 |
| airQualityPM10AverageLastHour   | float    | 2.5 @2019-10-22T09:42:939000+00:00     |
| RealTimeAQI                     | float    | 1 @2019-10-22T09:42:939000+00:00       |
| airQualityPM2_5AverageLastHour  | float    | 1.9 @2019-10-22T09:42:939000+00:00     |
| airQualityPM2_5RealTimeDelta    | float    | 1.1914 @2019-10-22T09:42:939000+00:00  |

**Linked Open Graph**

Select a SPARQL endpoint:  
Snap4City SmartCity Ontology (by DISIT)  
http://www.disit.org/km4city/resource/iot/orion/Finland/Helsinki/373773207E330127

Search for Keyword:  
373773207E330127

**Linked Open Graph**

Identifier:  
http://www.disit.org/km4city/resource/iot/orion/Finland/Helsinki/373773207E330127

Image:

Info:  
[http://www.w3.org/2003/01/geo/wgs84\\_pos#lat](http://www.w3.org/2003/01/geo/wgs84_pos#lat)  
60.15858  
[http://www.w3.org/2003/01/geo/wgs84\\_pos#long](http://www.w3.org/2003/01/geo/wgs84_pos#long)  
24.921349  
<http://schema.org/name>  
373773207E330127  
<http://www.disit.org/km4city/schema#format>  
json  
<http://www.disit.org/km4city/schema#macaddress>  
00:0C:29:00:00:00

**Network Graph:** A graph showing relationships between entities like 'AirQualityObject', 'Sensor', 'SystemCapability', and 'hasAttribute'.

## Linked Open Data

LOG: <https://log.disit.org>

LOD

**Linked Open Graph**

**SiiMobility (by DISIT)**  
Examples:  
• [VIA GIACOMO MATTEOTTI](#)  
• [Baiano a ripoli](#)  
• [Firenze](#)  
Choose a class:  
Search for keyword  
keyw  
uri

**Your data**  
sparql endpoint: (optional)  
http://...  
uri: http://... Request  
**Status**  
Requests:  
http://www.disit.dinfo.unifi.it/SiiMobility/MUSE  
Remove Clear

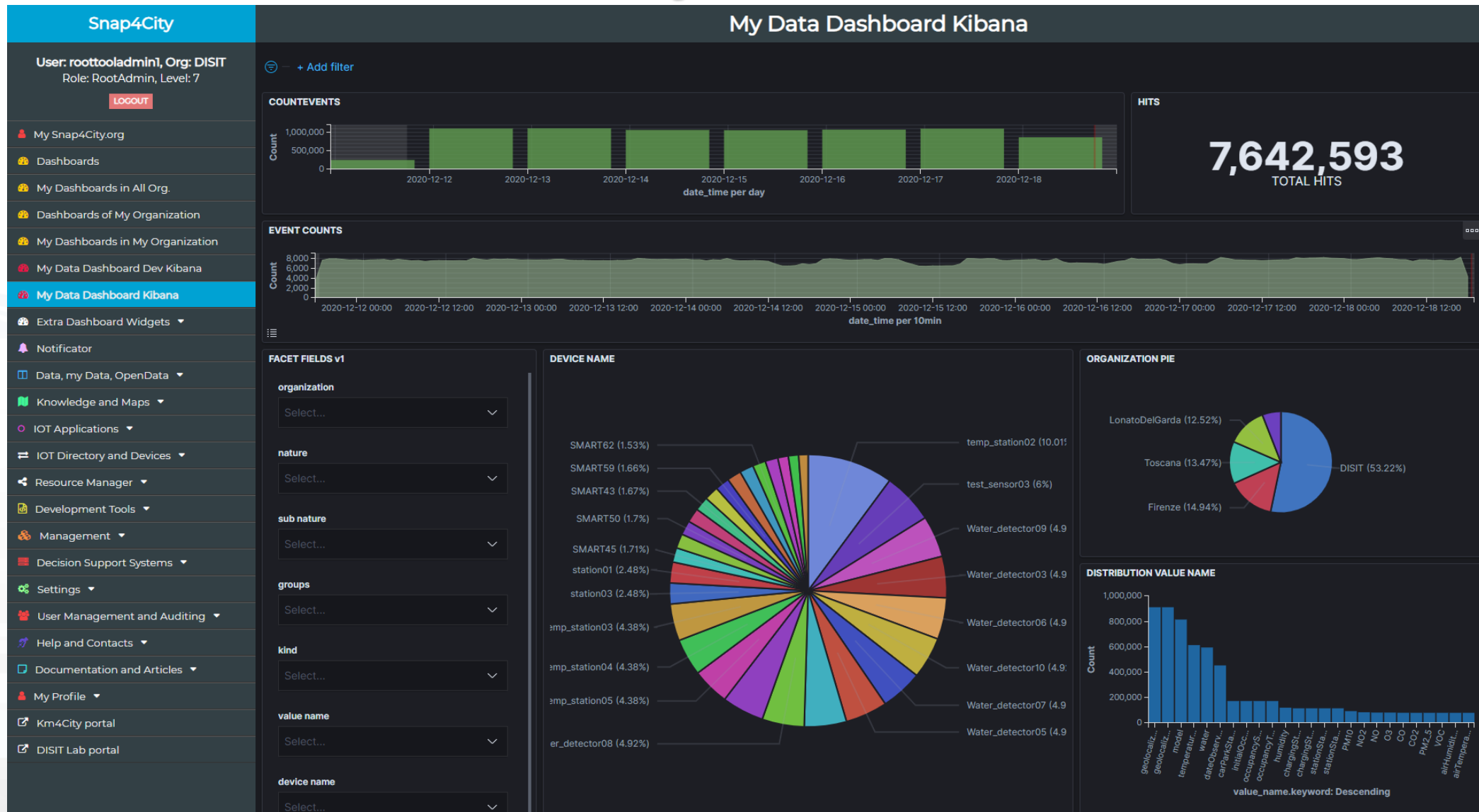
**Type of relations**  
Select all Deselect all Invert Hide all inverse  
 belongTo  
 contains  
 ends  
 has  
 hasExternalAccess  
 hasProvince  
 hasStreetNumber  
 isIn  
 isPartOfProvince  
 managingAuthority  
 placedIn  
 seeAlso  
 coincideWith  
 depiction  
 forming  
 hasAccess  
 hasMunicipality  
 hasRule  
 inMunicipalityOf  
 isPartOf  
 isPartOfRegion  
 ownerAuthority  
 sameAs  
 starts

**Linked Open Graph**

**museo ferragamo**  
DESCRIPTION  
Relations of Museo Ferragamo with the road graph

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

# DevDash: My Data Dashboard





TOP

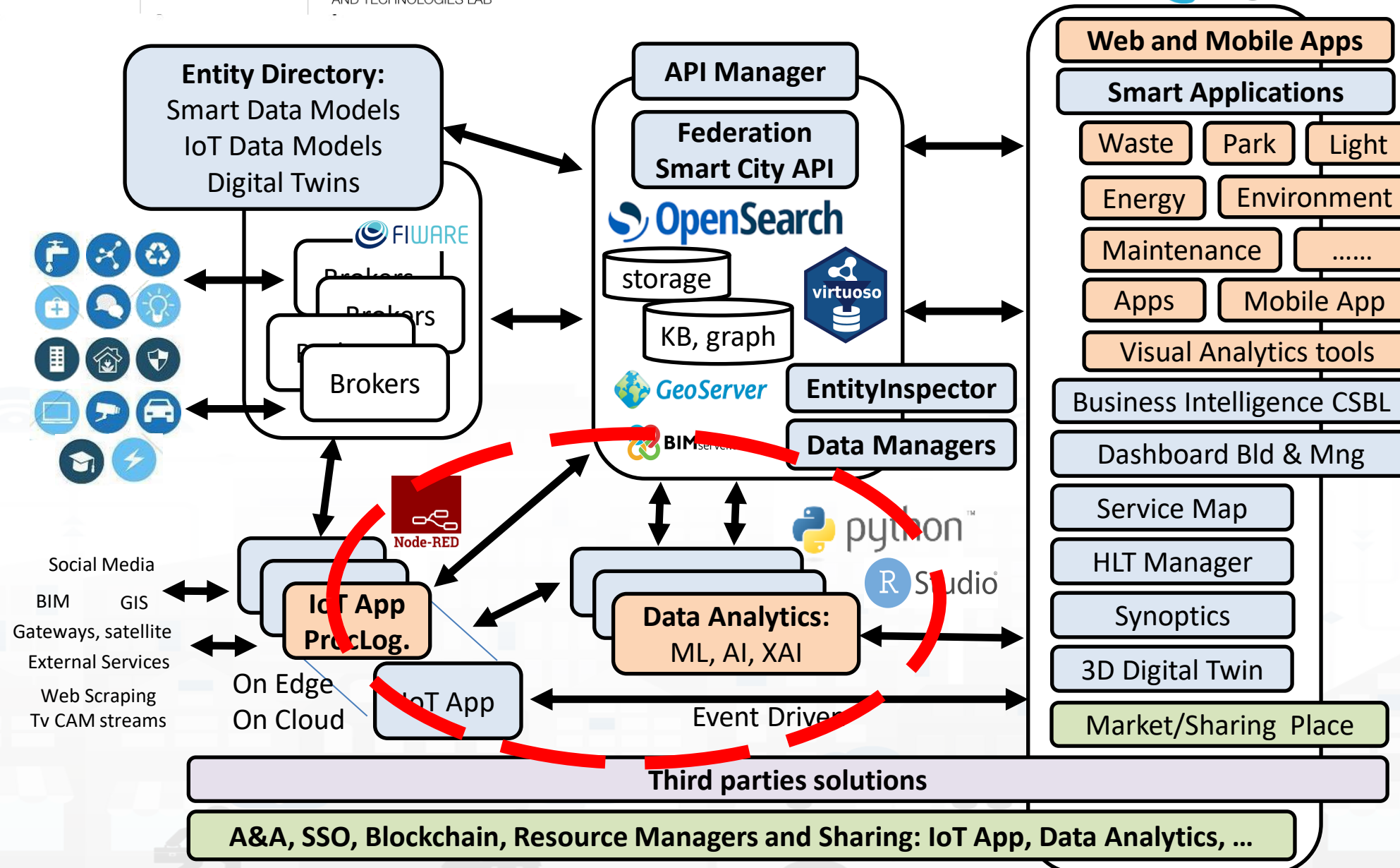
# Design & Develop of Data Analytics



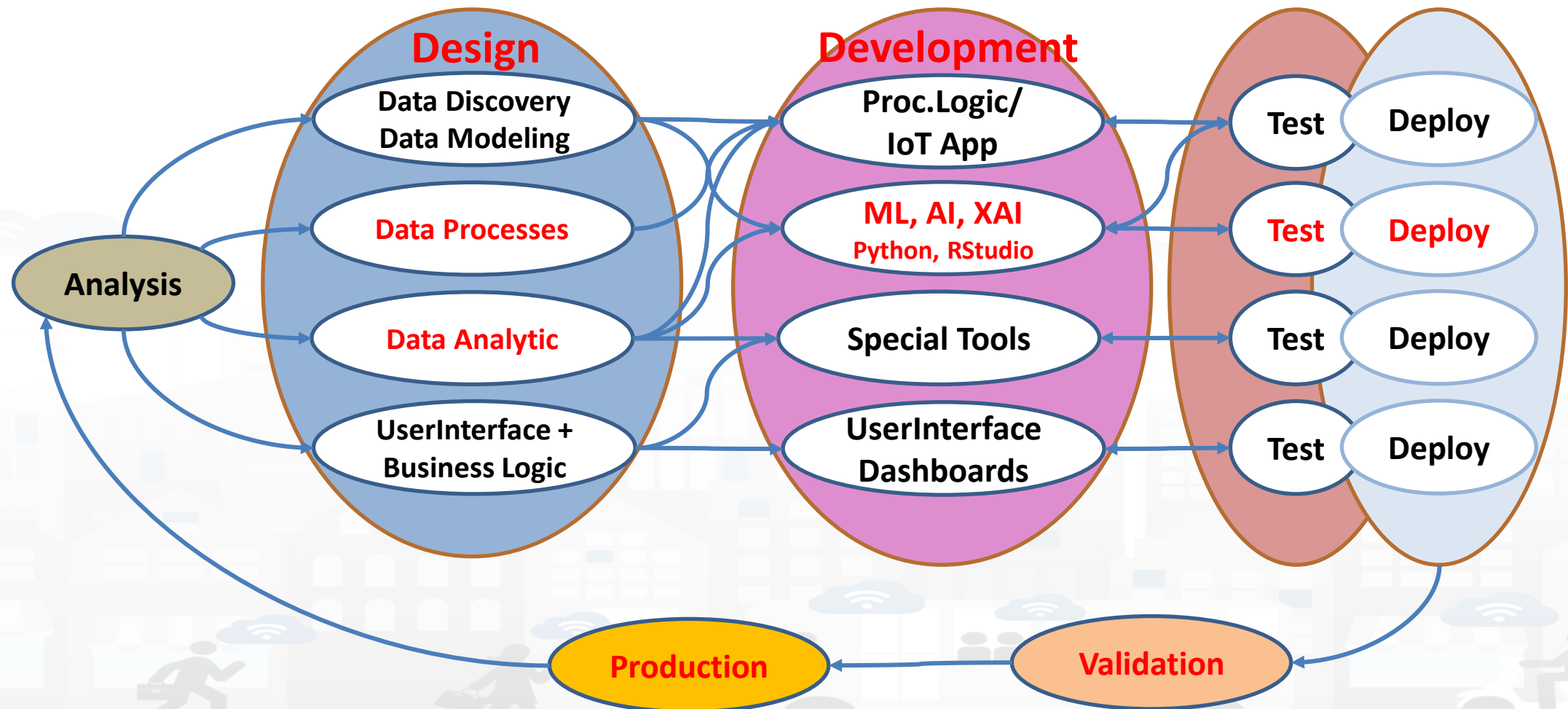
Part 4



# Tech Arch



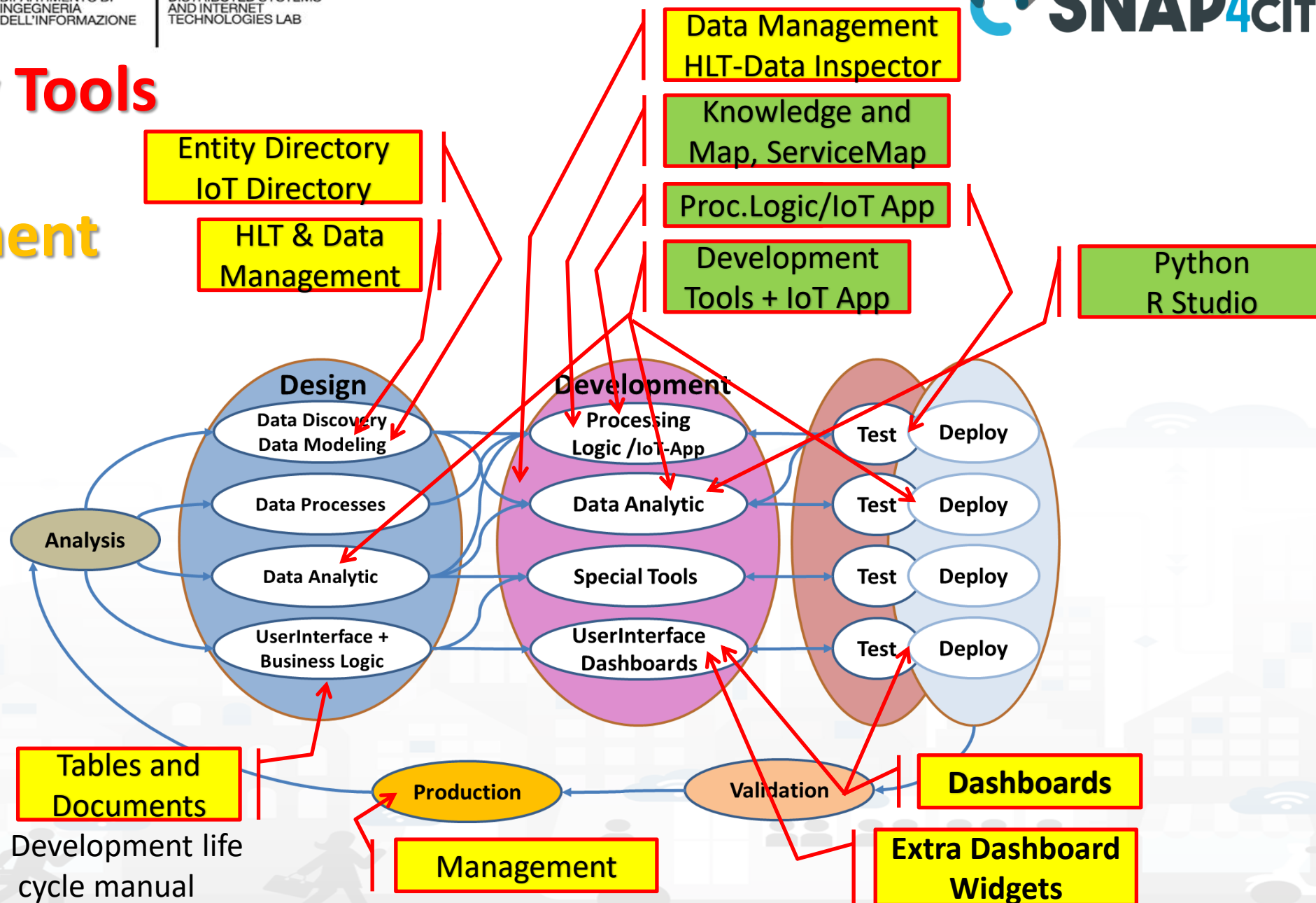
# Development Life Cycle Smart Solutions



# Snap4City Tools

vs

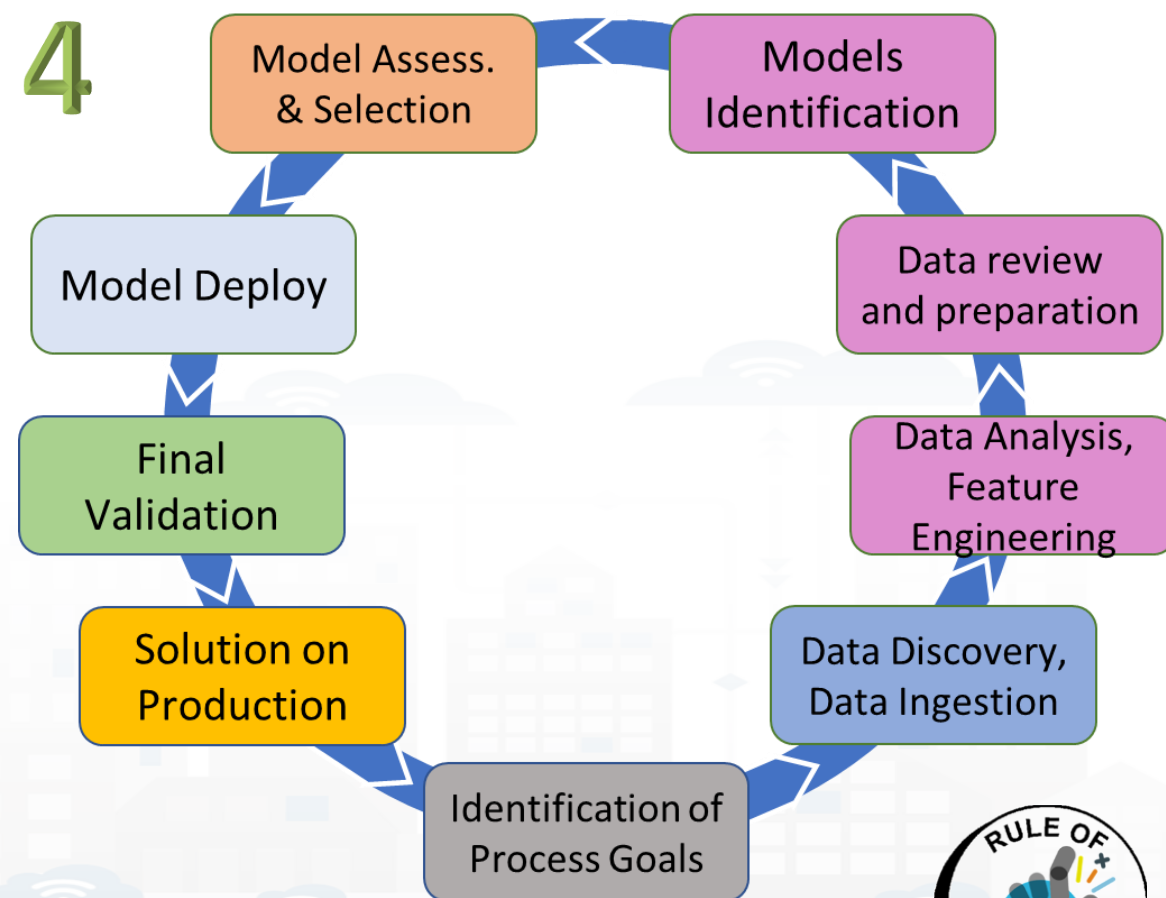
# Development Life Cycle



# Model/Technique Development/testing

- **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.)**

## Part 4



TOP

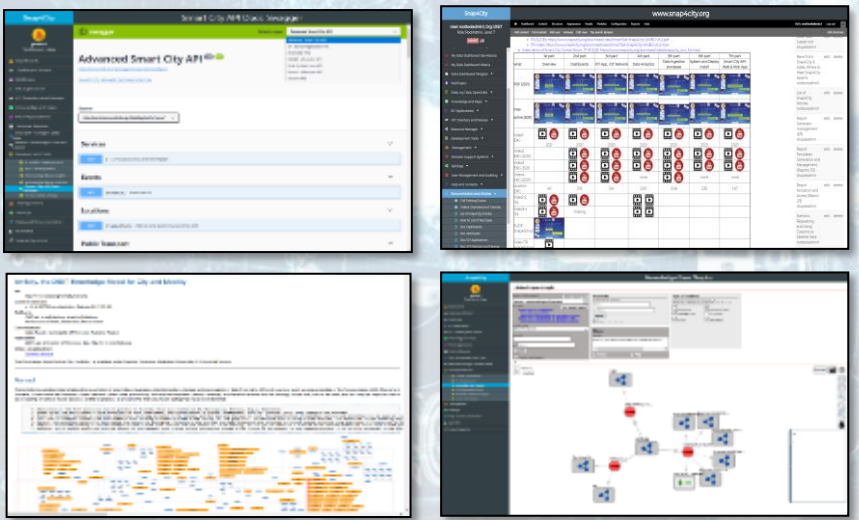
# *Data Analytics on Cloud: Snap4City Infrastructures*



# Data Analytics on Snap4City platform



Swagger

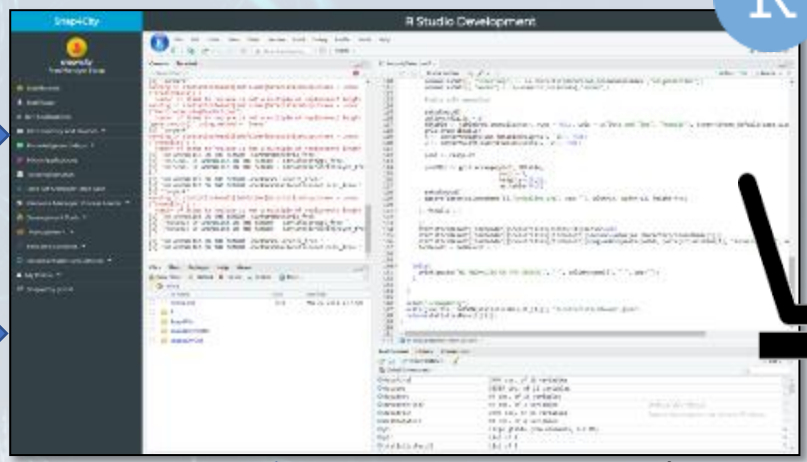


Ontology Schema

LOG.disit.org



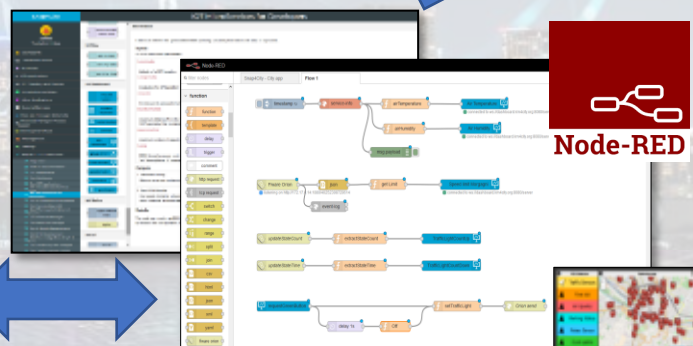
Smart City API from Knowledge Base and other tools



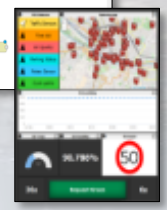
Creating MicroServices



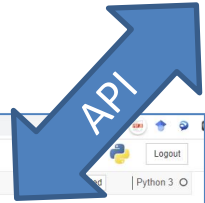
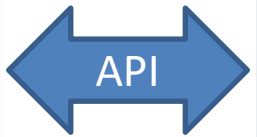
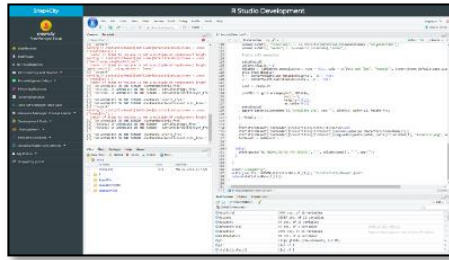
Saving / Sharing reusing



Using them into IOT Applications



# Development



On Server  
Or  
On PC

On PC as Anaconda

```

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

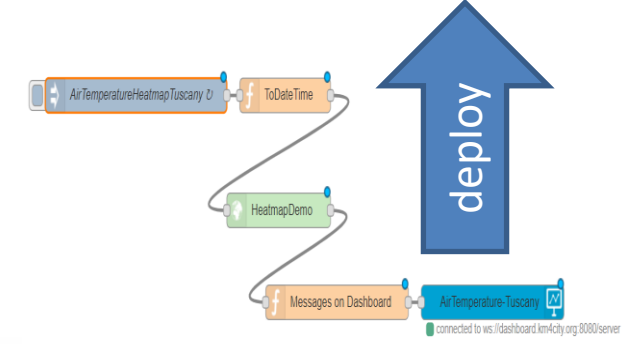
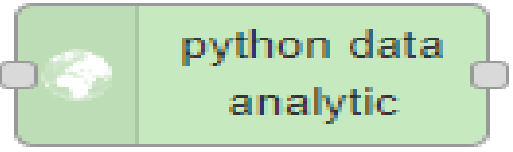
```



File.py  
AI Model  
Mapping  
Data..



Load  
File.py  
or .zip



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**

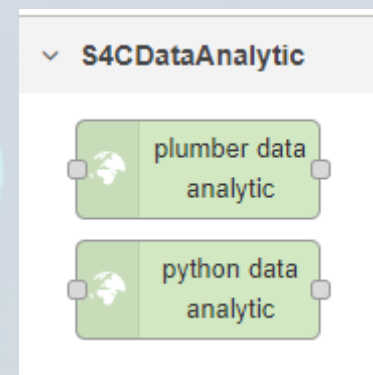
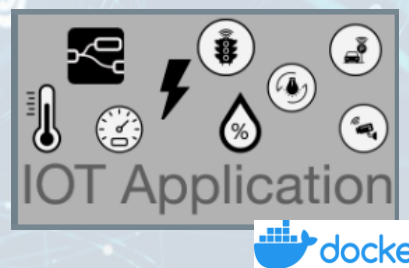




# Data Analytic Container



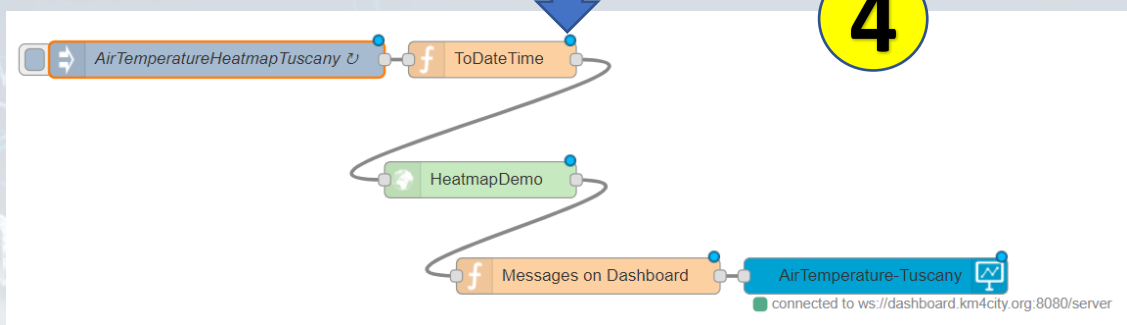
**2** Open an Advanced IoT App / Node-RED



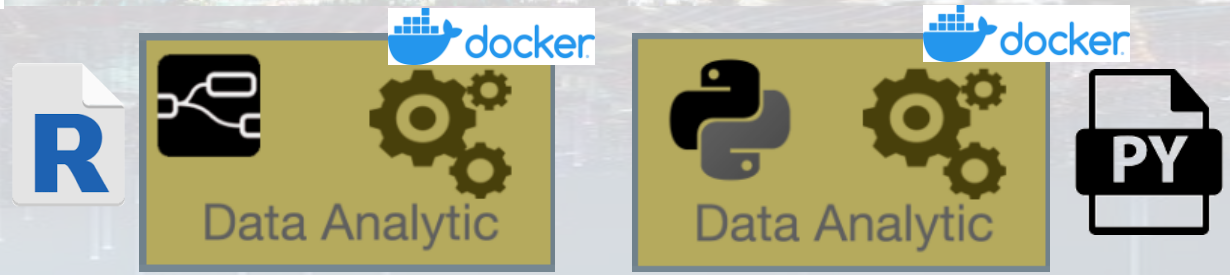
**3** Use Snap4City Data Analytic Node, and load in the code you developed

**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>

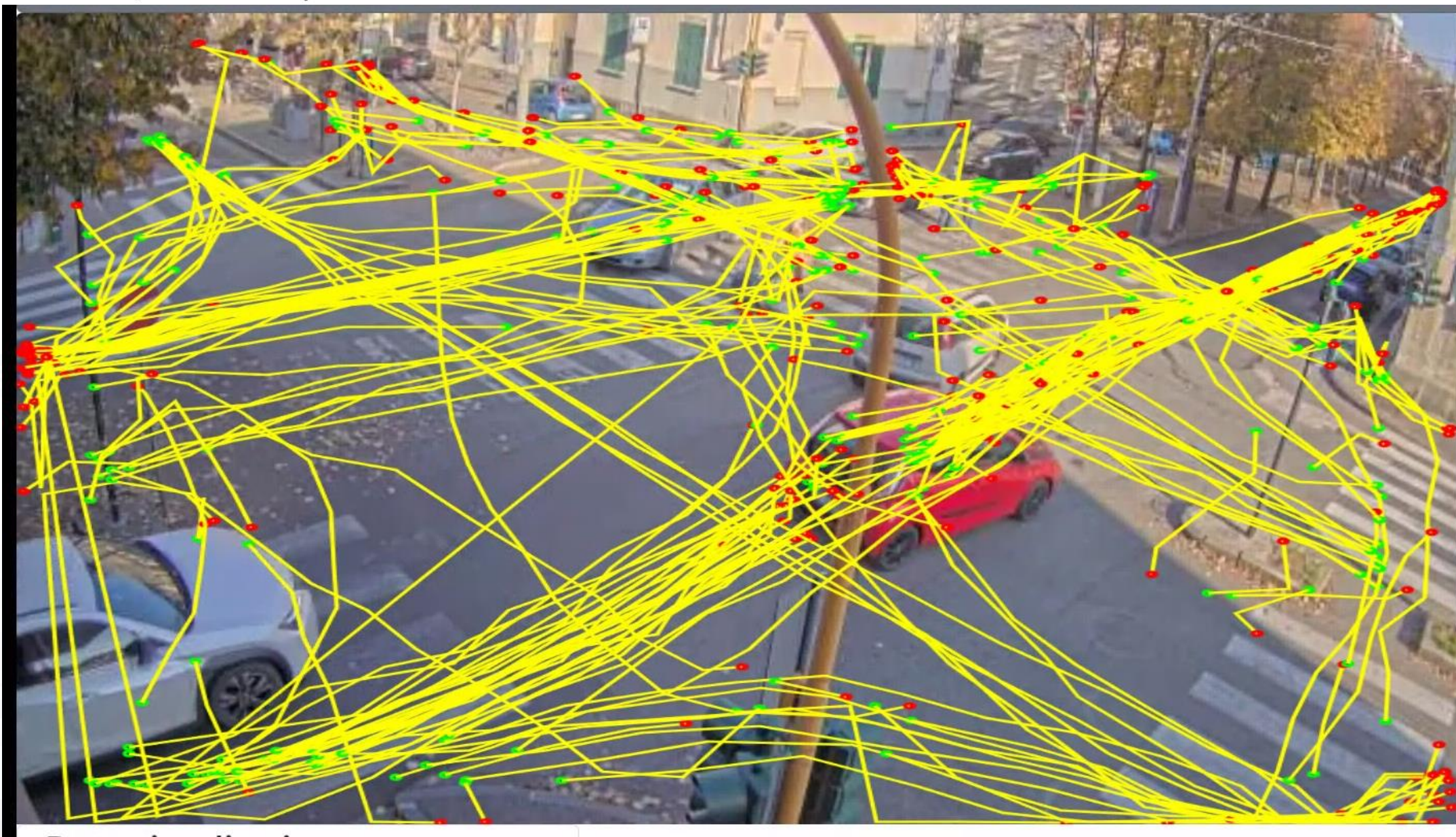


**5** Deploy the IoT App → Snap4City Container Manager based on Marathon/Mesos is creating a Container for your Data Analytic code



# *analytics example*







IoT edge on  
TV Camera

1

Send Trajectories

Send data to Broker

2

Device: CrossVenaria2  
with trajectories

IOT Broker

3

Save data

Big Data  
Store  
Facility

show data

4

Data Inspector

The Data Inspector interface displays a map of a city area with a red trajectory line. A data table is shown below the map, listing various data points for the device 'CROSSVENARIA2'. A time-series graph at the bottom shows the 'dist' variable over a 4-hour period, with a peak value of 15.9.

