100% OPEN SOURCE

EU GDPR COMPLIANT

PEN Test Passed

Powered by FIWARE

SNAP4CITY on EUROPEAN OPEN SCIENCE CLOUD MARKETPLACE

Node-RED

SNAP4CITY

KM4CITY

SNAP4 Appliances and Dockers Installations

Be smart in a SNAP!

SMARTCITY EXPO WORLD CONGRESS

7-9 November 2023, Barcelona, Spain

Visit Snap4City in Hall 1

IoT App. / Proc.Logic
Server Side Business Logic

Sept. 2023, Course, Part 3
https://www.snap4city.org/944
https://www.snap4city.org/577

LIVING LAB

SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

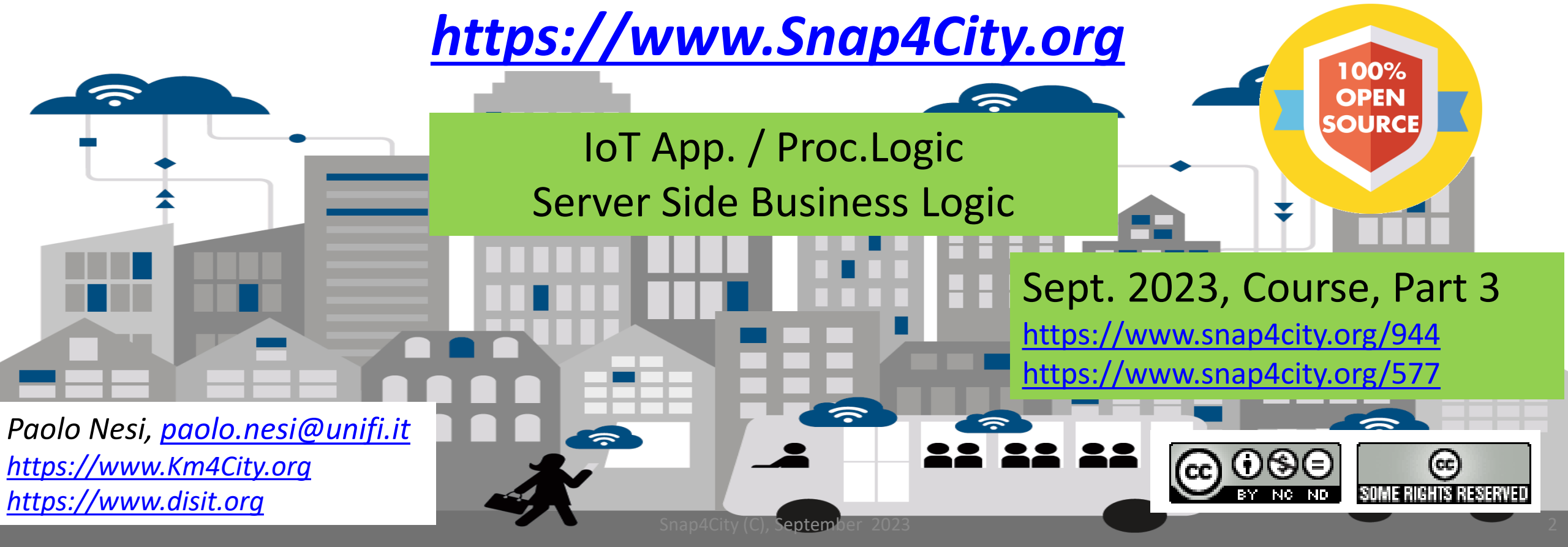
UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

SNAP4CITY

Powered by KM4CITY

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

https://www.Snap4City.org



100% OPEN SOURCE

IoT App. / Proc.Logic
Server Side Business Logic

Sept. 2023, Course, Part 3
https://www.snap4city.org/944
https://www.snap4city.org/577

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

100% OPEN SOURCE

Powered by FIWARE

FREE TRIAL

PEN Test Passed

EU GDPR COMPLIANT

SNAP4 Appliances and Dockers Installations

EUROPEAN OPEN SCIENCE CLOUD

Node-RED

JS Foundation

E015 digital ecosystem

NVIDIA

SNAP4CITY KM4CITY

SMART SOLUTIONS AND DECISION SUPPORT SYSTEMS

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

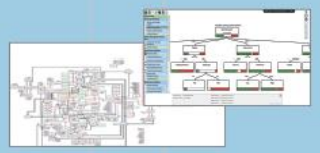
DASHBOARDS - VISUAL ANALYTICS - SYNOPTICS - DIGITAL TWIN - GRAPHICAL WIDGETS - ANALYTICS - GUI CUSTOM STYLES - VISUAL PROGRAMMING

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



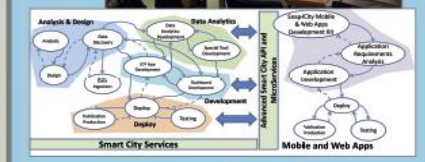
EXPERT SYSTEM, KNOWLEDGE BASE
SEMANTIC REASONING
SMART DATA MODEL
IOT DEVICE MODELS, STORAGE

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

https://www.snap4city.org/577

On Line Training Material (free of charge)

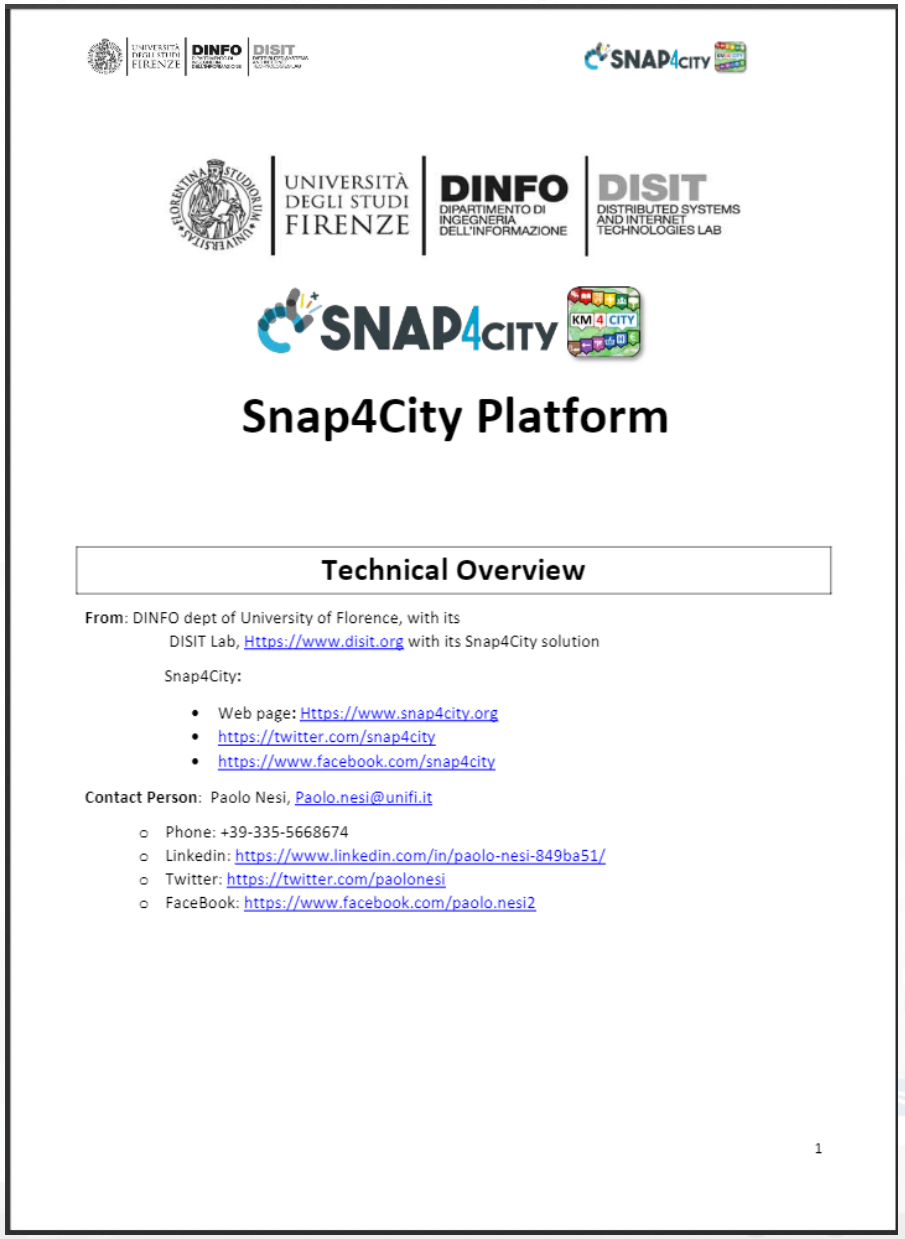
https://www.snap4city.org/944

	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

Note on Training Material

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



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

SNAP4CITY KM4CITY

Snap4City Platform

Technical Overview

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

Snap4City:

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

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

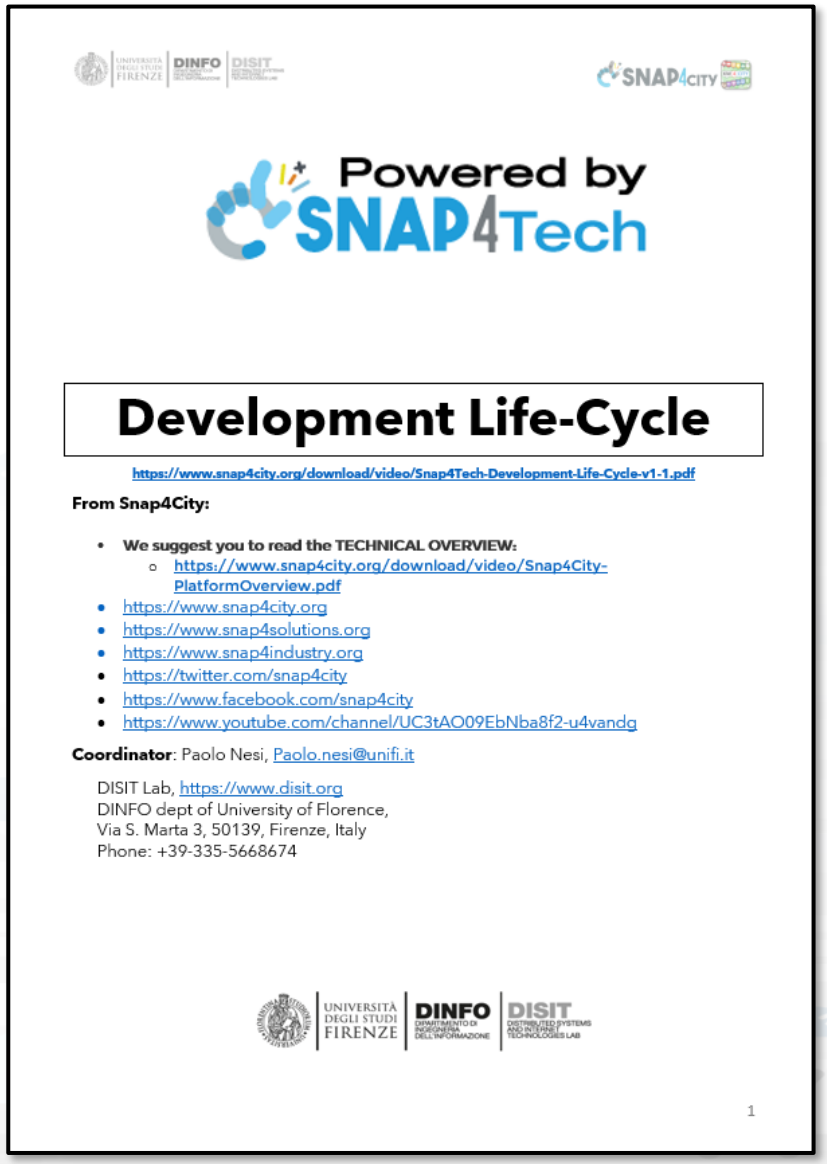
- Phone: +39-335-5668674
- Linkedin: https://www.linkedin.com/in/paolo-nesi-849ba51/
- Twitter: https://twitter.com/paolonesi
- FaceBook: https://www.facebook.com/paolo.nesi2

1

Tech Overview

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

Development



Powered by SNAP4Tech

Development Life-Cycle

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

From Snap4City:

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

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

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

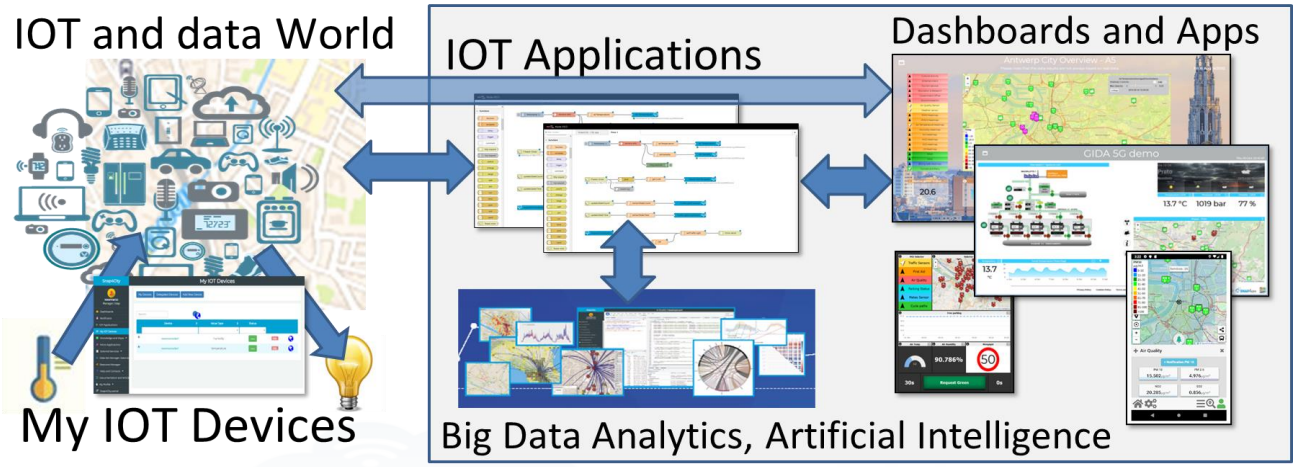
UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

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

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

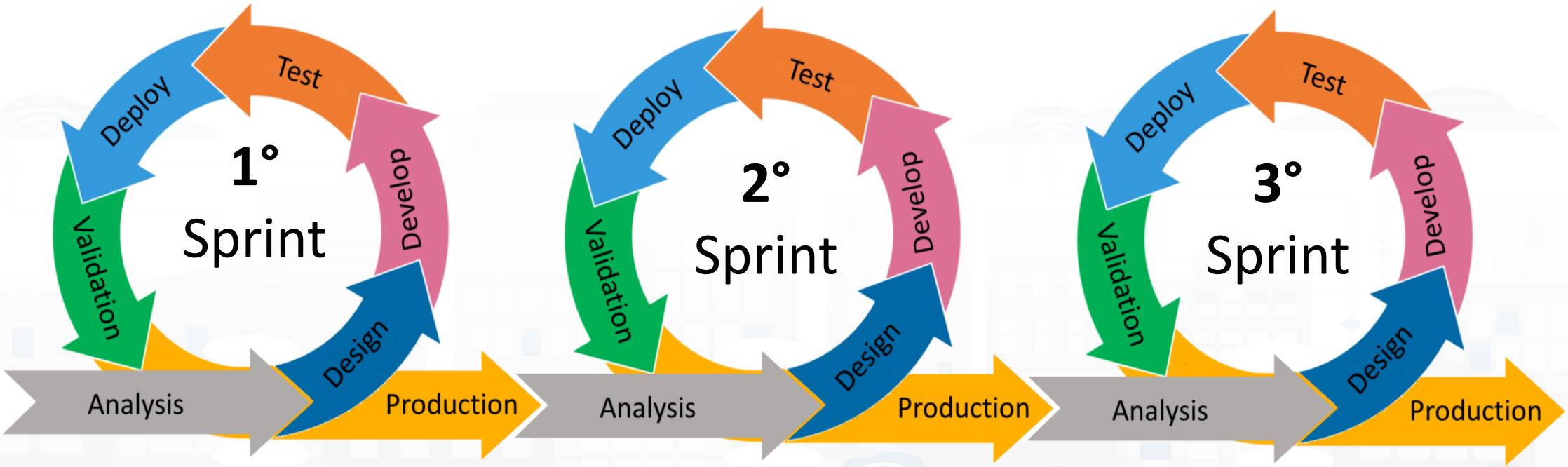


IF you need to go more in deep you can ask us to pass at the next Role becoming full AreaManager with full rights of development, also for Data Analytics, machine learning, etc.

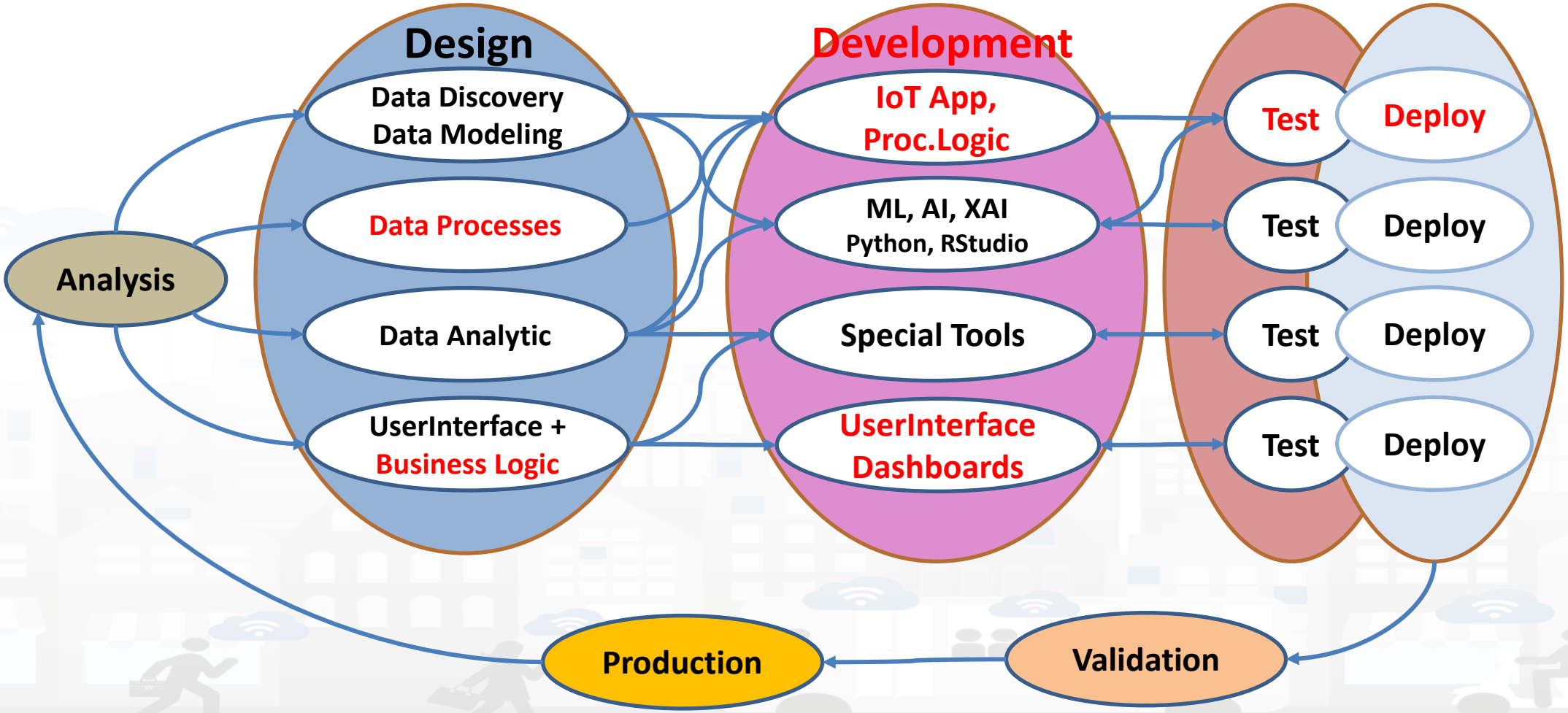
Agenda of third part

- Recall on Snap4City Architecture
- Node-RED
- IOT App = Node-RED + Snap4City
  - IoT App === Proc.Logic
- Examples of IOT App for Smartening Solutions
- Exploiting/Generating data by using: IoT App/Proc.Logic
- External Service ←→ IoT App/Proc.Logic
- Dashboards ←→ IoT App/Proc.Logic
  - Server Side Business Logic

Development Life Cycle Smart Solutions

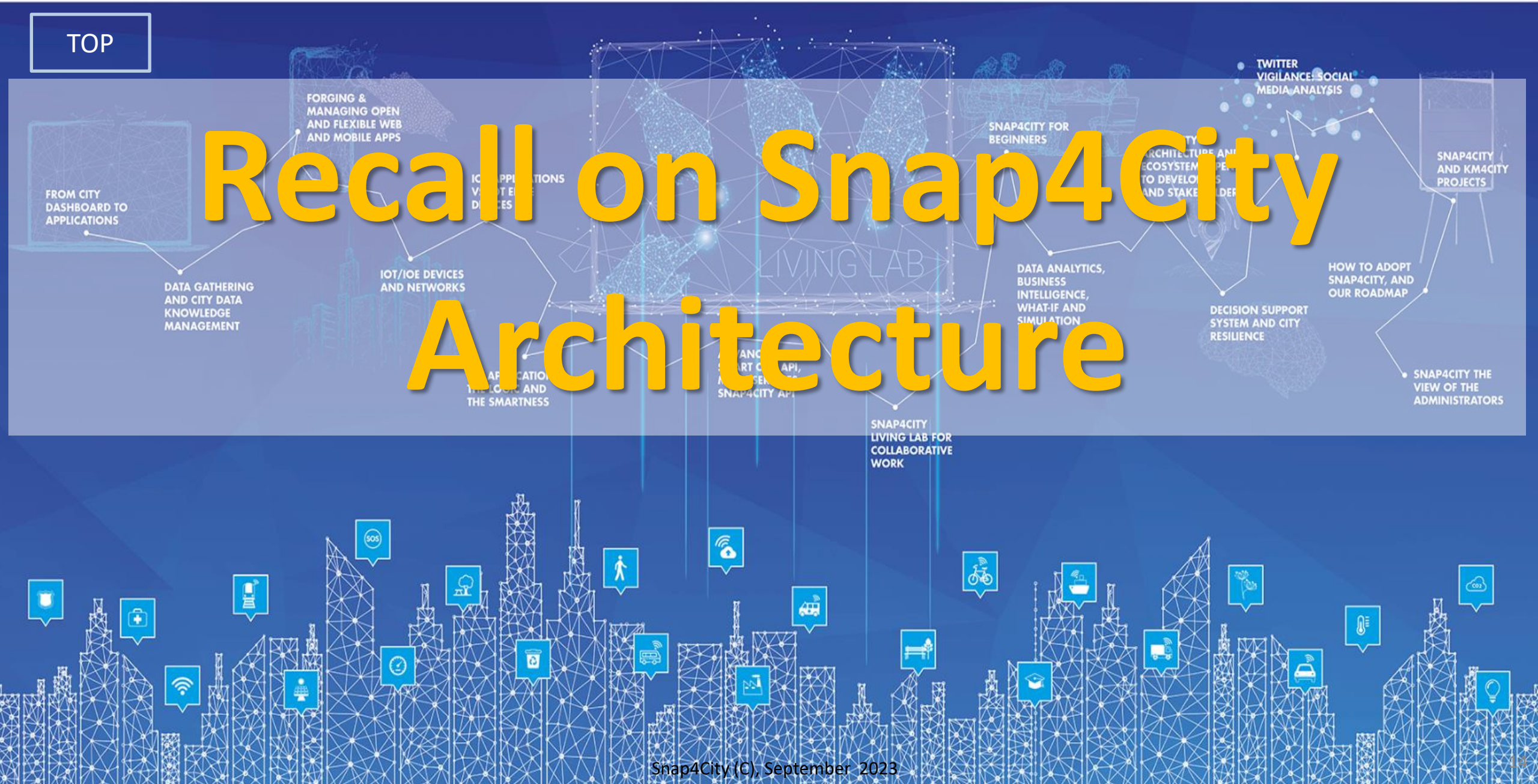


Development Life Cycle Smart Solutions



TOP

Recall on Snap4City Architecture



UNIVERSITÀ DEGLI STUDI FIRENZE

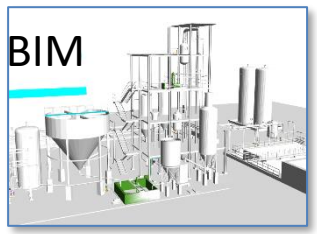
DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

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

Concept

SNAP4CITY KM4CITY

BIM

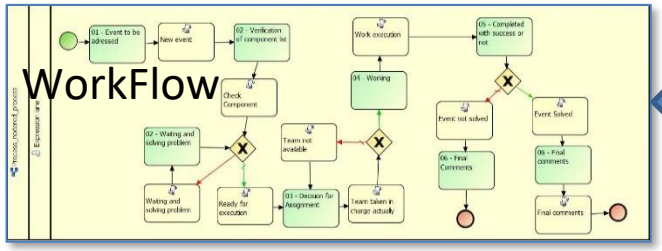


KPI, POI, MyKPI, …

API, External Services

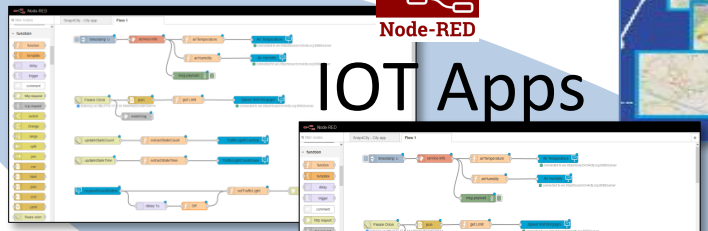
Web Scraping

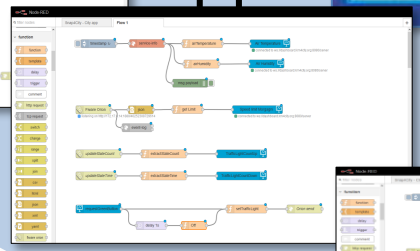
WorkFlow

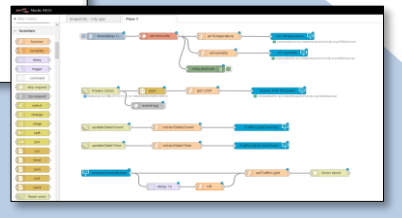


Node-RED

IOT Apps







Data Analytics, Artificial Intelligence

IOT Brokers

IOT Broker

IOT Broker

FIWARE

GIS

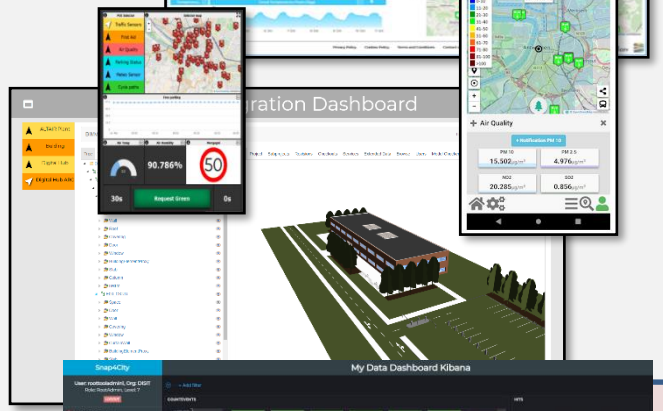
Big Data

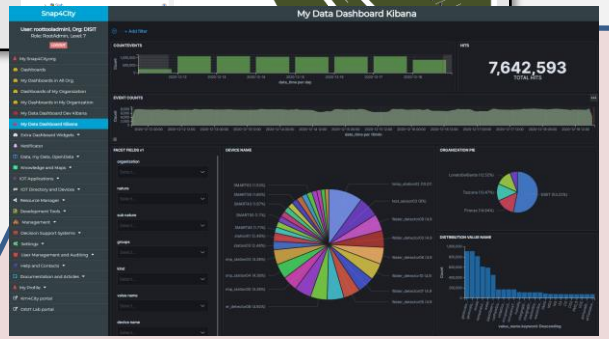
KB

LD, LOD

ckan

FI-WARE

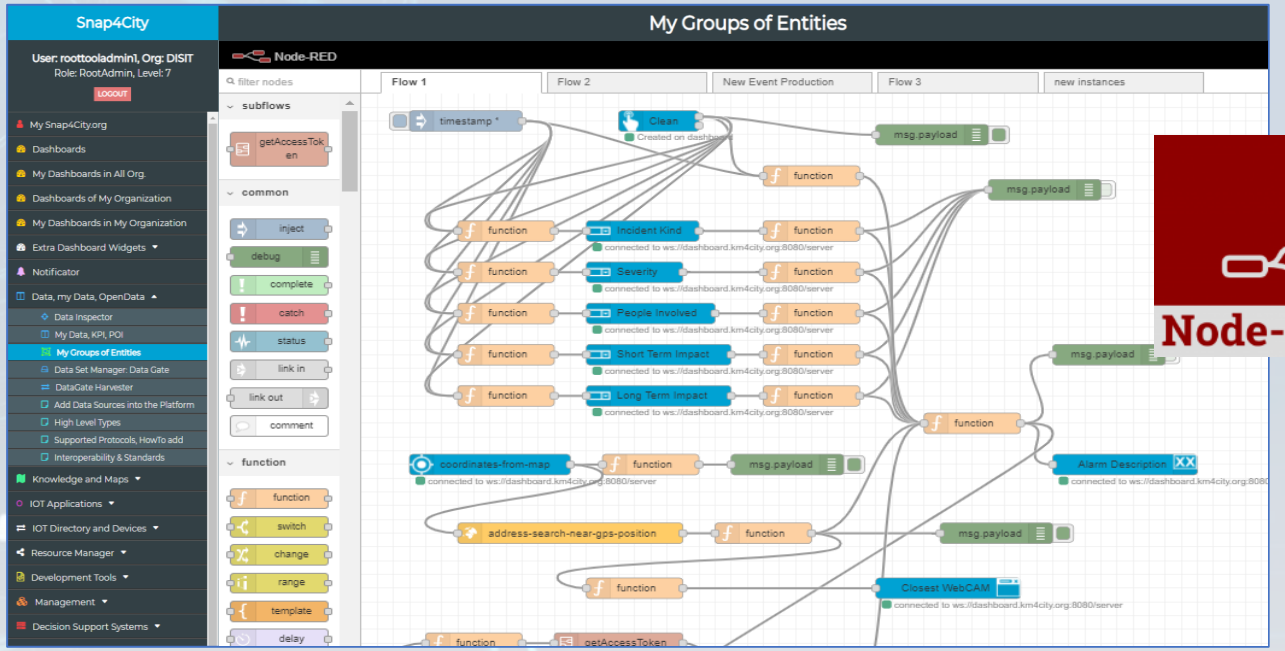


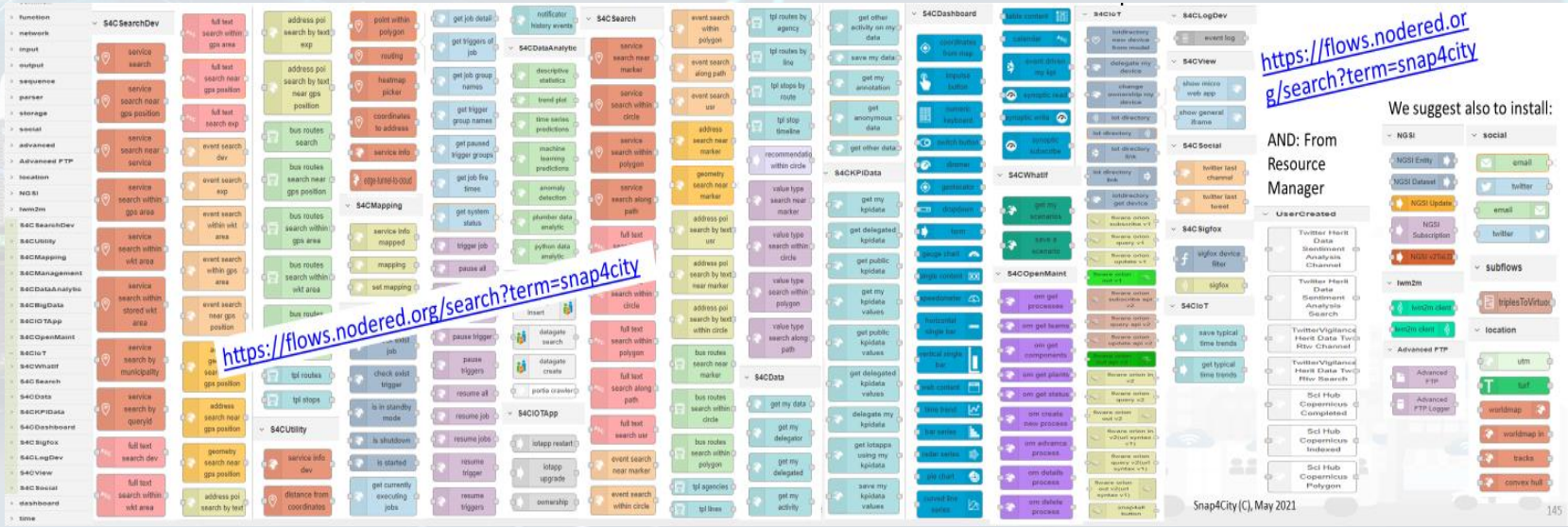


Dashboards and Apps

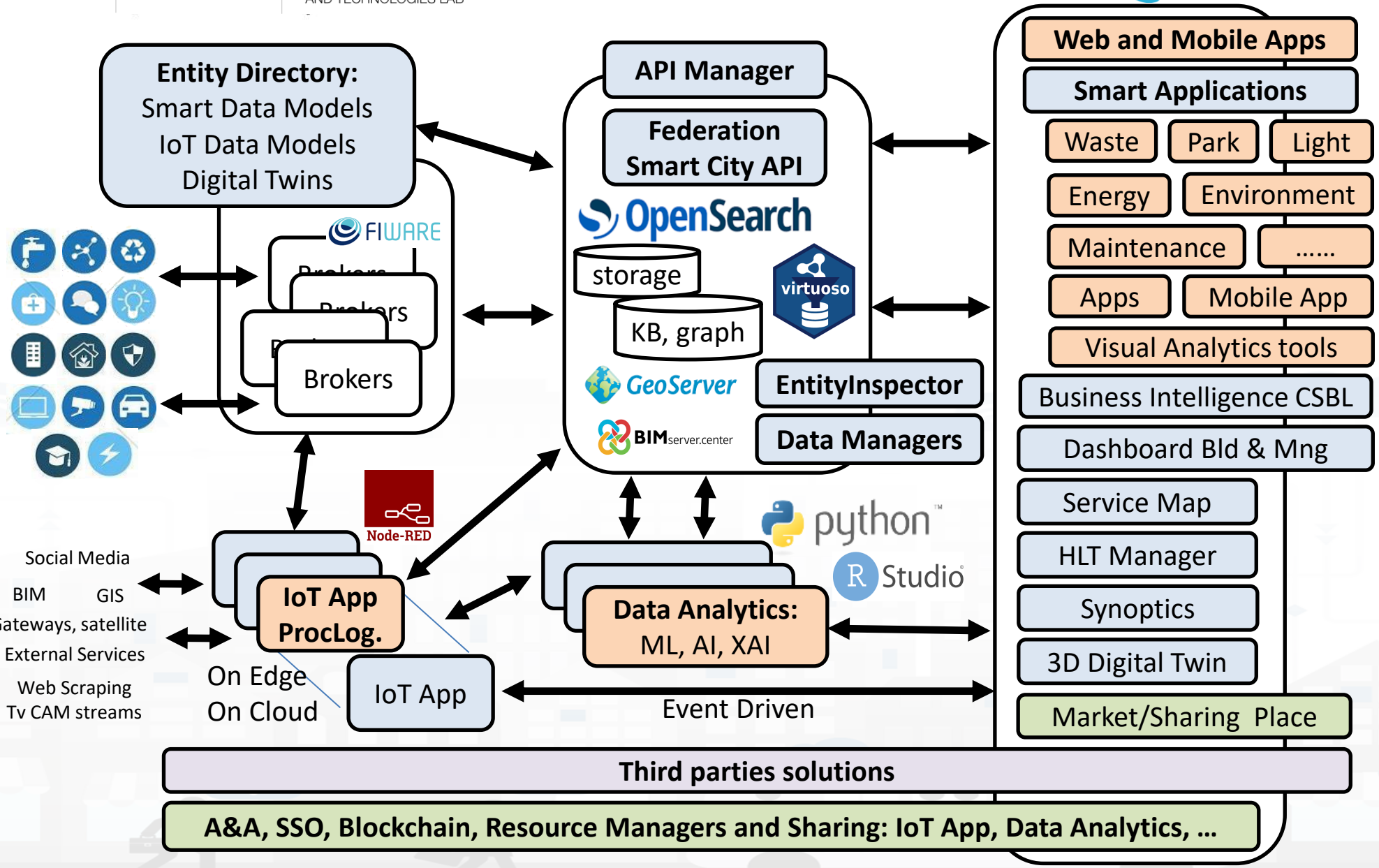
Ingestion, aggreg. → exploitation

- **IoT App Visual Programming, no coding**
  - Data transformation
  - Integration, Interoperab.
  - Scripting Data Analytics
  - Data ingestion
  - Business logic
- **Edge and Cloud**
- **MicroServices** data driven develop via visual language Node-RED

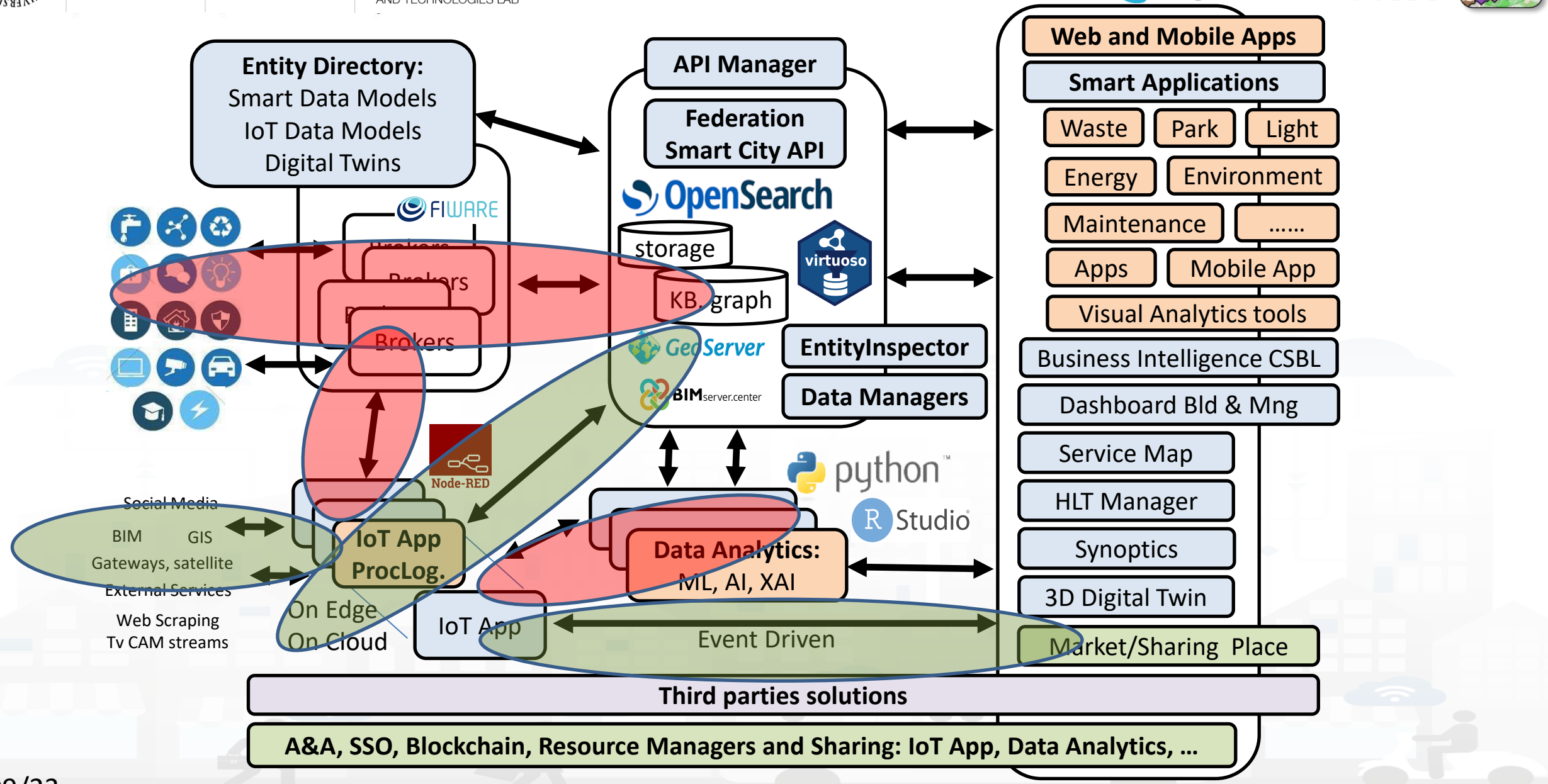




Tech Arch

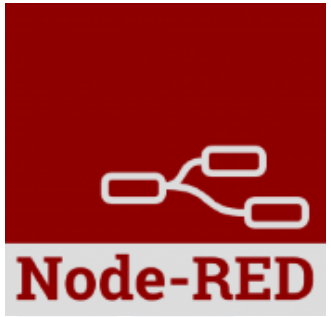

Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
EntityInspector
BIMserver.center
Data Managers
Node-RED
python
RStudio
Social Media
BIM
GIS
Gateways, satellite
External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

Tech Arch


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

IoT App / Proc.Logic

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



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

Part 4

Part 5

SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES

TOP

FROM CITY DASHBOARD TO APPLICATIONS

DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT

FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS

IOT/IOE DEVICES AND NETWORKS

IOT APPLICATIONS VS IOT EDGE DEVICES

IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS

ADVANCED SMART CITY API, MICROSERVICES, SNAP4CITY API

SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK

DATA ANALYTICS, BUSINESS INTELLIGENCE, WHAT-IF AND SIMULATION

SNAP4CITY FOR BEGINNERS

SNAP4CITY ARCHITECTURE AND ECOSYSTEM. OPENED TO DEVELOPERS AND STAKEHOLDERS

TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS

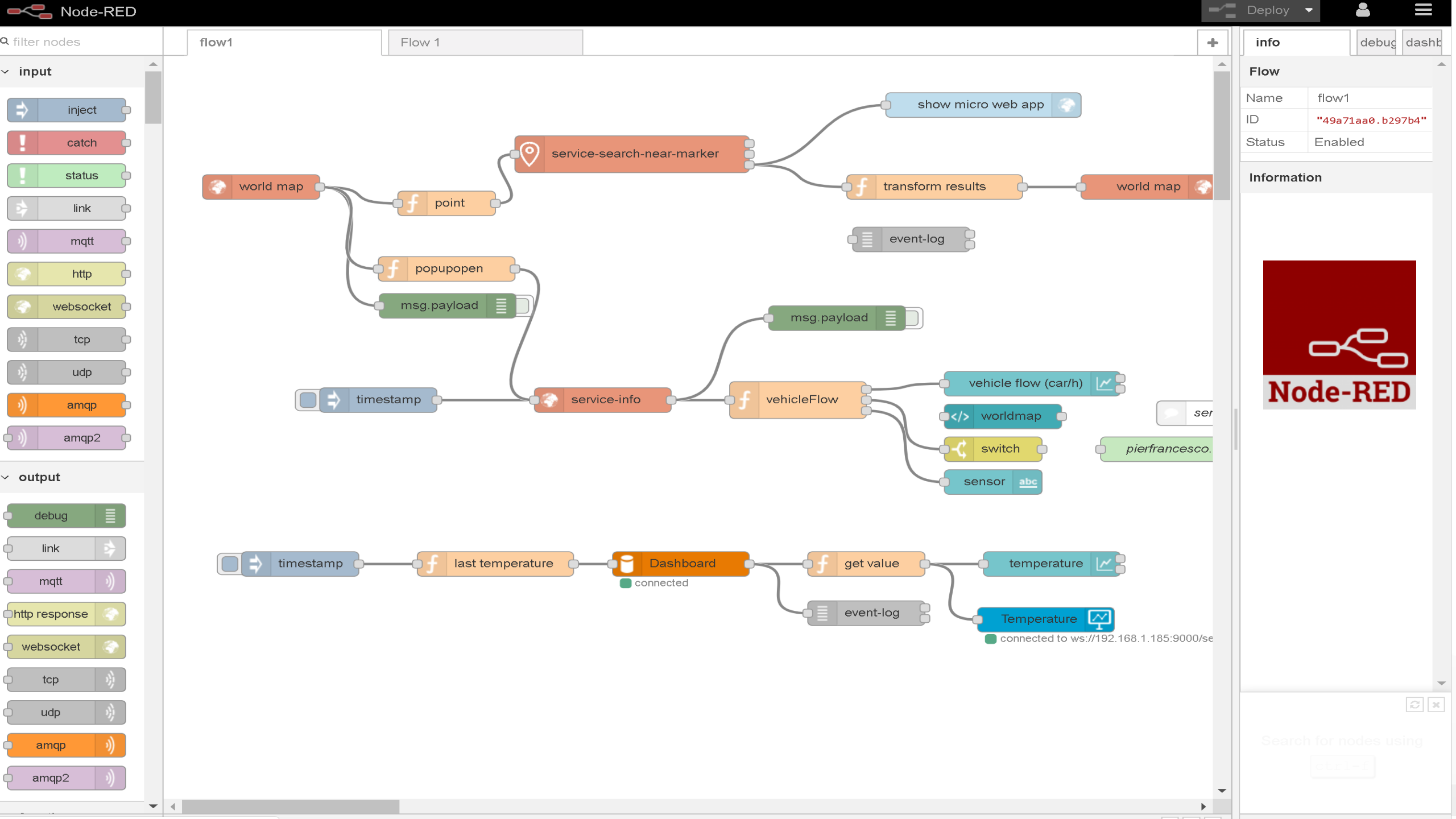
HOW TO ADOPT SNAP4CITY, AND

SNAP4CITY AND KM4CITY PROJECTS

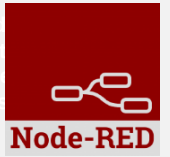
...AP4CITY THE ...W OF THE ...MINISTRATORS

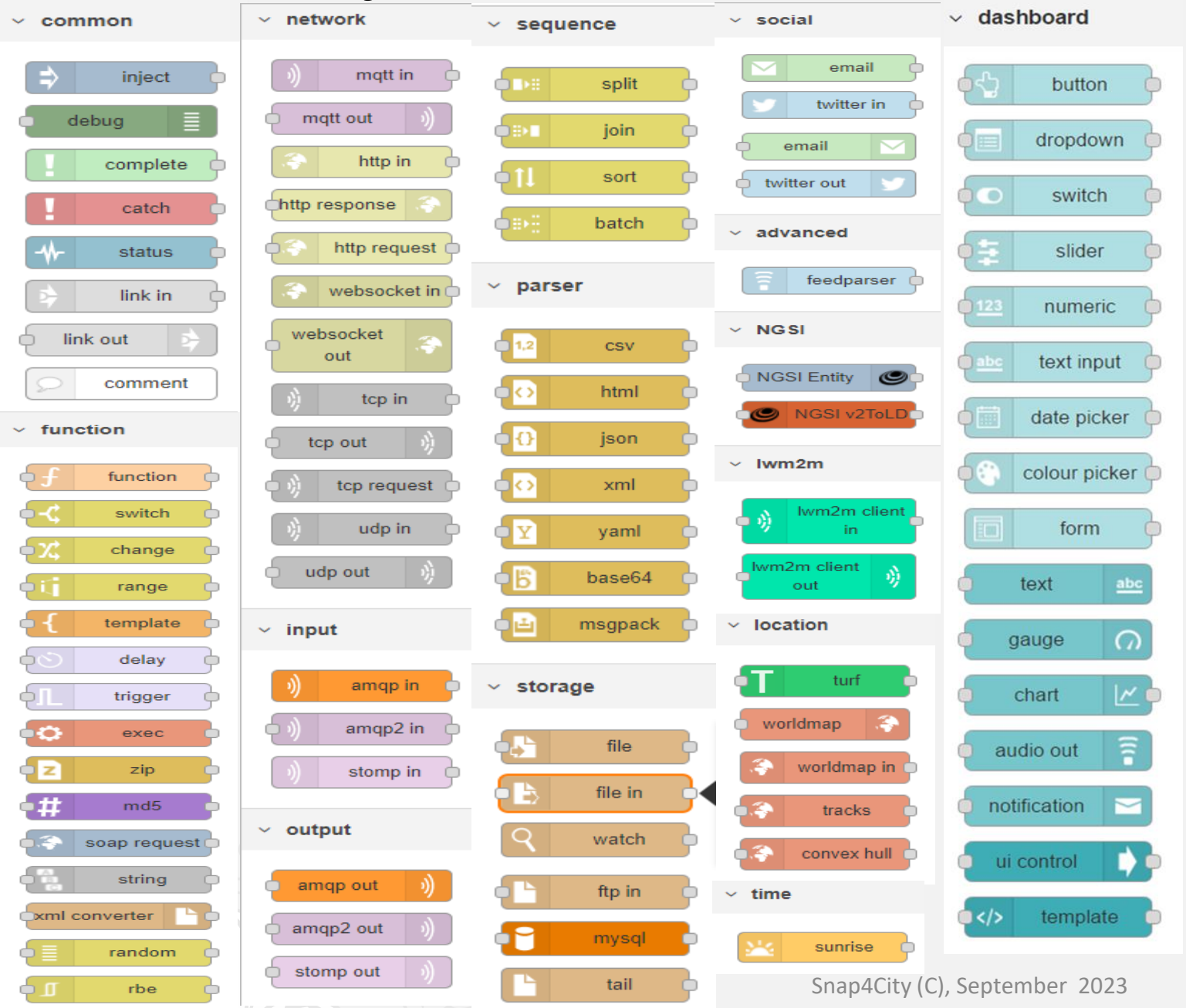
Node-RED

Node-RED

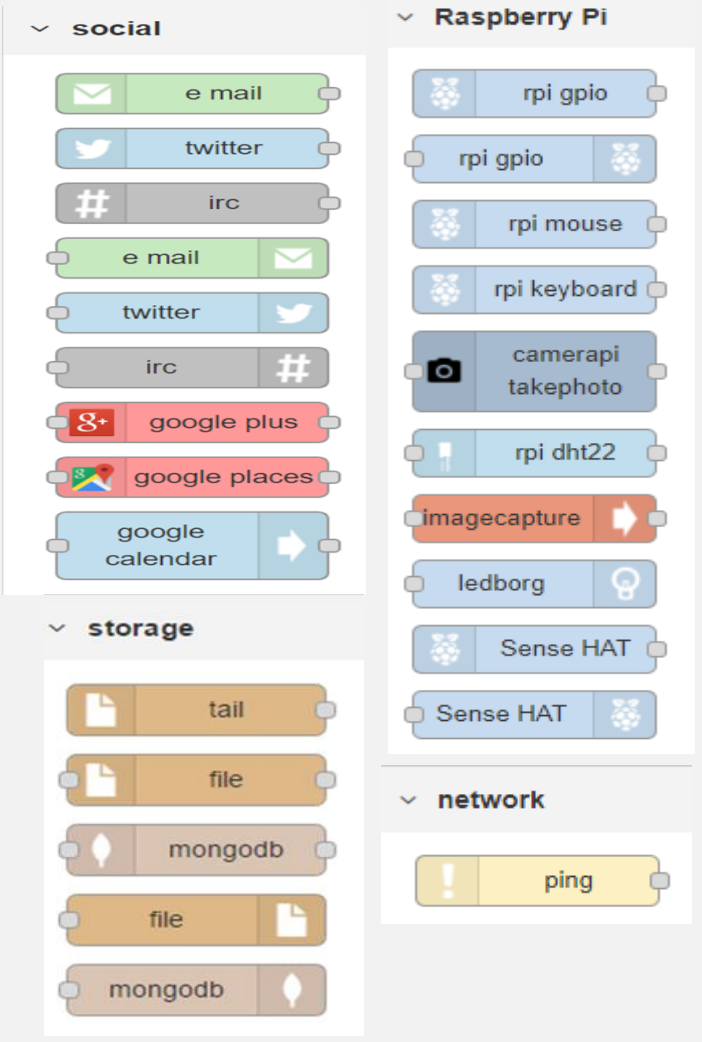

Node-RED
Deploy
filter nodes
flow1
Flow 1
input
inject
catch
status
link
mqtt
http
websocket
tcp
udp
amqp
amqp2
output
debug
link
mqtt
http response
websocket
tcp
udp
amqp
amqp2
show micro web app
service-search-near-marker
world map
point
transform results
world map
event-log
popupopen
msg.payload
msg.payload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

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





+ on IOT Edge Raspberry

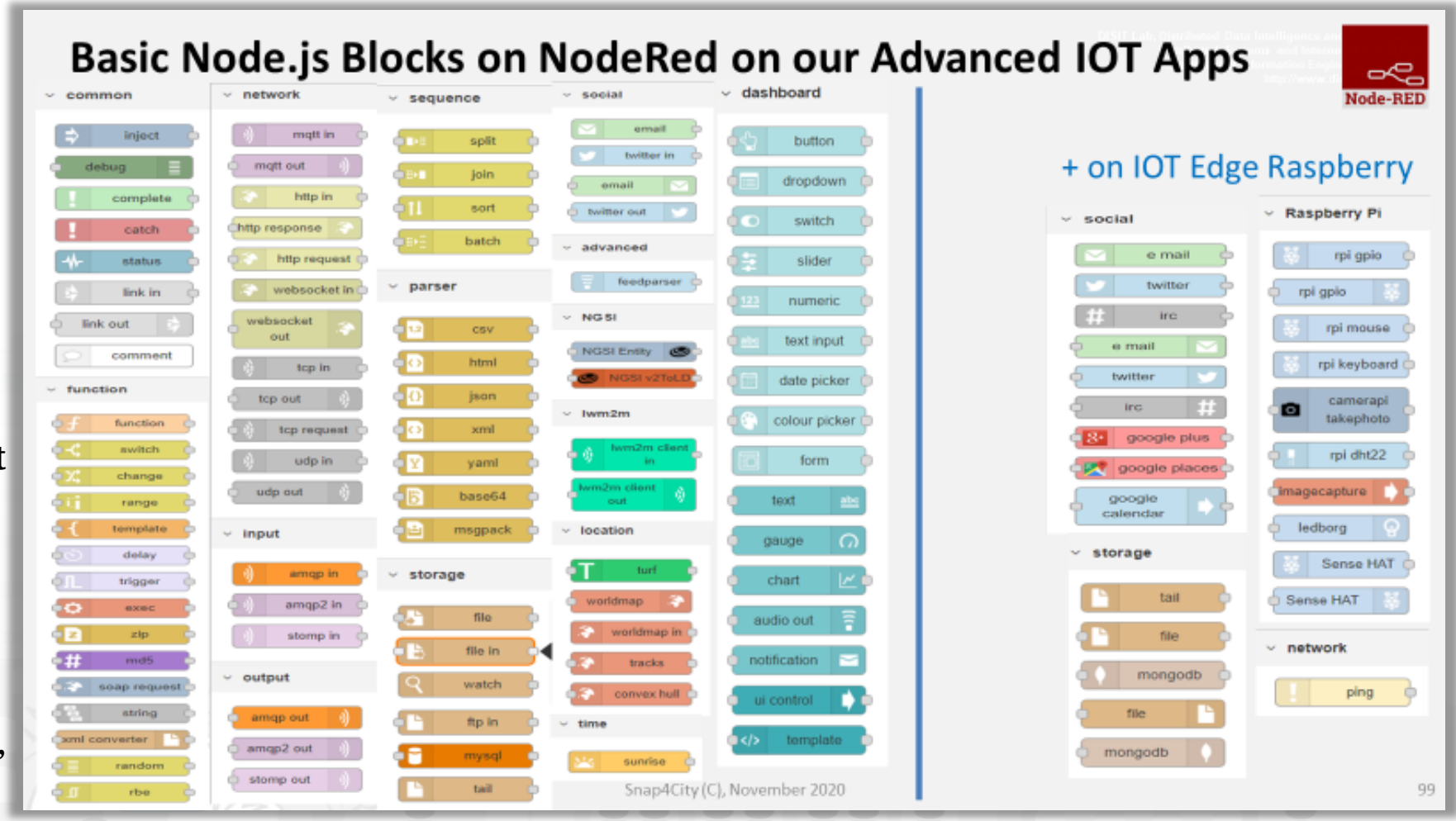


Node-RED Basic Blocks

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

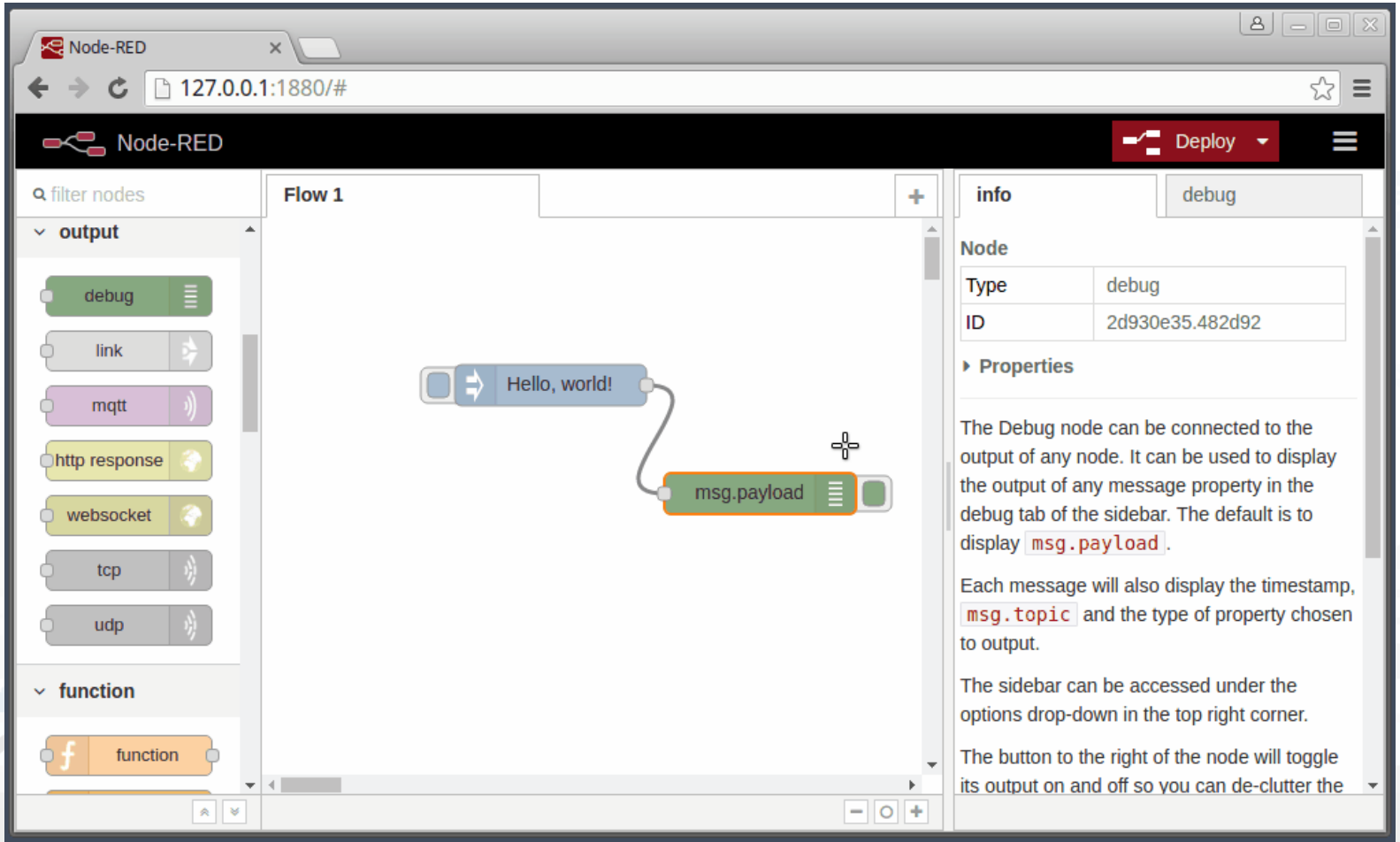
Despite to its diffusion, for the usage in the context of Smart City it was **not powerful** to cope with the **basic requirements** of the domain.

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



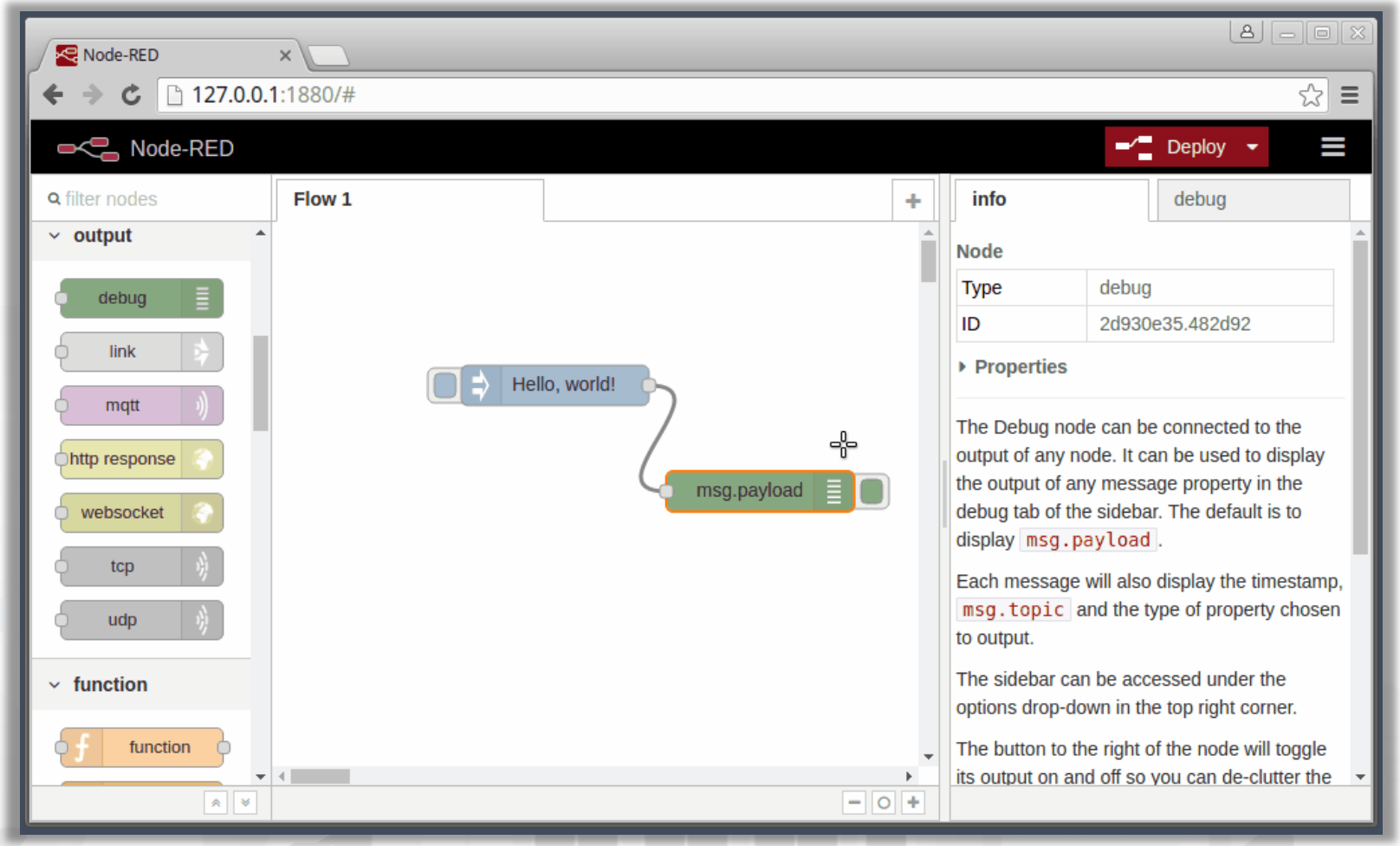
Hello World of Node-RED

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



Node-RED

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

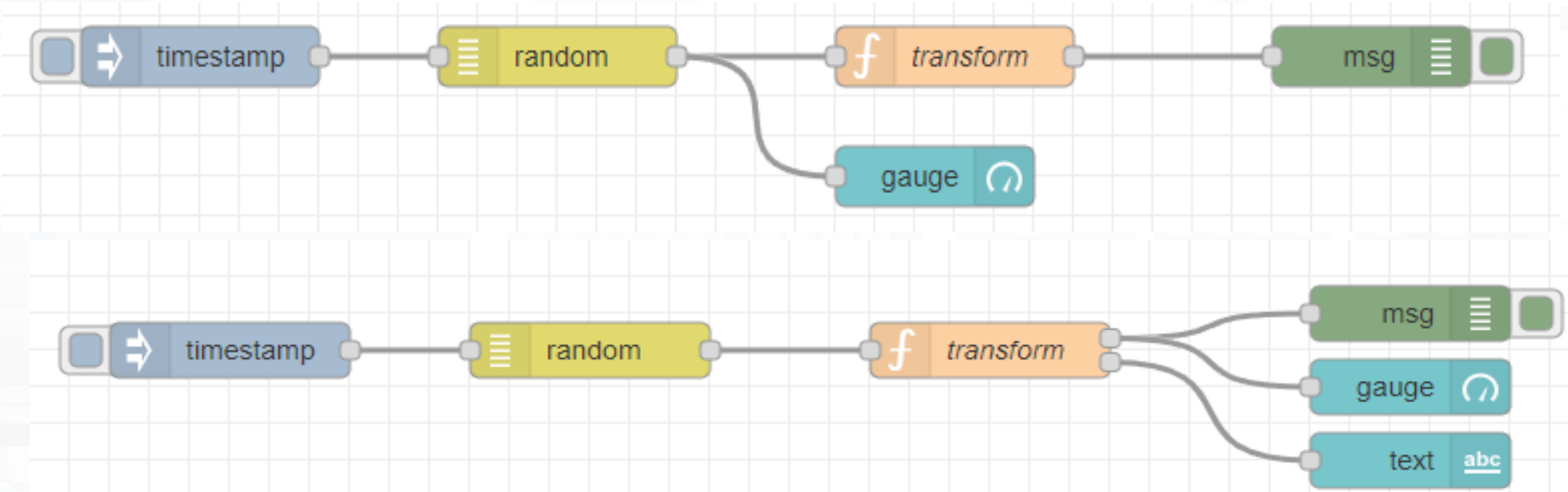


Node-RED Demo 0

Course 2023 IoT App / Proc.Logic

- Pattern:

- Pattern:



recap

- Inject messages
  - Structure of messages, msg.payload
- Function for coding
  - Function with two outputs
- Connection on Dashboard element
  - Tab of dashboard, setting for color, position, etc.
- Real time update on gauge and trends
- Debug tab, timestamp, copy value, copy path, etc.
- Etc.

Node-RED Demo 1

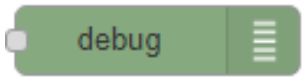
Example of simple IoT Application

In this demo let's create an IoT Application that:

- generate a random value,
- the value is switched on the correct path
- the value is showed in the local dashboard of NodeRed

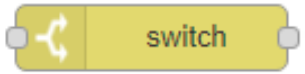
Nodes for flow

Generates an input for the other nodes. It can be repeated at predefined intervals, entered manually and of various types (timestamp, string, number, boolean, JSONetc)

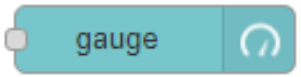


Each message that enters the debug node is shown in the "debug" tab on the right of node-red (you can choose which part of the message to show)

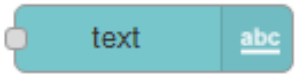
Generates a random number. You can configure the number generation interval and the type (integer or float).



Evaluates the input message and routes it to the correct output according to the desired configuration



Shows a number inside a gauge counter.

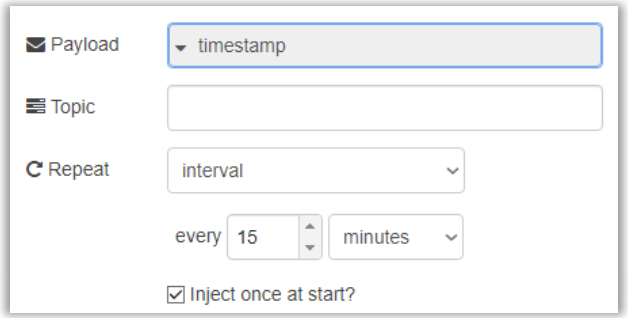


Shows a text inside the local dashboard

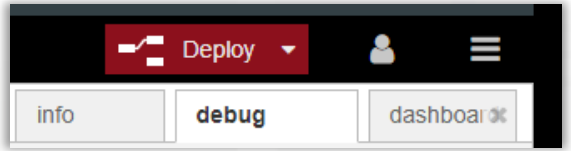
Step 1

- Inject and Debug

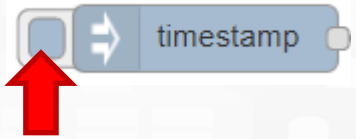
- Connect
- Configure



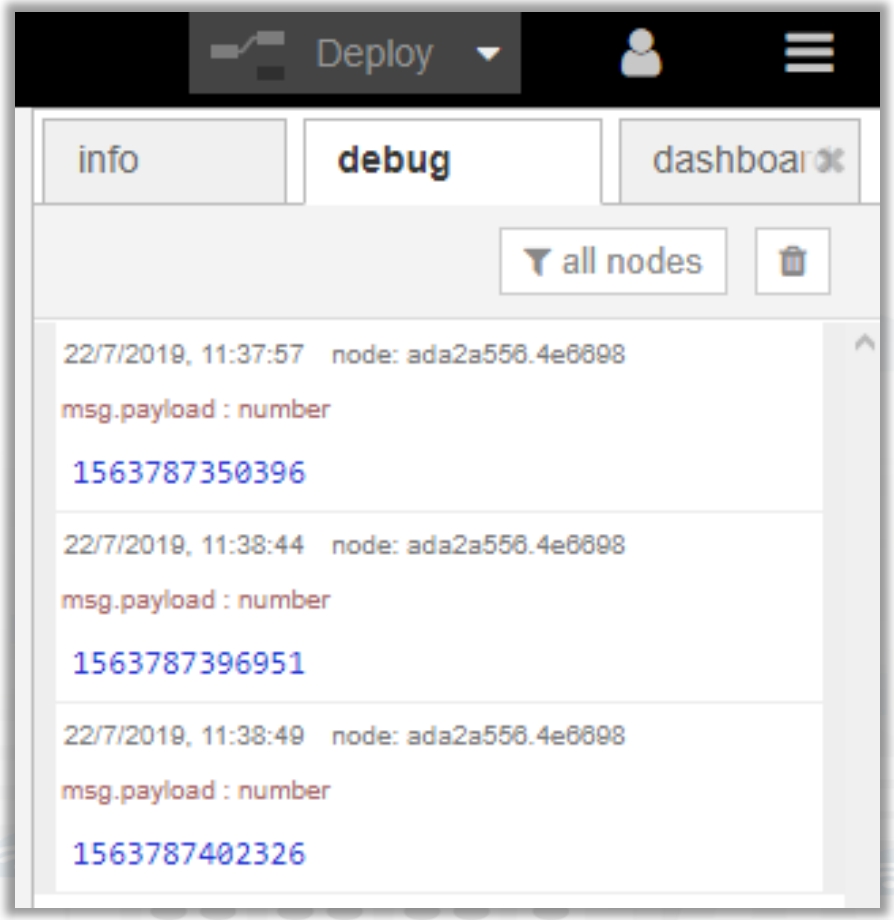
- Deploy



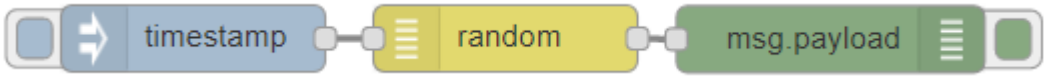
- Click



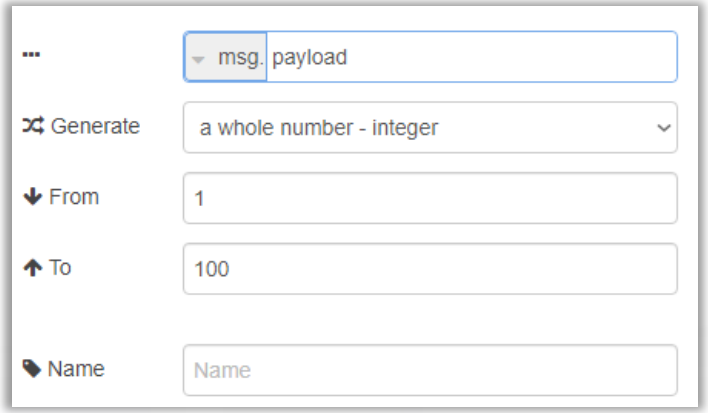
- Observe



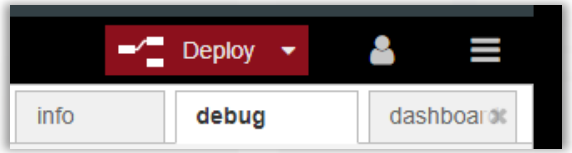
Step 2

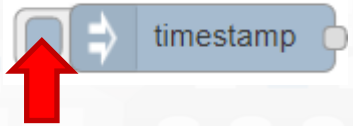


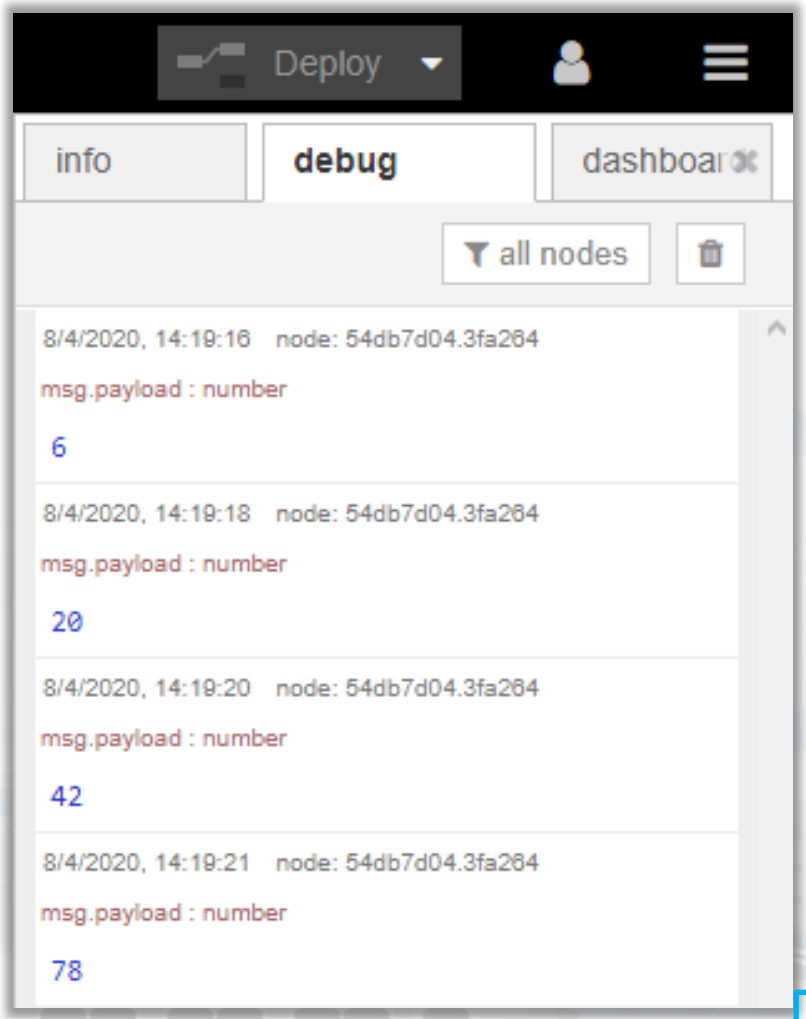
- Random
- Connect
- Configure



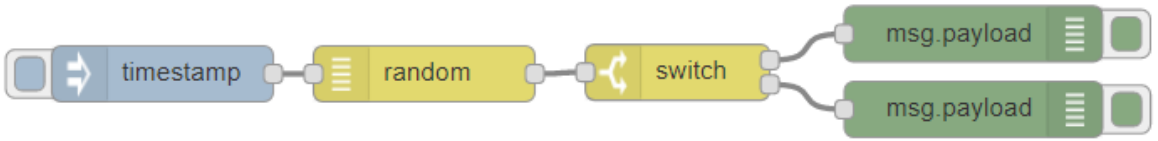
- Deploy
- Click
- Observe



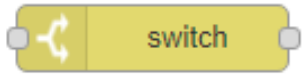




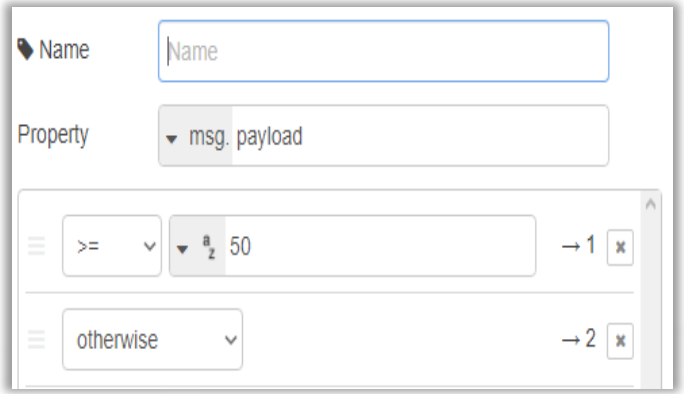
Step 3



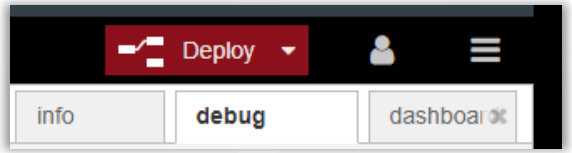
- Switch



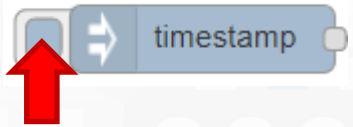
- Connect
- Configure



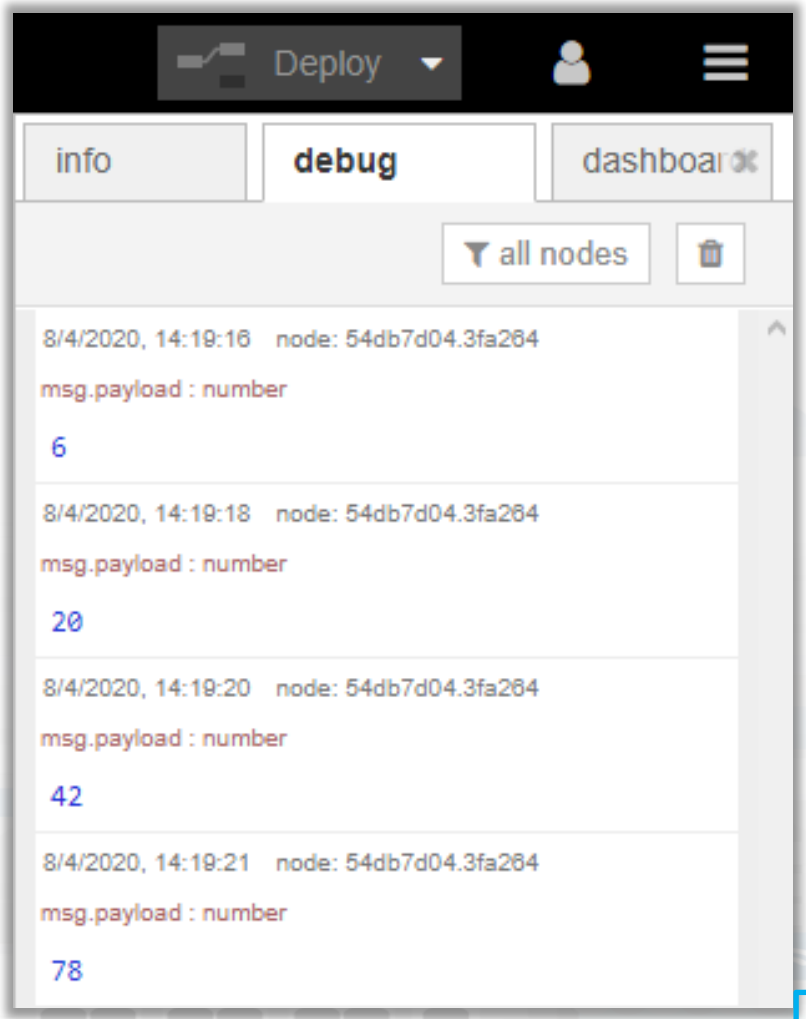
- Deploy



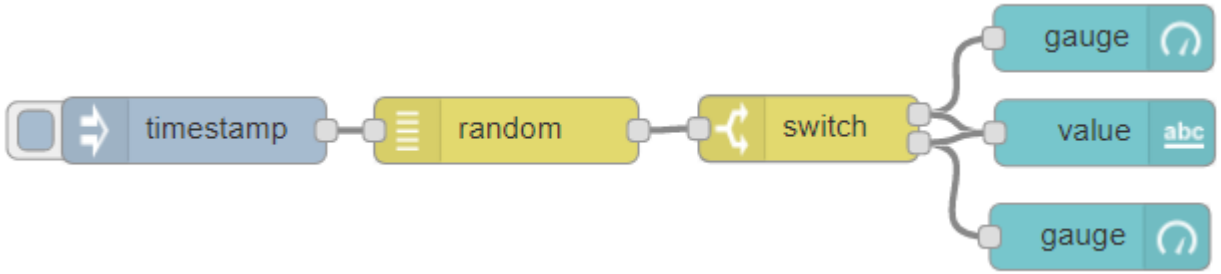
- Click



- Observe

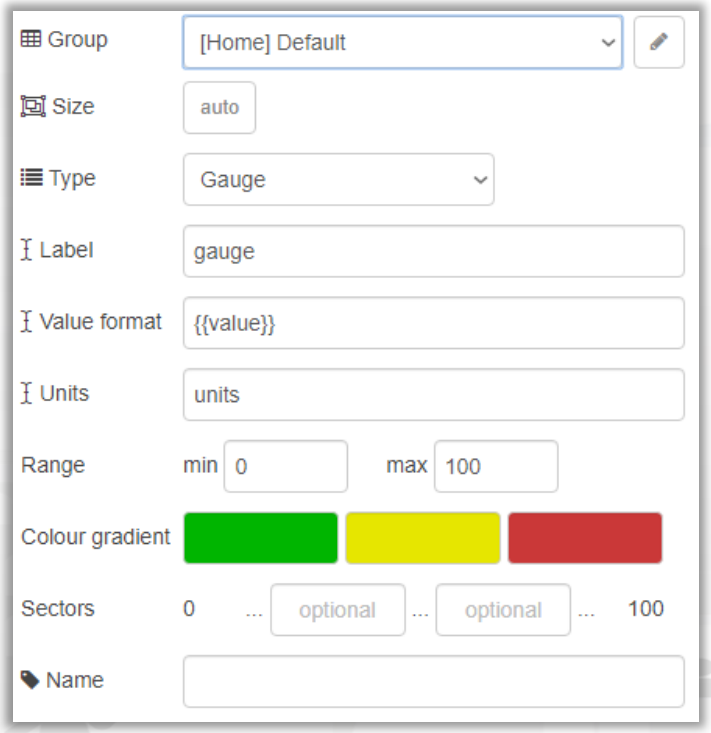


Step 4

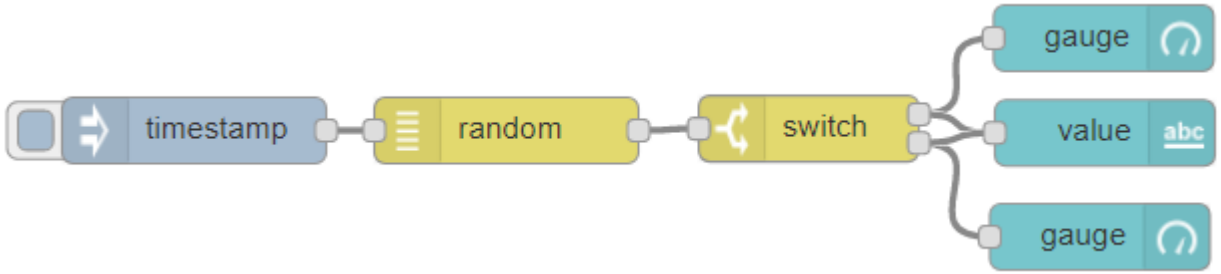


- Gauge and text 

- Connect
- Configure gauge

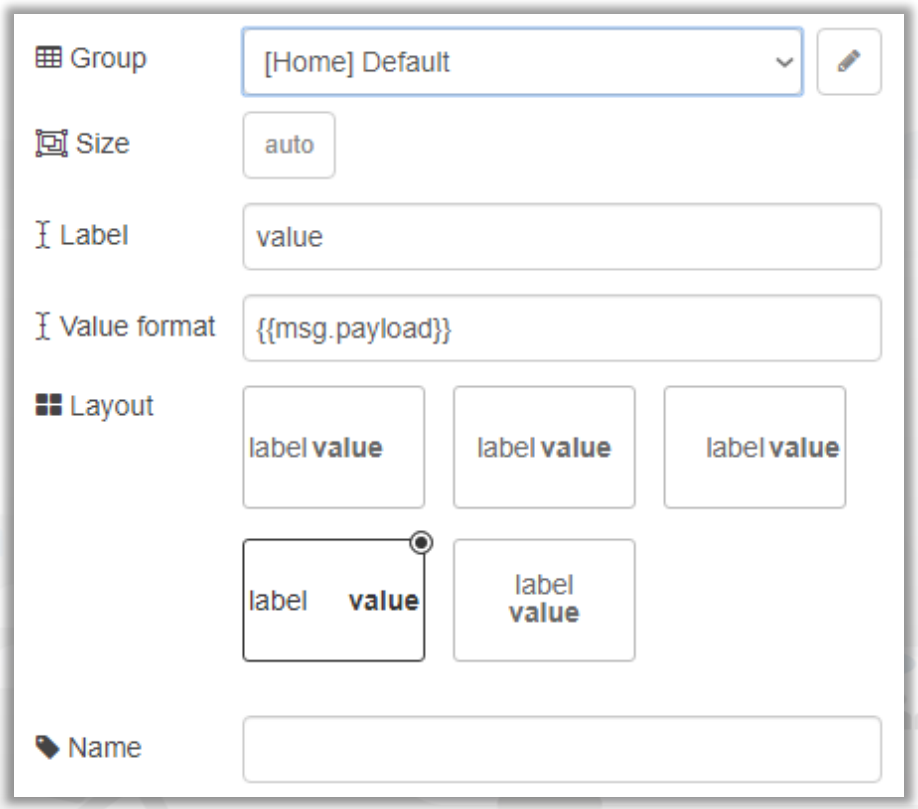


Step 4 Bis

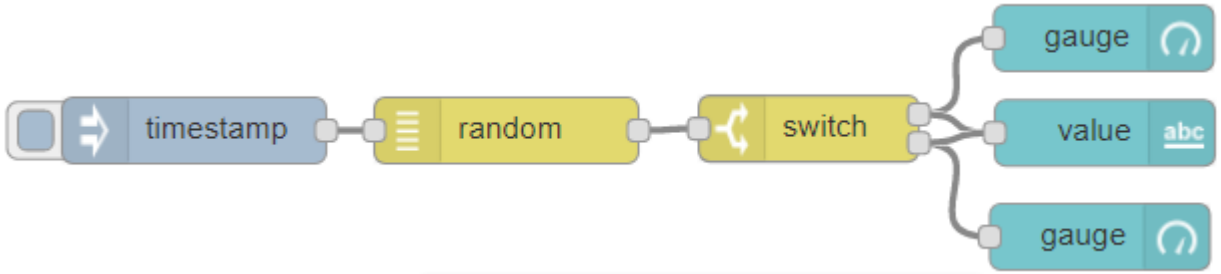


- Gauge and text 

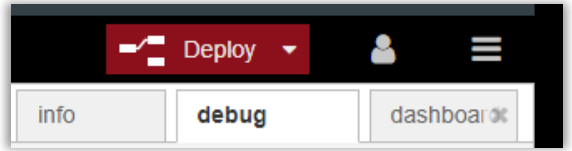
- Connect
- Configure text



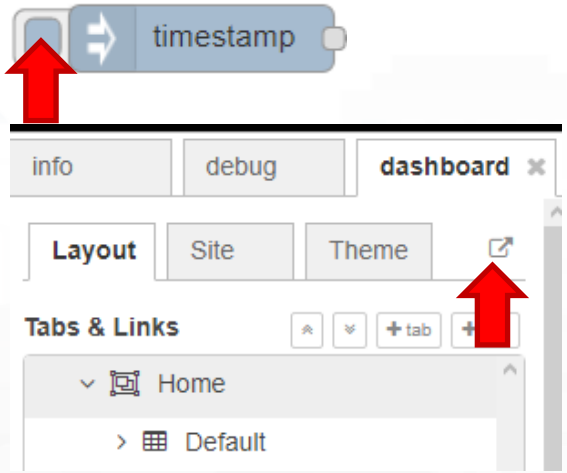
Step 5



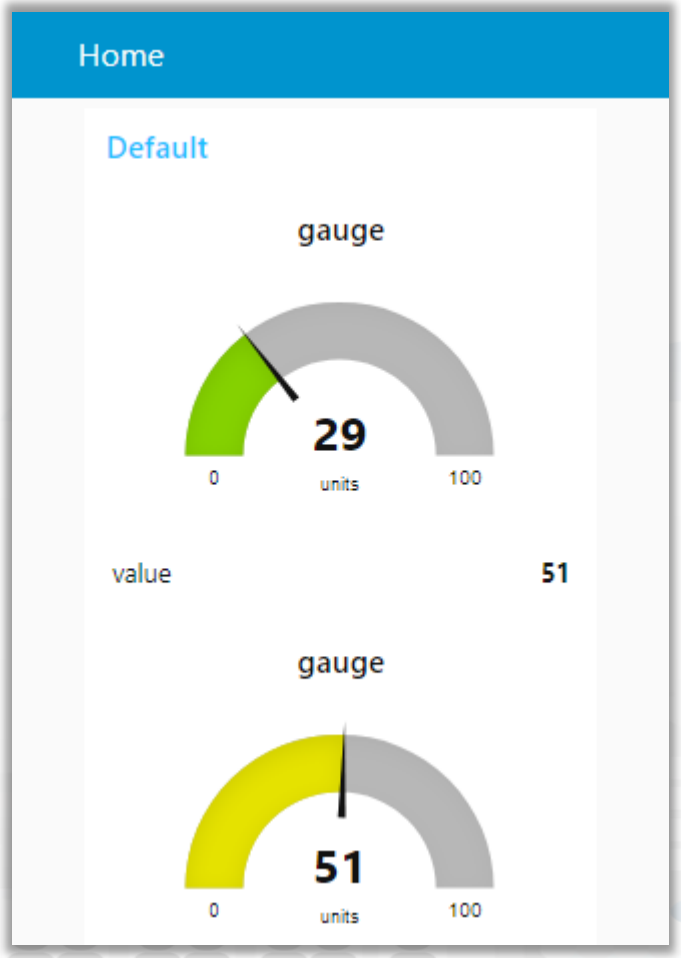
- Deploy



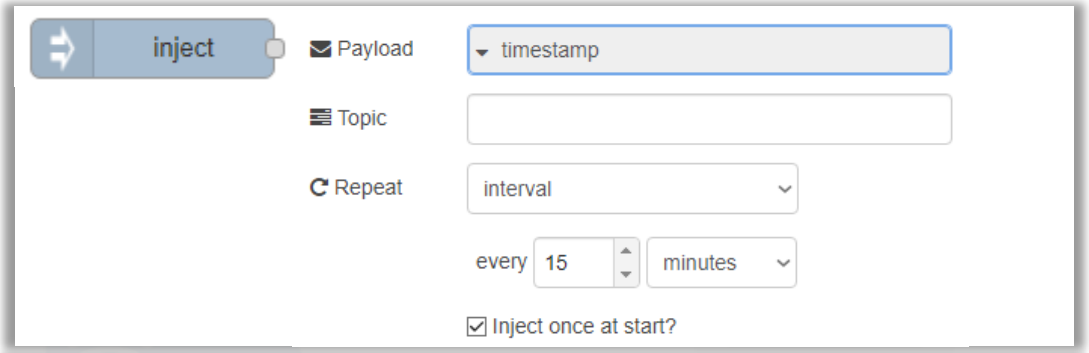
- Click

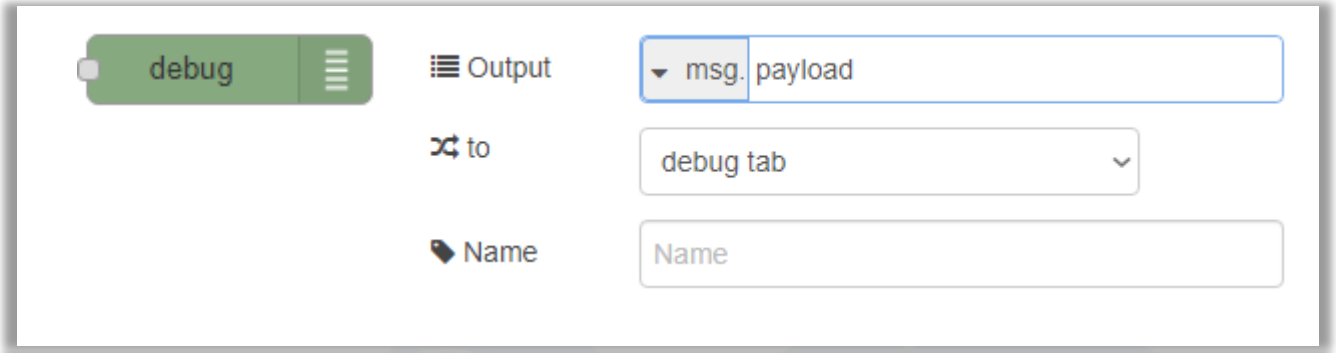


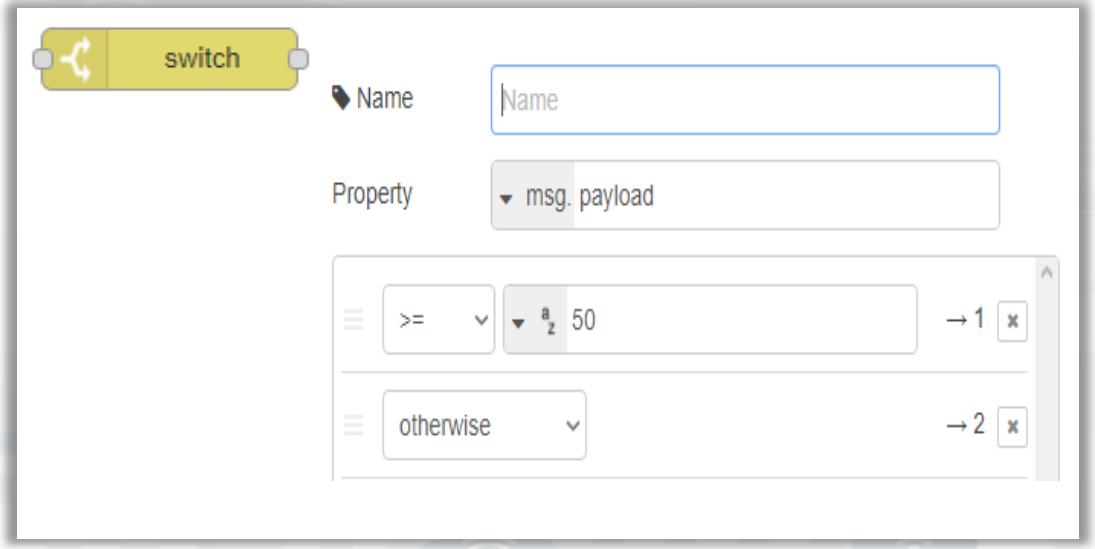
- Click
- Observe

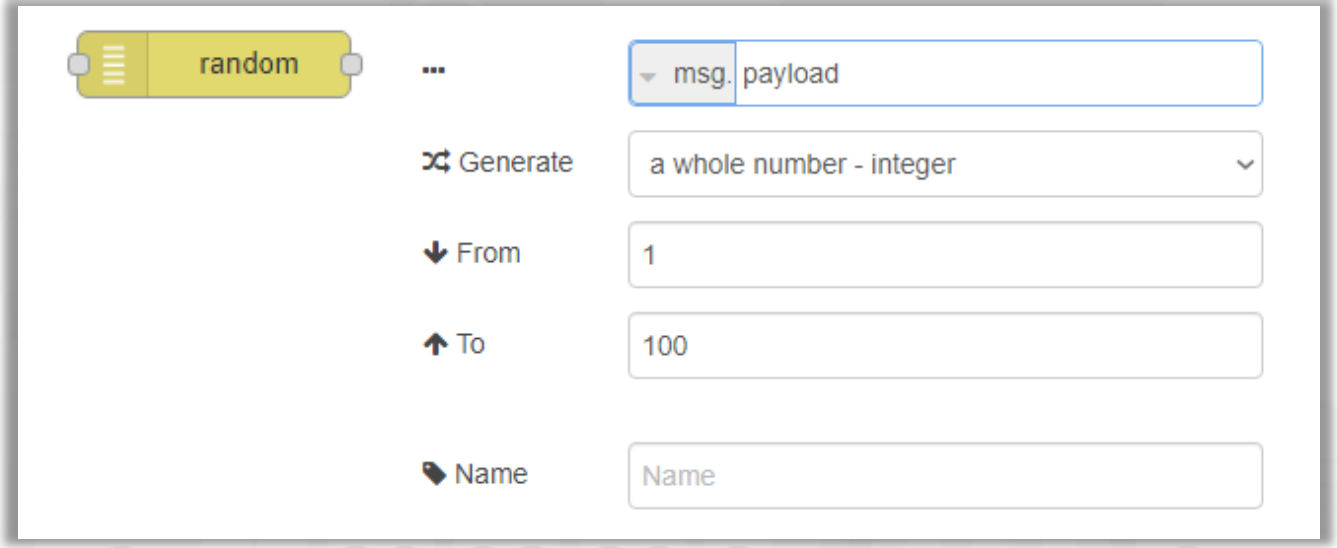


Nodes configuration 1/2

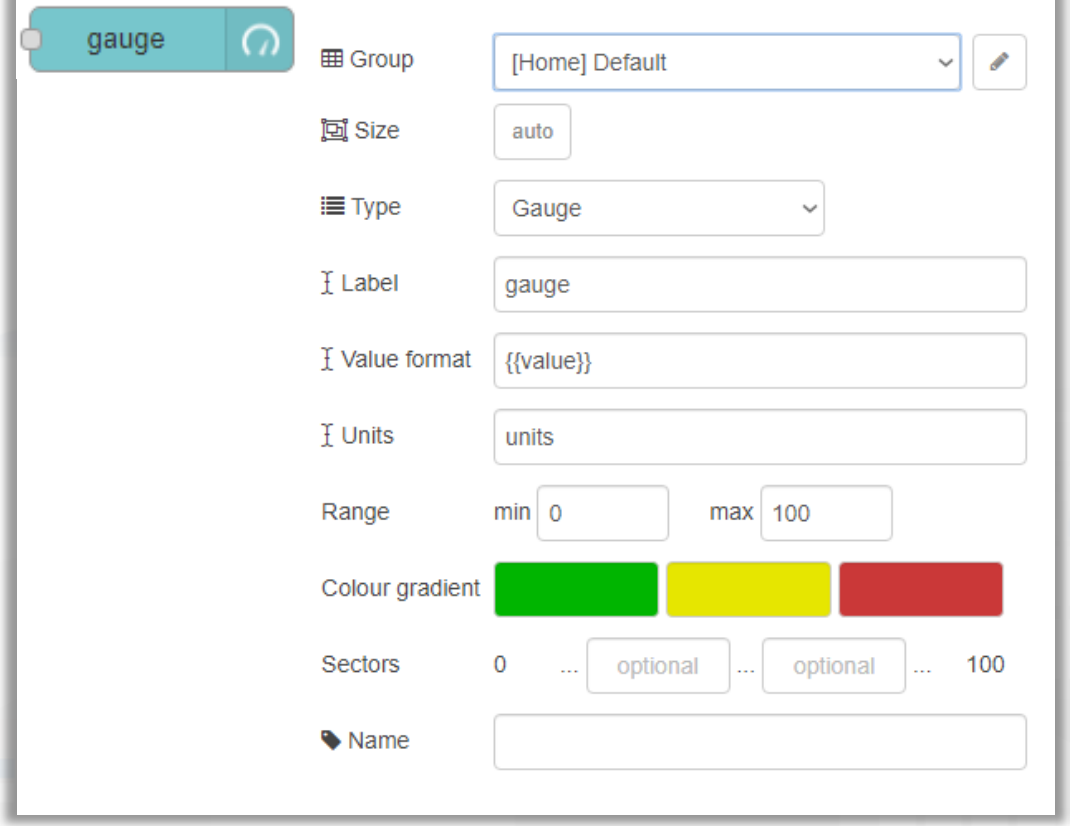


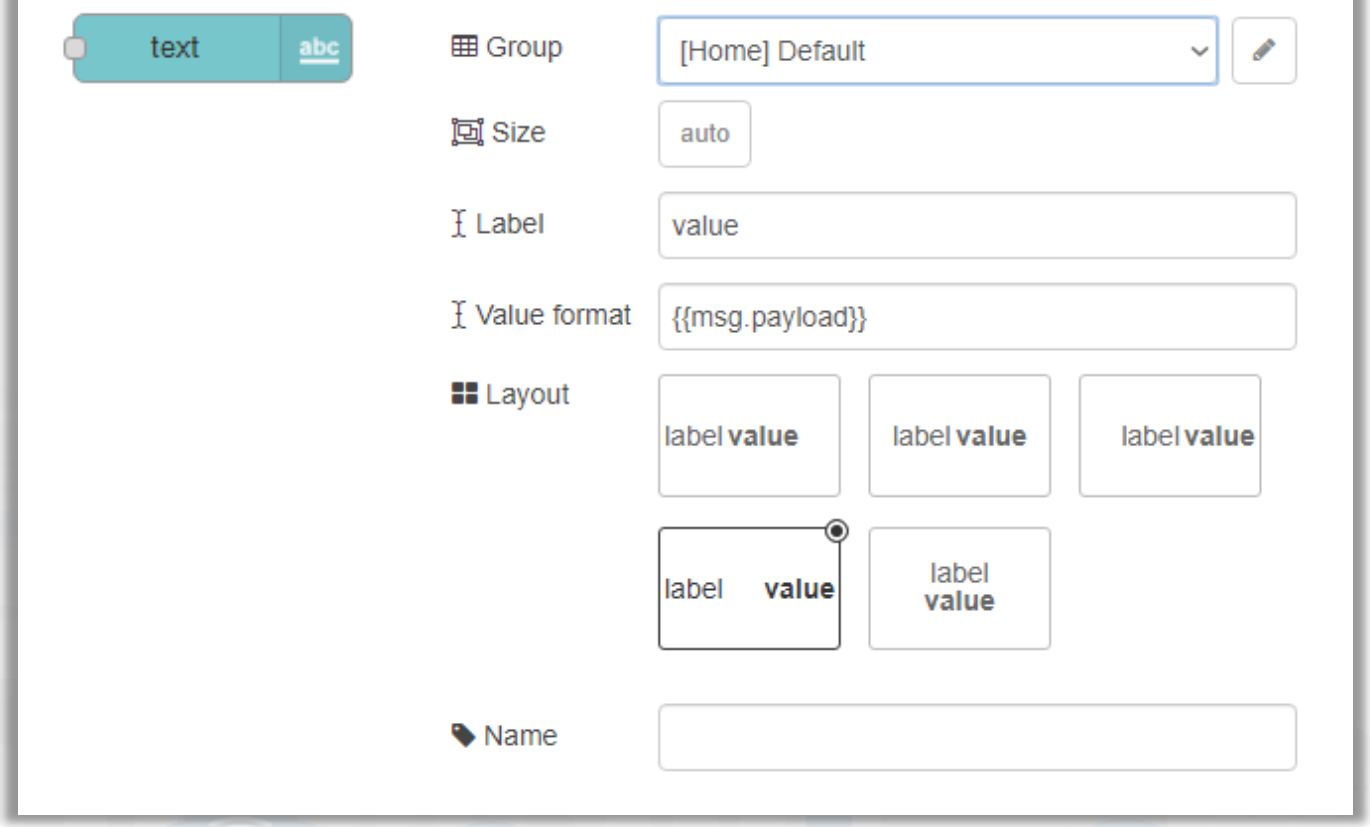




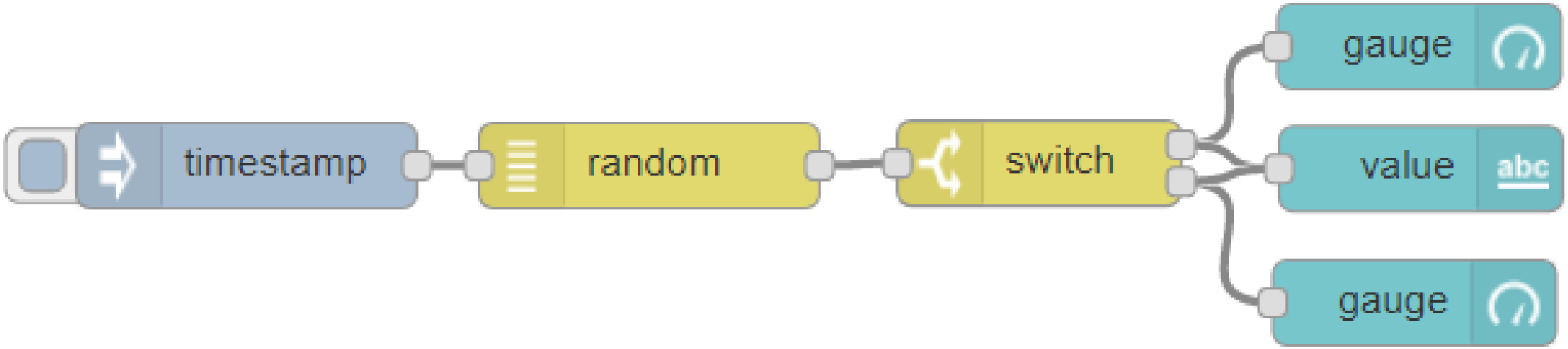


Nodes configuration 2/2

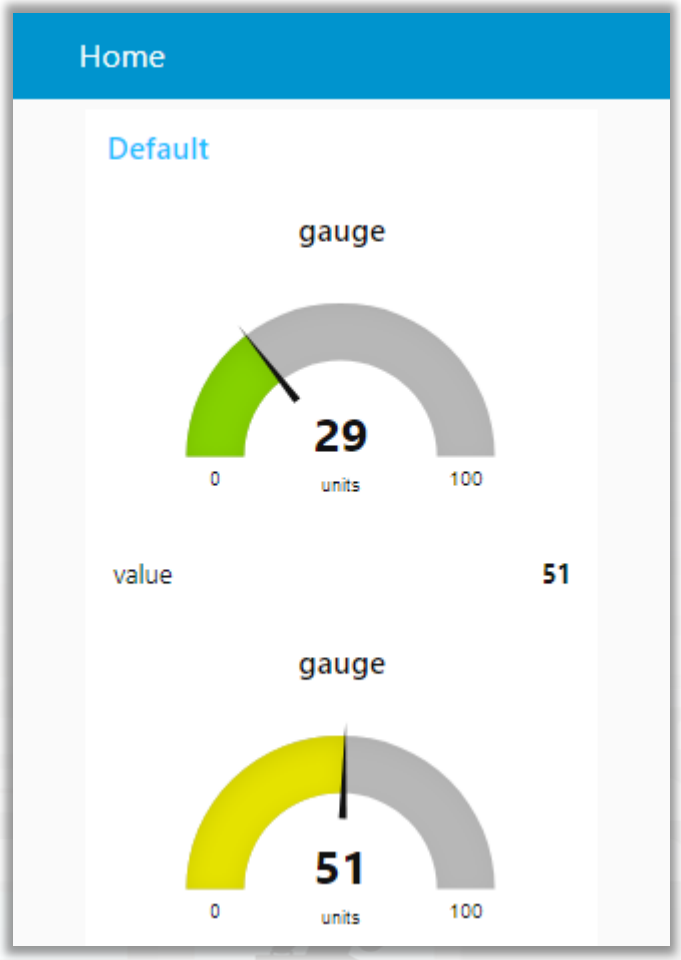




Nodes connections



Resulting Dashboard



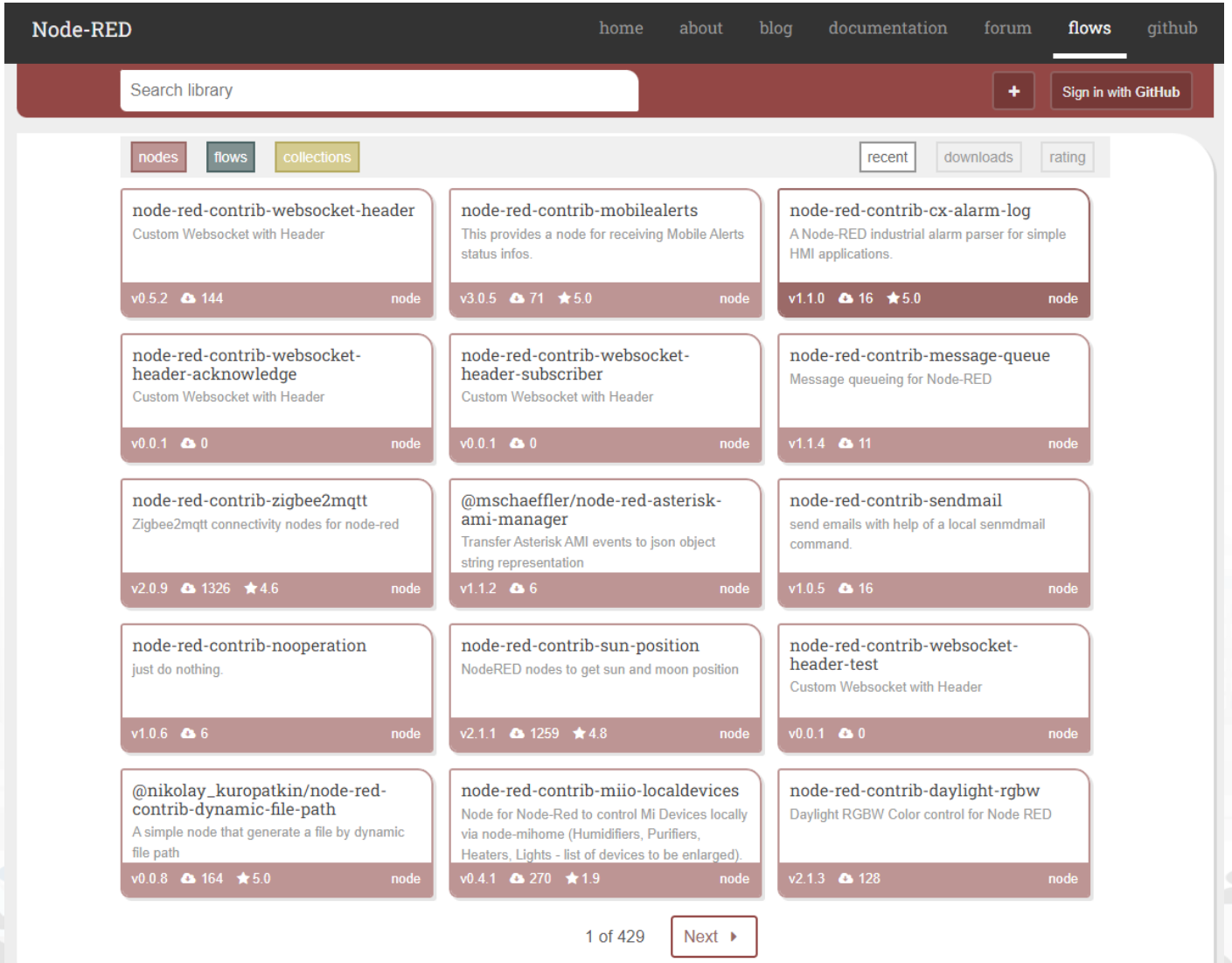
This is a local Node-RED dashboard.
Simple to be created, very limited for many aspects.

Snap4city dashboards are more :

- Powerful
- Flexible
- Secure
- nicer

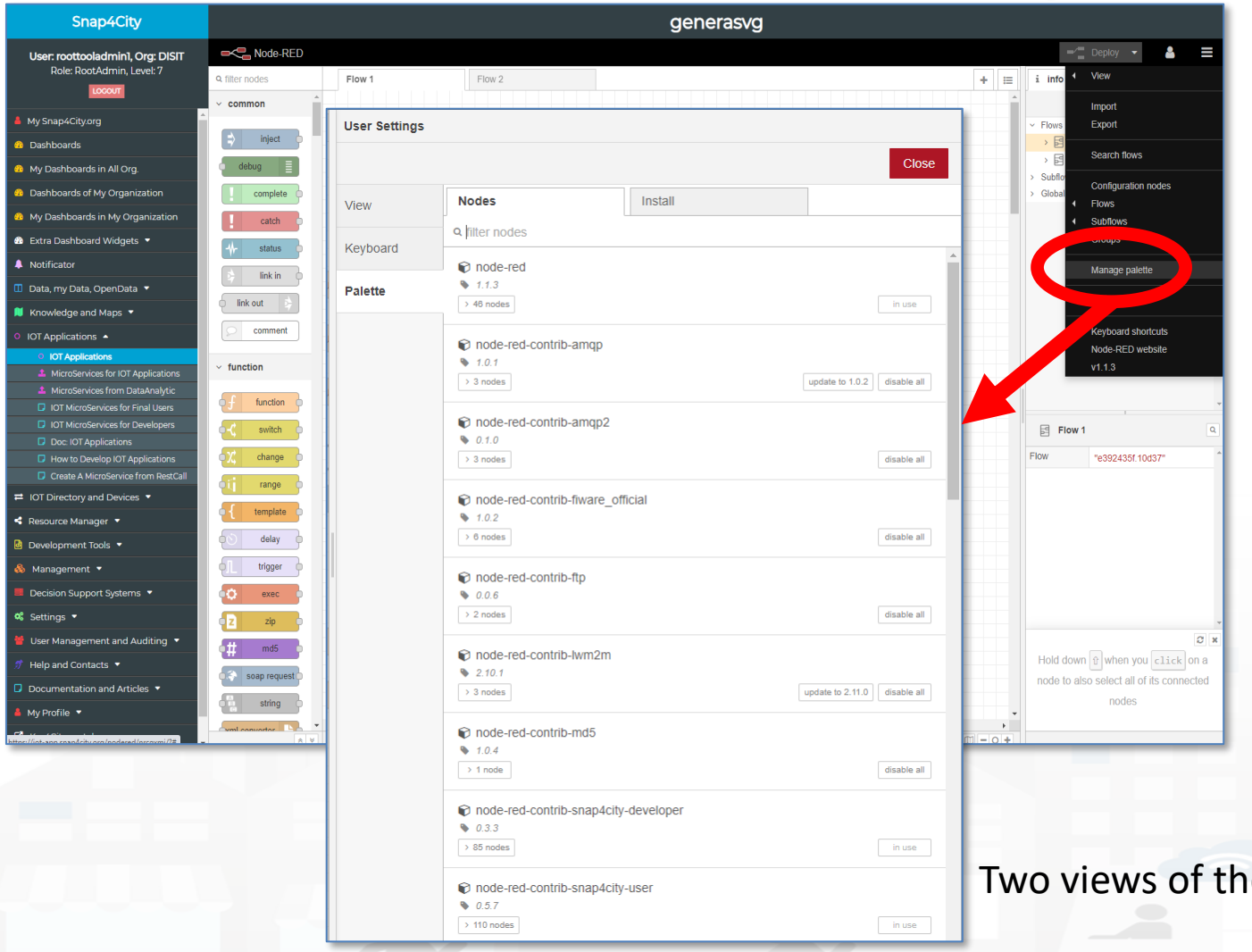
Node-RED Libraries

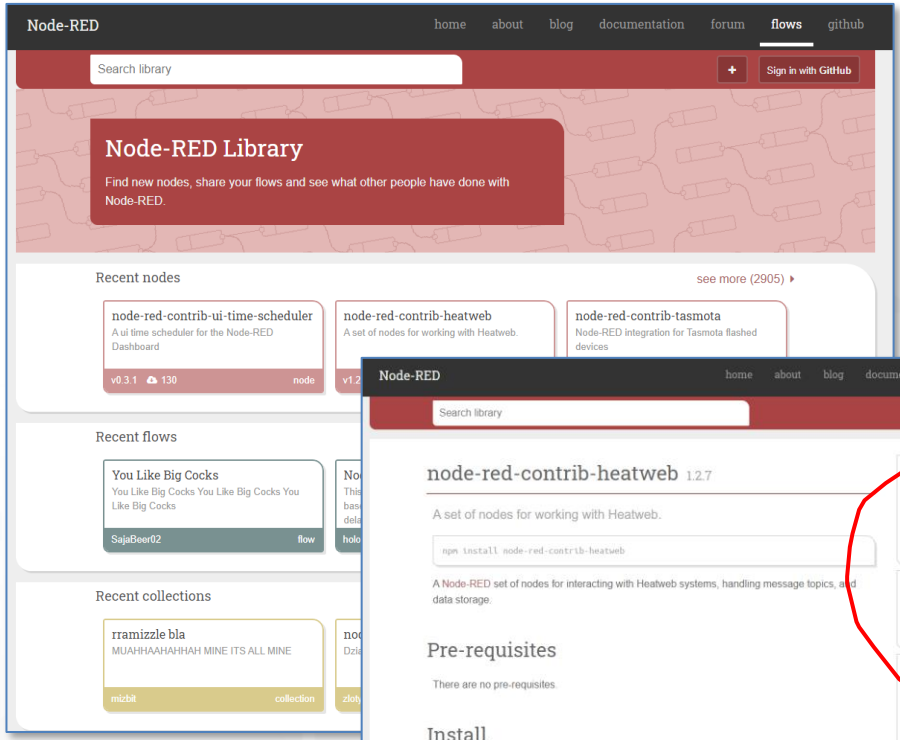
https://flows.nodered.org/search?term=

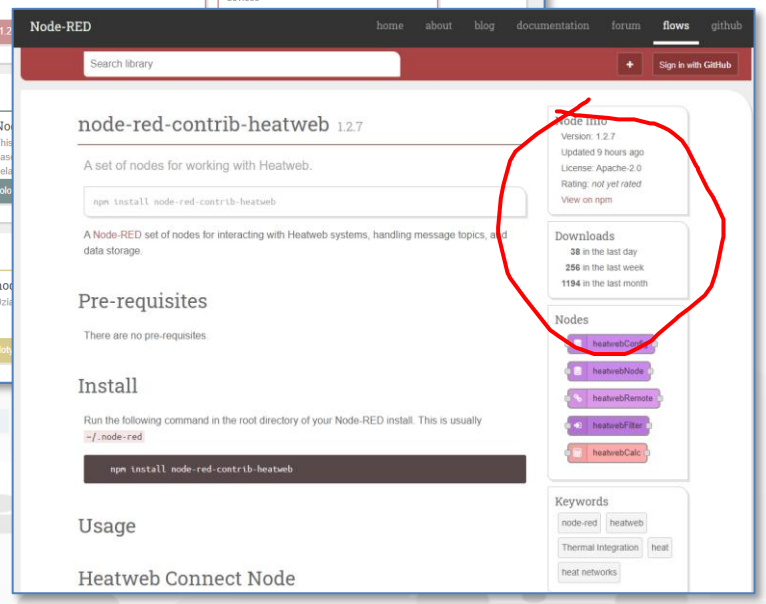


Load Library from Palette

https://flows.nodered.org/







Two views of the same libraries

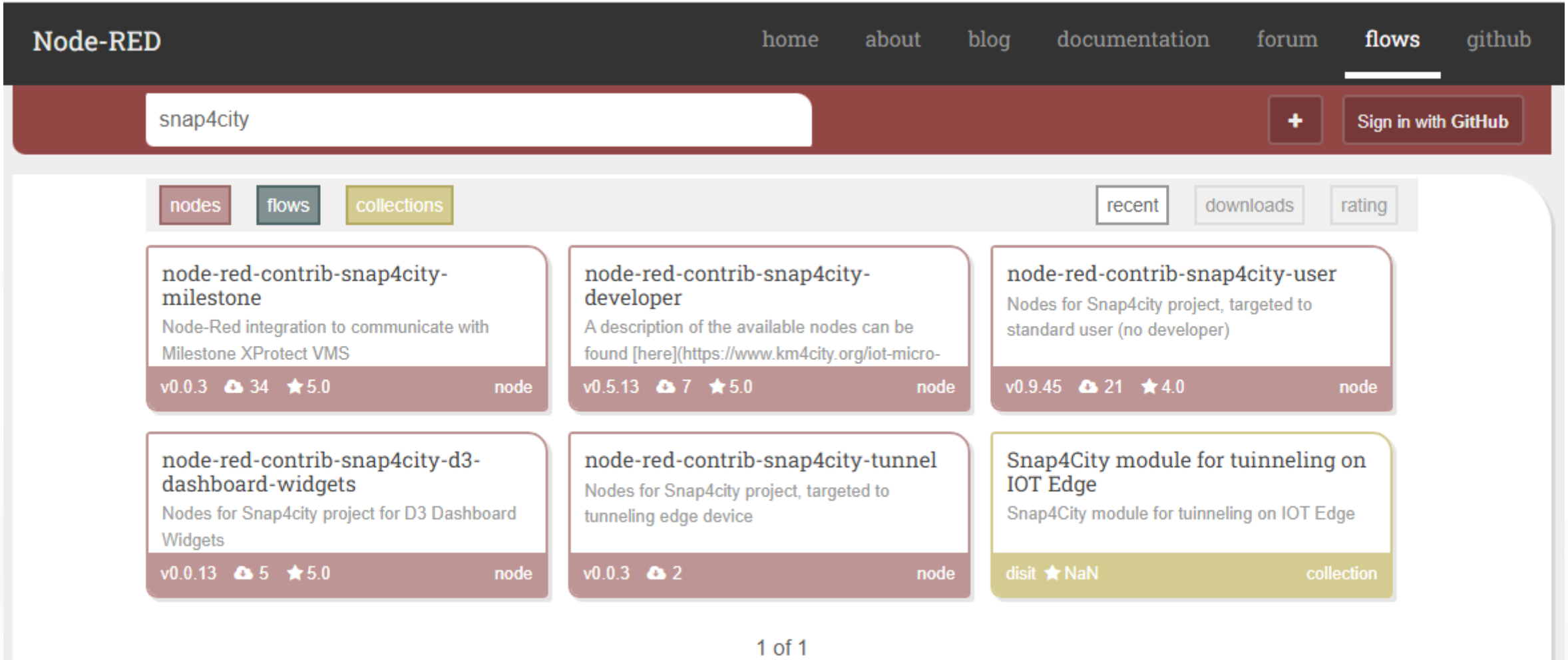
Proc.Logic / IoT App Editor: NODE-RED

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

IoT App / Proc.Logic Editor: NODE-RED

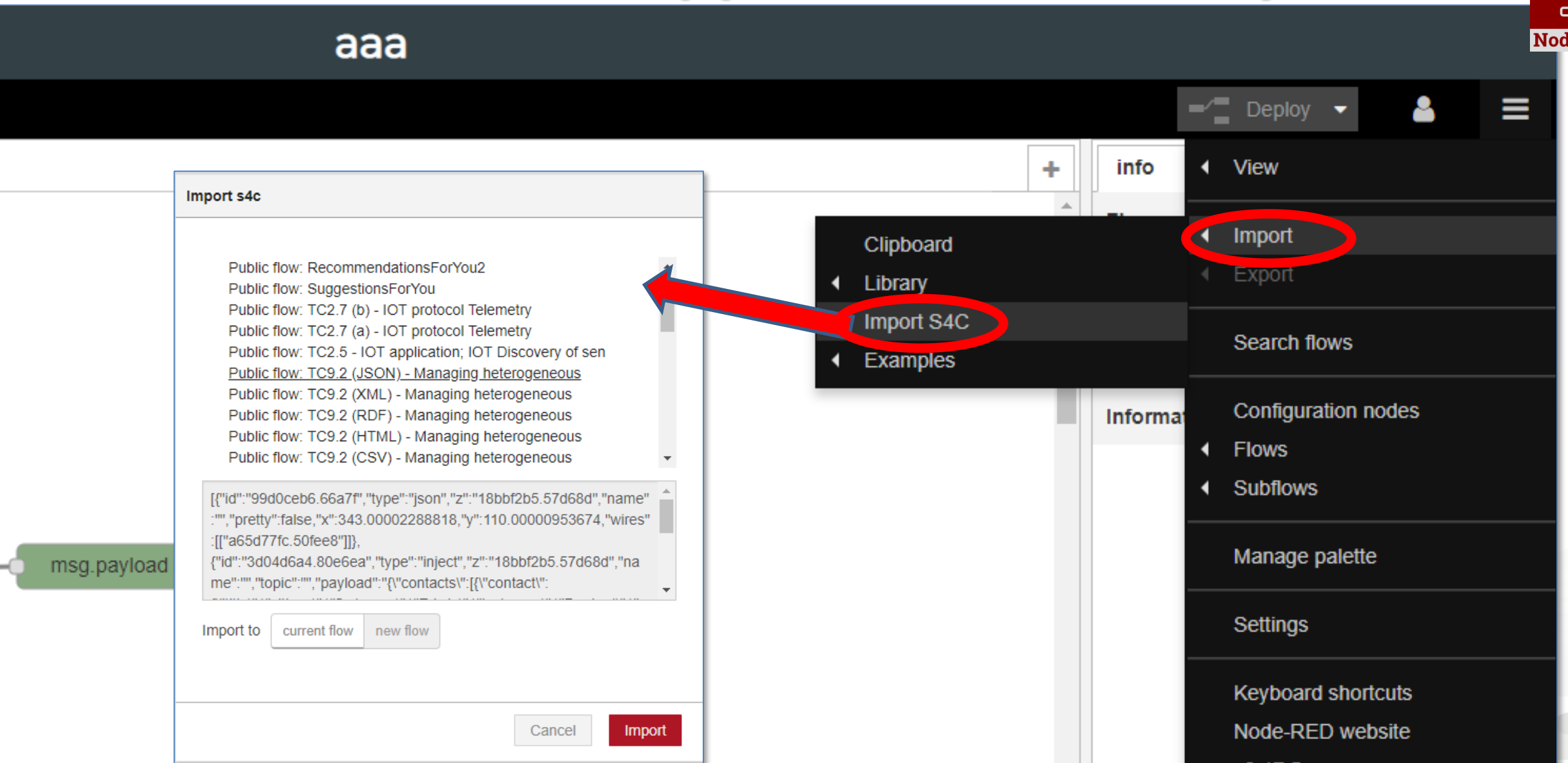
- In the **IoT Apps / Proc.Logic of Snap4City**, it is possible to Extend the Capabilities:
  - Load other Nodes, segments of flow and entire flows from several sources: email, libraries, S4C repository, etc.
  - Load other libraries of MicroServices/Nodes/Blocks from **Manage Palette**
    - A large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

Snap4City Libraries on Node-RED



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

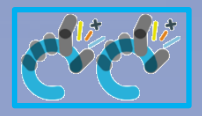
Load an IOT application of example

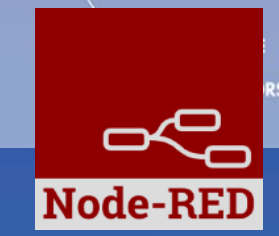


TOP

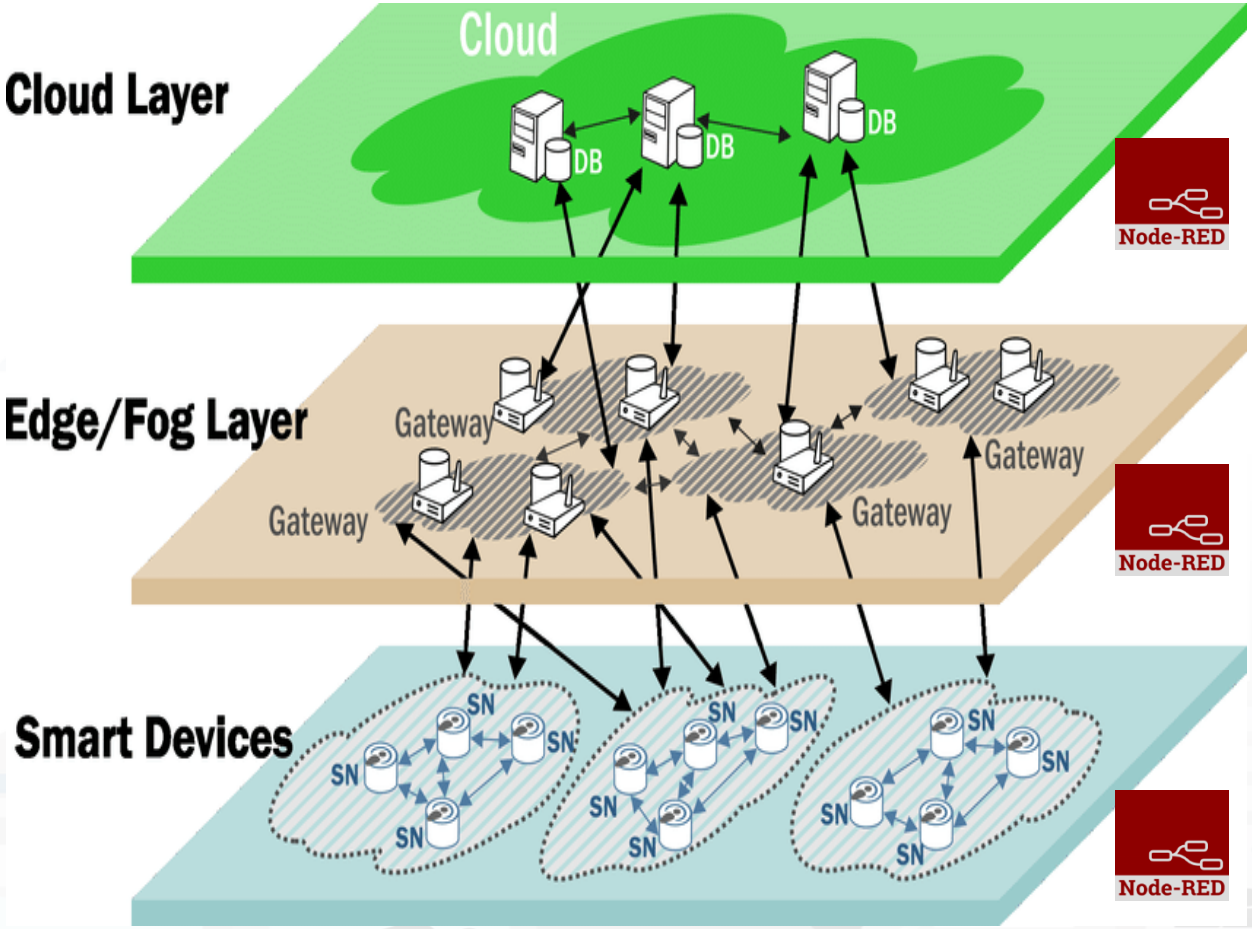
IOT App = Node-RED + Snap4City

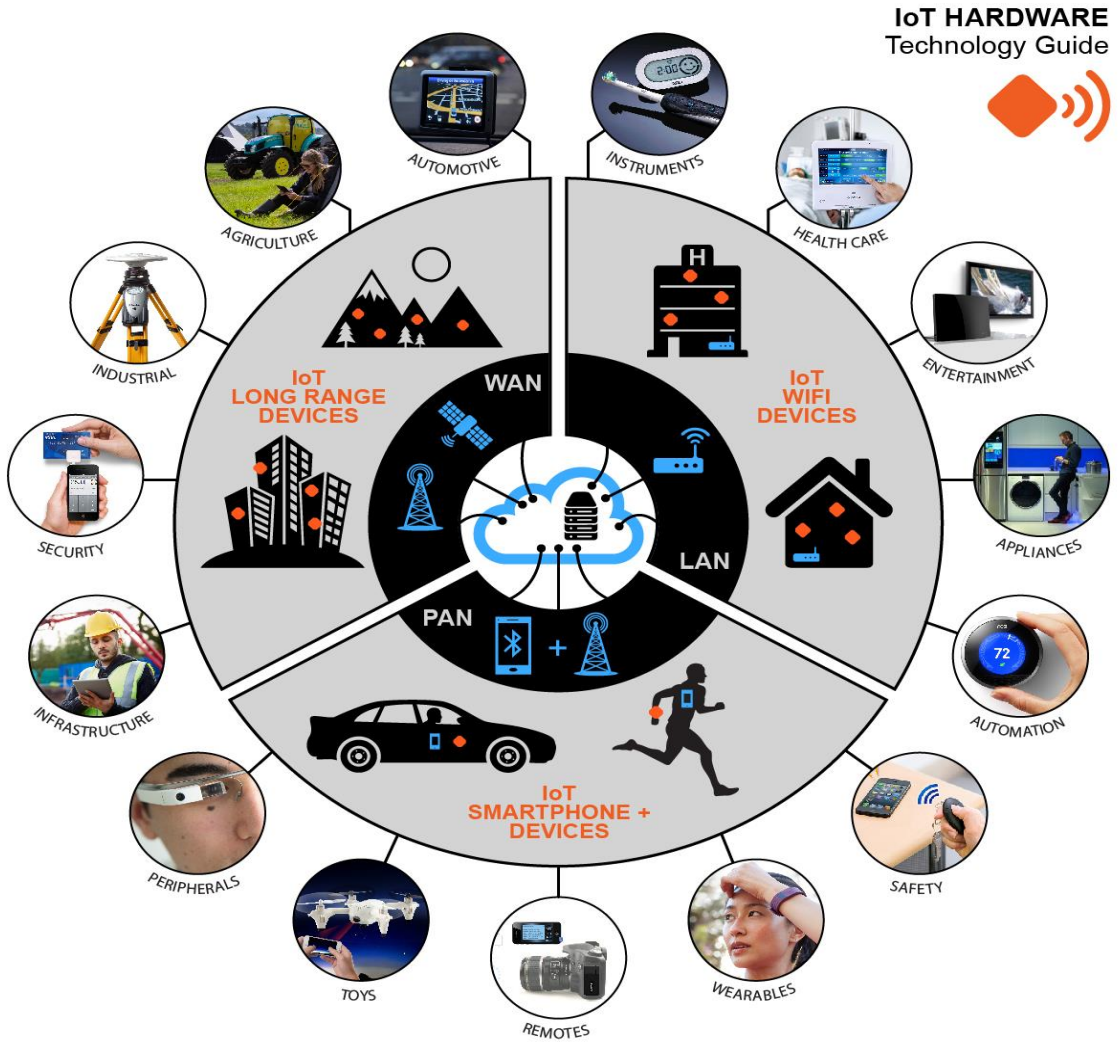
IoT App === Proc.Logic





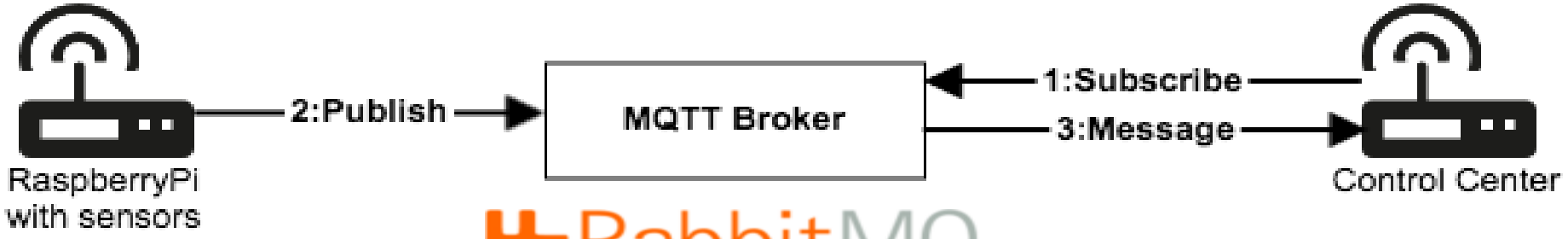
WoT: Cloud vs Fog/Edge Computing





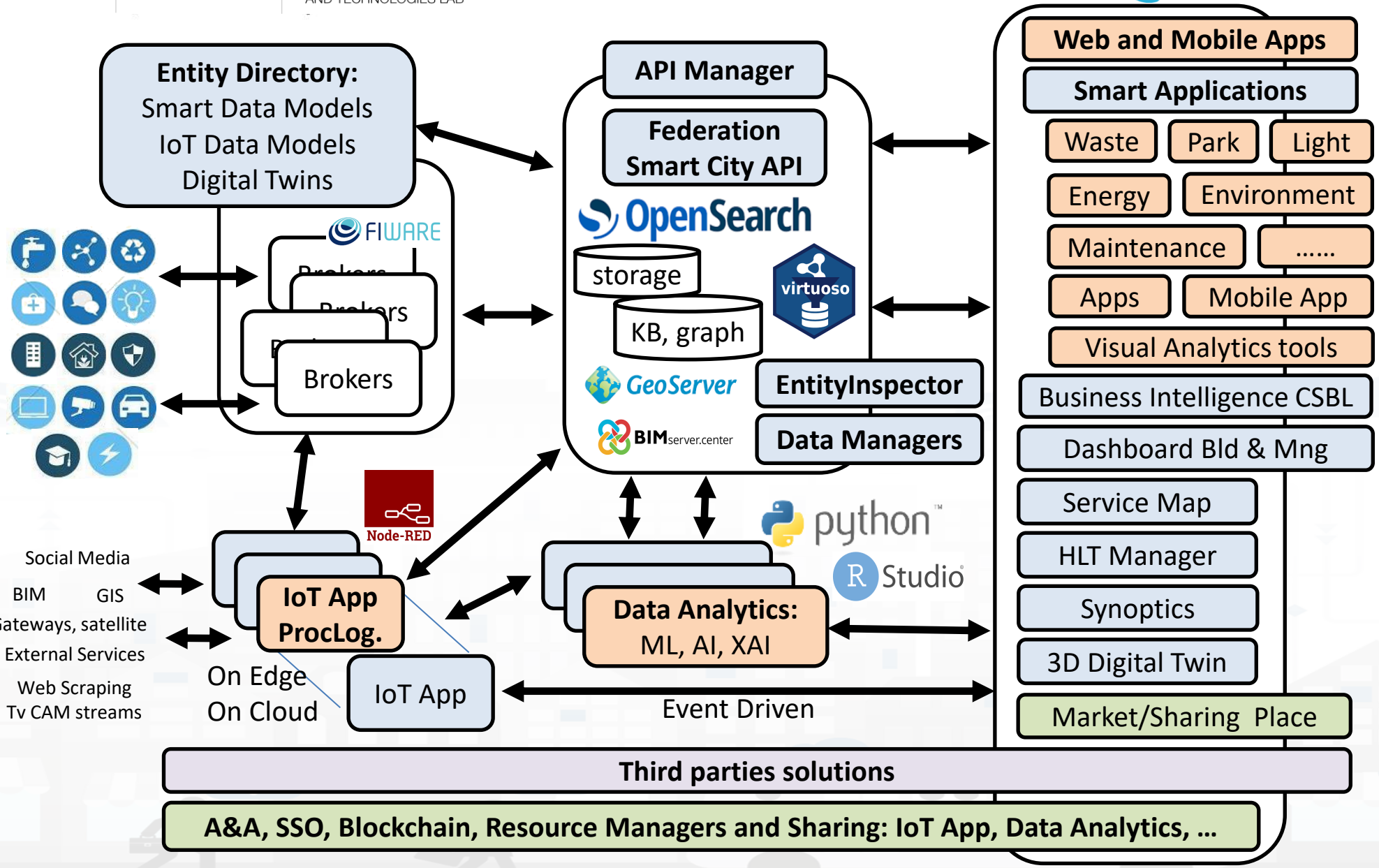
Brokers

	AMQP	STOMP	JMS	COAP	NGSI	MQTT OASIS
RabbitMQ	X	X	X	X		X
Mosquitto						X
ActiveMQ	X	X	X			X
StormMQ	X					
HIVEMQ			X			X
ORION BROKER				X	X	X