| Description | Value | Last Value | Last 4 hours | Last 24 hours | Last 7 days | Last 30 days | Last 90 days |
|-------------|-------|------------|--------------|---------------|-------------|--------------|--------------|
| dist        | 13.7  | Last Value | Last 4 hours | Last 24 hours | Last 7 days | Last 30 days | Last 90 days |
| en          | 308   | Last Value | Last 4 hours | Last 24 hours | Last 7 days | Last 30 days | Last 90 days |

IoT edge on  
TV Camera



Send Trajectories

Send data to Broker

**IOT Broker**

**Devices:**

- CrossVenaria2VehicleFlowTrajectoriesV2
- VenariaConteggio



**e**

Send data to Broker

**f**

Save data

Save Counting per Cluster

**Big Data Store Facility**

Get data

**c**

**Device:**  
CrossVenaria2  
with  
trajectories

Periodically

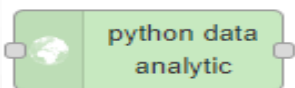
**b**

Activate



**d**

From Trajectories  
to clusters.  
Counting in/out  
and flows

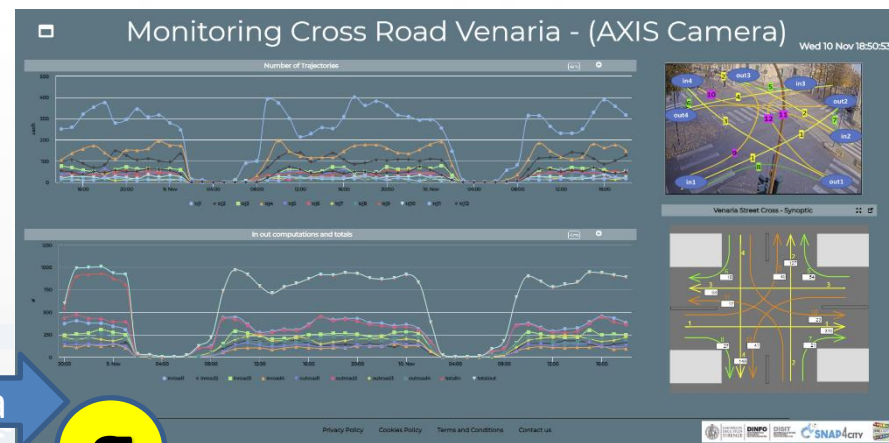


**a**

show data

**g**

Create and use a Dashboard

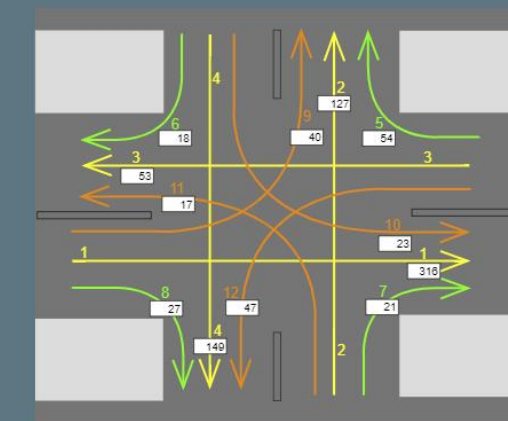
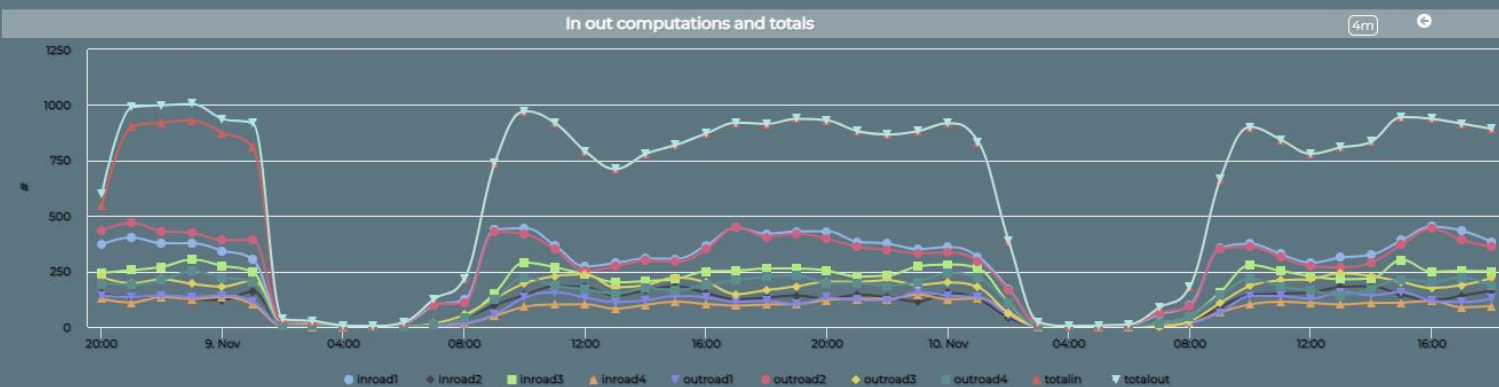
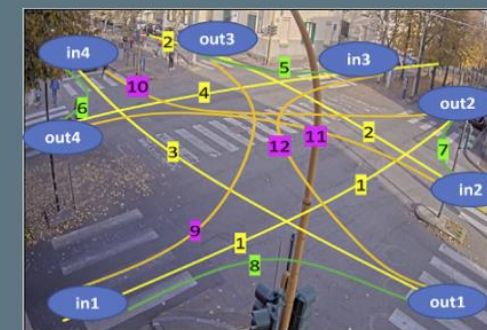
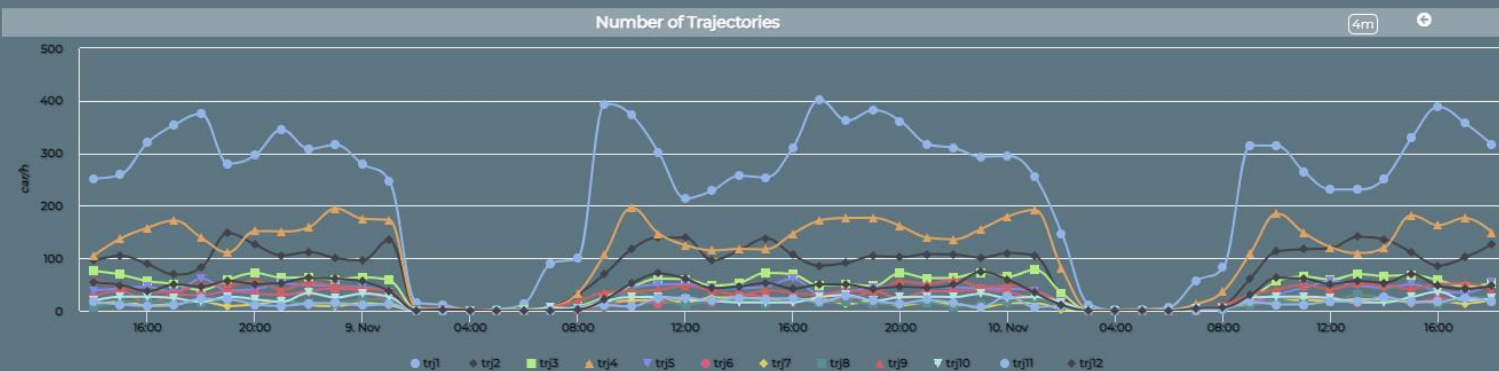


## Traffic Flow Analysis via TV Camera and Clustering on cloud

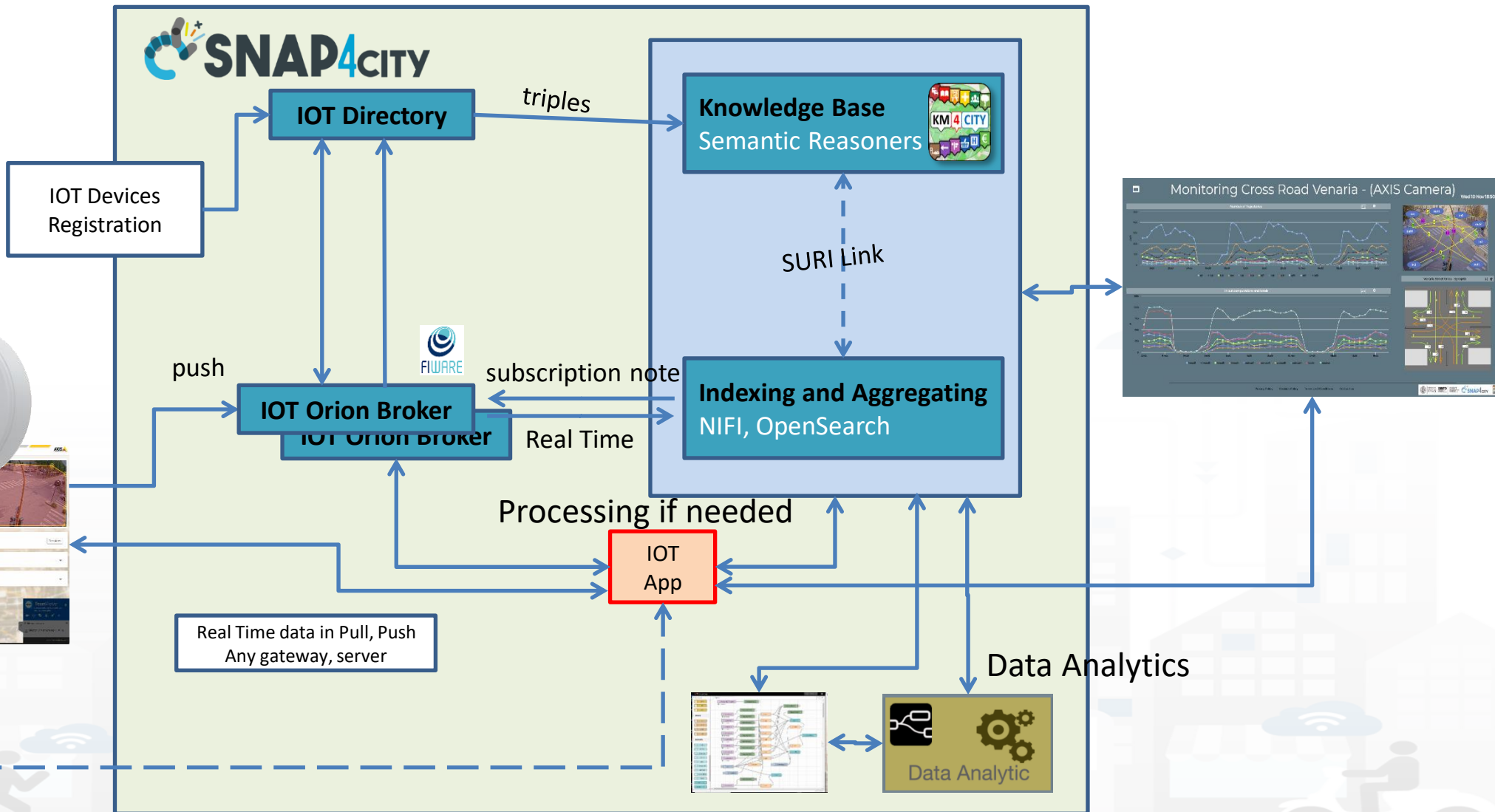


### Monitoring Cross Road Venaria - (AXIS Camera)

Wed 10 Nov 18:00



# Managing TV Cam



# *Data Analytics on Dedicated Computer or HPC*

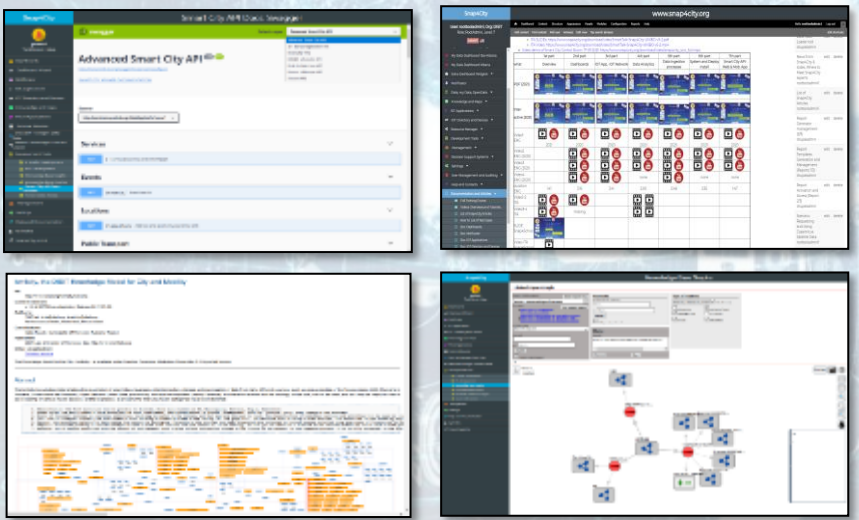




# Data Analytics on Snap4City platform



Swagger

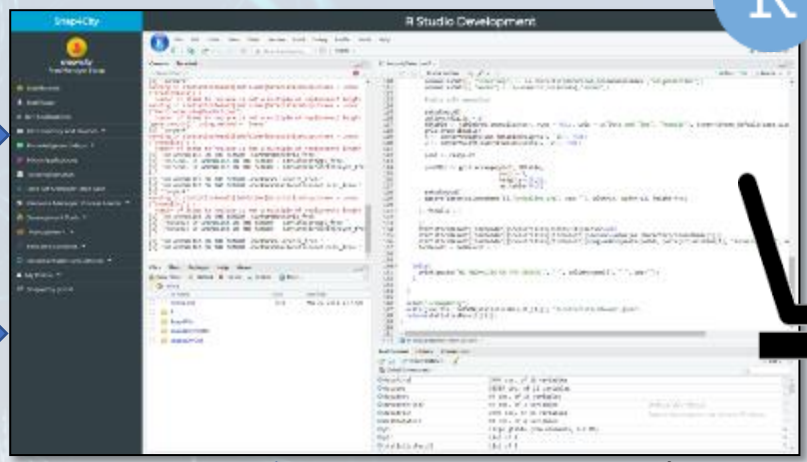


Ontology Schema

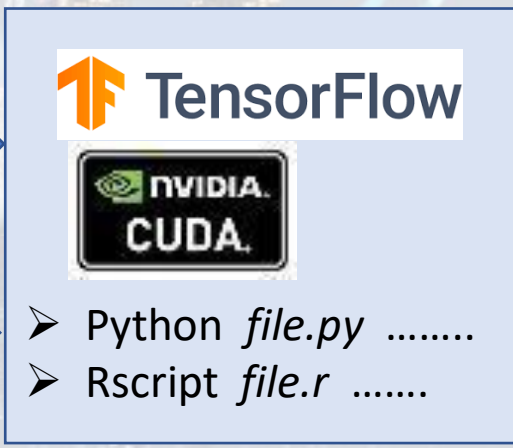
LOG.disit.org



Smart City API from Knowledge Base and other tools

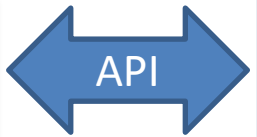
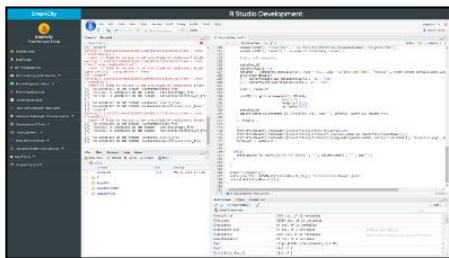


Saving / Sharing reusing



Resource Manager





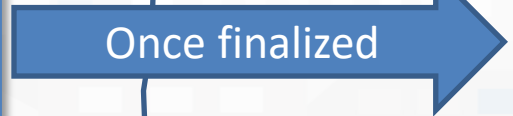
## EXECUTION



On Server  
Or  
On PC

On PC as Local Environment

DEVELOPMENT



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

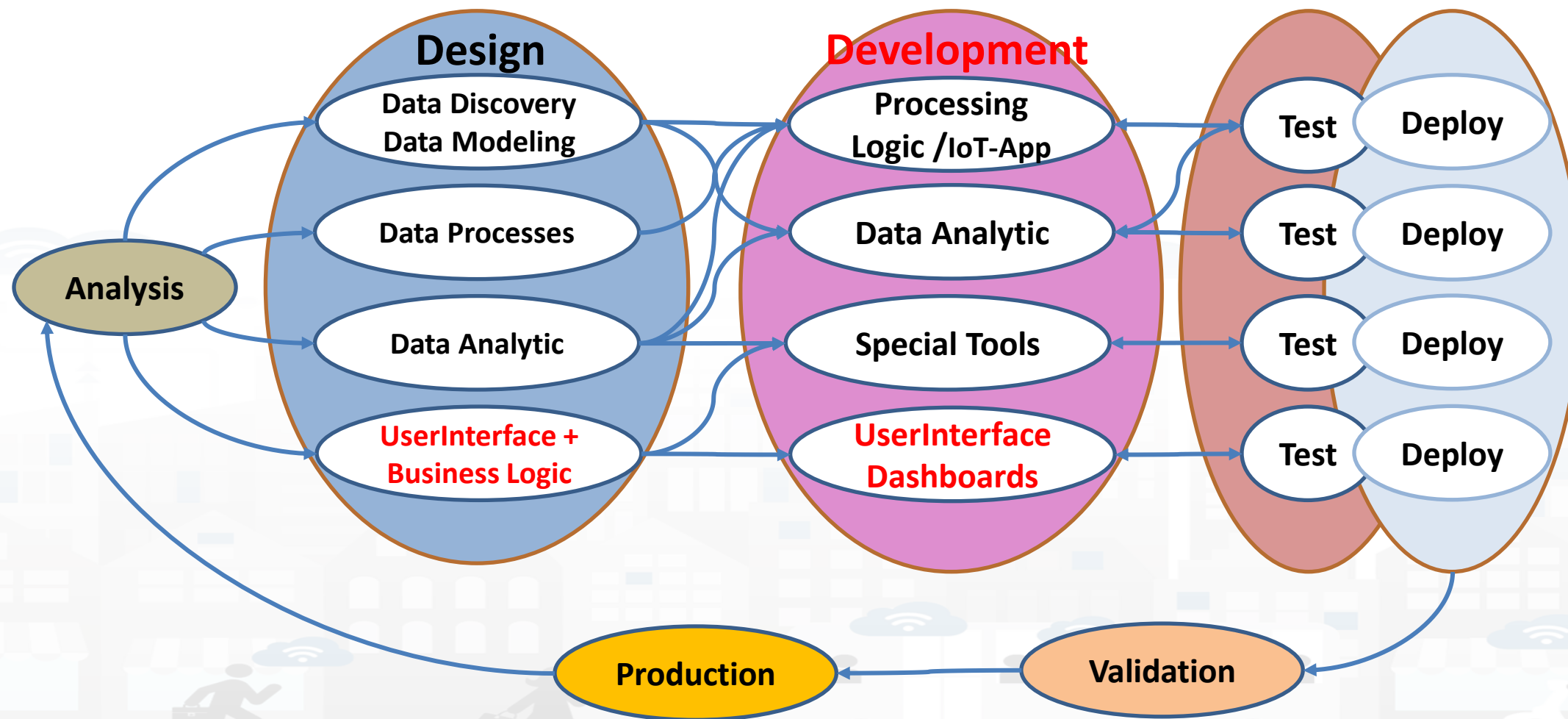
TOP

# Design & Develop: user interfaces, visual tools

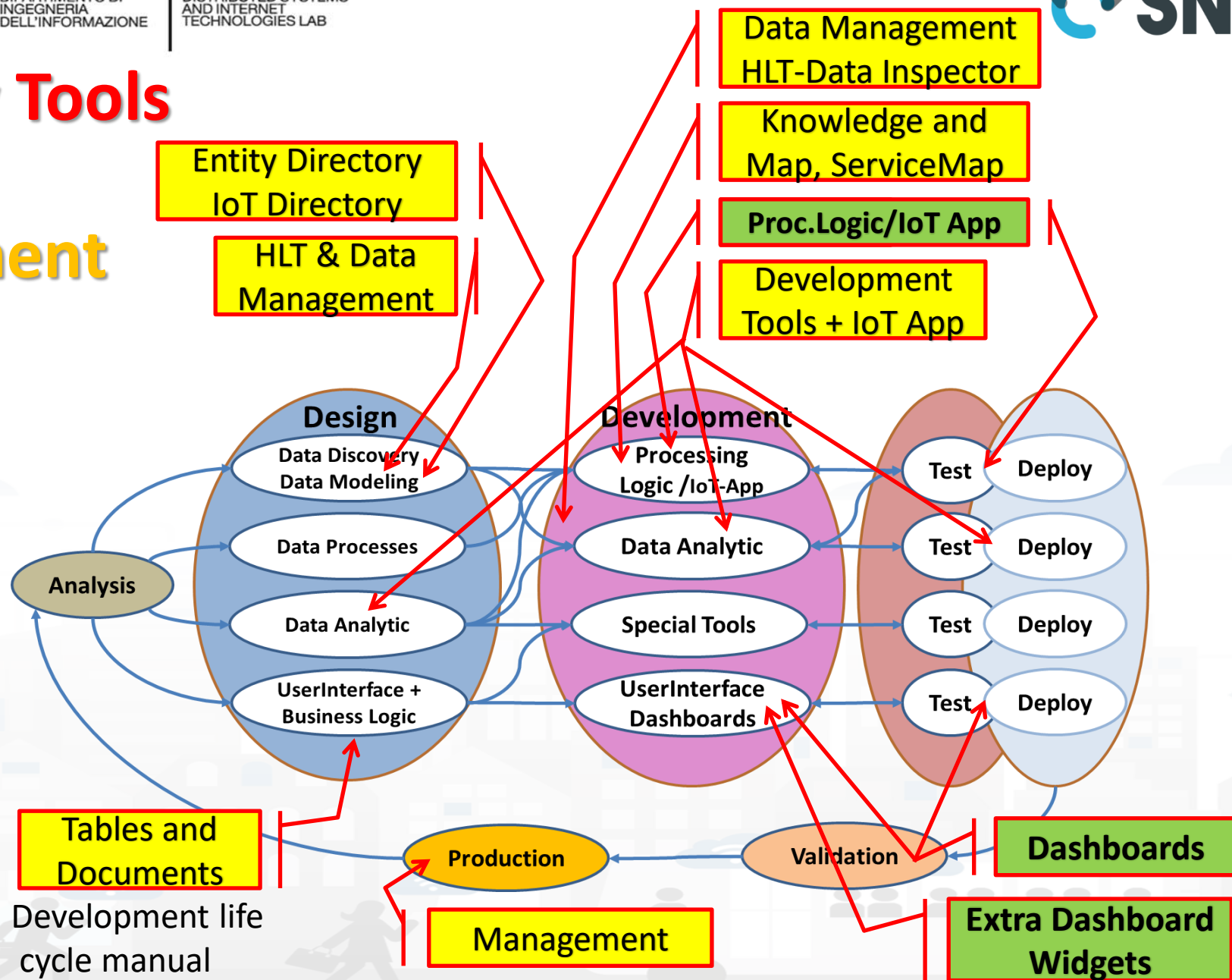
Part 2  
Part 3



# Development Life Cycle Smart Solutions

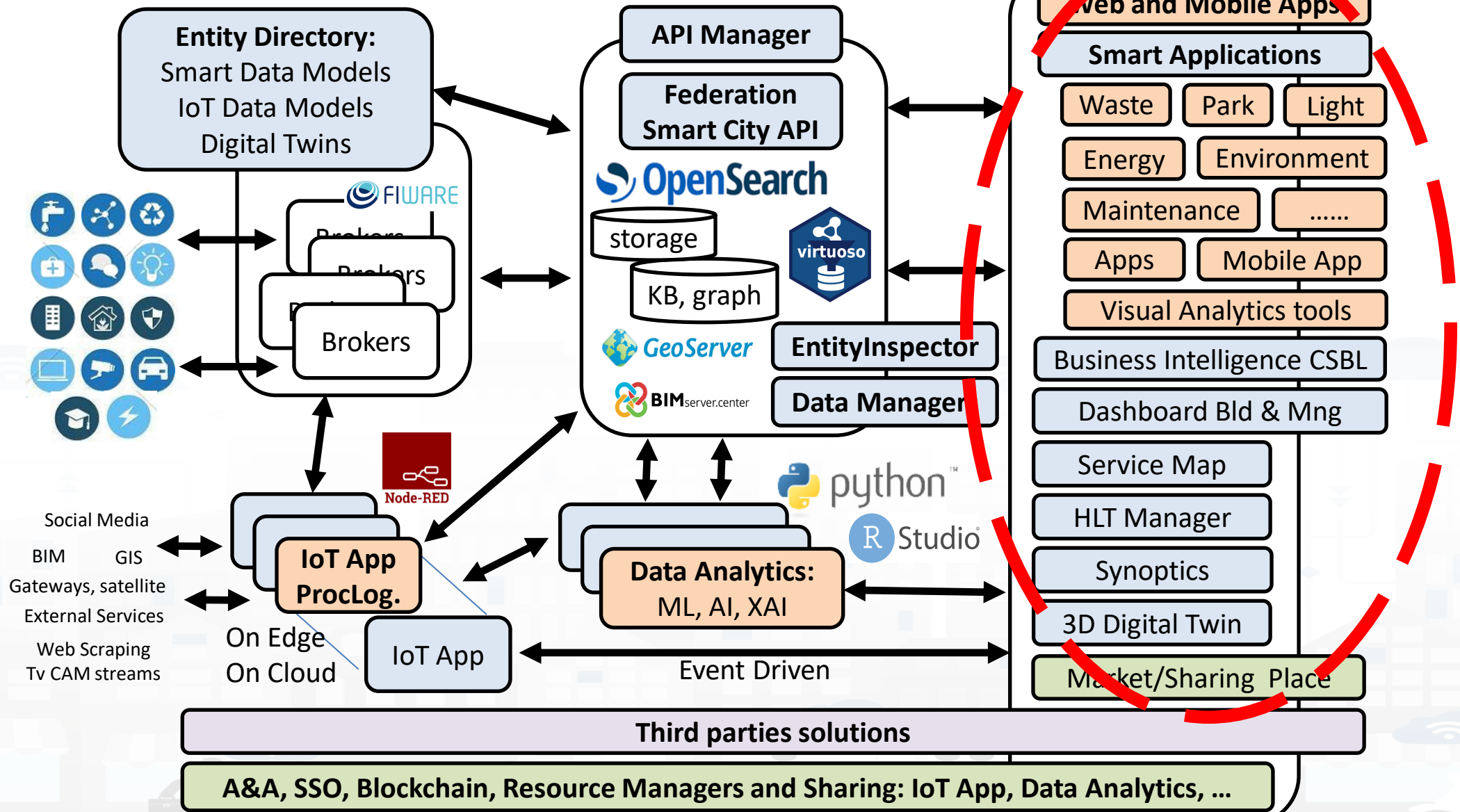


# Snap4City Tools vs Development Life Cycle



Development life cycle manual

# Tech Arch





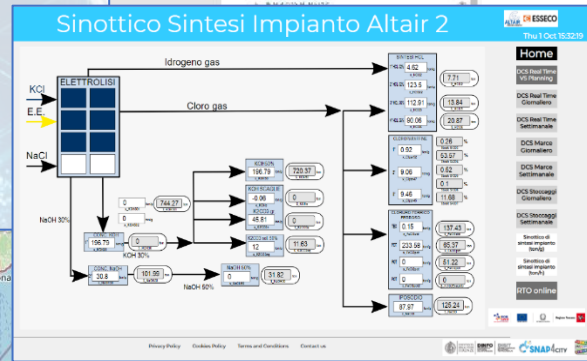
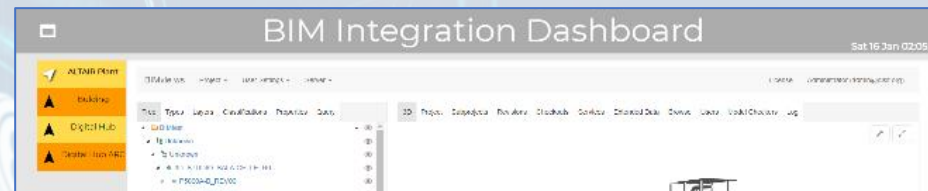
# Snap4City Dashboard Builder (2023) vs Kibana/Grafana

| 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             |

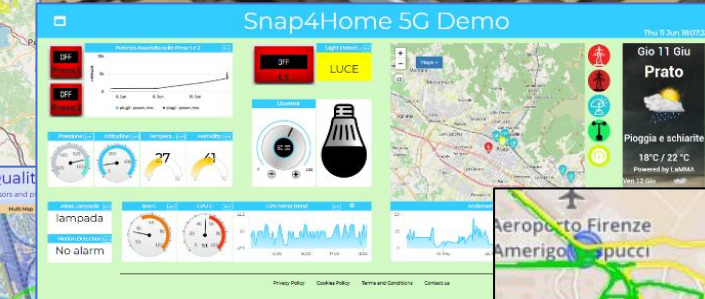
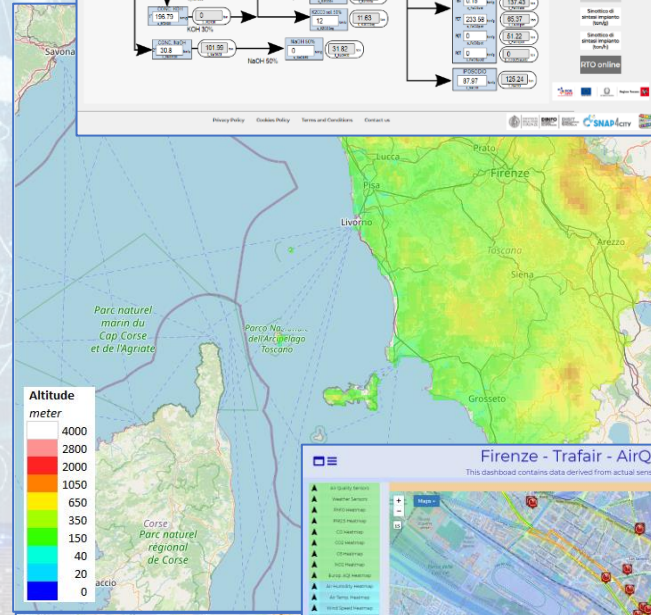
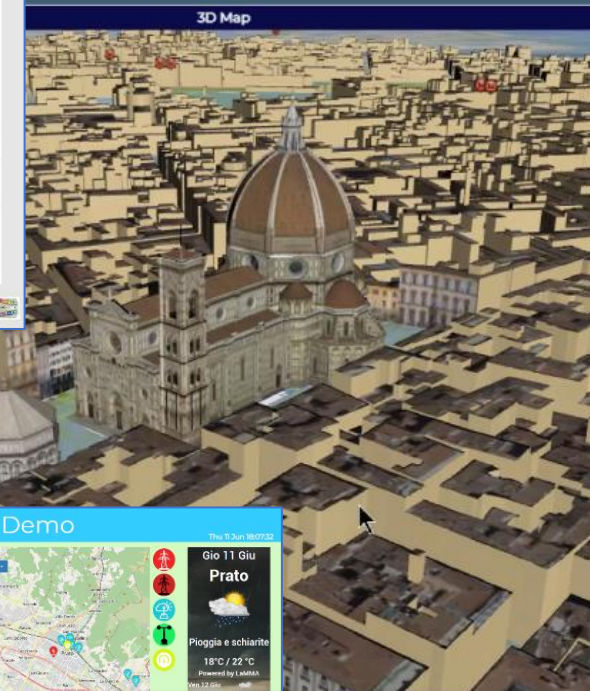
# High Level Types