RabbitMQ
Open Source Enterprise Messaging

Tech Arch


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

What About IoT Devices, Time Series

IOT Device

Entity: IOT Device

Sends a message

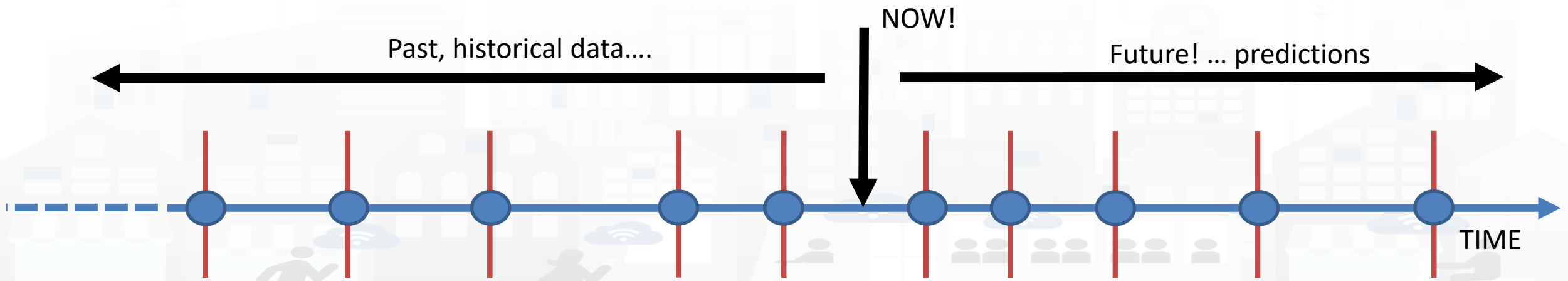
Message (
timestamp: 02-04-2020 at 10:30,
Temperature: 29.34,
Humidity: 35
)

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

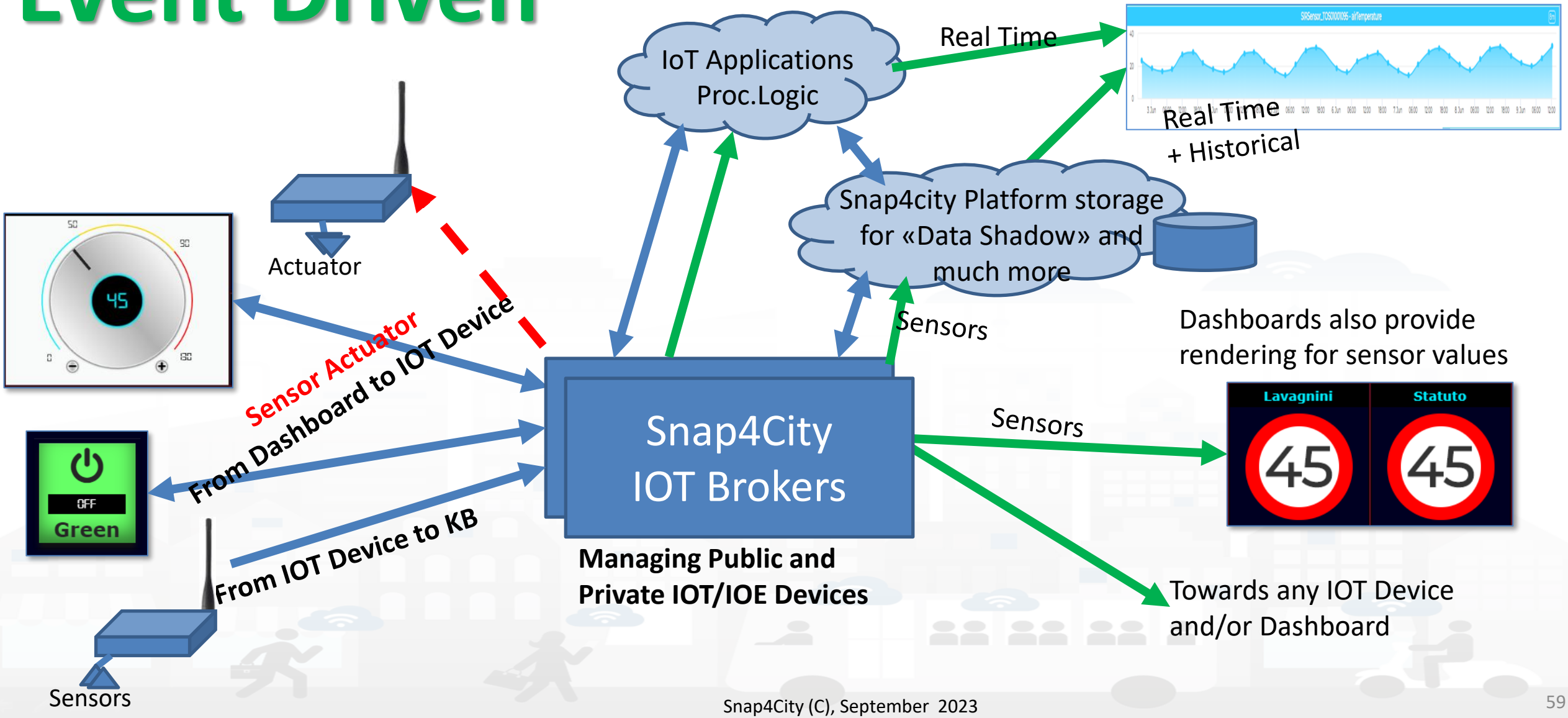
- A set of data coming from an IoT Device with multiple sensor become a time series of values for devices.
  - For example: taking a new measure every 10 minutes (Red Lines)
  - Non regular rates can be valid data as well.
- Each new measure in Snap4City is conventionally time located in «dateObserved»,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

Time Series: they are data streams

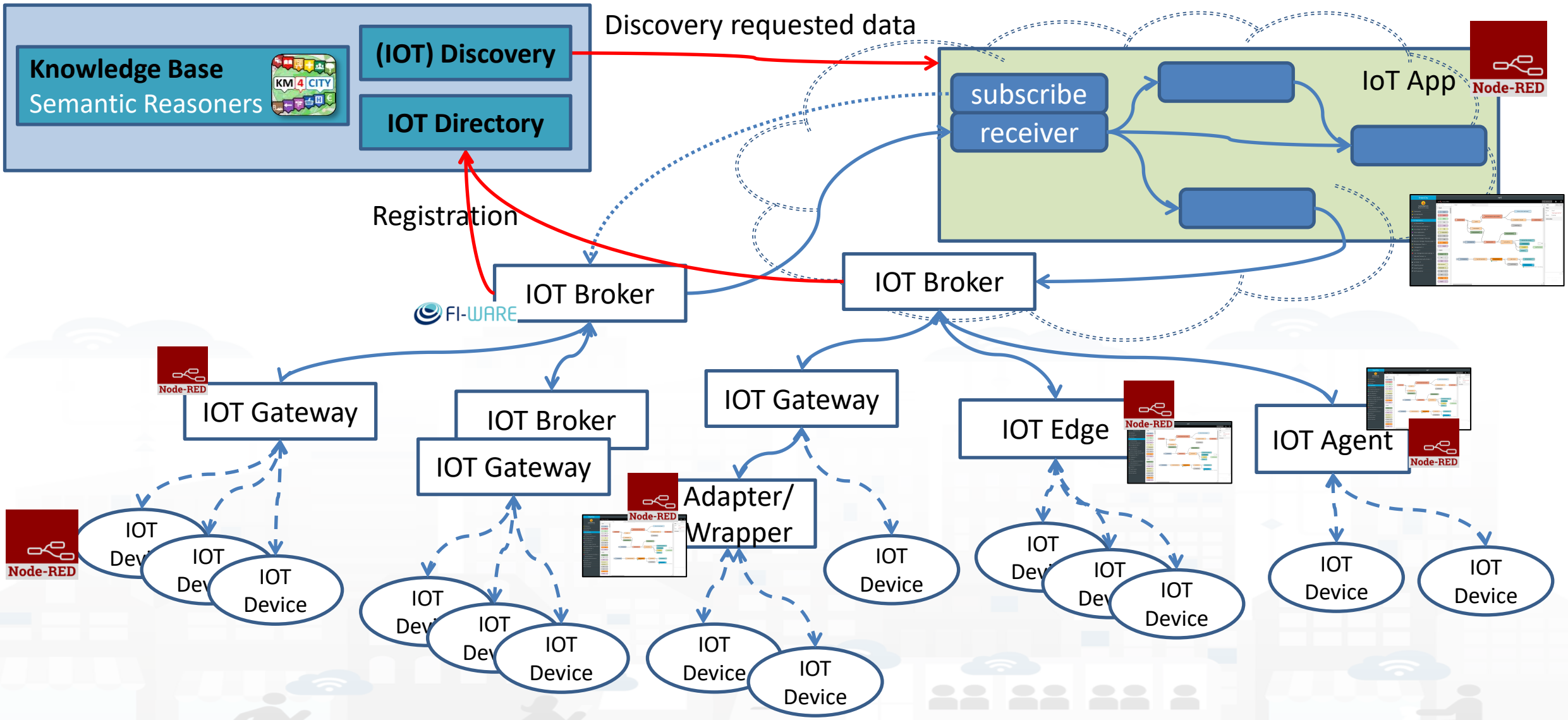
- As soon as you have registered an Entity Instance / IoT Device
  - You are ready to get Future data, may be arriving in PUSH
  - Recall and store historical data as well, but they have to be
    - recalled in PULL with some IoT App.
    - Loaded in PULL with some File or Data Table Loader



Event Driven

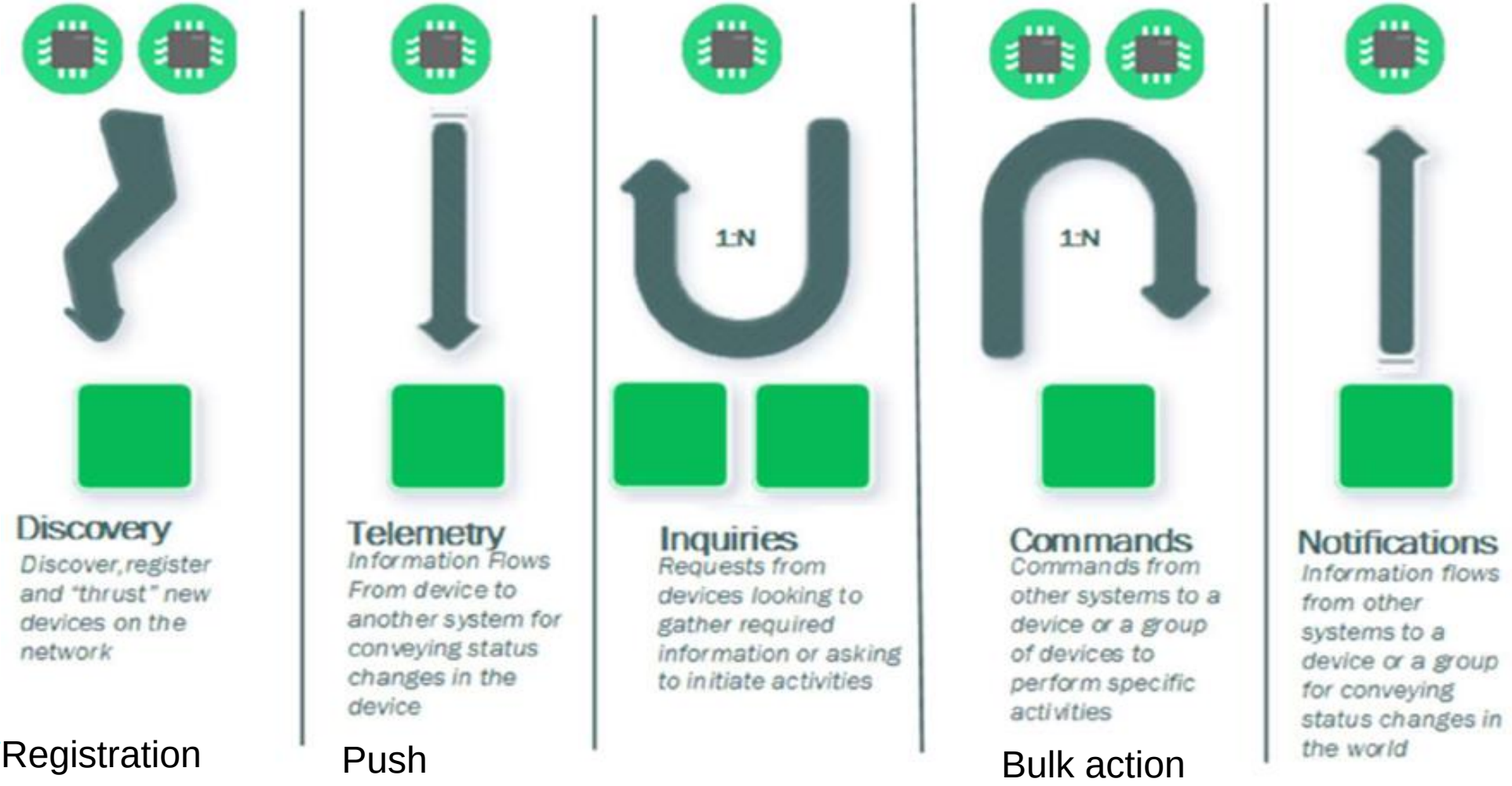


IoT Network


Knowledge Base
Semantic Reasoners
(IOT) Discovery
IOT Directory
Discovery requested data
Registration
subscribe
receiver
IoT App
Node-RED
FI-WARE
IOT Broker
IOT Broker
IOT Gateway
IOT Broker
IOT Gateway
IOT Gateway
Adapter/
Wrapper
IOT Edge
IOT Agent
IOT Device

IoT/IoE Protocols

Communication Patterns



- MQTT
- HTTP(s)
- AMQP
- COAP
- NGSI
- OneM2M
- WebSockets
- ……..
- Etc.

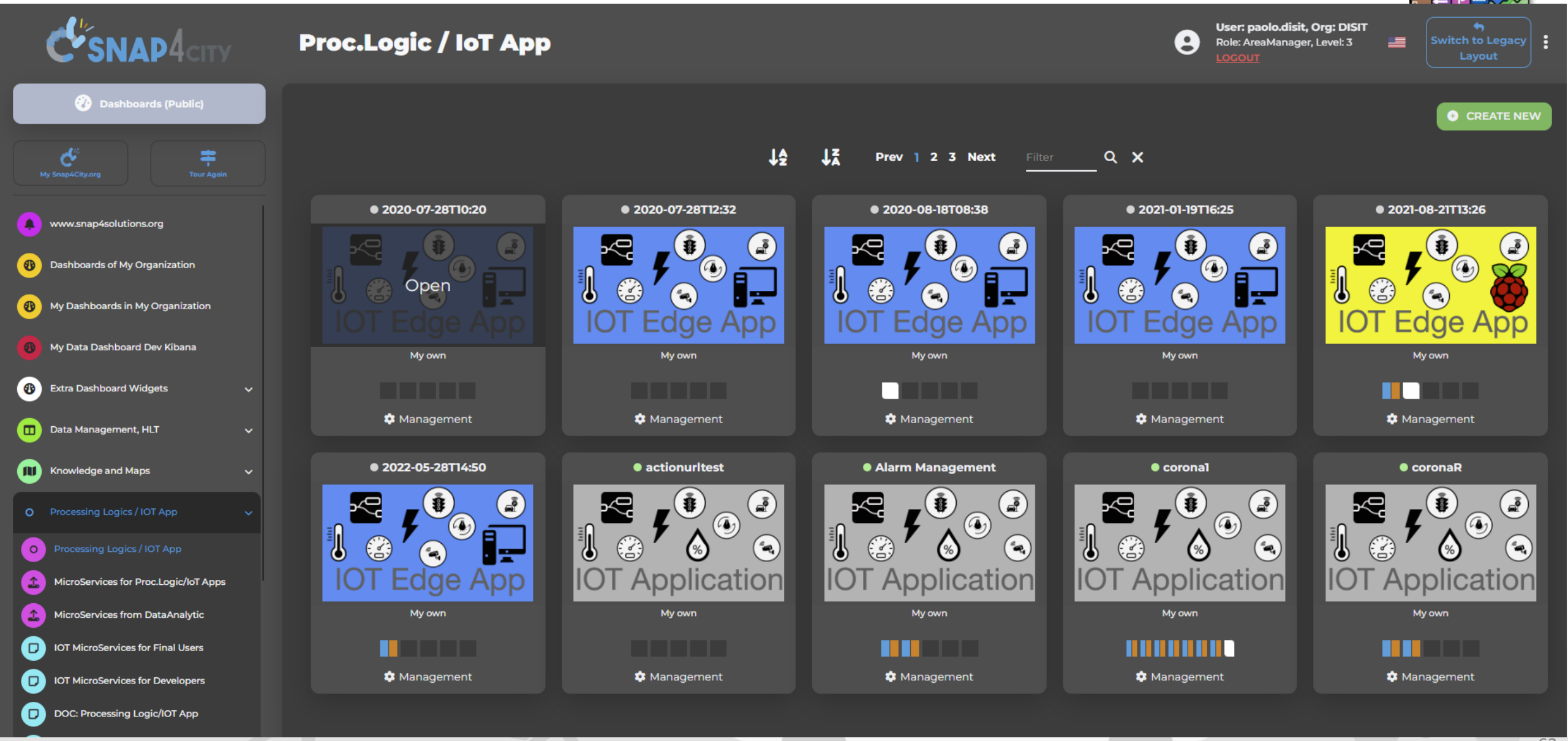
Registration

Push

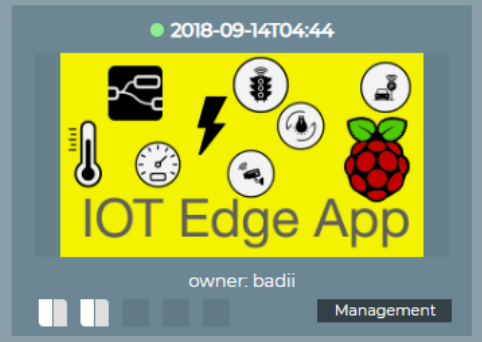
Bulk action

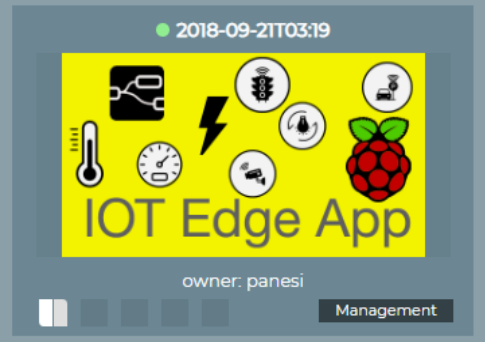
Note on Communication patterns

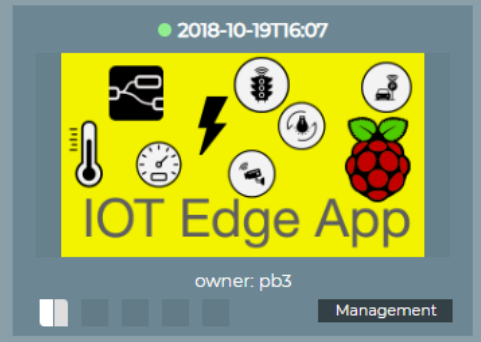
- Not all Communication Patterns are supported by all Protocols
- Not all Communication Patterns are supported by all Platforms
- Protocols implement Patters, + formats, + sequences, etc.
- They are referred at level of communications
  - IoT Device ← → IoT Gateway ←→IoT Broker
- IoT Protocols mostly used at level of IP are:
  - NGSI V1/2, MQTT, COAP, AMQP, OneM2M, WS, ModBUS,
- Radio protocols are: Lora, ZigBee, 3G, Wi-Fi, etc.
- Formats: JSON, Geo-JSON, Linked Data, XML, CSV,

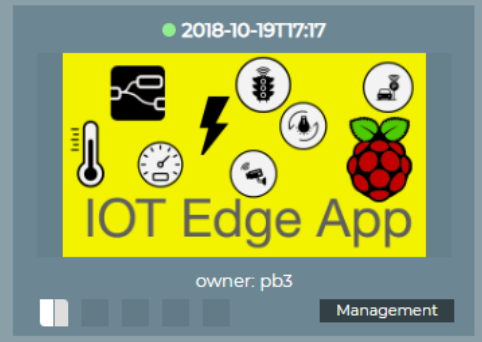

SNAP4CITY
Proc.Logic / IoT App
User: paolo.disit, Org: DISIT
Role: AreaManager, Level: 3
LOGOUT
Switch to Legacy Layout
Dashboards (Public)
My Snap4City.org
Tour Again
www.snap4solutions.org
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
Extra Dashboard Widgets
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Processing Logics / IOT App
MicroServices for Proc.Logic/IoT Apps
MicroServices from DataAnalytic
IOT MicroServices for Final Users
IOT MicroServices for Developers
DOC: Processing Logic/IOT App
CREATE NEW
Prev 1 2 3 Next
Filter
2020-07-28T10:20
Open
IOT Edge App
My own
Management
2020-07-28T12:32
IOT Edge App
My own
Management
2020-08-18T08:38
IOT Edge App
My own
Management
2021-01-19T16:25
IOT Edge App
My own
Management
2021-08-21T13:26
IOT Edge App
My own
Management
2022-05-28T14:50
IOT Edge App
My own
Management
actionurltest
IOT Application
My own
Management
Alarm Management
IOT Application
My own
Management
corona1
IOT Application
My own
Management
coronaR
IOT Application
My own
Management

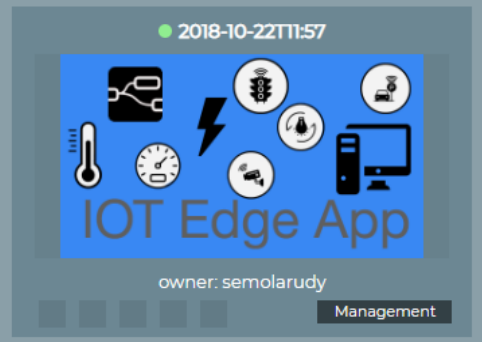
IOT Applications

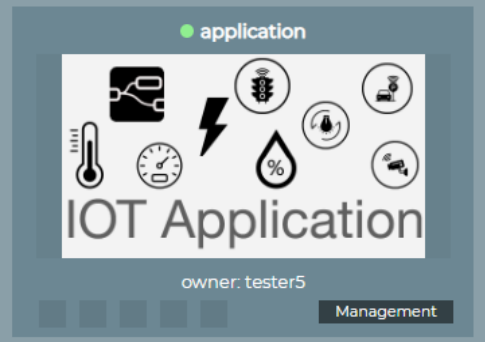


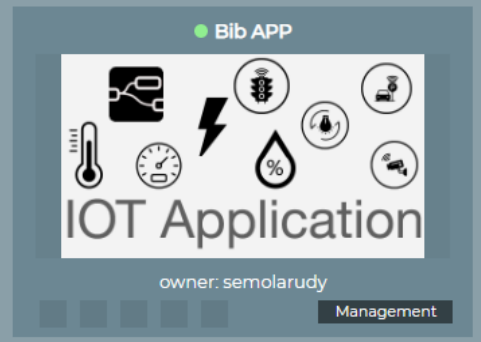


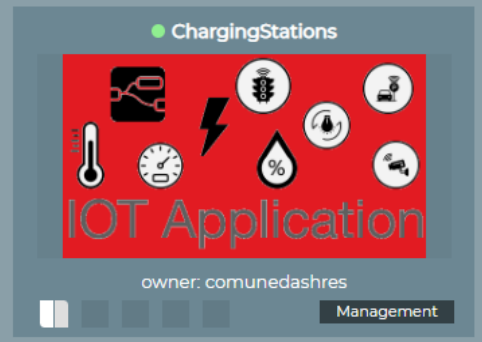


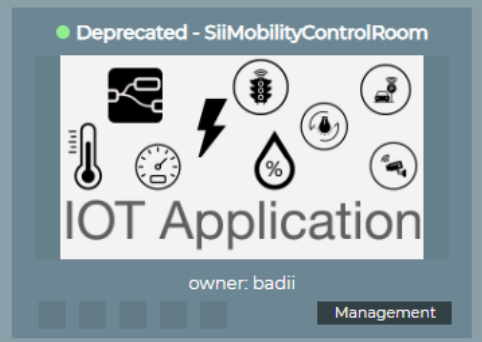


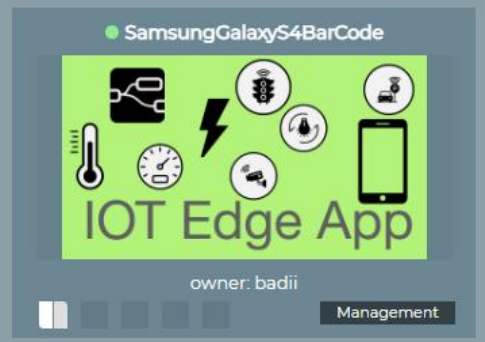


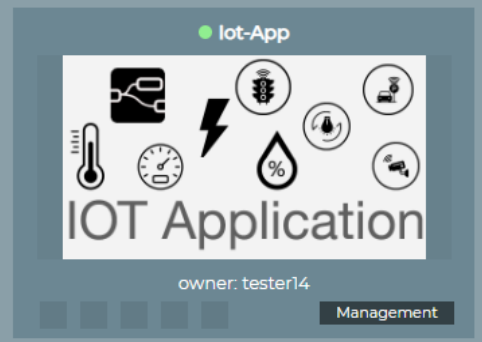






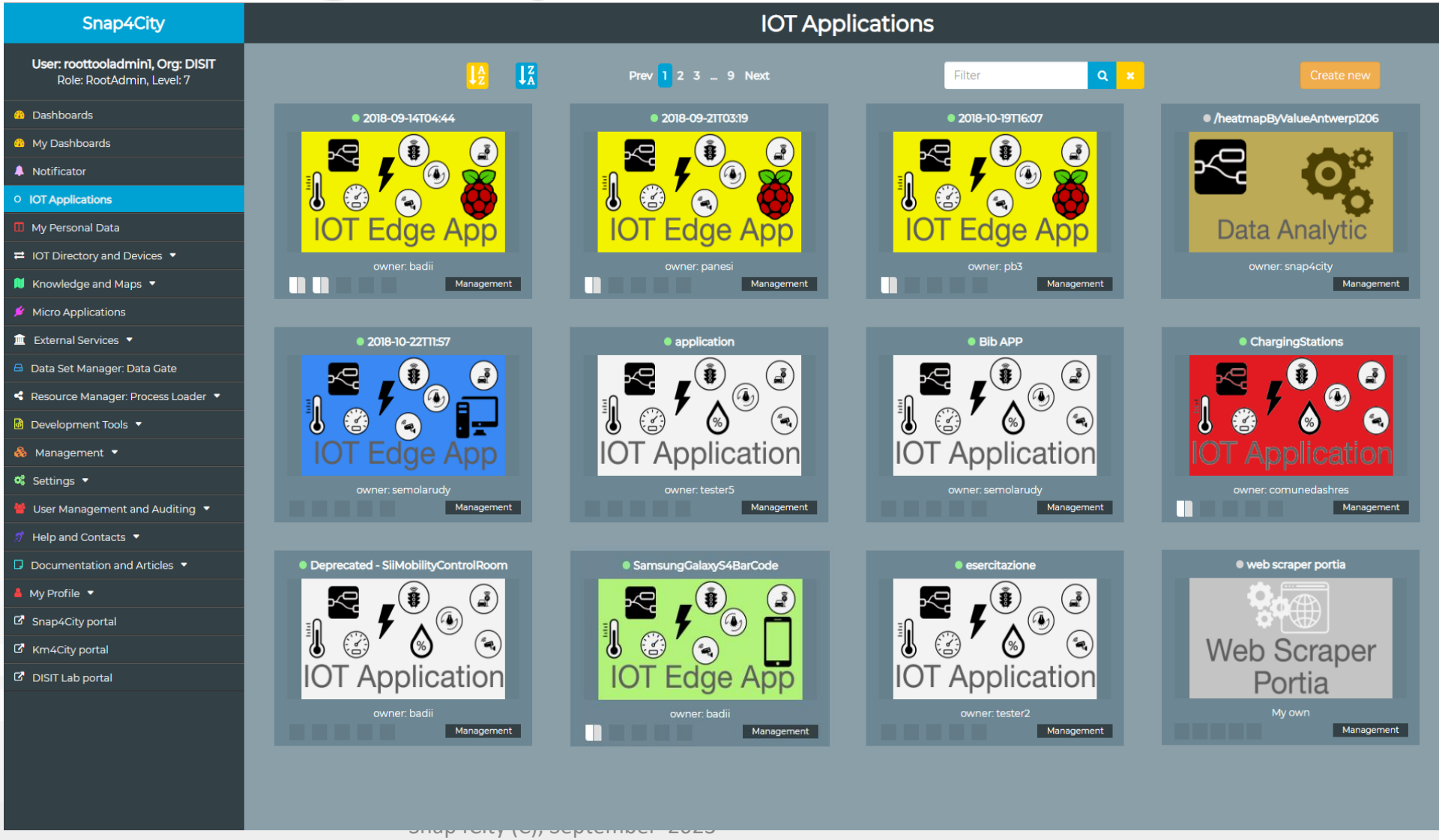






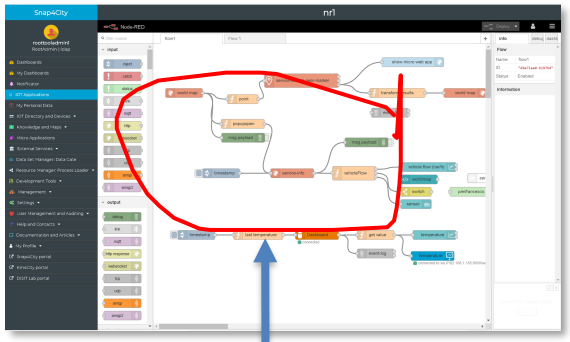
IOT Application Listing, they can be

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

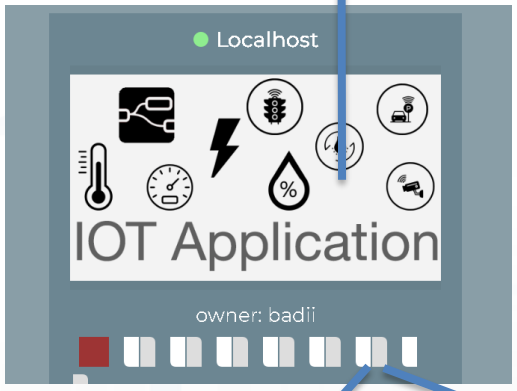


IOT Applications Listing

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

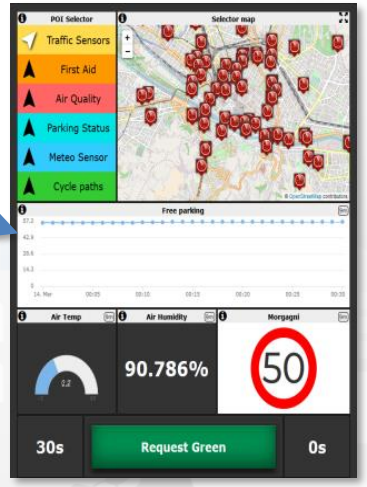


EDIT IOT APP



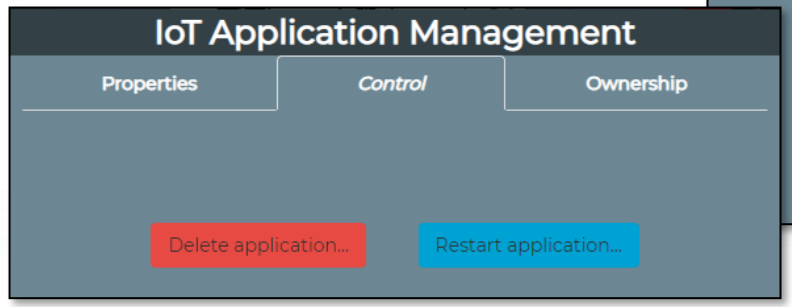
VIEW

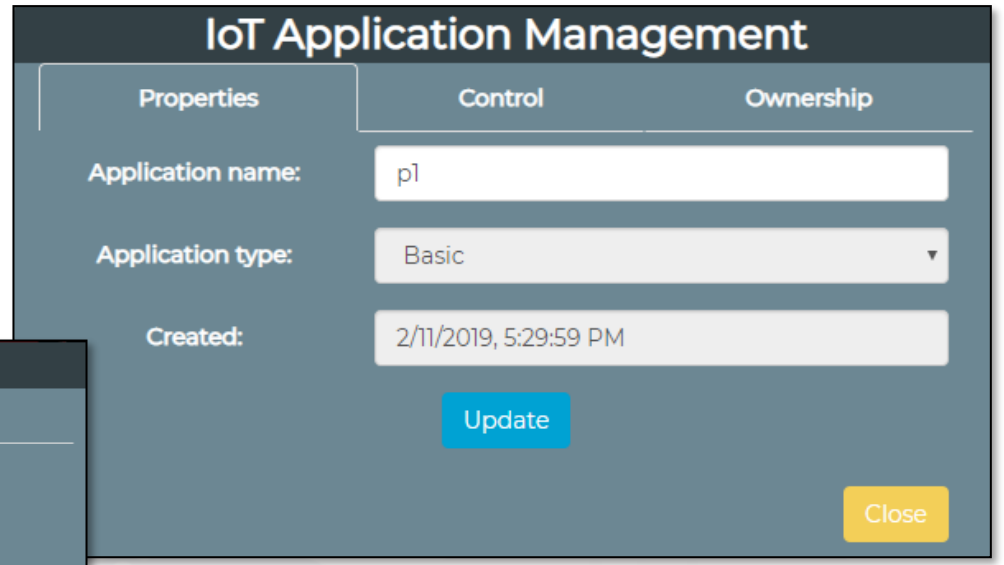
EDIT

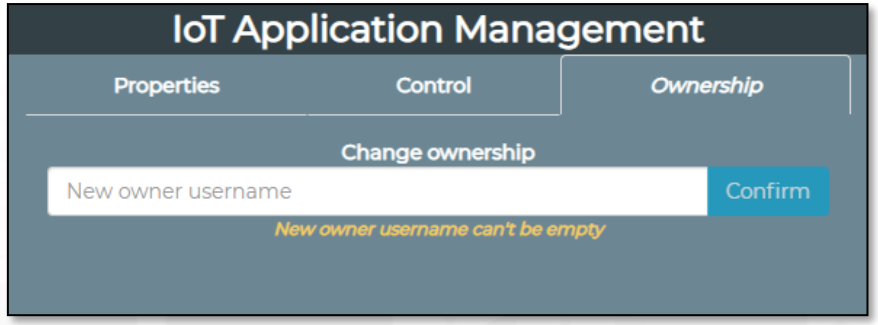


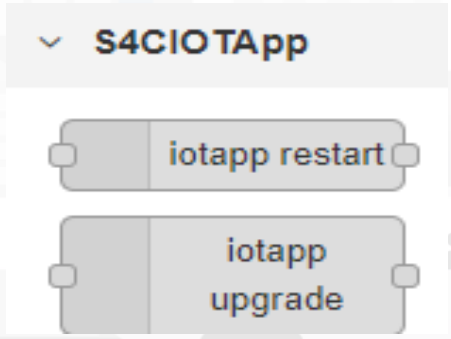
IOT Application Self Control

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







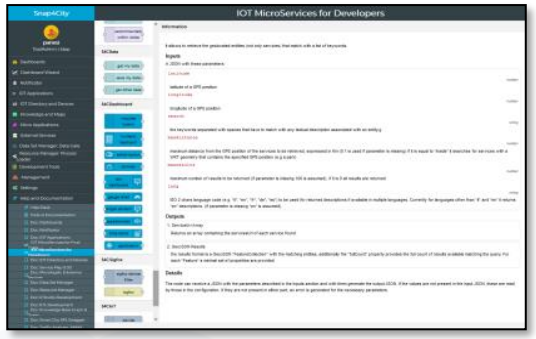


Automating restart and update

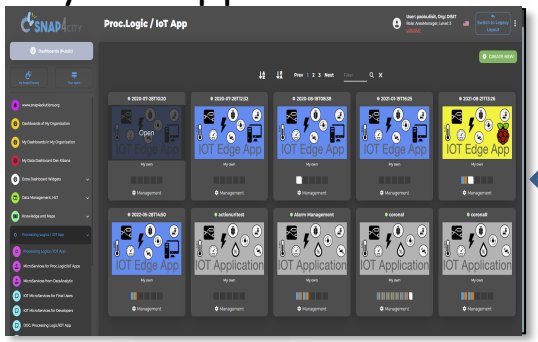
IOT Applications Development

IOT Discovering

MicroServices collections

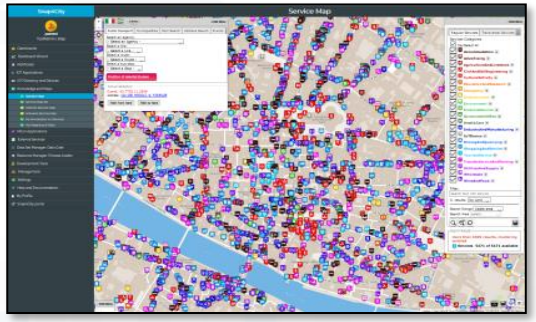


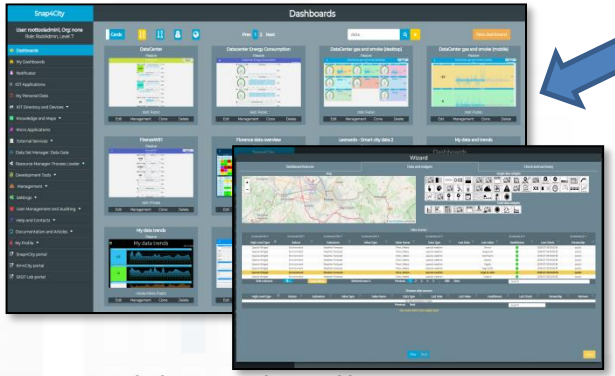
My IOT Applications



IOT App. Editor

Generating IOT App With Dashboard



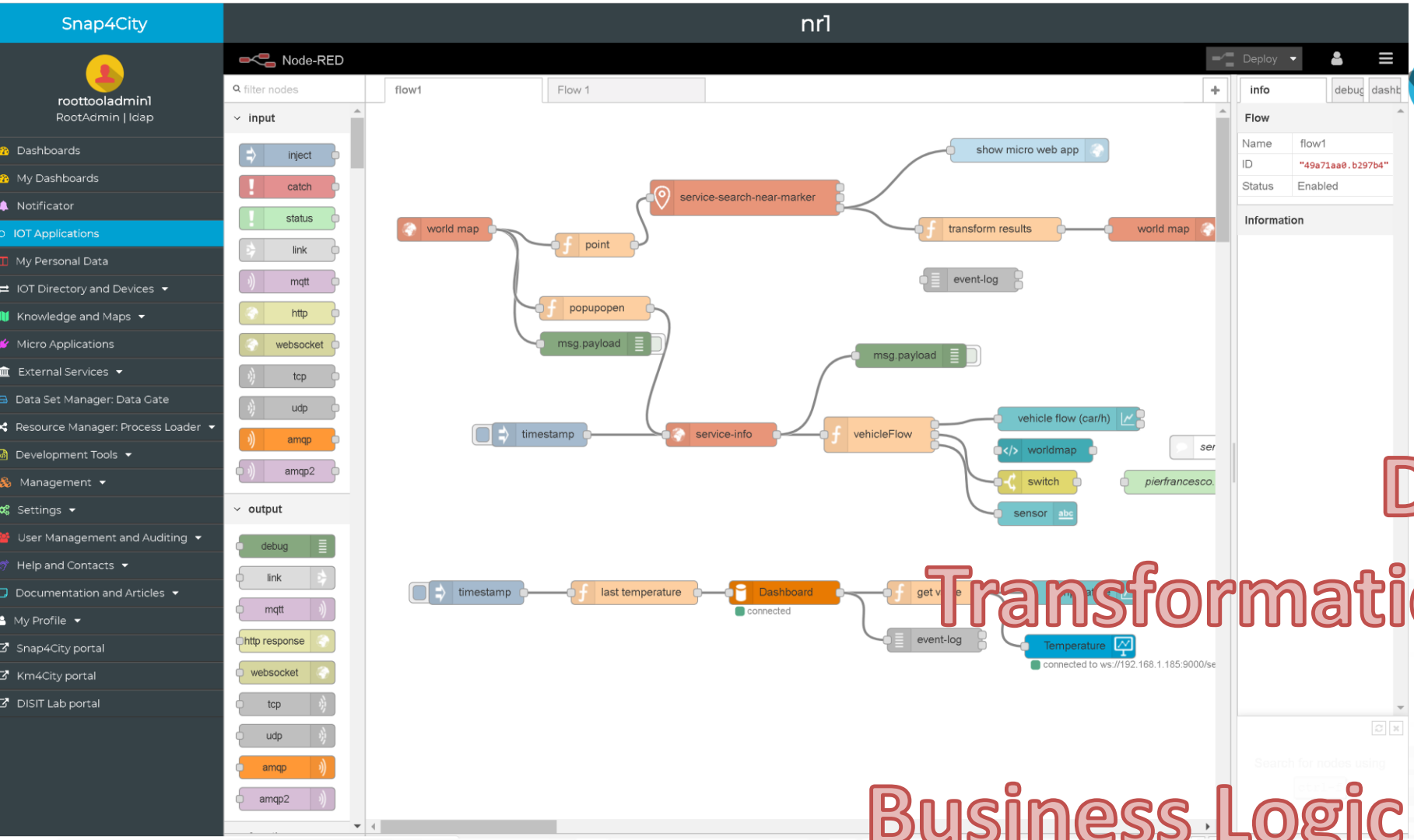


Sharing/saving reusing IOT App

Resource Manager

ServiceMap Discovery

Dashboard Collection, Editor and Wizard



Data Adapation

Transformation, Conversion

Integration

Business Logic vs Dashboards

Data Analytics control

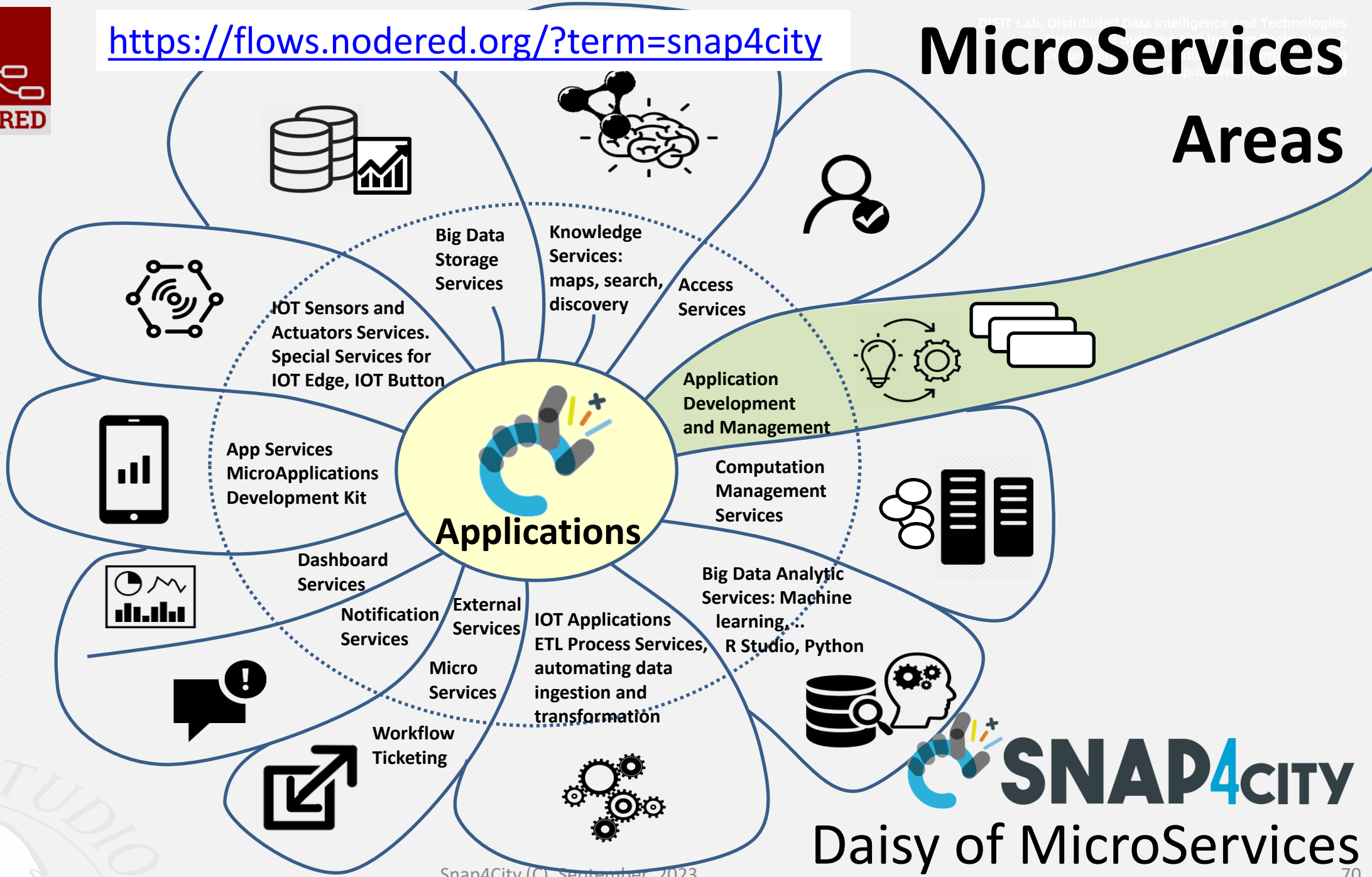
Editing IOT Applications

Everywhere: Cloud, on IoT Edge Devices

- input
- output
- function
- social
- storage
- analysis
- advanced
- NGSI
- Iwm2m
- S4CSearchDev
- S4CUtility
- S4CMapping
- S4CManagement
- S4CDataAnalytic
- S4CBigData
- S4CIOTApp
- S4CSearch
- S4CData
- S4CKPIData
- S4CDashboard
- S4CSigfox
- S4CIoT
- S4CLogDev
- S4CView
- S4CSocial
- location
- dashboard

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

MicroServices Areas



SNAP4CITY

Daisy of MicroServices

The Processing Logic (IoT App) microservices

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

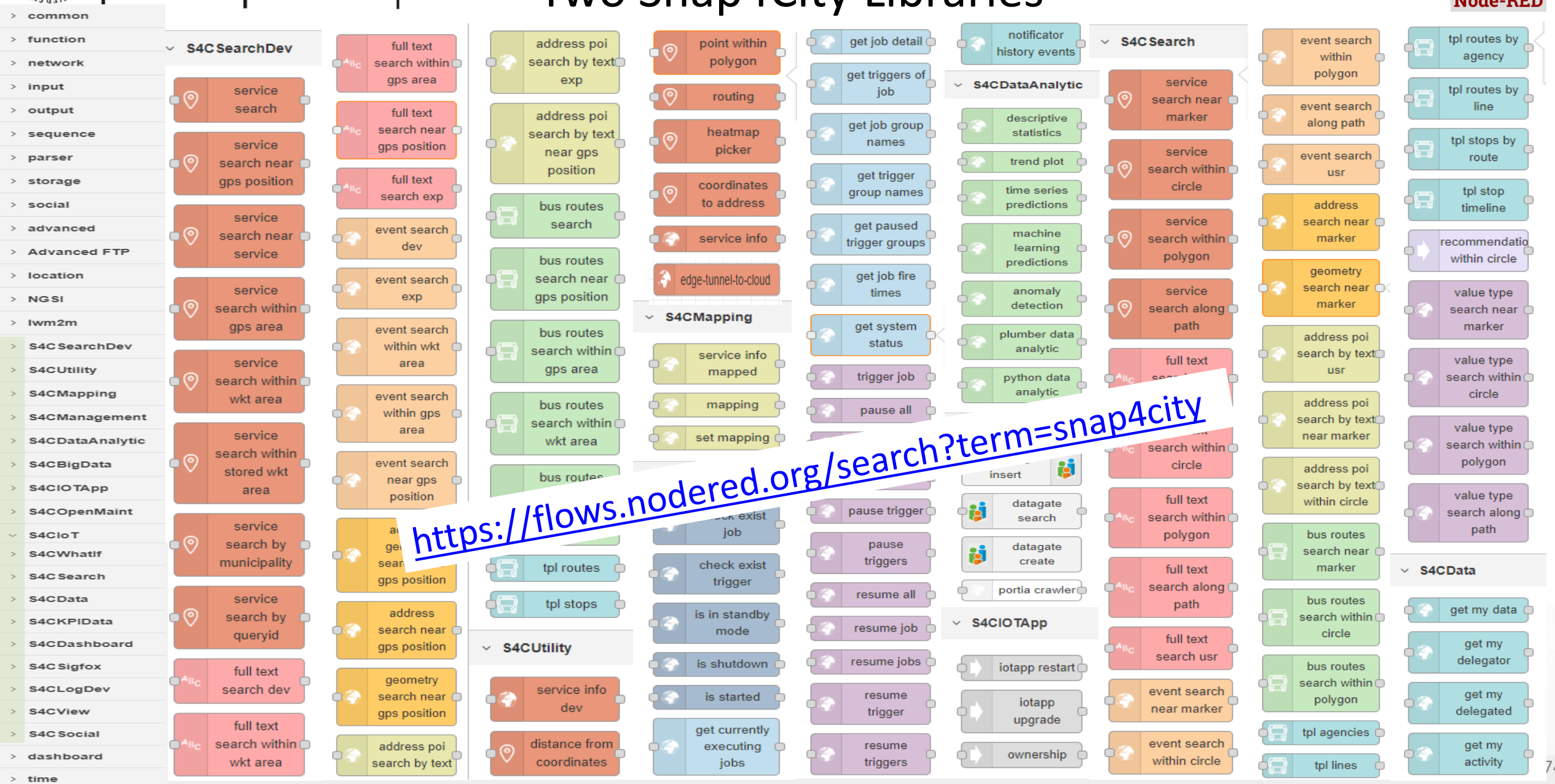
- **Data ingestion**: more than 100 protocols IOT and Industry 4.0, web Scraping, external services, any protocol database, etc.
- **Data access**: save/retrieve data, query search on expert system, georeverse solution, search on expert system Km4City ontology, call to Smart City API, etc.
- **Data Transformation/transcoding:**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

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

Sept 2023 collection

Two Snap4City Libraries

SNAP4CITY | Node-RED



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

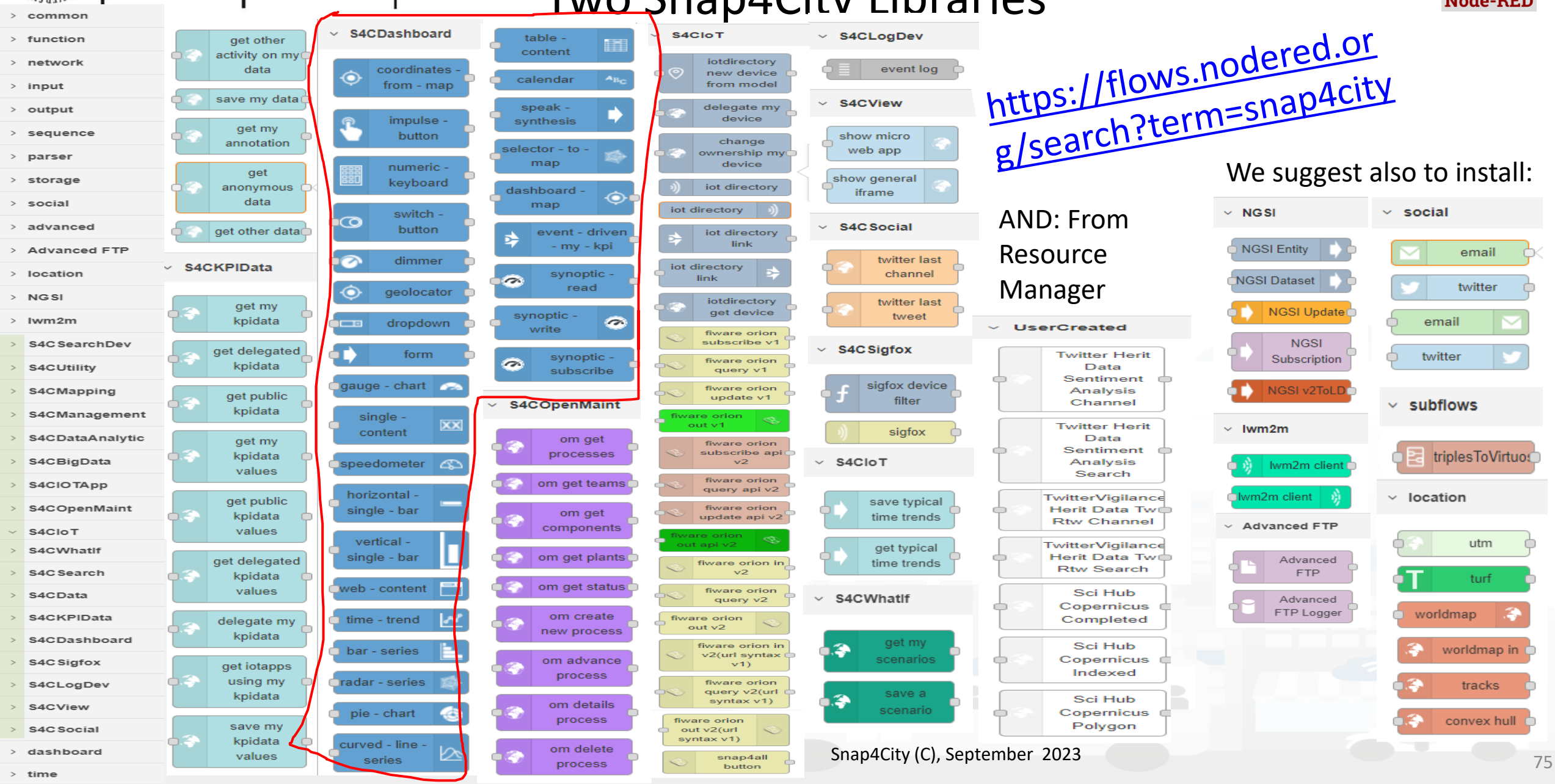
Sept 2023 collection

Two Snap4City Libraries

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

We suggest also to install:

AND: From Resource Manager



Standards and Interoperability (6/2023)

Compliant with:

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

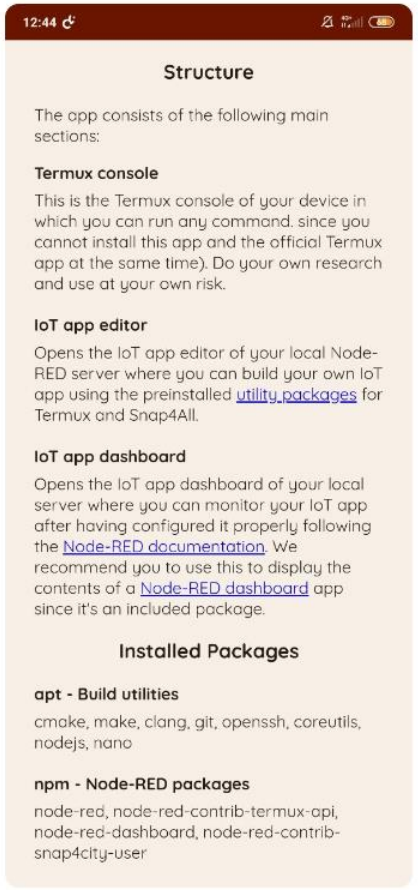
https://www.snap4city.org/65

Snap4All Mobile App

Node-RED on Android

Snap4All mobile app for Android

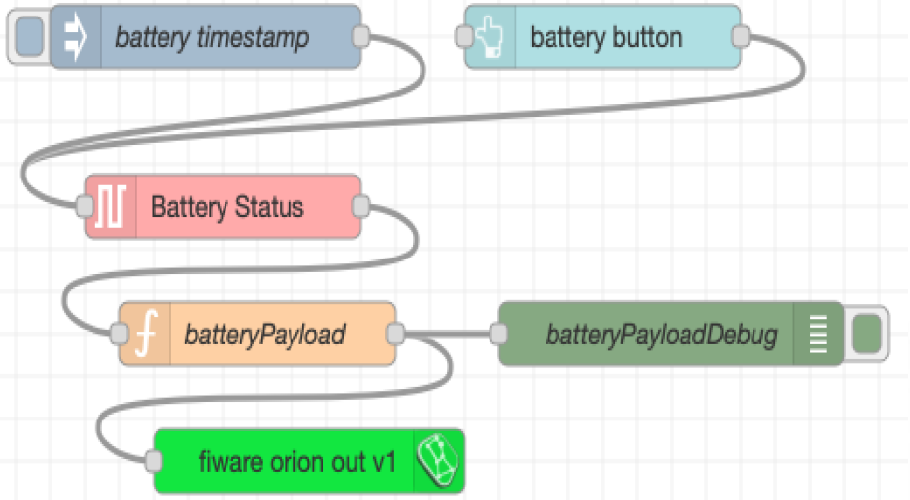
(a) Home

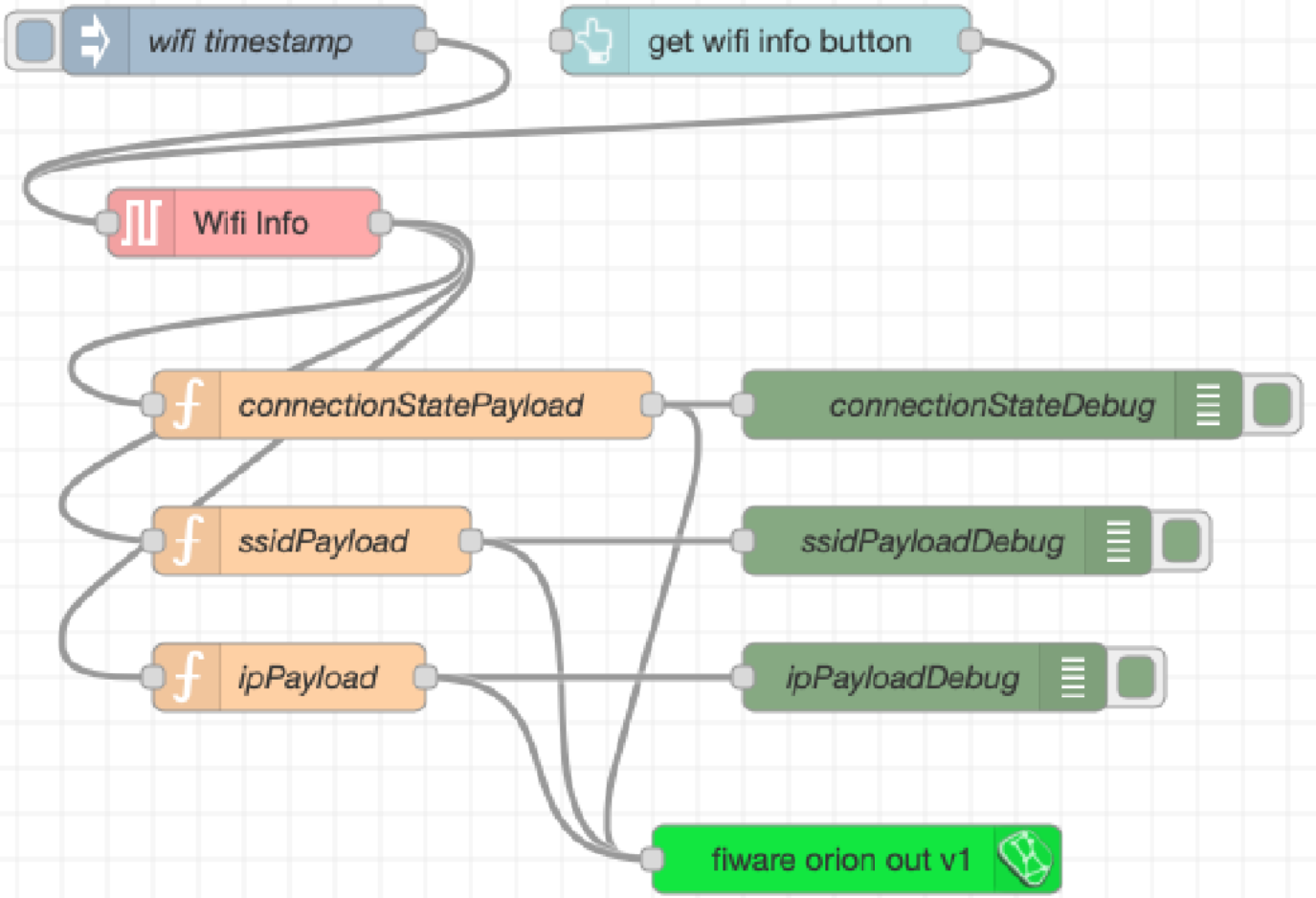


(b) Info

(c) Console

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



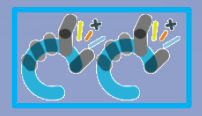


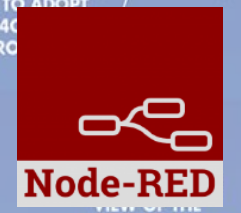
Technical Manual https://www.snap4city.org/drupal/sites/default/files/files/Snap4All-TechnicalManual-2022.pdf

COFFEE BREAK

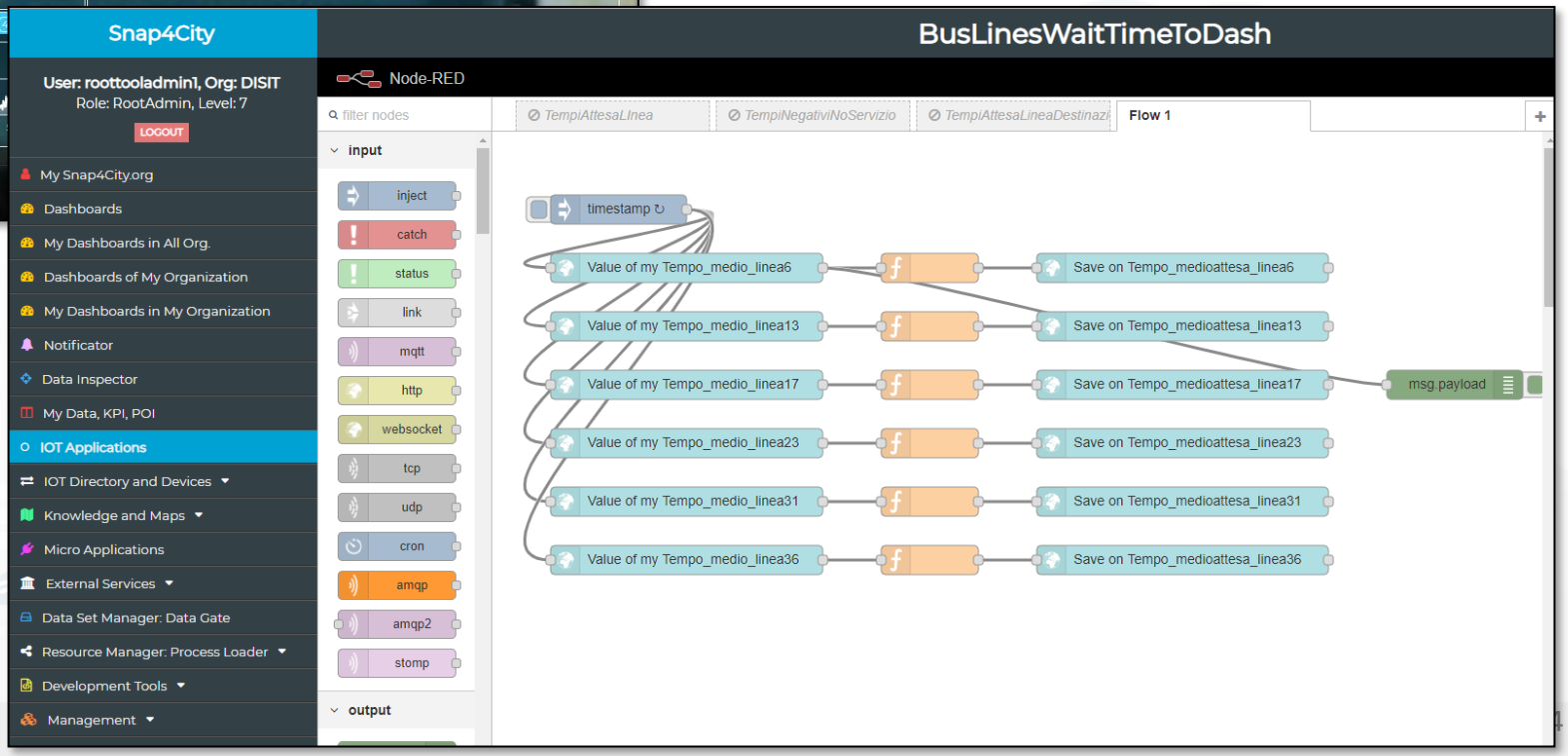
TOP

Examples of IoT App/Proc.Logic for Smartening Solutions





Estimation of the mean waiting time at bus stops



IoT App Smartening Devices and Dashboards

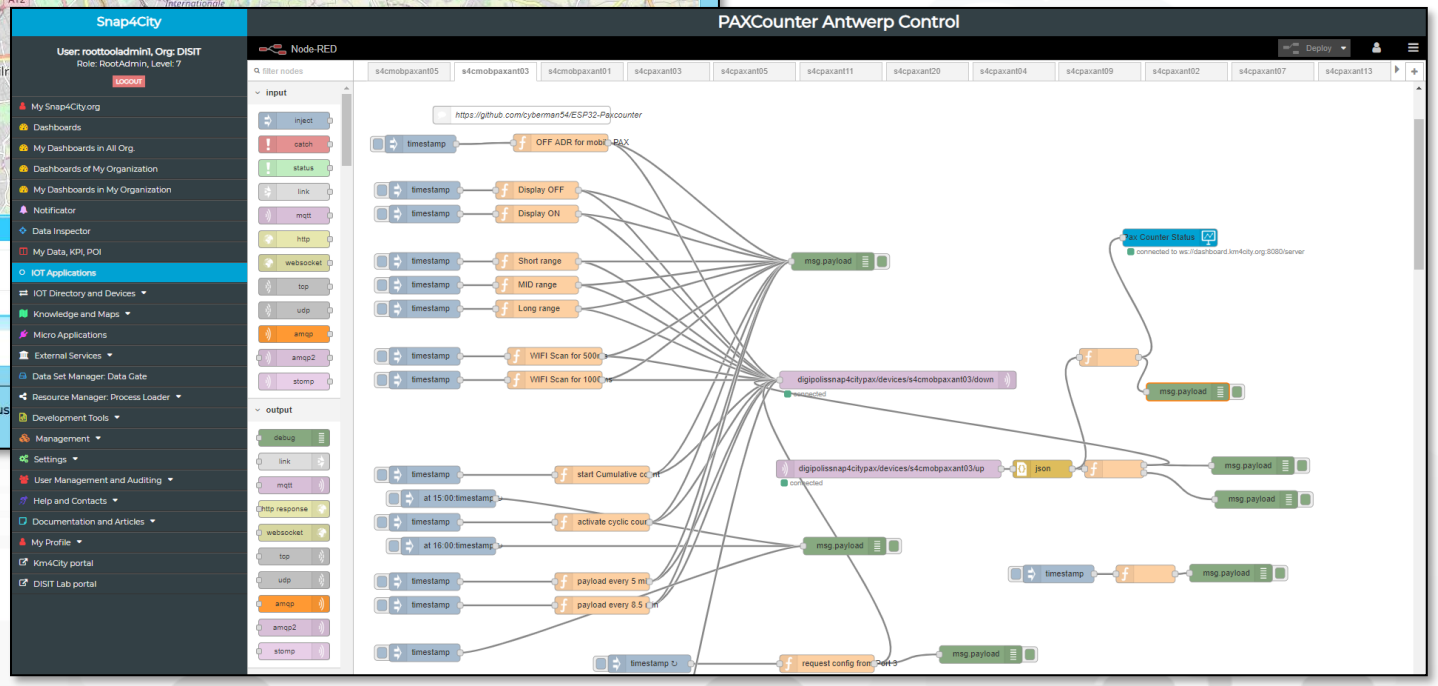
PaxCounter devices

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

LoRa® Wi Fi Bluetooth™

https://www.snap4city.org/456

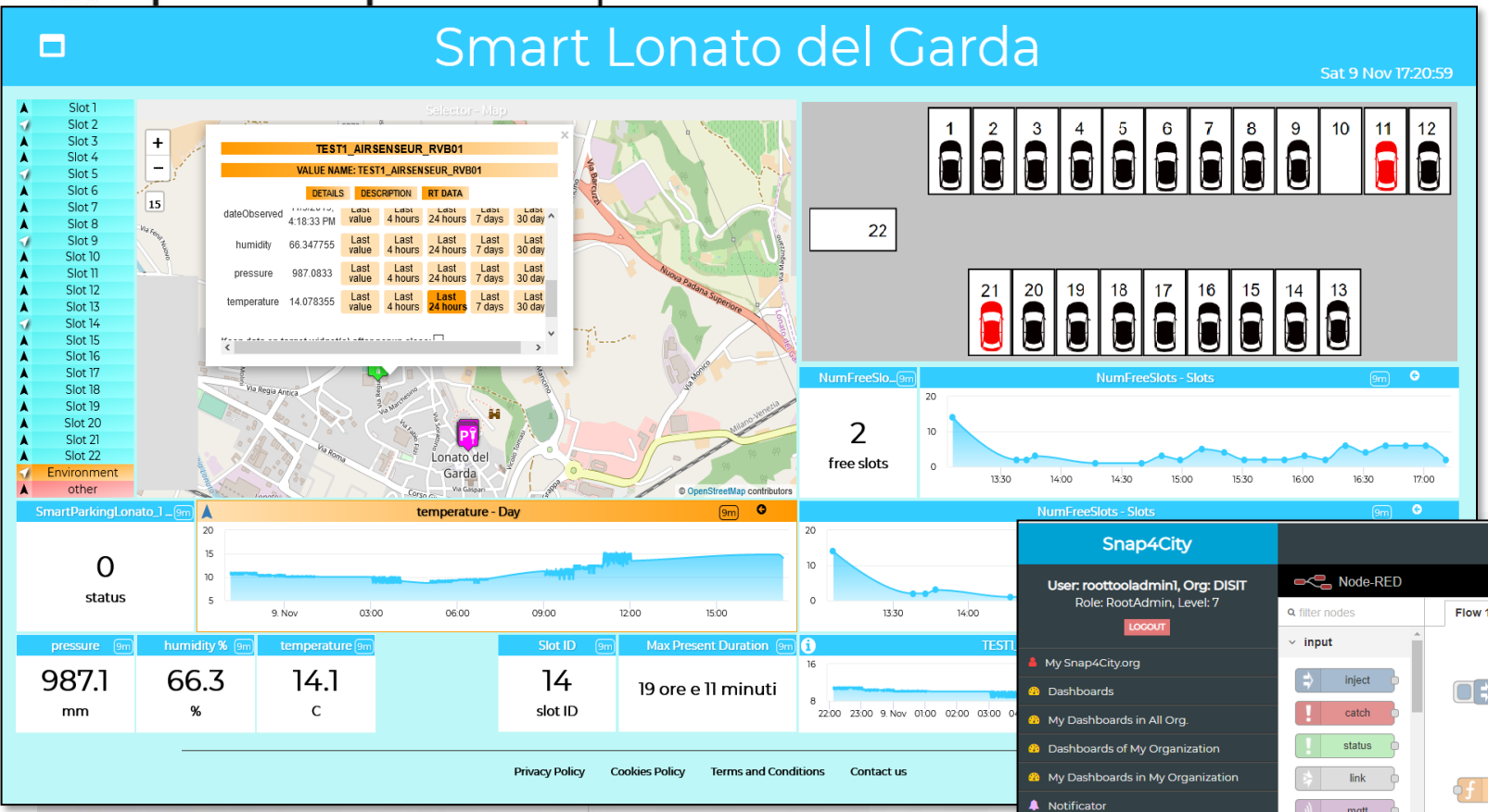
Programmable PAX counting

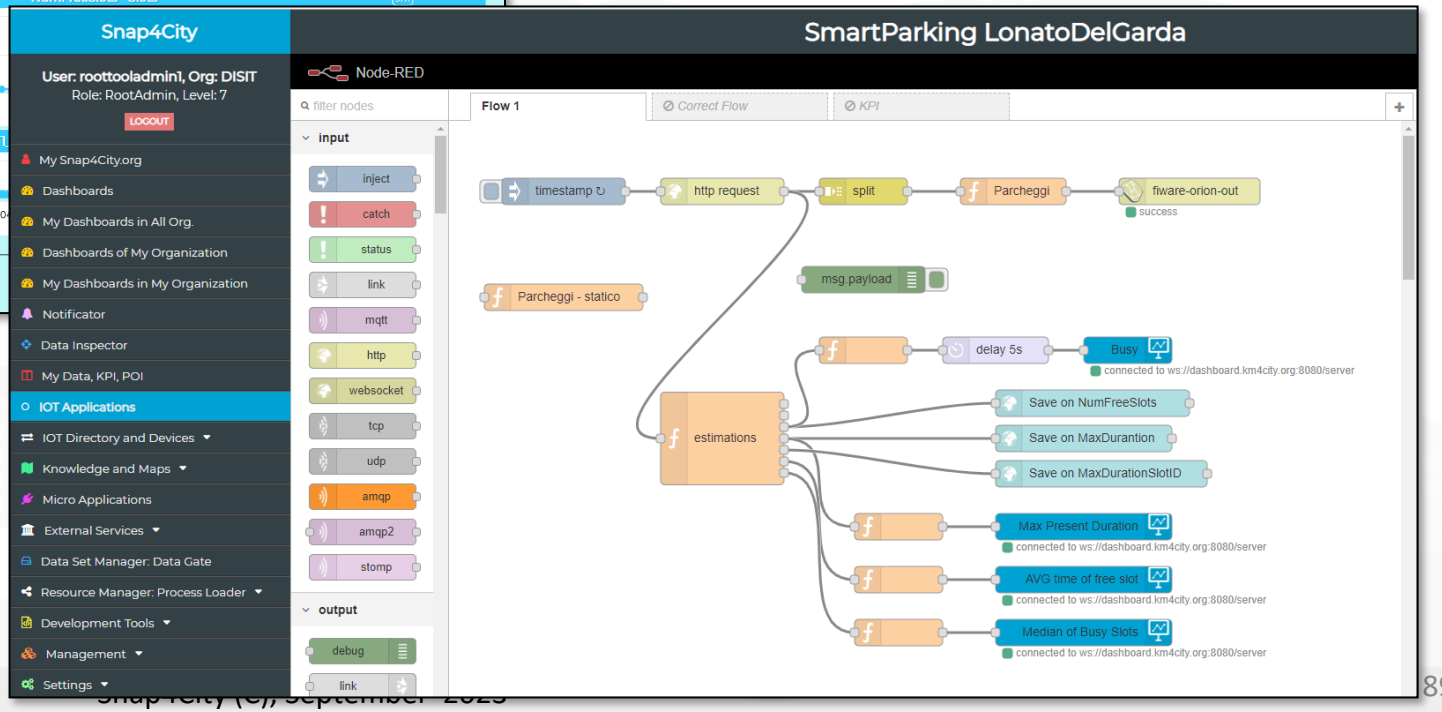


Antwerp

IoT App Smart Parking

Smart Parking Monitoring (SVG, statistic, overparking)





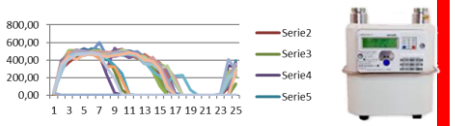
IoT App Smart Industry 4.0
Snap4Industry

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

https://www.snap4city.org/369

SNAP4CITY KM4CITY

External Services



Fleet management

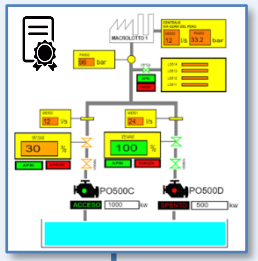
IoT Devices/Edge

IoT Broker

Internet

SECURE

Admin



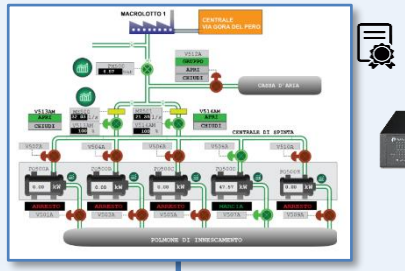
DCS

IoT Broker

PLC

IoT Devices/Edge

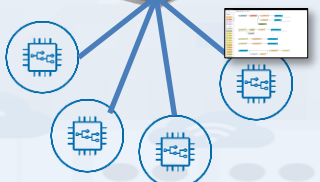
Industry Plant1.....