Snap4City (C), September 2023

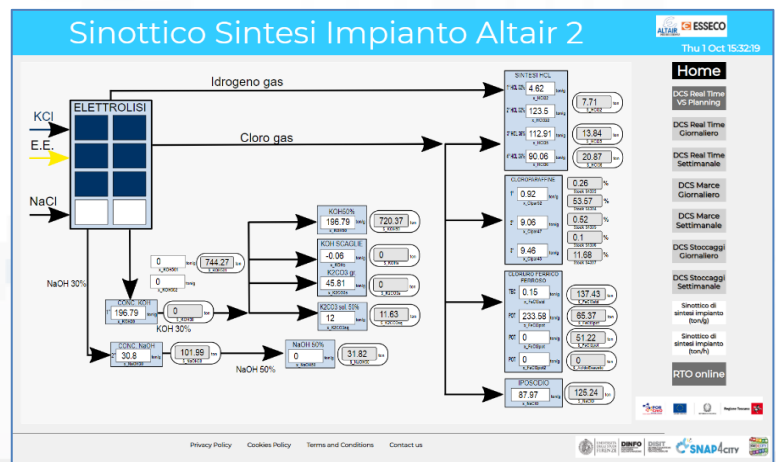
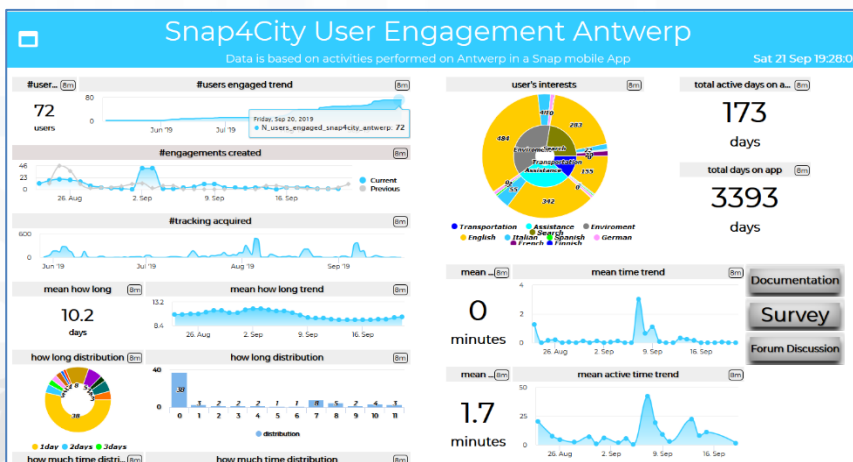
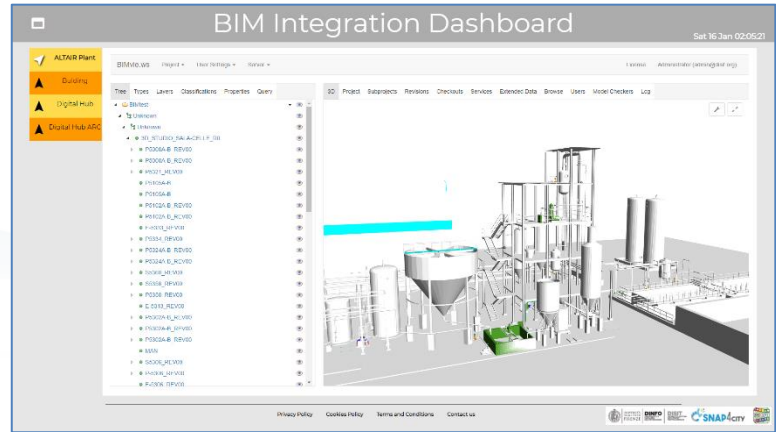
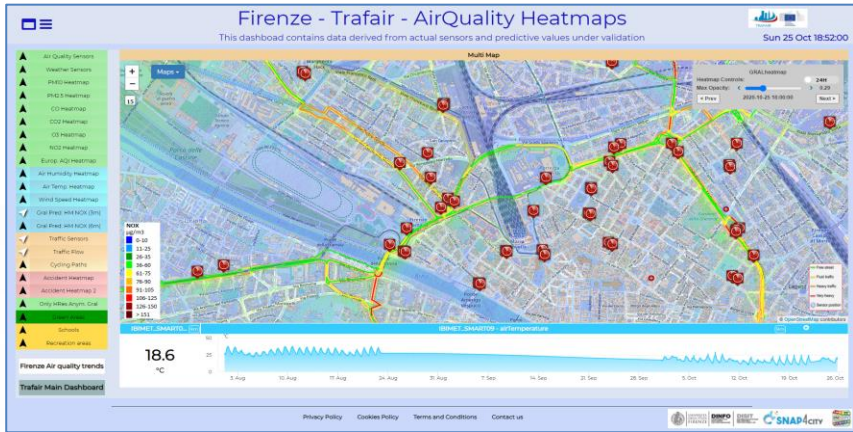
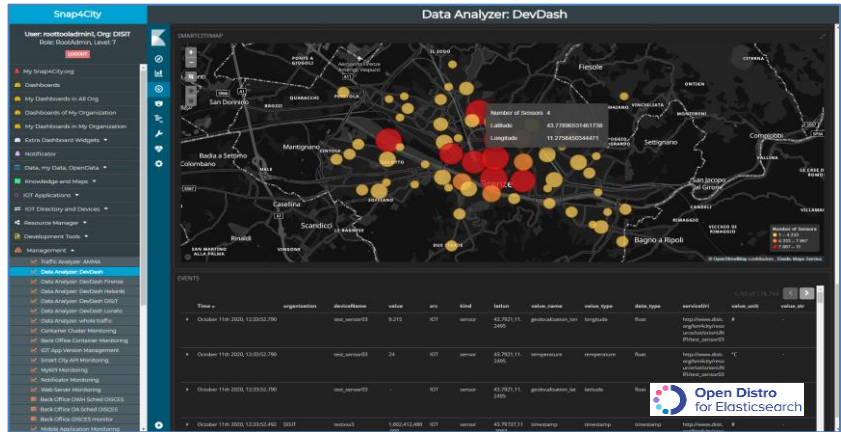
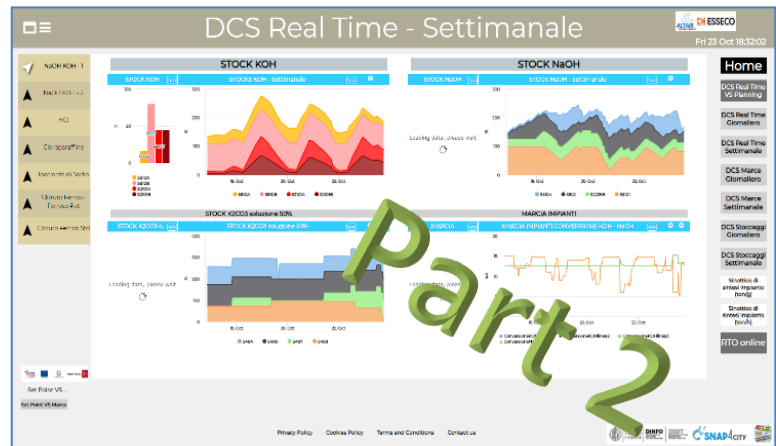
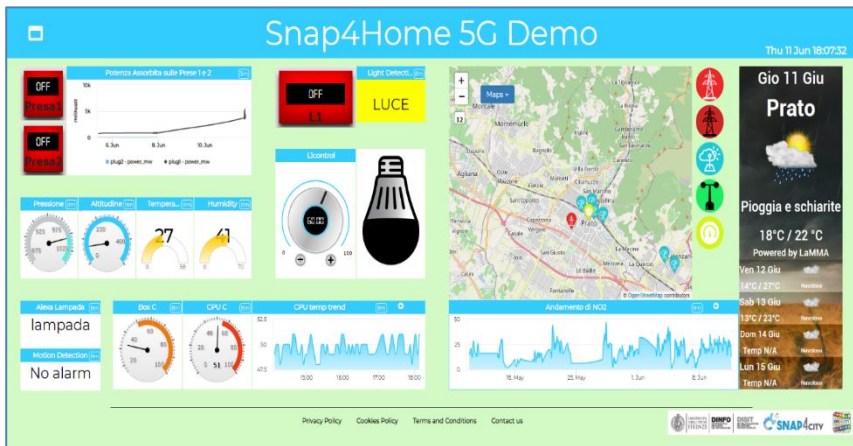
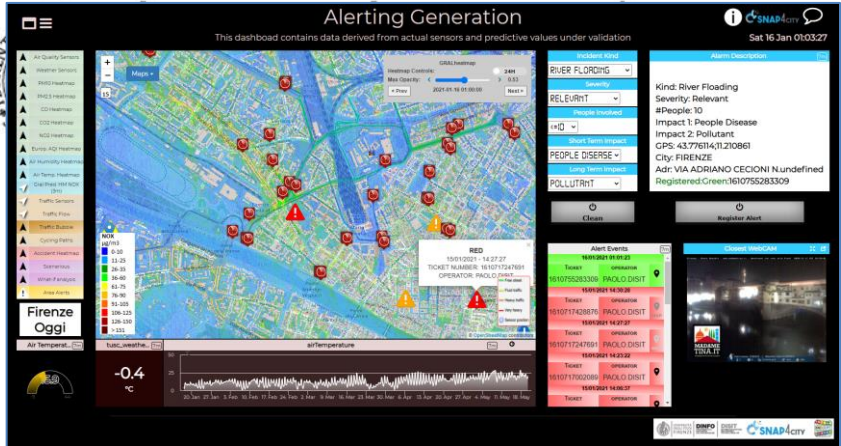
- POI, IOT Devices, shapes, ...
  - FIWARE Smart Data Models,
  - IoT Device Models
- GIS, maps, orthomaps, WFS/WMS, GeoTiff, calibrated heatmaps, ...
- Satellite data, ...
- traffic flow, typical trends, ...
- trajectories, events, Workflow, ...
- 3D Models, BIM, Digital Twins, ...
- OD Matrices of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



**SNAP4CITY**  
- Digital Twin Global - Fire  
demonstrator







Snap4City (C), September 2023

# Different Themes

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

**Legacy**

Multi Map  
GRAL Heatmap  
Heatmap Controls: Max Opacity: 0.25  
2025-10-25 18:00:00

IBIMET\_SMART09 - air/temperature  
18.6 °C

Firenze Air quality trends  
Traffair Main Dashboard

Privacy Policy Cookies Policy Terms and Conditions Contact us

**FIRENZE - TRAFAIR - AIRQUALITY HEATMAPS - NEWGUI**  
This dashboard contains data derived from actual sensors and predictive values under validation  
Tue 3 May 20:42:51

**BaloonDark Part 2**

Multi MAP  
GRAL Heatmap  
Heatmap Controls: Max Opacity: 0.25  
2022-04-27 21:00:00

IBIMET\_SMART09 - AIRTEMPERATURE  
20.6 °C

Traffair Main Dashboard

**3D MULTI DATA MAP - DIGITAL TWIN GLOBAL - FIRENZE - NEWGUI**  
Tue 3 May 14:31:42

**Baloon**

GRAL Heatmap  
Heatmap Controls: Max Opacity: 0.25  
2022-04-27 21:00:00

Firenze-IPIL/TrafficRealTime  
Traffic Heatmap Controls: Max Opacity: 1  
2022-05-03 14:21:00

METRO8 - VEHICLEFLOW  
240 car/h

**Traffic Flow Manager On Multiple Cities - Newgui**  
Wed 8 Jun 23:22:04

**Gea**

GRAL Heatmap  
Heatmap Controls: Max Opacity: 0.25  
2022-05-18 23:00:00

Firenze-IPIL/TrafficRealTime  
Traffic Heatmap Controls: Max Opacity: 1  
2022-06-08 22:20:00

METRO588 - AverageSpeed  
9m

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

Part 2

**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)

# Dashboard List and Editor

Part 2

Snap4City
Dashboards

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

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

Cards
⬆️ A Z
⬆️ Z A
🔍
🔄

Prev 1 2 Next

🔍
✖️

New dashboard

**DataCenter**  
Passive

disit: Public

Edit Management Clone Delete

**Datacenter Energy Consumption**  
Passive

disit: Public

Edit Management Clone Delete

**DataCenter gas and smoke (desktop)**  
Passive

disit: Public

Edit Management Clone Delete

**DataCenter gas and smoke (mobile)**  
Passive

disit: Public

Edit Management Clone Delete

**FirenzeWiFi**  
Passive

disit: Private

Edit Management Clone Delete

**Florence data overview**  
Passive

disit: Public

Edit Management Clone Delete

**Leonardo - Smart city data 2**  
Passive

Leonardo: Public

Edit Management Clone Delete

**My data and trends**  
Passive

nicola.mitolo: Public

Edit Management Clone Delete

**My data trends**  
Passive

nicola.mitolo: Public

Edit Management Clone Delete

**Notificator monitoring**  
Passive

disit: Public

Edit Management Clone Delete

**Pisa Real Time Data**  
Passive

mitolo: Public

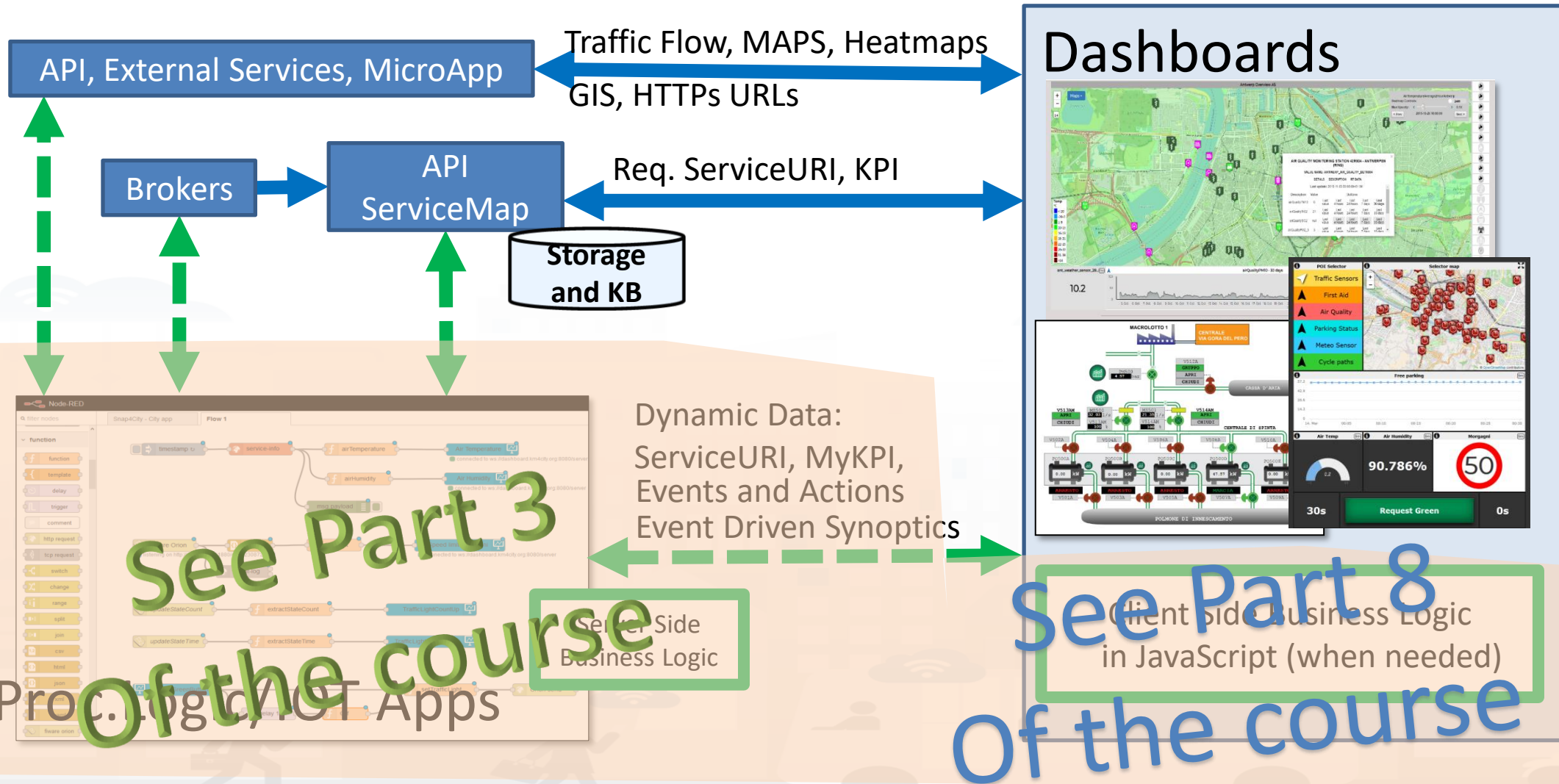
Edit Management Clone Delete

**Real Time Sensors via ServiceMap3D**  
Passive

disit: Public

Edit Management Clone Delete

# How the Dashboards exchange data



# Dashboard List and Editor

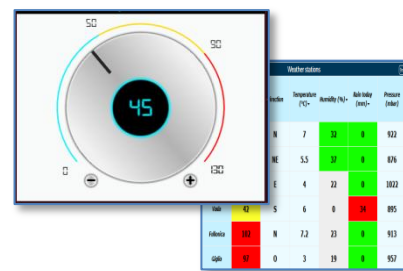
**Snap4City**  
User: root@rootadmin, Org: none  
Role: RootAdmin, Level: 7

**Dashboards**

- My Dashboards
- Notifier
- IoT Applications
- My Personal Data
- IoT Directory and Devices
- Knowledge and Maps
- Micro Applications
- External Services
- Data Set Manager: Data Gate
- Resource Manager: Process Loader
- Development Tools
- Management
- Settings
- User Management and Auditing
- Help and Contacts
- Documentation and Articles
- My Profile
- Snap4City portal
- IoT4City portal
- DISIT Lab portal

**Dashboards**

- DataCenter
- DataCenter Energy Consumption
- DataCenter gas and smoke (desktop)
- DataCenter gas and smoke (mobile)
- FlorenceWiFi
- Florence data overview
- Leonardo - Smart city data 2
- My data and trends
- My data trends
- Notifier monitoring
- Plus Real Time Data
- Real Time Sensors via ServiceMap3D



| CRID                        | 2018-09-05-10                                                      | S.P.N. 73 DI MARMANTILE - ISTITUZIONE TEMPORARY TRAFFIC LIGHTS | 16/03/2018 | 00:00:00 | 5 |
|-----------------------------|--------------------------------------------------------------------|----------------------------------------------------------------|------------|----------|---|
| CRID: 2018-09-05-11         | S.P.N. 105 DI TORRANCIANICO - ISTITUZIONE TEMPORARY TRAFFIC LIGHTS | 12/03/2018                                                     | 00:00:00   | 5        |   |
| <b>INCIDENTI SOLO DANNI</b> |                                                                    |                                                                |            |          |   |
|                             | AGGIUNTA                                                           | 11/03/2018                                                     | 10:06:12   | 1        |   |
| <b>INCIDENTI CON FERITI</b> |                                                                    |                                                                |            |          |   |
|                             | AGGIUNTA                                                           | 11/03/2018                                                     | 06:30:23   | 1        |   |
| <b>INCIDENTI SOLO DANNI</b> |                                                                    |                                                                |            |          |   |
|                             | AGGIUNTA                                                           | 11/03/2018                                                     | 05:58:48   | 1        |   |
| <b>INCIDENTI CON FERITI</b> |                                                                    |                                                                |            |          |   |
|                             | AGGIUNTA                                                           | 11/03/2018                                                     | 05:38:41   | 1        |   |



**Cam Firenze 1** | **Cam Firenze 2**

Antwerp | Helsinki | Firenze | Current | Blue

**PeopleNumber**

None

|   |   |        |
|---|---|--------|
| 7 | 8 | 9      |
| 4 | 5 | 6      |
| 1 | 2 | 3      |
| 0 | . | Cancel |

Confirm

Green

Yellow

Red



**Snap4City - Mobility Operator**

5.1

Temperature: 8.88°C

Humidity: 40.6%

City parking: 100%

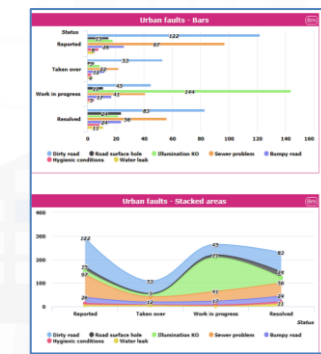
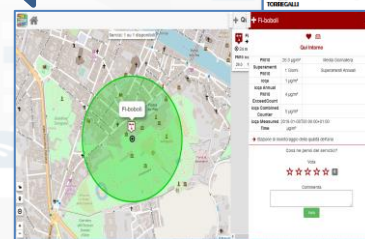
45

45

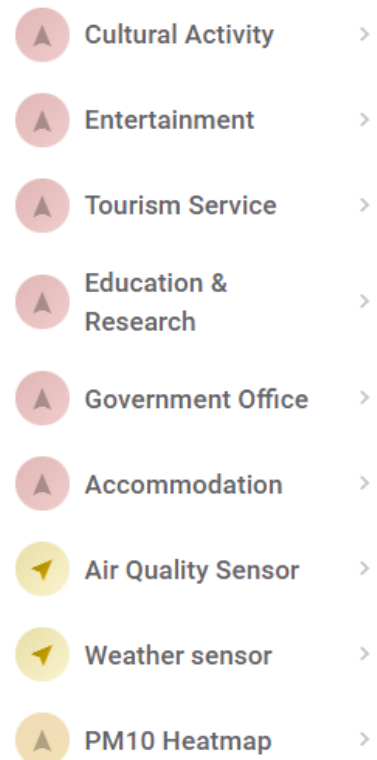
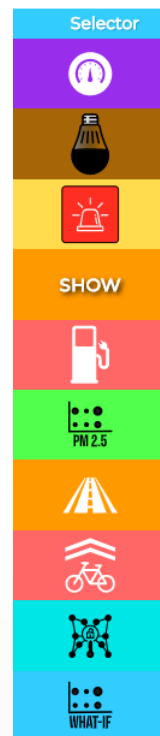
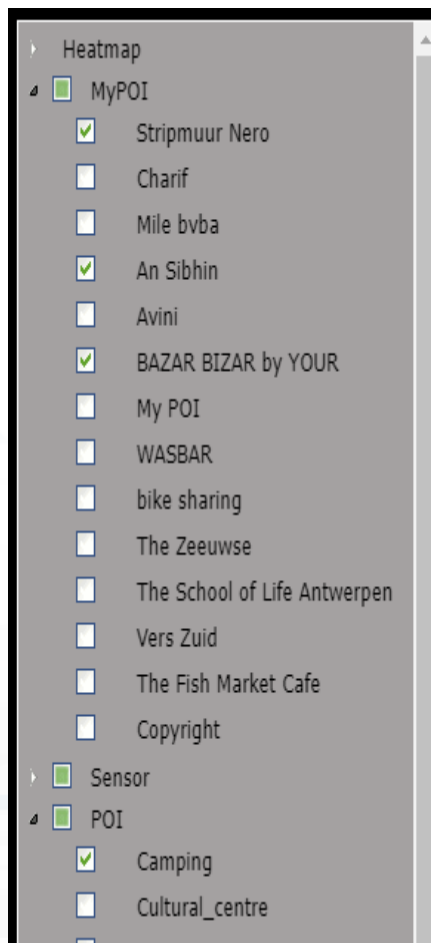
45

45

OFF Green



# 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

EDITING

The image shows the widget editor interface. On the left, a 'Selector' panel contains various icons for different data types: a purple icon with a person, a brown icon with a lightbulb, a yellow icon with a bell, an orange icon with a speaker, a red icon with a gas pump, a green icon with 'PM 2.5', and a blue icon with a bicycle. A 'Properties' panel is visible at the top right of the editor. A context menu is open over the yellow bell icon, listing options: 'Hide header', 'Hide dim ctrl', 'Header color', 'Title color', 'Background color', 'Border color', 'More options' (circled in red), 'Delete widget', and 'Quit'.

The main dashboard is titled 'Custom Pins on Map - test GP' and shows the date 'Sat 31 Oct 11:35:41'. The central map displays a city street grid with numerous custom pins, including gas pumps and bells. A legend in the bottom right corner identifies map features: 'Free street' (green line), 'Fluid traffic' (yellow line), 'Heavy traffic' (orange line), 'Very heavy' (red line), and 'Sensor position' (purple circle). Below the map is a speed graph titled 'METRO19 - averageSpeed' showing speed in km/h over time from 12:00 on Oct 25 to 12:00 on Oct 31. The graph shows several peaks in speed. At the bottom of the dashboard, there are links for 'Privacy Policy', 'Cookies Policy', 'Terms and Conditions', and 'Contact us', along with logos for the partner institutions.

<https://www.snap4city.org/dashboardSmartCity/view/index.php?iddashboard=Mjk5MA==>



# The Selector is the Map Controller

MoreOptions

## Custom Pins on Map - test GP

No subtitle

Sat 31 O

Properties Wizard Add widget Separator Embedding Screenshot Save Preview

Pin Icon

| Data widgets     | Default View Mode | Map Icon color | Alternate View Mode |
|------------------|-------------------|----------------|---------------------|
| Nothing selected |                   |                | Bub                 |
| METRO19          | Pir               |                | Cus                 |
| METRO19          | Pir               |                | Cus                 |
| METRO19          | Pir               |                | Cus                 |
| METRO19          |                   | Symbol         | Cus                 |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |
| Nothing selected |                   |                |                     |

• Targeting the data to be shows on other data Widgets

Selector

Selector - Map

| PSVG3B             |         |             |              |               |             |  |
|--------------------|---------|-------------|--------------|---------------|-------------|--|
| VALUE NAME: PSVG3B |         |             |              |               |             |  |
|                    | DETAILS | DESCRIPTION | RT DATA      |               |             |  |
| str1               | eccolo  | Last value  | Last 4 hours | Last 24 hours | Last 7 days |  |
| val1               | 0       | Last value  | Last 4 hours | Last 24 hours | Last 7 days |  |
| val2               | 2       | Last value  | Last 4 hours | Last 24 hours | Last 7 days |  |
| val3               | 34      | Last value  | Last 4 hours | Last 24 hours | Last 7 days |  |
| val4               | 4       | Last value  | Last 4 hours | Last 24 hours | Last 7 days |  |

METRO19 - av...[9m]

0.1 Km/h

val3 - 4 Hours

TOP

# *Develop Dashboard*

## *Main Concepts*



# Snap4City Dashboards main concepts



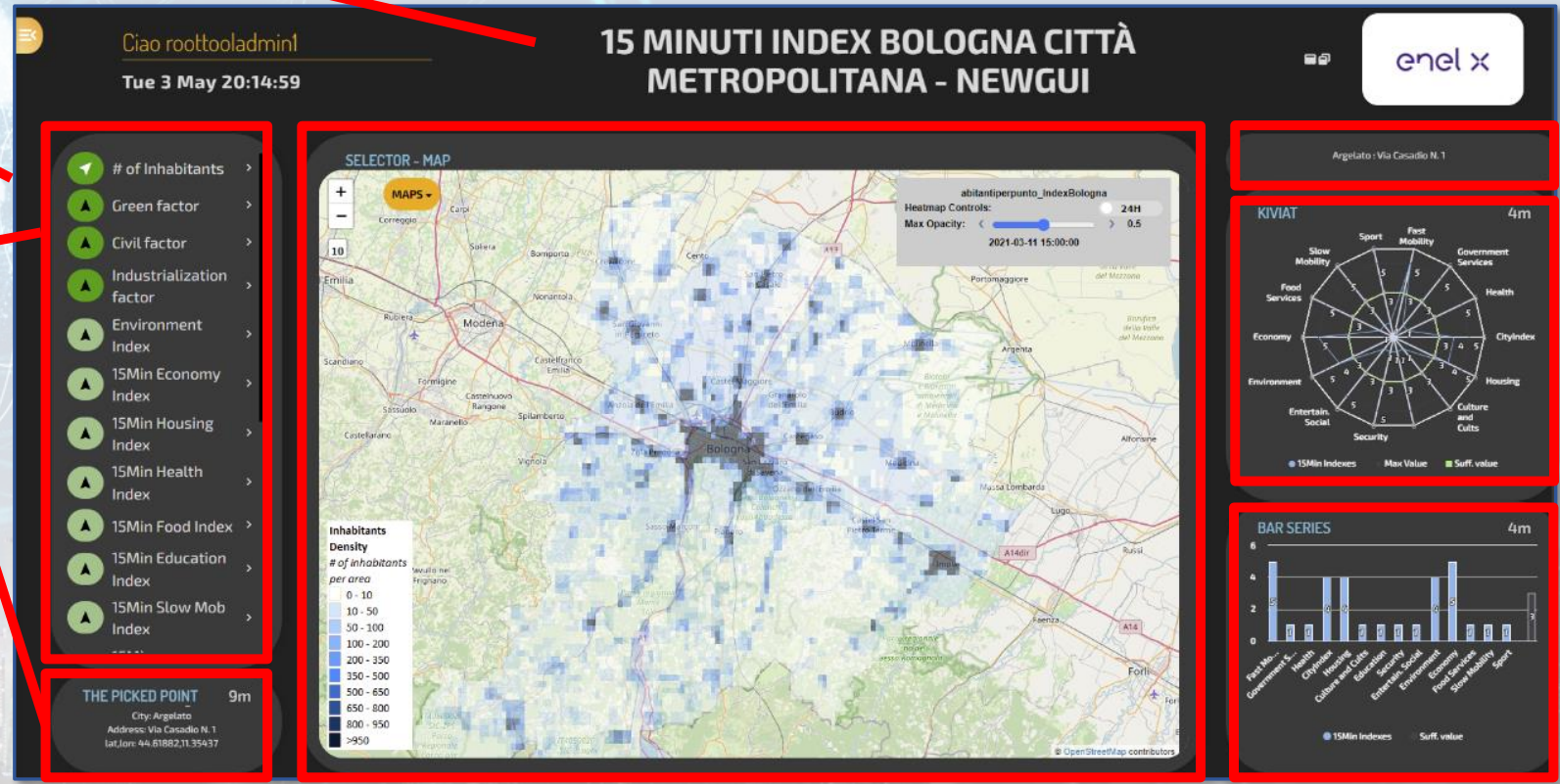
Header

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                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>Aim</b>            | Display vehicle information and measured values                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>Purpose</b>        | Monitoring                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Status</b>         | Draft                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Missing</b>        | None                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>Preferred size</b> | PC                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Style</b>          | PA                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Chat enabled</b>   | No                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Kind</b>           | Active                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <b>Data vs Widget</b> | <p><b>Map Widget</b></p> <ul style="list-style-type: none"> <li>Description: map showing the vehicle position over time</li> <li>Kind: monitoring only</li> <li>Preferred Data representation: map</li> <li>Data: <code>Vehicle.latitude</code>, <code>Vehicle.longitude</code></li> </ul> <p><b>DataTable Widget</b></p> <ul style="list-style-type: none"> <li>Description: table reporting the vehicle events</li> <li>Kind: monitoring only</li> <li>Preferred Data representation: table</li> <li>Data: <code>VehicleEvent.eventID</code>, <code>VehicleEvent.dateObserved</code>, <code>VehicleEvent.status</code>, <code>VehicleEvent.kind</code></li> </ul> <p><b>SingleContent Widget</b></p> <ul style="list-style-type: none"> <li>Description: single content showing the total km travelled by the vehicle</li> <li>Kind: IoT App</li> <li>Preferred Data representation: single number</li> <li>Data: <code>Vehicle.kmTotal</code></li> </ul> <p><b>Synoptic Widget</b></p> <ul style="list-style-type: none"> <li>Description: battery shaped synoptic to represent the available energy percentage</li> <li>Kind: monitoring only</li> </ul> |

|                                   |                                                                                                                                                                                                                                                                                                                                       |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Client Side Business Logic</b> | <ul style="list-style-type: none"> <li>To be developed in JavaScript into the Dashboard Widget</li> <li>Event driven: .....</li> </ul>                                                                                                                                                                                                |
| <b>Server Side Business Logic</b> | <ul style="list-style-type: none"> <li>To be developed in IoT App with S4C Dashboard Nodes</li> <li>IoT Application →               <ul style="list-style-type: none"> <li>Event driven: .....</li> </ul> </li> <li>IoT Application →               <ul style="list-style-type: none"> <li>Event driven: .....</li> </ul> </li> </ul> |

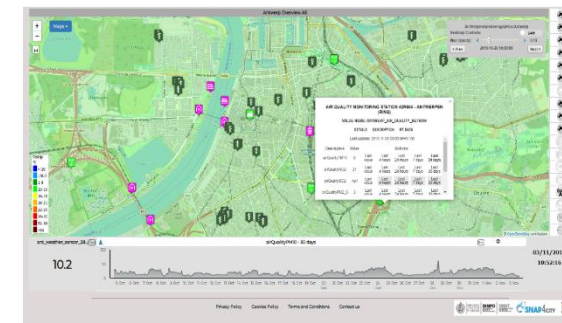
As can be seen in the example dashboard schema above, several information must be specified:

- 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

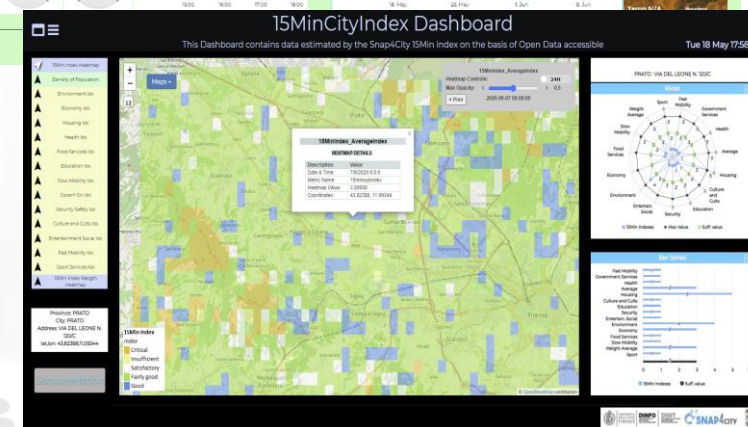
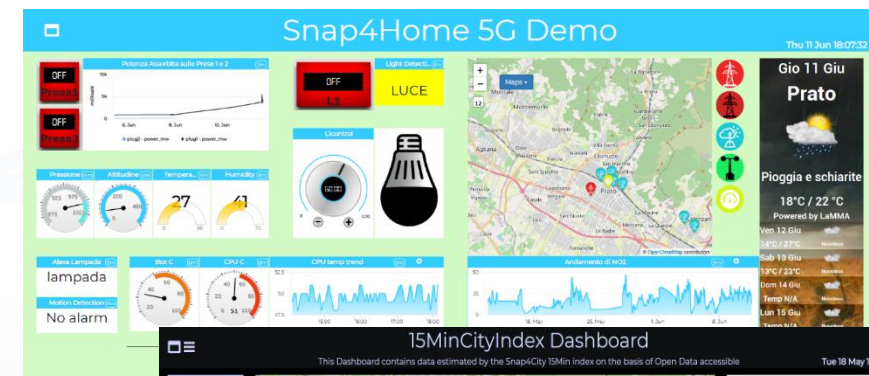
- Passive dashboards may have Selectors, maps, etc., and a lot of visualization without changing the status on Server, no sending commands to the Server Side.