SCADA

PLC/ RTU

IoT Broker

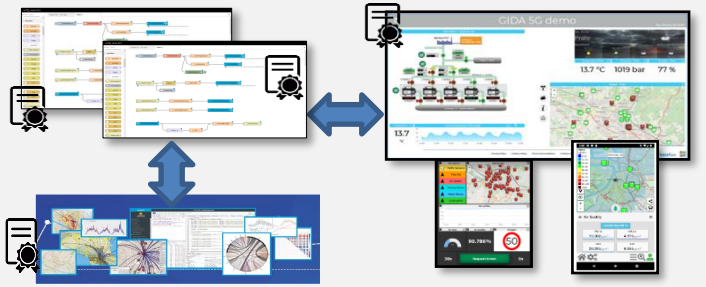


IoT Devices/Edge

Industry Plant2.....

IOT Applications

Dashboards and Apps



Big Data Analytics, Artificial Intelligence

Control and Supervision on Multiple Supply Chains

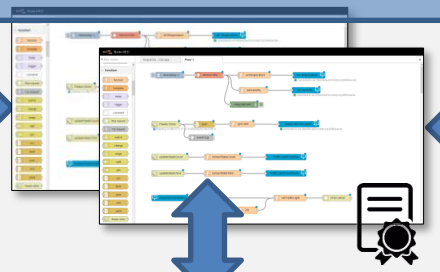
Industry 4.0 as a Service


Prato
https://www.snap4city.org/758
Smart City vs Industry 4.0

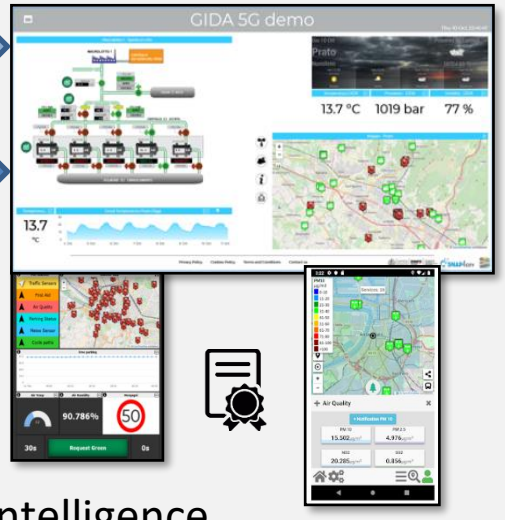
GIDA set up

Smart City data from many sources

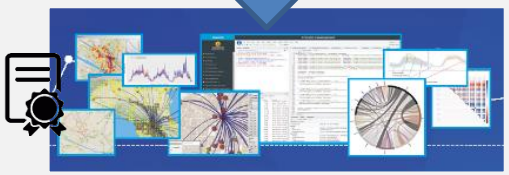
IOT Applications



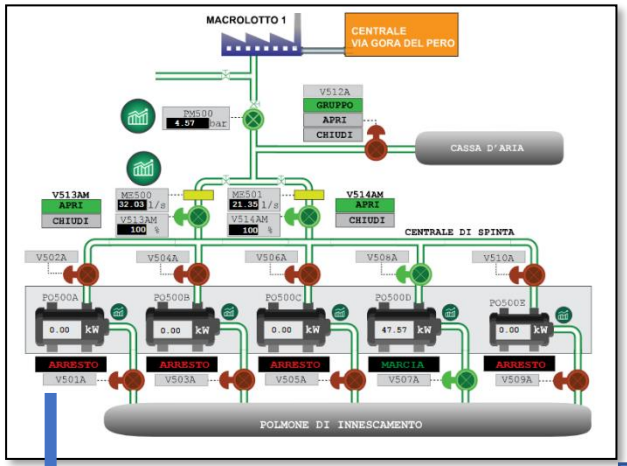
Dashboards and Apps



IOT Data Shadow Snap4City



Big Data Analytics, Artificial Intelligence



ModBus to Snap4City Gateway Edge

5G network devices

Telemonitoring Telecontrol

https://www.snap4city.org/369

https://www.snap4city.org/369

Measuring any kind of sensors values

Controlling Energy Power

Measuring
Energy Consumption

Any kind of notification channel

IOT Edge:
Node-RED
+
Snap4City

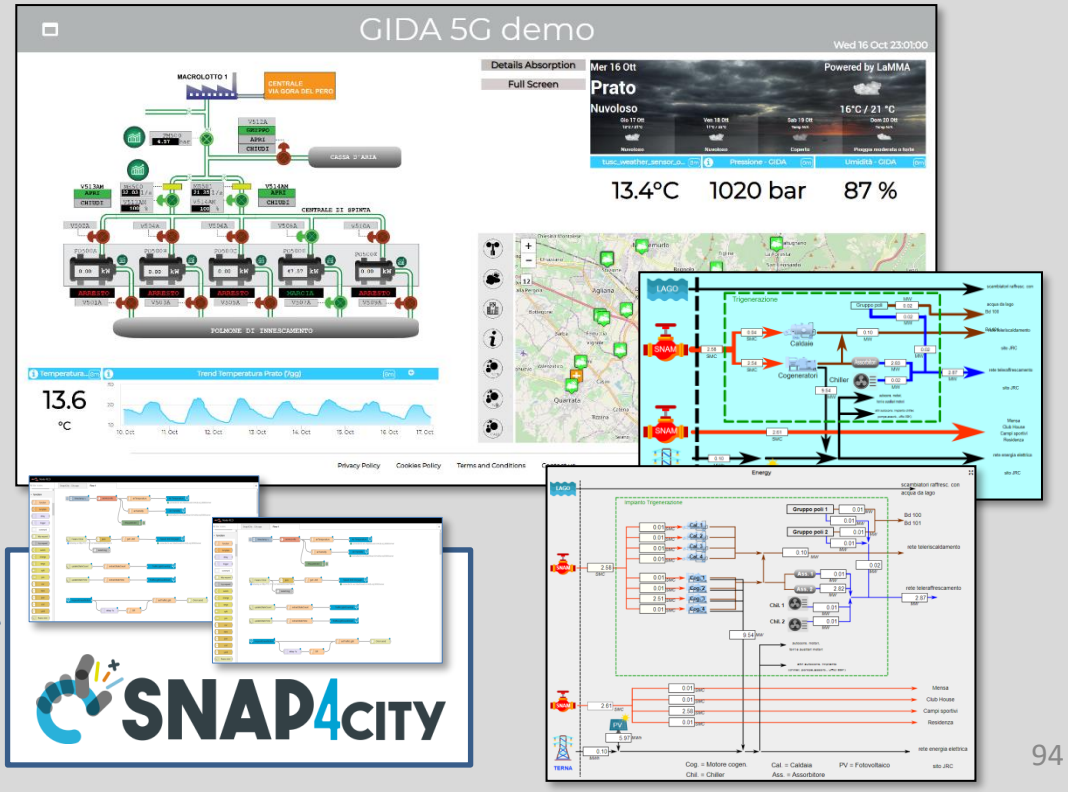
Local Control

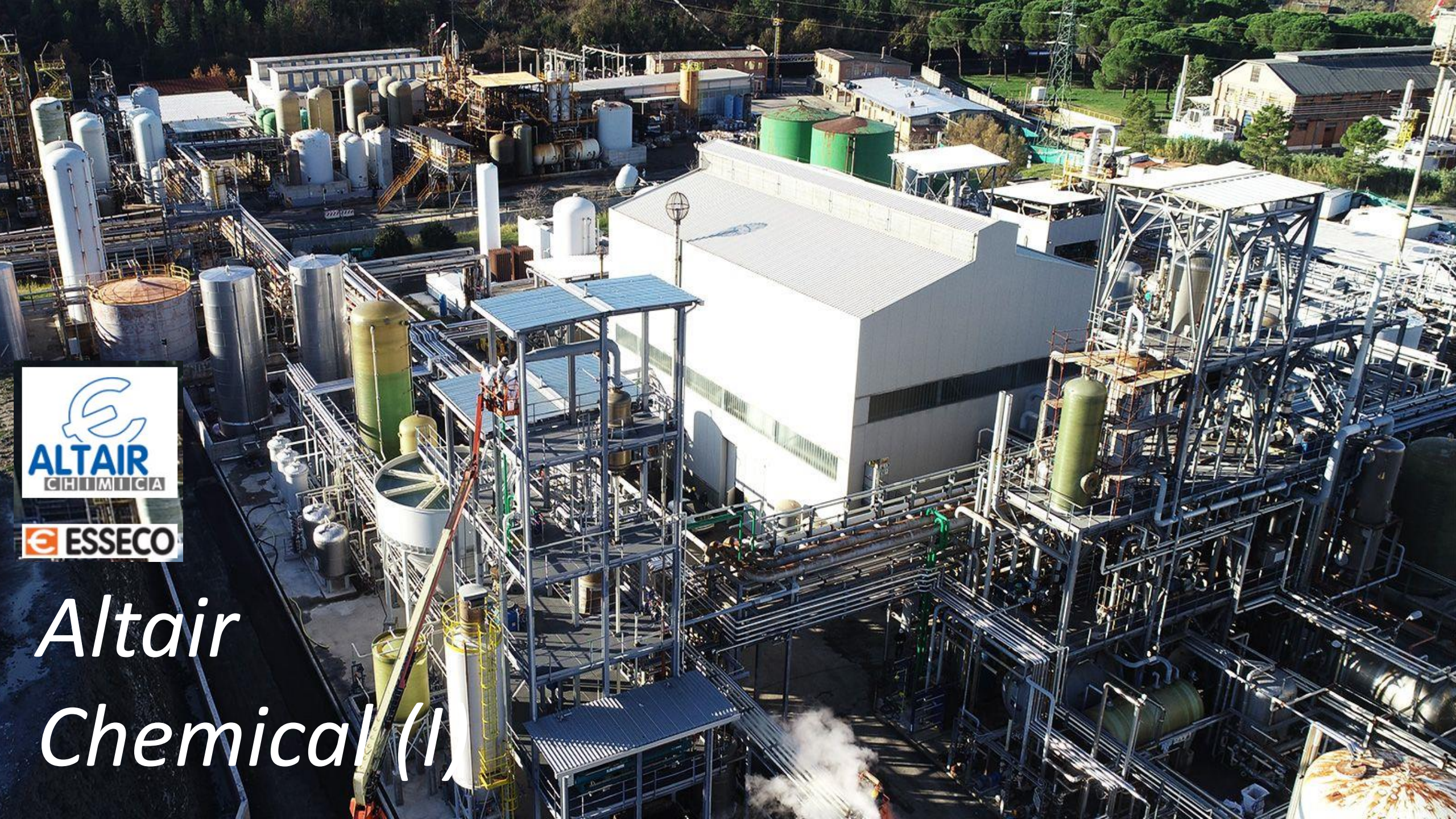
DCS

Administrative Servers

Alexa: Voice Commands

Contextual (smart city/home) data, Data Analytics
Historical Data, Remote Control, Mobile App



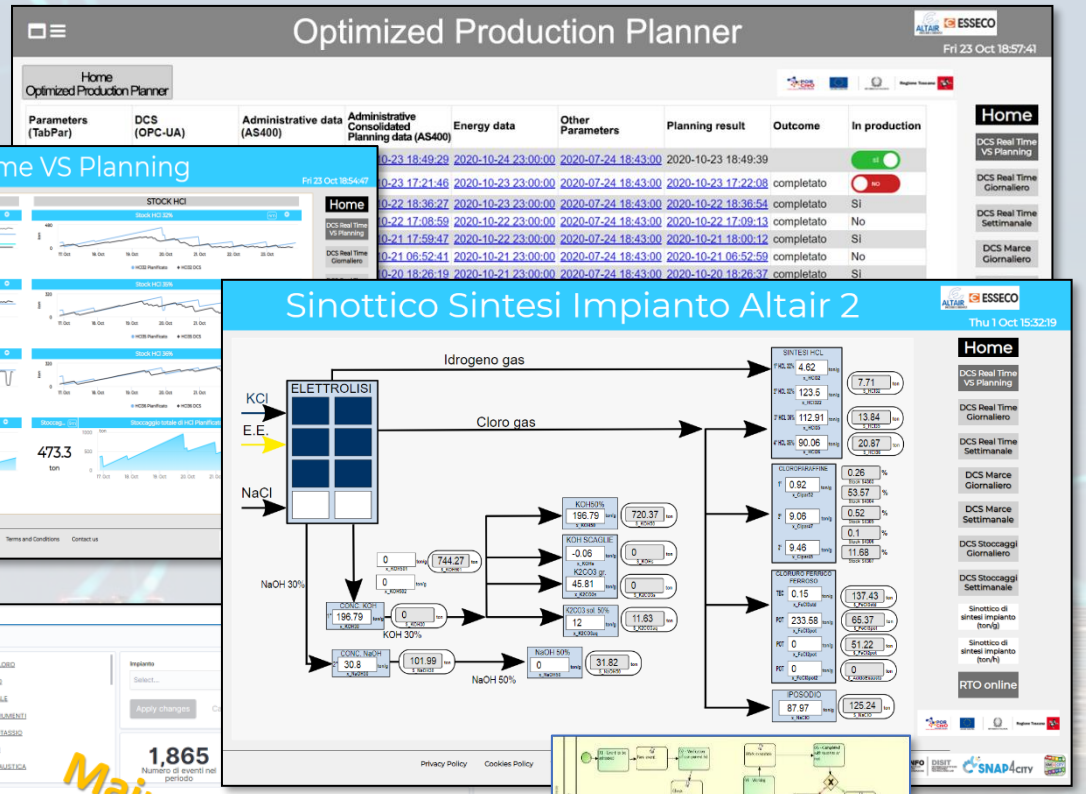


Altair Chemical (I)

Snap4Altair Decision Support supervision and control, Industry 4.0

reference

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



Industry Plant Supervision and Maintenance

Aims

- **Control Room**: Higher level supervision and monitoring (since 2020)
  - Management of Production Plan *Optimization*
  - Control of Perimeter with drone and sensors
- **Maintenance ticketing** (since 2017)
  - *predictive* (in development)
  - 3D Digital Twin (in development)

MicroService Architecture

Energy Service

Data Connections and Transformation

IoT App/DA: Real Time & Stream Processing

Predictive Maintenance

Prod. Plan Optimization

API/MicroServices

Maintenance Intelligence

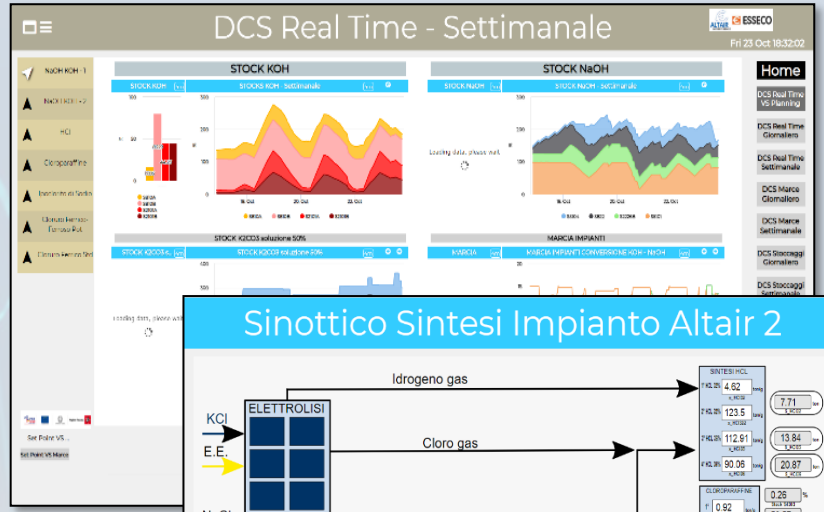
Digital Twin Local / BIM

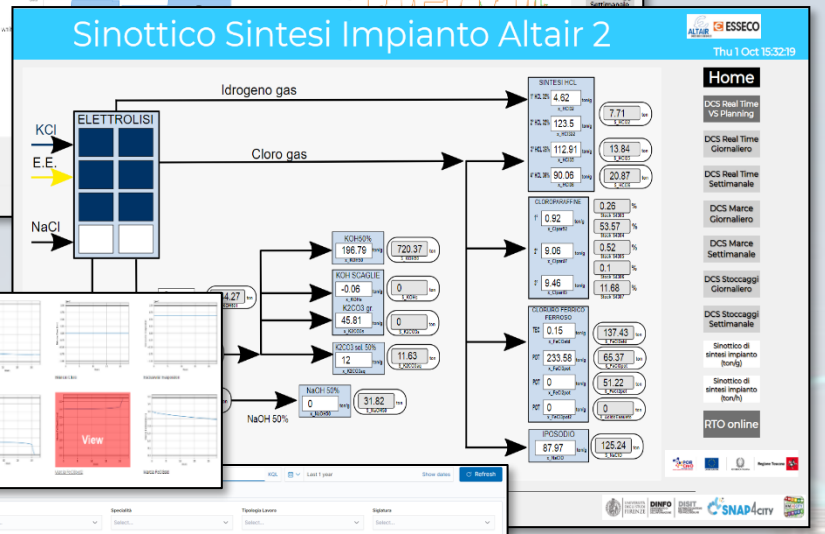
.....

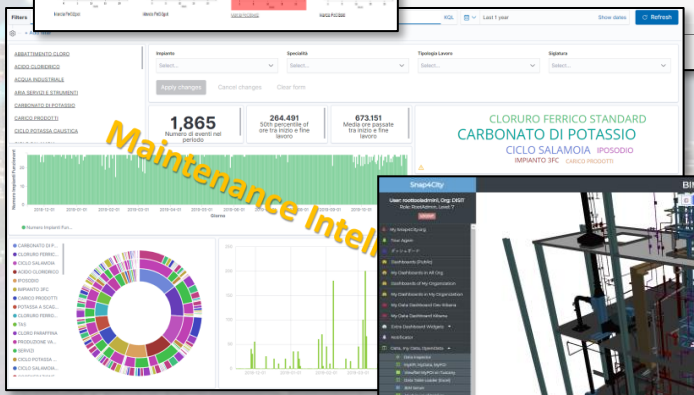
Data Storage

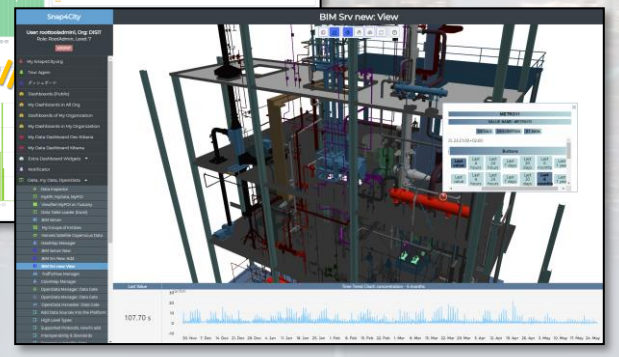
Management, Auth./Autoriz.

Snap4City Dashboard Builder

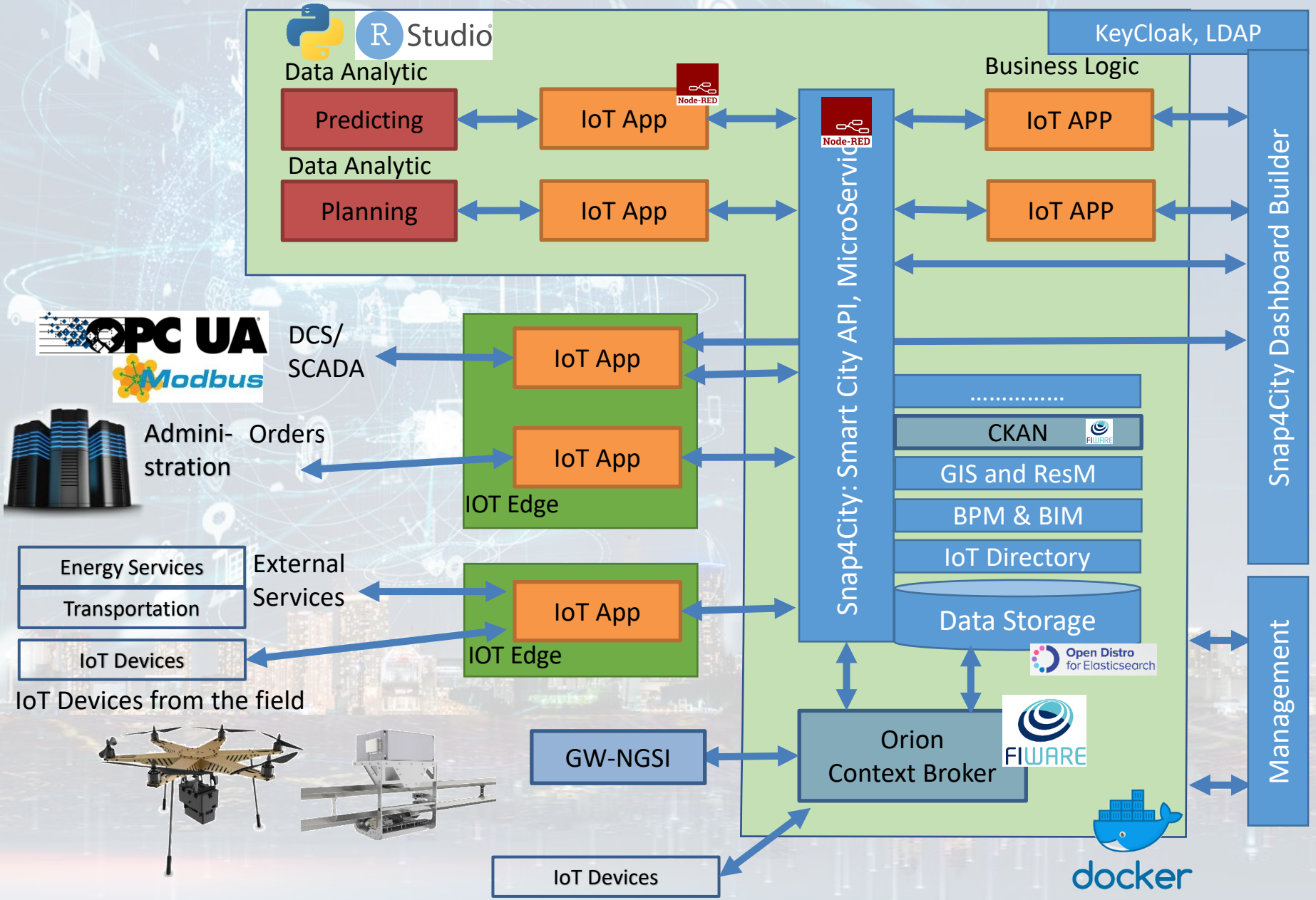






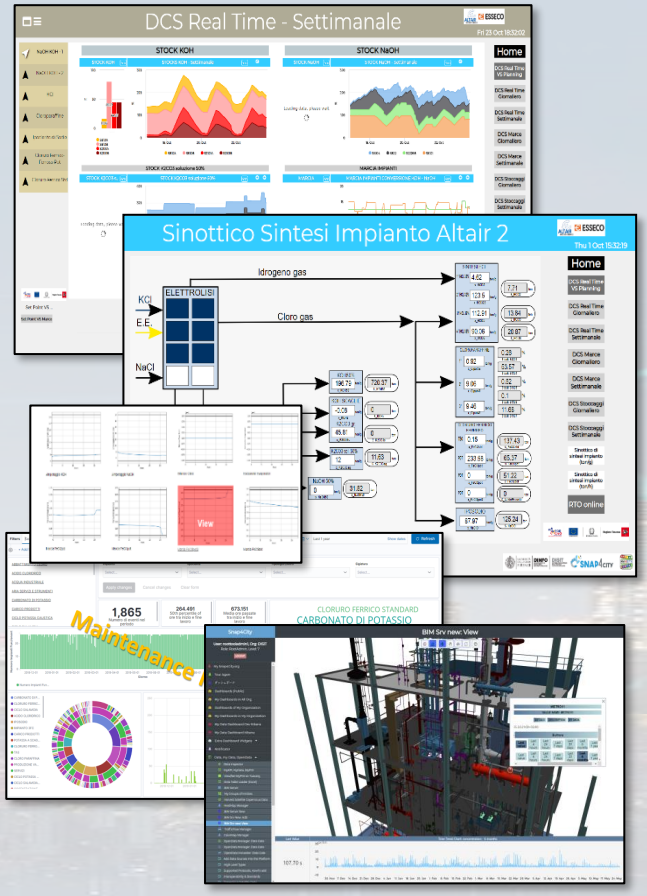


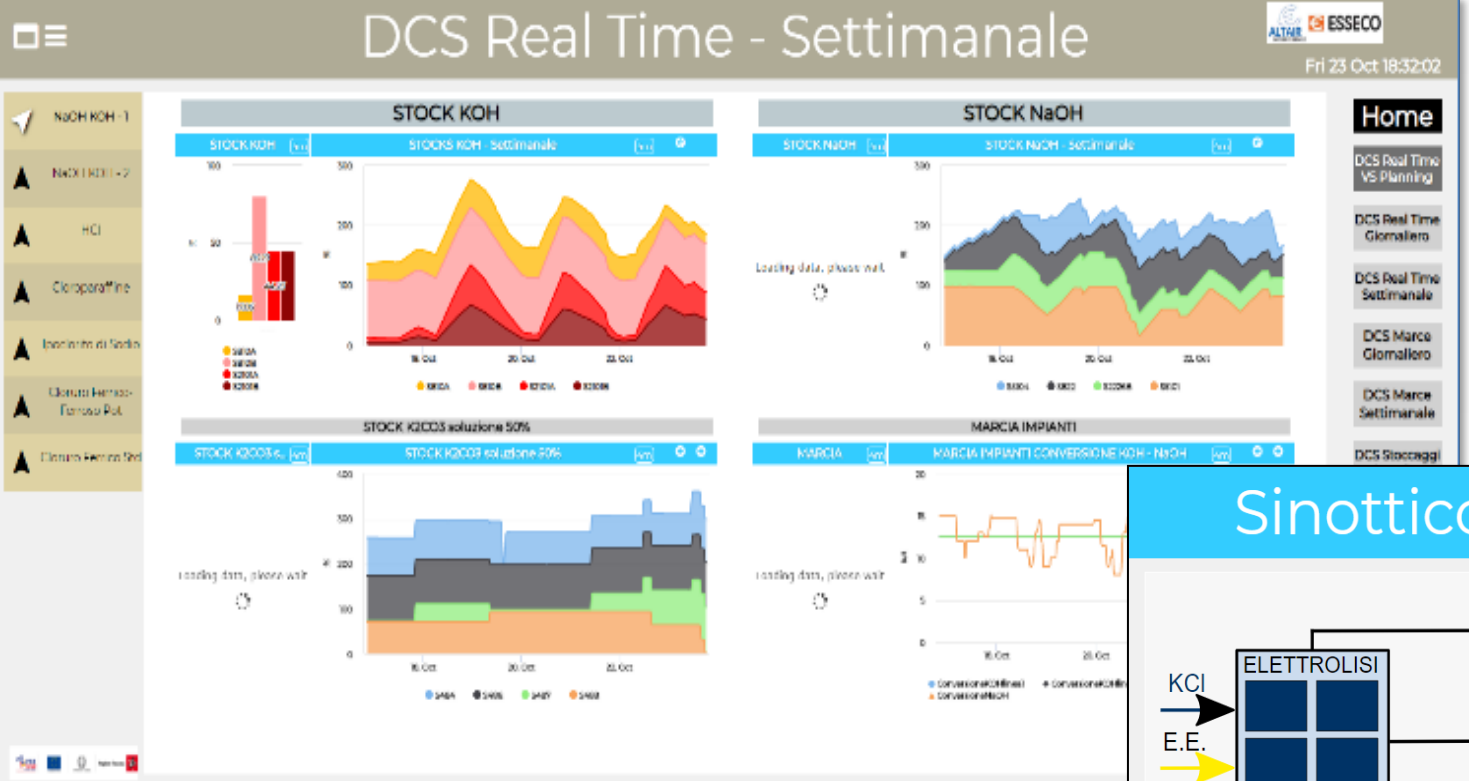
Snap4City/Industry Detailed Architecture

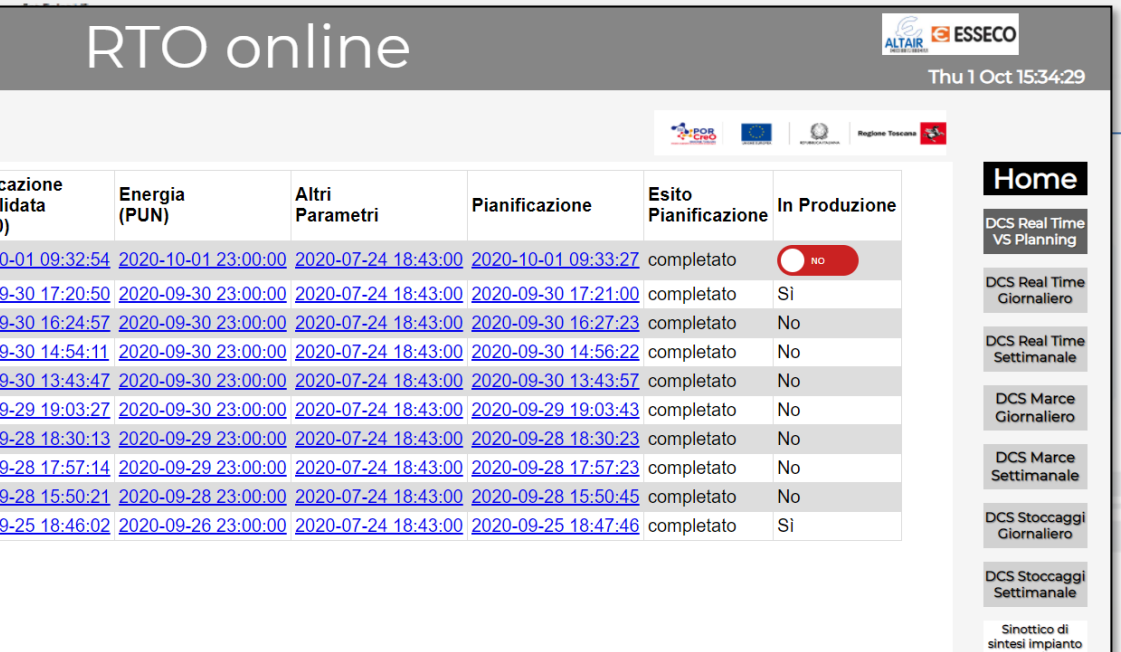


Production Parameters

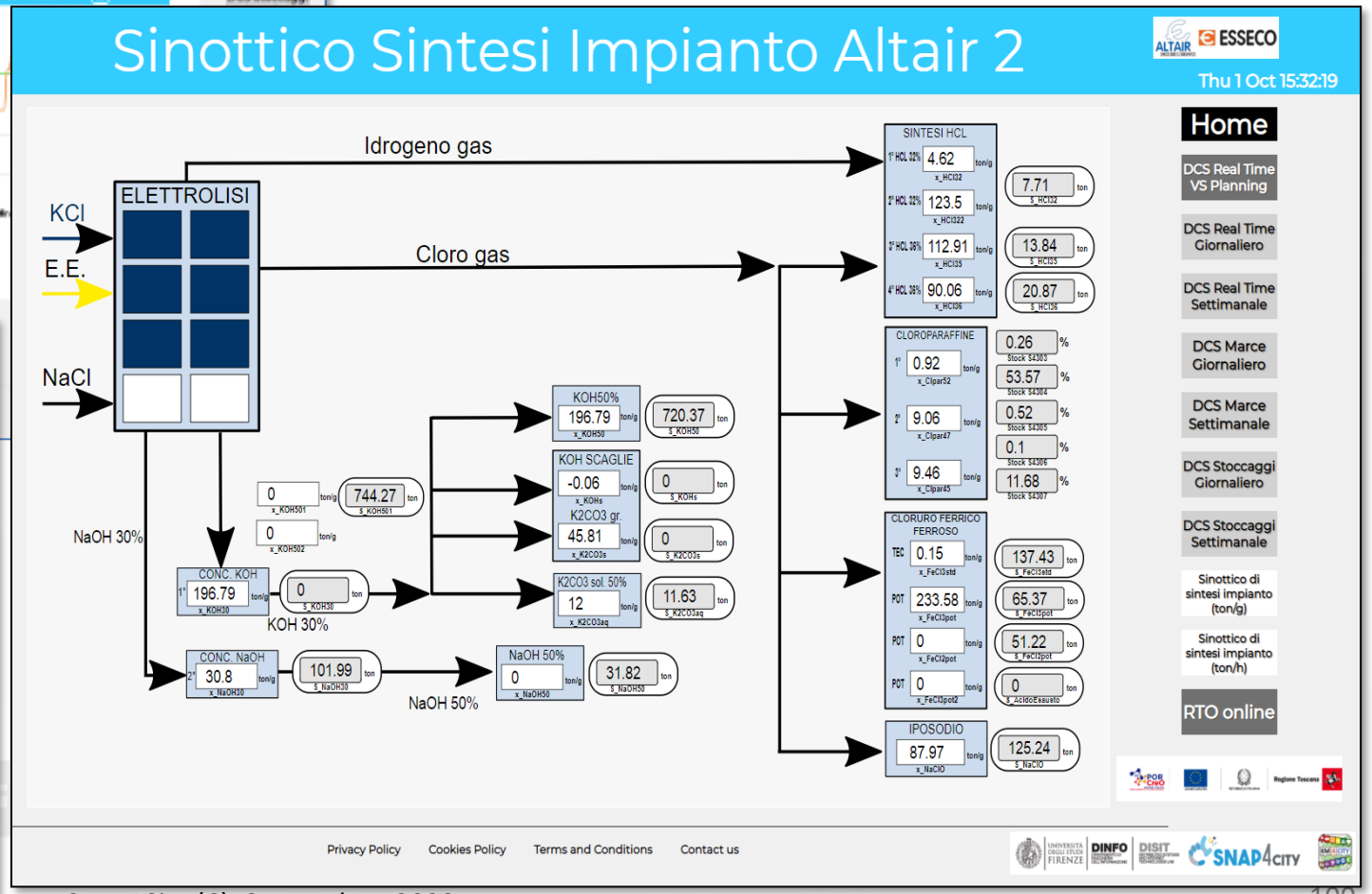
Dashboards, Visual Analytics, Synoptics, 3D, Maps



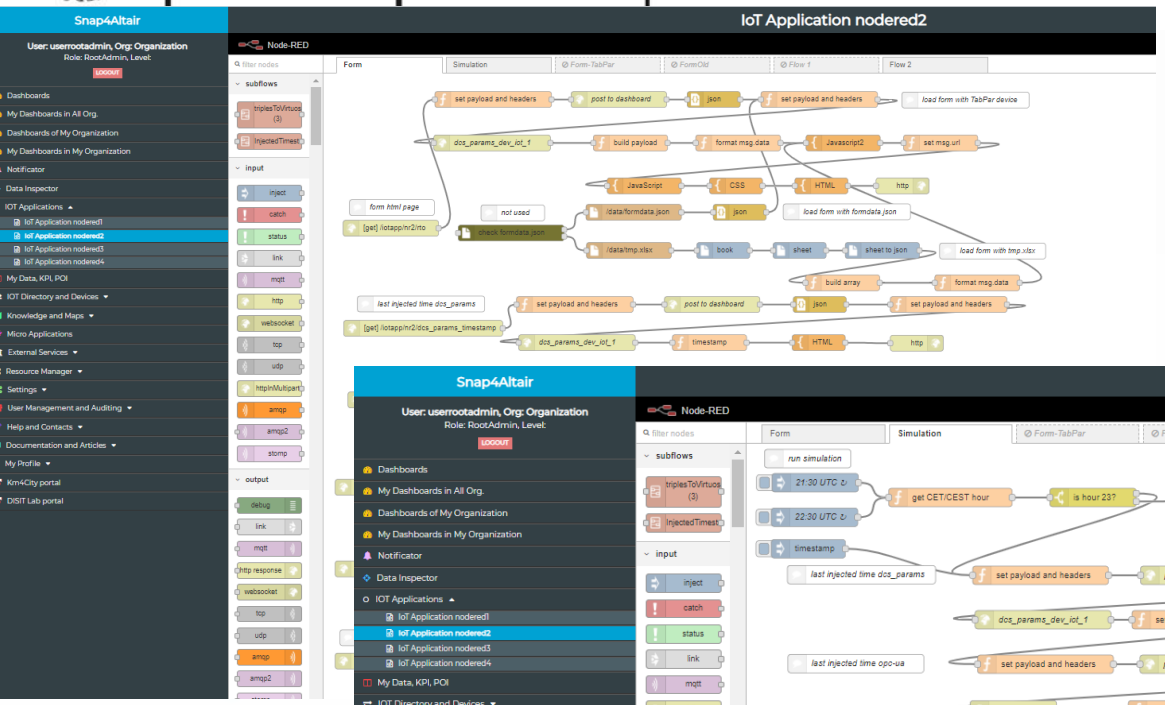


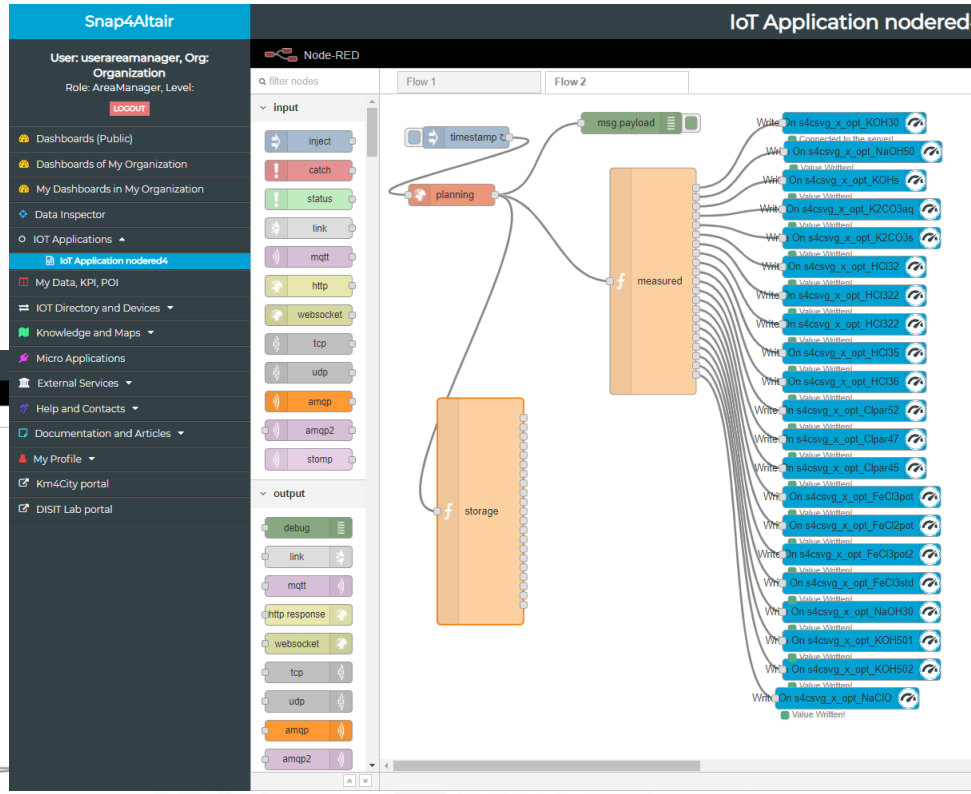


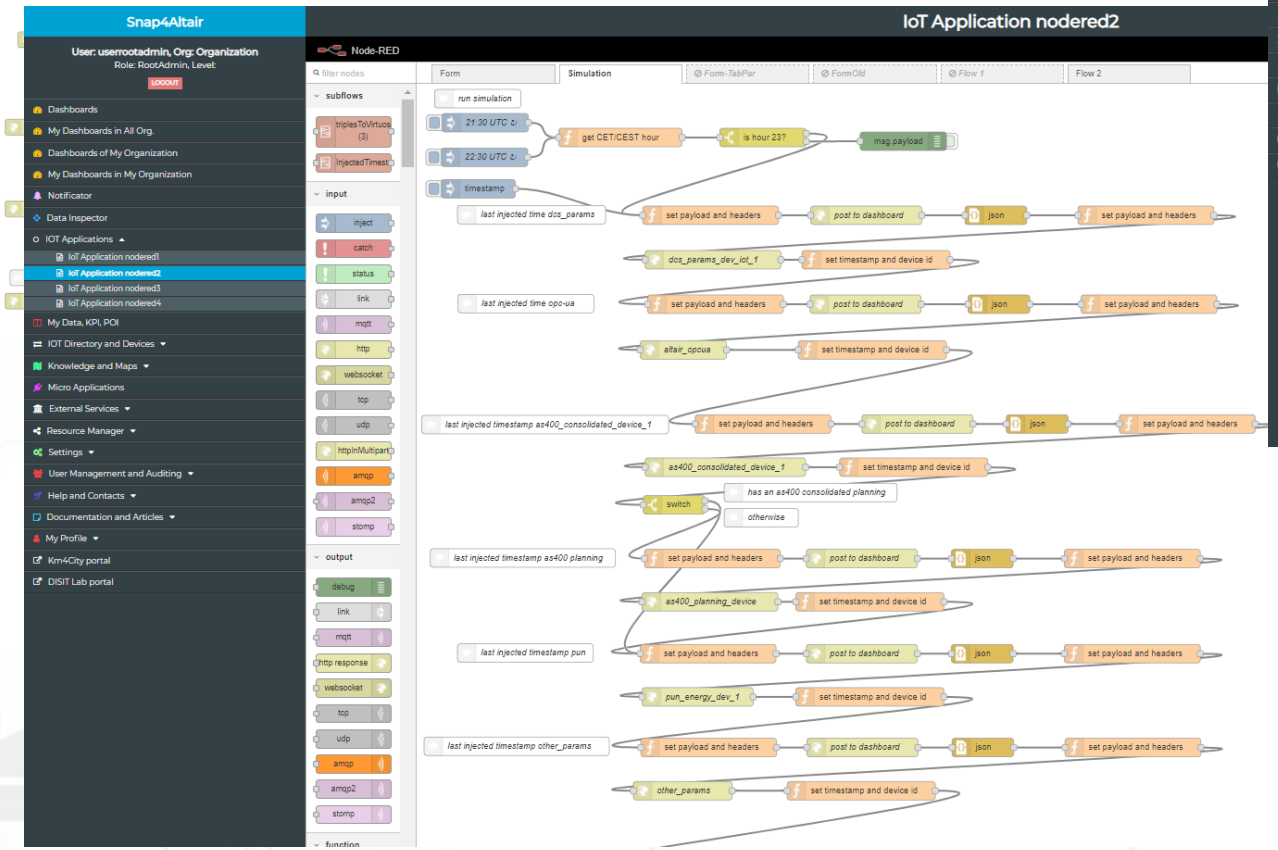
cazione lidata)	Energia (PUN)	Altri Parametri	Pianificazione	Esito Pianificazione	In Produzione
0-01 09:32:54	2020-10-01 23:00:00	2020-07-24 18:43:00	2020-10-01 09:33:27	completato	NO
9-30 17:20:50	2020-09-30 23:00:00	2020-07-24 18:43:00	2020-09-30 17:21:00	completato	Sì
9-30 16:24:57	2020-09-30 23:00:00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



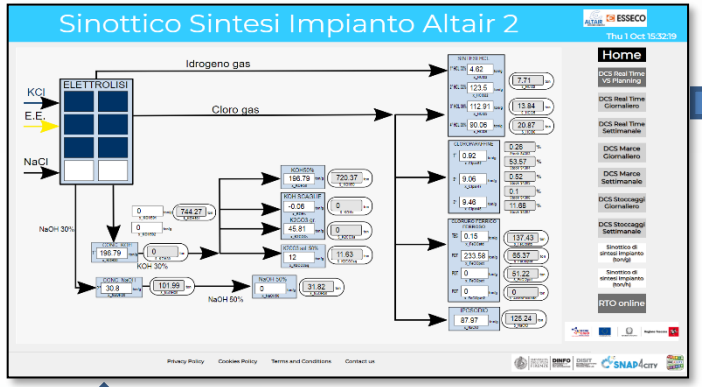
Some Flows


Snap4Altair
IoT Application nodered2
Node-RED


Snap4Altair
IoT Application nodered
Node-RED

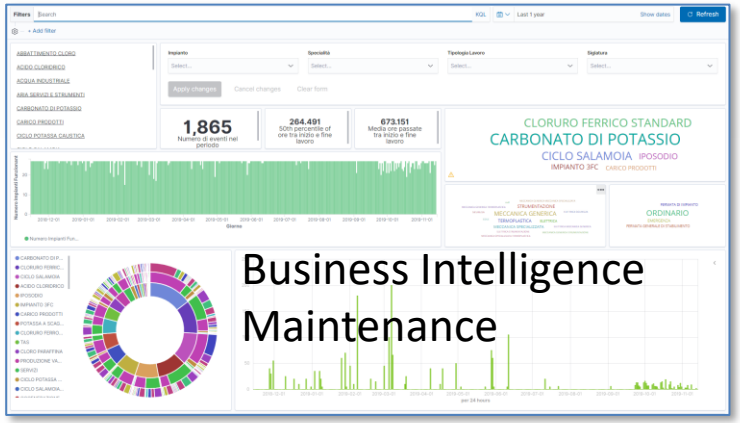

Snap4Altair
IoT Application nodered2
Node-RED

Workflow for Ticket management

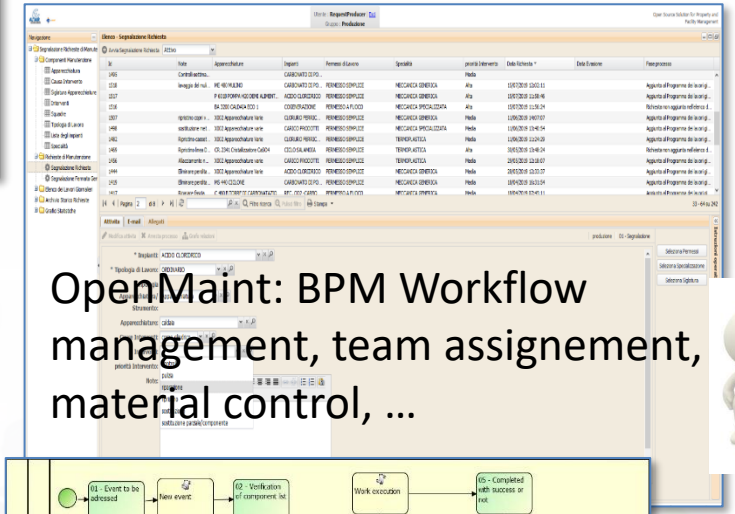


Dashboards and actions

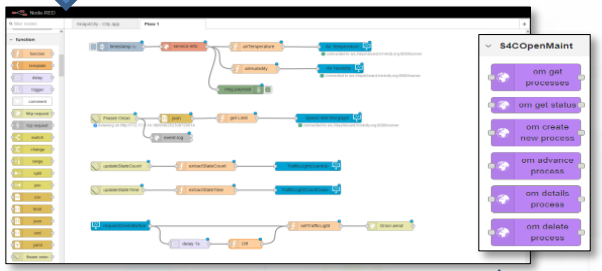
Consumptions/productions



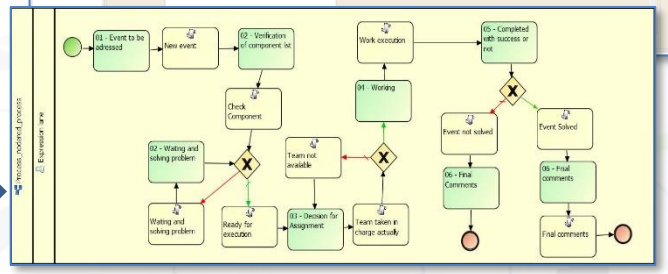
Events/actions

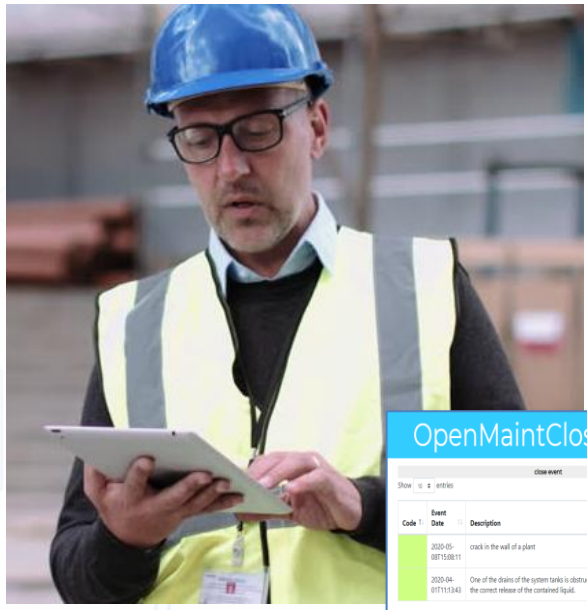


OpenMaint: BPM Workflow management, team assignement, material control, …



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

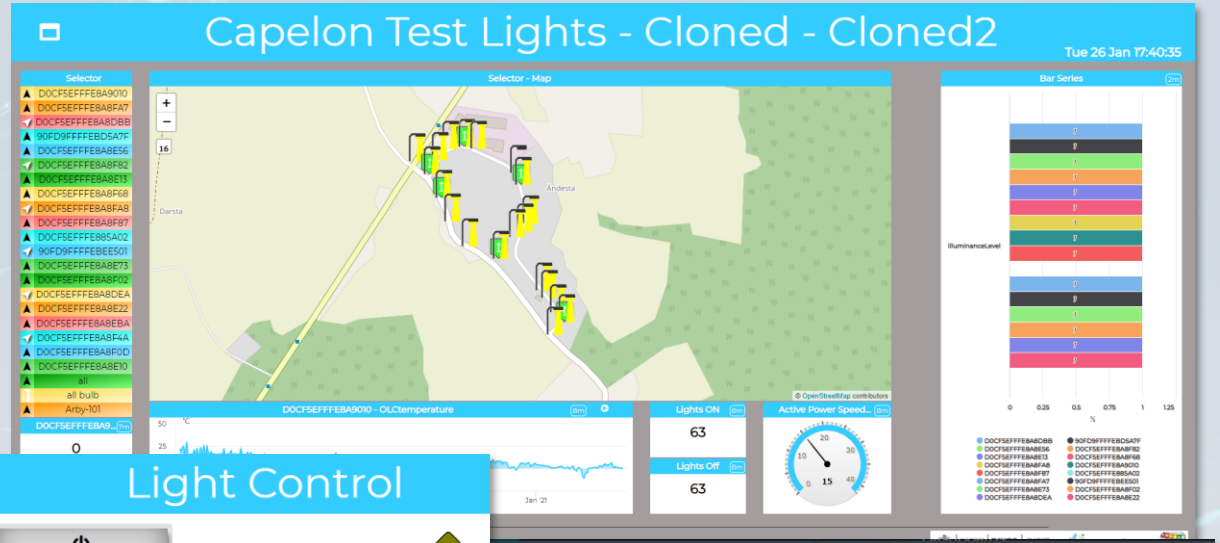


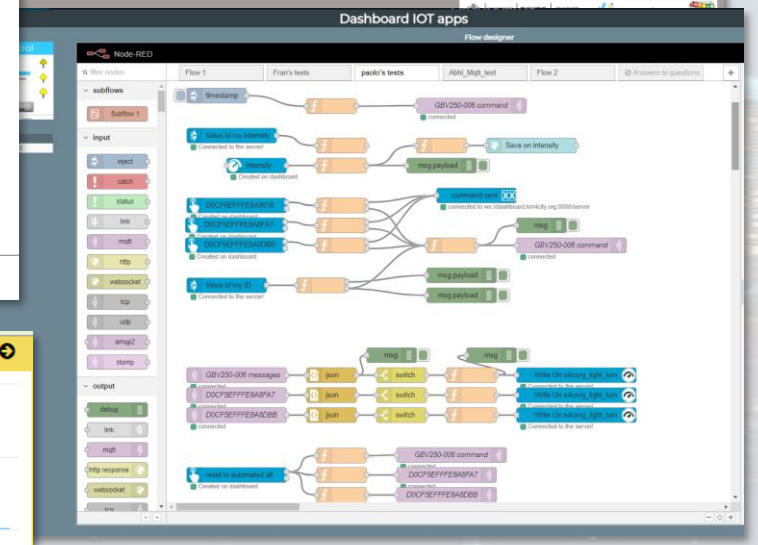


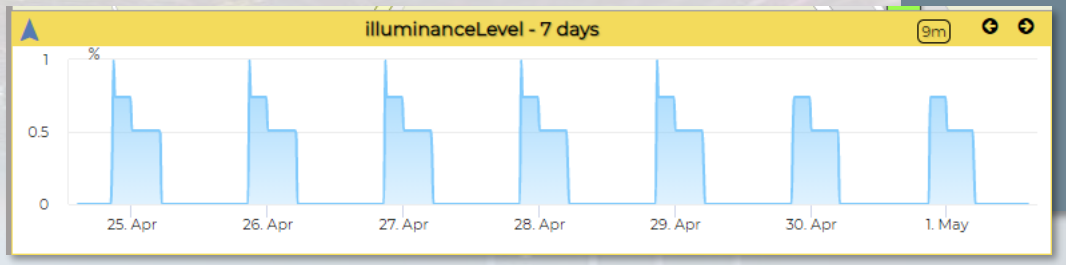
Västerås, Sweden

Smart Light Control of CAPELON *reference*

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

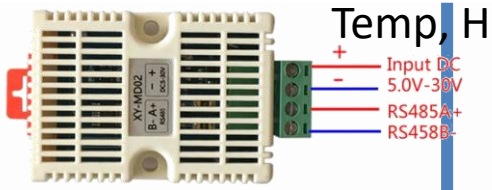






IoT App Smart Industry 4.0
ModBus Integration

Devices

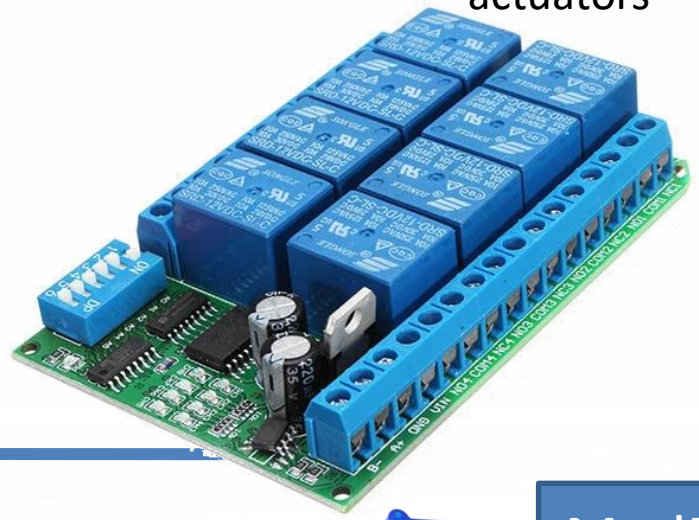


Temp, Humidity

Energy Metering

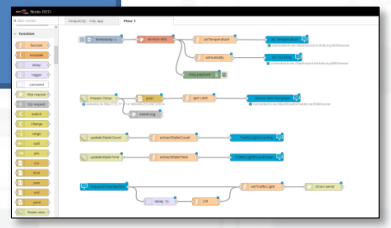
Modbus

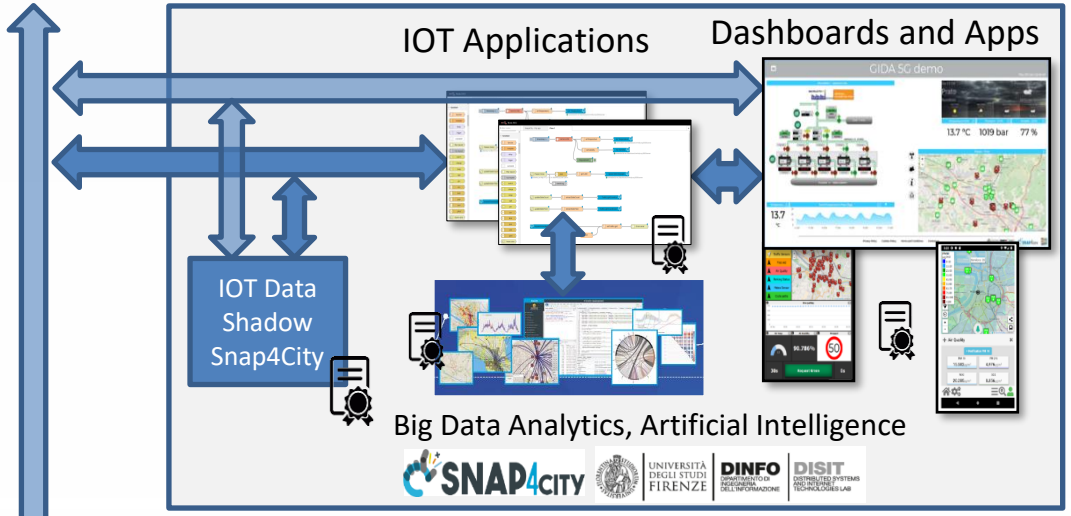
8 Relais, actuators



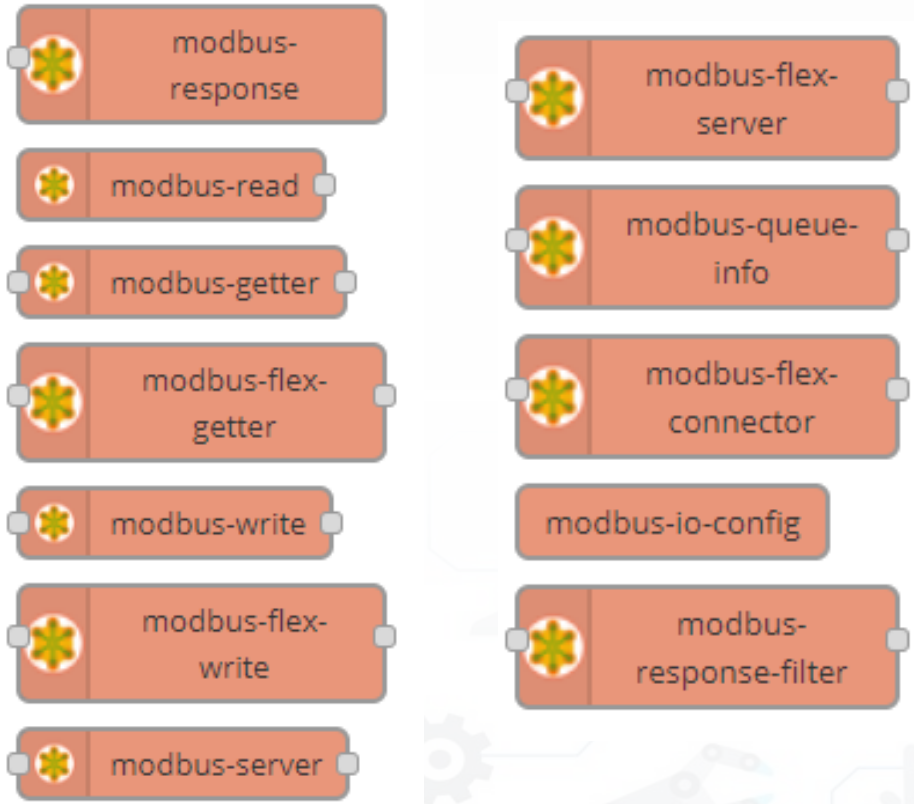
USB -- ModBus

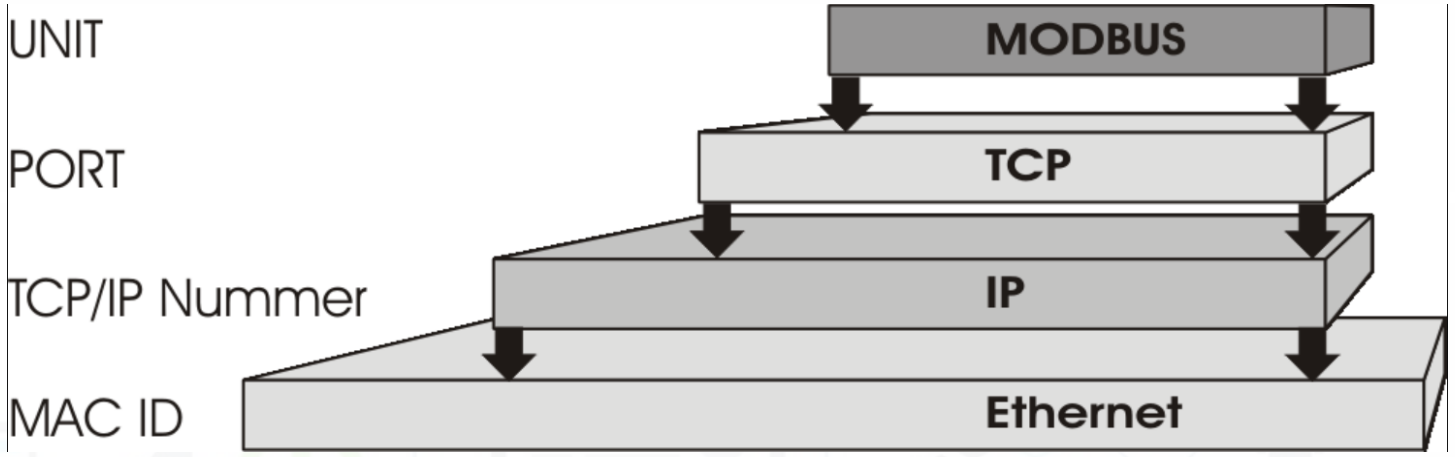
ModBus to Snap4City Gateway Edge





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


modbus-response
modbus-read
modbus-getter
modbus-flex-getter
modbus-write
modbus-flex-write
modbus-server
modbus-flex-server
modbus-queue-info
modbus-flex-connector
modbus-io-config
modbus-response-filter


UNIT
PORT
TCP/IP Nummer
MAC ID
MODBUS
TCP
IP
Ethernet

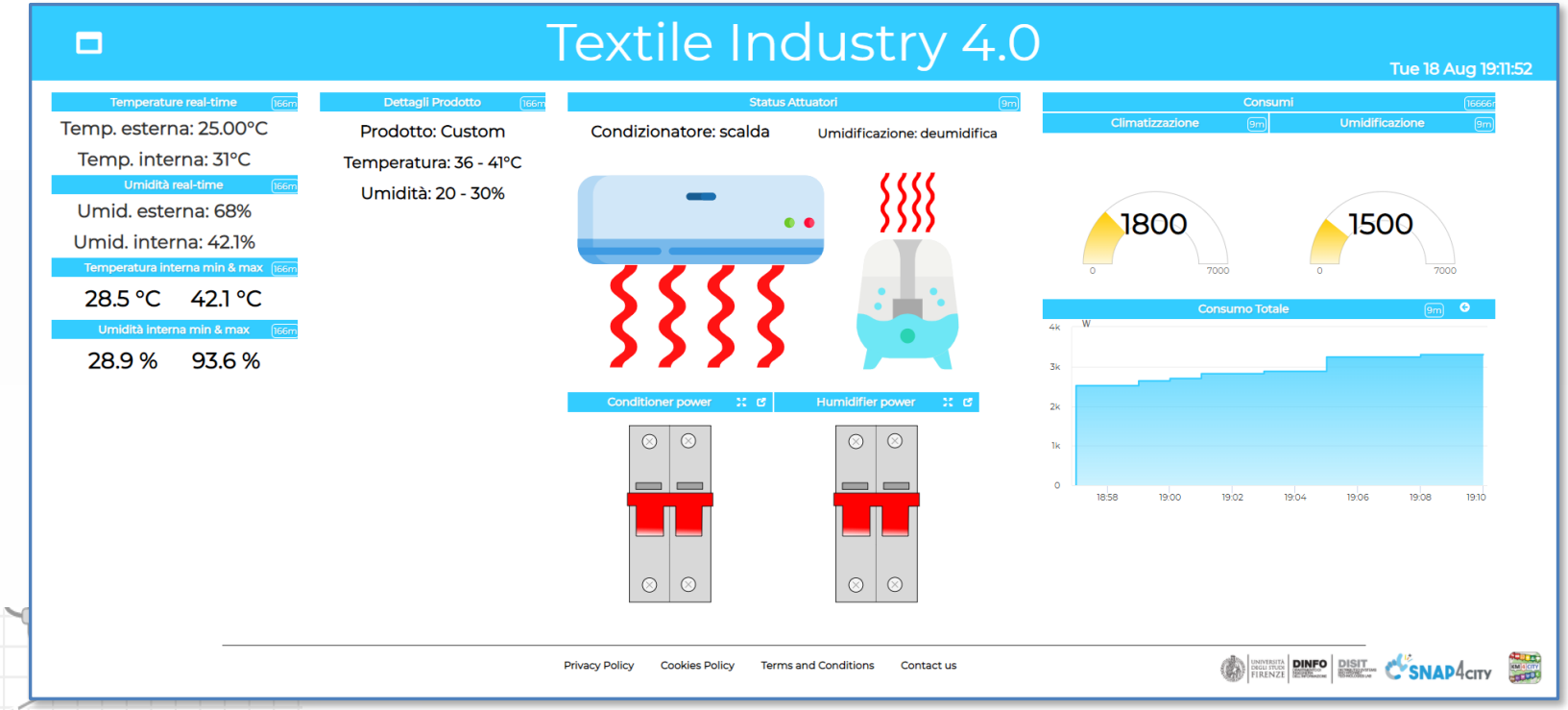
- ..

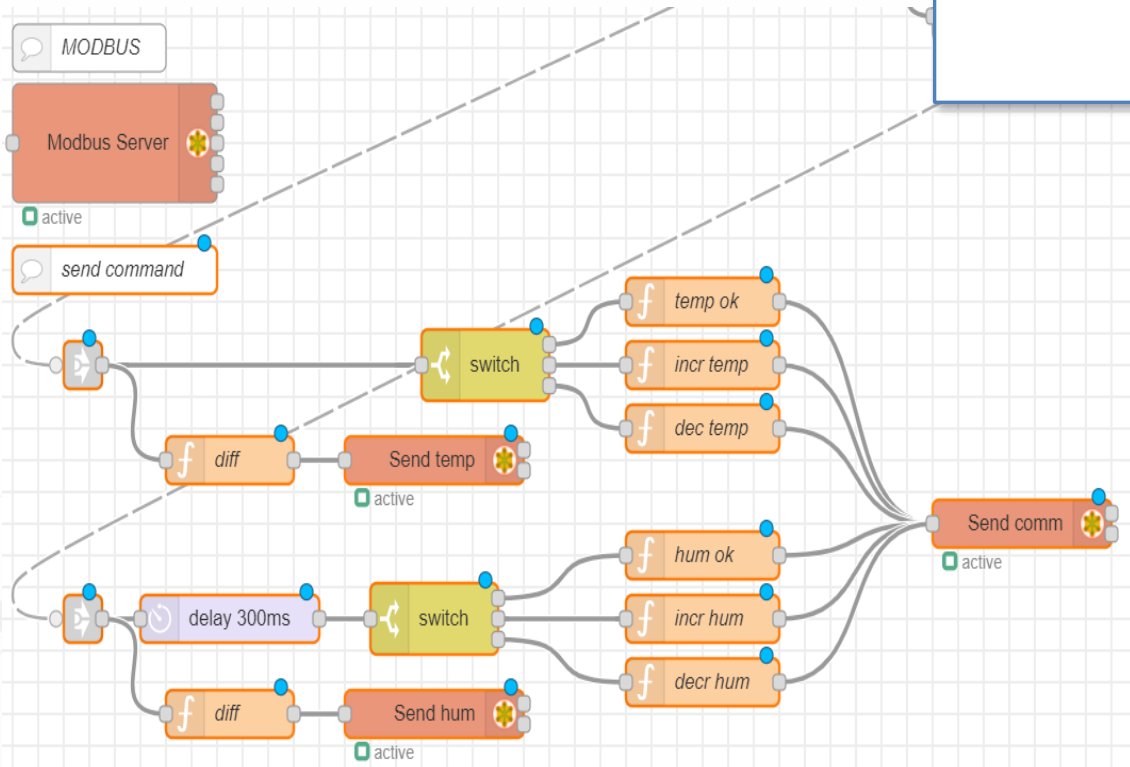
UNIVERSITÀ DEGLI STUDI FIRENZE

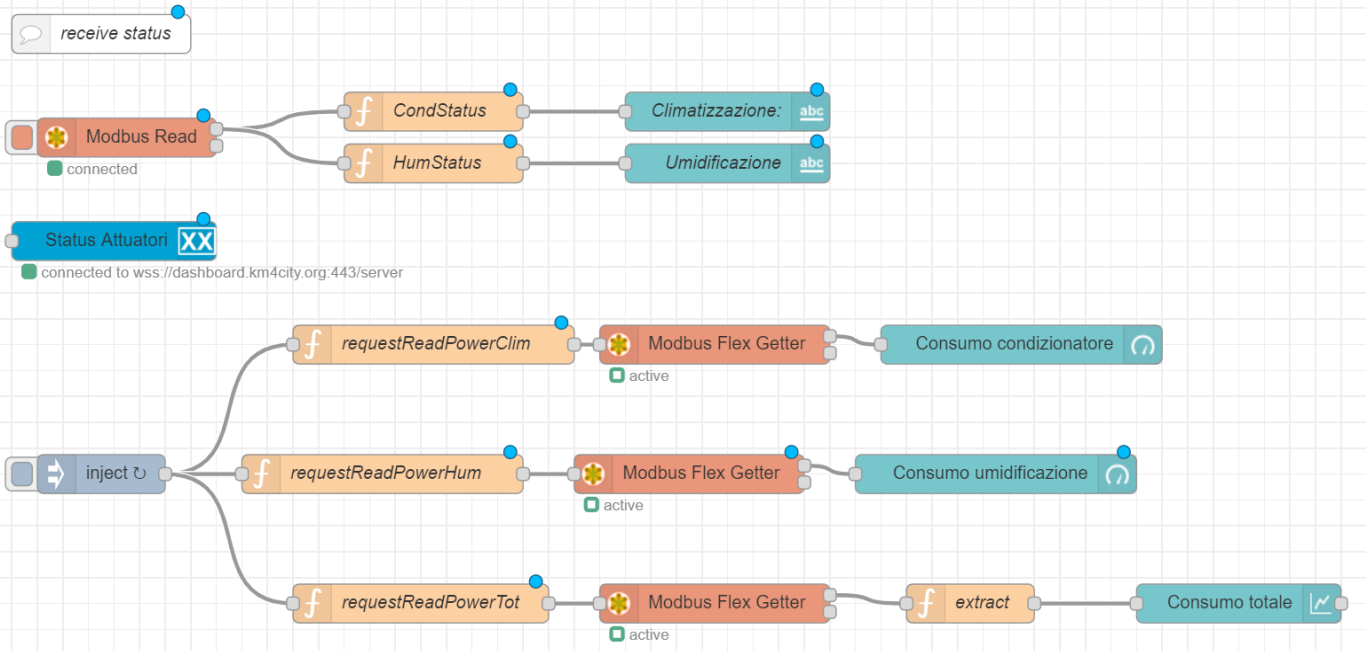
DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

Modbus







IoT App vs Smart Home
Snap4Home

Snap4Home

Sonoff: Controlling Energy Power

Philips Hue: Controlling Lights

Hue: Motion Control / Alarm

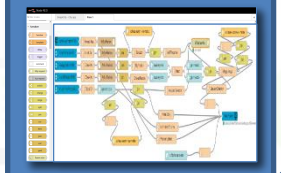
Measuring
Energy Consumption

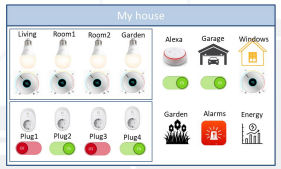
TP Link: Controlling / Measuring Energy Plugs

Alexa: Voice Control

IOT Edge:

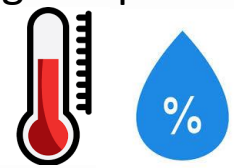
Raspberry pi: Node-RED + Snap4City





Local Control

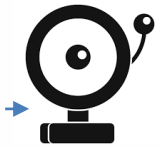
Measuring Temperature and Humidity



Garage Door

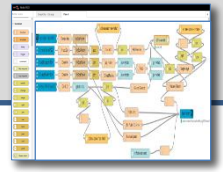
Window
Roller Shutters

Controlling Motors

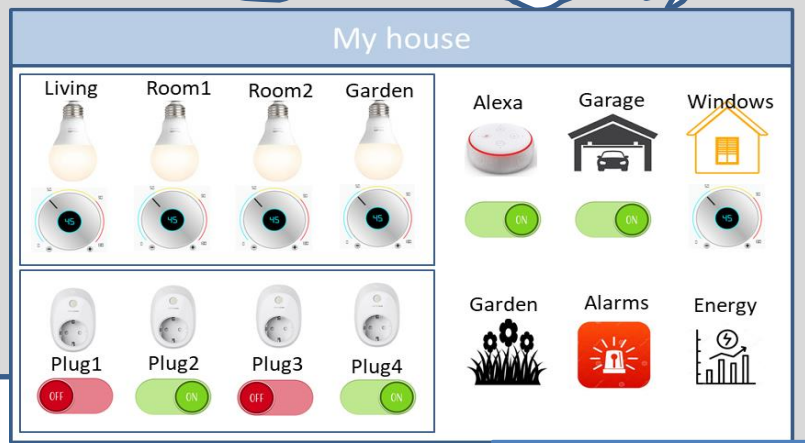


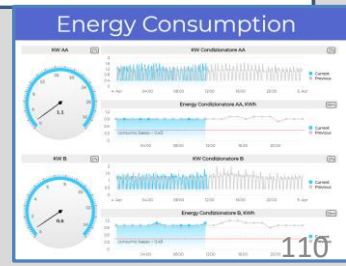
Alarm sound
and light

Controlling
Irrigators



SNAP4CITY

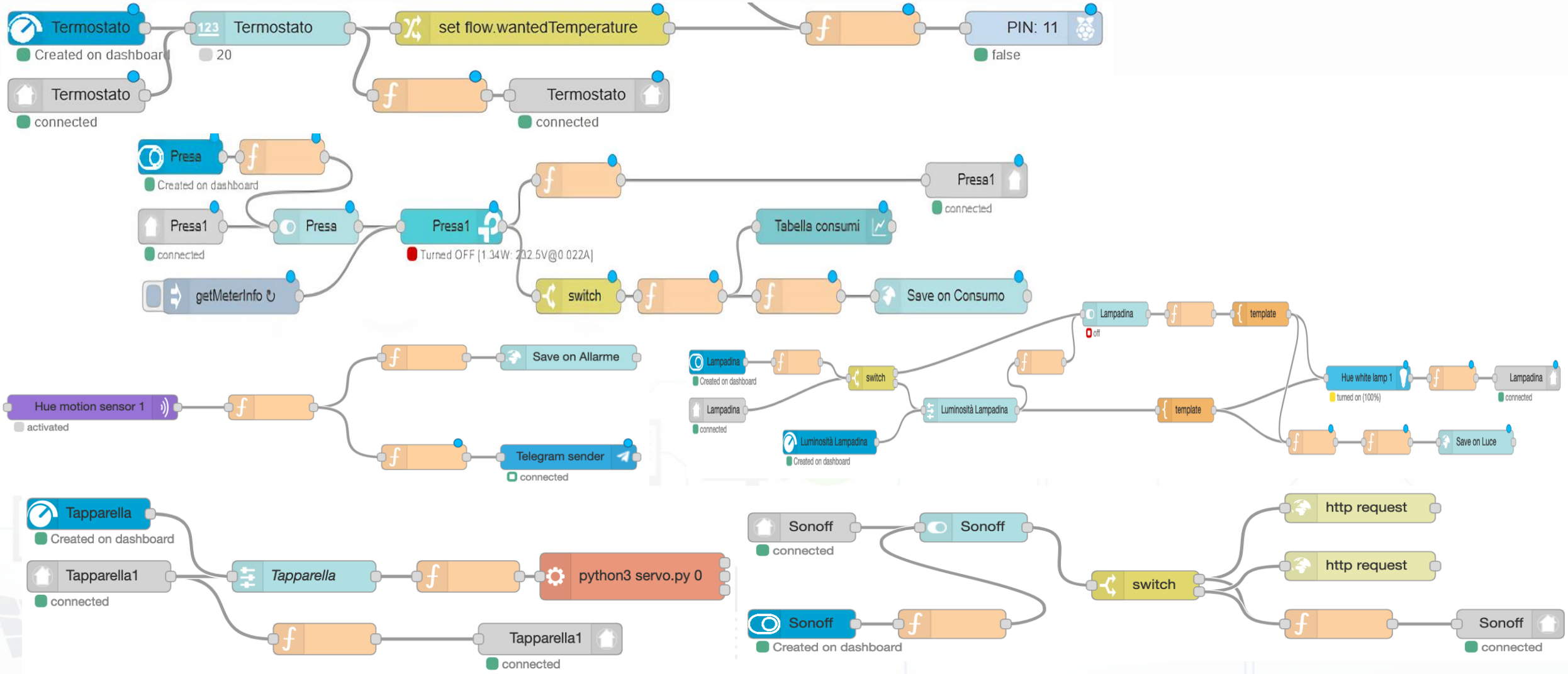




Environmental Contextual data from the city
Historical Data, Remote Control, Mobile App

https://www.snap4city.org/620

Example: IOT App on Snap4Home



Snap4Home

Hue Hub

Motion Control / Alarm

TP Link plugs: meter

Alexa: Voice Control

SNAP4CITY

IOT Edge:

Raspberry pi: Node-RED + Snap4City MicroService Library

5G gateway

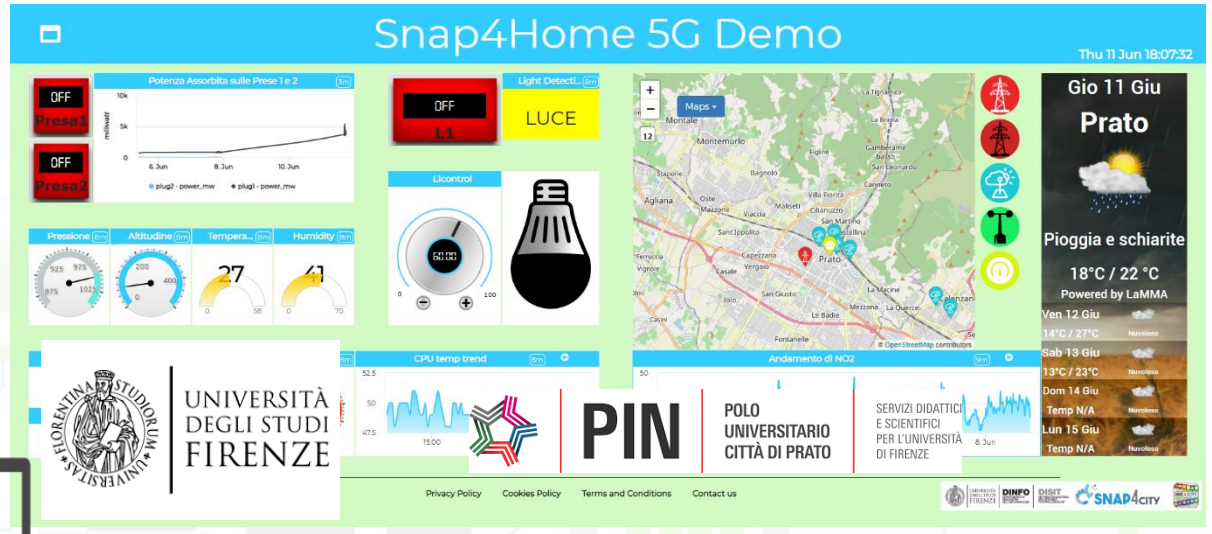
FIWARE Orion Broker

Dashboards

Advanced Smart City API

SNAP4CITY

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



open fiber WIND 3 ZTE estra

https://www.snap4city.org/369

Snap4Home

Philips Hue: Controlling Lights

Hue Hub

Hue: Motion Control / Alarm

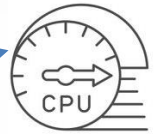
TP Link: Controlling / Measuring Energy Plugs

Alexa: Voice Control

IOT Edge:

Raspberry pi: Node-RED + Snap4City

Measuring: Temperature, Humidity, light in the room

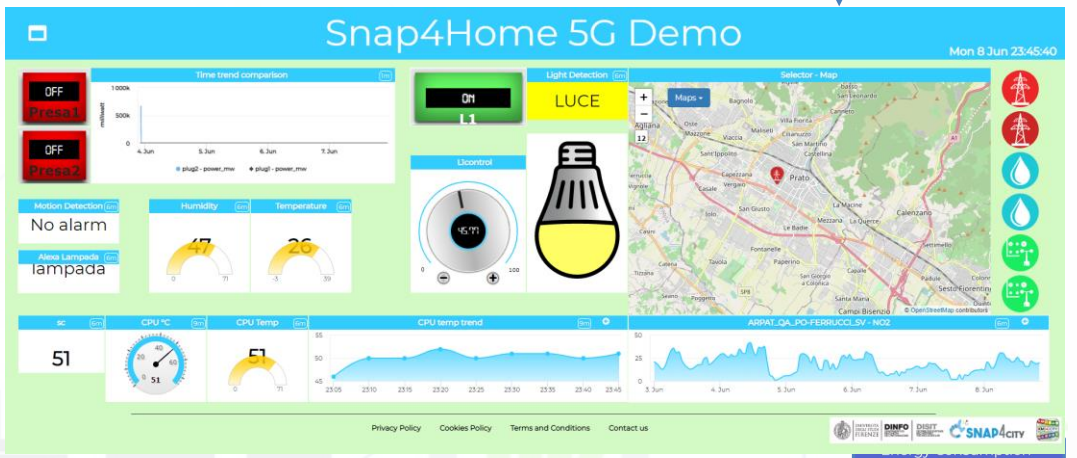


Monitoring: CPU clock, status

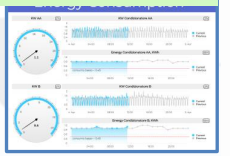
5G gateway

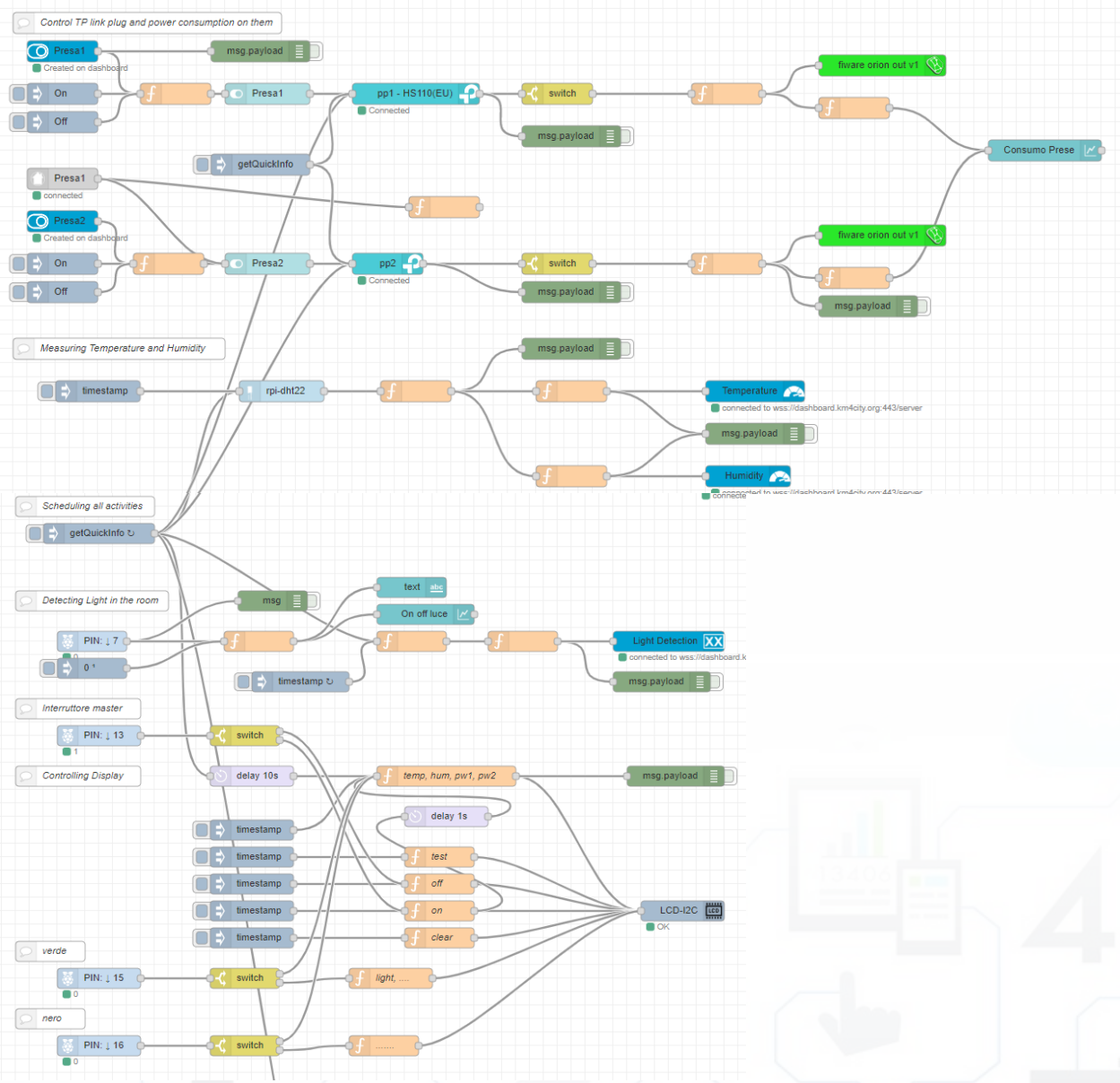
Internet

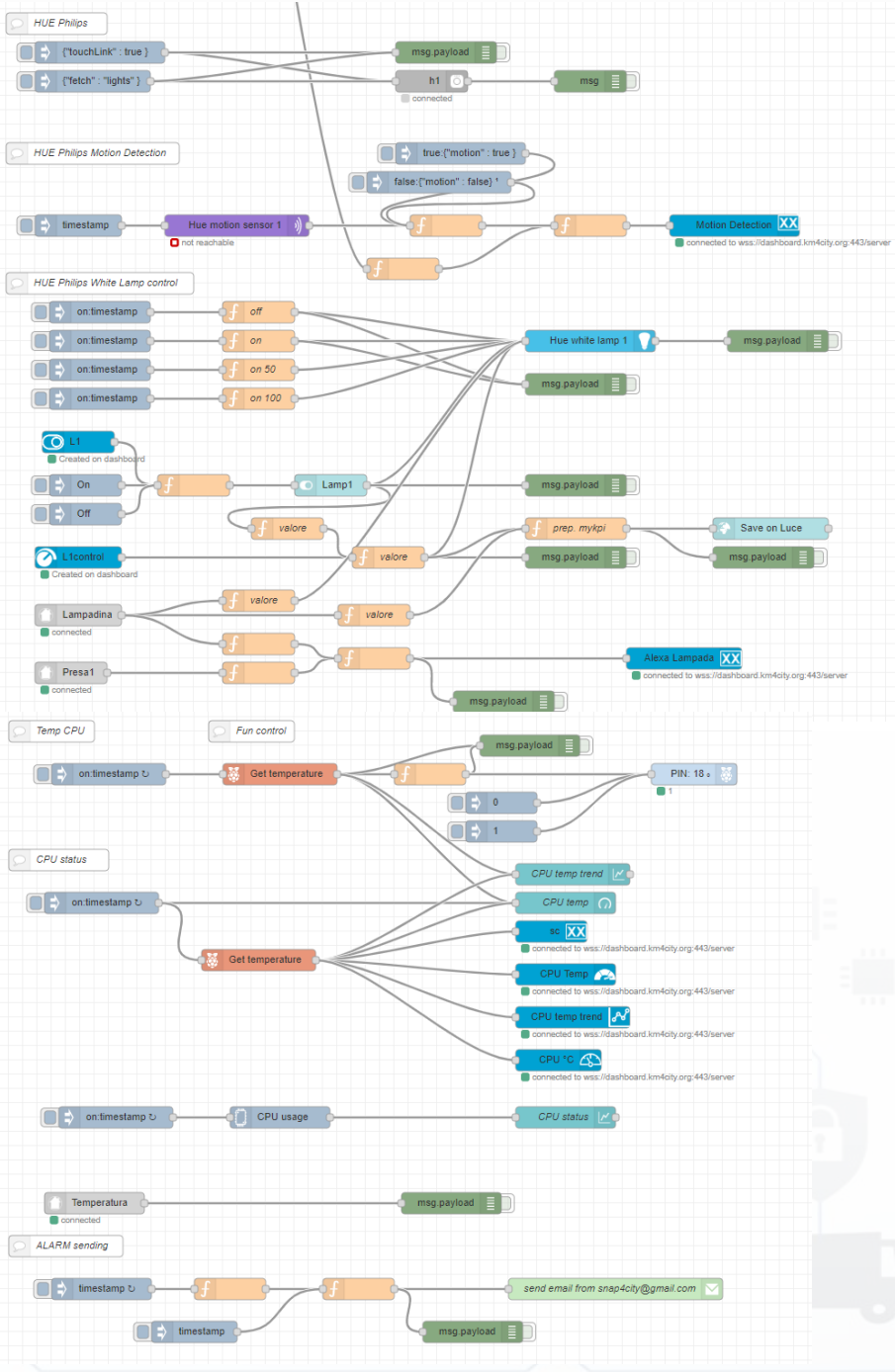
SNAP4CITY



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



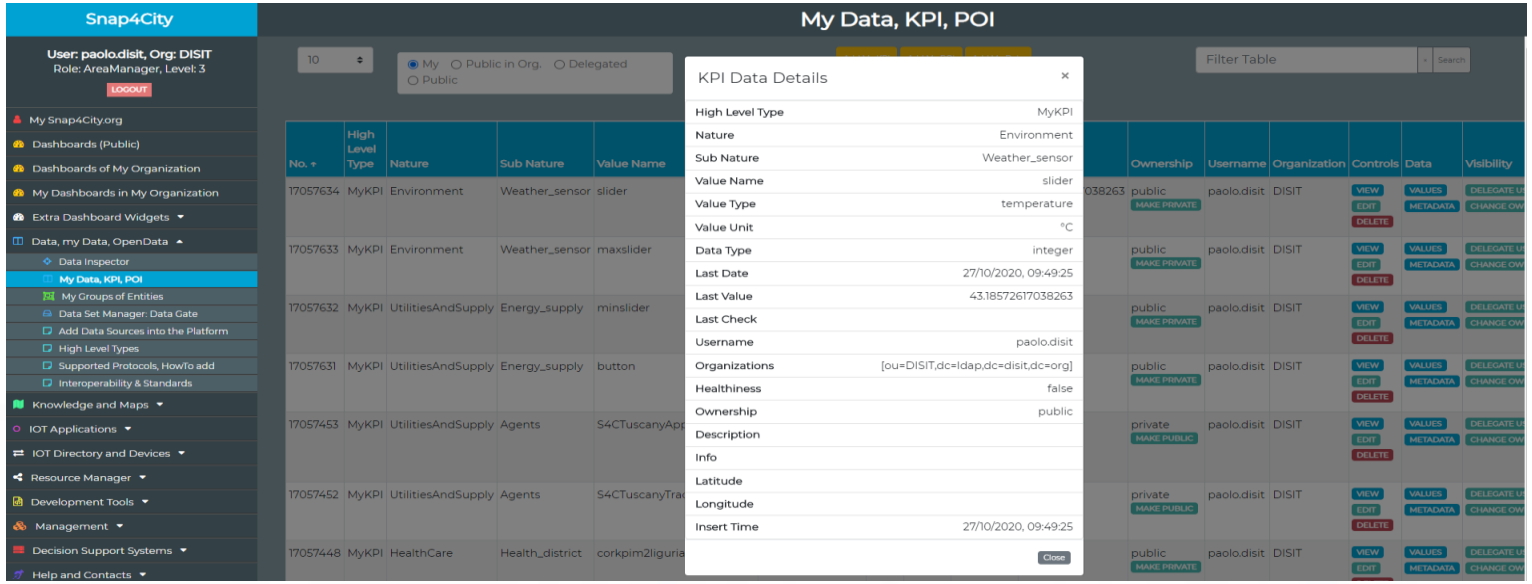

Control TP link plug and power consumption on them
Presa1
Created on dashboard
msg.payload
On
Off
Presa1
pp1 - HS110(EU)
Connected
switch
fiware orion out v1
msg.payload
Consumo Prese
getQuickInfo
Presa1
connected
Presa2
Created on dashboard
On
Off
Presa2
pp2
Connected
switch
fiware orion out v1
msg.payload
Measuring Temperature and Humidity
timestamp
rpi-dht22
msg.payload
Temperature
connected to wss://dashboard.km4city.org:443/server
msg.payload
Humidity
Scheduling all activities
getQuickInfo
Detecting Light in the room
msg
text
On off luce
PIN: 7
0
Light Detection
connected to wss://dashboard.k
msg.payload
timestamp
Interruttore master
PIN: 13
1
switch
Controlling Display
delay 10s
temp, hum, pw1, pw2
msg.payload
delay 1s
timestamp
timestamp
timestamp
timestamp
timestamp
test
off
on
clear
LCD-I2C
OK
verde
PIN: 15
0
switch
light, ...
nero
PIN: 16
0
switch
.....

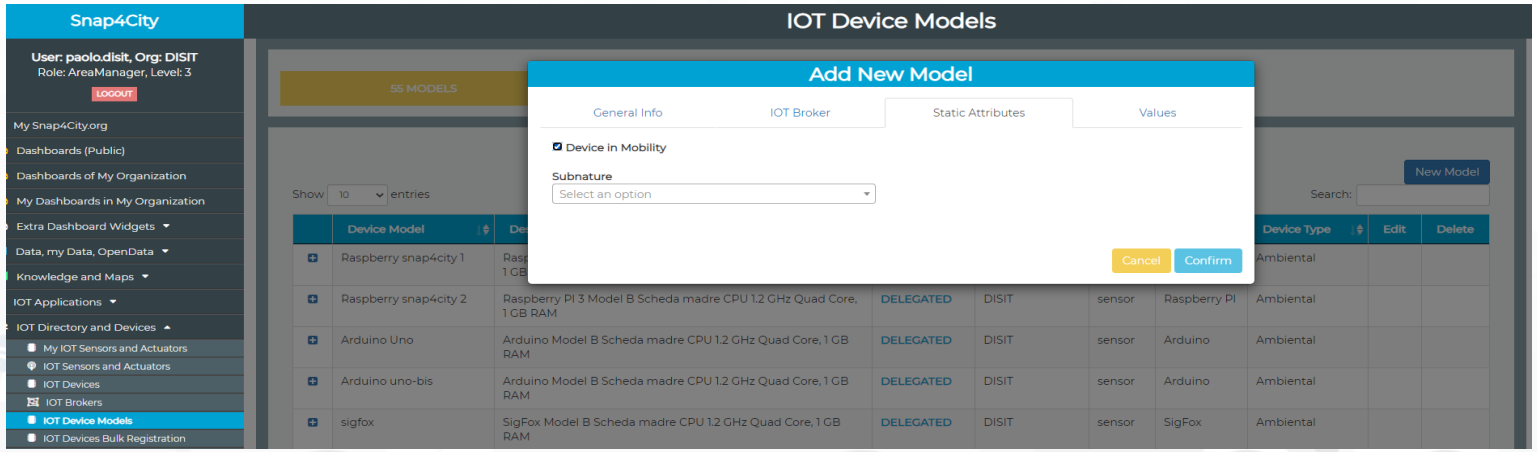

HUE Philips
{"touchLink" : true }
{"fetch" : "lights" }
msg.payload
h1
connected
msg
HUE Philips Motion Detection
true:{"motion" : true }
false:{"motion" : false} *
timestamp
Hue motion sensor 1
not reachable
Motion Detection
connected to wss://dashboard.km4city.org:443/server
HUE Philips White Lamp control
on:timestamp
on:timestamp
on:timestamp
on:timestamp
off
on
on 50
on 100
Hue white lamp 1
msg.payload
msg.payload
L1
Created on dashboard
On
Off
Lamp1
msg.payload
valore
prep. mykpi
Save on Luce
msg.payload
L1control
Created on dashboard
valore
valore
msg.payload
Lampadina
connected
valore
valore
Presa1
connected
Alexa Lampada
connected to wss://dashboard.km4city.org:443/server
msg.payload
Temp CPU
Fun control
on:timestamp
Get temperature
msg.payload
PIN: 18
1
0
1
CPU status
on:timestamp
Get temperature
CPU temp trend
CPU temp
sc
connected to wss://dashboard.km4city.org:443/server
CPU Temp
connected to wss://dashboard.km4city.org:443/server
CPU temp trend
connected to wss://dashboard.km4city.org:443/server
CPU °C
connected to wss://dashboard.km4city.org:443/server
on:timestamp
CPU usage
CPU status
Temperatura
connected
msg.payload
ALARM sending
timestamp
timestamp
send email from snap4city@gmail.com
msg.payload

Moving IoT/WoT Entities, Tracking Devices

Working with Sensor Data from Moving Devices

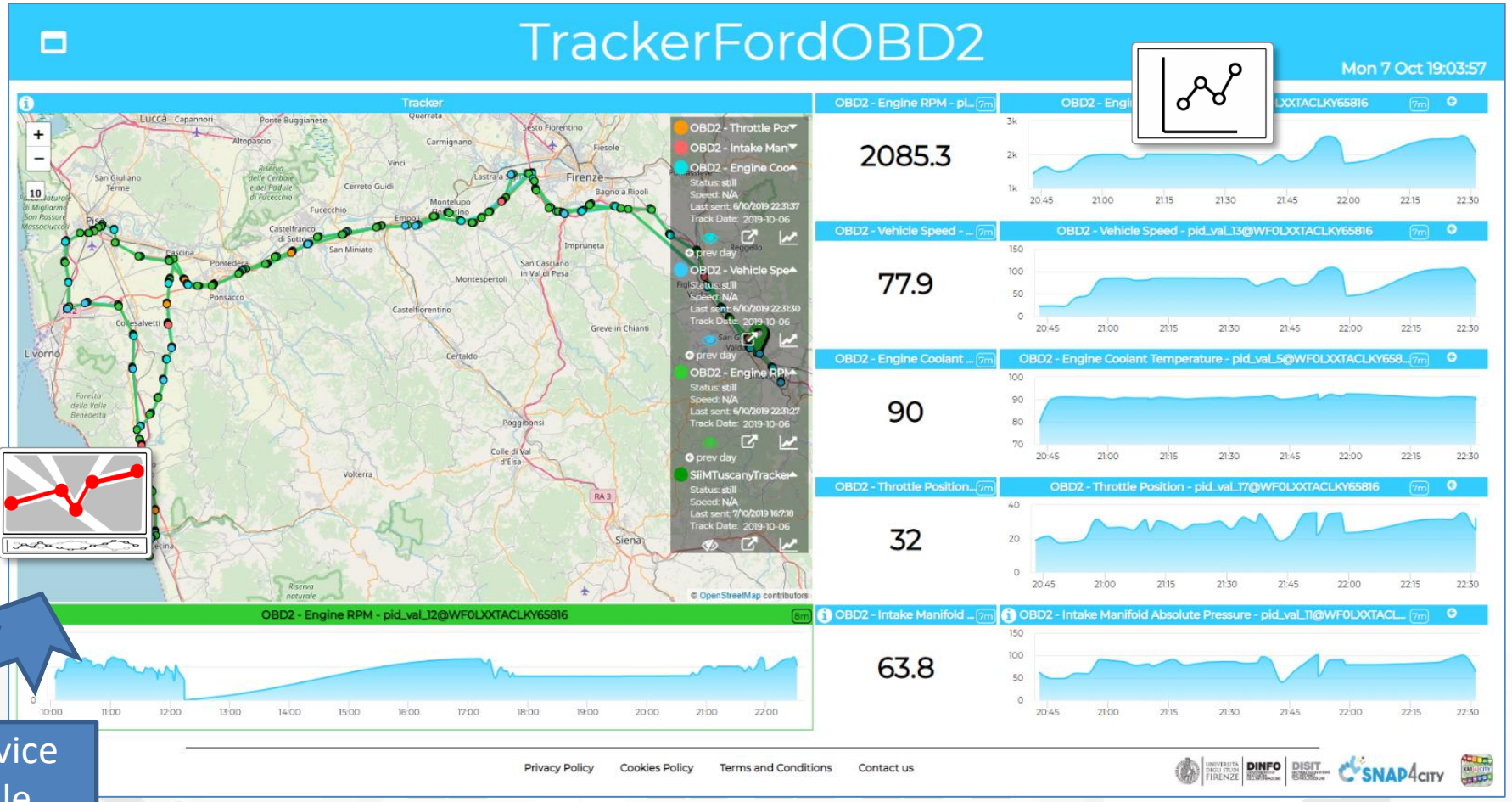
- Moving data can be collected by using:
  - **MyKPI**: in which each MyKPI has a ValueName, Unit, Type, etc.. And also GPS location
  - **IOT Device in Mobility**: which generates a new HLT SensorMobile





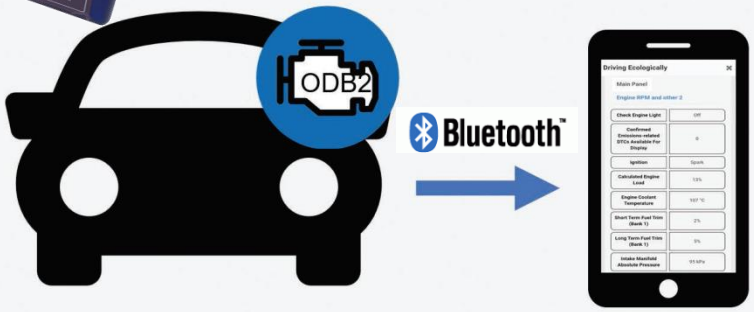
MyKPI: Tracking of Devices and Mobiles

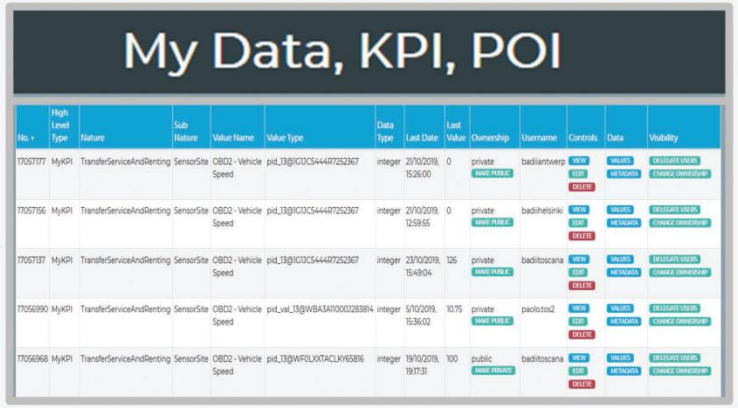
- Real Time Trajectories for
  - Mobile Phone
  - Moving IOT Devices
  - OBU, Vehicular Kits
  - Multiple tracks
  - Day by day
- Micro Application

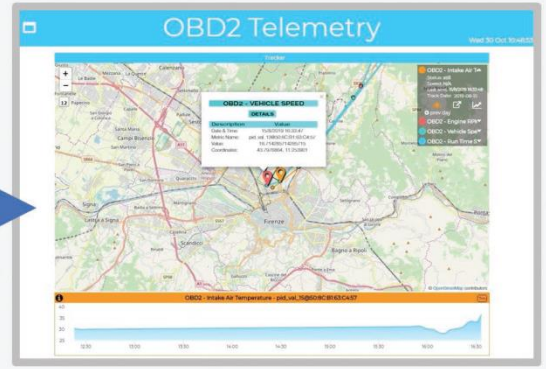


IOE – Vehicle Monitoring

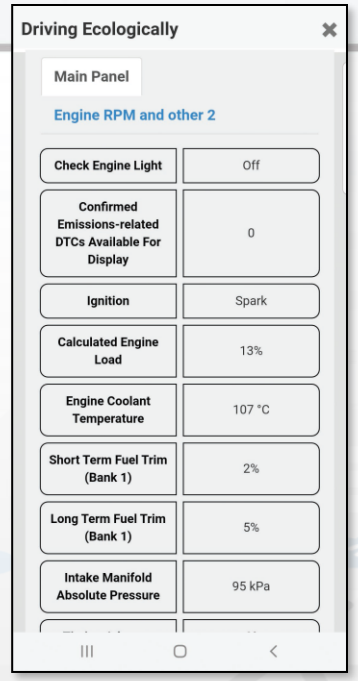
CANBUS sniffer

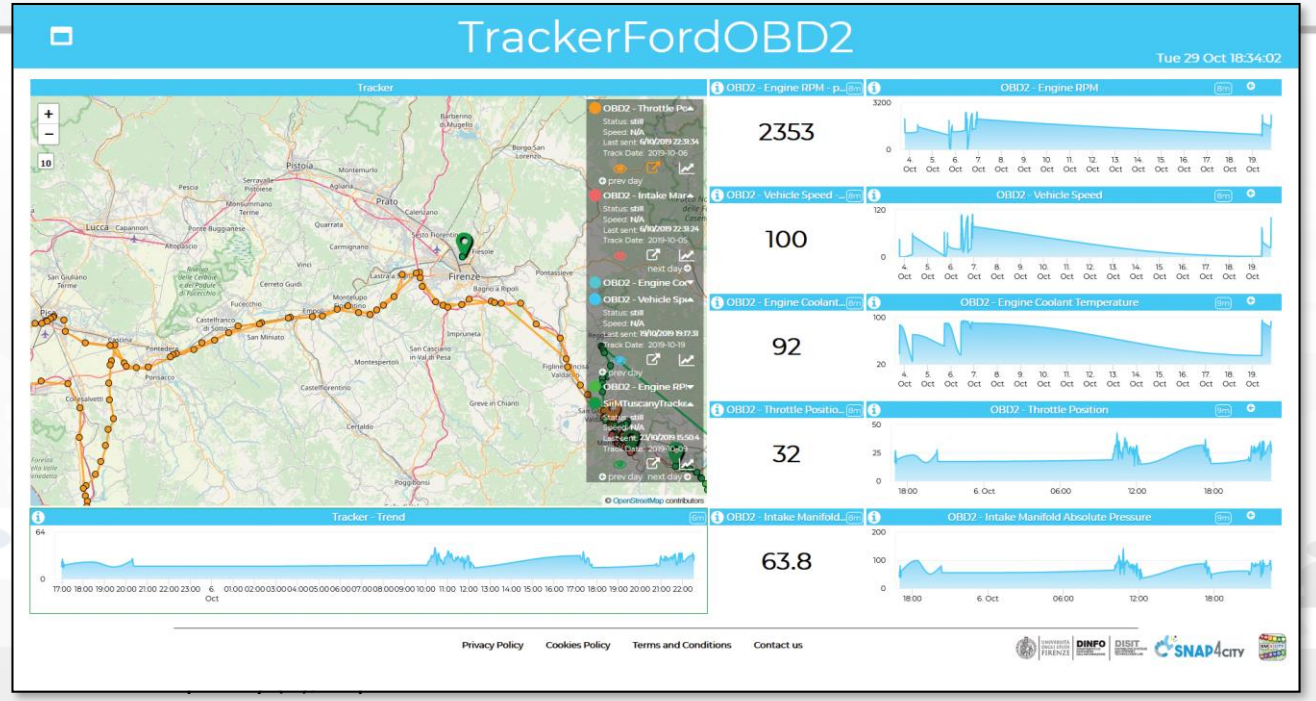






Tuscany in a Snap Mobile App on Android





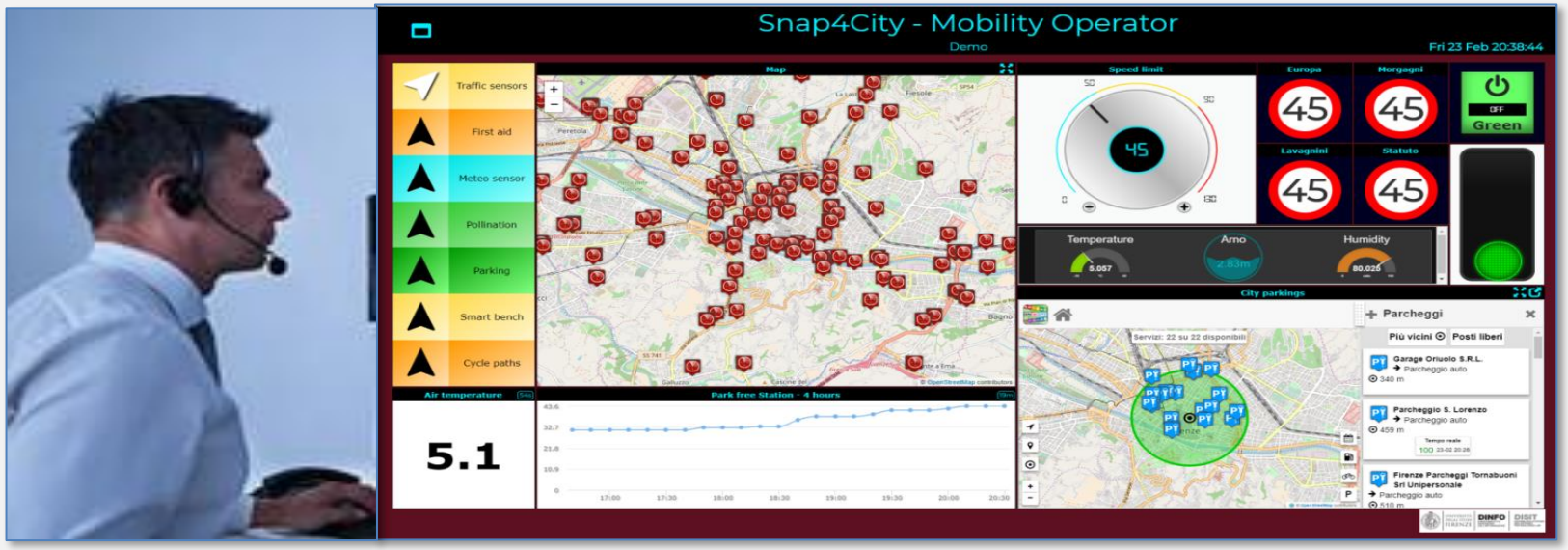
IOT App vs Smart City Solutions

City Dashboard + IOT App

Control Room Operator

Would like to:

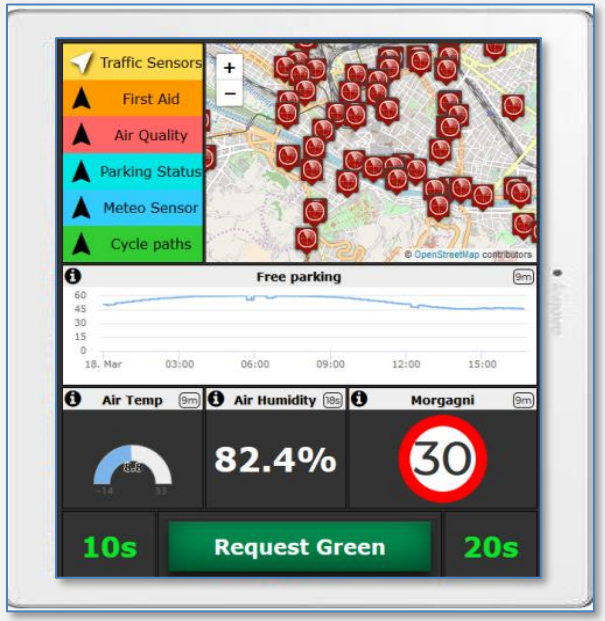
- **Monitor** traffic flow, Environment, Car parking, Cycling, First aid, temp., ..
- **Act and** monitor Dynamic Plates
- **Act and** monitor red lights

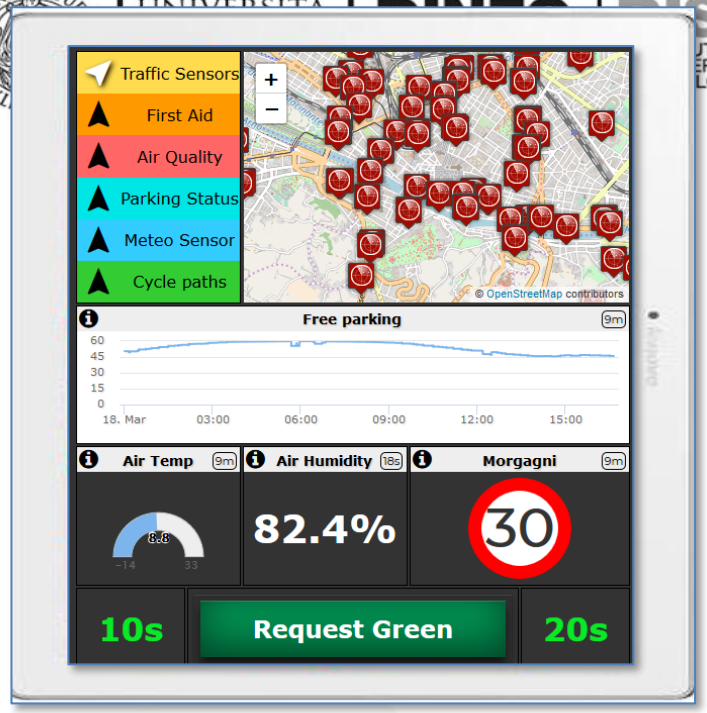


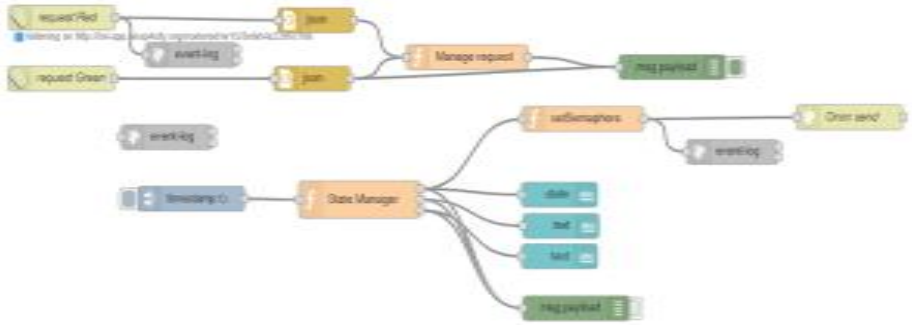
Driver, Policeman

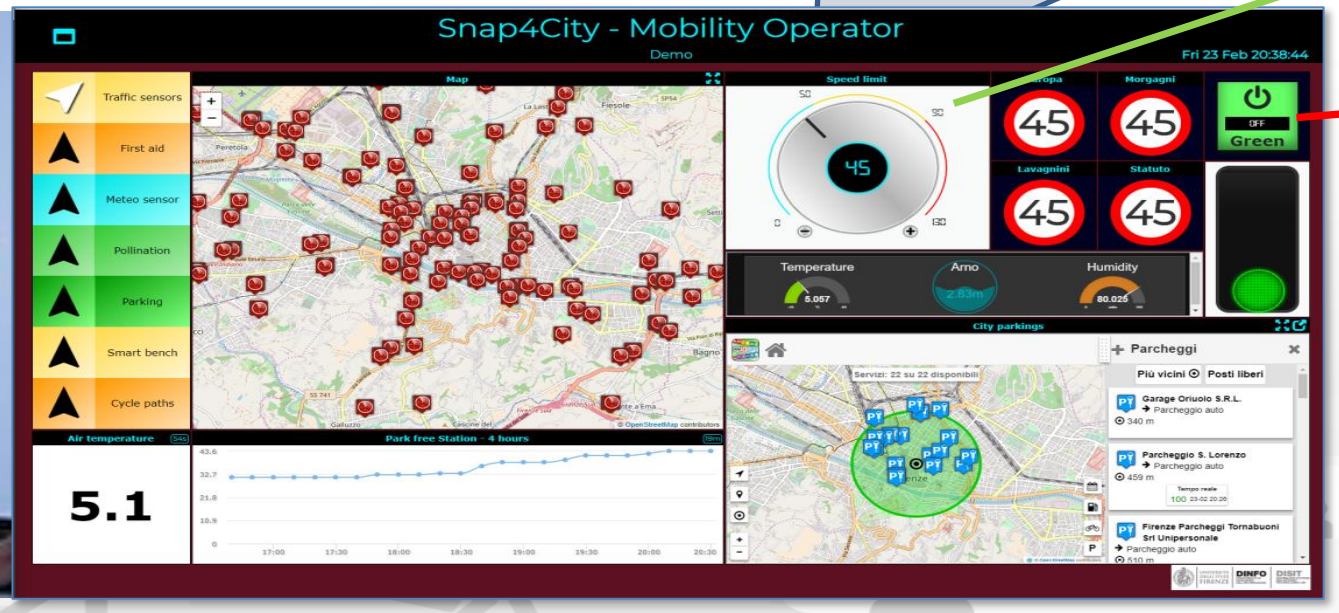
Would like to:

- Monitor traffic, Parking, traffic events, speed limit, …
- **Act and** monitor red lights




Traffic Sensors
First Aid
Air Quality
Parking Status
Meteo Sensor
Cycle paths
© OpenStreetMap contributors
Free parking
9m
60
45
30
15
0
18. Mar
03:00
06:00
09:00
12:00
15:00
Air Temp
9m
Air Humidity
18s
Morgagni
9m
82.4%
30
10s
Request Green
20s



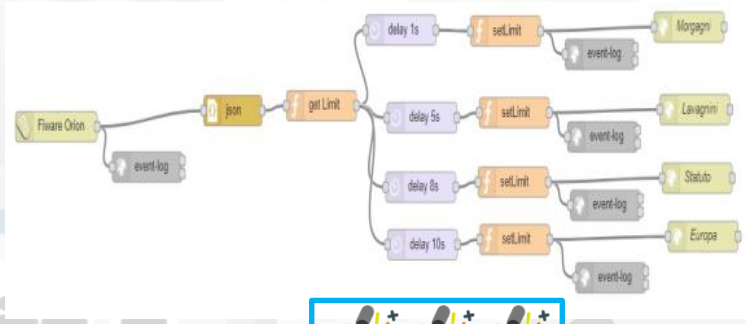

Snap4City - Mobility Operator
Demo
Fri 23 Feb 20:38:44
Traffic sensors
First aid
Meteo sensor
Pollination
Parking
Smart bench
Cycle paths
Map
Speed limit
Morgagni
Lavagnini
Statuto
Green
Temperature
Arno
Humidity
City parkings
Parcheggi
Air temperature
Park free Station - 4 hours
5.1


30


30

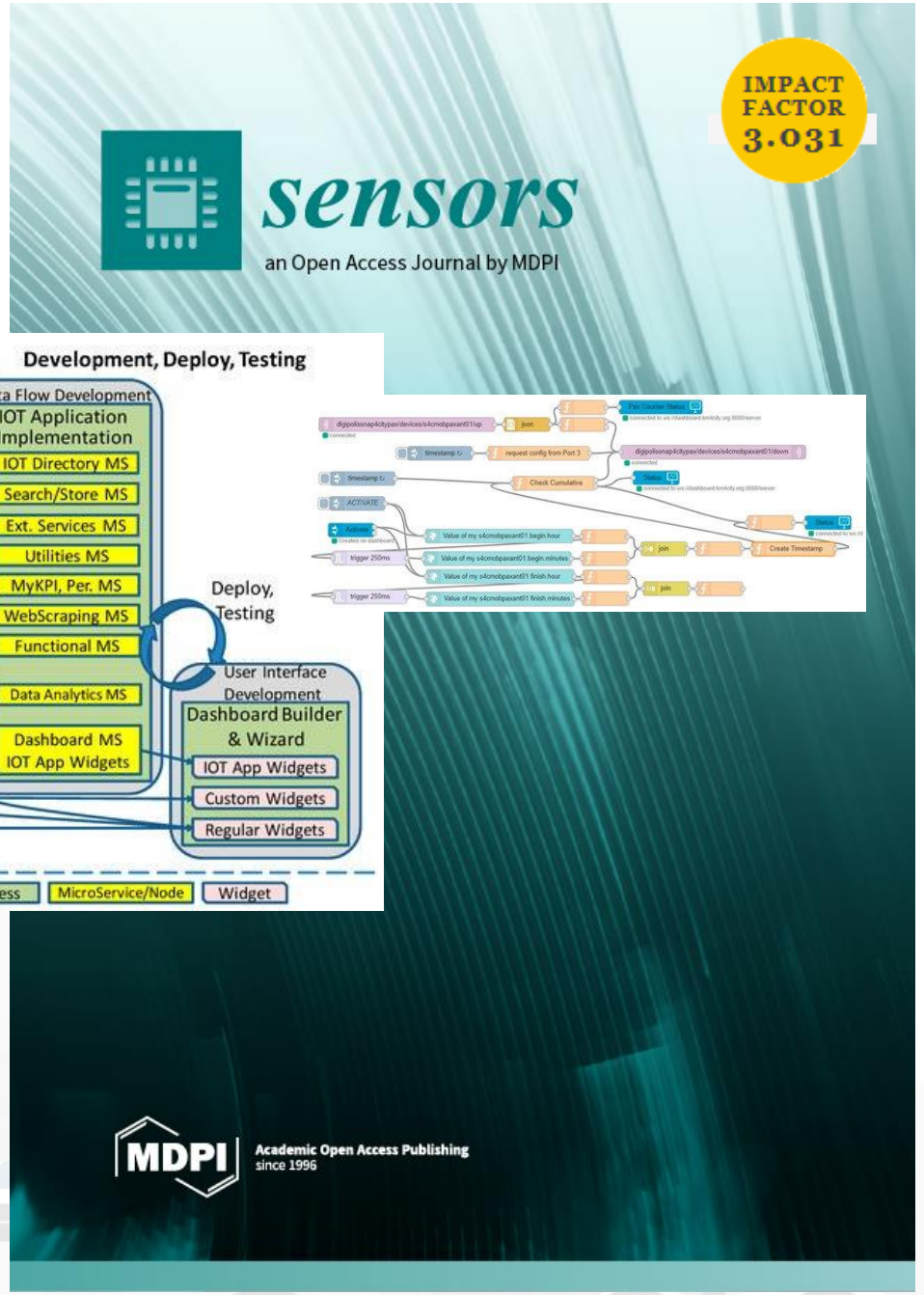

30


30


Fiware Orion
event-log
json
get Limit
delay 1s
delay 5s
delay 8s
delay 10s
setLimit
event-log
Morgagni
Lavagnini
Statuto
Europa

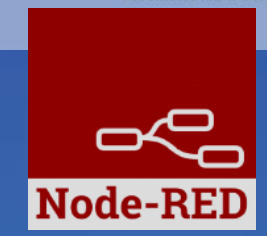
MicroServices Suite for Smart City

- Badii, C.; Bellini, P.; Difino, A.; Nesi, P.; Pantaleo, G.; Paolucci, M. MicroServices Suite for Smart City Applications.
- *Sensors* **2019**, *19*, 4798.
- https://www.mdpi.com/1424-8220/19/21/4798/pdf



TOP

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



IoT App / Proc.Logic

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

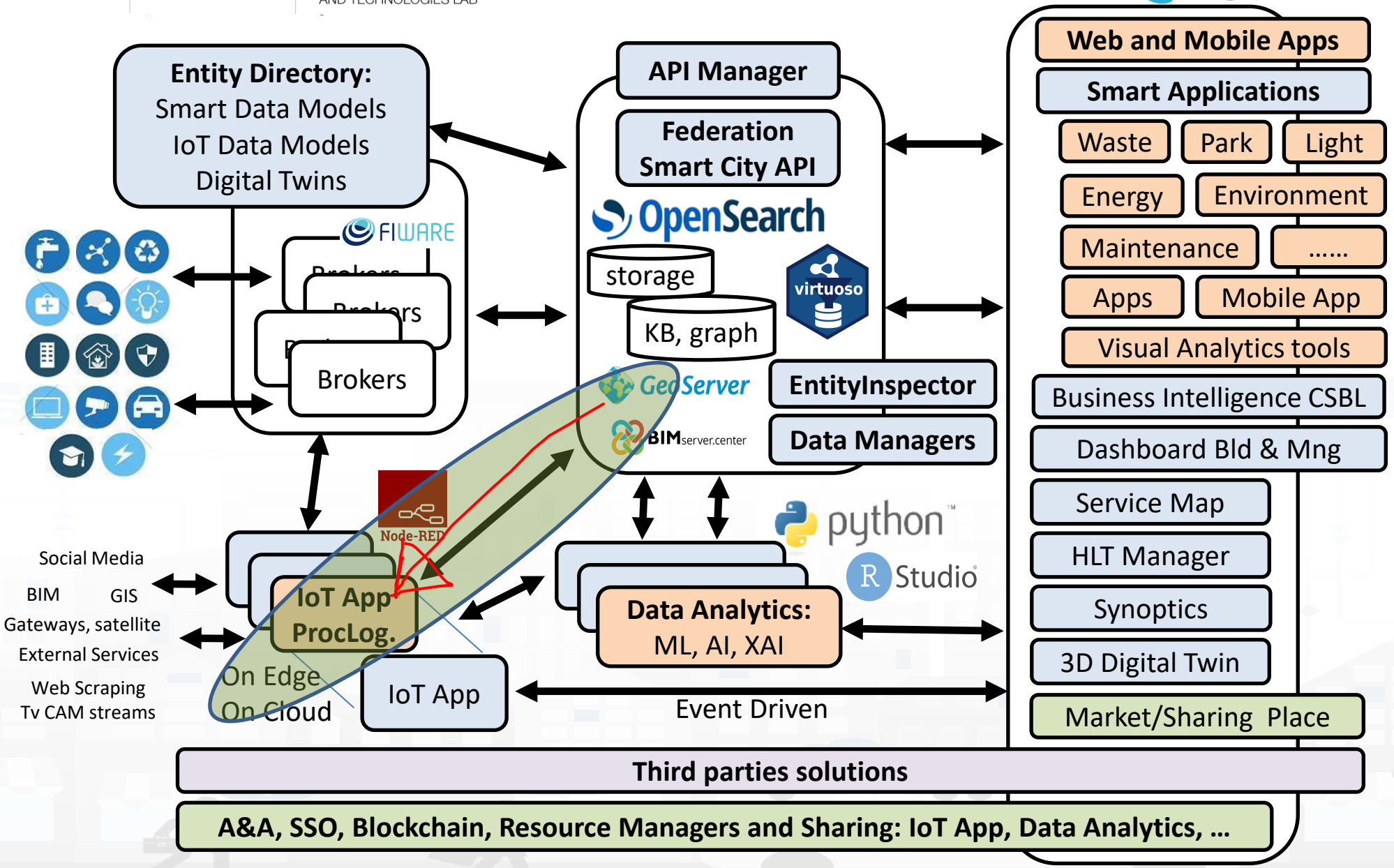
Node-RED

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

Part 4

Part 5

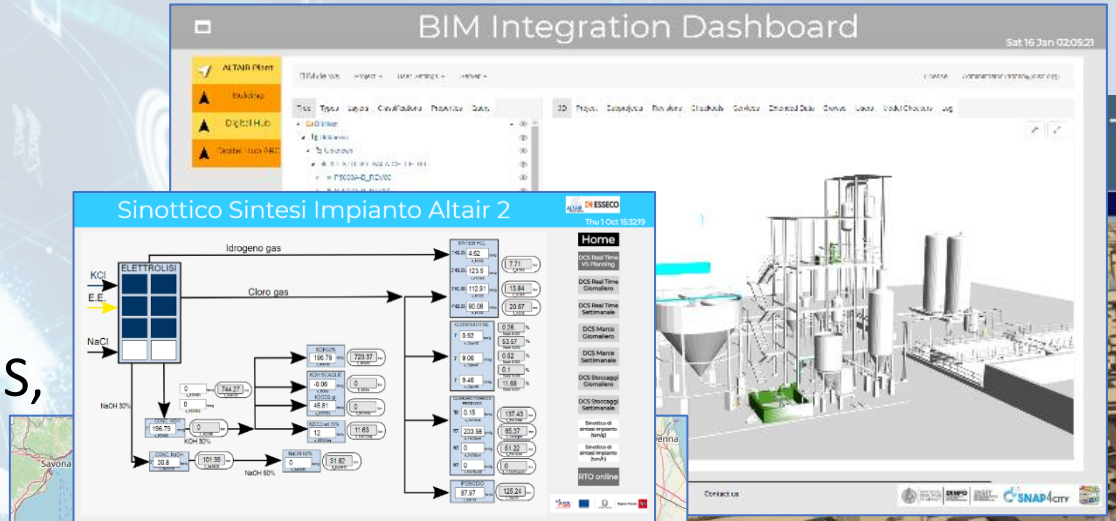
Tech Arch

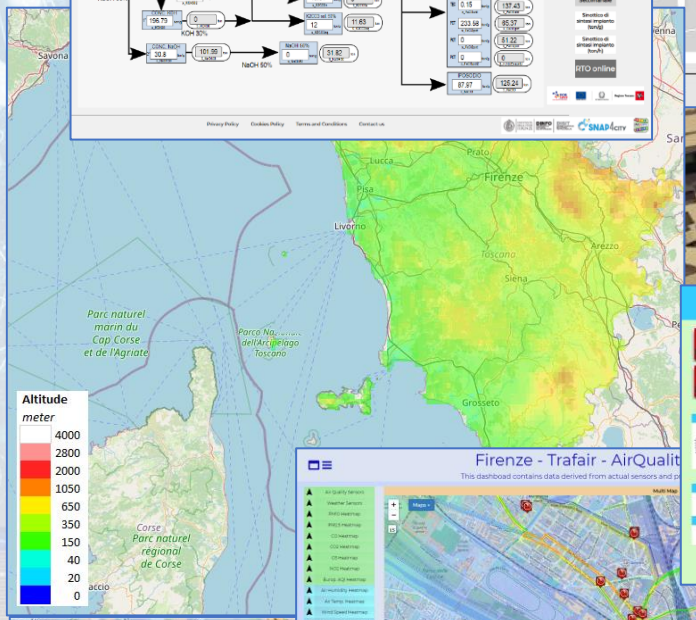

Entity Directory:
Smart Data Models
IoT Data Models
Digital Twins
FIWARE
Brokers
Brokers
Brokers
API Manager
Federation
Smart City API
OpenSearch
storage
KB, graph
virtuoso
GeoServer
BIMserver.center
EntityInspector
Data Managers
Node-RED
python
RStudio
Social Media
BIM
GIS
Gateways, satellite
External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

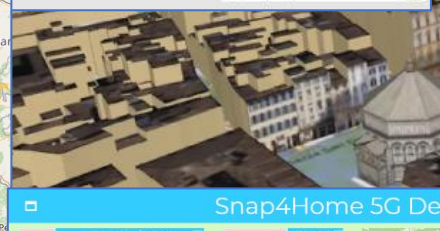
High Level Types

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.