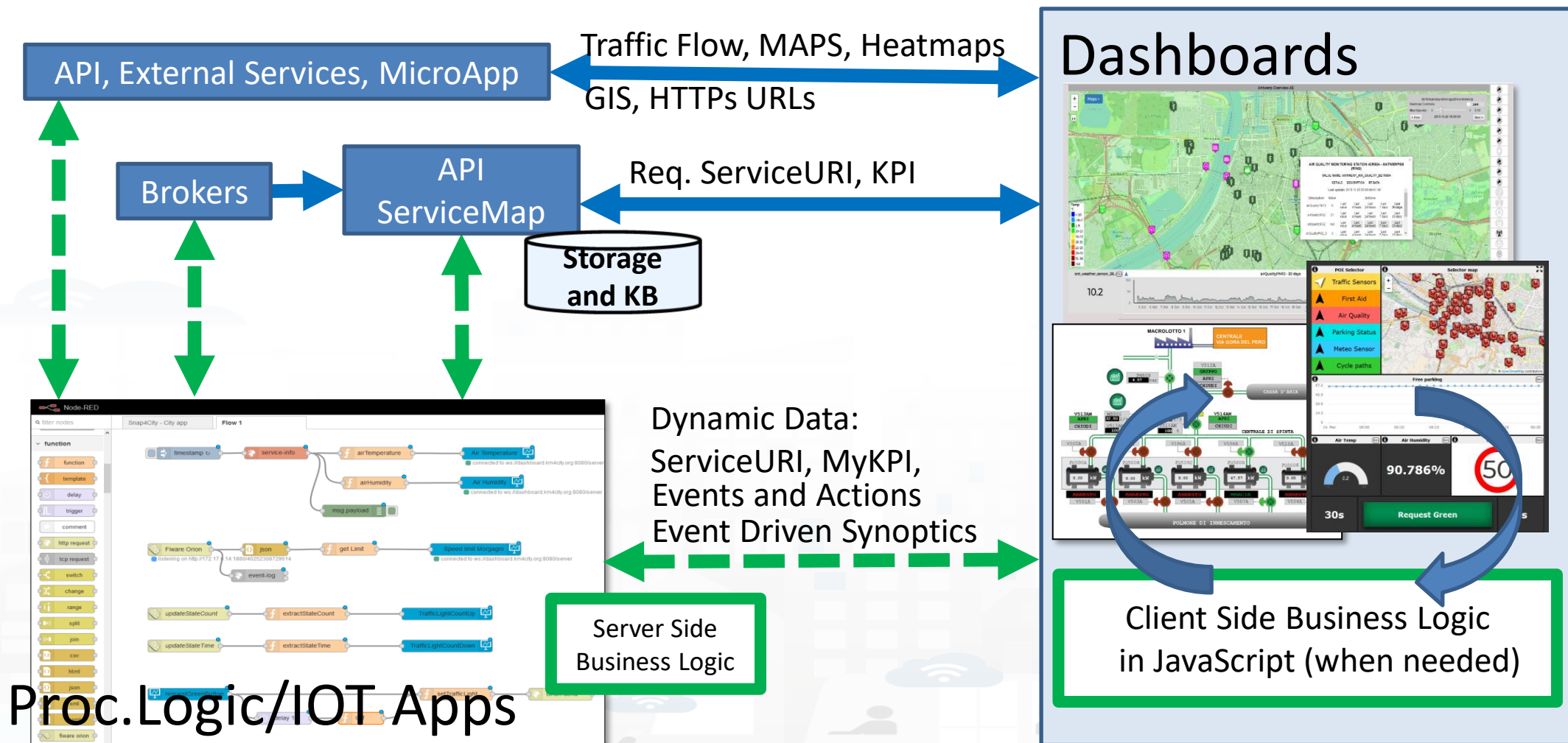
- **Custom Business Logic**

- **Active Dashboards,** which can be those sending or receiving commands to/from the logic coded somehow and in particular for

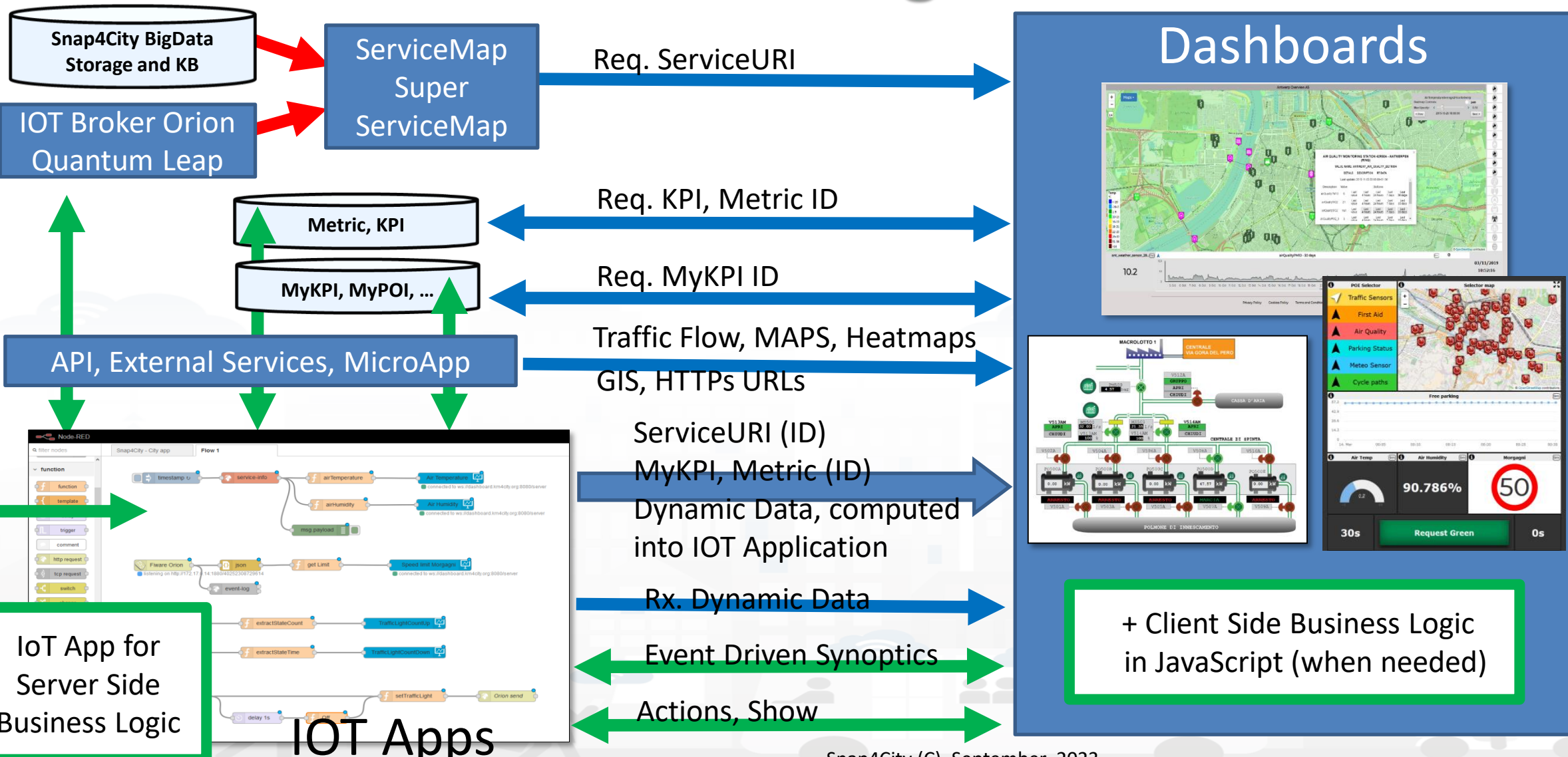
- **Server Side Business Logic** → logic on IoT Apps with Snap4City Dashboard Nodes, which is easier to be programmed begin based on Node-RED visual programming.
- **Client Side Business Logic** → logic on JavaScript on specific Dashboard Widgets only for skilled developers of Snap4City Platform. We 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



TOP

# *Develop: via Dashboard Wizard*



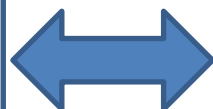
# Dashboard Builder: Development

Data Transformation  
Business Logic

IOT Applications

Knowledge Base,  
Km4City

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



Widget Collection

Micro Applications

External Services

Custom Widgets/  
Synoptics



Dashboard Wizard

Dashboard Editor

Public Dashboard Collection

My Own Dash/App

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



# New Dash ?

**My Dashboards in My Organization**

User: paolo.disit, Org: DISIT  
Role: AreaManager, Level: 3  
[LOGOUT](#) [Switch to Legacy Layout](#)

**NEW DASHBOARD**

Prev 1 2 3 Next My own

| Dashboard Title                                    | Proc.Logic / IoT App | Ownership              |
|----------------------------------------------------|----------------------|------------------------|
| Alerting Generation                                | Proc.Logic / IoT App | My own: Public (DISIT) |
| Andamenti Nazionali e Regionali infezione COVID-19 | Proc.Logic / IoT App | My own (DISIT)         |
| Andamento Regione Toscana e Province, COVID-19     | Proc.Logic / IoT App | My own (DISIT)         |
| Case 1 SVG ws3                                     | Proc.Logic / IoT App | My own (DISIT)         |
| case 2 SVG WS3                                     | Proc.Logic / IoT App | My own (DISIT)         |
| Case4 svg                                          | Passive              | My own (DISIT)         |
| Change Alert Color Status                          | Proc.Logic / IoT App | My own: Public (DISIT) |
| DataCenter new Device DHT                          | Proc.Logic / IoT App | My own: Public (DISIT) |
| Device Table Testing double                        | Proc.Logic / IoT App | My own (DISIT)         |
| DIDA data 2                                        | Passive              | My own: Public (DISIT) |

Privacy Policy Cookies Policy

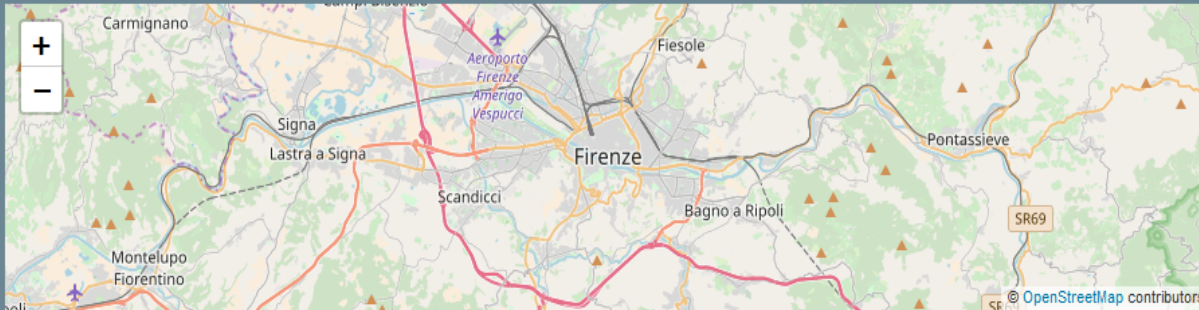
## Wizard



## Dashboard features

## Data and widgets

## Map



## Single data widgets



## Multi data widgets



## Data sources

| High-Level Type | Nature      | Subnature        | Value Type  | Value Name  | Data Type       | Last Date | Last Value | Healthiness | Last Check          | Ownership |
|-----------------|-------------|------------------|-------------|-------------|-----------------|-----------|------------|-------------|---------------------|-----------|
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Previ_Meteo | Previ_Meteo | special weather |           |            |             | 2018-07-08 16:00:18 | public    |

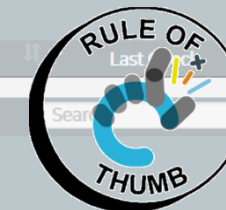
- Select the area of your interest: panning and zooming

- Select the

- graphic aspect of your interest, or
- High Level Type of your interest, or
- Make a search if you 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



Close

SNAP4CITY Dashboards Wizard

Dashboard features | Data and widgets | Check and summary

Map

Single data widgets

Multi data widgets

| High-Level Type | Nature      | Subnature        | Value Type | Value Name      | Last Date           | Last Value | Healthiness | Last Check          | Ownership |
|-----------------|-------------|------------------|------------|-----------------|---------------------|------------|-------------|---------------------|-----------|
| Special Widget  | Environment | Weather Forecast | Prev_Meteo | special weather | 2018-07-08 16:00:18 | Vernio     | ●           | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Prev_Meteo | special weather | 2018-07-08 16:00:18 | Vergemoli  | ●           | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Prev_Meteo | special weather | 2018-07-08 16:00:18 | Vecchiano  | ●           | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Prev_Meteo | special weather | 2018-07-08 16:00:18 | Vaiano     | ●           | 2018-07-08 16:00:18 | public    |
| Special Widget  | Environment | Weather Forecast | Prev_Meteo | special weather | 2018-07-08 16:00:18 | Vaglia     | ●           | 2018-07-08 16:00:18 | public    |

Selected rows: 0

Chosen data sources

No data available in table

You must select one widget type

Prev Next



# Dashboard Wizard

Wizard

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

## Test api from Time

Thu 8 Mar 09:18:52

Selector

- Traffic Sensors
- First Aid
- Smart waste
- Meteo sensor in via Bolognese
- Air quality
- Pollination
- Parking Status
- Smart bench
- Bike sharing (Pisa)

Map

METRO759

DETAILS DESCRIPTION RT DATA

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

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

Vehicle Flow - 30 days

TOP

# *Develop:* *Dashboard Synoptics*

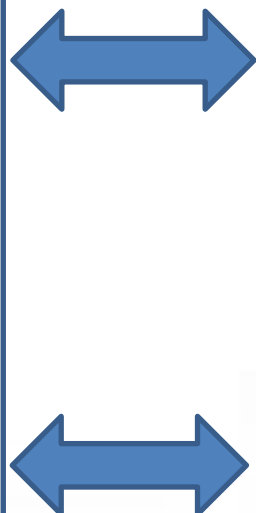


## Custom Widget / Synoptic / PIN Development

IOT Applications

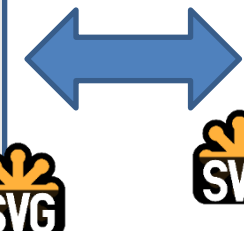
Knowledge Base, Km4City

Knowledge and Storage Data from the Field and City



SVG Symbols Collection

0.04 SMC

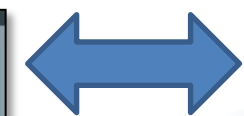


Inkscape editor on your computer

Create, save a Custom Widget in SVG

Public Dashboard Collection

My Own Dash/App



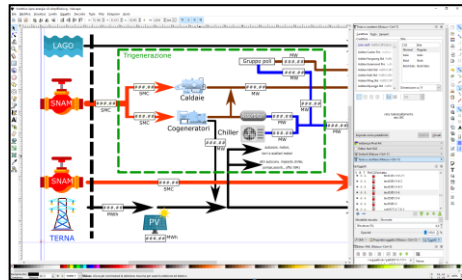
Dashboard Editor

Create, save, load, delegate, grant access

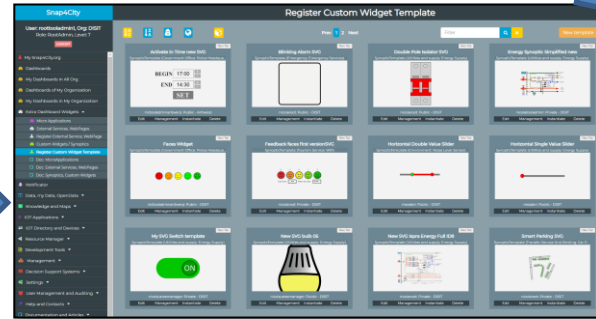
1. Create and Load a Custom SVG
2. Select/Reuse an SVG
3. Make and Instance of Synoptic by Associate Variables with MyKPI
4. Create on Dashboard a Widget based on Synoptic HLT such as Ext. Srv.:
  - <https://www.snap4city.org/synoptic/v2/synoptic.html?id=xxxx>

**CW with a single READ Variable are automatically usable as PINS**

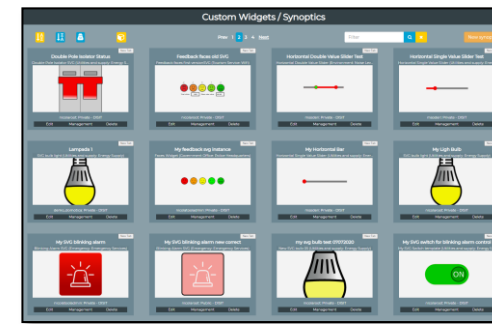
**Create, save a Custom Widget in SVG**



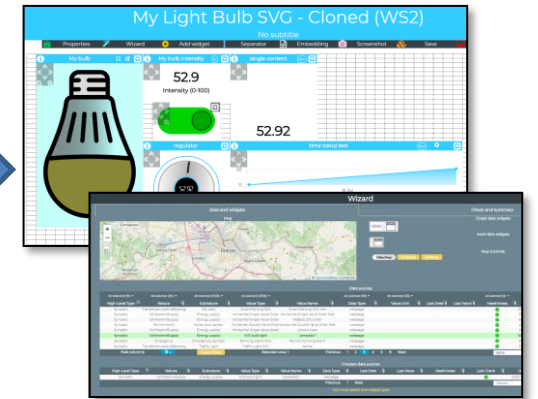
**Upload as Custom Widget Template**



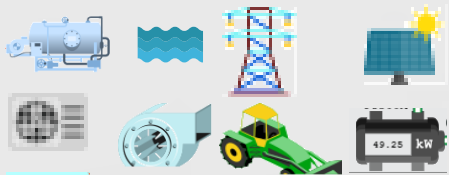
**List of Custom Widgets / Synoptics**



**Dashboard Editing/wizard**

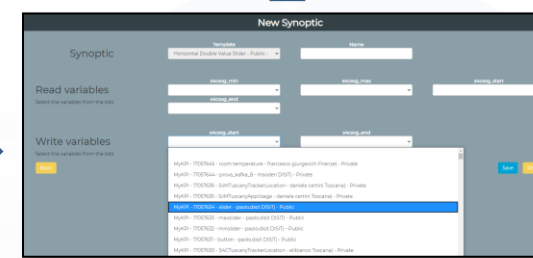
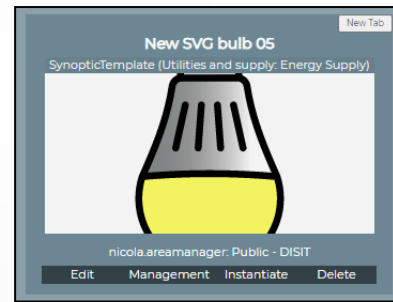


**SVG Symbols Collection**

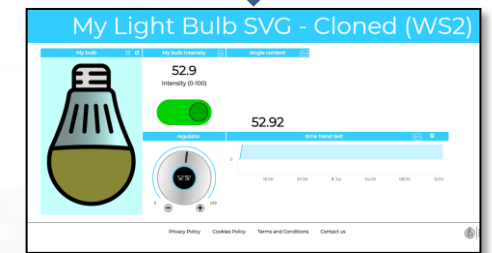


**From any open library**

**select**

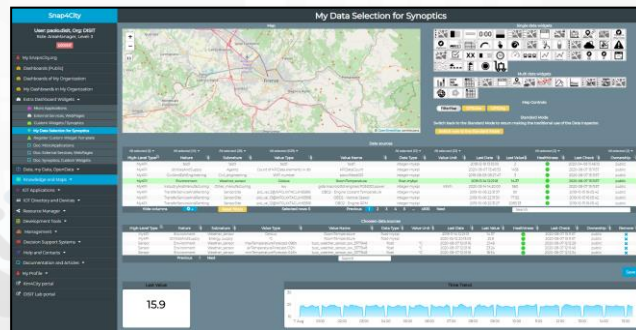


**Instantiate as Custom Widgets / Synoptics  
Connect with WebSockets**



**Final Dashboard**

**Select MyKPI and Sensor Data for Synoptics cases**

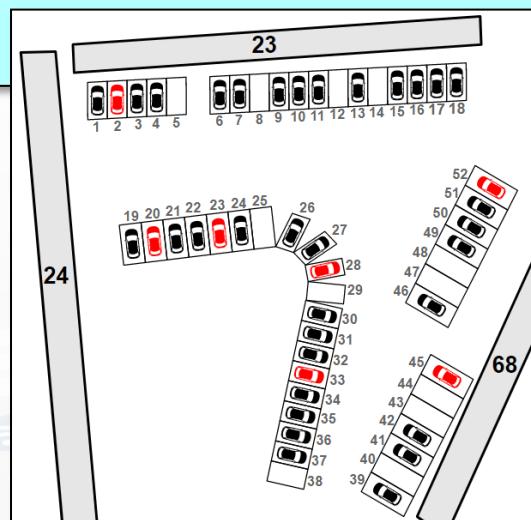
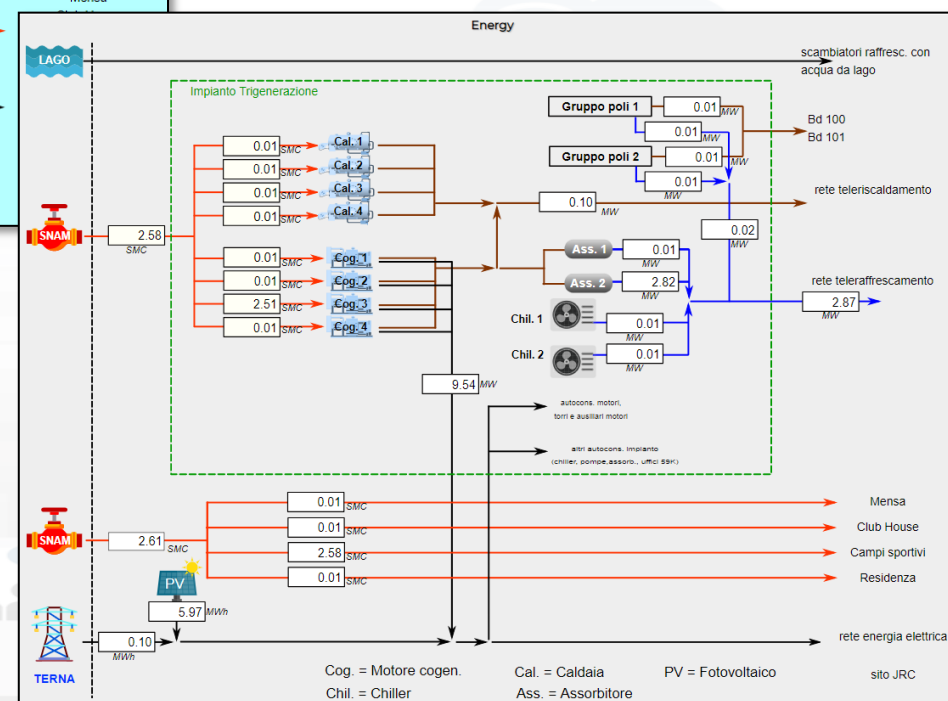
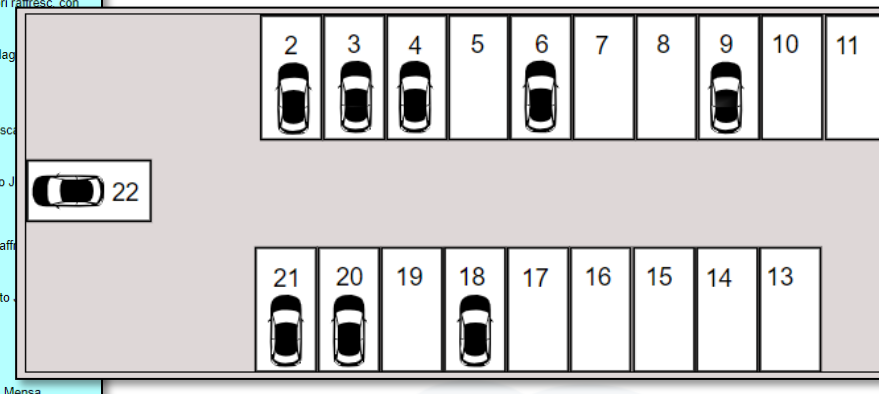
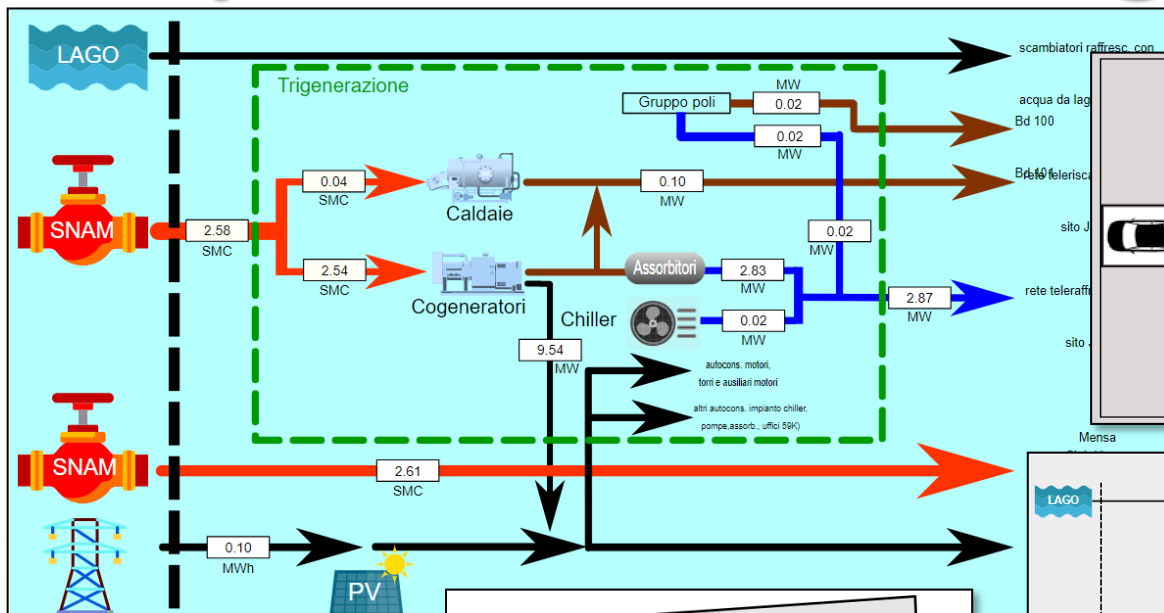


**Part 3**



## Special Custom Widgets

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



SVG

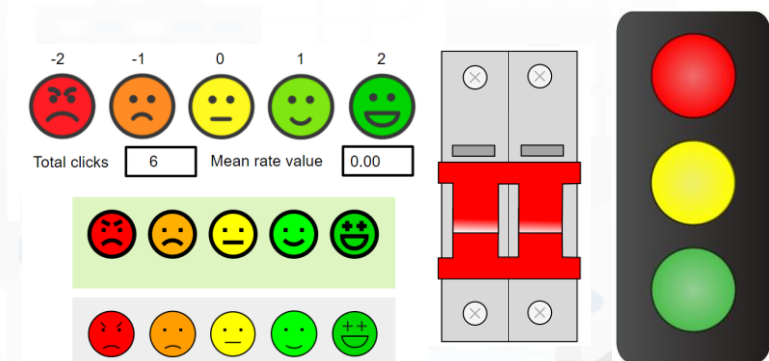
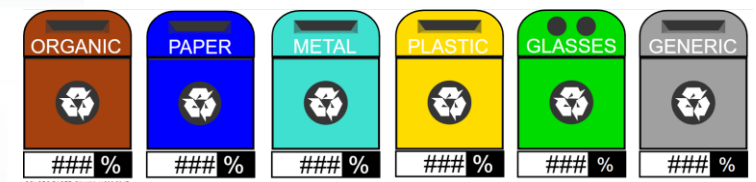
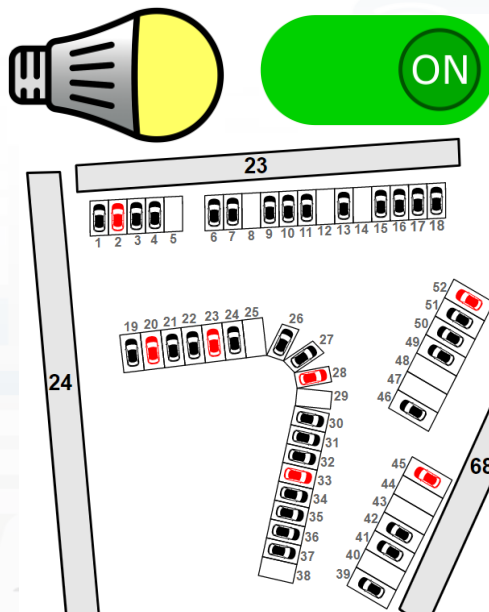
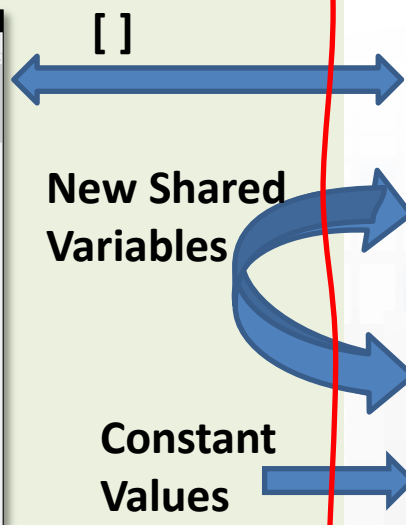
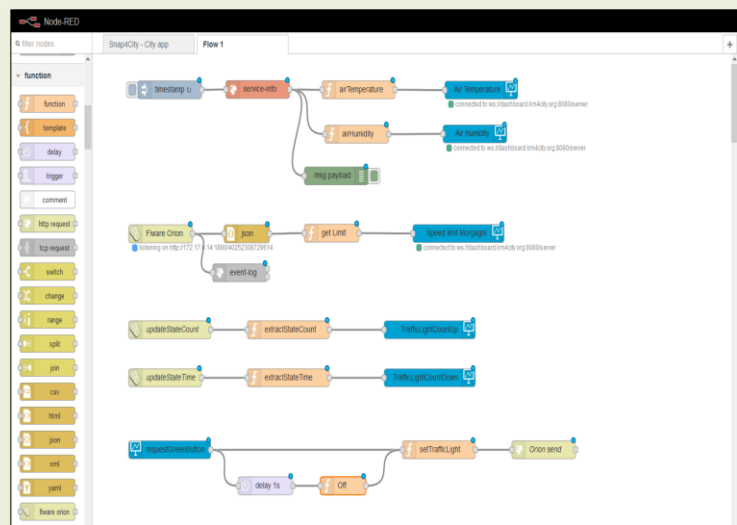
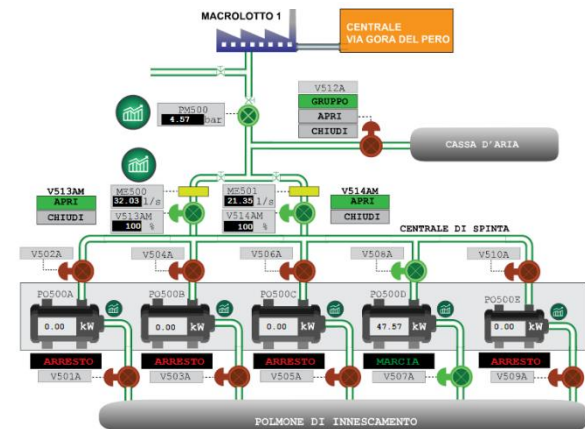
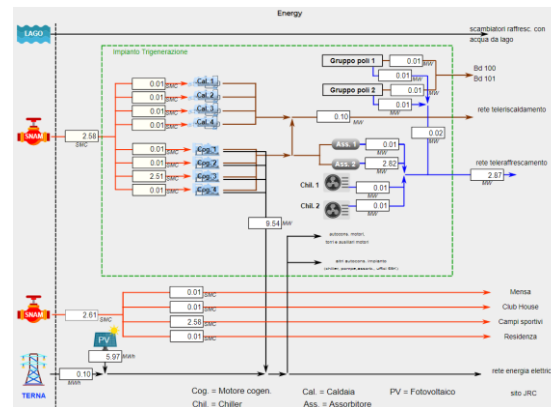
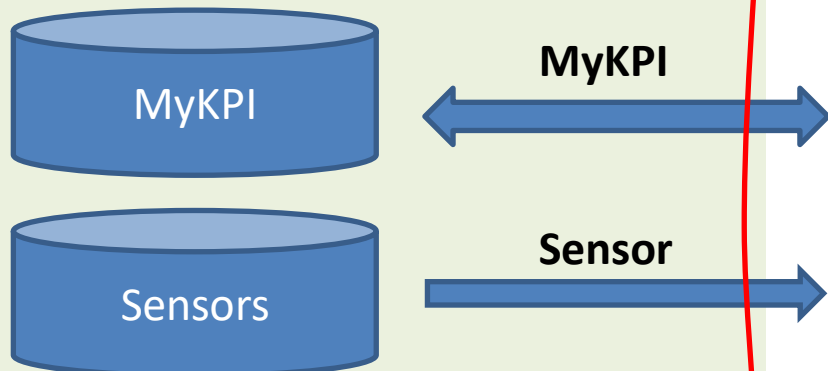
-2 -1 0 1 2

Total clicks 6 Mean rate value 0.00

Begin 17:00

Finish 4:00

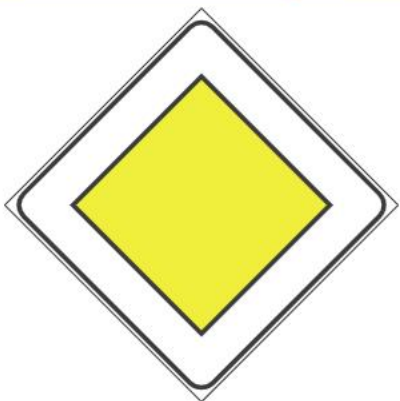
## From-To Custom Widgets / Synoptics to Storage in WS



## SVG Custom Widgets Examples

Sat 16 Jan 01:07:39

Precedence Italians Road signals



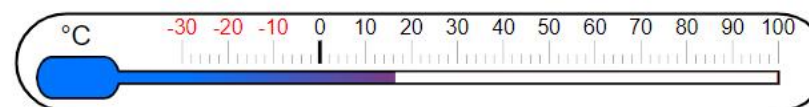
Select a code from 0 to 11 to change the road sign

| New     |   |      | Last confirmed |  |  |
|---------|---|------|----------------|--|--|
|         |   |      | None           |  |  |
| 7       | 8 | 9    |                |  |  |
| 4       | 5 | 6    |                |  |  |
| 1       | 2 | 3    |                |  |  |
| 0       | . | Canc |                |  |  |
| Confirm |   |      |                |  |  |

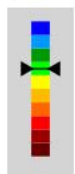
Smart Light Luminosity



Air Temperature in Florence



PM10 level - Bologna

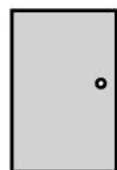


fan

Fan velocity



open/...



Dynamic Prohibition...