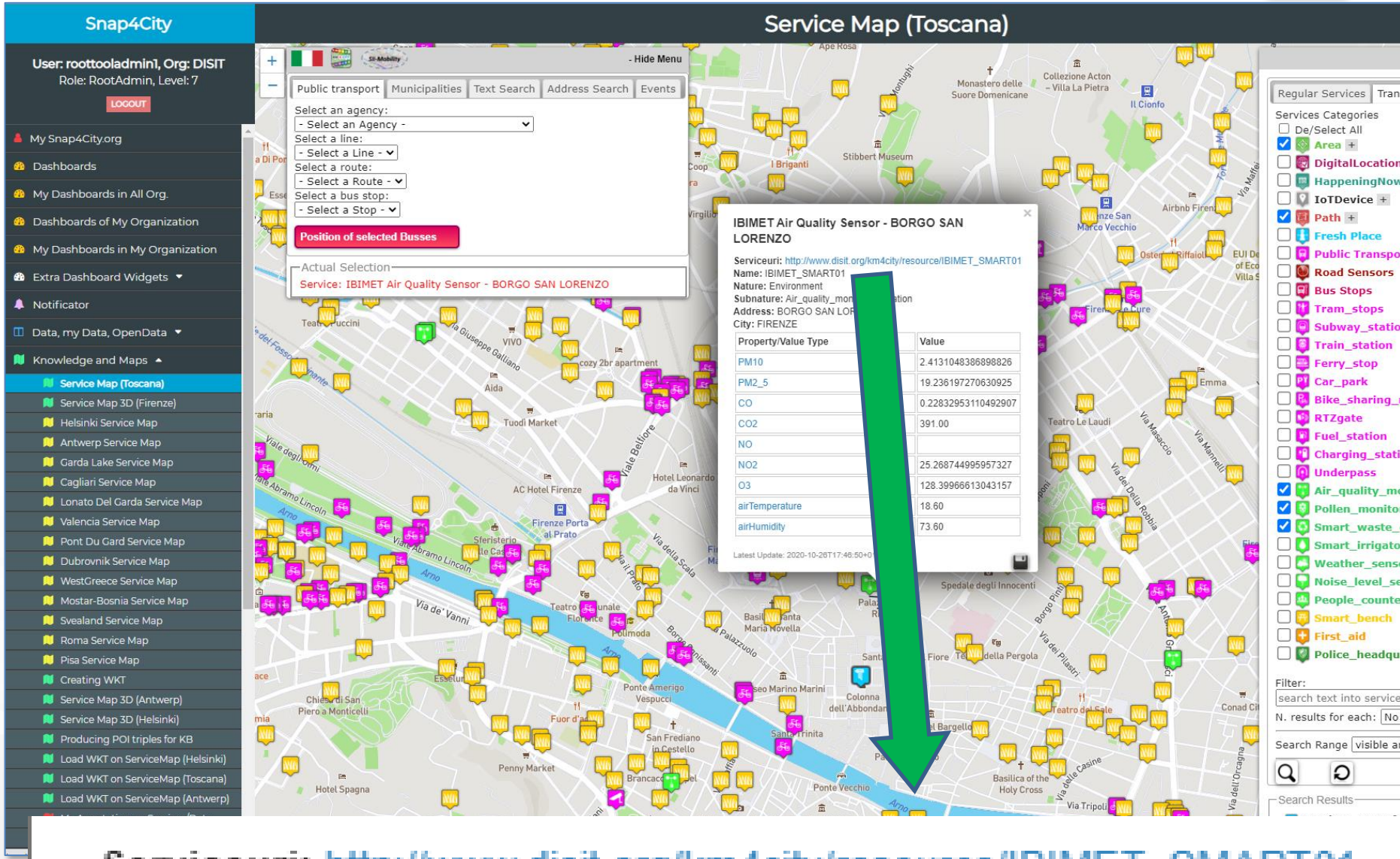






ServiceMap

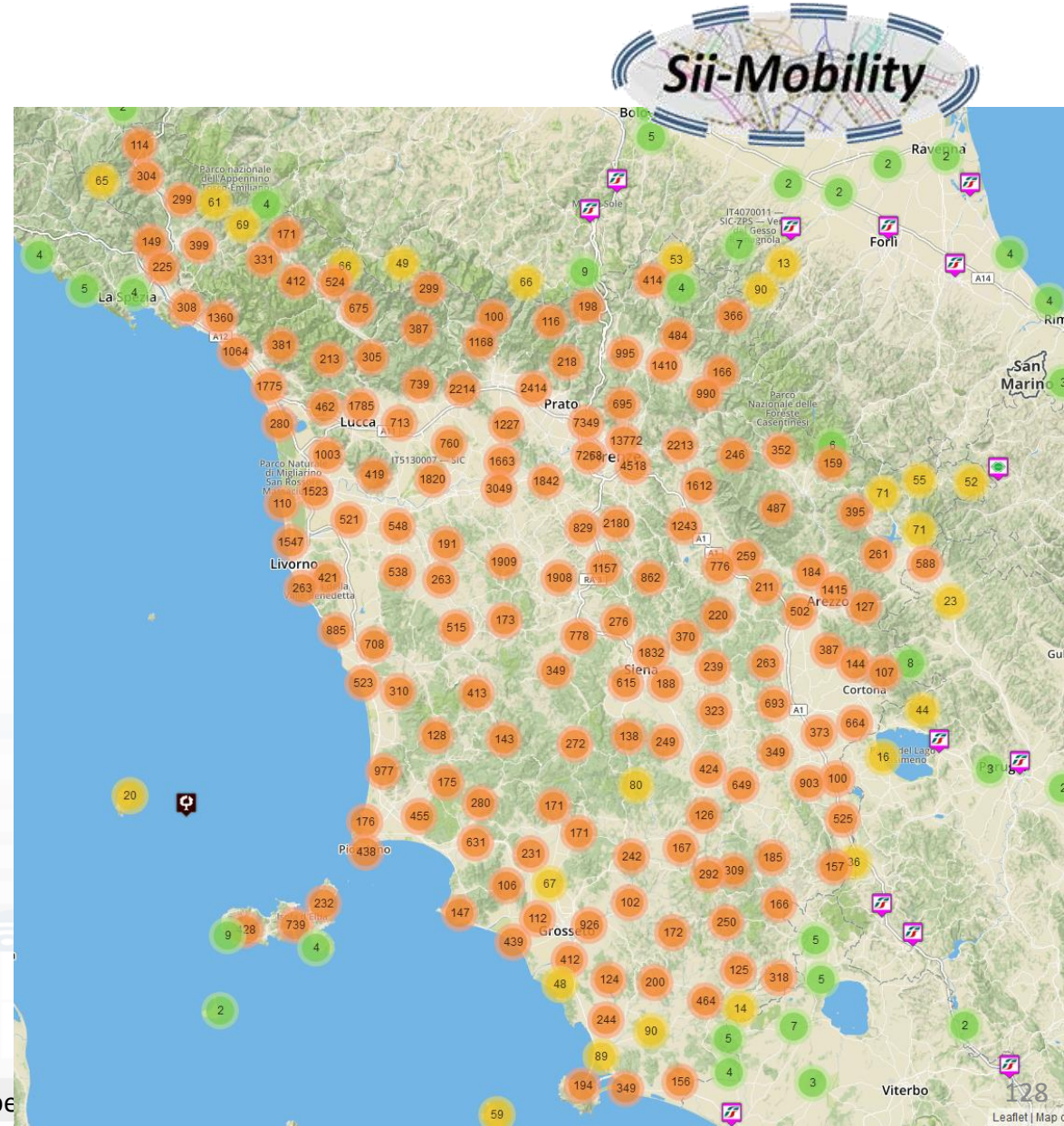
- **For PUBLIC**:
  - IOT Devices, Sensors,
  - Sensor mobile,
  - Actuators,
  - Virtual Sensors,
  - POI, etc.
- See as
  - ServiceURI



Serviceuri: http://www.disit.org/km4city/resource/IBIMET_SMART01

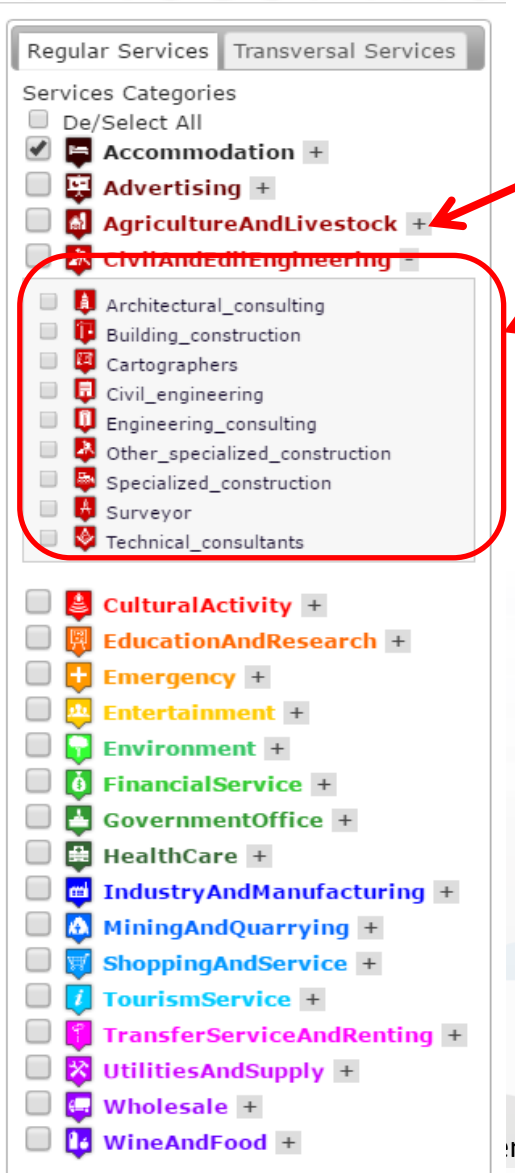
Data Domains

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



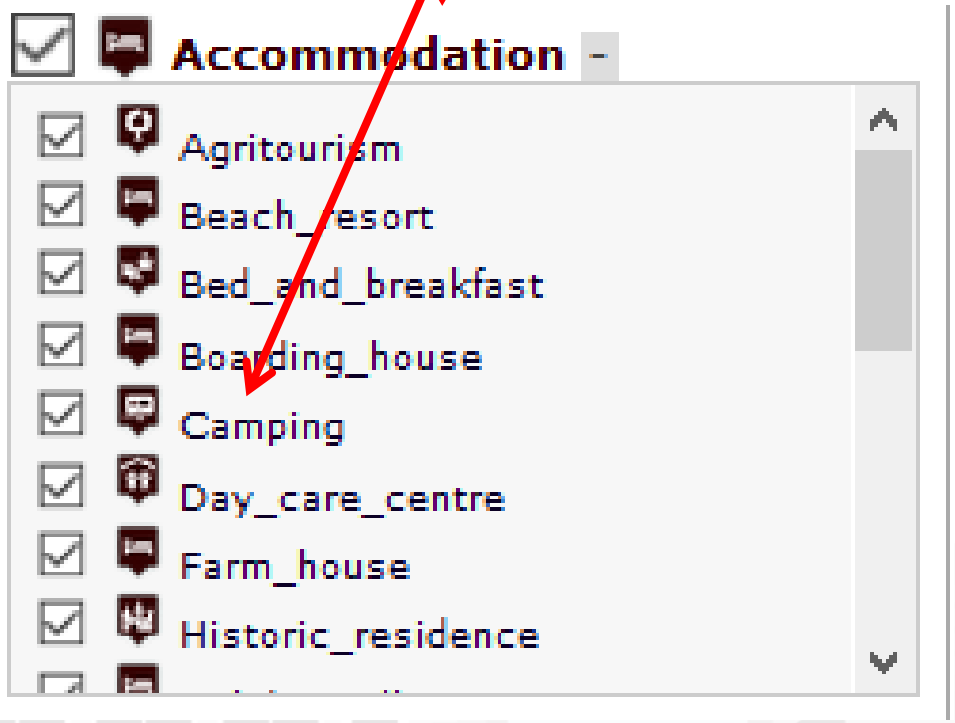
Concepts of Services: Macro and subcathegory

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



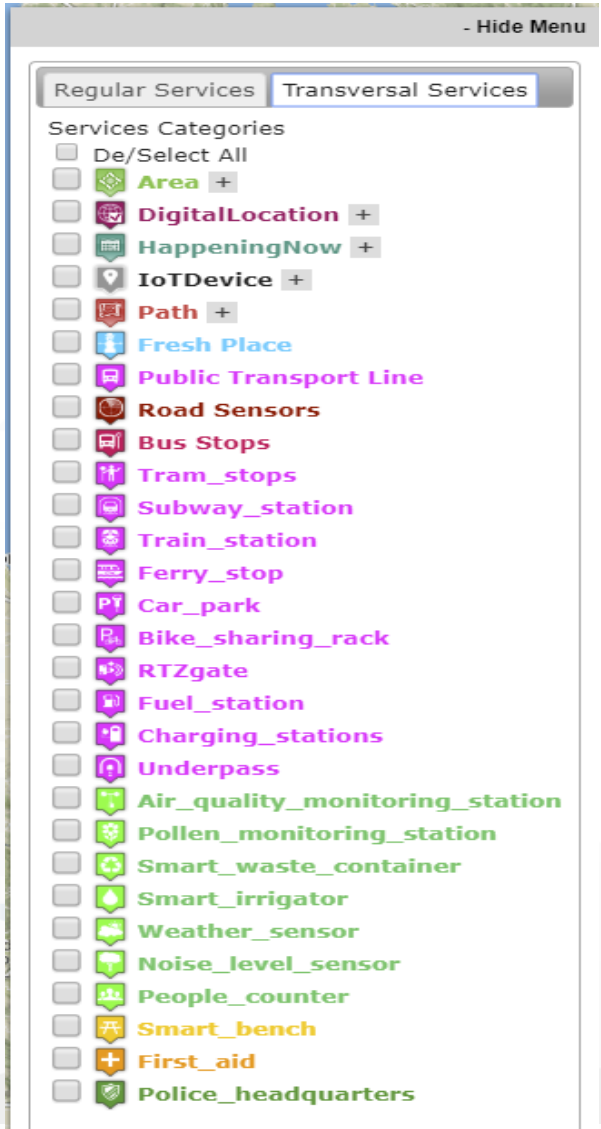
20 Service Macro Classes (The Nature)

Service subClasseses (the SubNature)



Access to Entities

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

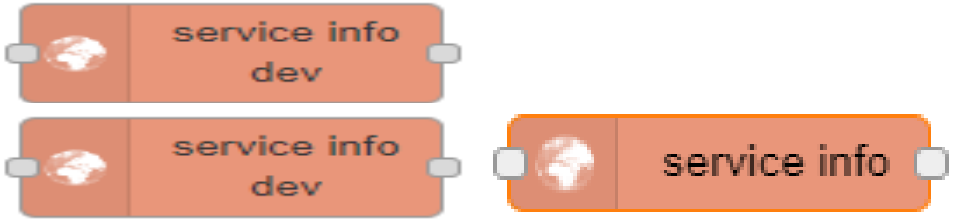


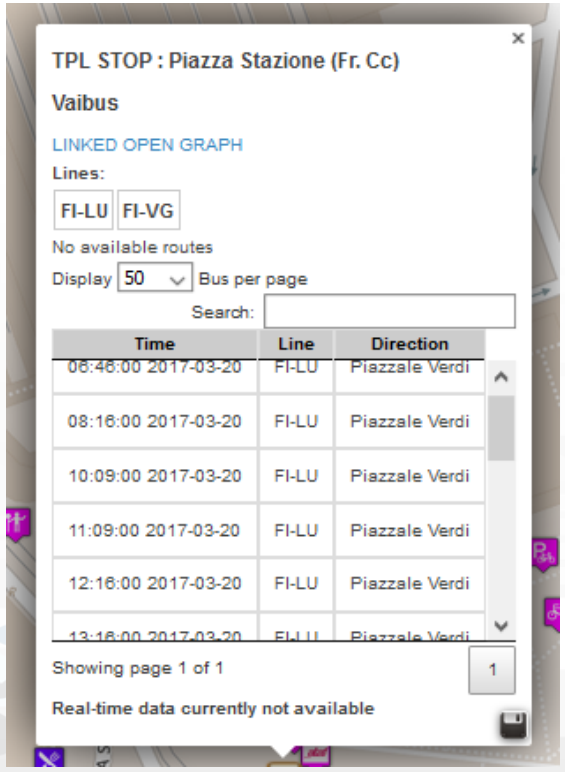
S4CSearch

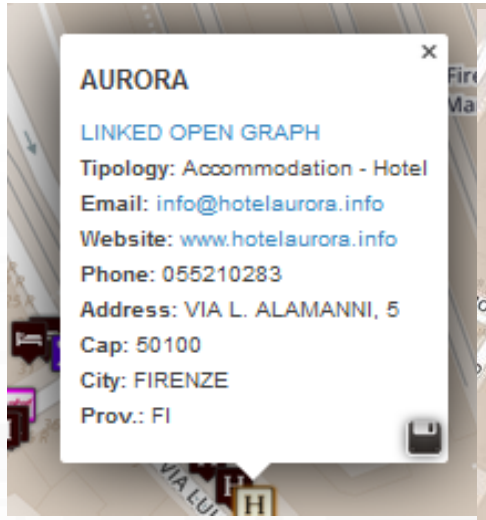
Node-RED

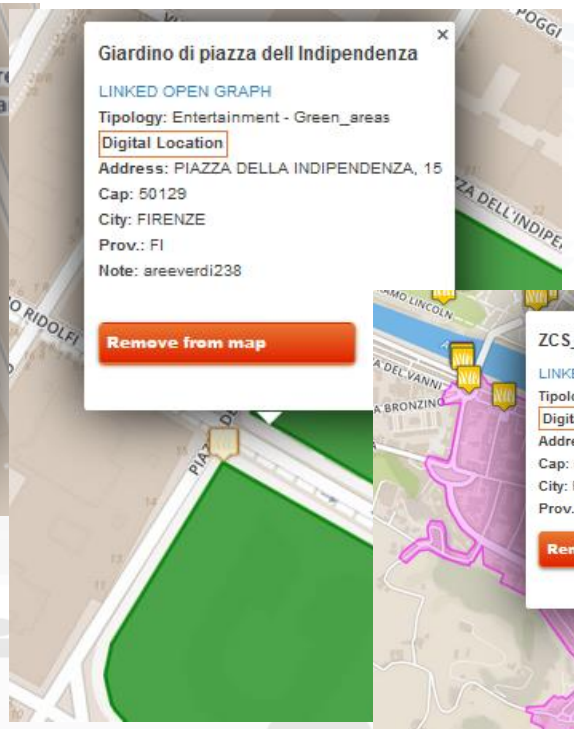
S4CUtility

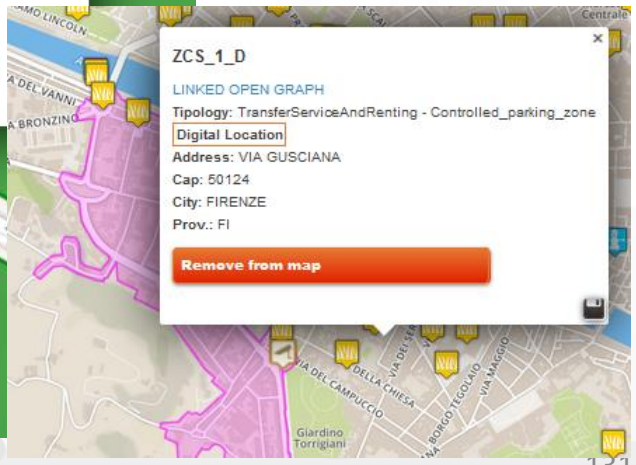
- ANY kind of sensors
- **To Get DATA of a Service / POI /sensor**
  - Historical and real time
  - Real Time







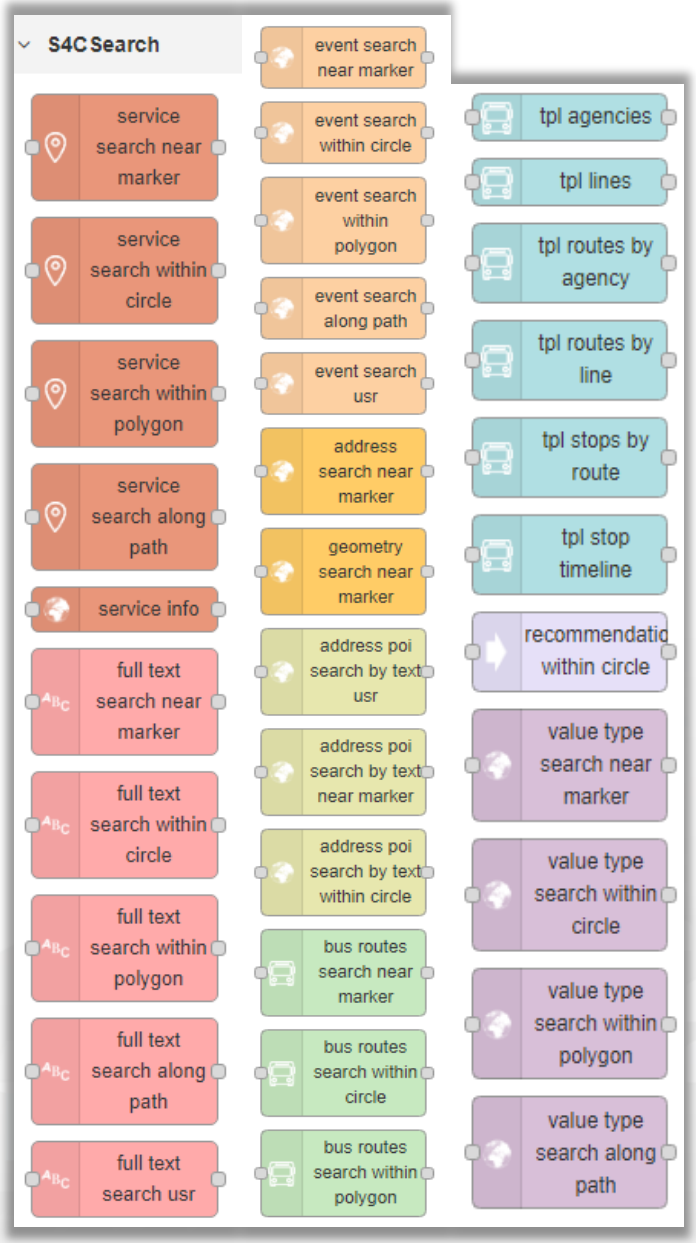




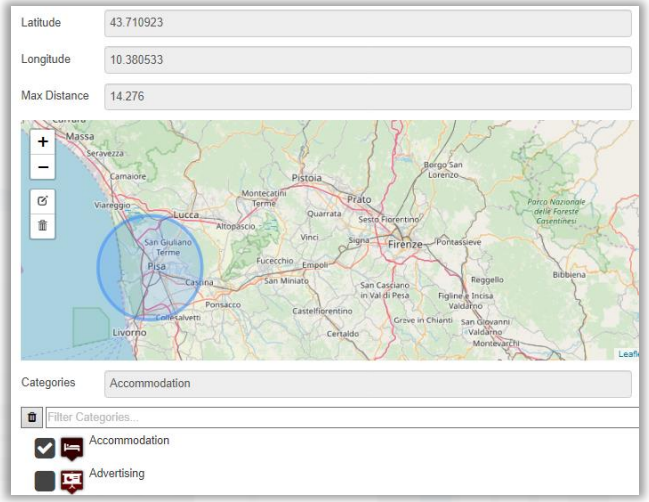
S4CSearch

Smart City Entities Search

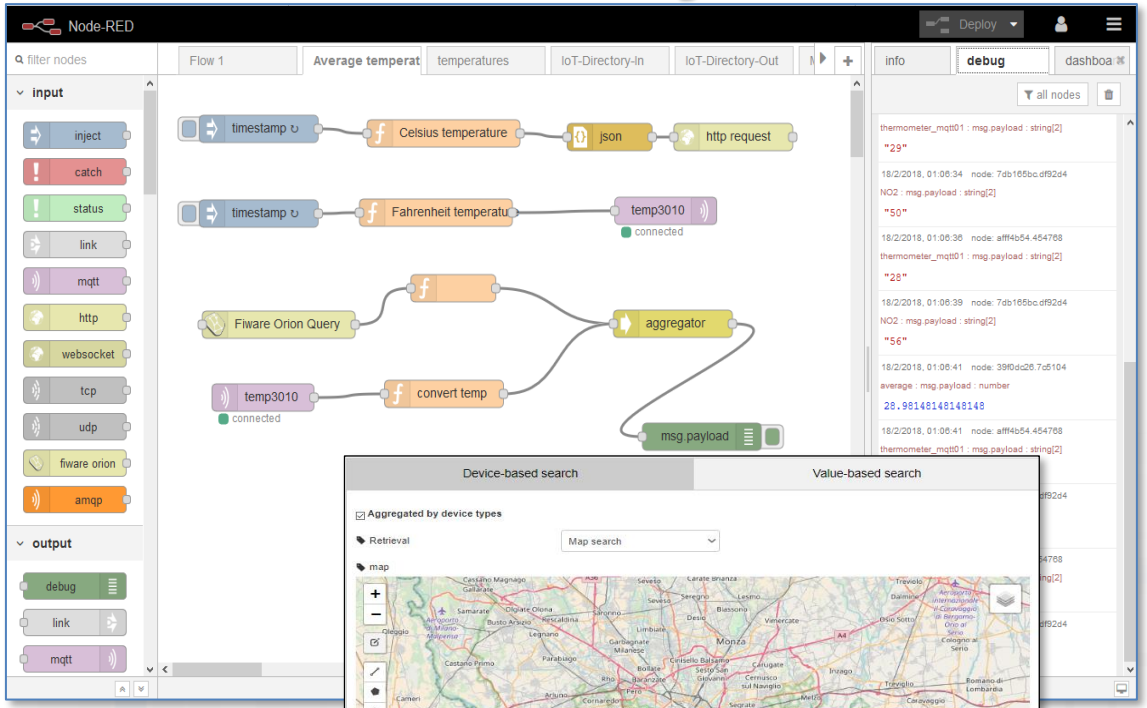
Simple and Fast

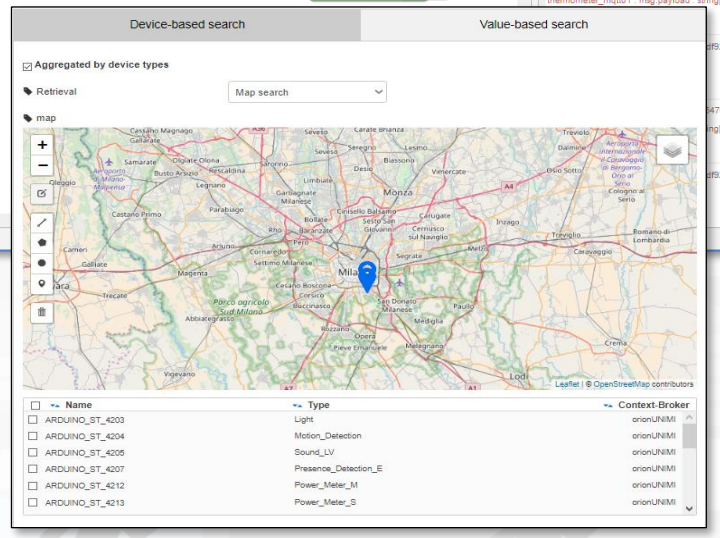


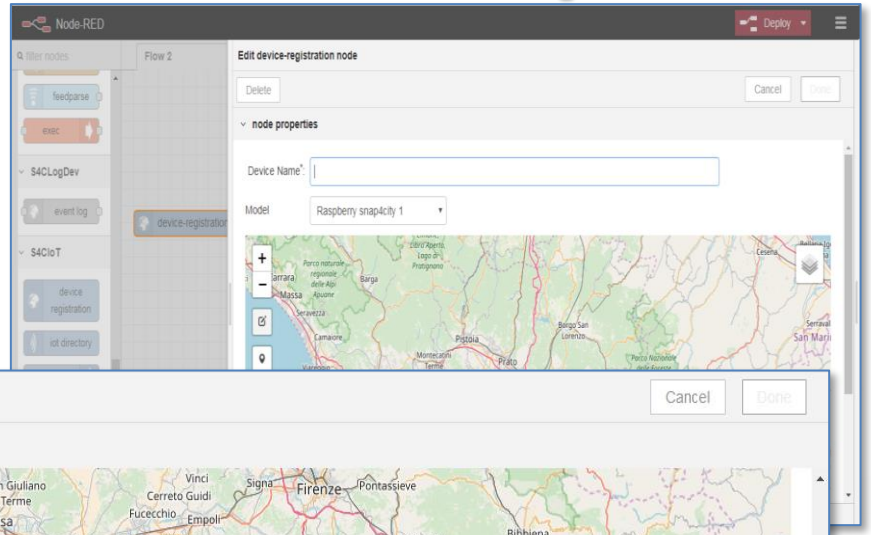
- **For example to search for:**
  - POIs:
    - near a GPS position, from text, along a path, in an area, etc..
  - Public Transport information / data
  - Suggestions
  - Public Transport Means Routes/Paths
  - Events in the area
  - Value Type (kind of data)
  - Etc.
- **To Get DATA of a Service /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

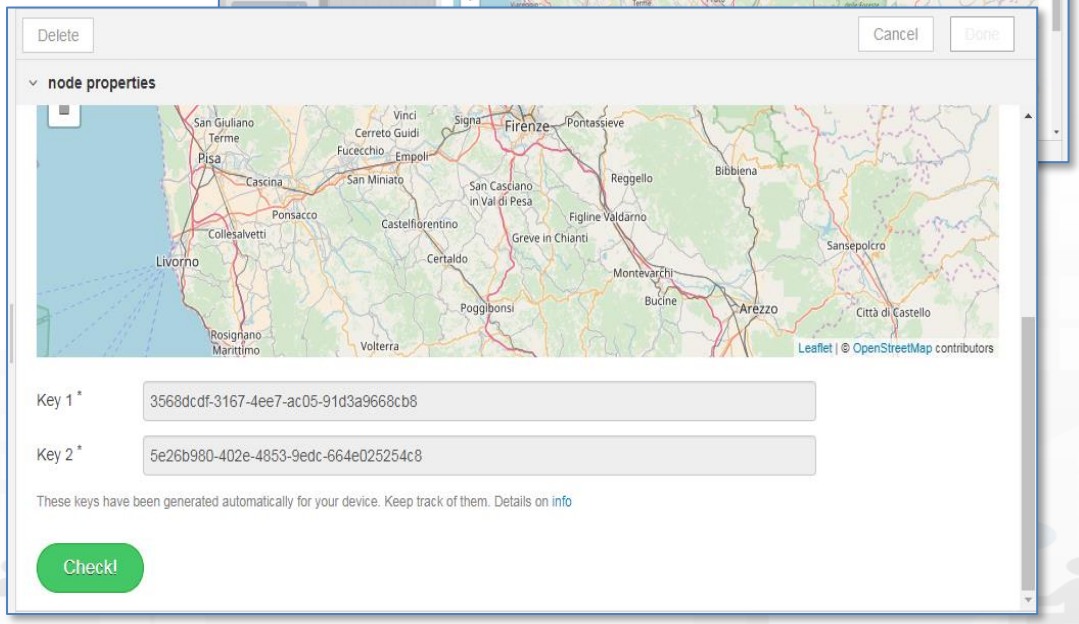


IOT Discovery on IOT Application Development







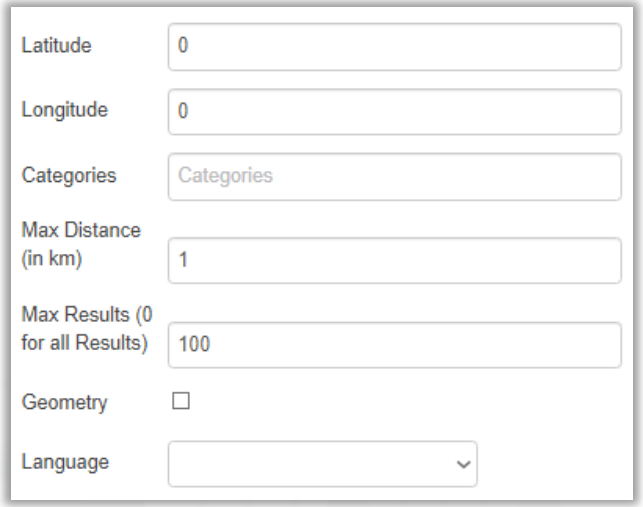


S4CSearch Adv

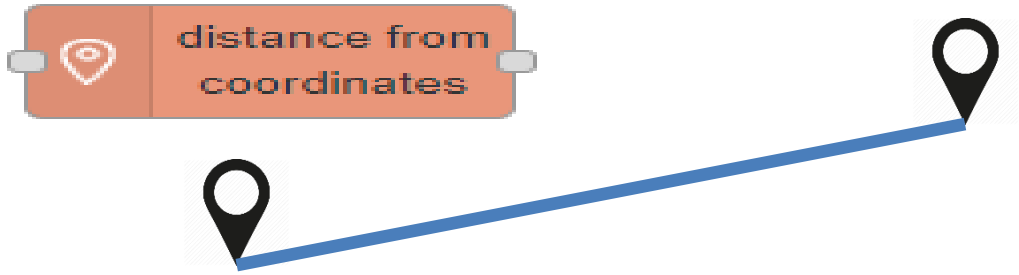
Smart City Entities Advanced Search

Flexiblity

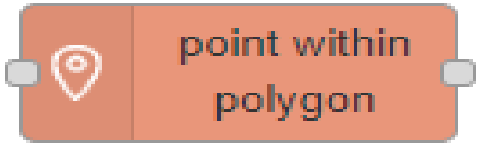
- Similar to basic Search functions but with more flexibility of the function for programming the search
- Adding Dynamic behavior:
  - Getting in input JSON with parameters
- **To Get DATA of a Service / POI /sensor**
  - Historical and real time
  - ANY kind of sensors



- **Distance from GPS point**



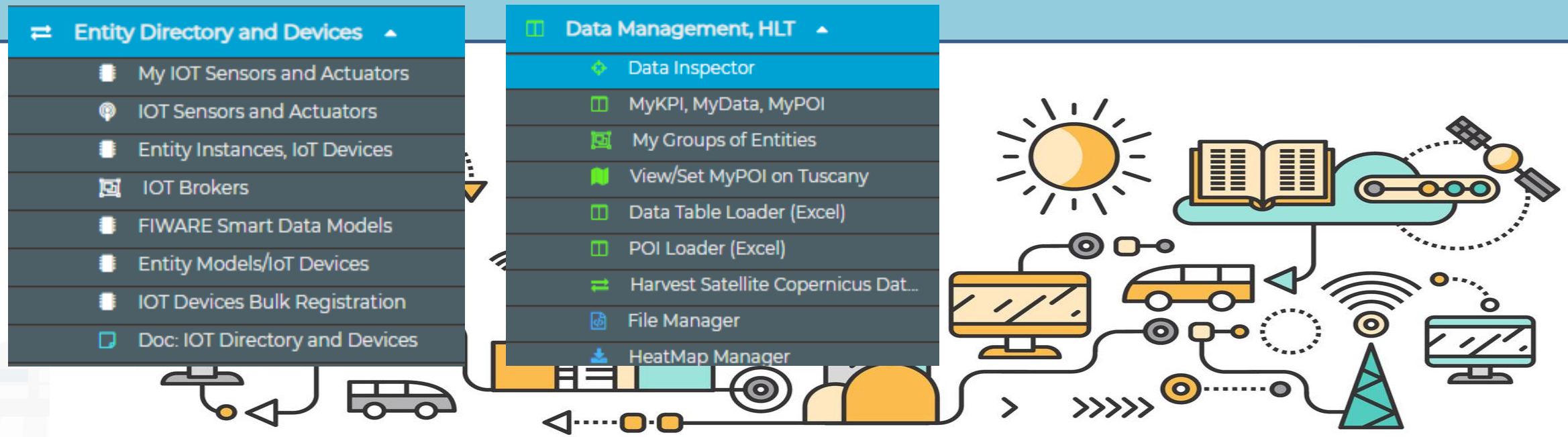
- **Point is in Polygon ?**
  - **Polyline as WKT**



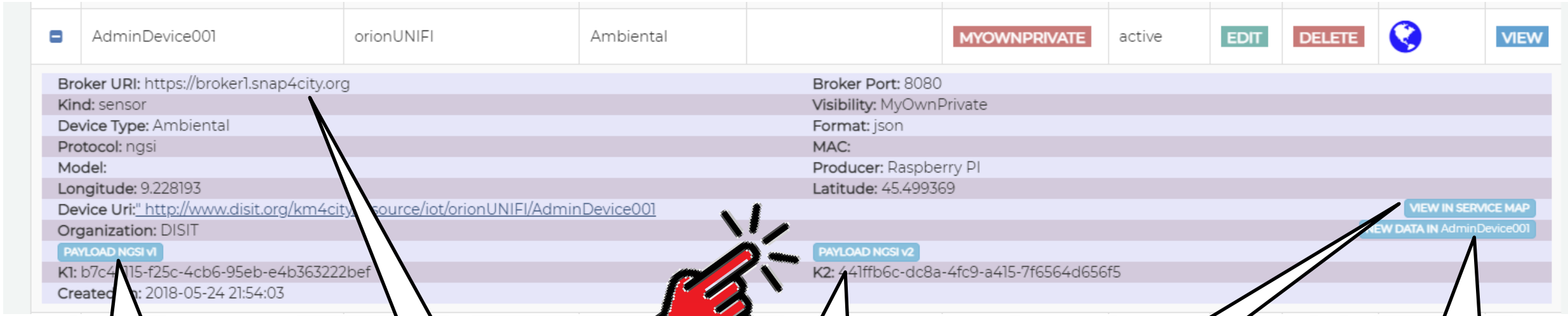
On video

- Example of searching of a IoT device on Service map
  - Identification of the service URI
  - Go on Super Service map, multi Org
- Example on Inspector the same device
  - See the Digital Twin view of the inspector
- Example on use Inject → function → service info dev → debug
  - Retrieve a data, retrieve a set of data in the last 24 hours

search vs services, the ServiceURI



Understanding / Testing an Entity/ IoT Device



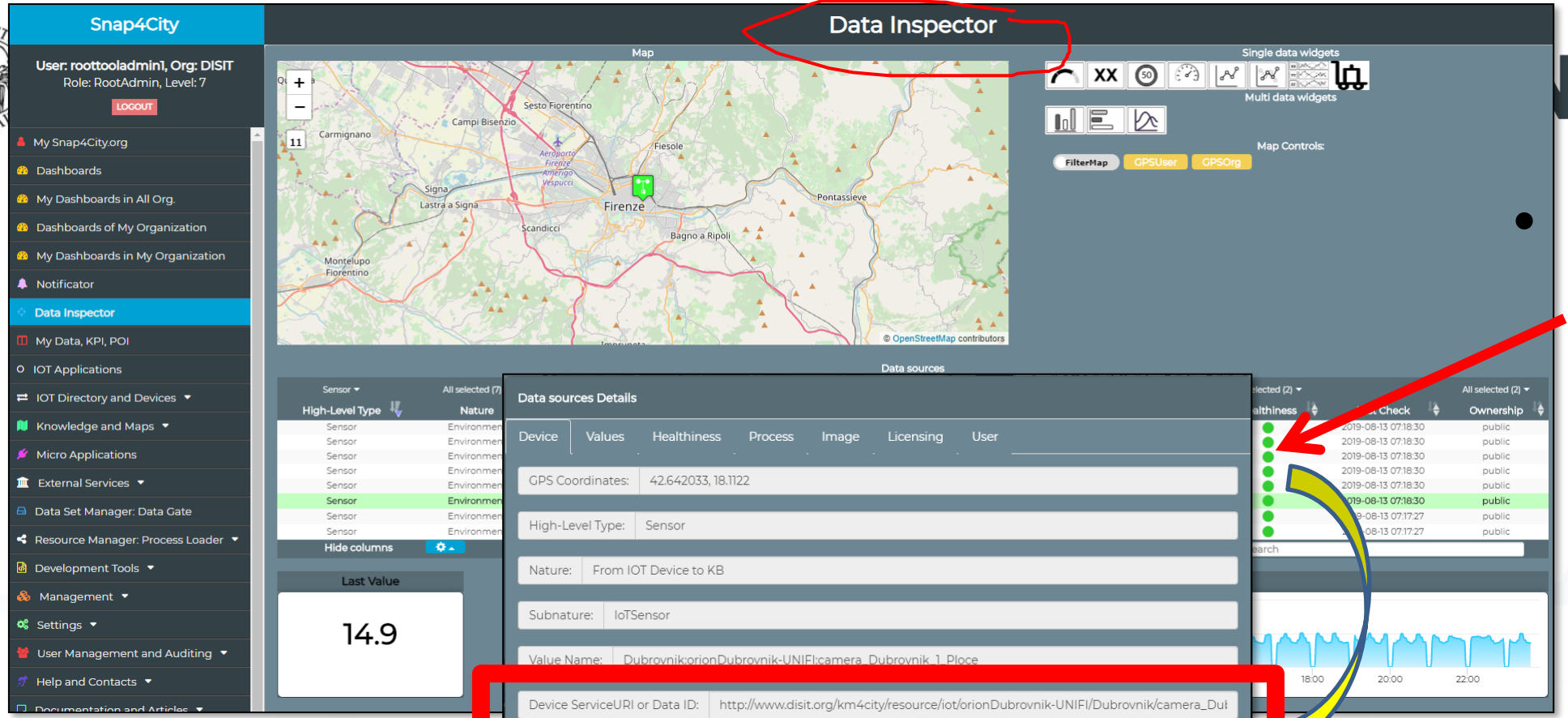
See Payload NGSI V1 in JSON directly from the Broker, Last message of the broker

The Broker

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

See IoT Device on ServiceMap

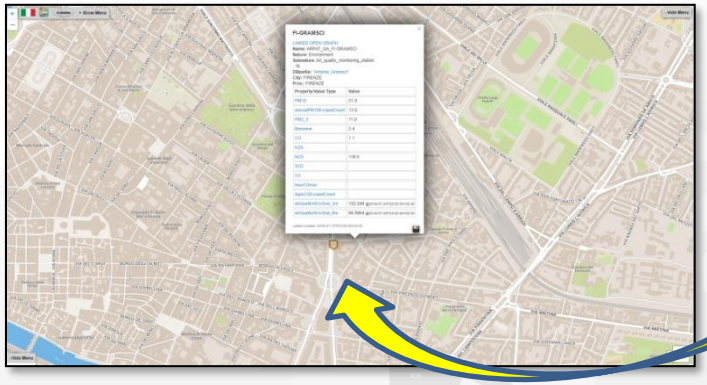
Create a Message to be sent at the IoT broker regarding this device.

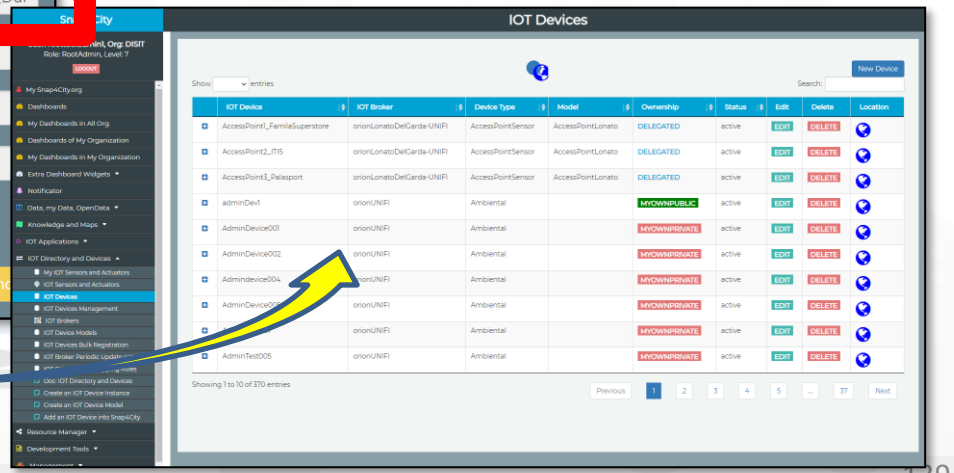


Knowledge Base view

- Click with the mouse on it

Data Inspector Wizard





Some functionalities are limited to certain roles

Notation Terminology

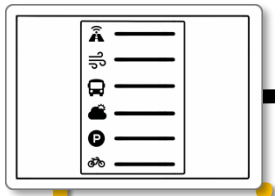
WHERE	Are synonymous at level of service which can be IoT device or entity with data and references to	Are synonymous at level of the single attribute of the entity, device, service, etc.
IoT Directory, Entity Directory	IoT Device, Entity Instance, Device URI	Sensor, Actuator, Attributes, Values (value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

ServiceURI, SURI of a sensor device:

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

ServiceURI, SURI extended with attribute/variable/value:

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

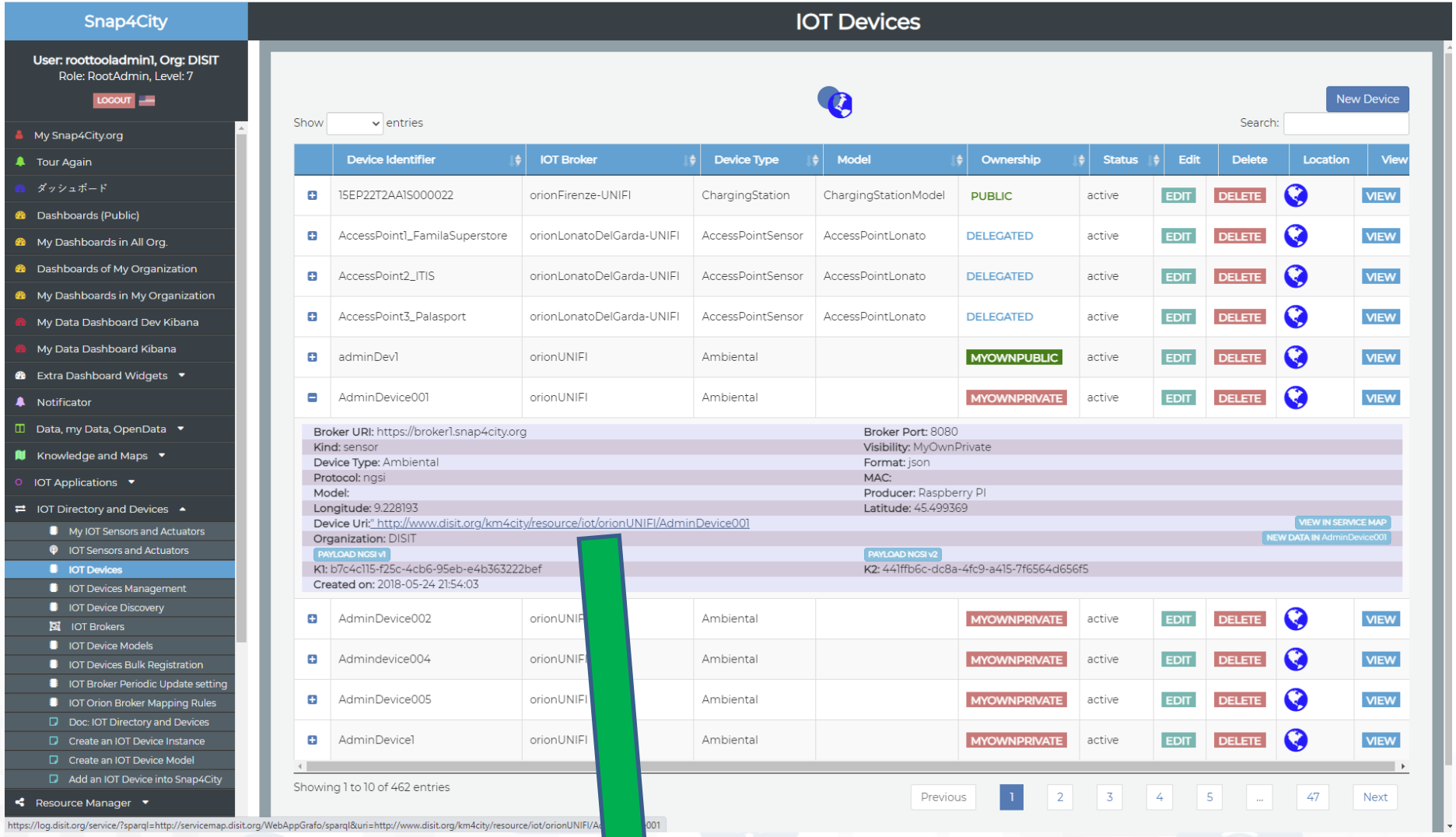



Dashboard Usage and recipe: Event map target

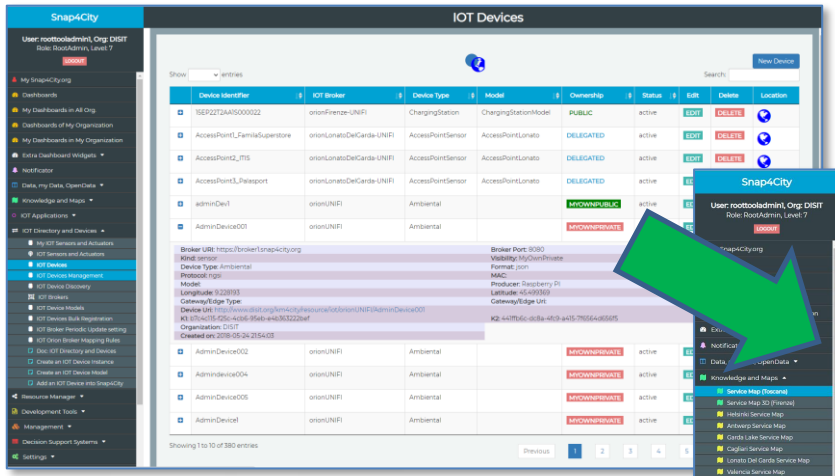
- **Selector to Show on Map a**
  - **category of Map positioned elements**
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.08694333811321;8.791809082031252;44.93758500391093;14.065246582031252&categories=Traffic_sensor&maxResults=0&maxDists=0.1&text=&model=&value_type=&format=json
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?queryId=e5f39066cd68ffe259ed8877bcee222b&format=json
  - **Entity by Model**
    - https://www.disit.org/superservicemap/api/v1?selection=59.36535064975547;13.457822799682619;59.39031474260852;13.566999435424806&model=SmartLightCapelon&format=json
  - **Single Entity**
    - https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/iot/orionFirenze2/Firenze/SHT20lab_new&format=json&fromTime=3-day
  - **Heatmap among many**
    - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=Florence_PM10
  - **Traffic flow**
    - https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=FirenzeFIPILITrafficRealtime&trafficflowmanager=true
    - https://firenzetraffic.km4city.org/trafficRTDetails/roads/read.php
  - **Origin Destination Map**
    - https://odmm.snap4city.org/api/get?precision=communes&from_date=&organization=Toscana&inflow=True&longitude=11.255751&latitude=43.769710&od_id=mobile_Toscana_1000&perc=True
- **Events which are also PIN on map**
- **Il Service URI as the unique identifier of the Entity**
  - **http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO632**

IoT Directory

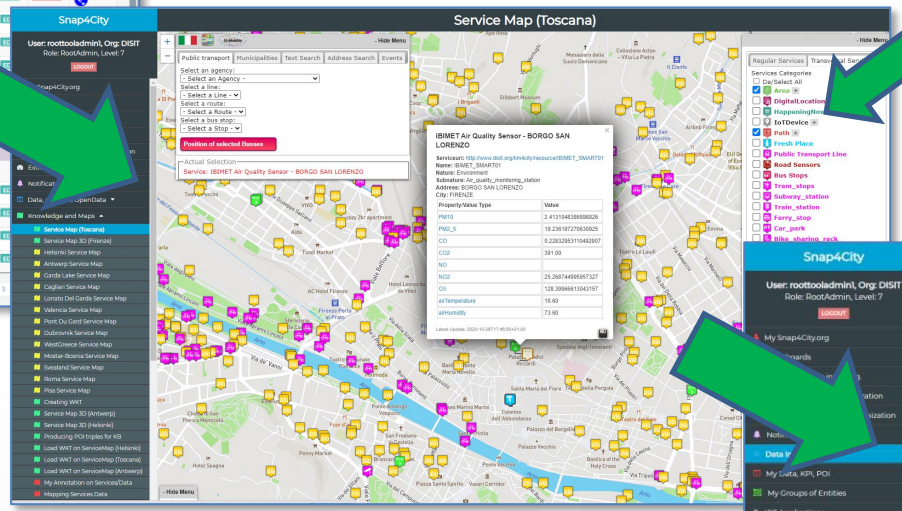
- **For:**
  IOT Devices, Sensors, Sensor mobile, Actuators, Virtual Sensors, etc.
- Accessible as
  - ServiceURI
  - **Device URI**



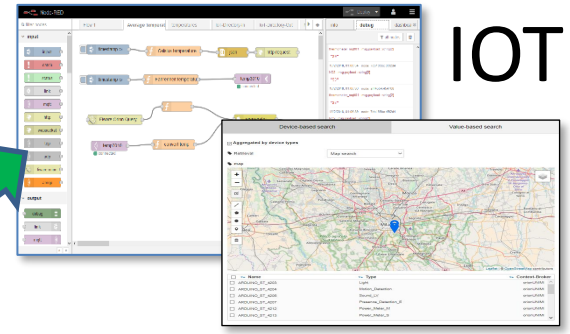
Data Registration Flow at a Glance



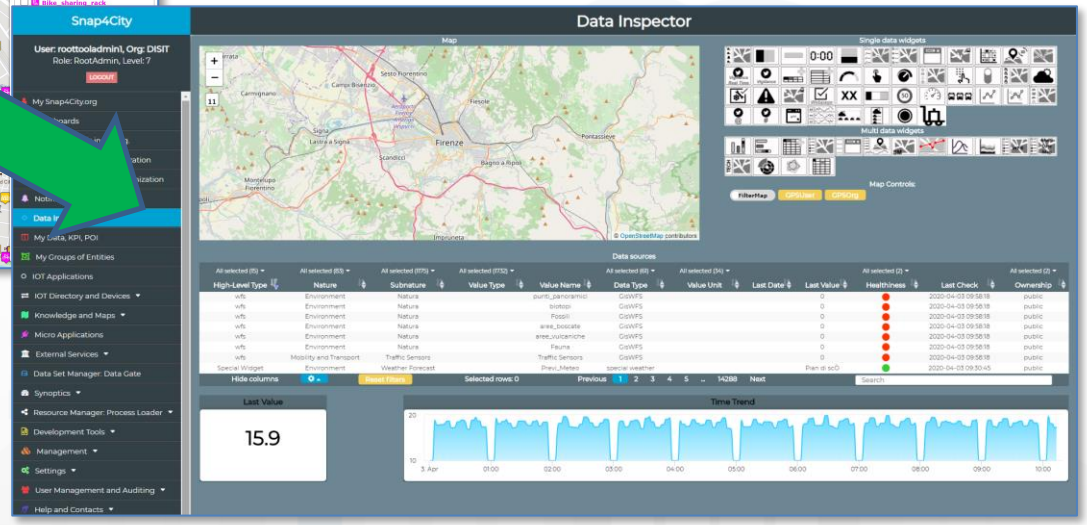
IOT Directory:
Devices…
Sensors..
Actuators…



Knowledge Base,
ServiceMap,
SuperServiceMap
SmartCity API,
ASCAPI



IOT Apps

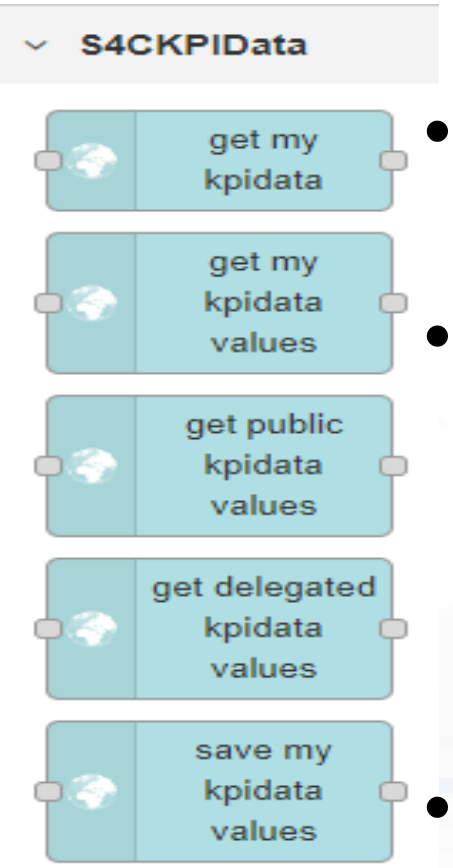


DataInspector
Dashboard Wizard

MyKPI Nodes


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

S4C KPI Data



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

On video

- Opening of the MyKPI
- Editing a MyKPI

SCALABLE SMART ANALYTIC APPLICATION BUILDER FOR SENTIENT CITIES

TOP

FORGING & MANAGING OPEN AND FLEXIBLE WEB

FROM CITY DASHBOARD TO APPLICATIONS

DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT

TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS

SNAP4CITY AND KM4CITY PROJECTS

HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP

DECISION SUPPORT SYSTEM AND CITY RESILIENCE

SNAP4CITY THE VIEW OF THE ADMINISTRATORS

THE LOGIC AND THE SMARTNESS

SNAP4CITY API

SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK

External Service ←→

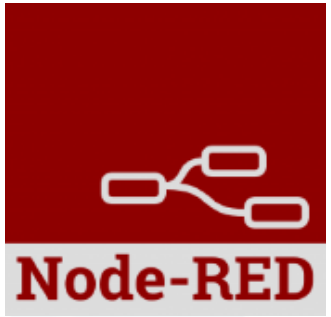
IoT App / Proc.Logic

- Processing Logics / IOT App
  - Processing Logics / IOT App
  - MicroServices for Proc.Logic/IoT App
  - MicroServices from DataAnalytic
  - IOT MicroServices for Final Users
  - IOT MicroServices for Developers
  - DOC: Processing Logic/IOT App
  - How to Develop Proc.Logic / IoT App
  - Create A MicroService from RestCall

Node-RED

IoT App / Proc.Logic

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

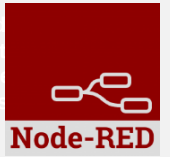


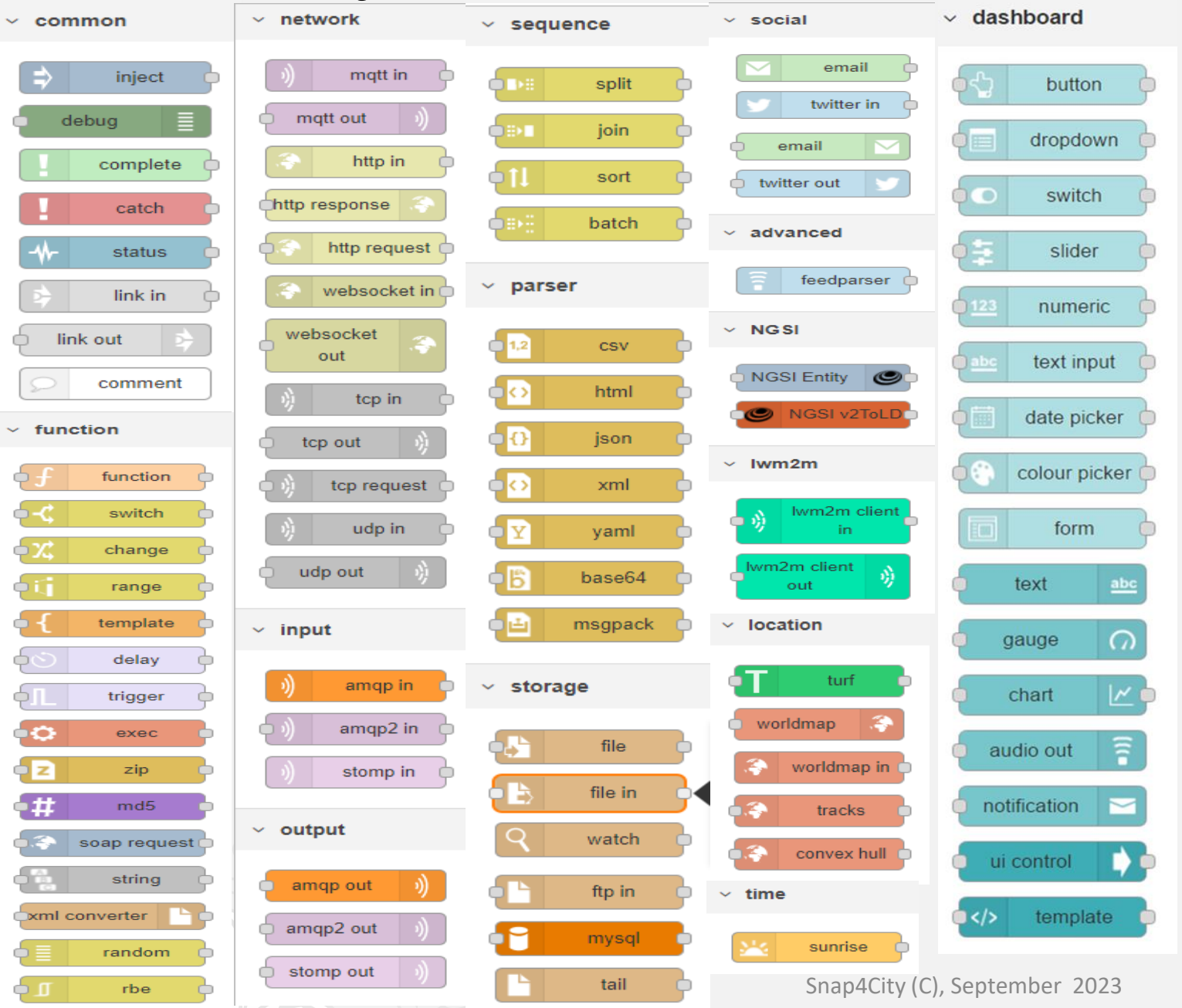
- Data Analytics ←→ IoT App / Proc.Logic
- Broker → Storage
- IoT App / Proc.Logic → Broker
- Broker → IoT App / Proc.Logic
- IoT App / Proc.Logic → Storage

Part 4

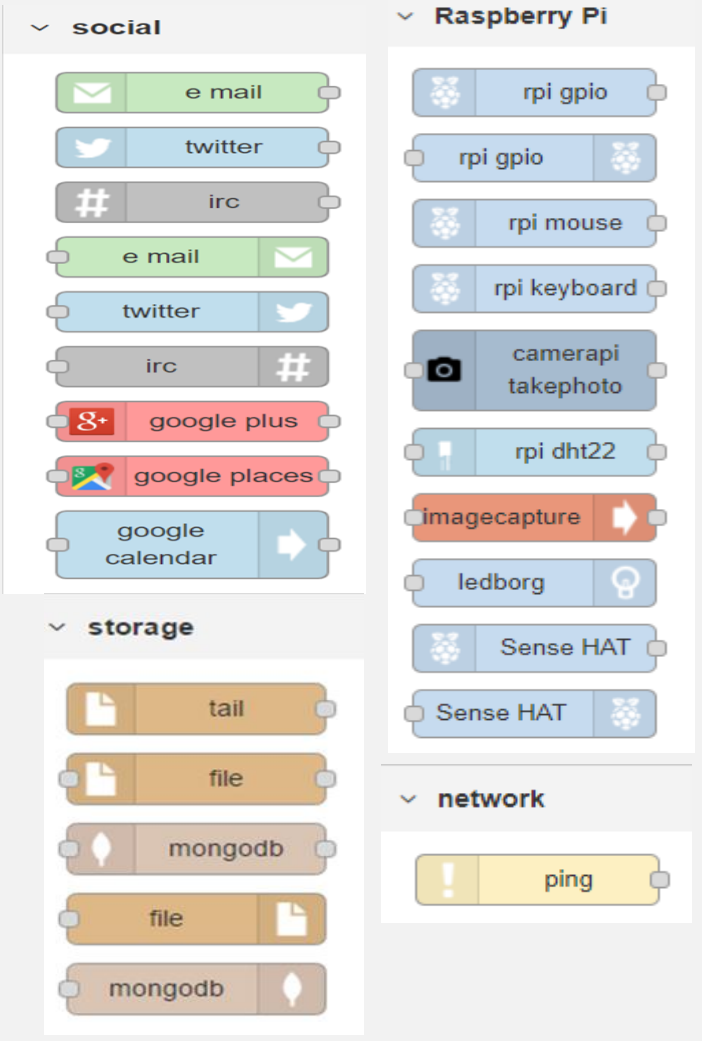
Part 5

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

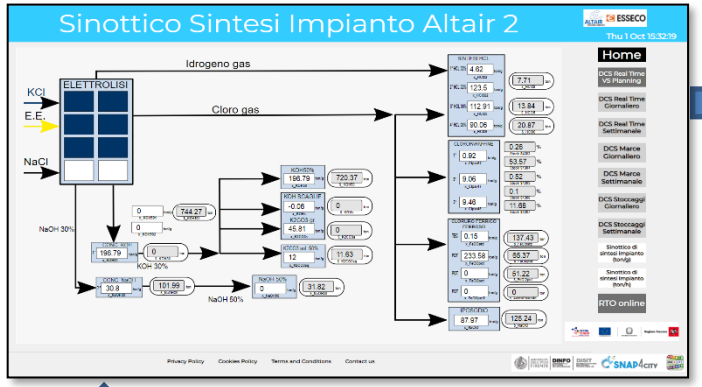




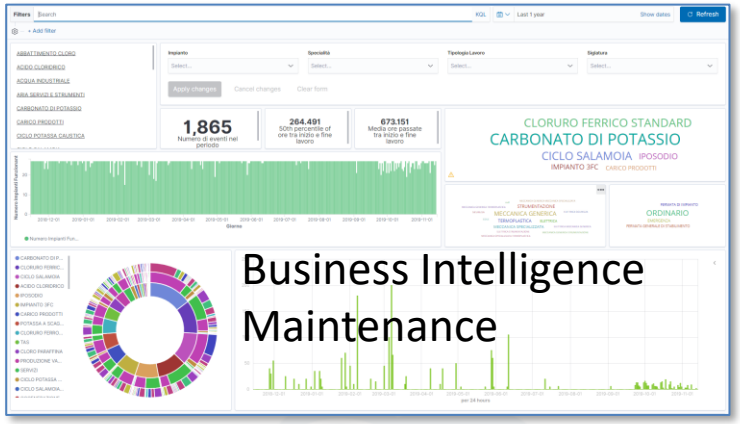
+ on IOT Edge Raspberry



Example of Integrated workflow



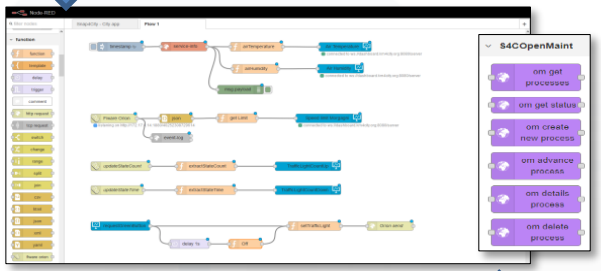
Consumptions/productions

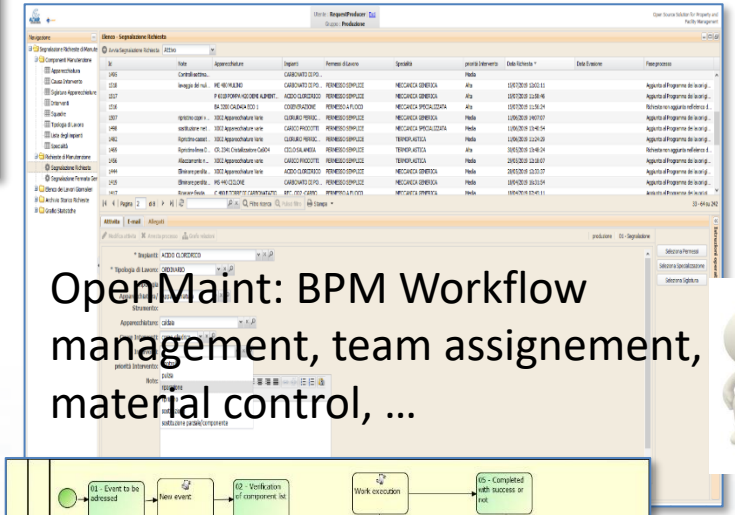


Events/actions

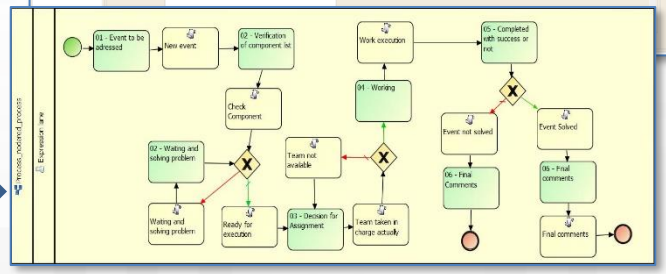
Business Intelligence
Maintenance

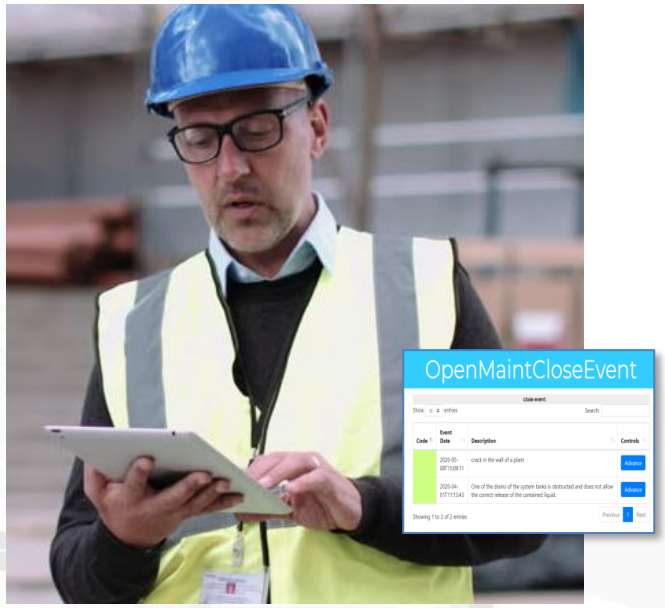
Dashboards and actions





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



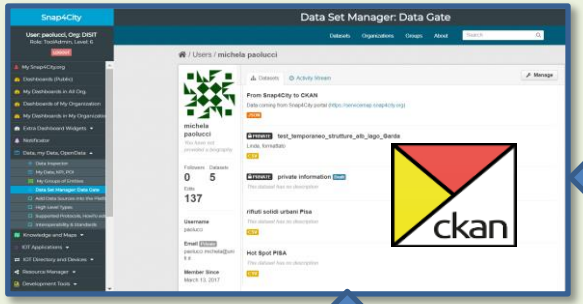


IOT App, Data event firing, event detection and firing Critical event management

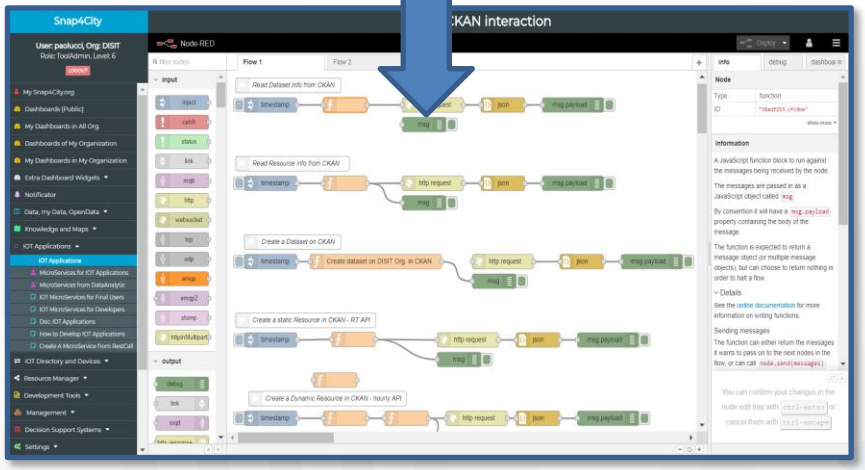
Snap4City vs CKAN

Snap4City Portal and Integrated tools

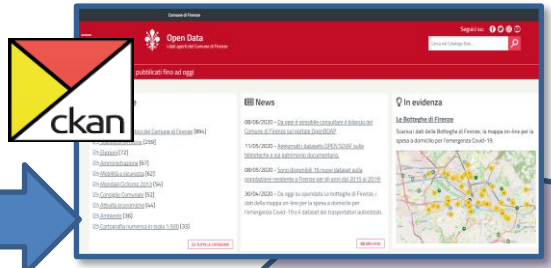
Datagate

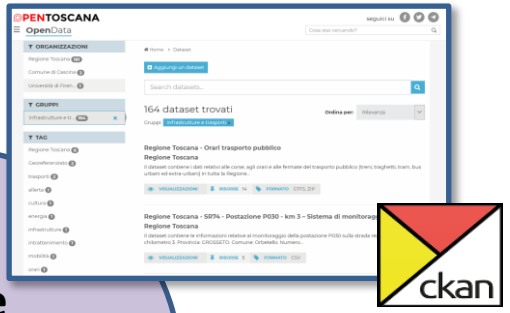


Advanced Snap4City APIs and Micro Services

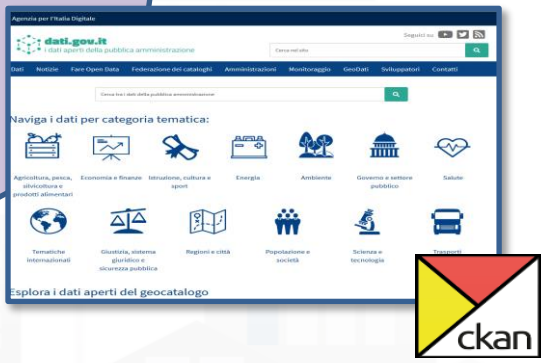


Harvesting and Publishing





Open or Private External CKAN Data Portals

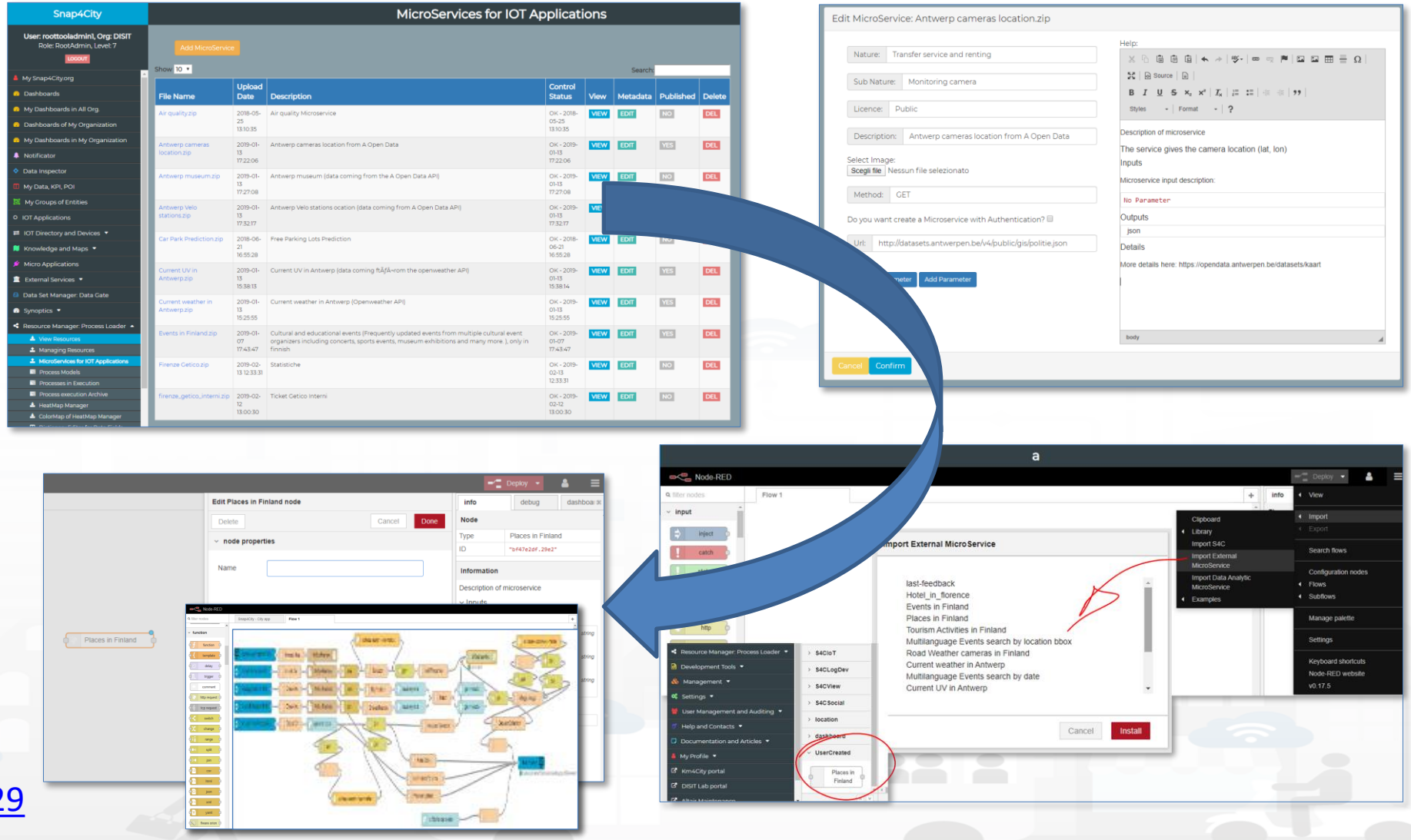


Automatize:

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

External REST Call API vs MicroServices

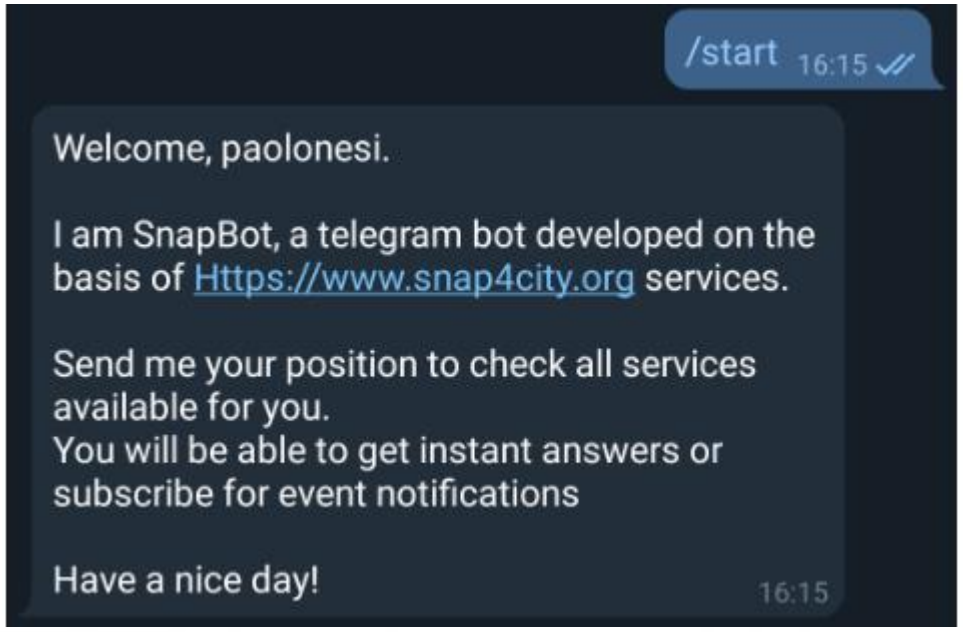
- Each Rest Call API can be automaticaly transformed into e MicroService for the IOT Applications

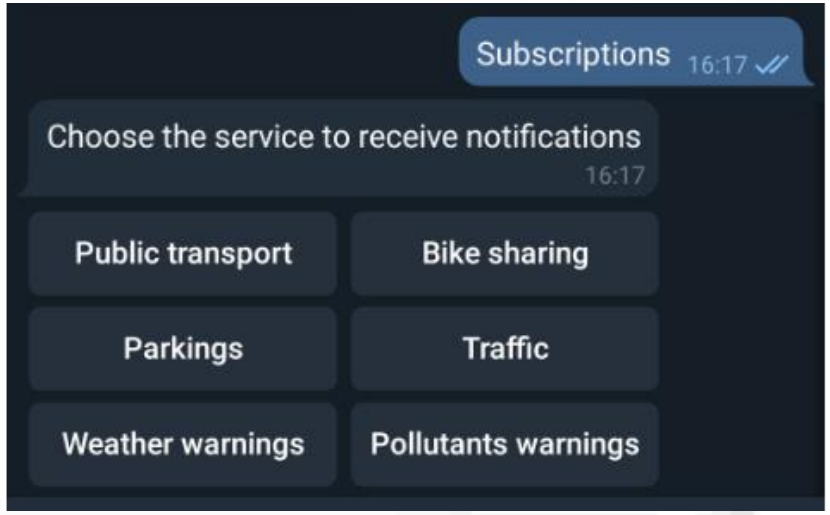


https://www.snap4city.org/129

SnapBot

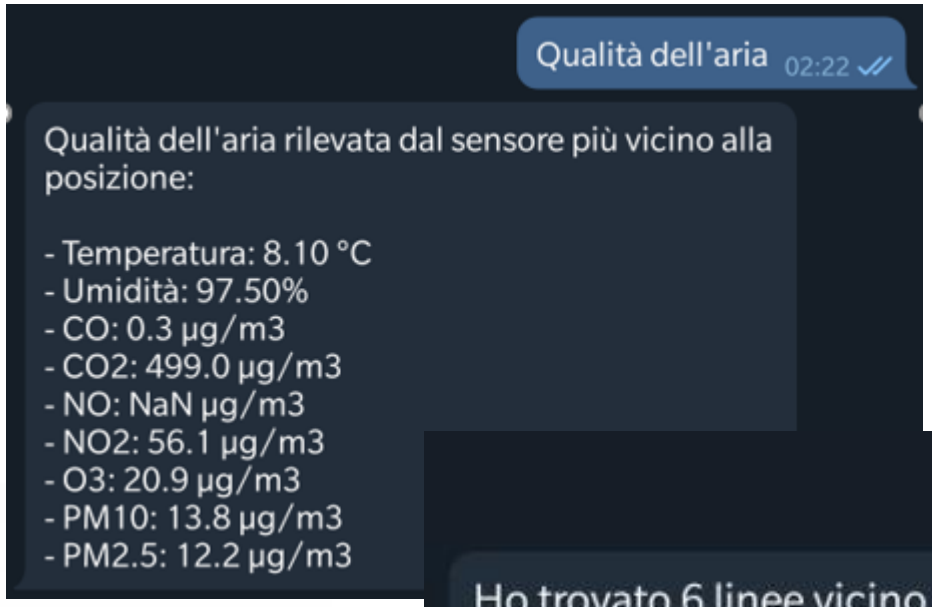
- provides real time smart city services to Telegram users, geolocalized, when you like, what you like
- active on Tuscany in all provinces and cities according to the data accessible on Https://www.snap4city.org
- Services on
  - Public Transport (more than 10 different operators),
  - bike sharing,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

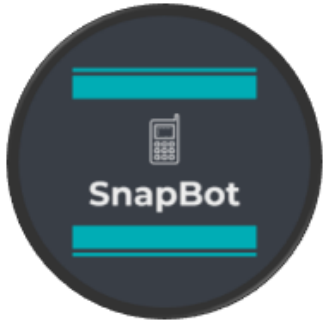


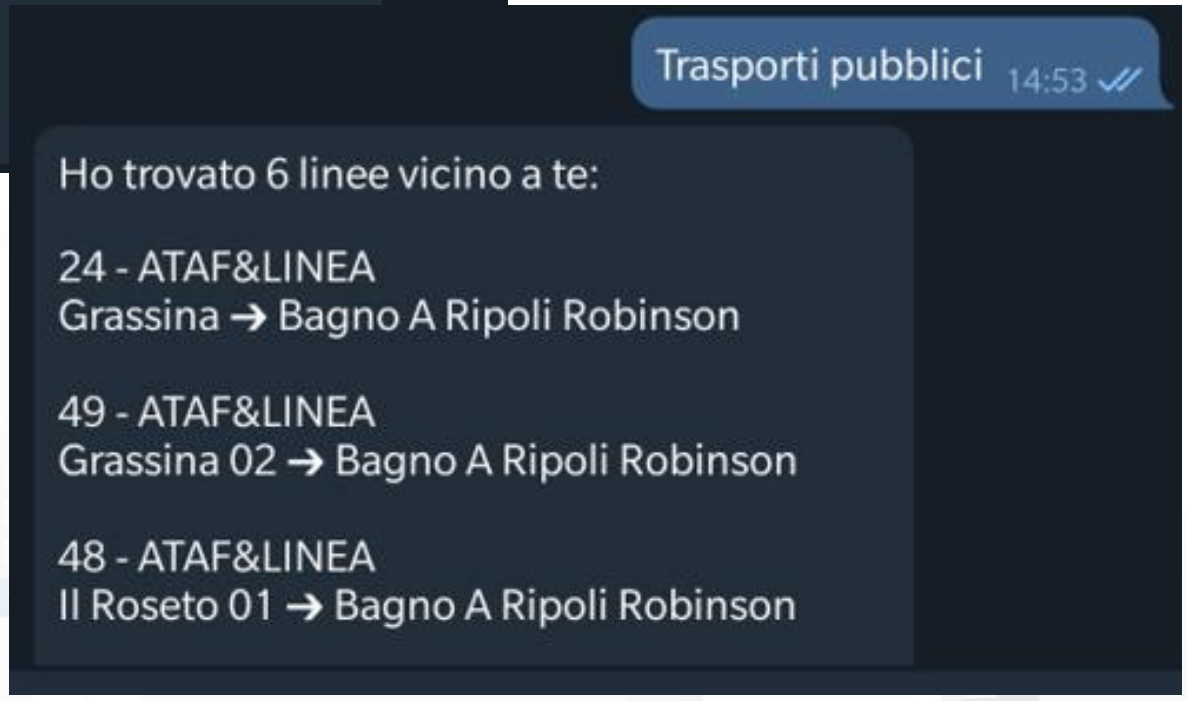


SnapBot

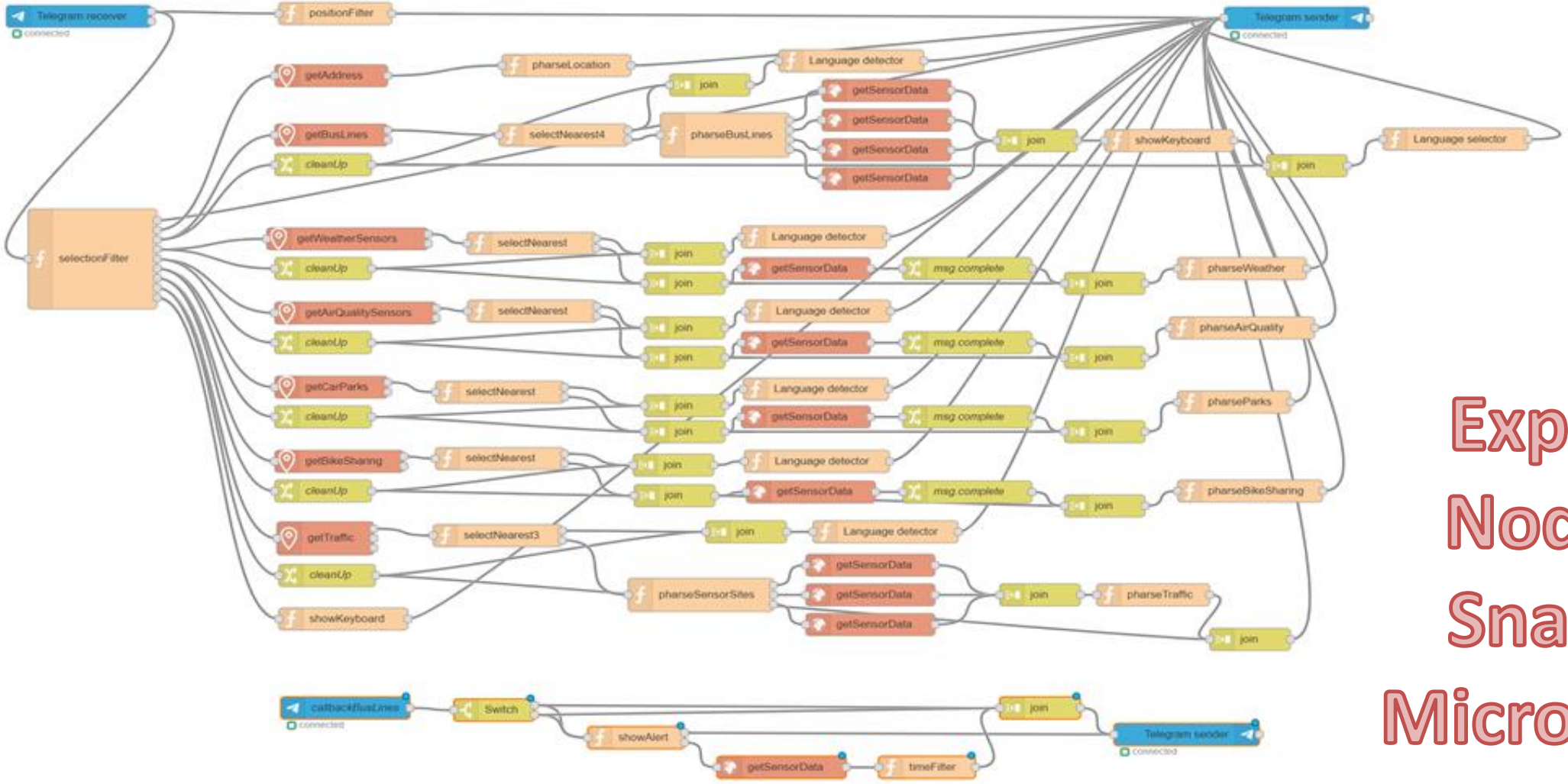
SNAP4CITY KM4CITY







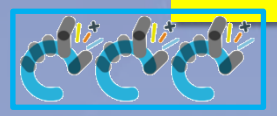
IOT App of SnapBot: OneShot Services



Exploiting Node-RED Snap4City MicroServices

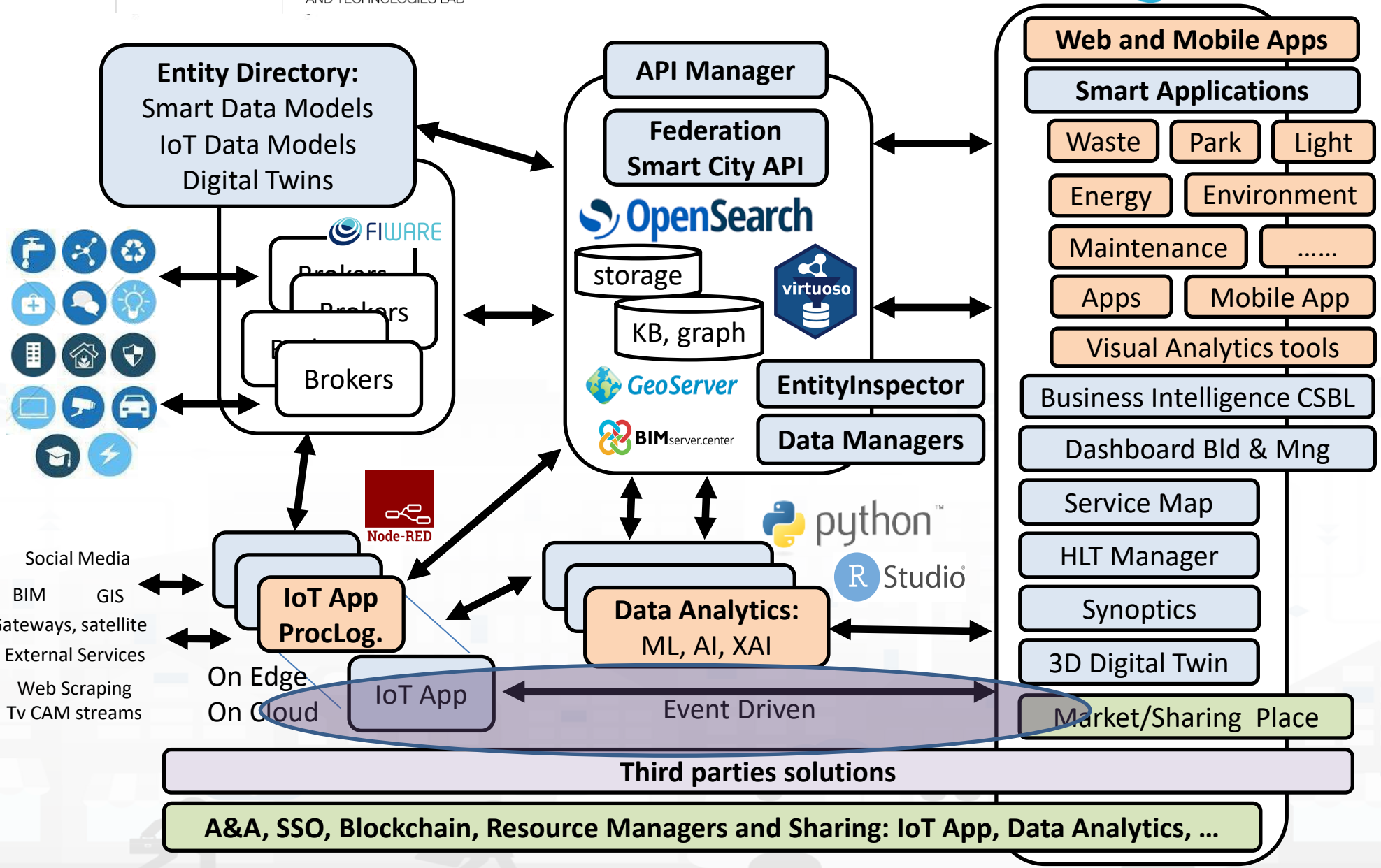
TOP

Dashboards ←→ IoT App / Proc.Logic
Server Side Business Logic

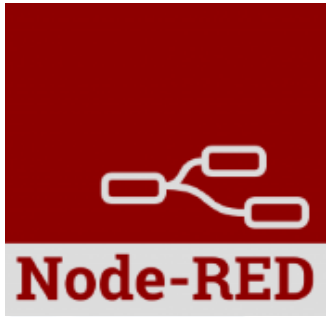
- Processing Logics / IOT App
  - Processing Logics / IOT App
  - MicroServices for Proc.Logic/IoT App
  - MicroServices from DataAnalytic
  - IOT MicroServices for Final Users
  - IOT MicroServices for Developers
  - DOC: Processing Logic/IOT App
  - How to Develop Proc.Logic / IoT App
  - Create A MicroService from RestCall

Tech Arch


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

IoT App / Proc.Logic

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

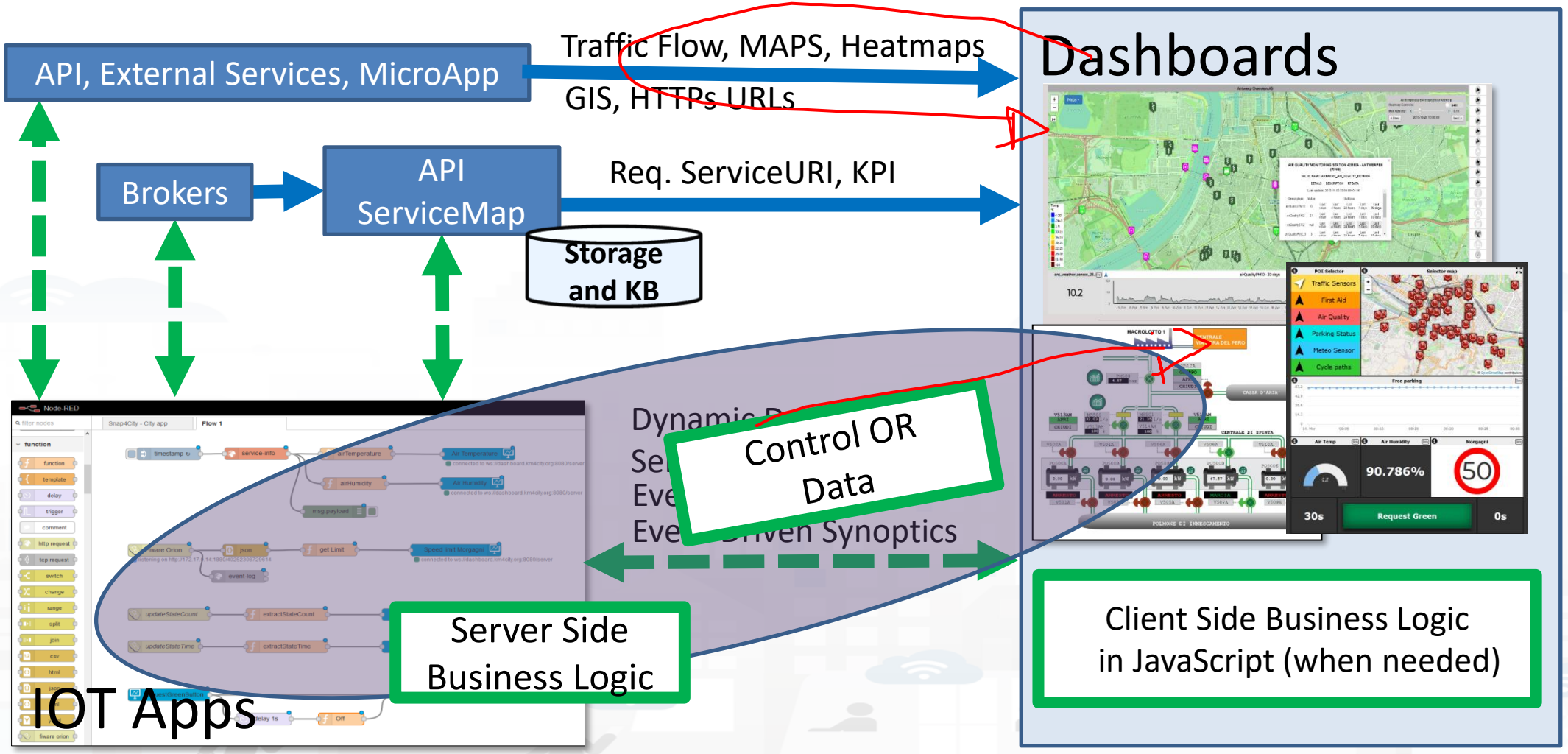


- Data Analytics ←→ IoT App / Proc.Logic
- Broker → Storage
- IoT App / Proc.Logic → Broker
- Broker → IoT App / Proc.Logic
- IoT App / Proc.Logic → Storage

Part 4

Part 5

How the Dashboards exchange data

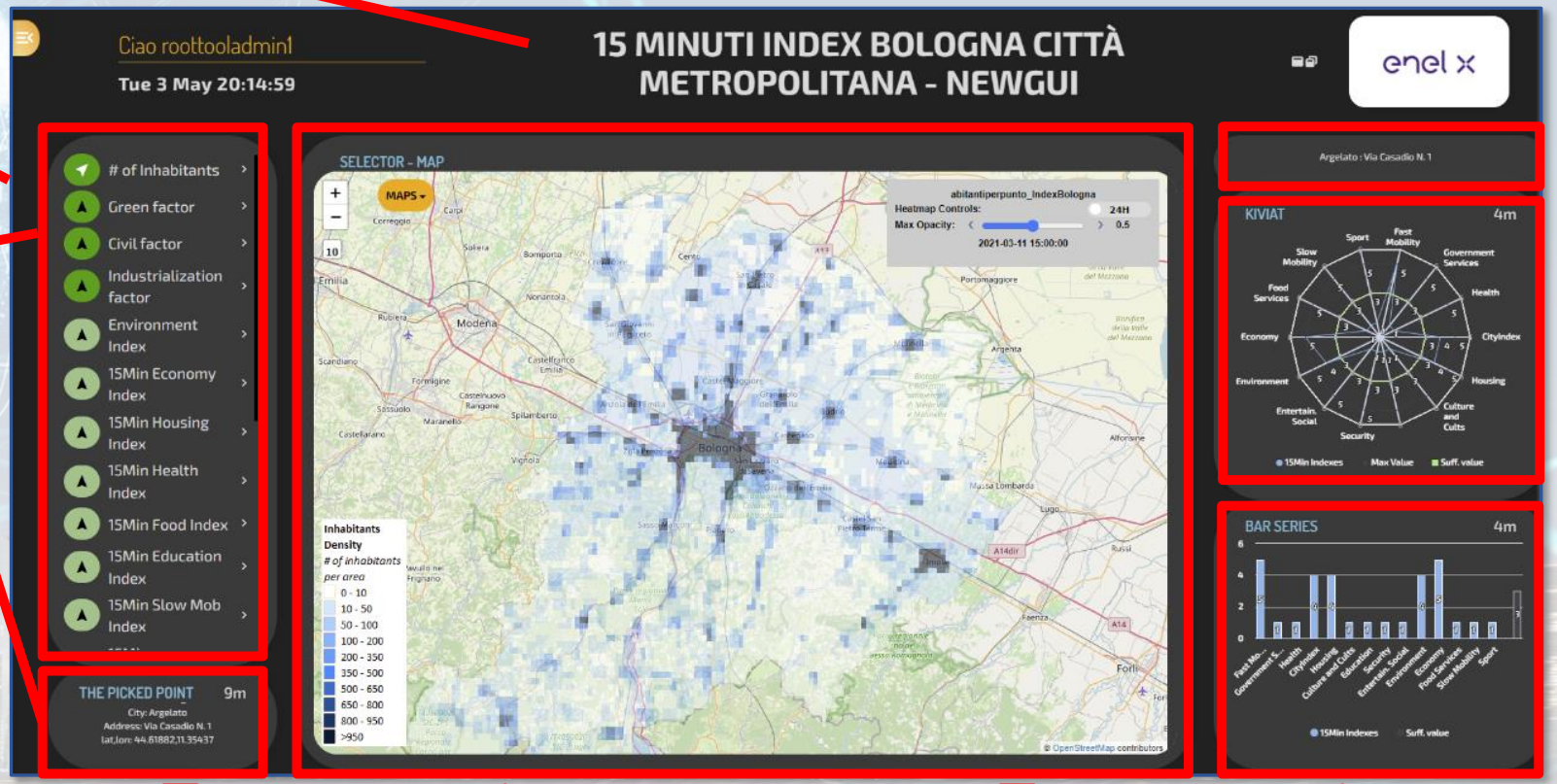


Snap4City Dashboards main concepts

Header

Dashboard

Interactive Widgets



Server Communication

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

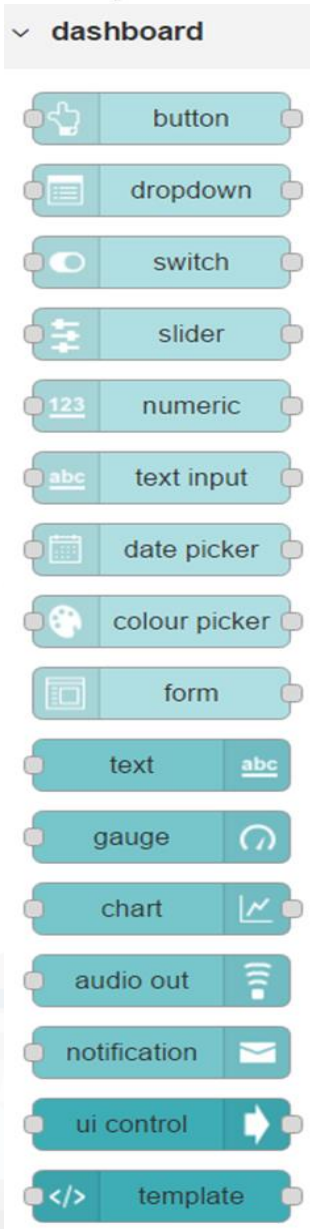
Inter Widget Communication:

Client Side Business Logic

See part 8 of the Course

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

Dashboard Native Local or Snap4City

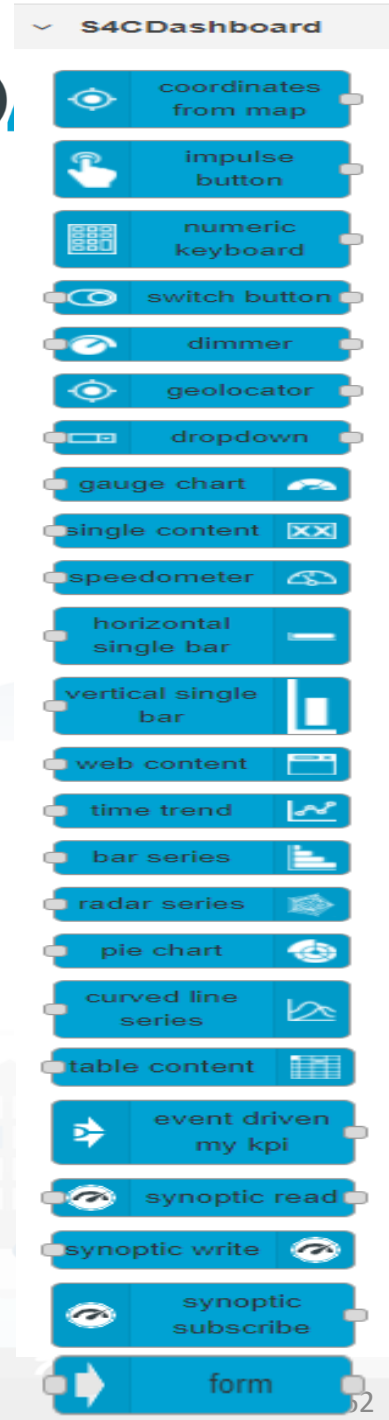


- Input/output
- non secure
- Limited in graphics
- No authentication
- No HLT
- No integration
- No historical data
- No Synoptics
- Etc..

- Local on IOT Edge

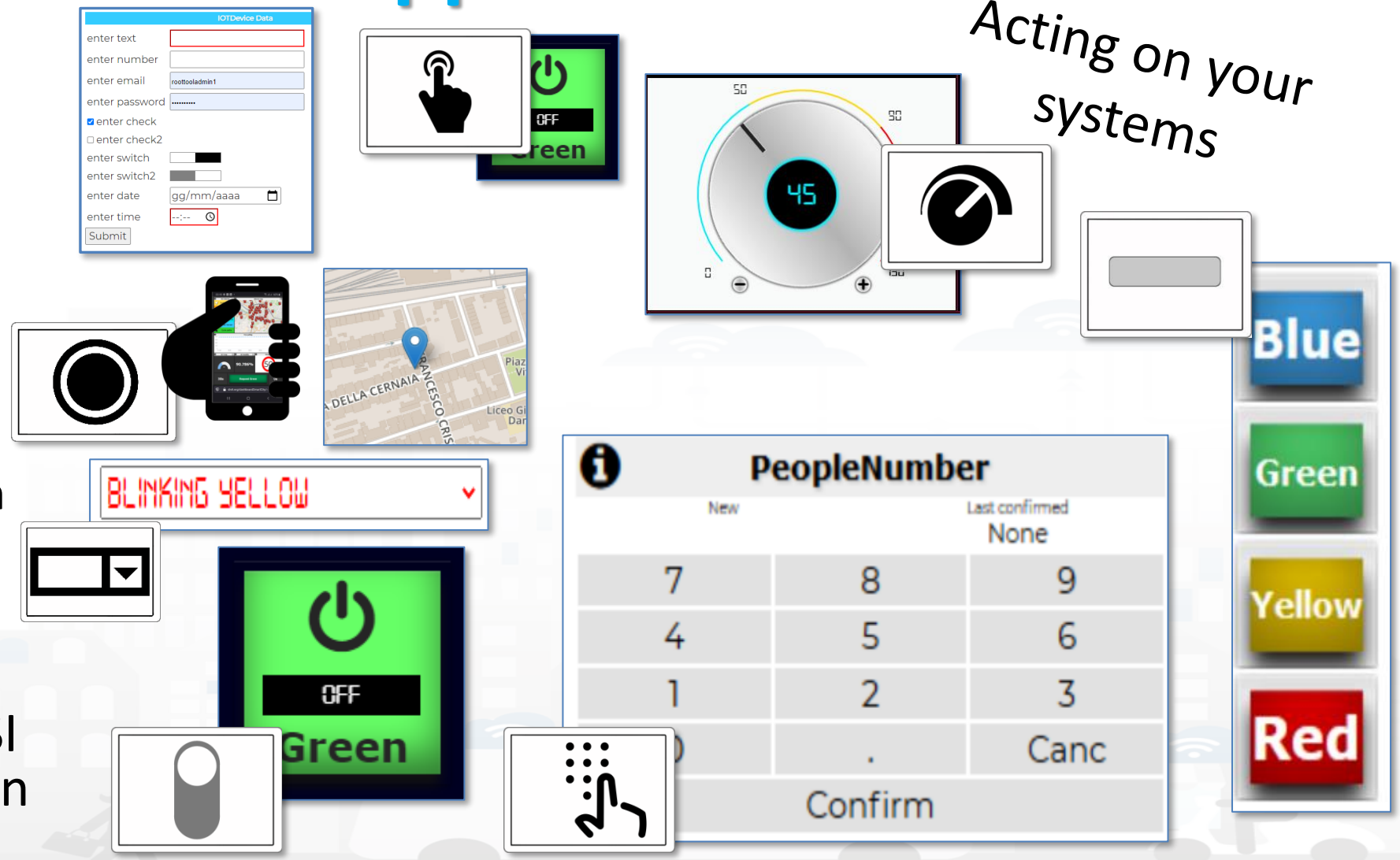
- Input/output
- Secure
- Advanced in graphics
- Single Sign On
- Several HLT
- Fully integrated
- Historical data
- Full Synoptics
- Etc..

- Remote for IOT Edge via WebSocket Secure



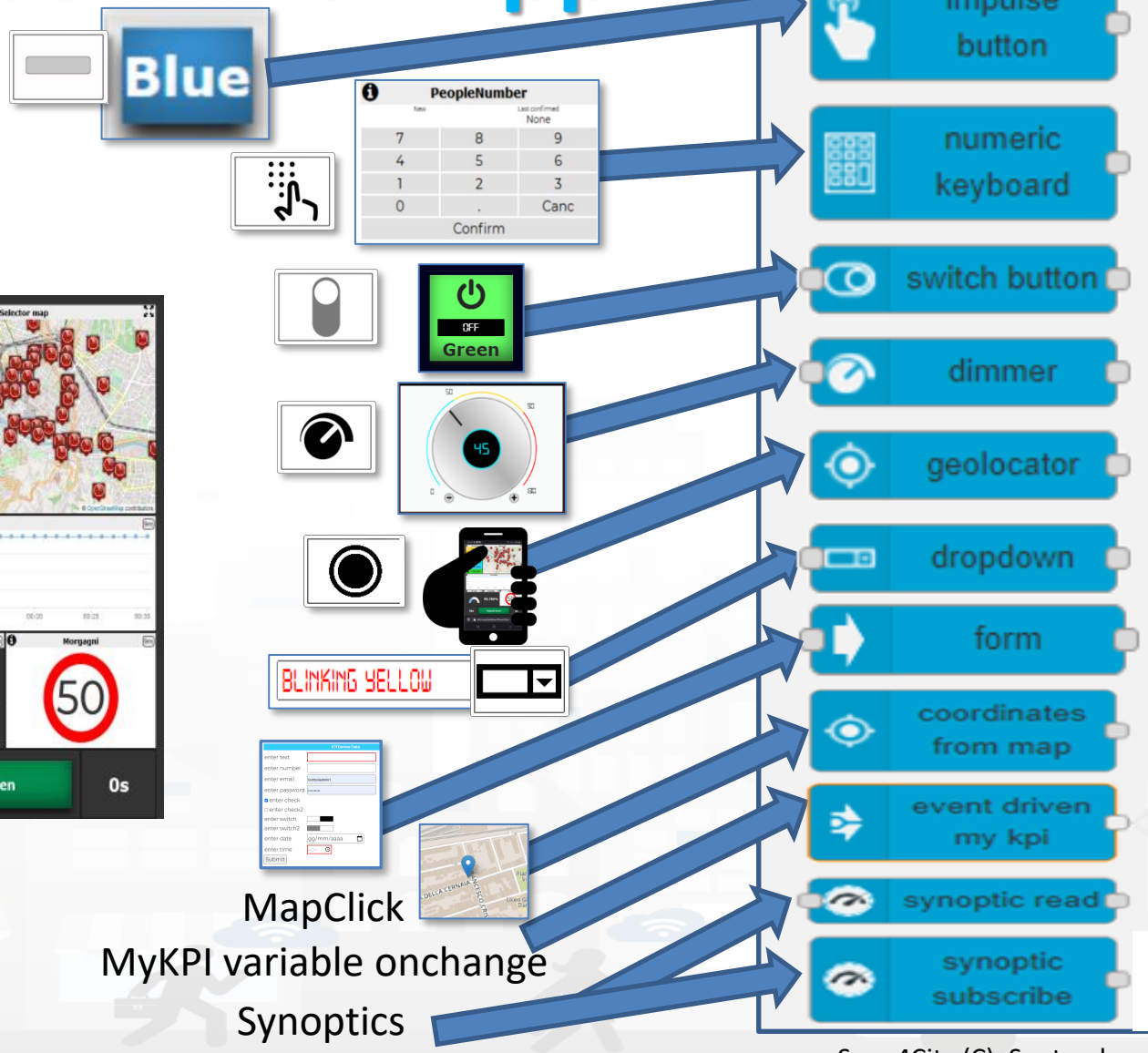
From Dashboard to IoT App

- **Widgets:**
  - Impulse Button
  - Button
  - Switch
  - Dimer/Knowb
  - KeyPad
  - Geolocator
  - Selection/Dropdown
  - Form
  - Map Picking
- **Registered** on some IOT brokers with NGSI mutual authentication



Nature

Dashboard-IoT App

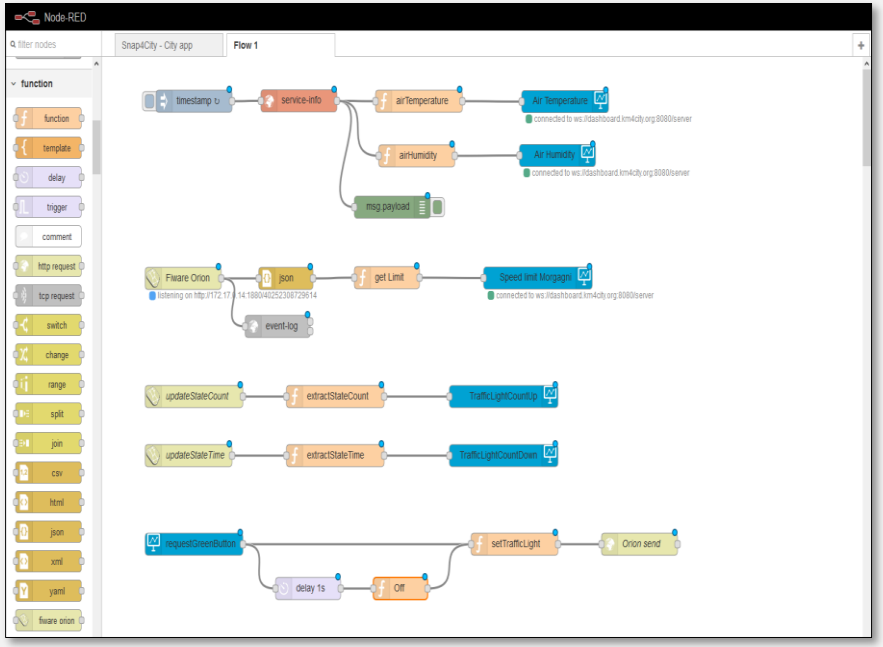


MapClick

MyKPI variable onchange

Synoptics

From Dashboard to IOT App

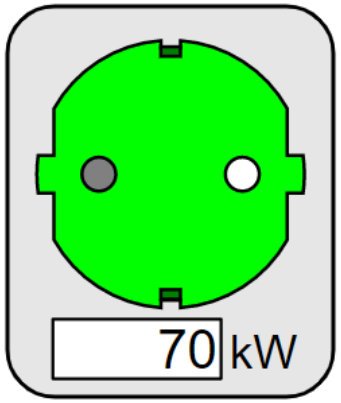


IOT Application

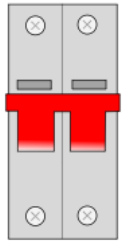
SVG Custom Widgets Examples 2

Tue 17 Nov 18:46:47

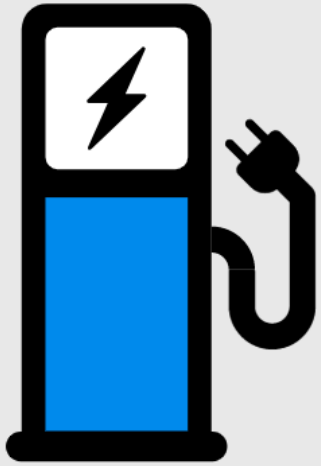
SVG shucko plug



Schuko switch



Charging Station Status



Legenda

Charging Station Status

Set on the keypad one of the following values

0 = ERROR (RED)

1 = AVAIBLE (GREEN)

2 = BOOKED (YELLOW)

3 = CHARGING

9999 = white icon

Charging Station status

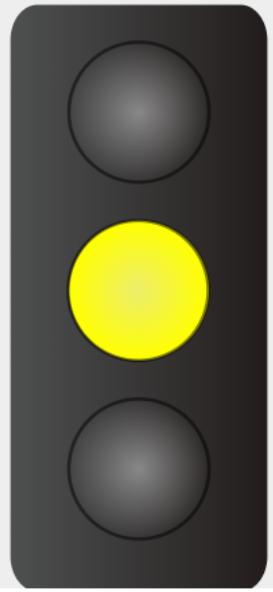
New | Last confirmed: None

7	8	9
4	5	6
1	2	3
0	.	Canc
Confirm		

Underpass

Set tunnel st...

Traffic Light



Traffic Light status set

YELLOW LIGHT

Speed Limit Set

New | Last confirmed: None

7	8	9
4	5	6
1	2	3
0	.	Canc
Confirm		

Dynamic Speed Limit Sign

Speed Limit Explaination

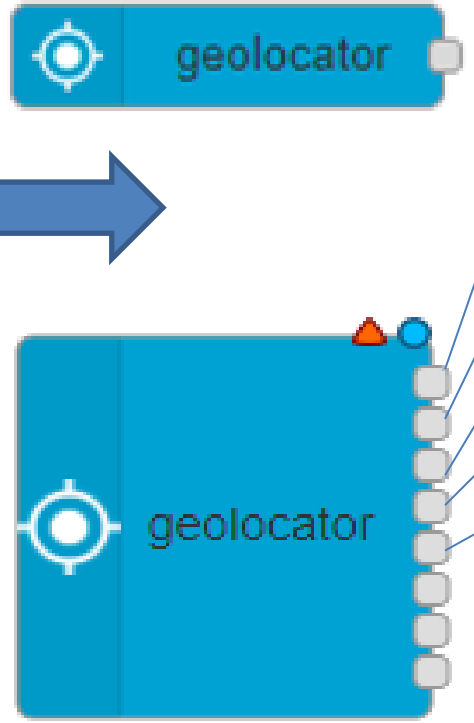
Speed Limit Custom Widget example

Write the speed limit by using the keypad and click CONFIRM.

9999 = white sign.

https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=Mjk4Ng==

Geolocation of Mobile Device



- Complete message
  - Returns a JSON containing all information about geolocation
- Latitude
  - Returns the latitude
- Longitude
  - Returns the longitude
- Accuracy
  - Returns the accuracy of latitude and longitude
- Altitude
  - Returns the altitude
- Altitude Accuracy
  - Returns the altitude accuracy
- Heading
  - Returns the heading
- Speed
  - Returns the speed

Web Browser GPS data rendering the Snap4City Dashboard can be passed to IOT Applications and saved ☺

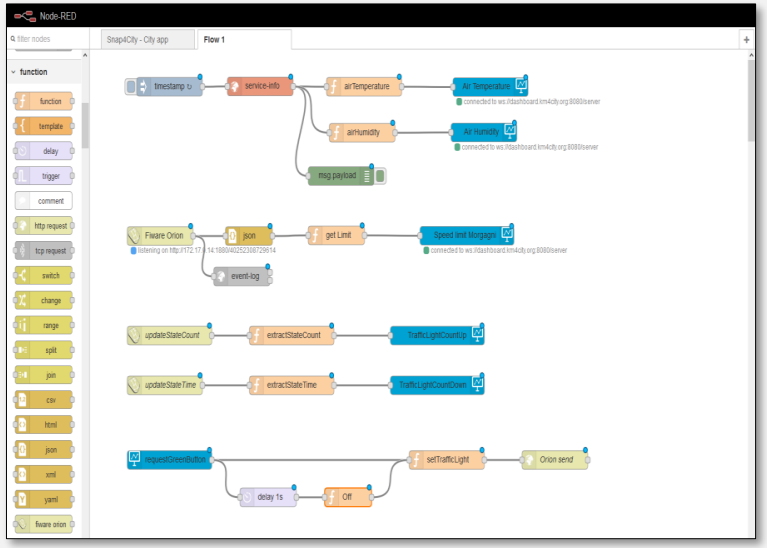
UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE | DISIT DISTRIBUTED SYSTEMS AND INTERNET TECHNOLOGIES LAB

Nature

SNAP4CITY | Node-RED

Dashboard-IOT App

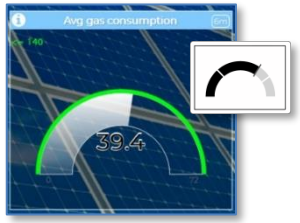
From IoT App to Dashboard

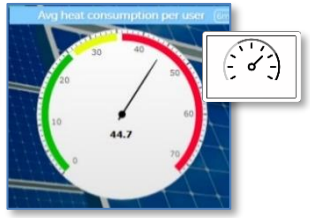


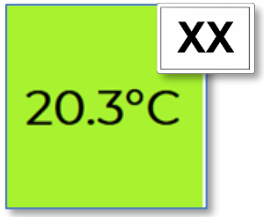
IOT Application

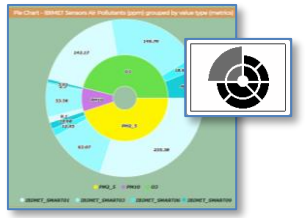
- gauge chart
- single content
- speedometer
- horizontal single bar
- vertical single bar
- web content
- time trend
- bar series
- radar series
- pie chart
- curved line series
- table content
- calendar
- speak synthesis
- synoptic write
- Selector - Map

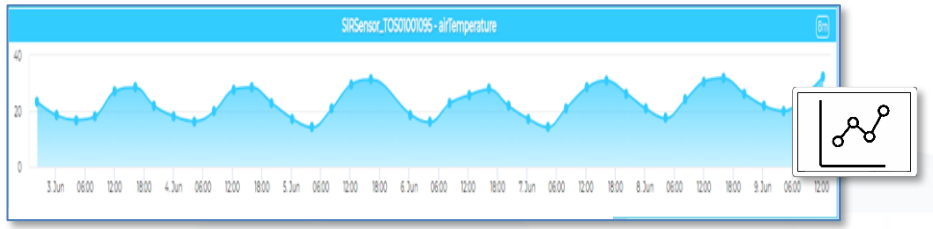
- Snap4D3
- dashboard - map
- event table
- device table

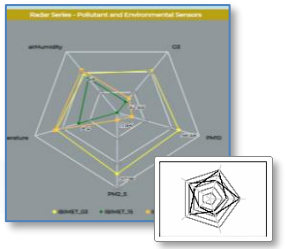


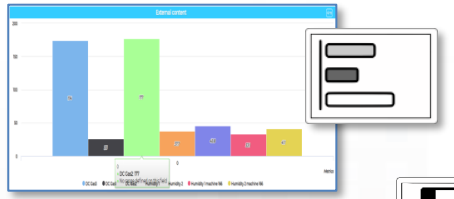


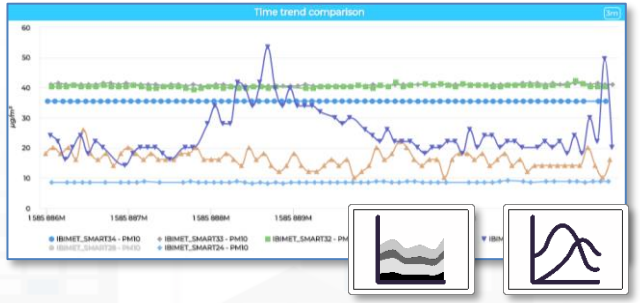


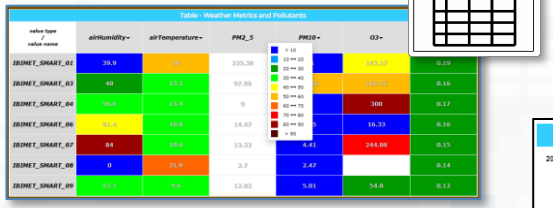


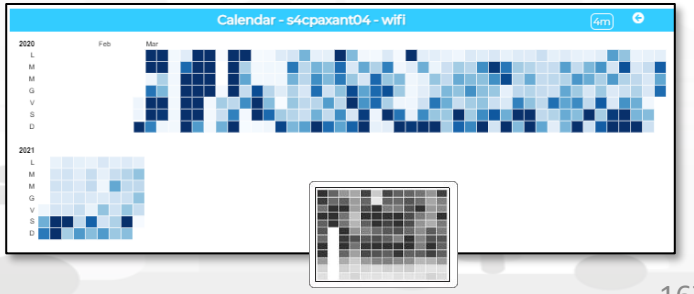




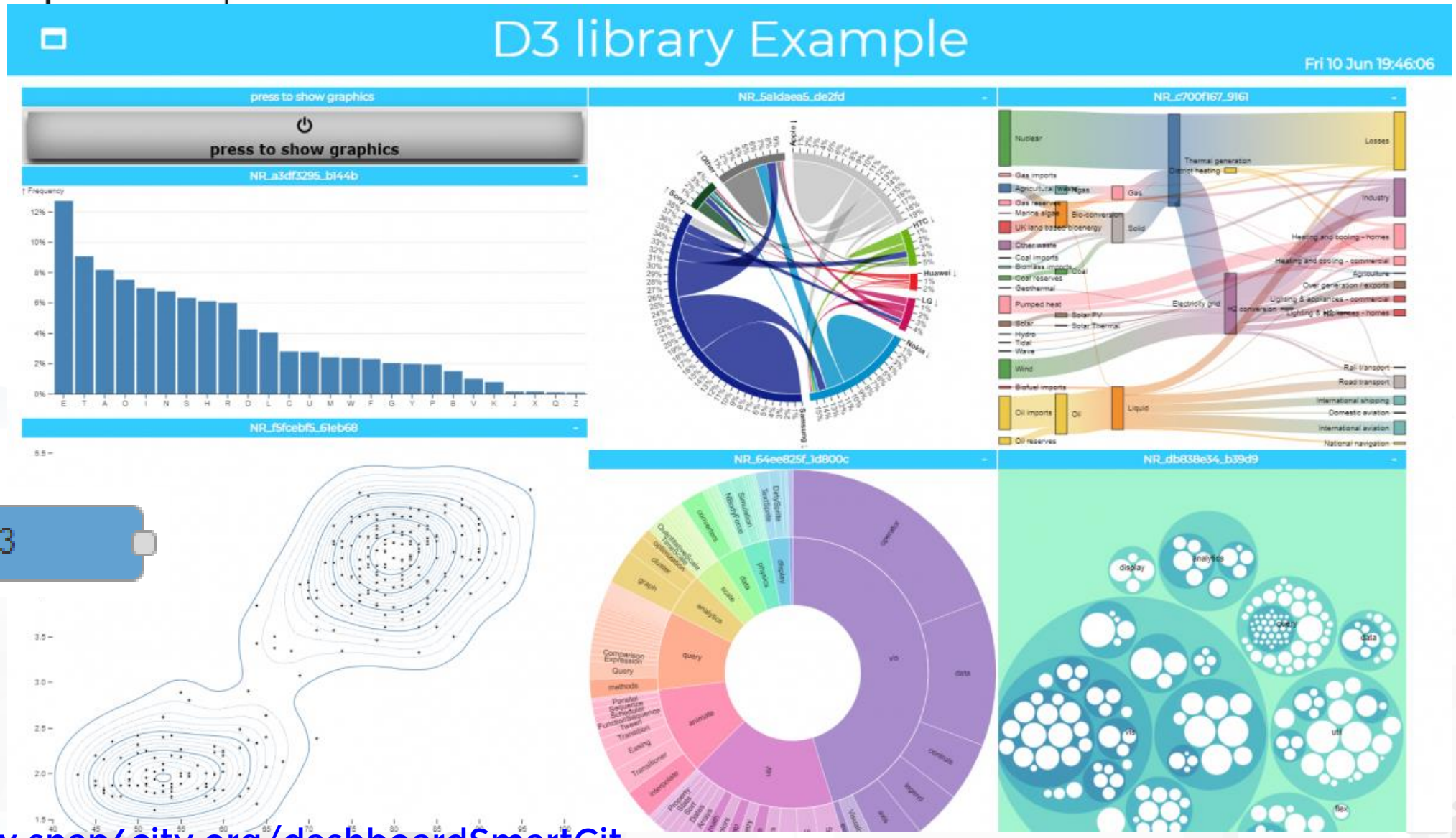






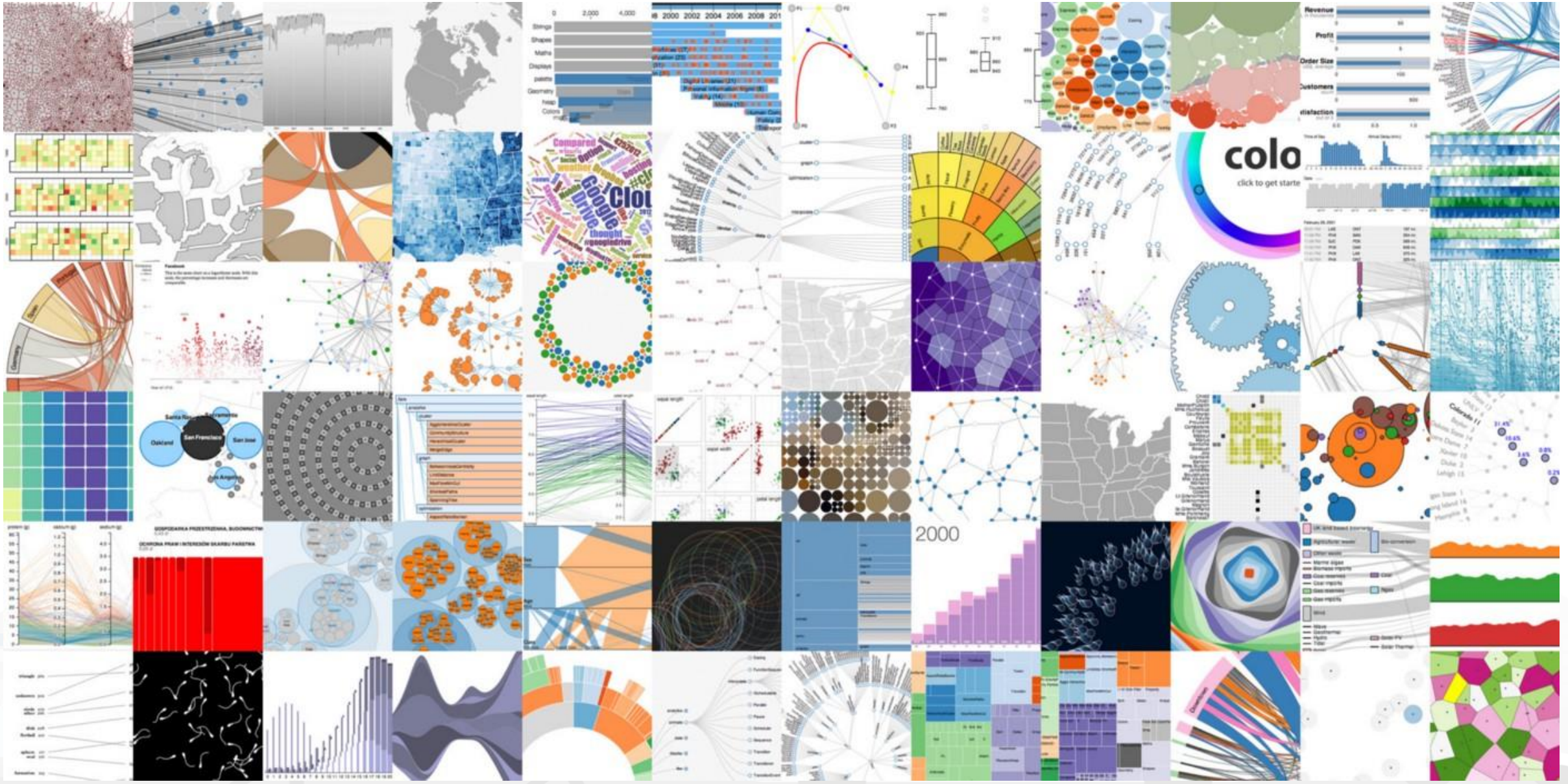


+ D3.JS Widgets



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

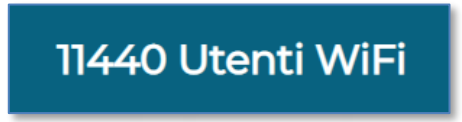
D3.js graphs



Single Content Widget (flexibility)

From Dashboard Editor and IOT Applications, accepts in input:

- Numbers
- String
- HTML code



https://www.snap4city.org/578

Controlling Single Content Wdgs from IoT App

```
{
   "metricId": [The ID of a KPI], // or [Service Uri Of A Sensor]
   "metricHighLevelType": "MyKPI", // or "Sensor"
   "metricName": [The name of the MyKPI], // or [The name of the Sensor],
   "metricType": [The type of the measure saved in the KPI] // or [A measure recorded by the sensor]
}
```

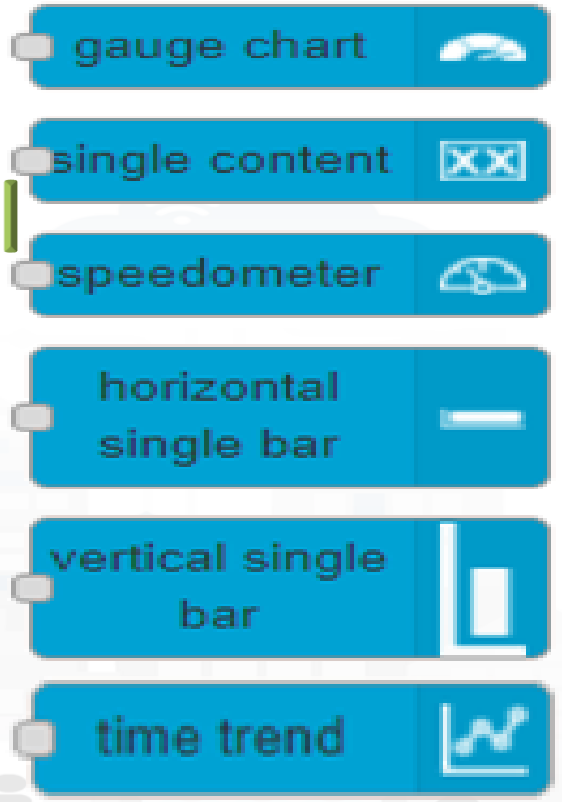
Two examples:

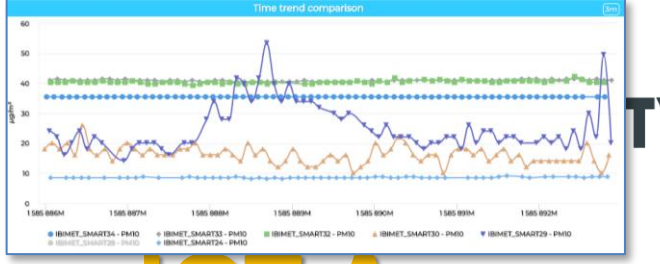
```
{
   "metricId": "17056579",
   "metricHighLevelType": "MyKPI",
   "metricName": "OBD2 - Intake Air Temperature",
   "metricType": "Temperature"
}

{
   "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
   "metricHighLevelType": "Sensor",
   "metricName": "tusc_weather_sensor_ow_3166540",
   "metricType": "airTemperature"
}
```

Send Control The Widget will get the data from storage

Similarly





https://www.snap4city.org/575

Controlling MultiSeries from IOT App

Expected JSON in input

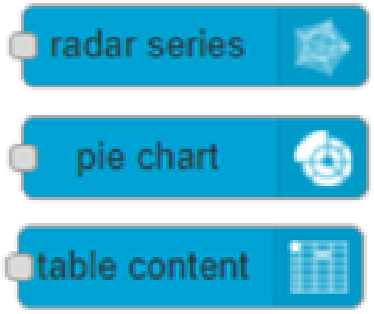
```
[
  {
    "metricHighLevelType": "Sensor",
    "metricName": [The name of the Sensor],
    "smField": [A measure recorded by the sensor],
    "serviceUri": [Service Uri Of A Sensor]
  },
  {
    "metricHighLevelType": "MyKPI",
    "metricName": [The name of the MyKPI],
    "smField": [The type of the measure saved in the KPI],
    "serviceUri": [The ID of a KPI]
  },
  {
    "metricHighLevelType": "Dynamic",
    "metricName": [The name of the dynamic data],
    "smField": [The type of the dynamic data],
    "metricValueUnit": [The unit of the dynamic data],
    "value": [An array of array of timestamp and value]
  }
]
```

```
[
  {   "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
    "metricHighLevelType": "Sensor",   "metricName": "tusc_weather_sensor_ow_3166540",
    "metricType": "airTemperature"
  },
  {   "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3182522",
    "metricHighLevelType": "Sensor",   "metricName": "tusc_weather_sensor_ow_3182522",
    "metricType": "airTemperature"
  },
  {   "metricId": "17057447",
    "metricHighLevelType": "MyKPI",   "metricName": "OBD2 - Intake Air Temperature",
    "metricType": "Temperature"
  },
  {   "metricId": "17056579",
    "metricHighLevelType": "MyKPI",   "metricName": "OBD2 - Intake Air Temperature",
    "metricType": "Temperature"
  },
  {   "metricId": "",
    "metricHighLevelType": "Dynamic",   "metricName": "BatteryGalaxyNote",
    "metricType": "Temperature",   "metricValueUnit": "°C",
    "measuredTime": "2019-11-21T14:51:00Z",   "value": 42
  },
  {   "metricId": "",
    "metricHighLevelType": "Dynamic",   "metricName": "Storage",
    "metricType": "Space",   "metricValueUnit": "Gb",
    "measuredTime": "2019-11-21T14:51:00Z",   "value": 12
  }
]
```

https://www.snap4city.org/575

Controlling BarSeries from IOT App

Similarly



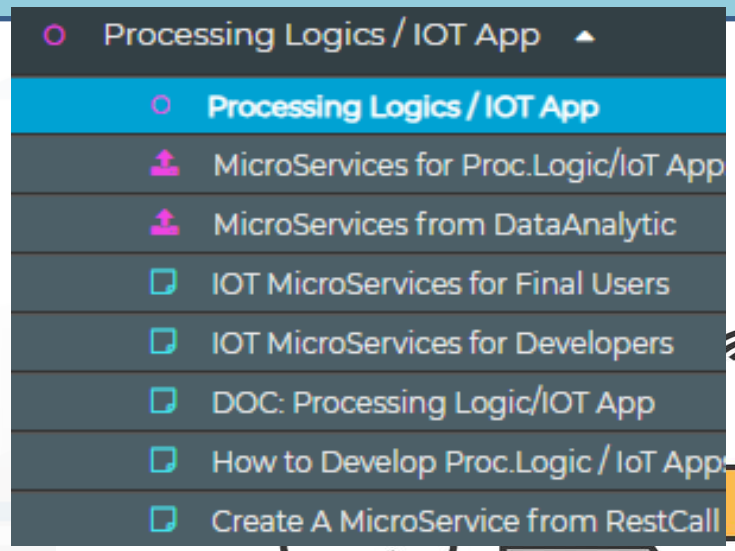
Expected JSON in input

```
[
  {
    "metricId": [Service Uri Of A Sensor],
    "metricHighLevelType": "Sensor",
    "metricName": [The name of the Sensor],
    "metricType": [A measure recorded by the sensor]
  },
  {
    "metricId": [The ID of a KPI],
    "metricHighLevelType": "MyKPI",
    "metricName": [The name of the MyKPI],
    "metricType": [The type of the measure saved in the KPI]
  },
  {
    "metricId": "",
    "metricHighLevelType": "Dynamic",
    "metricName": [The name of the dynamic data],
    "metricType": [The type of the dynamic data],
    "metricValueUnit": [The unit of the dynamic data],
    "measuredTime": [The ISO time of the measure of dynamic data],
    "value": [A dynamic value]
  }
]
```


```
[
{"metricId":"17056320",
        "metricHighLevelType":"MyKPI",
        "metricName":"SiiMTuscanyTrackerLocation",
        "metricType":"Velocity"},
{"metricId":"http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
        "metricHighLevelType":"Sensor",
        "metricName":"tusc_weather_sensor_ow_3166540",
        "metricType":"airTemperature"},
{"metricId":"https://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.d...",
        "metricHighLevelType":"Sensor",
        "metricName":"tusc_weather_sensor_ow_3182522",
        "metricType":"airTemperature"},
{"metricId":"",
        "metricHighLevelType":"Dynamic",    "metricName":"BatteryTemperatureGalaxyNote",
        "metricType":"Gradi Centigradi",    "metricValueUnit":"°C",
        "measuredTime":"2019-11-21T14:51:00Z",
        "value":55.395468539280635},
{"metricId":"",
        "metricHighLevelType":"Dynamic",    "metricName":"BatteryTemperaturemia",
        "metricType":"Gradi Centigradi",    "metricValueUnit":"°C",
        "measuredTime":"2019-11-21T14:51:00Z",
        "value":51.396725502373464},
{"metricId":"",
        "metricHighLevelType":"Dynamic",    "metricName":"BatteryTemperaturemia",
        "metricType":"airTemperature",      "metricValueUnit":"°C",
        "measuredTime":"2019-11-21T14:51:00Z",
        "value":29.150364690965127}
]
```

COFFEE BREAK

Device Tables vs IoT App data
Getting data from Dashboards



Node-RED

https://www.snap4city.org/795

https://www.snap4city.org/809

Device Table

Similar ma More Generic of the Event Table IoT App block and Widget

Device Table Node accepts multiple formats:

msg.payload =

{ ordering: "dateObserved",

query: "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",

actions: ["https://upload.wikimedia.org/wikipedia/commons/thumb/6/6d/Windows_Settings_app_icon.png/1024px-Windows_Settings_app_icon.png", "pin"],

columnsToShow: ["dateObserved", "vehicleFlow"]

}

ASCAPI

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

- query: "https://www.snap4city.org/superservicemap/api/v1/iot-search/?selection=43.77;11.2&maxDists=700.2&model=CarPark",
- query: "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**",
- query: "https://www.snap4city.org/superservicemap/api/v1/iot-search/?selection=43.77;11.2&maxDists=200.2&model=metrotrafficsensor&valueFilters=vehicleFlow>10;vehicleFlow<400&serviceUri=http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO1;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO10;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO11;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO13;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO14;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO15;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO16;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO17;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO18;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO19;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO2;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO20;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO21;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO22;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO23;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO24;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO25;http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO26 ",

https://www.snap4city.org/809

Device Table Testing double

Thu 1 Sep 13:44:38

DT2 4m

Show 5 Search:

First << Prev 1 2 3 Next >> Last

device	capacity	dateObserved	Actions
CarParkStazioneFirenzeS.M.N.	877	2022-09-01T11:33:01.681Z	
CarParkStazioneFortezzaFiera	530	2022-09-01T11:33:01.681Z	
CarParkS.Ambrogio	379	2022-09-01T11:33:01.681Z	
CarParkAlberti	313	2022-09-01T11:33:01.681Z	
CarParkPieracciniMeyer		2022-09-01T11:33:01.681Z	

DT1 43s

Show 5 Search:

First << Prev 1 2 3 Next >> Last

device	capacity	dateObserved	Actions
CarParkStazioneFortezzaFiera	530	2022-09-01T11:33:01.681Z	
freeParkingLots 277			
occupacy			
occupiedParkingLots 253			
status			
CarParkStazioneFirenzeS.M.N.	877	2022-09-01T11:33:01.681Z	
CarParkS.Ambrogio	379	2022-09-01T11:33:01.681Z	
CarParkParterre	1006	2022-09-01T11:33:01.681Z	
CarParkCareggi	514	2022-09-01T11:33:01.681Z	