Prohibition Traffic Signs Co...

| New     |   |      | Last confirmed |  |  |
|---------|---|------|----------------|--|--|
|         |   |      | None           |  |  |
| 7       | 8 | 9    |                |  |  |
| 4       | 5 | 6    |                |  |  |
| 1       | 2 | 3    |                |  |  |
| 0       | . | Canc |                |  |  |
| Confirm |   |      |                |  |  |

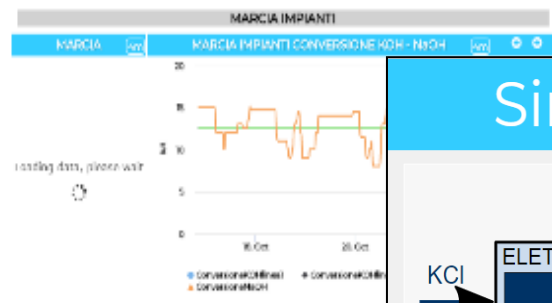
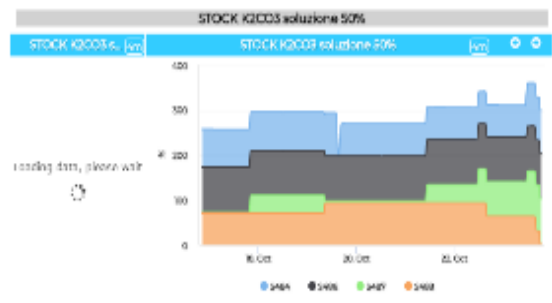
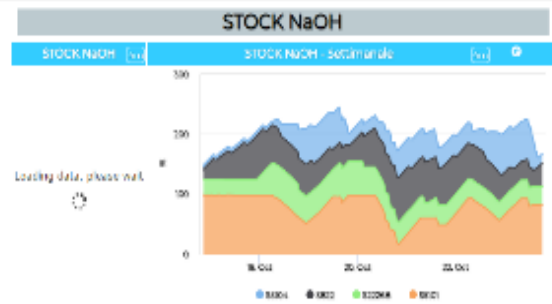
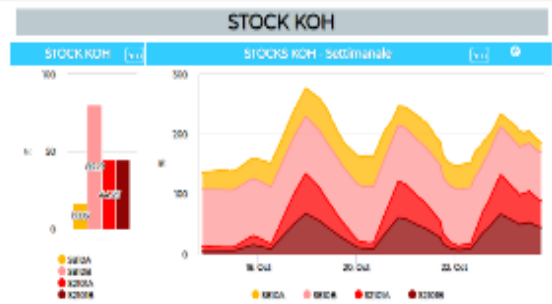


Prohibition Traffic Signs Legend



Symbols Legend

- ▲ NaOH KOH - 1
- ▲ NaOH KOH - 2
- ▲ HCl
- ▲ Cloroparaffine
- ▲ Idrogeno gas
- ▲ Cloruro Ferrico Ferroso POT
- ▲ Cloruro Ferrico Ferroso POT



- Home
- DCS Real Time VS Planning
- DCS Real Time Giornaliero
- DCS Real Time Settimanale
- DCS Marce Giornaliero
- DCS Marce Settimanale
- DCS Stocaggi



## RTO online

| Localizzazione (id data) | Energia (PUN)       | Altri Parametri     | Pianificazione      | Esito Pianificazione | In Produzione                       |
|--------------------------|---------------------|---------------------|---------------------|----------------------|-------------------------------------|
| 0-01 09:32:54            | 2020-10-01 23:00:00 | 2020-07-24 18:43:00 | 2020-10-01 09:33:27 | completato           | <input type="checkbox"/>            |
| 0-09 17:20:50            | 2020-09-30 23:00:00 | 2020-07-24 18:43:00 | 2020-09-30 17:21:00 | completato           | <input checked="" type="checkbox"/> |
| 0-09 16:24:57            | 2020-09-30 23:00:00 | 2020-07-24 18:43:00 | 2020-09-30 16:27:23 | completato           | <input type="checkbox"/>            |
| 0-09 14:54:11            | 2020-09-30 23:00:00 | 2020-07-24 18:43:00 | 2020-09-30 14:56:22 | completato           | <input type="checkbox"/>            |
| 0-09 13:43:47            | 2020-09-30 23:00:00 | 2020-07-24 18:43:00 | 2020-09-30 13:43:57 | completato           | <input type="checkbox"/>            |
| 0-29 19:03:27            | 2020-09-30 23:00:00 | 2020-07-24 18:43:00 | 2020-09-29 19:03:43 | completato           | <input type="checkbox"/>            |
| 0-28 18:30:13            | 2020-09-29 23:00:00 | 2020-07-24 18:43:00 | 2020-09-28 18:30:23 | completato           | <input type="checkbox"/>            |
| 0-28 17:57:14            | 2020-09-29 23:00:00 | 2020-07-24 18:43:00 | 2020-09-28 17:57:23 | completato           | <input type="checkbox"/>            |
| 0-28 15:50:21            | 2020-09-28 23:00:00 | 2020-07-24 18:43:00 | 2020-09-28 15:50:45 | completato           | <input type="checkbox"/>            |
| 0-25 18:46:02            | 2020-09-26 23:00:00 | 2020-07-24 18:43:00 | 2020-09-25 18:47:46 | completato           | <input checked="" type="checkbox"/> |

- Home
- DCS Real Time VS Planning
- DCS Real Time Giornaliero
- DCS Real Time Settimanale
- DCS Marce Giornaliero
- DCS Marce Settimanale
- DCS Stocaggi Giornaliero
- DCS Stocaggi Settimanale

### Sinottico Sintesi Impianto Altair 2

Thu 1 Oct 15:32:19

**Inputs:** KCl, E.E., NaCl

**Process:** ELETTROLISI

**Outputs:**

- Idrogeno gas
- Cloro gas
- SINTESI HCL: 4.62, 123.5, 112.91, 90.06 (ton)
- CLOROPARAFFINE: 0.92, 9.06, 9.46 (ton)
- CLORURO FERRICO FERROSO: 0.15, 233.58, 0 (ton)
- IPOSODIO: 87.97 (ton)
- NaOH 30%: 196.79, 0, 30.8 (ton)
- NaOH 50%: 720.37, -0.06, 45.81, 12, 101.99, 0 (ton)
- KOH50%: 196.79 (ton)
- KOH SCAGLIE: -0.06 (ton)
- K2CO3 gr.: 45.81 (ton)
- K2CO3 sol. 50%: 12 (ton)
- NaOH 50%: 31.82 (ton)

- Home
- DCS Real Time VS Planning
- DCS Real Time Giornaliero
- DCS Real Time Settimanale
- DCS Marce Giornaliero
- DCS Marce Settimanale
- DCS Stocaggi Giornaliero
- DCS Stocaggi Settimanale
- Sinottico di sintesi impianto (ton/g)
- Sinottico di sintesi impianto (ton/h)
- RTO online



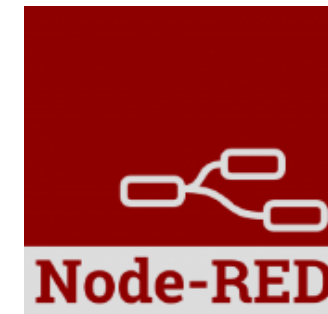
TOP

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



# IoT App / Proc.Logic

- Storage → IoT App / Proc.Logic
- External Service ↔ IoT App / Proc.Logic **Part 3**
- **Dashboards ↔ IoT App / Proc.Logic**

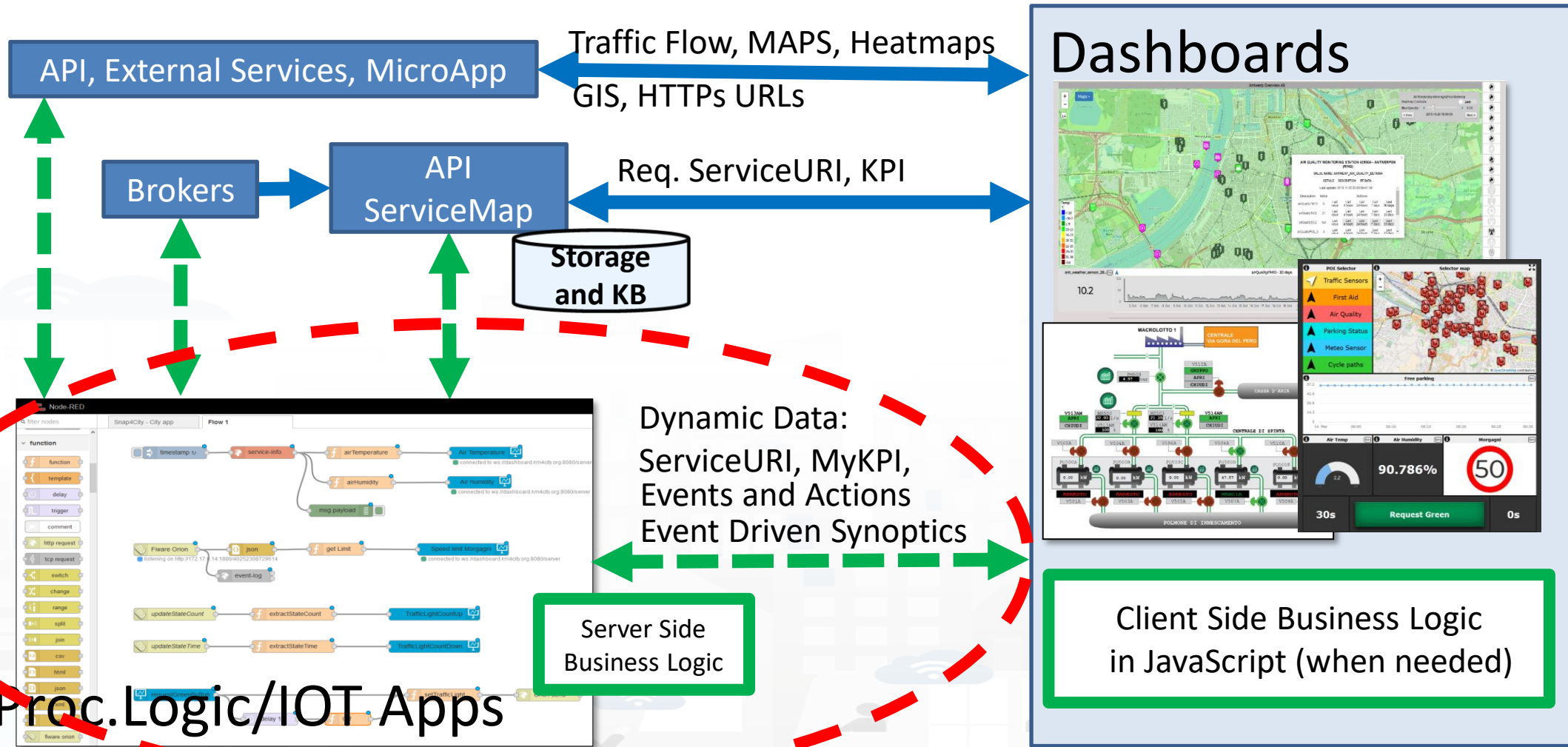


- Data Analytics ↔ IoT App / Proc.Logic **Part 4**
  - Broker → Storage
  - IoT App / Proc.Logic → Broker
  - Broker → IoT App / Proc.Logic
  - IoT App / Proc.Logic → Storage
- Part 5**

# Principles of Server Side Business Logic

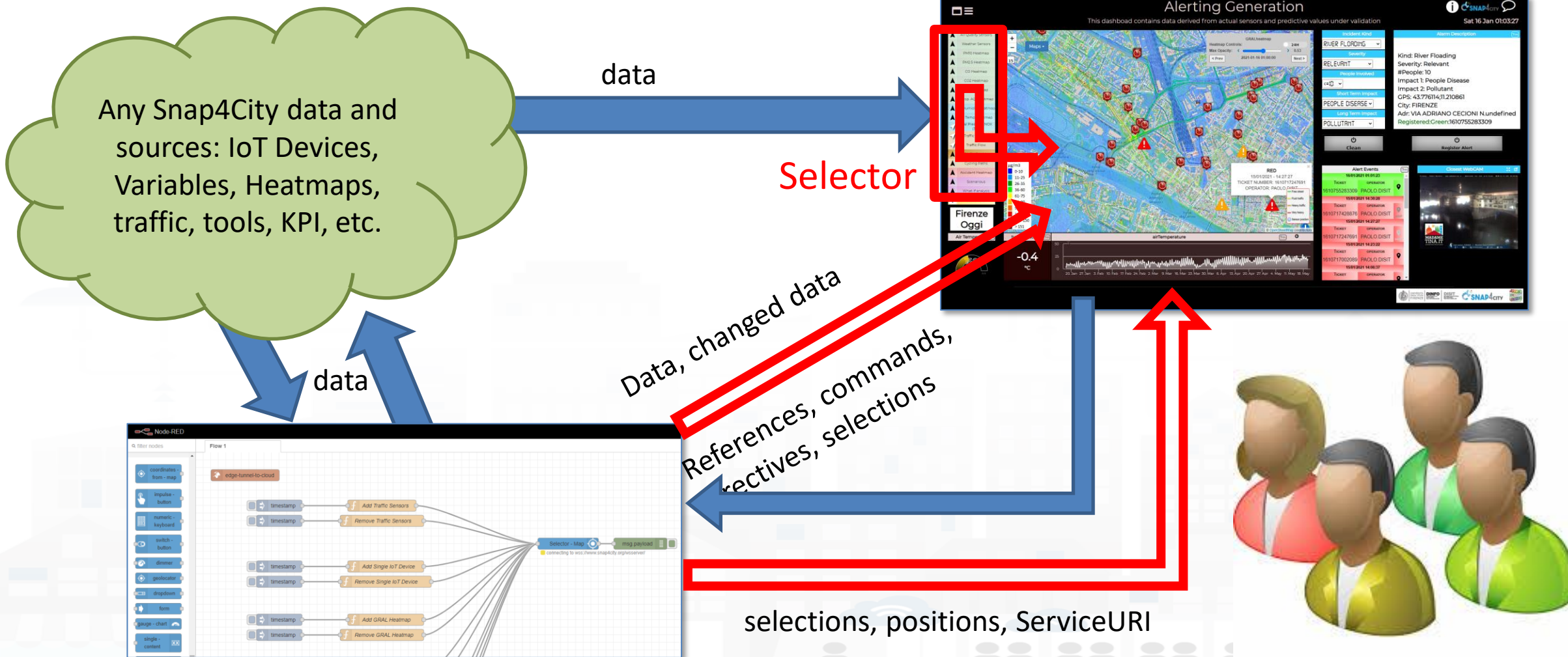
- It is possible to have one Processing Logic (IoT App) referring to multiple Dashboards, and one Dashboard referring multiple Processing Logic (IoT Apps)
- Let see a 1:1 relationship from Proc.Logic and Dashboard
  - Any Action performed on Dashboard is provided to the Proc.Logic, which may produce reactions on Dashboard.
  - The context of Proc.Logic  $\leftrightarrow$  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

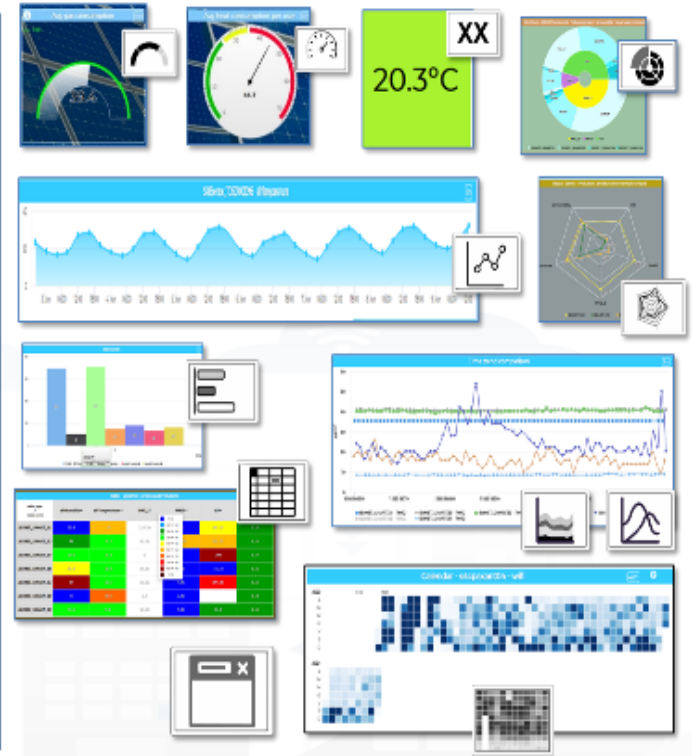
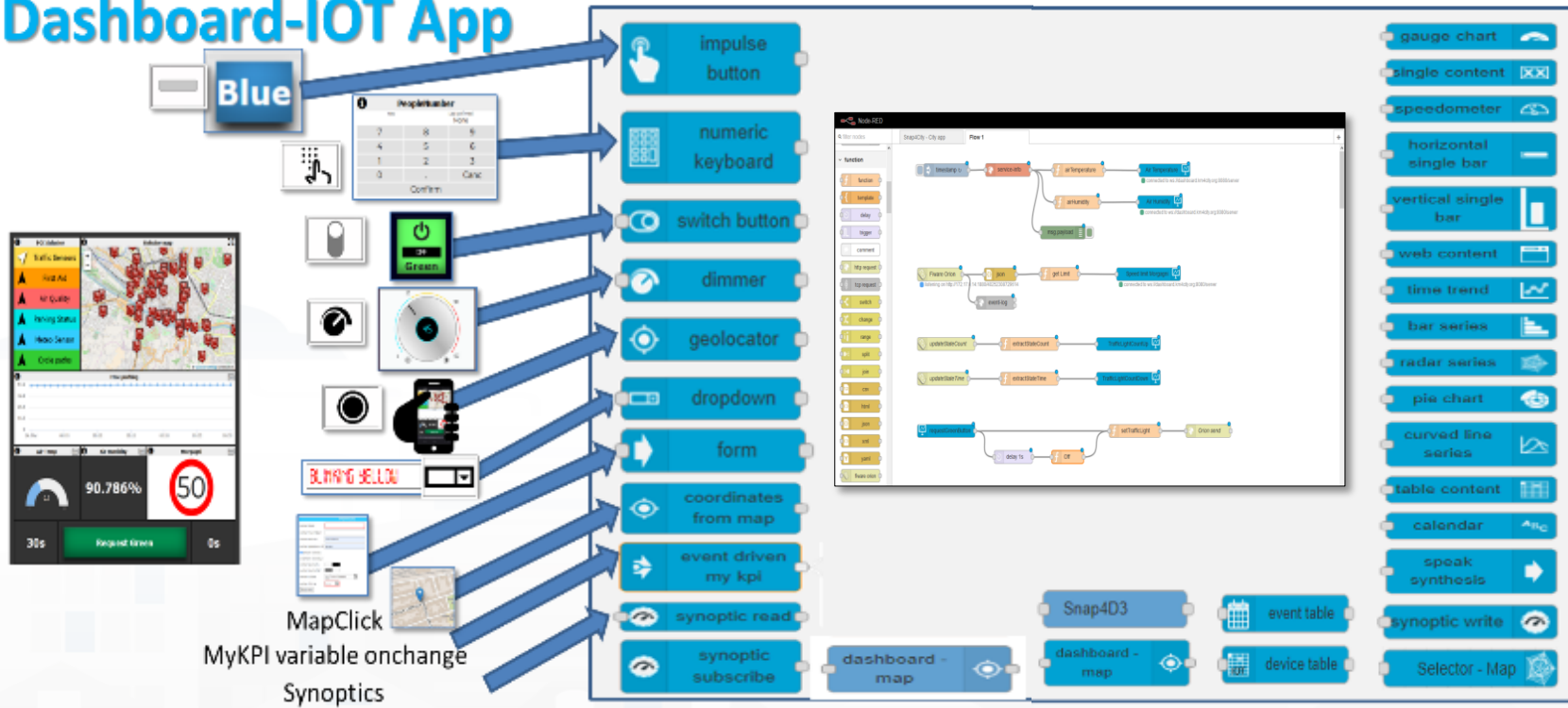




# Maps Server Side Business Logic vs IOT Apps



# Dashboard-IOT App



| Widgets ICONS | Widget Name, Description                          | IOT App | Dashboard-IOT App | KPI (metric)          | MyPersonalID ata | MyData | My KPI | Sensor |
|---------------|---------------------------------------------------|---------|-------------------|-----------------------|------------------|--------|--------|--------|
|               | Single Content                                    | X (cs)  | X (ED)            | X                     | X                | X      | X      | X      |
|               | 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      |
|               | Time Trend Compare                                |         |                   | X                     |                  |        | X      | X      |
|               | SpiderNet, radar, Kiviati                         | 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

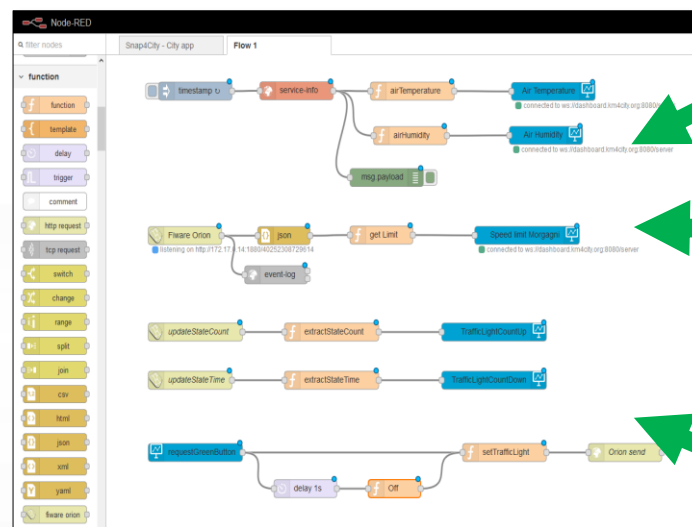
# Widgets and their counterpart Nodes

- **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**).

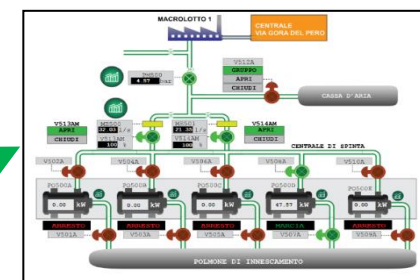
On Server-Side (into Proc.Logic) the developer can even create some HTML pages and provide them into a Dashboard Widget. And a mixt of Widgets in, out, in/out

# Advanced IOT Applications

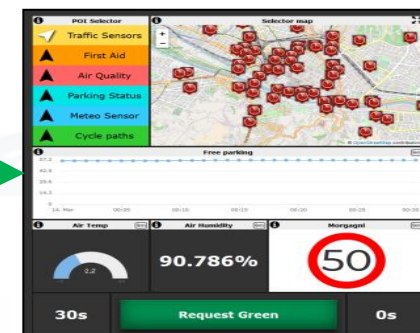
- **Synoptics** can .....
  - do all ☺
- **Widgets** can
  - send/receive dynamic data,
  - change data sources, etc.
  - Provide interactive maps
- **HTML pages** can
  - be dynamically generated
  - provide forms to produce data for IOT Applications
  - Collect files on web and system
  - produce files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



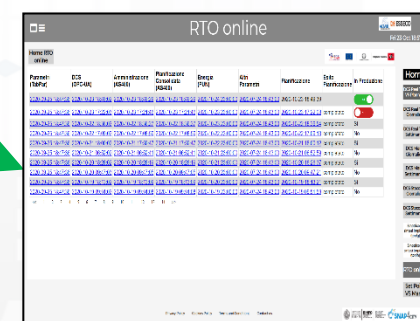
Synoptics  
Custom  
Widgets



Widgets  
Maps  
Buttons  
Keypads  
Controls



HTML pages  
HTML Forms  
HTML Tables



<https://www.snap4city.org/394>

<https://www.snap4city.org/596>

# Proc.Logic (IoT App) with Dynamic Web Pages

RTO online  
Fri 23 Oct 18:57:41

| Parametri (TabPar)                  | DCS (OPC-UA)                        | Amministrazione (AS400)             | Pianificazione Consolidata (AS400)  | Energia (PUN)                       | Altri Parametri                     | Pianificazione      | Esito Pianificazione | In Produzione            |
|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|---------------------|----------------------|--------------------------|
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-23 18:49:02</a> | <a href="#">2020-10-23 18:49:29</a> | <a href="#">2020-10-23 18:49:29</a> | <a href="#">2020-10-24 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-23 18:49:39 |                      | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-23 17:22:03</a> | <a href="#">2020-10-23 17:21:46</a> | <a href="#">2020-10-23 17:21:46</a> | <a href="#">2020-10-23 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-23 17:22:08 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-22 18:36:02</a> | <a href="#">2020-10-22 18:36:27</a> | <a href="#">2020-10-22 18:36:27</a> | <a href="#">2020-10-23 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-22 18:36:54 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-22 17:09:02</a> | <a href="#">2020-10-22 17:08:59</a> | <a href="#">2020-10-22 17:08:59</a> | <a href="#">2020-10-22 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-22 17:09:13 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-21 18:00:02</a> | <a href="#">2020-10-21 17:59:47</a> | <a href="#">2020-10-21 17:59:47</a> | <a href="#">2020-10-22 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-21 18:00:12 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-21 06:52:02</a> | <a href="#">2020-10-21 06:52:41</a> | <a href="#">2020-10-21 06:52:41</a> | <a href="#">2020-10-21 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-21 06:52:59 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-20 18:26:02</a> | <a href="#">2020-10-20 18:26:19</a> | <a href="#">2020-10-20 18:26:19</a> | <a href="#">2020-10-21 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-20 18:26:37 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-20 09:47:03</a> | <a href="#">2020-10-20 09:47:05</a> | <a href="#">2020-10-20 09:47:05</a> | <a href="#">2020-10-20 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-20 09:47:21 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-19 18:13:02</a> | <a href="#">2020-10-19 18:13:09</a> | <a href="#">2020-10-19 18:13:09</a> | <a href="#">2020-10-20 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-19 18:13:21 | completato           | <input type="checkbox"/> |
| <a href="#">2020-09-25 18:47:36</a> | <a href="#">2020-10-19 09:51:02</a> | <a href="#">2020-10-19 09:51:08</a> | <a href="#">2020-10-19 09:51:08</a> | <a href="#">2020-10-19 23:00:00</a> | <a href="#">2020-07-24 18:43:00</a> | 2020-10-19 09:51:59 | completato           | <input type="checkbox"/> |

RTO online  
Thu 1 Oct 15:33:23

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

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

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

Salva Parametri

**Elenco Parametri Iniziali Algoritmo RTO SODA4.0**  
(effettuare cambiamenti che saranno utilizzati dalla prossima esecuzione)

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

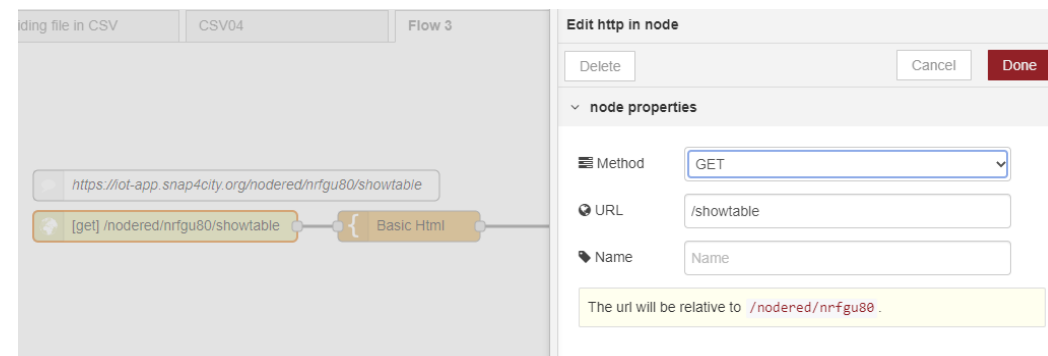
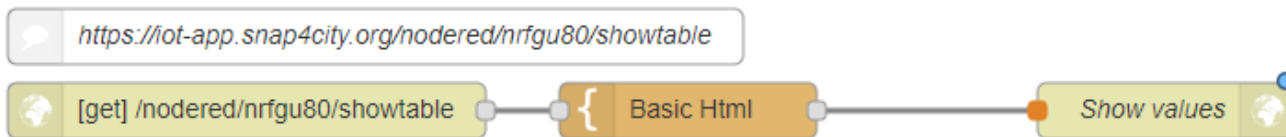
1 alpha\_eco (Peso che decide l'importanza relativa di Stock e PUN nella funzione obiettivo) [-]

0.31 Cl2\_HCl32 (Cl2 consumato per ton di prodotto) [ton Cl2/ton HCl32]

297.54 MAX\_HCl32\_s (Massimo stock HCl32) [ton]

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

# From IoT App to generate HTML pages, forms



```

<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

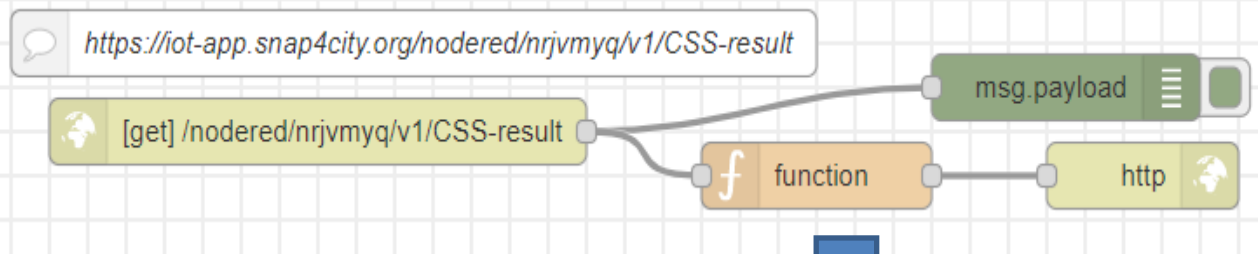
| Person 1 | Person 2 | Person 3 |
|----------|----------|----------|
| Emil     | Tobias   | Linus    |
| 16       | 14       | 10       |

# HTML & Tables on Dashboards

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



# From IoT App to API Get

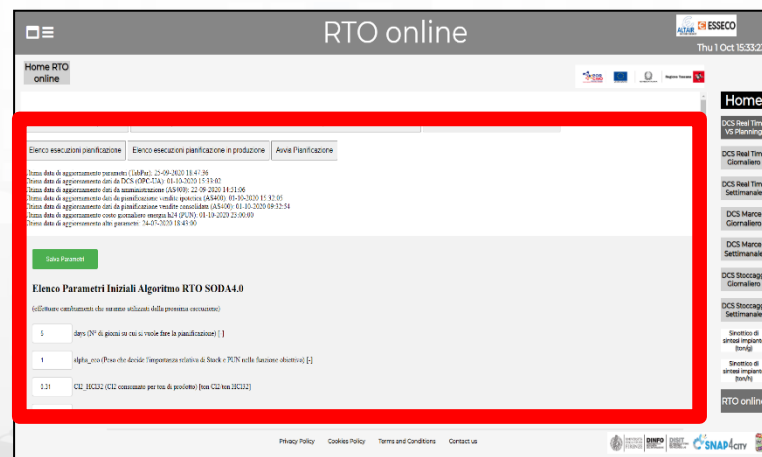


Function receives:

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

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

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



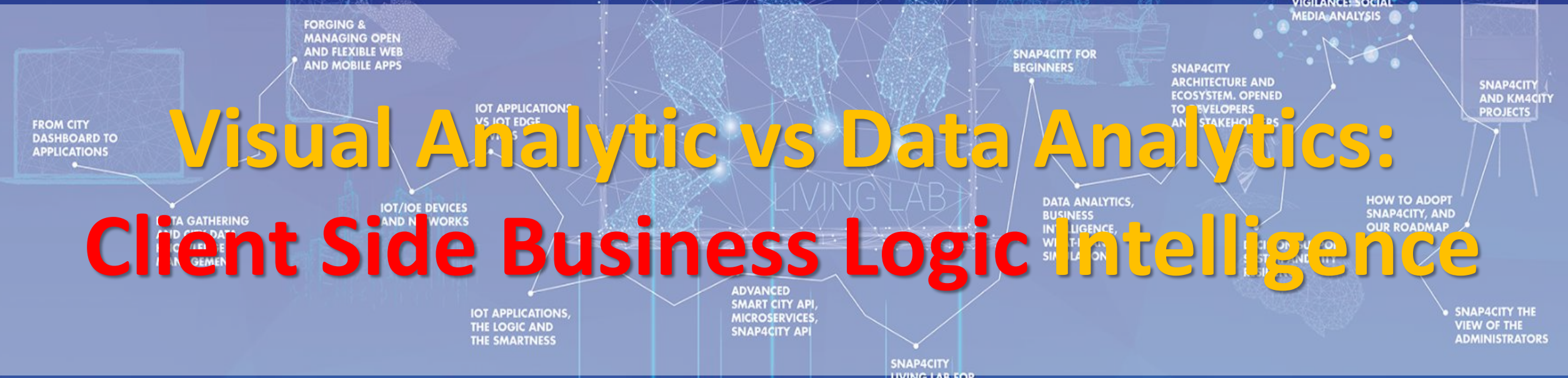
## Call on Browser:

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

Domain Prefix  
IoT App ID  
Your custom

TOP

# Visual Analytic vs Data Analytics: Client Side Business Logic Intelligence



*MS. 40*

# Client Side Business Logic

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

[Wdf](https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual)



Powered by  
**SNAP4Tech**

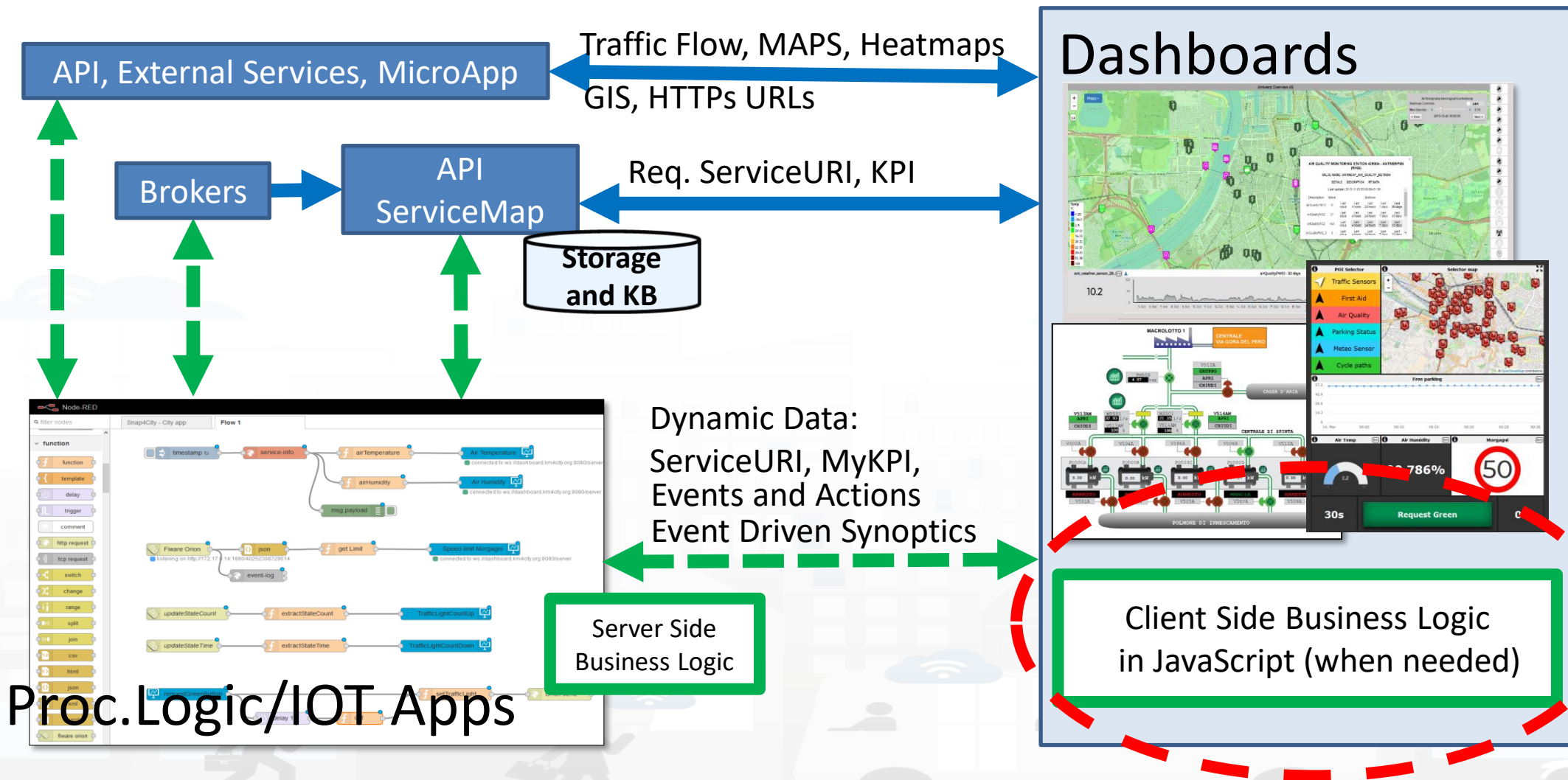
## Client-Side Business Logic Widget Manual

### 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/UC3TAO09EbNba8f2-u4vandu>

Coordinator: Paolo Nesi, [Paolo.nesi@unifi.it](mailto: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

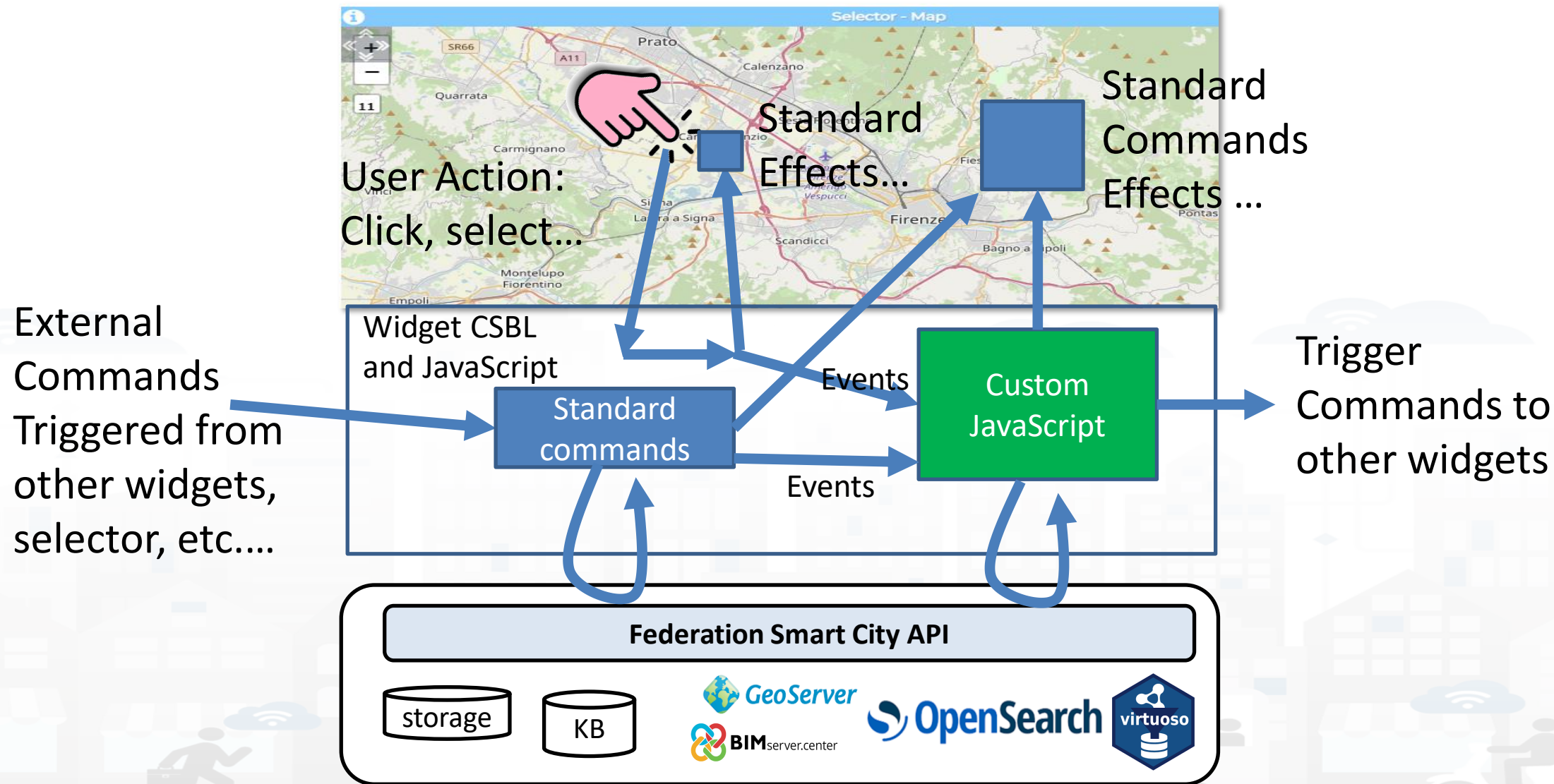
# How the Dashboards exchange data



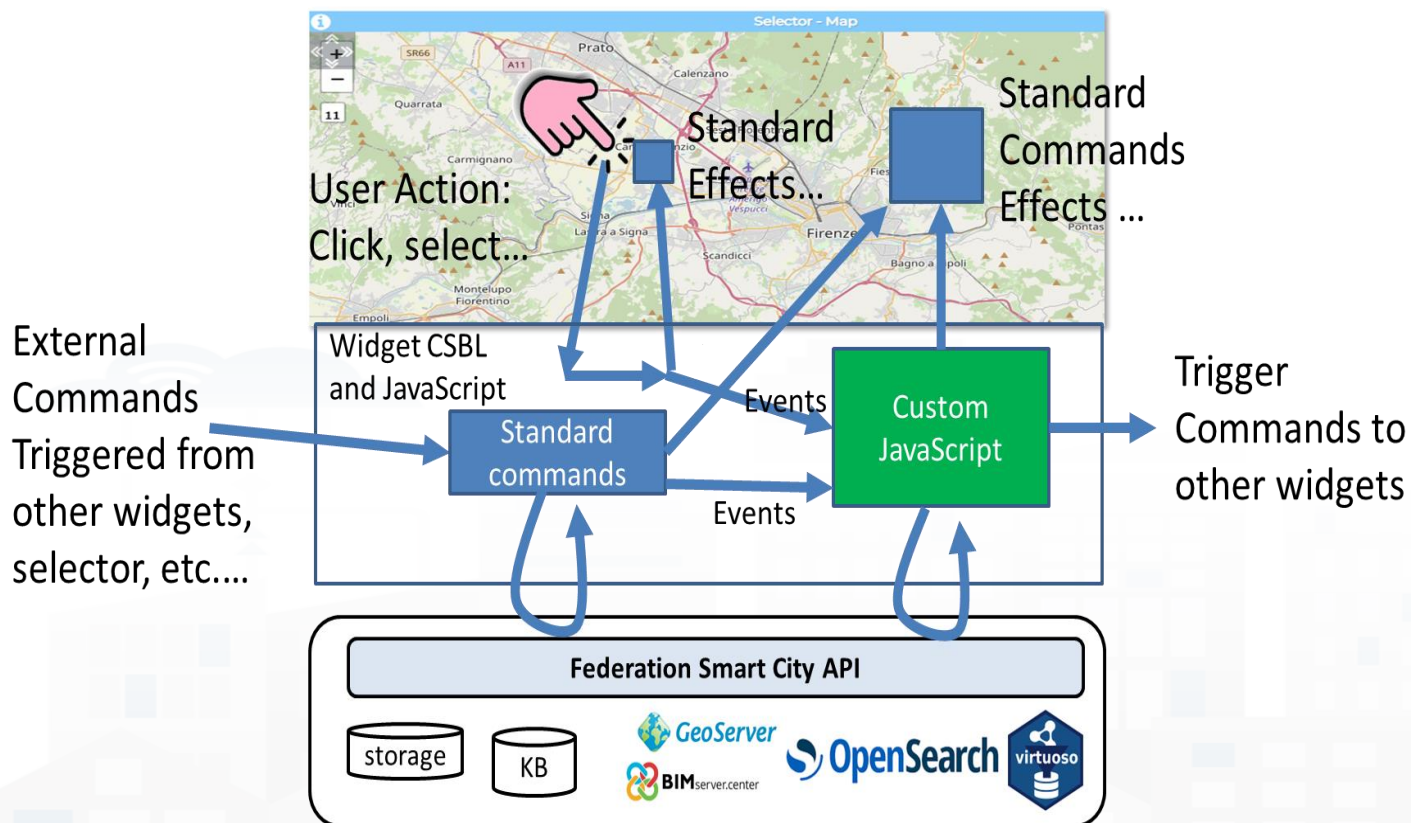
# Client Side Business Logic, CSBL

- 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](mailto: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

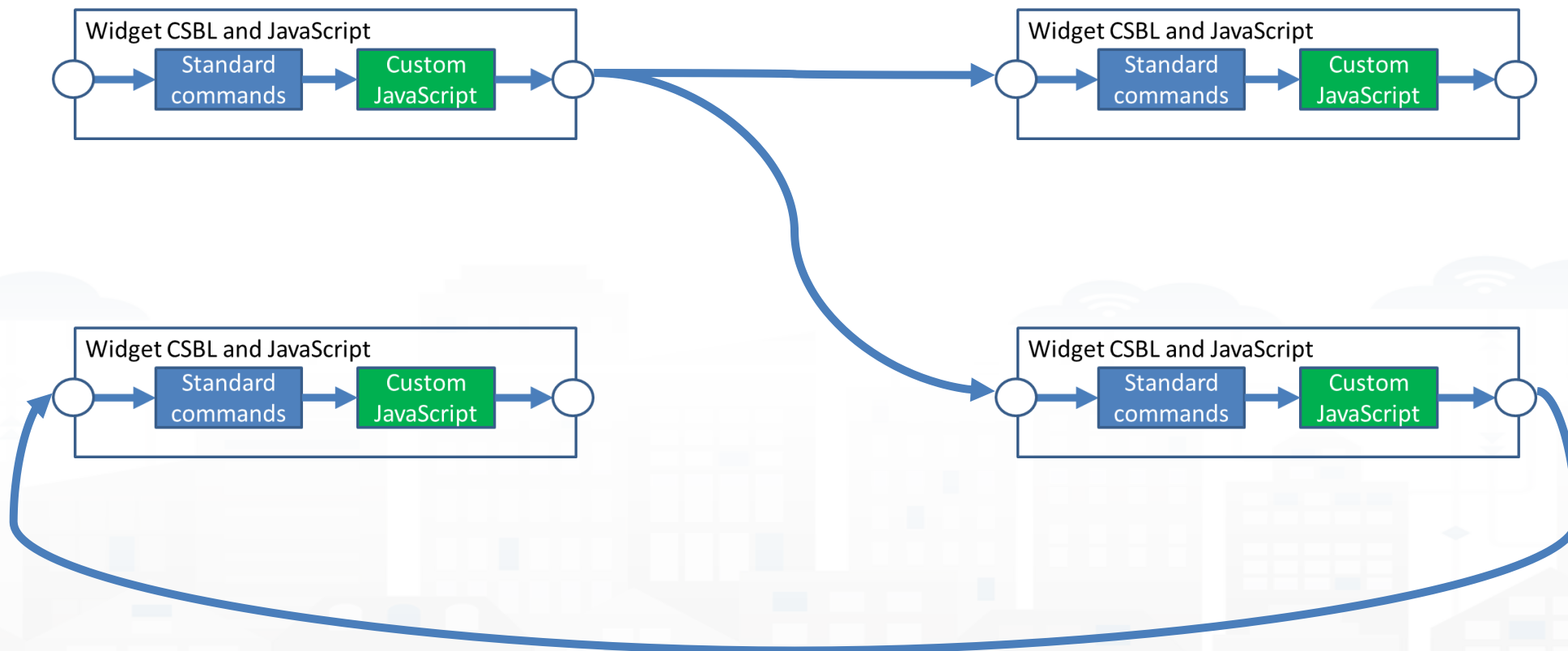
```
function execute ()
{
 Var e = JSON.parse(param)
 if (e.event == ".....") {

 }
 else if (e.event == ".....") {

 }
 else {

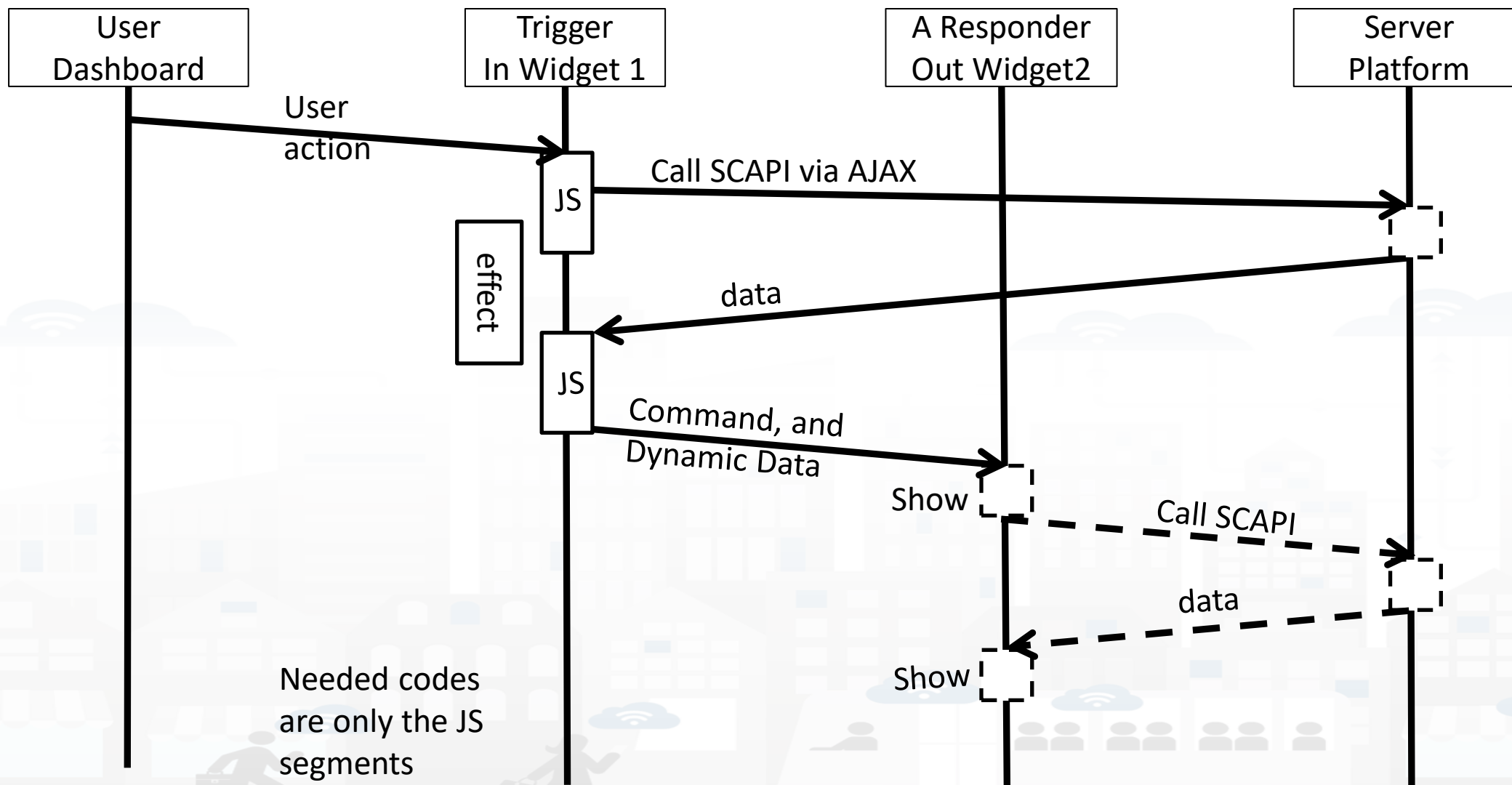
 $('body').trigger ({.....});
 }
}
```

# Composition

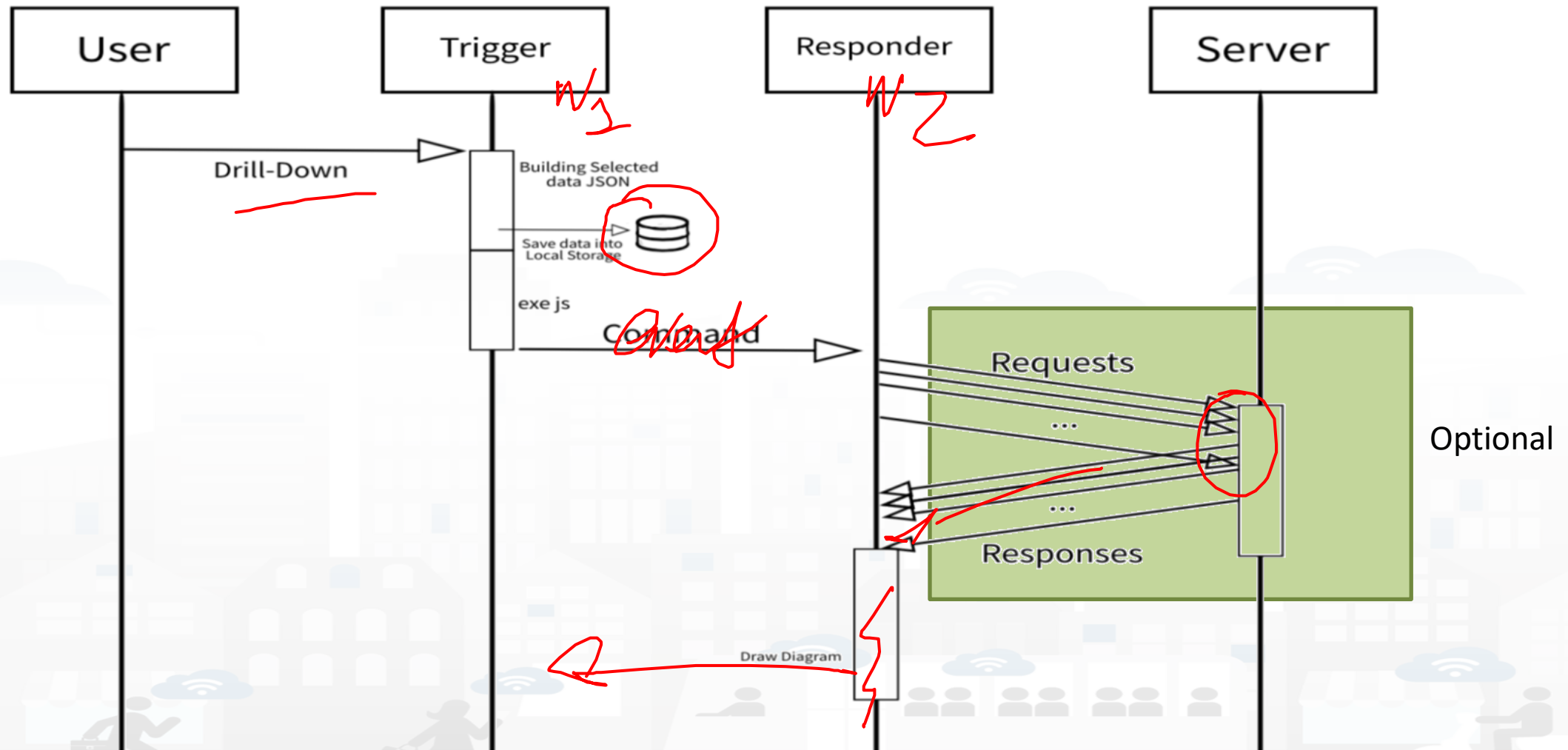


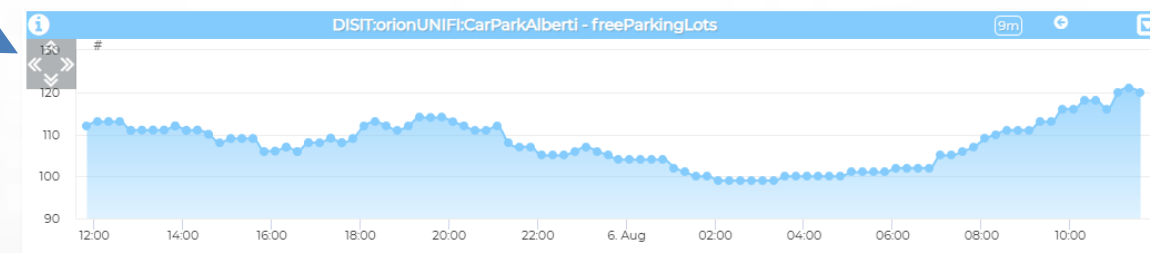
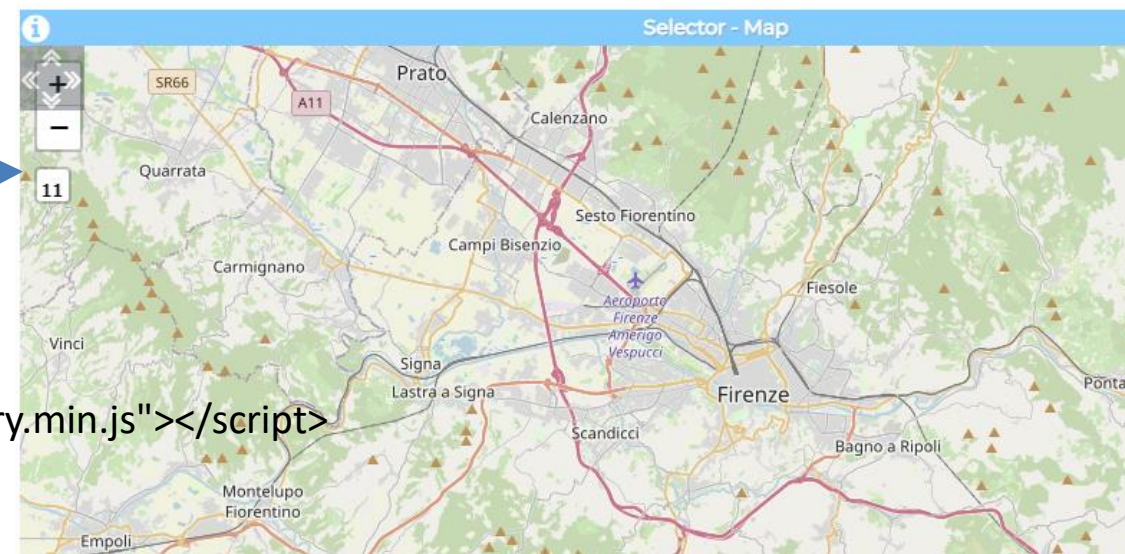


# User Actions, Triggers on Responder



# User Actions, Triggers on Responder





```
<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

The screenshot shows the configuration interface for an external content widget. It is divided into three main sections:

- Metric and widget choice:** Contains fields for Widget category (Data viewer), Metric (NR\_a2874619\_ebd078), Widget name (w\_NR\_a2874619\_ebd078\_2573), Widget type (widgetExte), Context, Widget link (https://rttvhd.snap4city.org/), and Metric description.
- Generic widget properties:** Contains fields for Title (Florence Da), Background color (rgba(2)), Content font size, Content font color, Header color (rgba(ε)), Header text color (rgba(2)), Period, Refresh rate (s), Height (41), Width (31), U/M, U/M position, Show header (Yes), and Font type (autosuggestion).
- Specific widget properties:** Contains fields for Widget mode (Web link), Enable fullscreen in new tab (Yes), Enable fullscreen in a popup (Yes), Zoom controls visibility (Always), Zoom factor (105%), and Zoom controls position (Top left).

At the bottom right, there is a section for "Enable CK Editor" (set to yes) and a text area for inserting HTML text. The text area contains the following code:

```

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

```

A blue arrow points from the "Specific widget properties" panel to the JavaScript code on the left side of the slide.

# Formalization of SSBL on In Widget More Options

- CK EDITOR

Modify widget

## Metric and widget choice

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

## Generic widget properties

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

## Specific widget properties

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

**Enable CK Editor** yes

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

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

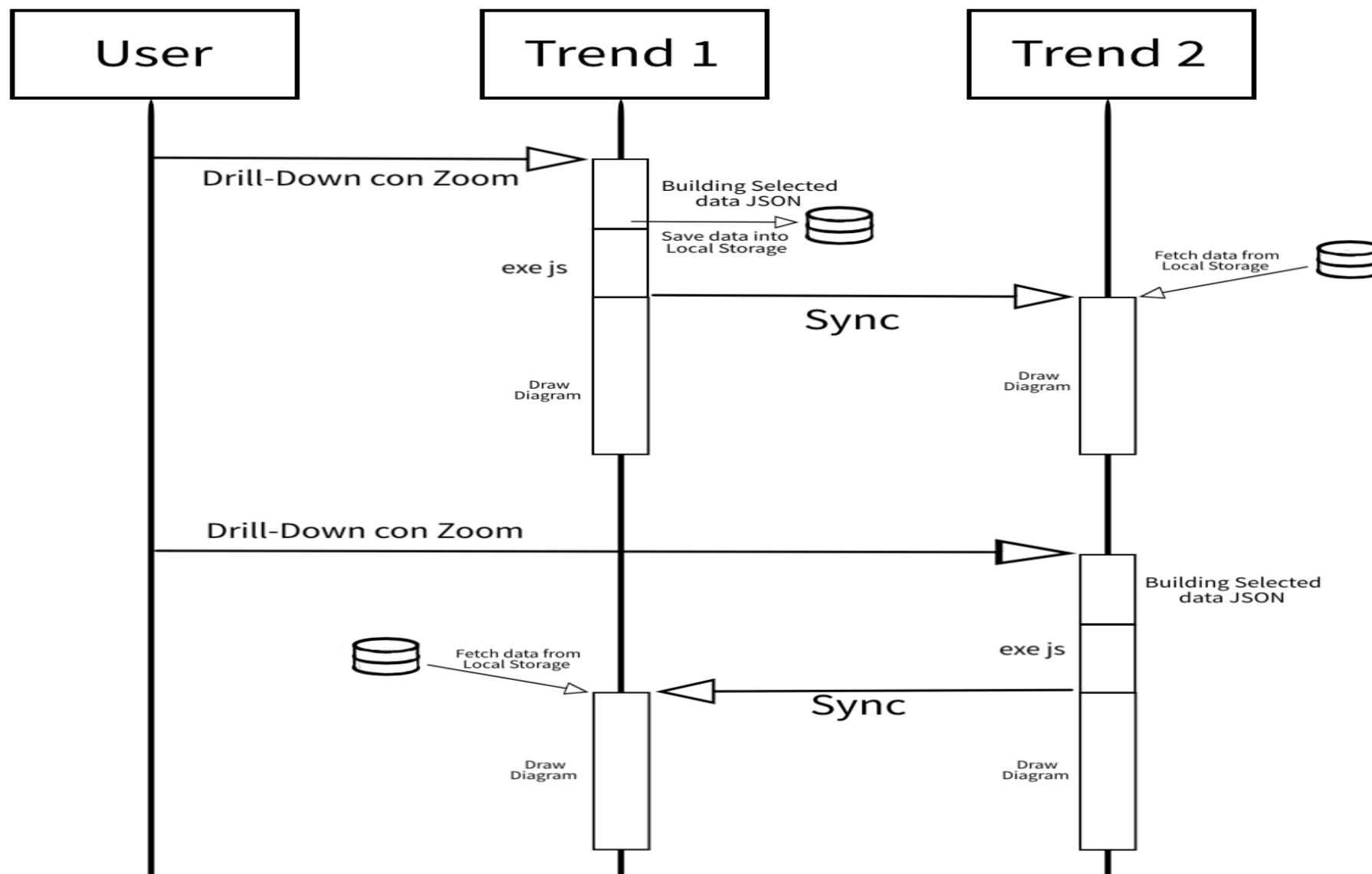
# Trigger map

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

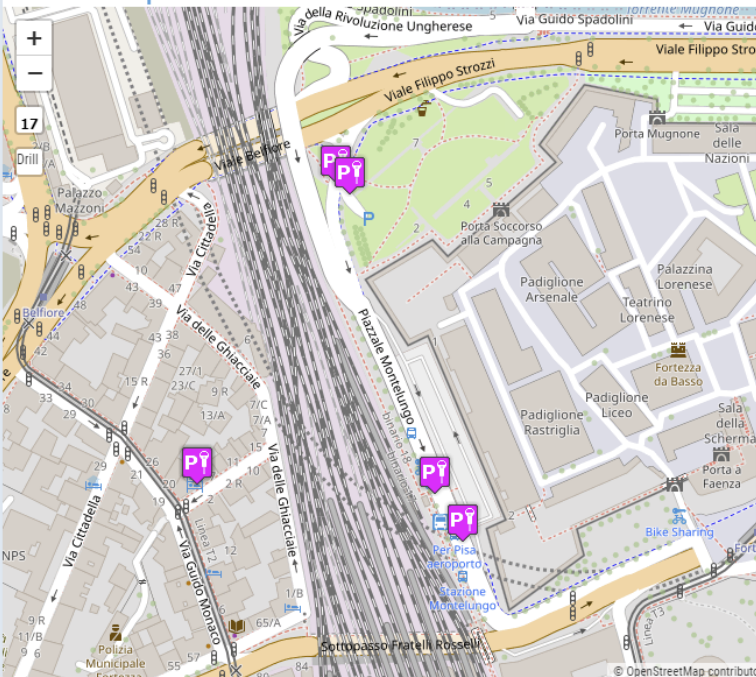
TOP

# *Client Side Business Logic Example*





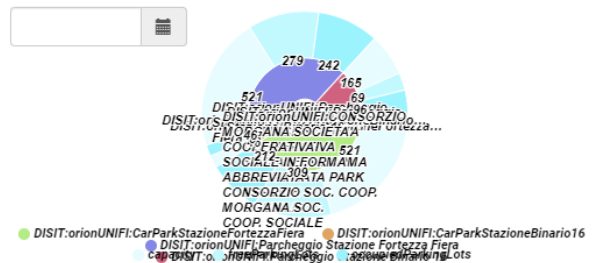
### Selector - Map



### Selector

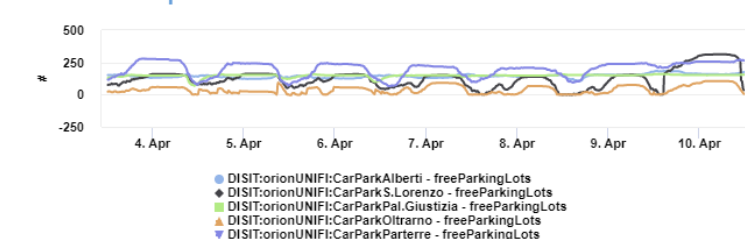
- Car\_park
- metrotrafficsensor
- Air\_quality\_monitoring\_station
- Weather\_sensor

### Pie Chart

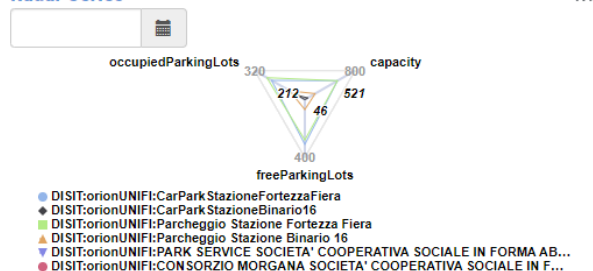


9m

### Time Trend Comparison

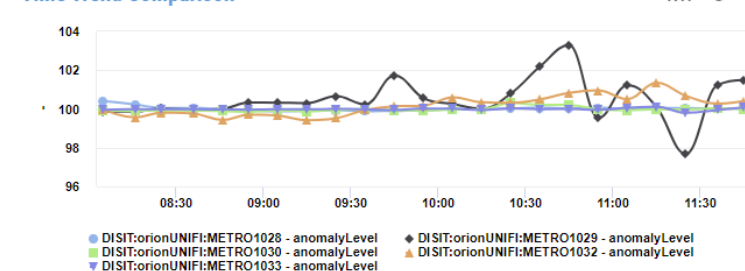


### Radar Series

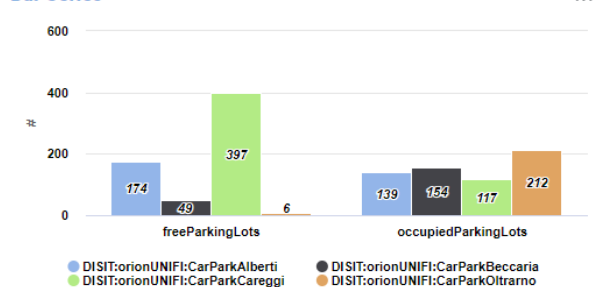


4m

### Time Trend Comparison

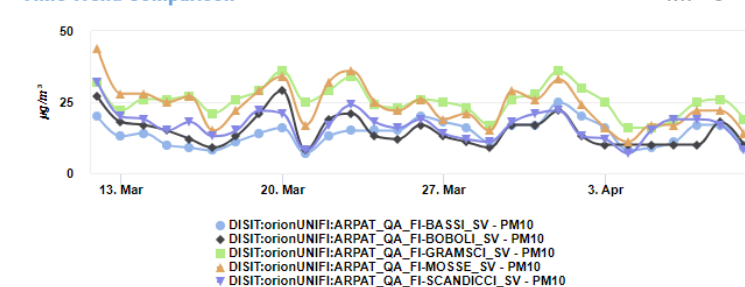


### Bar Series



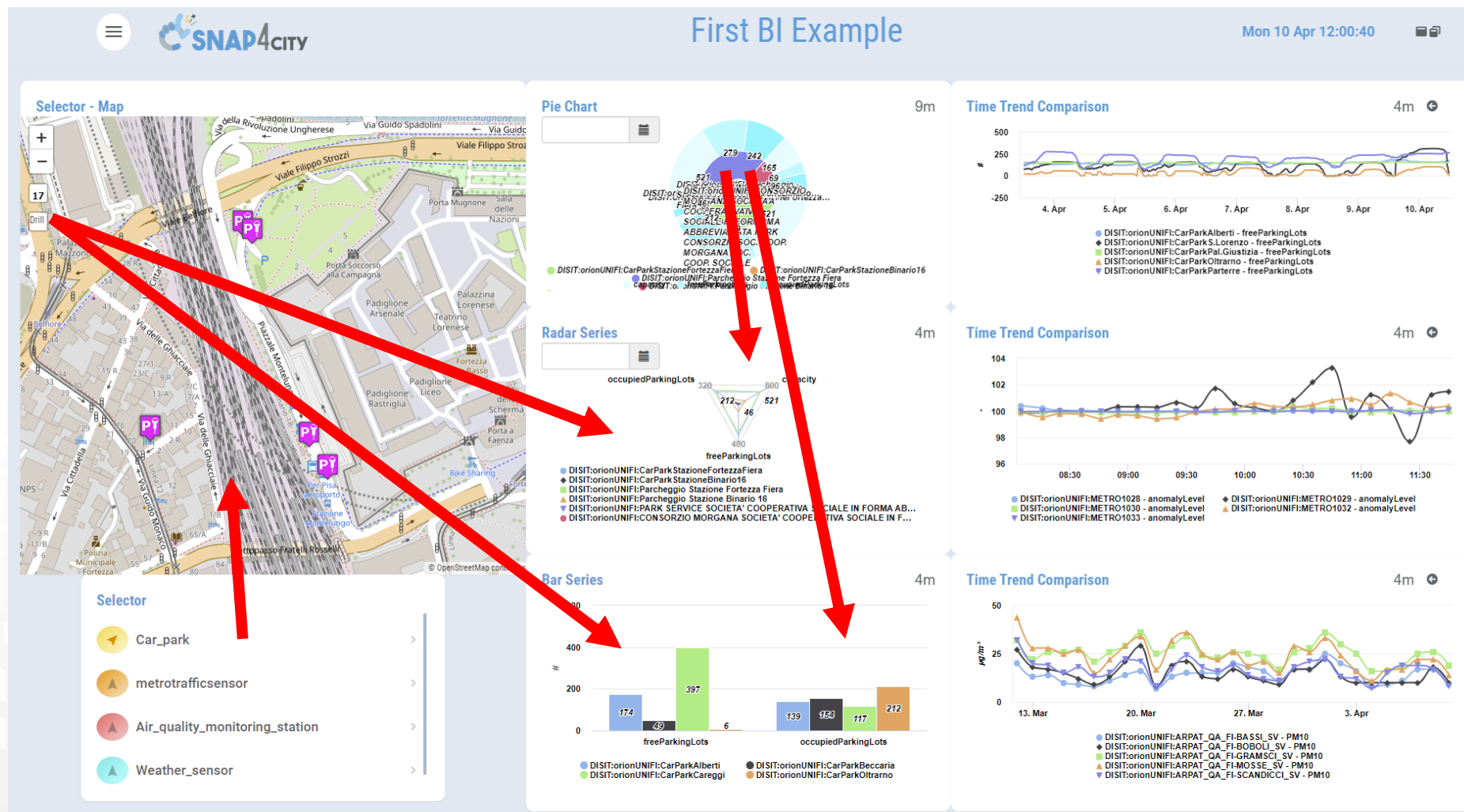
4m

### Time Trend Comparison



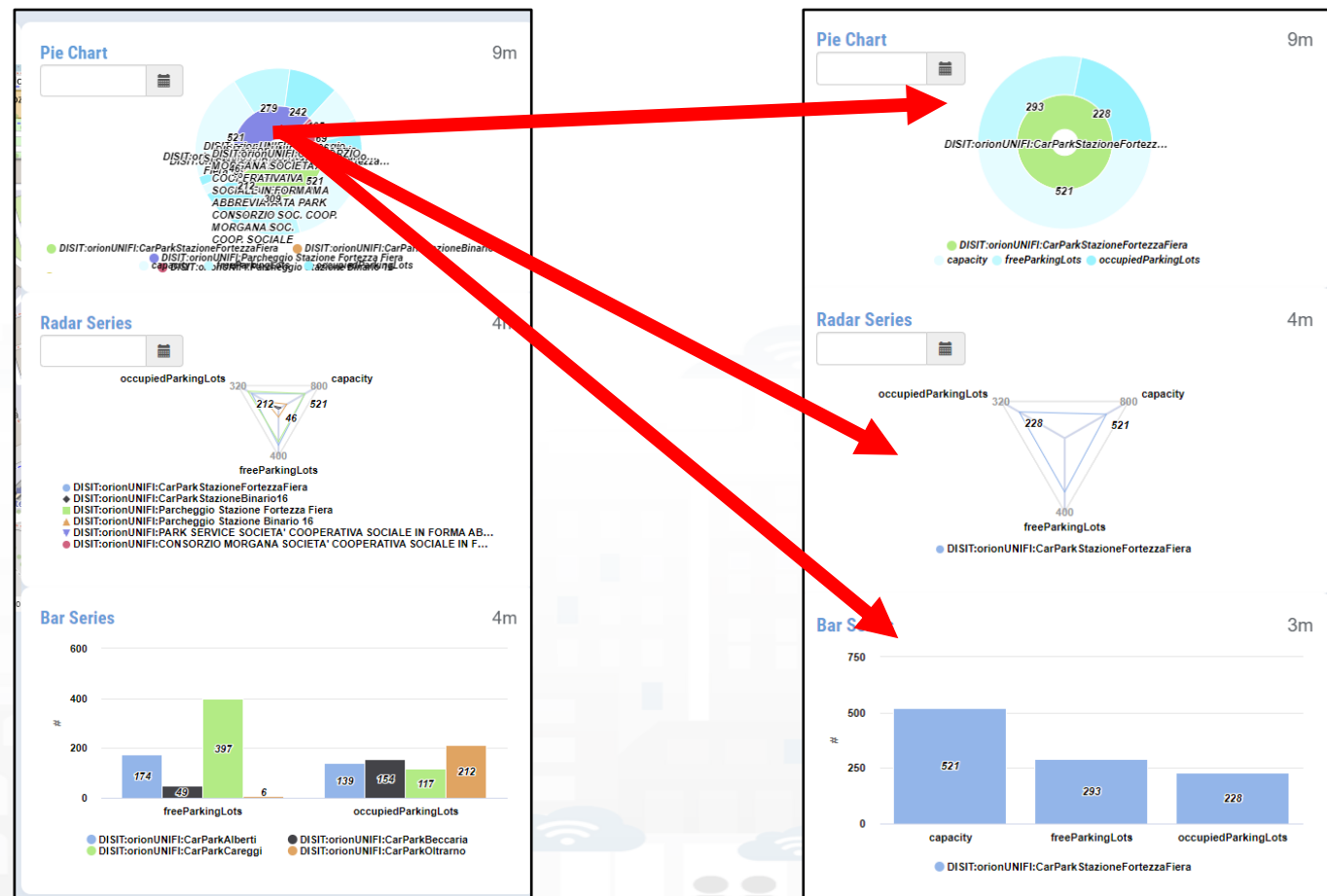
# Example: From Map to Graphs (spatial drill down)

- 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

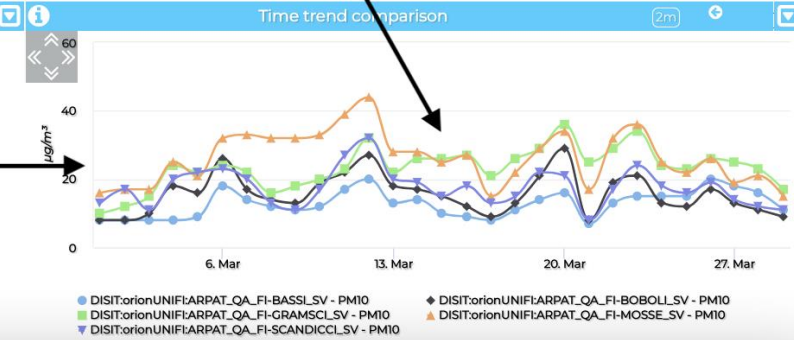
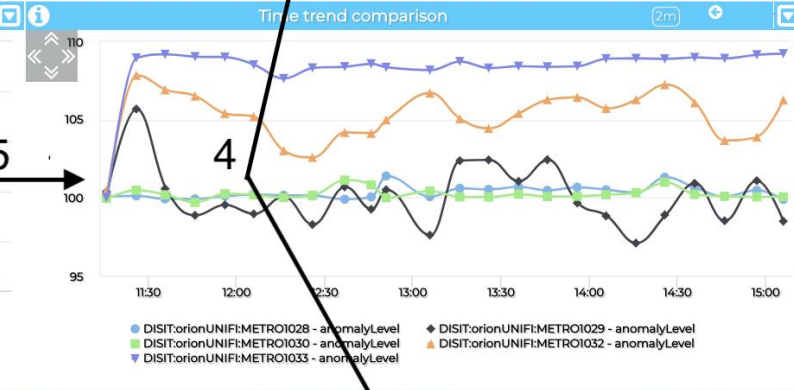
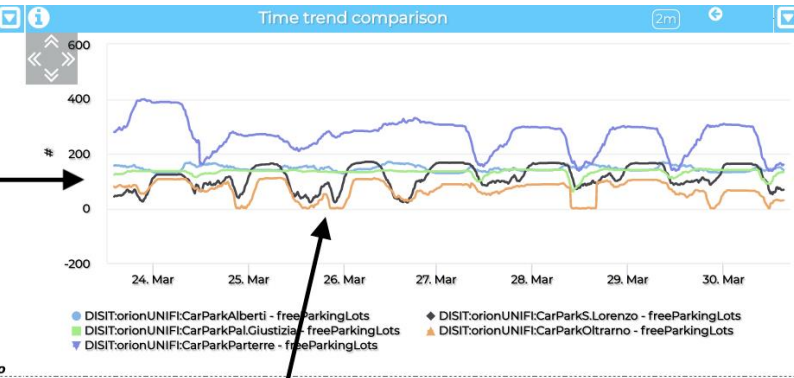
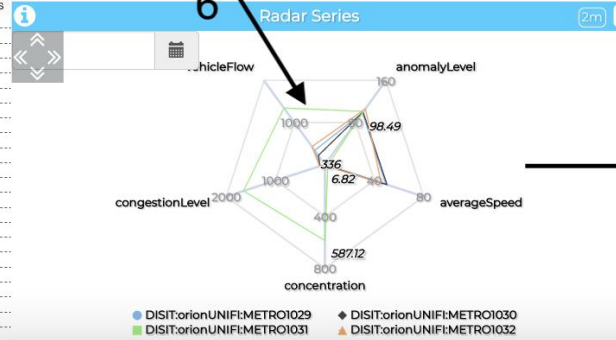
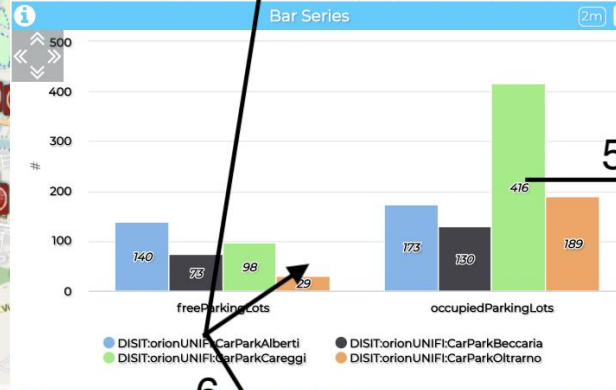
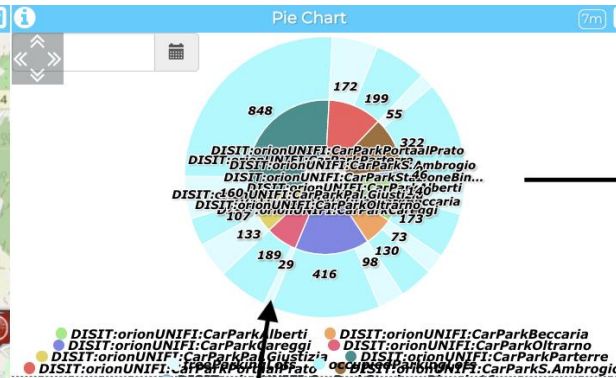
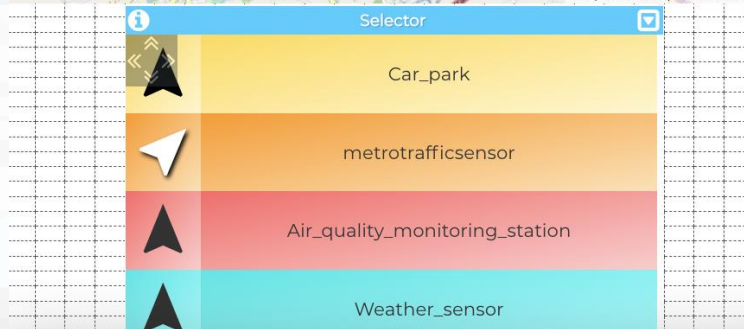
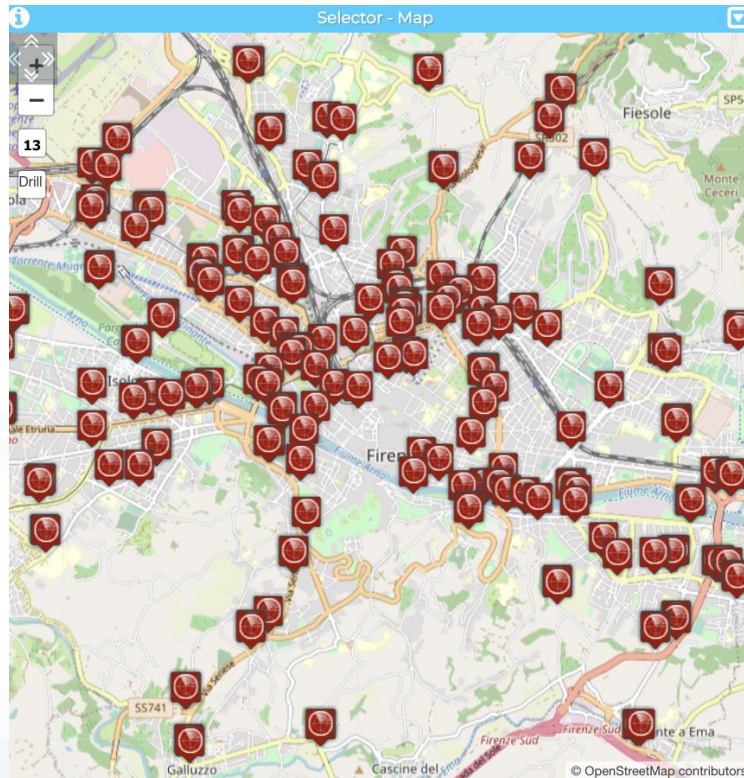


1) Click on the Legend of Bar Series

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



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.



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 SURJ, metric names and/or values, and show their corresponding values on a Pie Chart graph.
widgetRadarSeries	Receive a JSON containing a list of SURJ, metric names and/or values, and show their corresponding values on a Radar/Kiviati graph.
widgetBarSeries	Receive a JSON object containing a list of SURJ, metric names and/or values, and show their corresponding values on a Bar graph.
widgetSingleContent	<ul style="list-style-type: none"> <li>Receive a SURJ and a metric name, or a value, or a text string, and show the corresponding value.</li> <li>Receive and show a HTML/JS page</li> </ul>
widgetSpeedometer	Receive a SURJ and a metric name, or a value, and show the corresponding value on a speedometer graph.
widgetGaugeChart	Receive a SURJ and a metric name, or a value, and show the corresponding value on a gauge graph.
widgetTimeTrend	Receive a SURJ 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 SURJ, 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 SURJ, 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 SURJ 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 SURJ 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 SURJ 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

*Examp. X*

# 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.

TOP

# *Develop: Client Side Business Logic Dashboards with Synoptics*

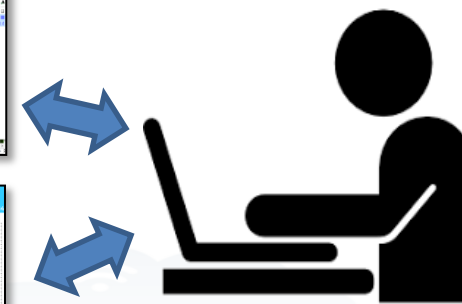


# Custom Widget / Synoptic / PIN Development

Inkscape editor on your computer



Create, save a Custom Widget in SVG



Create, save, load, delegate, grant access

IOT Applications

Knowledge Base, Km4City

Knowledge and Storage Data from the Field and City

SVG Symbols Collection

0.04 SMC

SVG

SVG

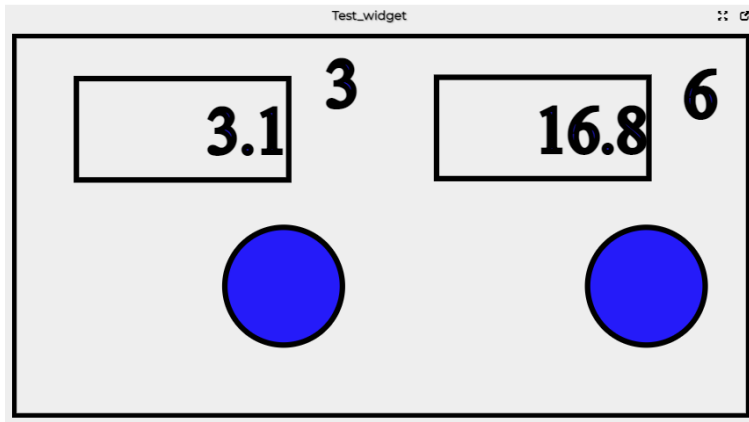
Public Dashboard Collection

My Own Dash/App

Dashboard Editor

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(.....)
```

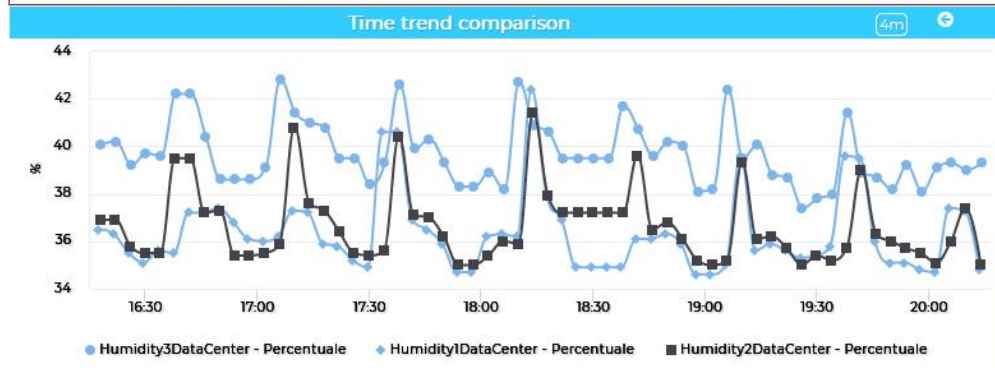
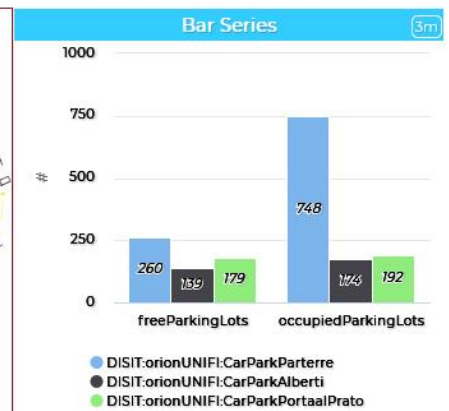
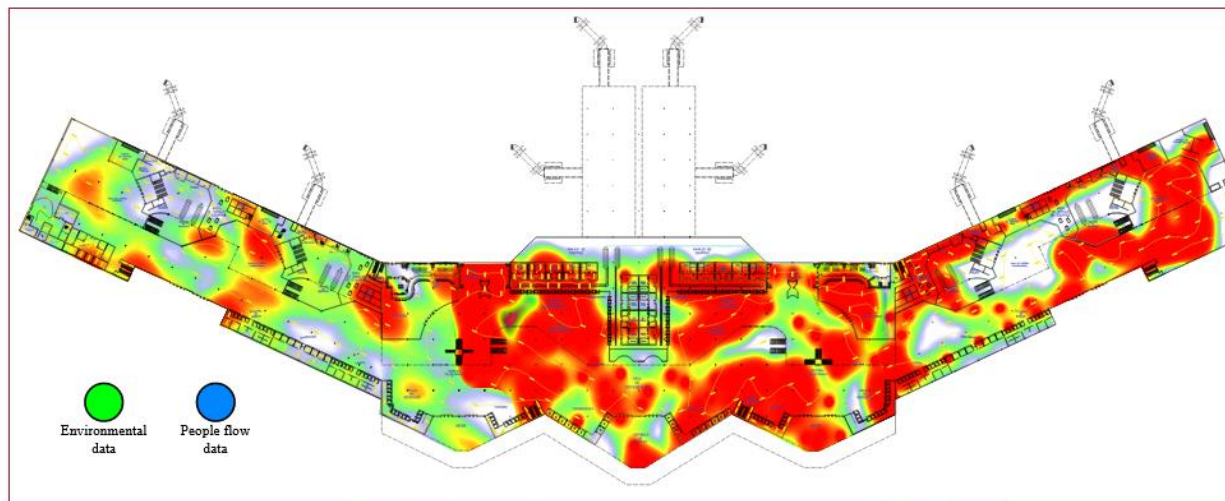
```
TextBoxWrite4(.....)
```

-- 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



button1

button1



TOP

# Design and Control of Smart Applications

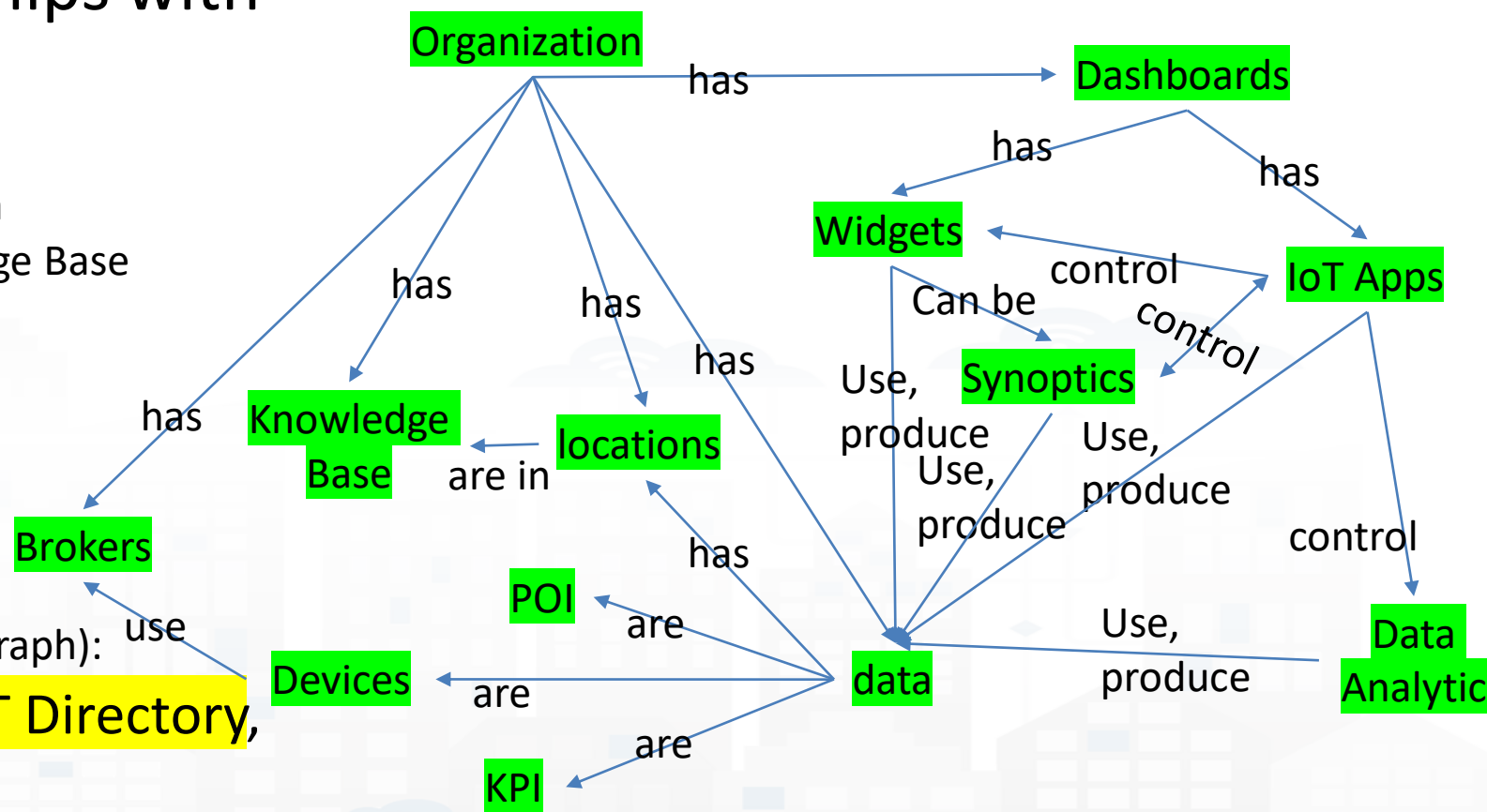
only for user with RootAdmin role  
partially accessible also for all Dashboard owners



# 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

testaxisvenaria: Public - (DISIT)

Edit Management Clone Delete

## Management

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

### Monitoring Cross Road Venaria - (AXIS Camera)

Change ownership

New owner username

*New owner username can't be empty*

Confirm

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

**Management**

Ownership Visibility Delegations Group Delegations Accesses Trends Structure Organization Thumbnail

Link to Graph

Dashboard Hierarchy

Dashboard: Monitoring Cross Road Venaria - (AXIS Camera)

- Widget: Trajectories legenda - (*widgetButton*)
- Use Data:
- Widget: Number of Trajectories per hour - (*widgetCurvedLineSeries*)
- Use Data:
  - sensor: CrossVenaria2VehicleFlowTrajectoriesV2
  - Query: <http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/CrossVen...>
  - Link to Data Inspector
  - Link to Graph log
  - Link to Servicemap
- Widget: In out computations and totals per hour - (*widgetCurvedLineSeries*)
- Use Data:
  - sensor: VenariaConteggio
  - Query: <http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/VenariaC...>
  - Link to Data Inspector
  - Link to Graph log
  - Link to Servicemap
- Widget: Venaria Street Cross - per hour - (*widgetExternalContent*)
- Use Data:
  - Query: <https://www.snap4city.org/synoptics/v2/synoptic.html?id=135648299>
  - Link to Data Inspector (root)
  - Link to Graph log

# For All Dashboard owners: Graph and Structure

- Go on Dashboard Management

**Management**

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

Link to Graph

Dashboard Hierarchy

Dashboard: - Energy -

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

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

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

## LOGraph

Linked Open Graph

Shows: 47 Endows: 31 Relations: 16

Identifier: <http://www.dinfo.org/km4city/resource/001/CarCount>

Image:

Infer: no other informations

SPARQL Query: 

```
graph TD; subgraph query; direction TB; Q1[SELECT ?subject ?property ?object]; end;
```

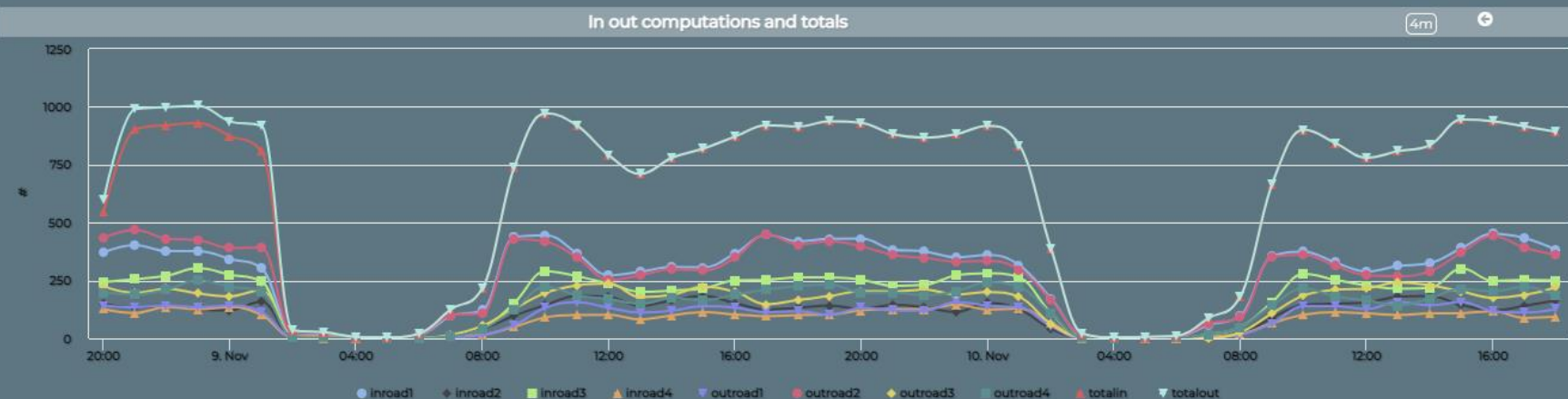
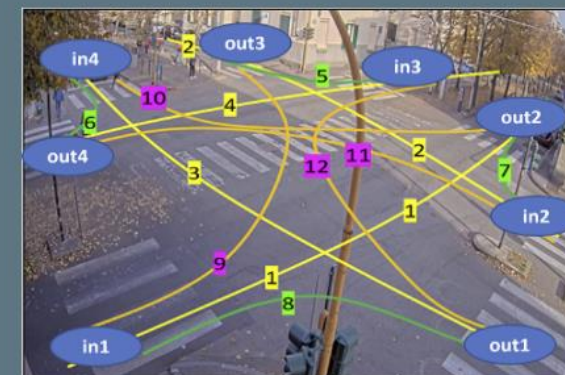
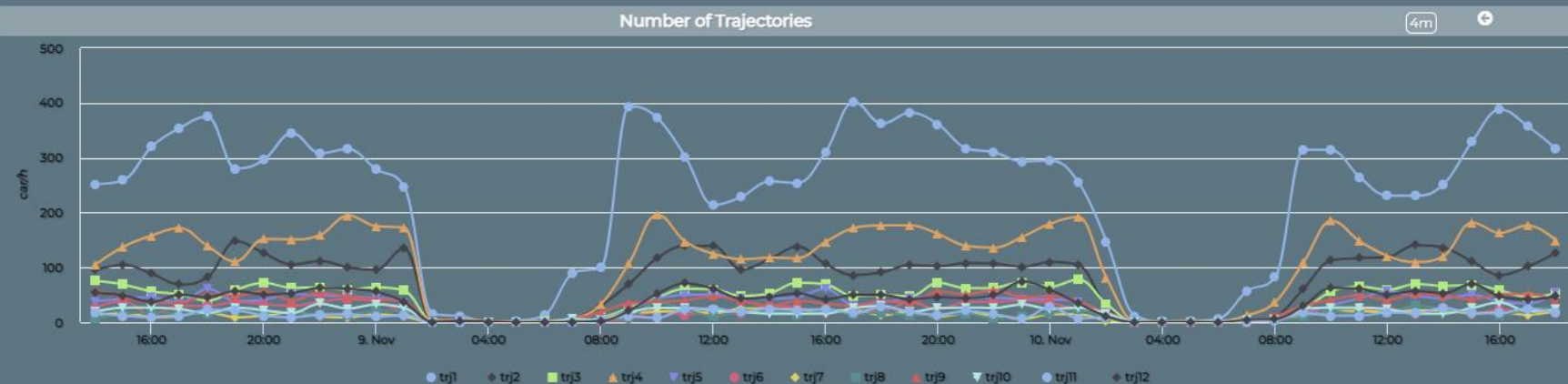
Type of relations

<input type="checkbox"/> Select all	<input type="checkbox"/> Deselect all	<input type="checkbox"/> Invert	<input type="checkbox"/> Hide all inverse
<input checked="" type="checkbox"/> expropriate	<input type="checkbox"/> hasDepiction		
<input checked="" type="checkbox"/> hasAttribute	<input checked="" type="checkbox"/> hasDashboard		
<input checked="" type="checkbox"/> hasSystemCapability	<input checked="" type="checkbox"/> hasWidget		
<input checked="" type="checkbox"/> implements	<input checked="" type="checkbox"/> observes		
<input type="checkbox"/> owl:sameAs	<input checked="" type="checkbox"/> rdf:type		
<input checked="" type="checkbox"/> rdfs:isA	<input checked="" type="checkbox"/> useData		
<input checked="" type="checkbox"/> usedByWidget			

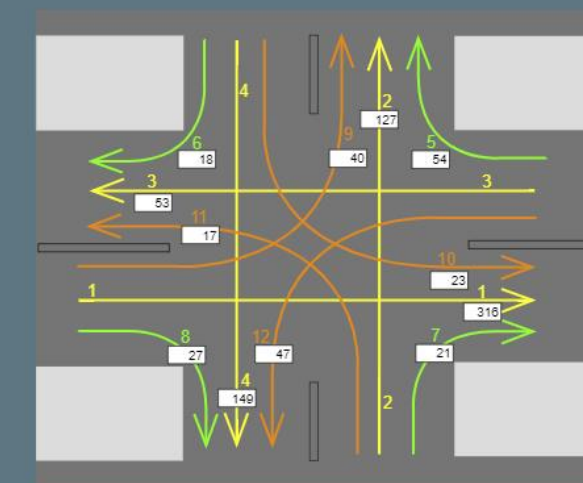


## Monitoring Cross Road Venaria - (AXIS Camera)

Wed 10 Nov 18:50:53



Venaria Street Cross - Synoptic



<https://www.snap4city.org/dashboardSmartCity/view/index.php?iddashboard=MzI5Ng==>

Linked Open Graph

Shown: 47  
Entities: 31  
Relations: 16

Show Endpoints Show User Status Hide Relations

Embed

CarCount Close

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

**Image:**

**Info:**

no other informations

**Sparql Query:**

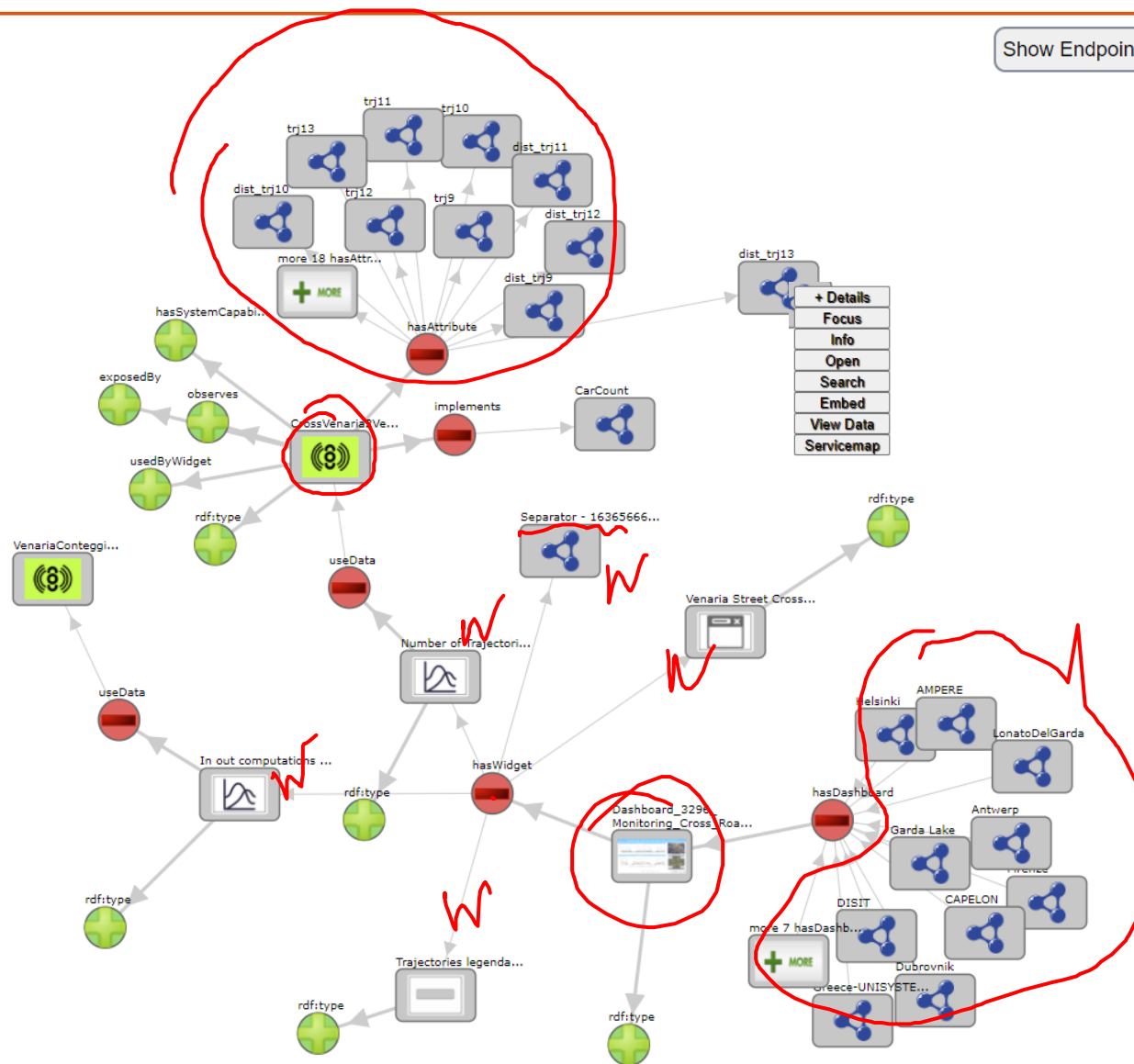
ENDPOINT:  
http://virtuoso-kb:8890/sparql

QUERY:  
SELECT ?subject ?property ?object

**Type of relations**

Select all Deselect all Invert  Hide all inverse

<input checked="" type="checkbox"/> exposedBy	<input type="checkbox"/> foaf:depiction
<input checked="" type="checkbox"/> hasAttribute	<input checked="" type="checkbox"/> hasDashboard
<input checked="" type="checkbox"/> hasSystemCapability	<input checked="" type="checkbox"/> hasWidget
<input checked="" type="checkbox"/> implements	<input checked="" type="checkbox"/> observes
<input type="checkbox"/> owl:sameAs	<input checked="" type="checkbox"/> rdf:type
<input checked="" type="checkbox"/> rdfs:seeAlso	<input checked="" type="checkbox"/> useData
<input checked="" type="checkbox"/> usedByWidget	



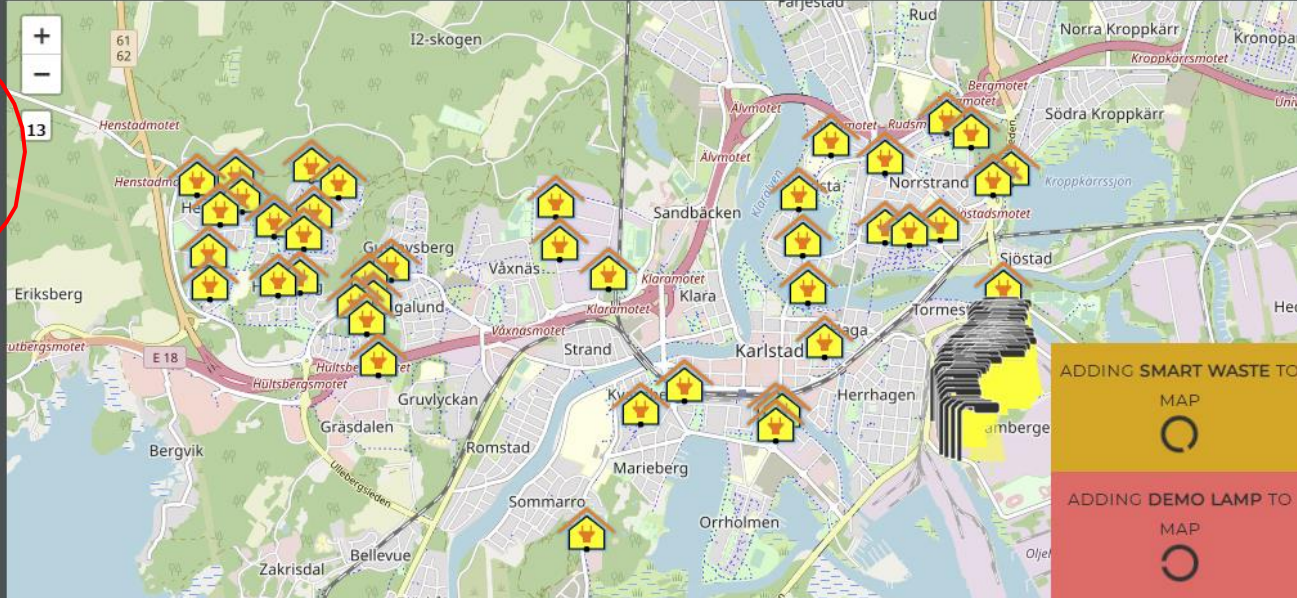


## Karlstad - Capelon

CAPELON


Sun 28 Nov 20:02:16


-  Cabinet
-  Smart Light
-  Demo Lamp
-  Smart Waste

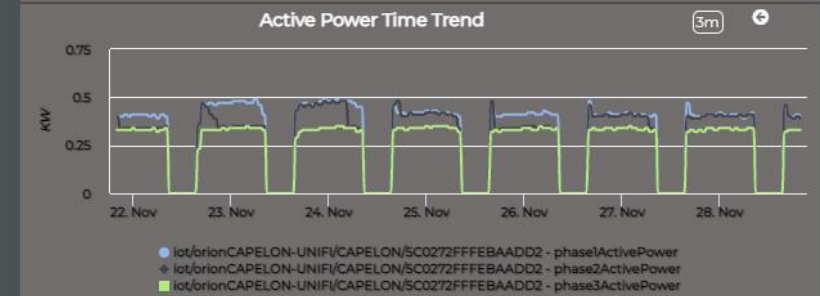
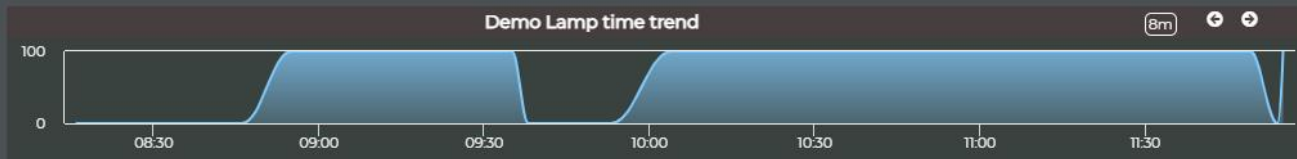
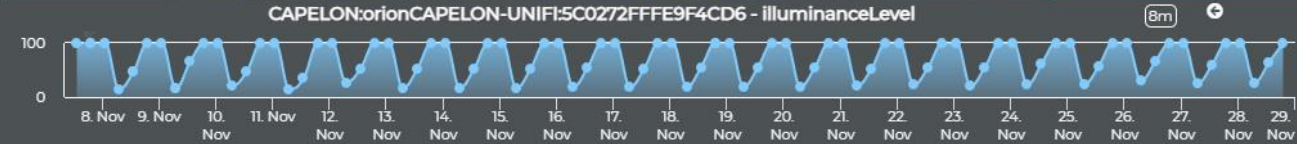


ADDING SMART WASTE TO MAP 

ADDING DEMO LAMP TO MAP 

 Lamp ON

 Lamp OFF





### Linked Open Graph

Shown: 45  
Entities: 31  
Relations: 14

Select a SPARQL endpoint:

Examples:  
uri:

Multiple endpoint search

---

**Your data**

sparql endpoint: (optional)

uri:

Multiple endpoint search

---

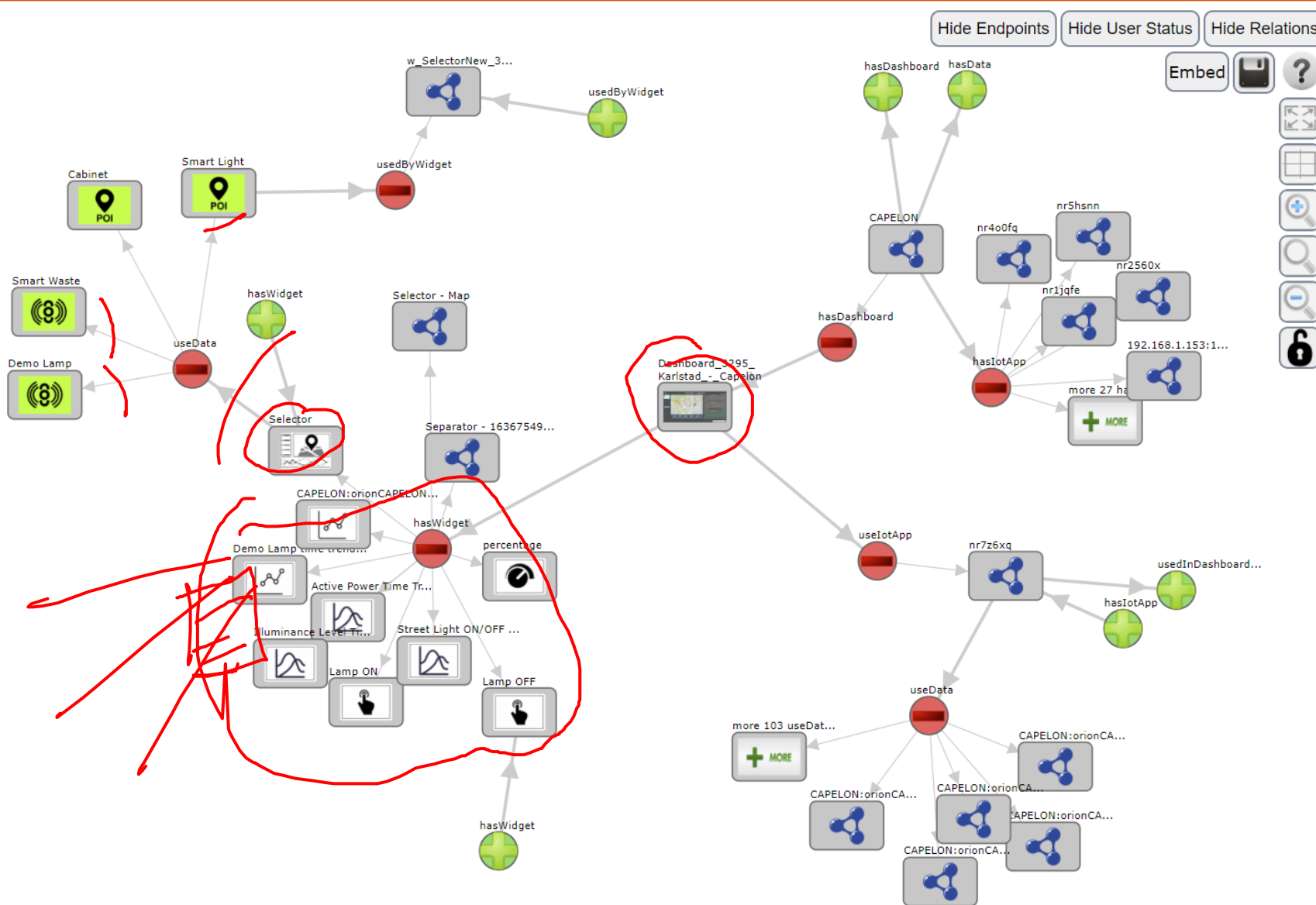
**Status**

Requests:

**Type of relations**

Select all  Deselect all  Invert  Hide all inverse

<input type="checkbox"/> foaf:depiction	<input checked="" type="checkbox"/> hasDashboard
<input checked="" type="checkbox"/> hasData	<input checked="" type="checkbox"/> hasIoTApp
<input checked="" type="checkbox"/> hasWidget	<input type="checkbox"/> owl:sameAs
<input type="checkbox"/> rdfs:type	<input checked="" type="checkbox"/> rdfs:seeAlso
<input checked="" type="checkbox"/> useData	<input checked="" type="checkbox"/> useIoTApp
<input checked="" type="checkbox"/> usedByWidget	<input checked="" type="checkbox"/> usedInDashboard



<https://www.snap4city.org/dashboardSmartCity/view/index.php?iddashboard=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

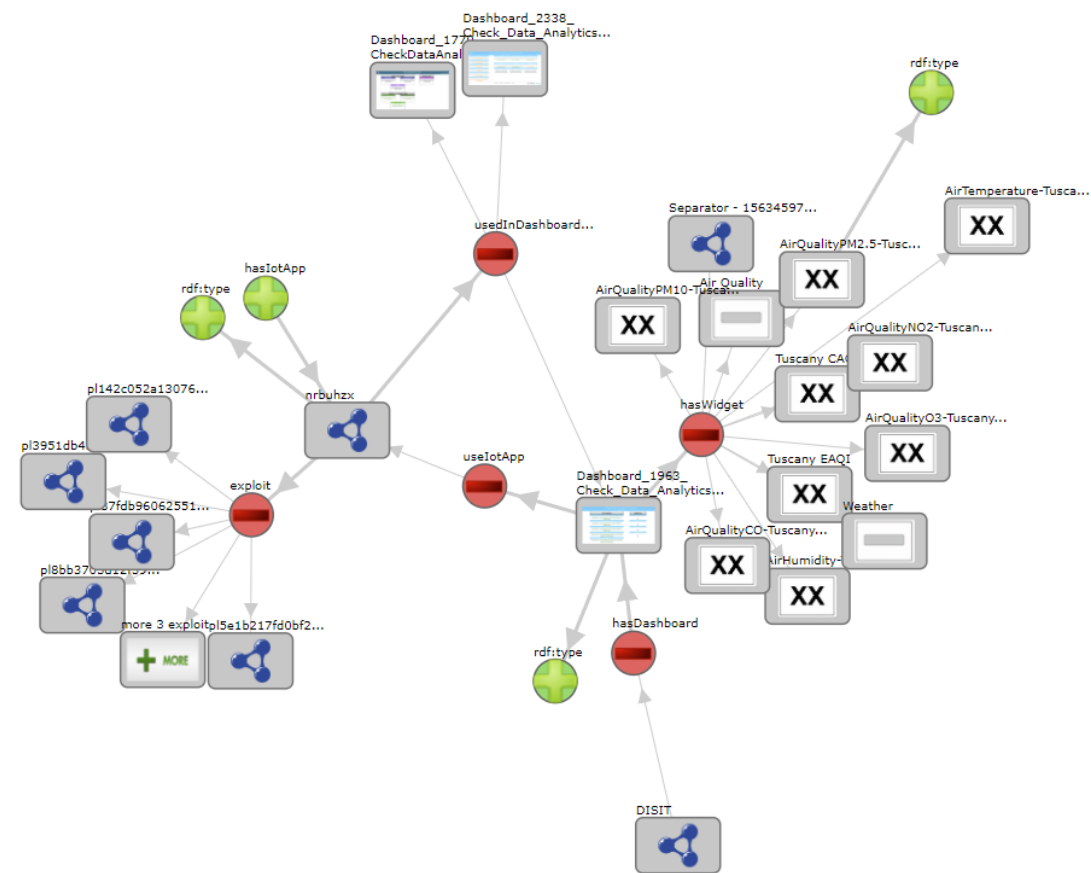
AirTemperature-Tuscany

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

DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT

FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS

IOT APPLICATIONS VS. IT USE CASES

IOT/IOE DEVICES AND NETWORKS

IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS

ADVANCED SMART CITY API, MICROSERVICES, SNAP4CITY API

SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK

SNAP4CITY FOR BEGINNERS

SNAP4CITY ARCHITECTURE AND ECOSYSTEM, OPENED TO DEVELOPERS AND PARTNERS

DATA ANALYTICS, BUSINESS INTELLIGENCE, WHAT IS AND IS NOT SMART

TWITTER VIGILANCE, SOCIAL MEDIA ANALYSIS

HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP

SNAP4CITY AND KM4CITY PROJECTS

SNAP4CITY THE VIEW OF THE ADMINISTRATORS

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

<https://www.snap4city.org/577>

- Data Streams from partecipanti, 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/577>



*On Line Training Material (free of charge)*

<https://www.snap4city.org/944>

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

Video1								
Video2								
Video3								
Video4				none		none	none	none


# 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>

[Switch To New Layout \(Beta\)](#)User: paolo.disit, Org: DISIT  
Role: AreaManager, Level: 3[LOGOUT](#) [Home](#) / [Tutorials and Videos](#) / Welcome: how to start using Snap4City for beginners

# 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](mailto:paonesi@gmail.com) !

[+ Share / Save](#)    ...[Add to your favorites](#)

Innovations



Interoperability



Installations



What People say



Mobile Apps



IOT Devices



IOT Applications



Data Analytics



Dashboards



Living Lab



Smart City API



Smart City Ontology



Work with Us



Articles



SNAP4CITY on EUROPEAN OPEN SCIENCE CLOUD MARKETPLACE



SNAP4CITY HACKATHON



INDUSTRY 4.0 Snap4Industry



Snap4Home

- 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](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](#)

Username: paolo.disit

## Search

**Training on Tools and Platform**Powered by [www.km4city.org](http://www.km4city.org)  

## Organization Groups

DISIT

- Developer
- Operativo

## Updates on Tools

Training Course Snap4City - 2023 Edition [new](#)  
drupaladminSnap4City Newsletter of April 2023 [new](#)  
roottooladmin1[My Snap4City.org](#)[Tour Again](#)[www.snap4solutions.org](#)[Dashboards \(Public\)](#)[Dashboards of My Organization](#)[My Dashboards in My Organization](#)[My Data Dashboard Dev Kibana](#)[Extra Dashboard Widgets](#)[Data Management, HLT](#)[Knowledge and Maps](#)[Processing Logics / IOT App](#)[Entity Directory and Devices](#)[Resource Manager](#)[Development Tools](#)[Management](#)[Decision Support Systems](#)[Deploy and Installation](#)[Help and Contacts](#)[Documentation and Articles](#)[My Profile](#)[Km4City portal](#)[DISIT Lab portal](#)



Dashboards (Public)



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



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.

Username: paolo.disit

## Search

Search input field with dropdown menu showing '-Any-'

WHAT IS Snap4City | LATEST NEWS | SELECT for Cities 1° Place award to SNAP4CITY | Snap4City Training on Tools and Platform | Tutorials | Scenarios | Organizations

SMARTCITY EXPO WORLD CONGRESS 15 - 17 NOVEMBER 2022 BARCELONA & ONLINE GET YOUR PASS

Flyer | Flyer | DATA ANALYTICS ARTIFICIAL INTELLIGENCE

Innovations | Interoperability | Installations

What People say | Mobile Apps | IOT Devices | IOT Applications | Data Analytics | Dashboards | Living Lab | Smart City API | Smart City Ontology | Work with Us

Articles | SNAP4CITY on EUROPEAN OPEN SCIENCE CLOUD MARKETPLACE | SNAP4CITY HACKATHON | INDUSTRY 4.0 | Snap4Industry | Snap4Home



Powered by www.km4city.org



## Organization Groups

- DISIT
  - Developer
  - Operativo

- 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/DBL\\_SNAP4SOLL.pdf](https://www.snap4city.org/download/video/DBL_SNAP4SOLL.pdf)

# 2022 booklets

- Snap4City



[https://www.snap4city.org/download/video/DPL\\_SNAP4CITY\\_2022-v02.pdf](https://www.snap4city.org/download/video/DPL_SNAP4CITY_2022-v02.pdf)

- Snap4Industry



[https://www.snap4city.org/download/video/DPL\\_SNAP4INDUSTRY\\_2022-v03.pdf](https://www.snap4city.org/download/video/DPL_SNAP4INDUSTRY_2022-v03.pdf)

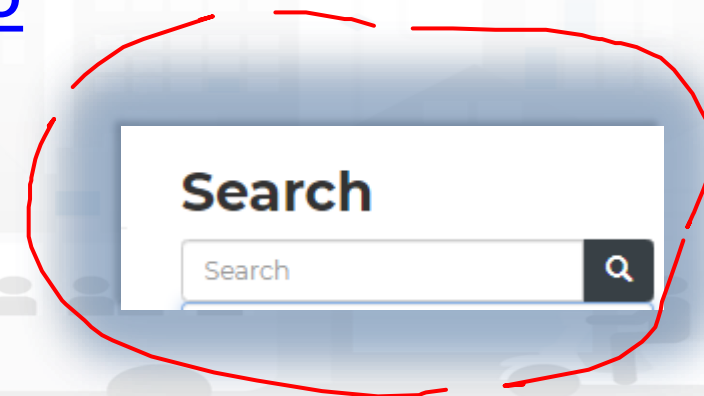
- Solutions
- Data Analytics



[https://www.snap4city.org/download/video/DPL\\_SNAP4SOLU.pdf](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>
- **Scenariious:** <https://www.snap4city.org/4>
- **Publications:** <https://www.snap4city.org/426>
- **WEB pages:** <https://www.snap4city.org/78>
- ***SEARCH on the right side***



### 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](mailto: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>

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



# Development

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



## 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](mailto: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

# Client Side Business Logic

<https://www.snap4city.org/download/video/ClientSideBusinessLogic->

[Wdf](https://www.snap4city.org/download/video/ClientSideBusinessLogic-)



## 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/UC3TAO09EbNba8f2-u4vandu>

Coordinator: Paolo Nesi, [Paolo.nesi@unifi.it](mailto: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

# Commercial Overview



- <https://fiware-foundation.medium.com/snap4-city-fiware-powered-smart-app-builder-for-sentient-cities-acfe24df49d5>
- [https://www.snap4city.org/drupal/sites/default/files/files/FF\\_ImpactStories\\_Snap4City.pdf](https://www.snap4city.org/drupal/sites/default/files/files/FF_ImpactStories_Snap4City.pdf)

SMART CITIES AND SMART INDUSTRY

**Snap4City:**  
**FIWARE powered smart app  
builder for sentient cities**

With the contribution of

TOP

# Acknowledgements

FROM CITY DASHBOARD TO APPLICATIONS

FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS

IOT APPLICATIONS VS IOT EDGE DEVICES

SNAP4CITY FOR BEGINNERS

SNAP4CITY ARCHITECTURE AND ECOSYSTEM. OPENED TO DEVELOPERS AND STAKEHOLDERS

TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS

SNAP4CITY AND KM4CITY PROJECTS

DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT

HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP

DECISION SUPPORT SYSTEM AND CITY RESILIENCE

SNAP4CITY THE VIEW OF THE ADMINISTRATORS

IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS

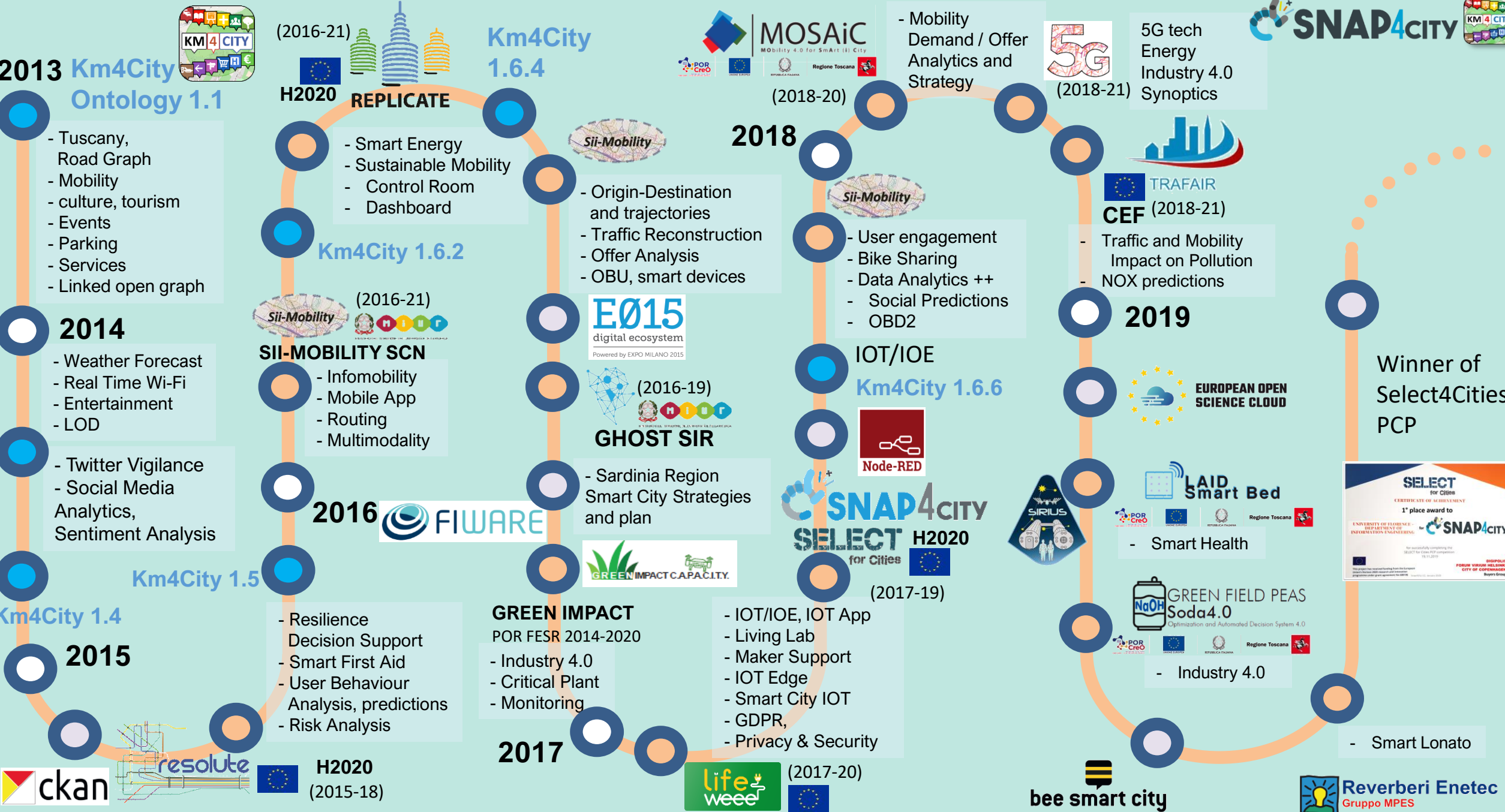
ADVANCED SMART CITY API, MICROSERVICES, SNAP4CITY API

WHAT-IF AND SIMULATION

SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK







**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**

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



**DISIT lab roadmap vs model and tools' usage**

**(2016-21) H2020 REPLICATE Km4City 1.6.4**

- Smart Energy
- Sustainable Mobility
- Control Room
- Dashboard

**Km4City 1.6.2**

- SII-MOBILITY SCN**
- Infomobility
  - Mobile App
  - Routing
  - Multimodality

**2016 FIWARE**

- GREEN IMPACT**  
POR FESR 2014-2020
- Industry 4.0
  - Critical Plant
  - Monitoring



**MOSAiC (2018-20)**

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

**E015 digital ecosystem**  
Powered by EXPO MILANO 2015

**(2016-19) GHOST SIR**

- Sardinia Region Smart City Strategies and plan

**SNAP4CITY SELECT for Cities H2020 (2017-19)**

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



- Smart Waste

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

- Mobility Demand / Offer
- Analytics and Strategy

**2018**

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

**IOT/IOE Km4City 1.6.6**



**LAID Smart Bed**

- Smart Health

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

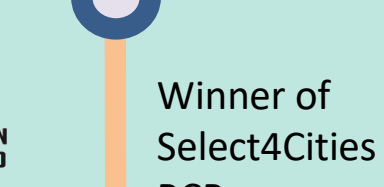


- Smart Lonato

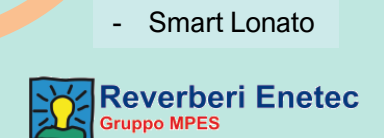


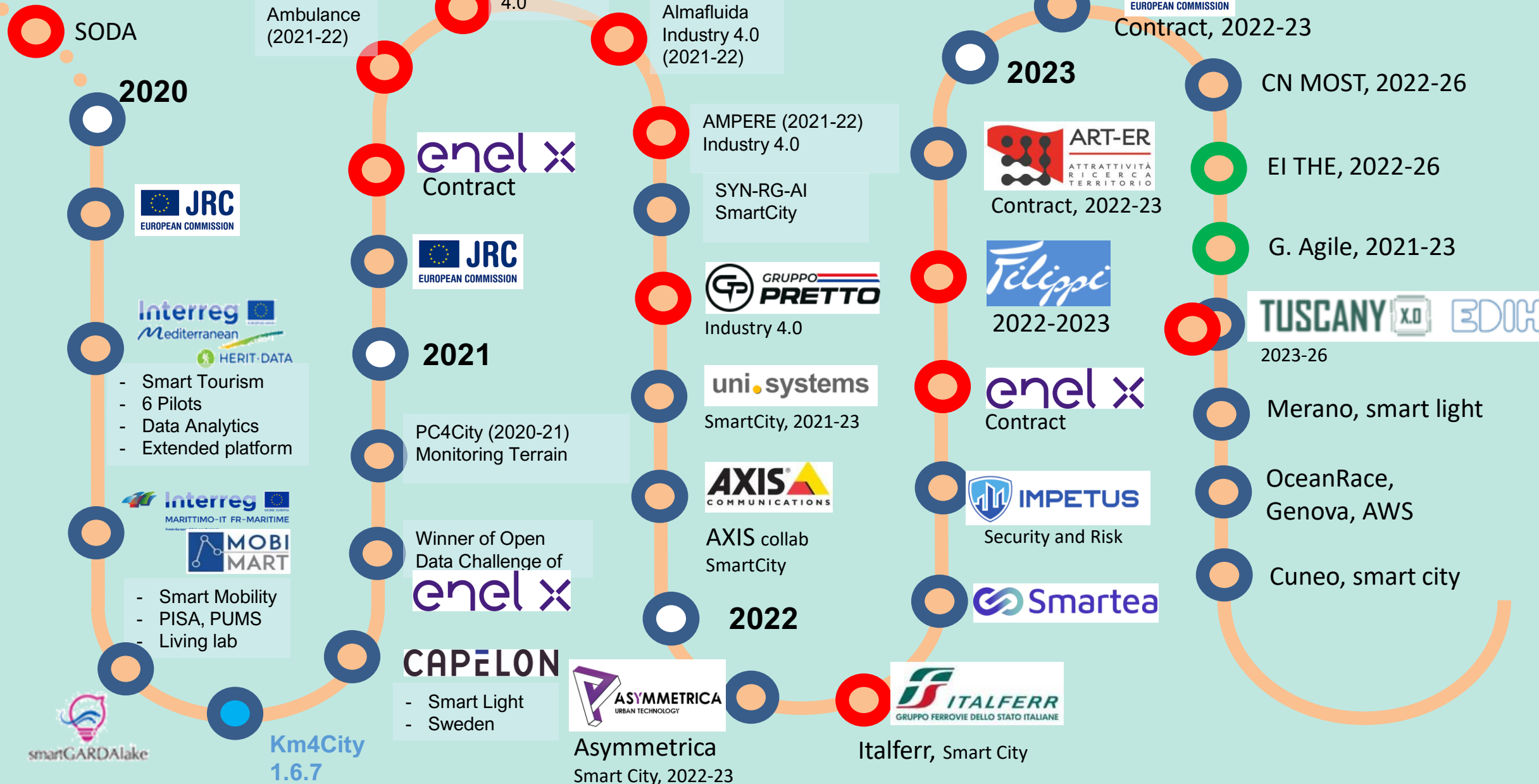
**2019**

- Traffic and Mobility Impact on Pollution
- NOX predictions



**Winner of Select4Cities PCP**





**2020**

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

- Smart Mobility
- PISA, PUMS
- Living lab

**2021**

**2022**

**2023**

**2023**

Smart Ambulance (2021-22)

Enterprise (2021-22) Industry 4.0

Almafluida Industry 4.0 (2021-22)

AMPERE (2021-22) Industry 4.0

SYN-RG-AI SmartCity

GRUPPO PRETTO Industry 4.0

uni.systems SmartCity, 2021-23

AXIS COMMUNICATIONS

AXIS collab SmartCity

ITALFERR GRUPPO FERROVIE DELLO STATO ITALIANE

Italferr, Smart City

enel x Contract

JRC EUROPEAN COMMISSION

PC4City (2020-21) Monitoring Terrain

Winner of Open Data Challenge of enel x

CAPELON - Smart Light - Sweden

ASYMMETRICA URBAN TECHNOLOGY

Asymmetrica Smart City, 2022-23

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.D EDIH 2023-26

Merano, smart light

OceanRace, Genova, AWS

Cuneo, smart city

smartGARDAlake Km4City 1.6.7

TOP



*Be smart in a SNAP!*



**SMARTCITY**

EXPO WORLD CONGRESS

7-9 November 2023, Barcelona, Spain

Visit Snap4City in Hall 1

## 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](http://www.snap4city.org)

 **SNAP4**  
Appliances and Dockers  
**Installations**

Email: [snap4city@disit.org](mailto: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