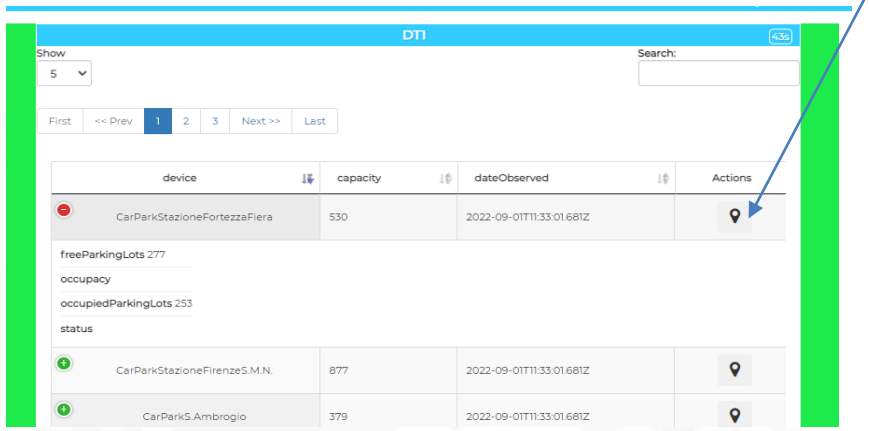
https://www.snap4city.org/809

Device Table

- The Node accepts in Input:
  - Ordering by a variable/attribute
  - List of SURI (one prefix and a number of Names)
  - List of Actions as icons to be clicked
  - List of Columns to be shown on the table
  - Query selection + filters
- Output:
  - The Action clicked by the user with the name of the SURI and ID

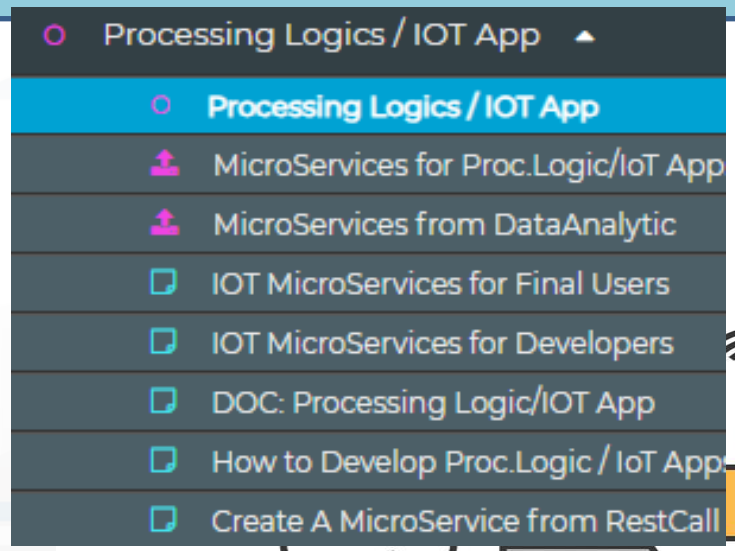
https://www.snap4city.org/809

```
msg.payload = {
ordering: "status",
prefix: "http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/",
devices: ["Alarm001", "Car001", "Velox001",
"Earthquake001","Theater002", "Landslide001","Theater002",
"Landslide001"],
actions: ["pin", "Action1", "Action1", "https://www.aa.org/my.img", ],
columnsToShow:["device", "startDate", "endDate", "status"]
}
```




```
{"device":"Car001",
"prefix":"http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/",
"ordering":"startDate",
"action":"Pin"
}
```

Widgets MAP with business intelligence on IoT Apps



Node-RED

Maps Business Logic vs IOT Apps

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

data

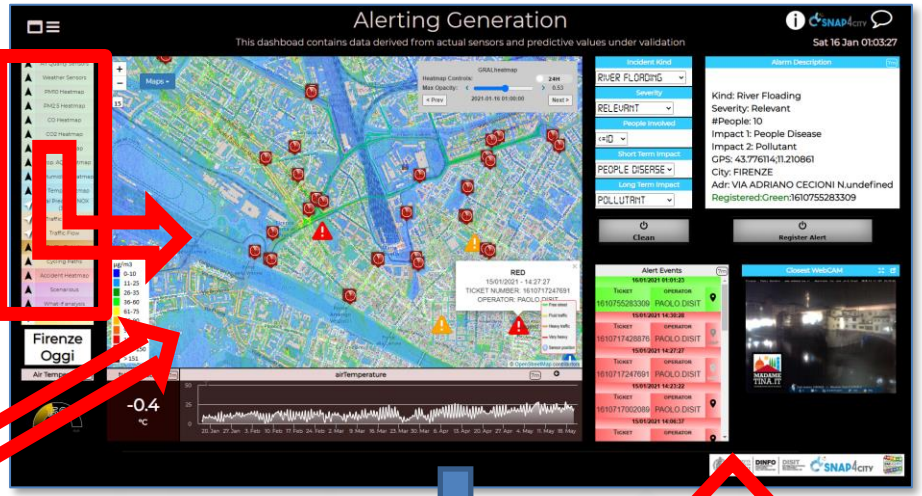
data

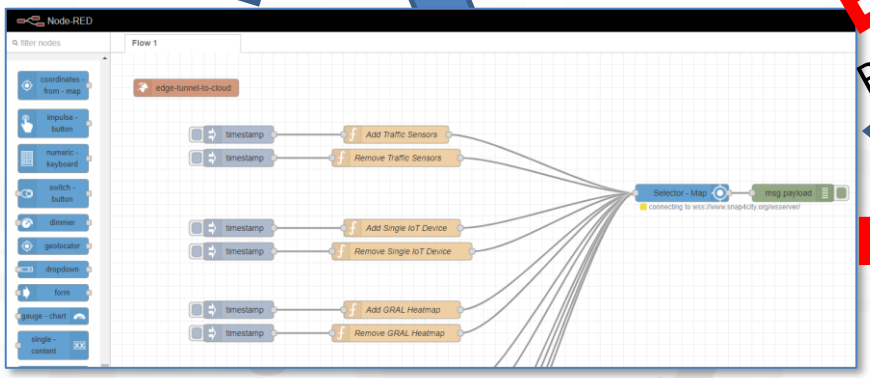
Selector

Data, changed data

References, commands, selections

selections, positions, ServiceURI

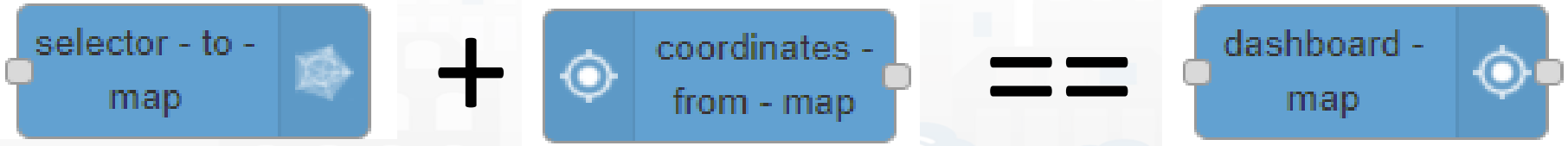




IoT App, Node-RED nodes: IoT App vs Dashbords

- **Coordinates From Map:**
  - Get GPS click from Map
  - Get Selected ServiceURI from Map
- **SelectorMap:**
  - Send commands, references, data to Maps

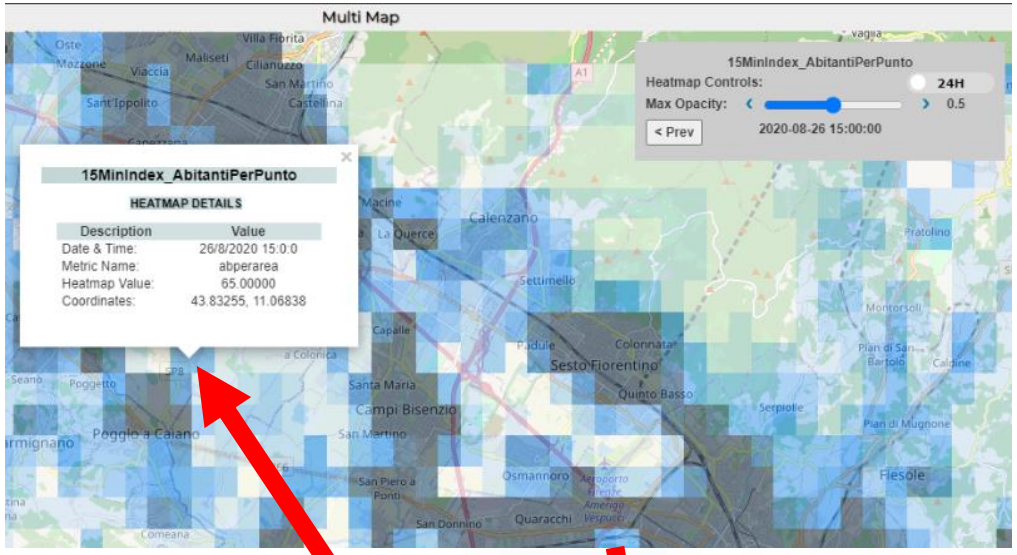
- **Dashboard Map provides:**
  - **All Features of**
    - **Coordinate From Map:**
    - **SelectorMap:**



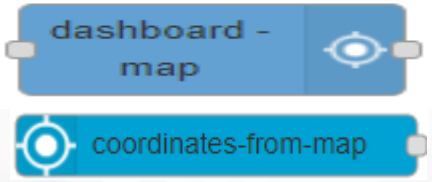
***SelectorMap* and *Coordinates From Map* will be Deprecated from Snap4City Library in late 2022**

Dashboard Map will be the only one supported since April 2022 and after

Multi Data Map GPS Location Picking vs IOT App



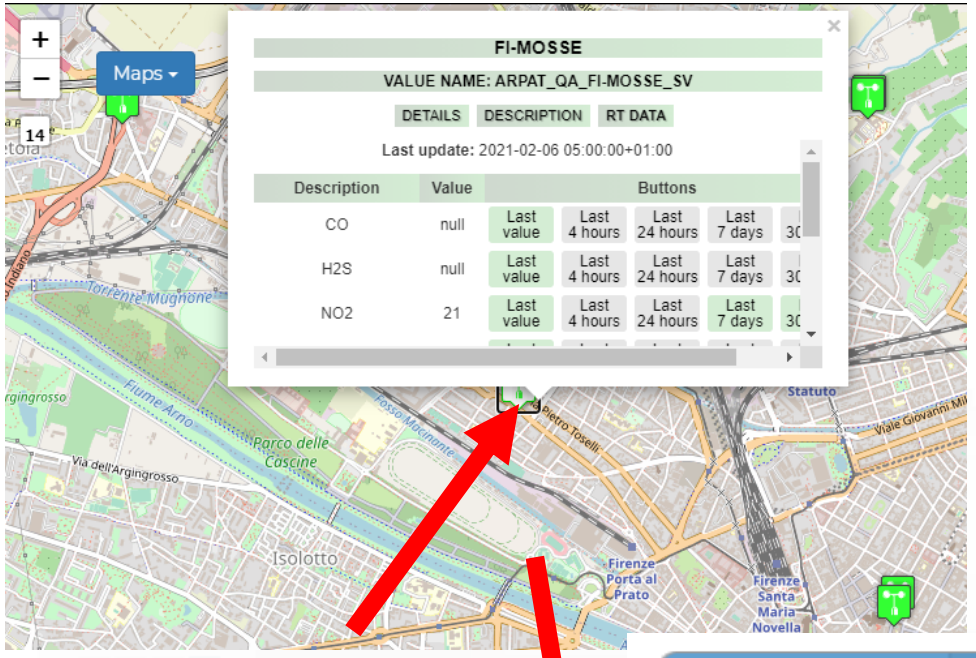
1) Click



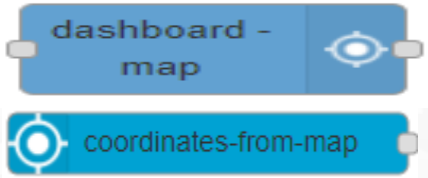
2) GET event with:
-- Lat,Long
-- ServiceURI

- 3) The click on the map passes GPS coordinates into IOT App. Thus you can use them to:
  - search for location
  - picking the value of one or more heatmaps
  - dynamically change data on widgets and dashboards
  - Etc.

Multi Data Map ServiceURI selection vs IOT App



1) Click on PIN

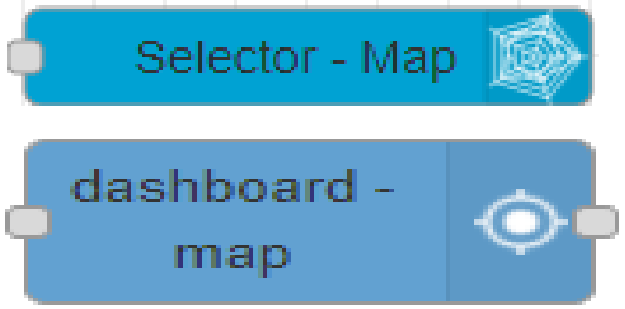


2) GET event with:
- Lat,Long
- **ServiceURI**

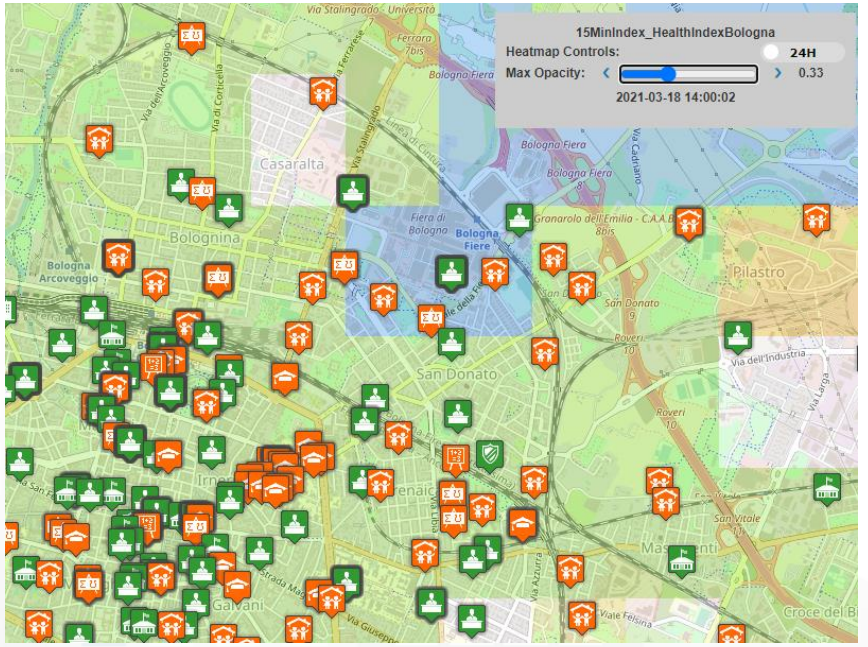
- 3) The click on the map passes GPS coordinates into IOT App and the ServiceURI. Thus you can use them to:
  - search for location
  - picking the value of one or more heatmaps
  - dynamically change data on widgets and dashboards
  - Get all the ServiceURI information and exploit them on Business Logic
  - Etc.

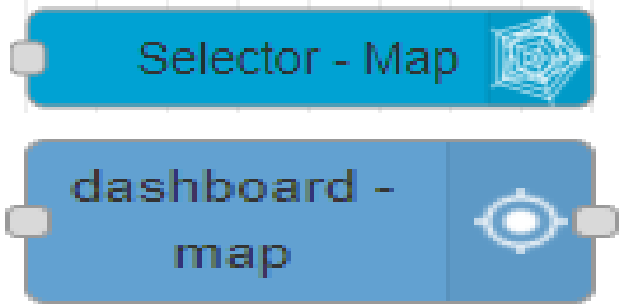
Selector Map

Controlling Maps from IOT Apps



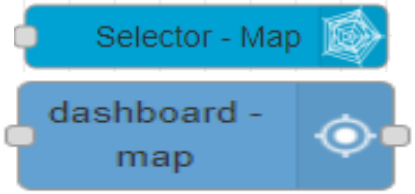
- User manual: https://www.snap4city.org/774
- To control Multi Data Map from IOT App
  - Add/remove a Category/SubCategory of Entities, via ***more option query***
  - Add/remove a single Device/PIN, MyPOI, MyKPI, Dynamic Pins, moving devices, etc.....
  - Add/remove cycling paths
  - Add/remove OD Matrix
  - Add/remove an Heatmap, a Traffic Flows, …
  - Add/remove multiple entities with multiple More Option Queries
  - Add/remove Special Tools: scenarios, whatif, etc.
  - Add/remove a set/single temporary GeoInfoPin






```
msg.payload = {
"type": "addSelectorPin",
"target": "w_Map_956_widgetMap32131",
"passedData": {
        "desc": "my studio temperature - GP -",
        "query": "datamanager/api/v1/poidata/17055853",
        "color1": "#ffdb4d",
        "color2": "#fff5cc",
        "display": "pins",
        "queryType": "MyPOI",
        "pinattr": "square",
        "pincolor": "#959595",
        "symbolcolor": "undefined",
        "iconTextMode": "text",
        "altViewMode": "None",
        "bubbleSelectedMetric": ""
        }
};
return msg;
```

Add MyKPI
Add MyPOI

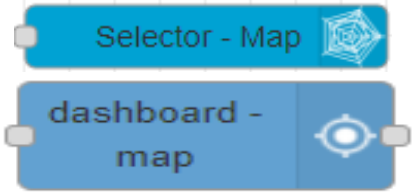


Add Heatmap

```
msg.payload = {
"type": "addHeatmap",
"target": "w_Map_956_widgetMap32131",
"passedData":"https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=FirenzeTrafficRealtime&trafficflowmanager=true",
"passedParams": {
    "desc": "Traffic Heatmap",
    "color1": "rgba(0,179,61,0)",
    "color2": "rgba(114,235,133,1)"
    }}
```

Remove Heatmap

```
msg.payload = {
"type": "removeHeatmap",
"target": "w_Map_956_widgetMap32131«,
"isTrafficHeatmap": true
}
}
```

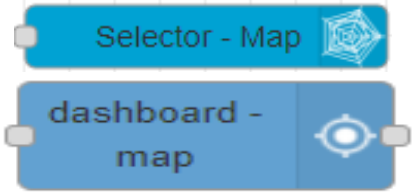


Add OD Matrix

```
payload : {
"type": "addOD",
"target": "w_Map_956_widgetMap32131",
"passedData" : "https://odmm.snap4city.org/api/get?precision=communes&from_date=2017-10-19%2000:00:00&organization=Tuscany&inflow=True&longitude=11.257123947143556&latitude=43.771837562821375",
    "passedParams": {
    "desc": "OD Matrix Toscana",
    "color1": "rgba(172,82,254,1)",
    "color2": "rgba(172,82,254,0.46)",
    } }
```

Remove OD Matrix

```
payload = {
"type": "removeOD",
"target": "w_Map_956_widgetMap32131",
}
```

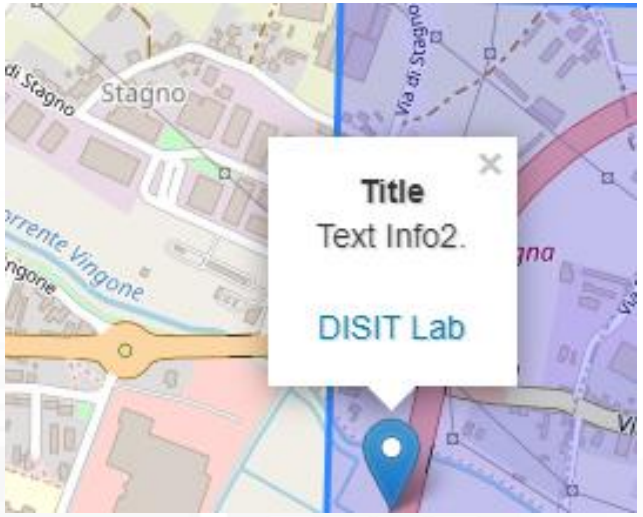


Add GeoInfoPin set of

```
payload : {
"type": "addGeoInfoPin",
"target": "w_Map_956_widgetMap32131",
"passedData": [{
      "id": "GeoInfoPin-01",
      "eventType": "GeoInfoPin",
      "textHtml": "<b>Title</b><br>Text Info2.<br><br><a href='http://www.disit.org/' target='_blank'>DISIT Lab</a>",
      "lat": " 43.76950",
      "lng": "11.125835"   }, … ]
}
```

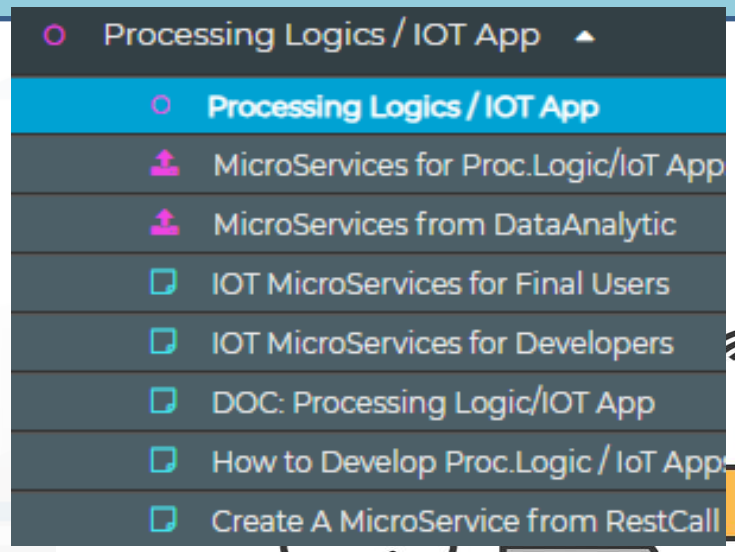
Remove GeoInfoPin set of

```
payload : {
"type": "removeGeoInfoPin",
"target": "w_Map_956_widgetMap32131",
"passedData": [{
      "id": "GeoInfoPin-01",
      "eventType": "GeoInfoPin",
      "lat": " 43.76950",
       "lng": "11.125835"
      }, ….]
}
```



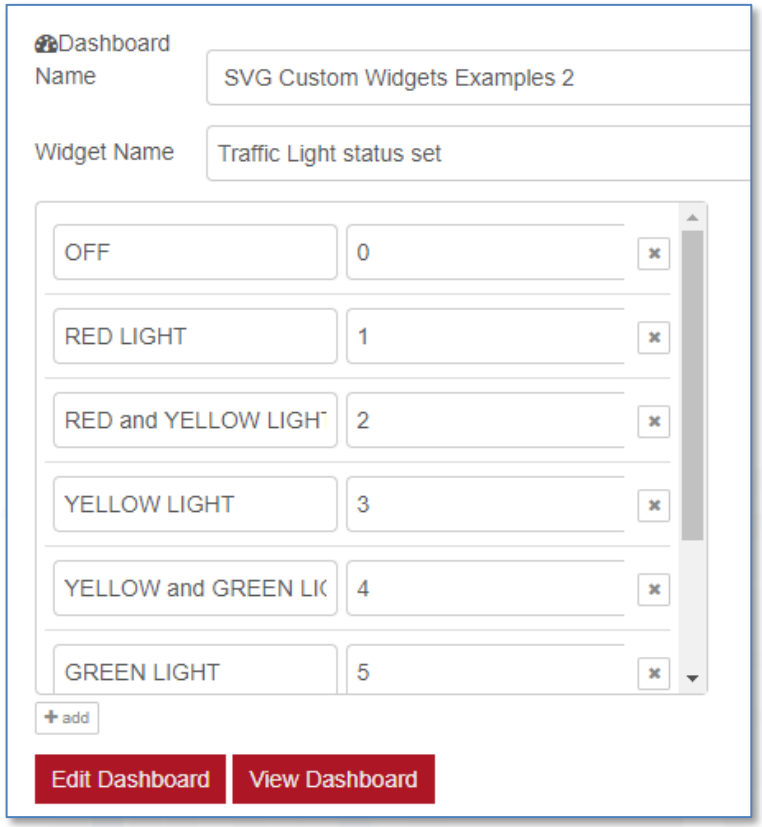
Selector and Forms vs IOT App data

Getting data from Dashboards

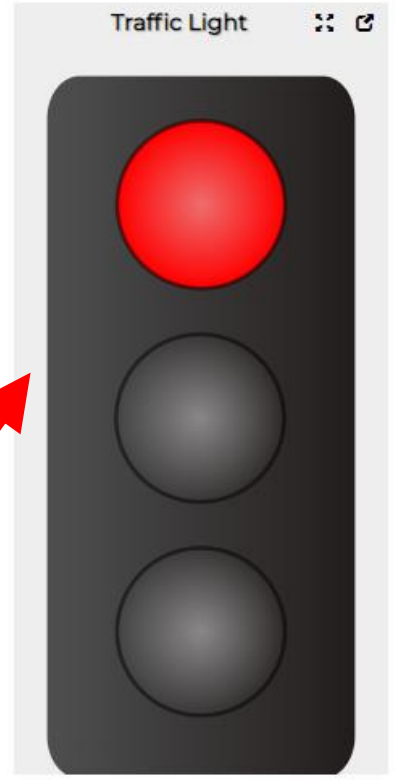


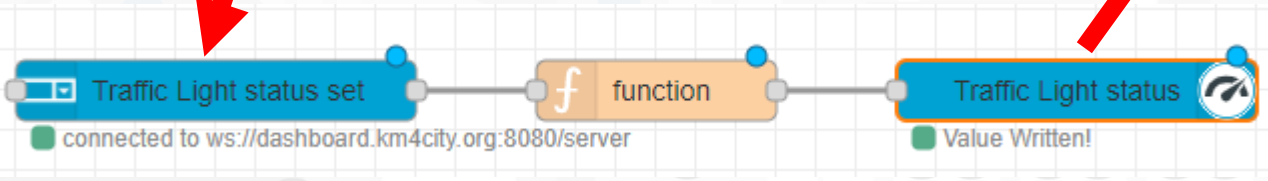
Node-RED

DropDown Selector



- Selecting MSG to be sent on the Business Logic IOT Application



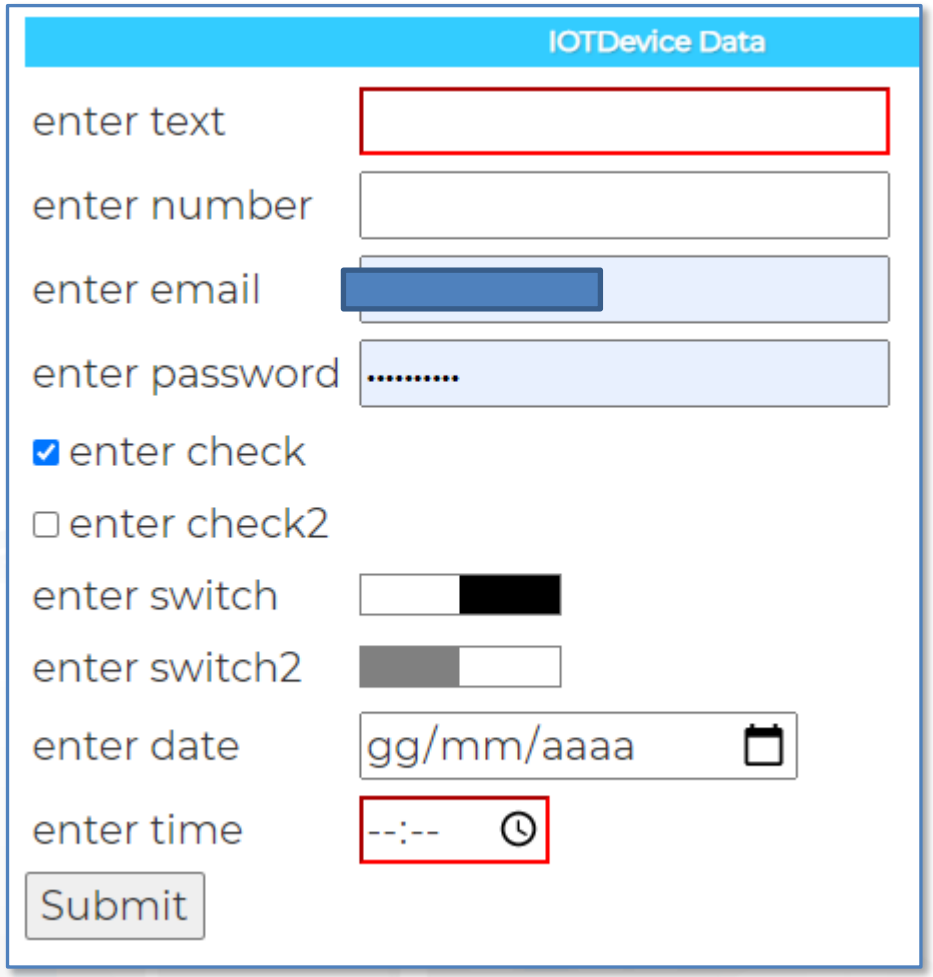


HTML

```
msg.payload = {value:JSON.parse(msg.payload).selected};
return msg;
```

Form

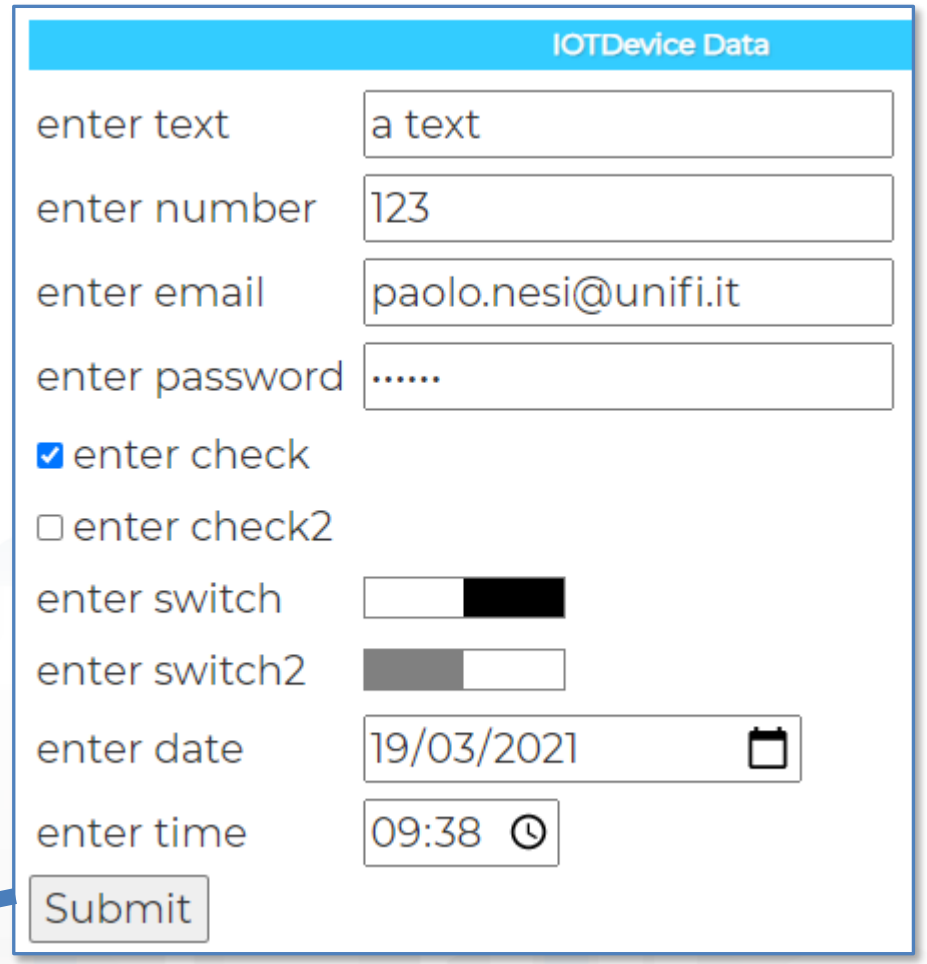

```
msg.payload ={ "form": { "options": [
{ "label": "enter text", "value": "", "type": "text", "required": true },
{ "label": "enter number", "value": "", "type": "number", "required": false },
{ "label": "enter email", "value": "", "type": "email", "required": false },
{ "label": "enter password", "value": "", "type": "password", "required": false },
{ "label": "enter check", "value": "checked", "type": "checkbox", "required": false },
{ "label": "enter check2", "value": "", "type": "checkbox", "required": false },
{ "label": "enter switch", "value": "on", "type": "switch", "required": false },
{ "label": "enter switch2", "value": "", "type": "switch", "required": false },
{ "label": "enter date", "value": "", "type": "date", "required": false },
{ "label": "enter time", "value": "", "type": "time", "required": true }
 ], "selected": [] } }
return msg;
```



HTML

Form

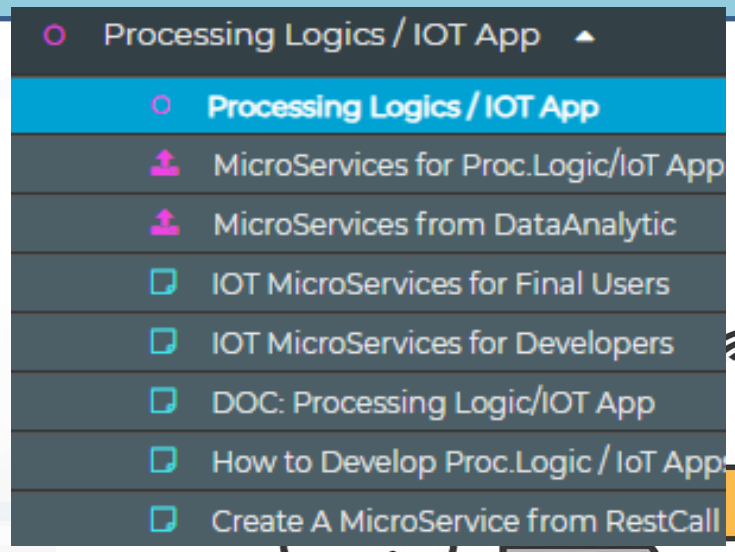

```
msg.payload ={ "form": { "options": [
{ "label": "enter text", "value": "", "type": "text", "required": true },
{ "label": "enter number", "value": "", "type": "number", "required": false },
{ "label": "enter email", "value": "", "type": "email", "required": false },
{ "label": "enter password", "value": "", "type": "password", "required": false },
{ "label": "enter check", "value": "checked", "type": "checkbox", "required": false },
{ "label": "enter check2", "value": "", "type": "checkbox", "required": false },
{ "label": "enter switch", "value": "on", "type": "switch", "required": false },
{ "label": "enter switch2", "value": "", "type": "switch", "required": false },
{ "label": "enter date", "value": "", "type": "date", "required": false },
{ "label": "enter time", "value": "", "type": "time", "required": true }
 ], "selected": [] } }
return msg;
```



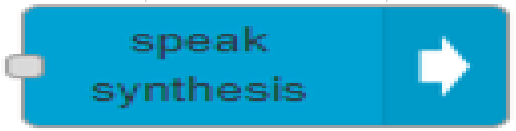
HTML

"selected":["a text","123", "paolo.nesi@unifi.it","aaaaaa", "checked","","on","","2021-03-19","09:38"]

Talk to your users
Producing voice on Dashboards



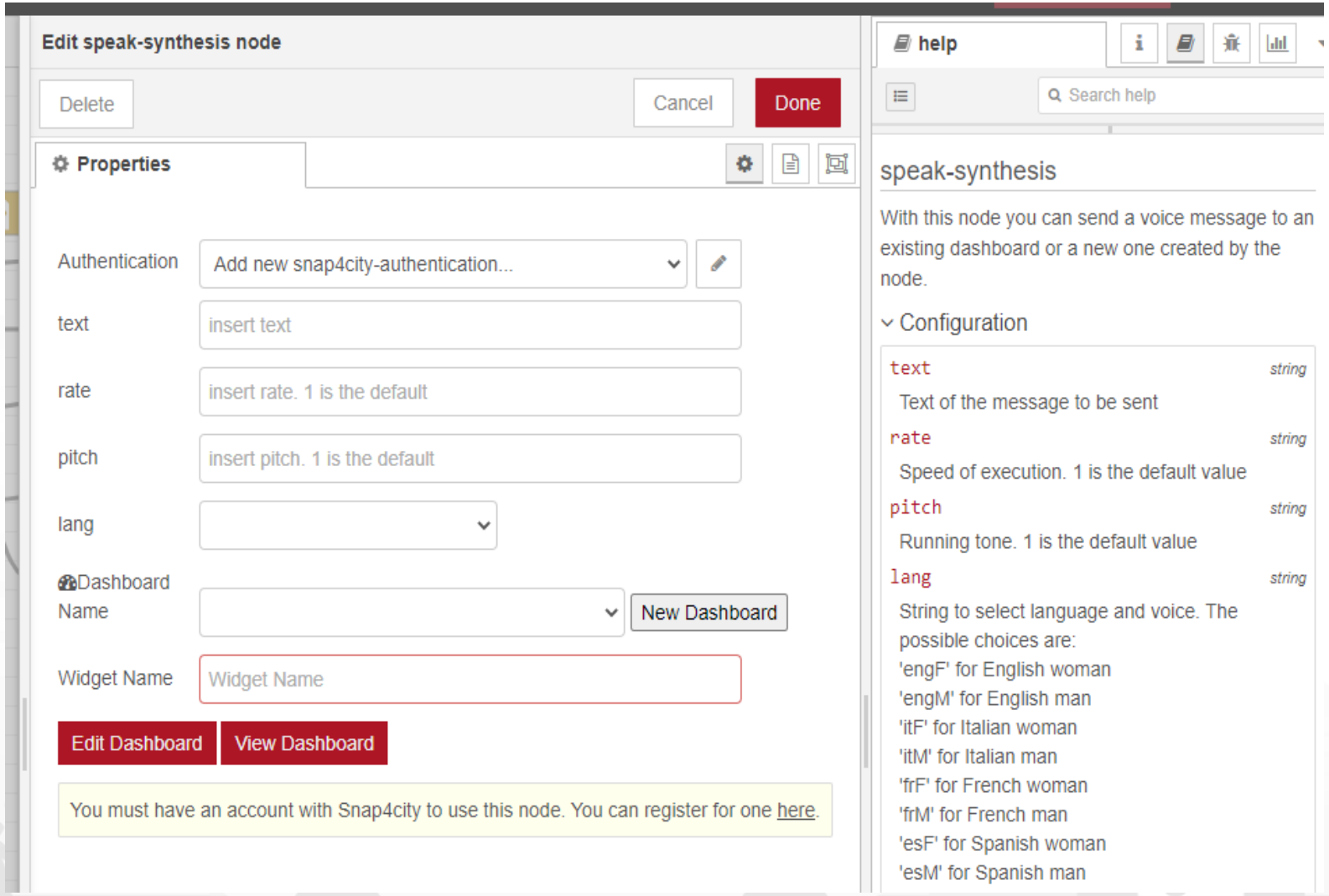
Node-RED

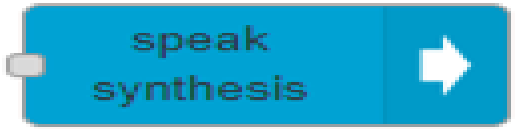


Send Voice Messages on Dashboards

- **Connectable** on all «String» Variables
- **Controllable** from IoT Applications
- Play button on Dashboard

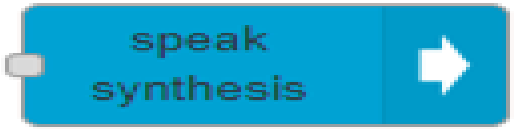
- **Configurable** as:
  - Voice Language and male, female, …
  - rate, pitch
  - ..





Send Voice Messages on Dashboards

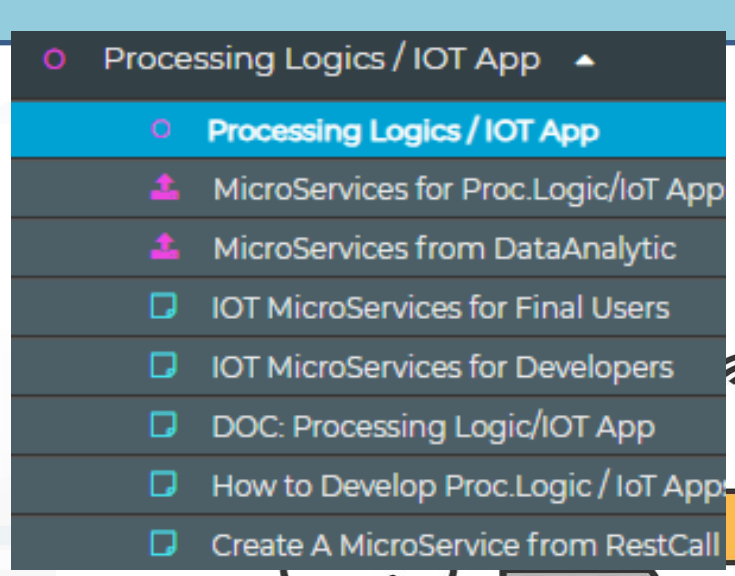
- Connectable on all «String» Variables
- Controllable from IoT Applications
- Simple Play button on Dashboard Widget
- Configured as:
  - Voice Language
  - Voice timbre: male, female, …
  - Voice Tone
  - Voice Volume
  - ..



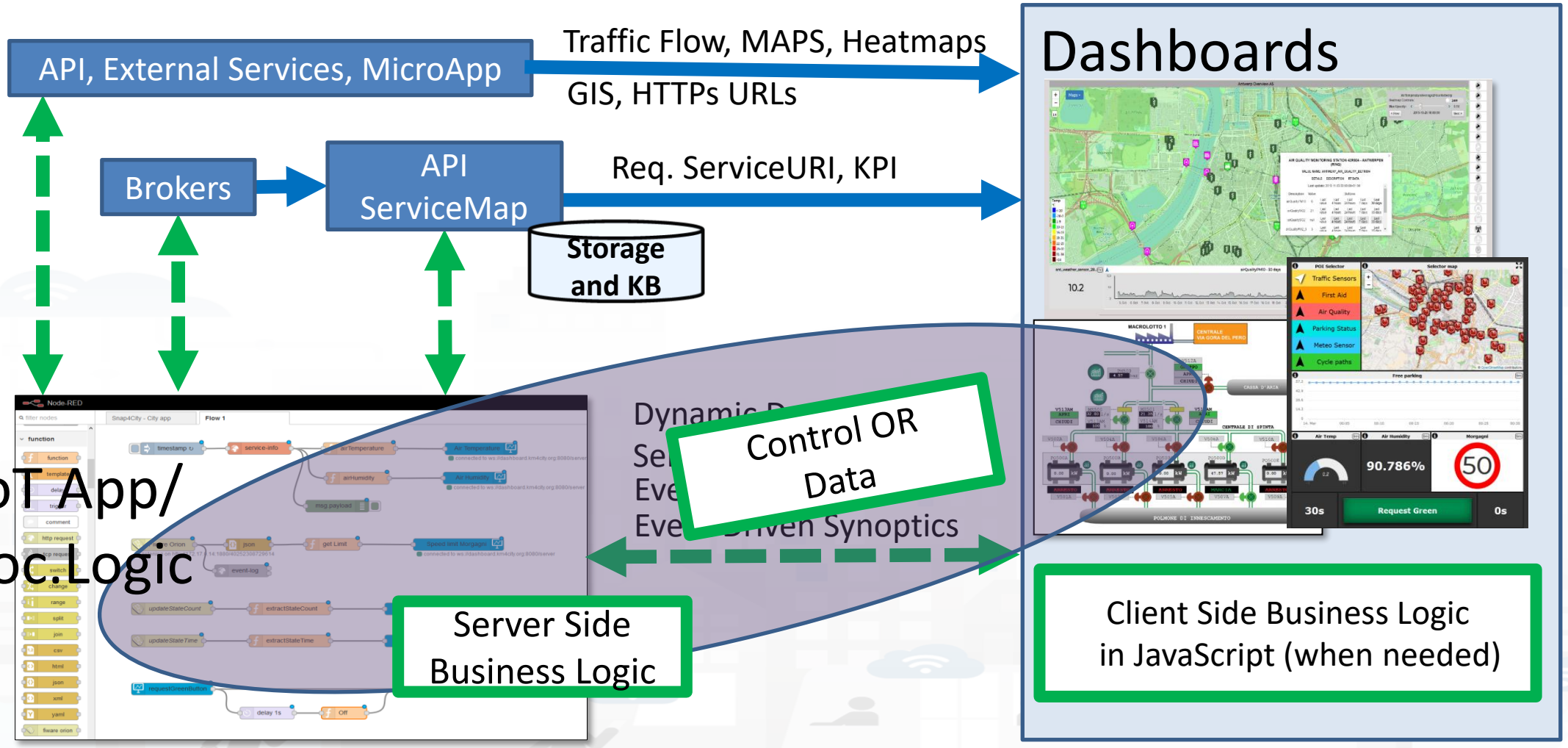
Send Voice Messages on Dashboards

- Connectable on all «String» Variables
- Controllable from IoT Applications
- Simple Play button on Dashboard Widget
- Configured as:
  - Voice Language
  - Voice timbre: male, female, …
  - Voice Tone
  - Voice Volume
  - ..

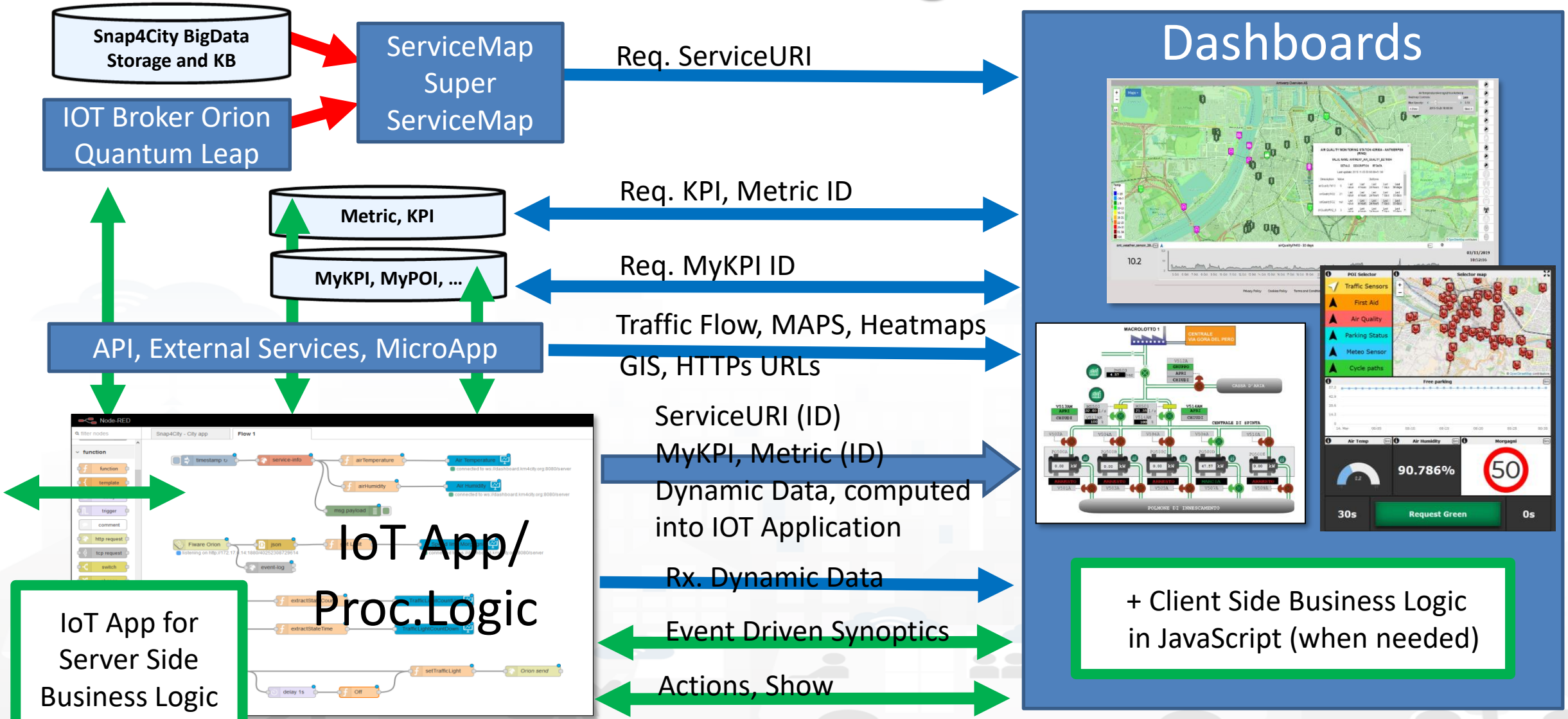
Dynamic Widgets data on Dashboard from IOT Applications



How the Dashboards exchange data



How the Dashboards exchange data



Dynamic (6/23)

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

DD: Event Driven

Legenda

- **IoT App / Proc.Logic column in previous table:**
  - **X**: means that from the IoT App you can send a new value or array to the widget directly, without the need to have is stored into Sensor or MyKPI variable, etc.
  - **CS, widget supports Change (data) Source**, in the sense that: from the IoT App is possible to send a command to the Widget to change the data source. E.g., selecting sources among: Sensors (serviceURI), MyKPI (ID), any value produced on the IoT App directly. (cs) recent additions
- **Dashboard IoT App column in previous table**:
  - **X:** there is a MicroService / node on IoT App to act on those widgets on dashboard. The data are visualized.
  - **ED, widget is Data/Event Driven**, in the sense that new data in push can be sent and the widget is updated in real time on web page without web page reloading

[TC4.9: New Support Widgets for Bars, Barseries, Trend, and Series, on Dashboards and IOT Applications](#) (partially obsolete)

Demo IoT Application exploiting Snap4City Dashboard

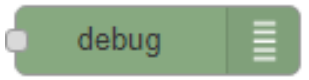
Example of complex IOT Application

In this demo let's create an IoT Application that:

- send random values on Snap4city's Dashboard
- create complex widget based on MyKPI e SURI

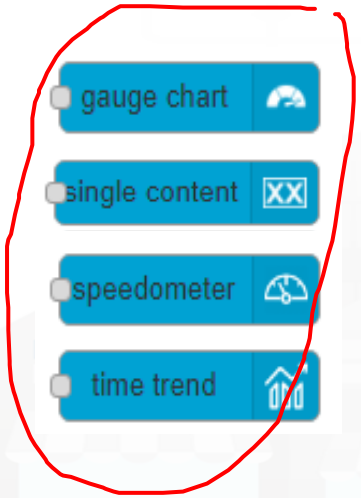
Nodes for flow

Generates an input for the other nodes. It can be repeated at predefined intervals, entered manually and of various types (timestamp, string, number, boolean, JSONetc)



Each message that enters the debug node is shown in the "debug" tab on the right of nodered (you can choose which part of the message to show)

Generates a random number. You can configure the number generation interval and the type (integer or float).

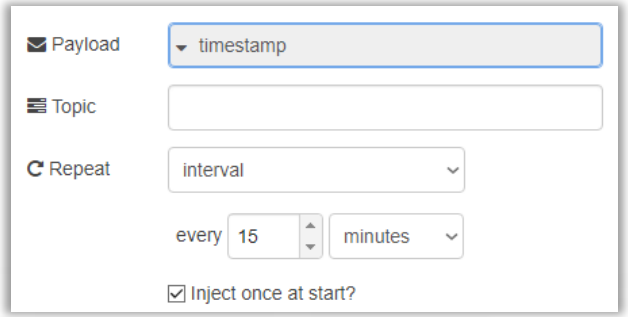


Display values in different modes on a dashboard. The node called single content accepts strings, numbers and html. The others only accept numbers.

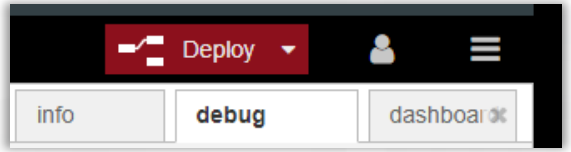
Step 1

- Inject and Debug

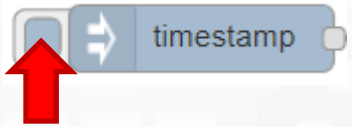
- Connect
- Configure



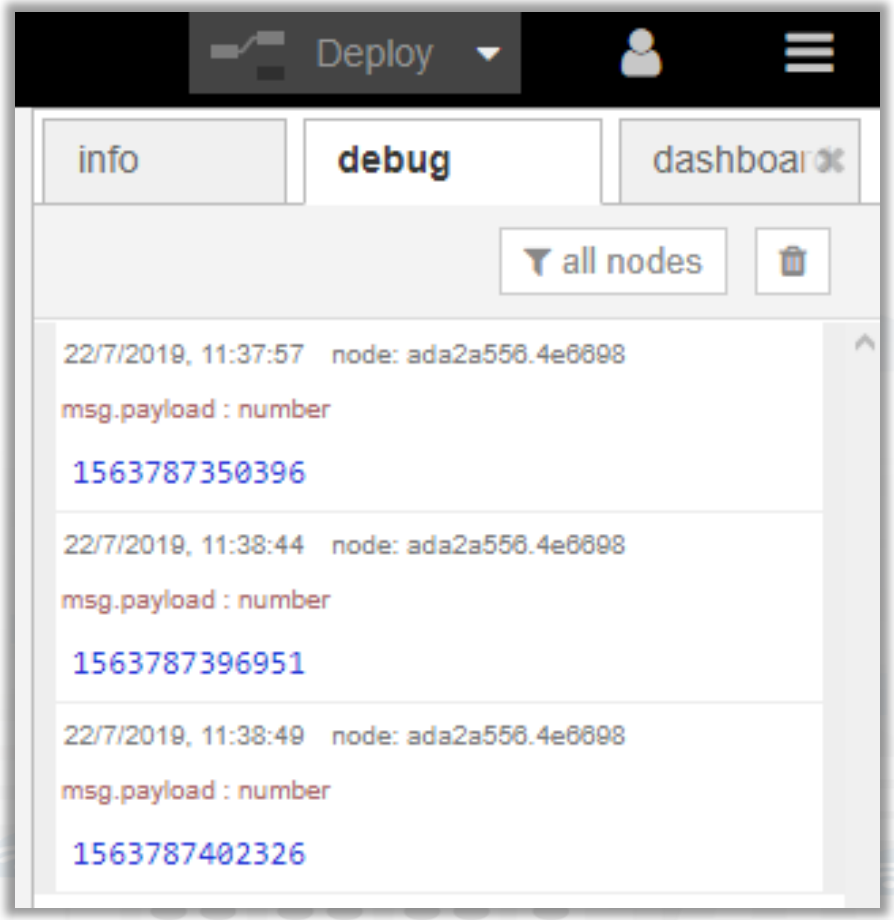
- Deploy



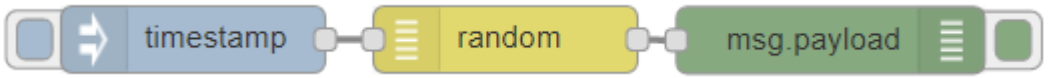
- Click



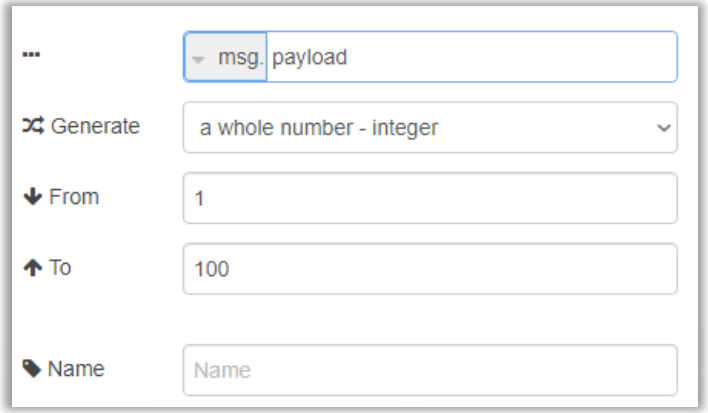
- Observe



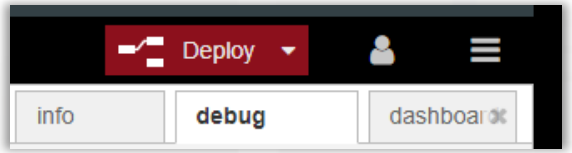
Step 2

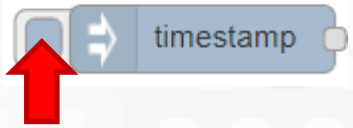


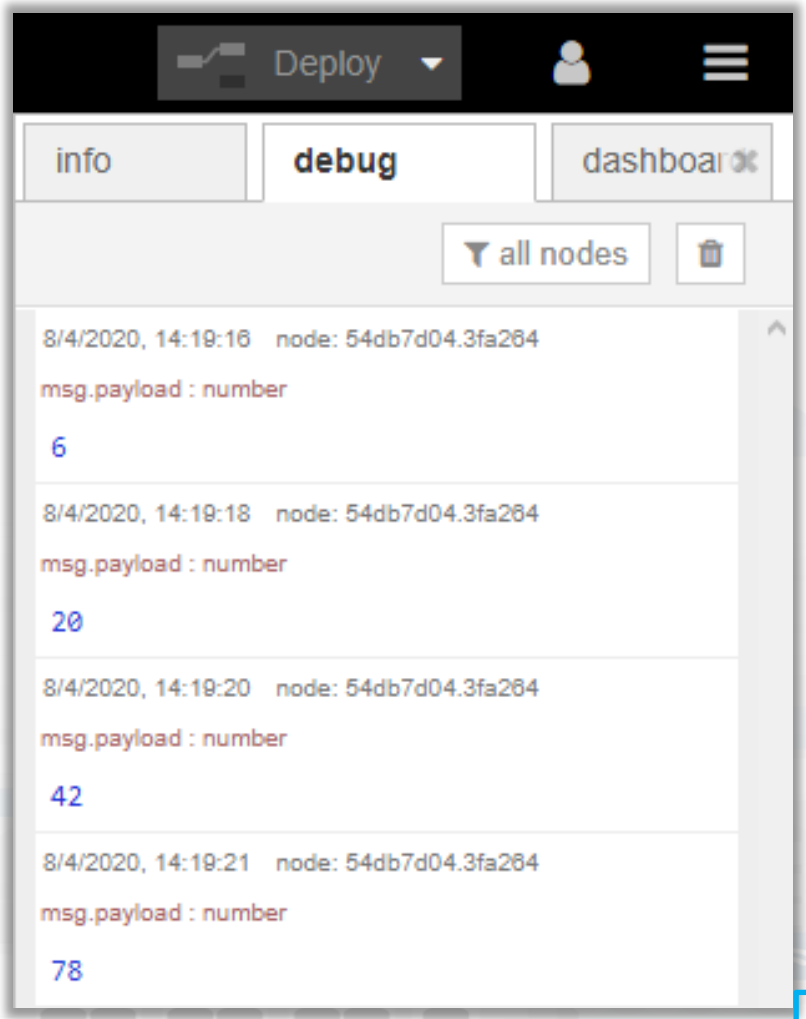
- Random
- Connect
- Configure



- Deploy
- Click
- Observe



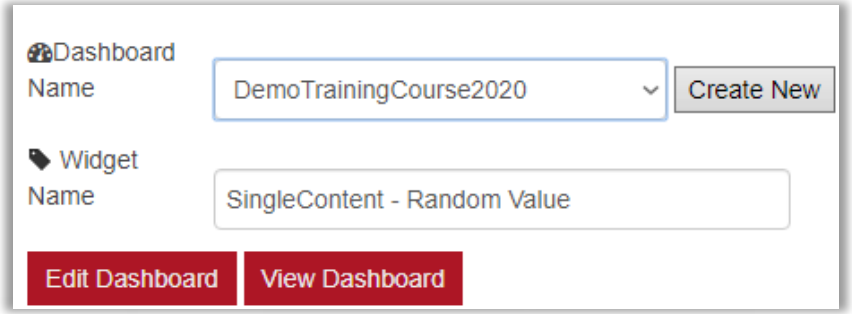




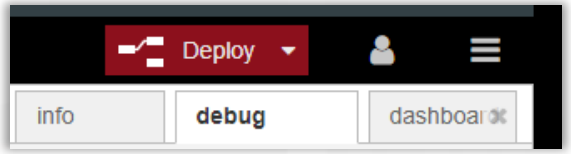
Step 3

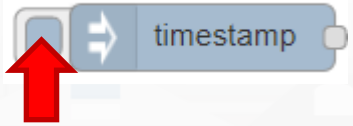
- Single content

- Connect
- Configure

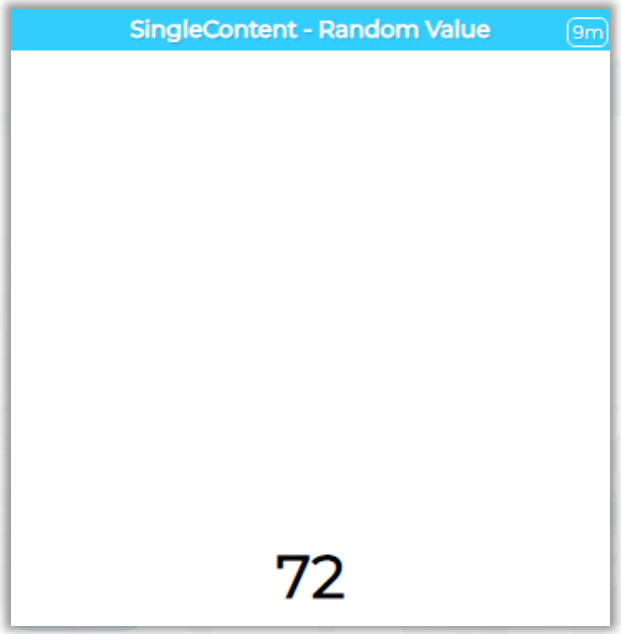


- Deploy

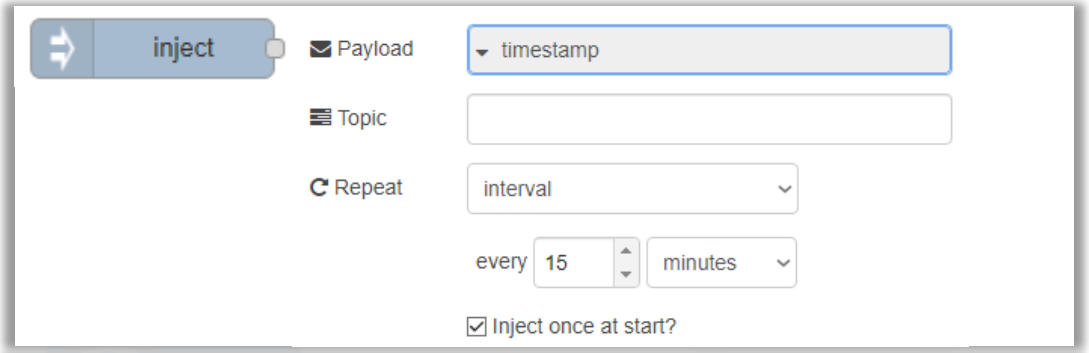


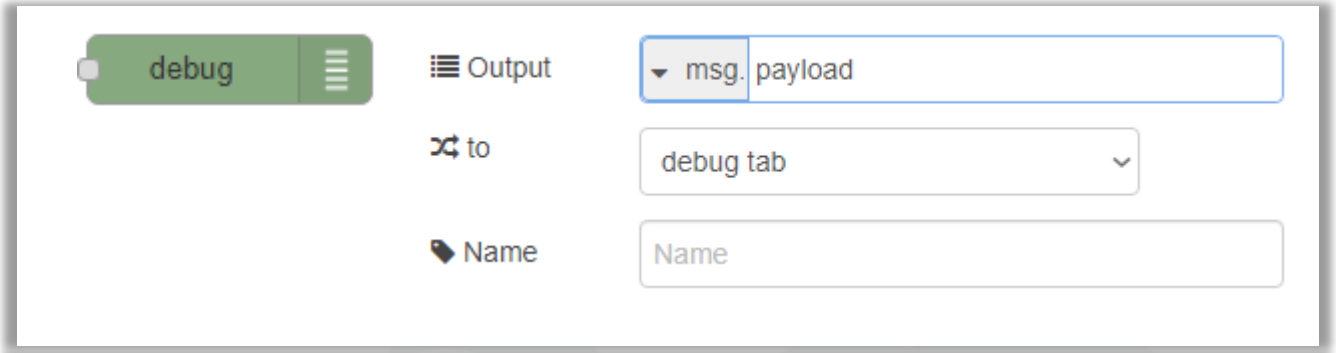
- Click  Click 

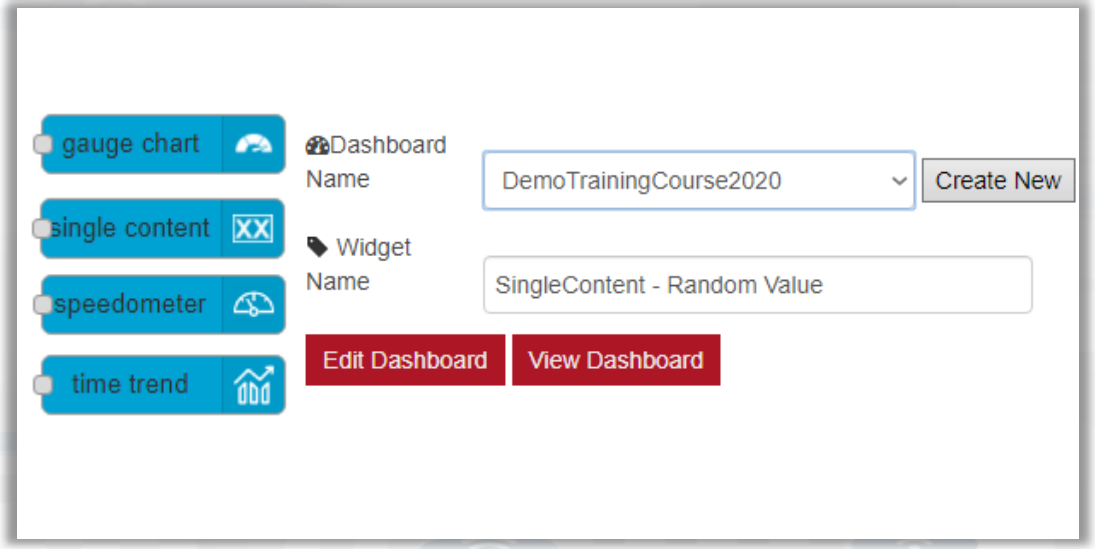
- Observe

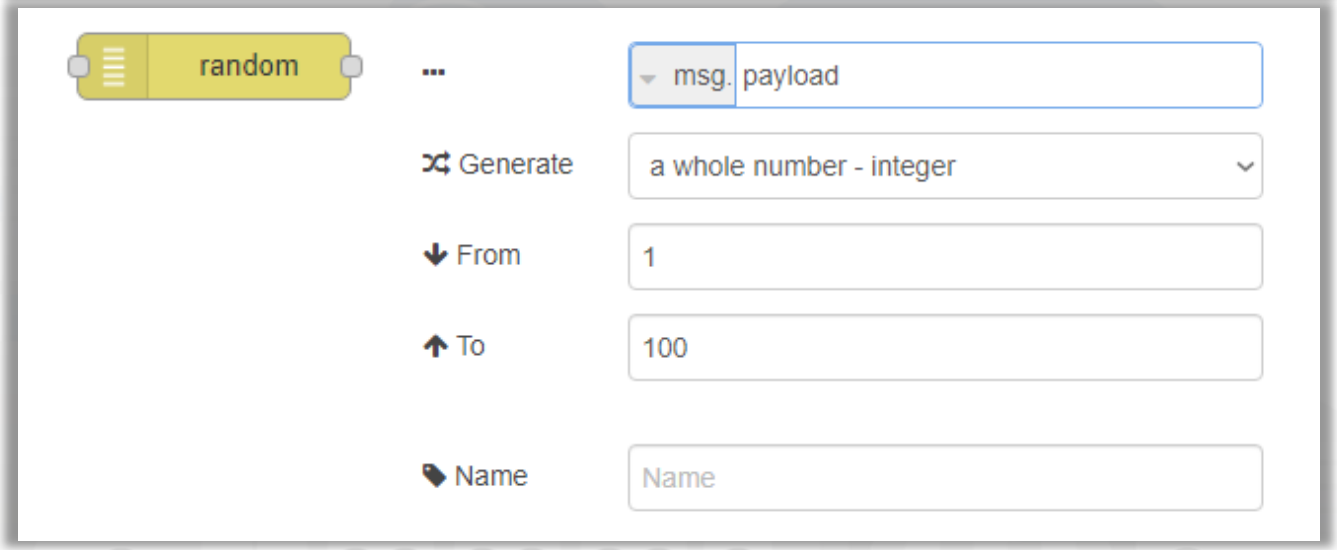


Nodes configuration

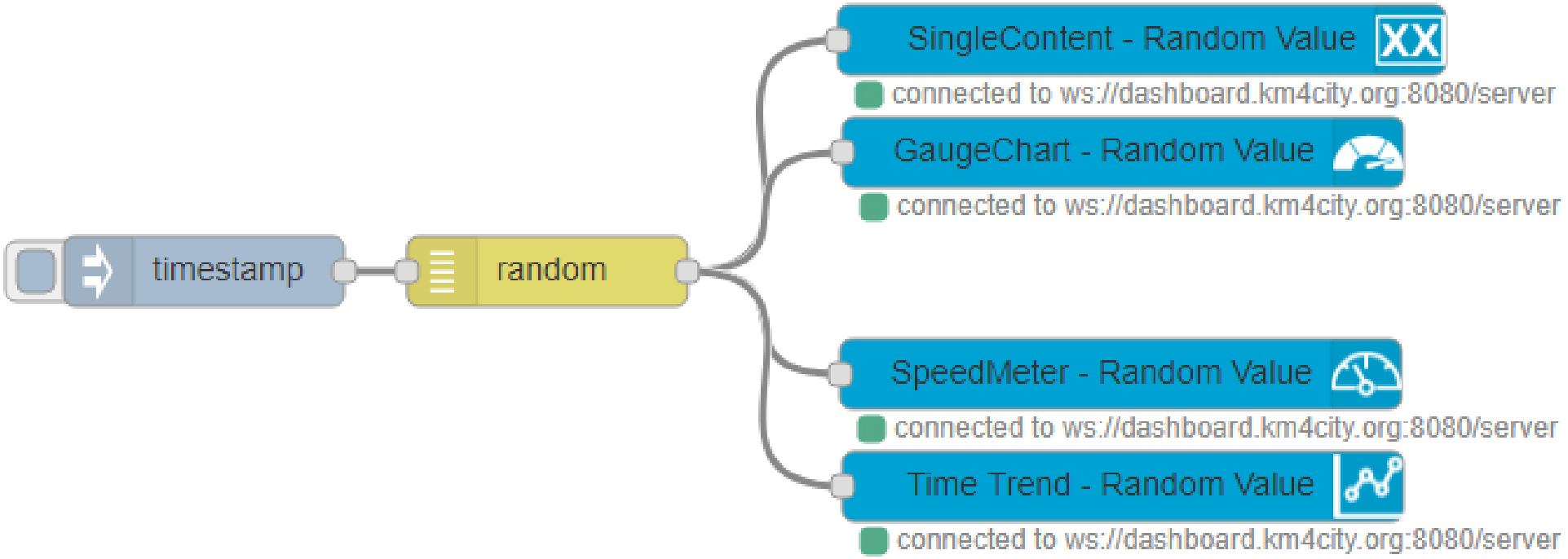








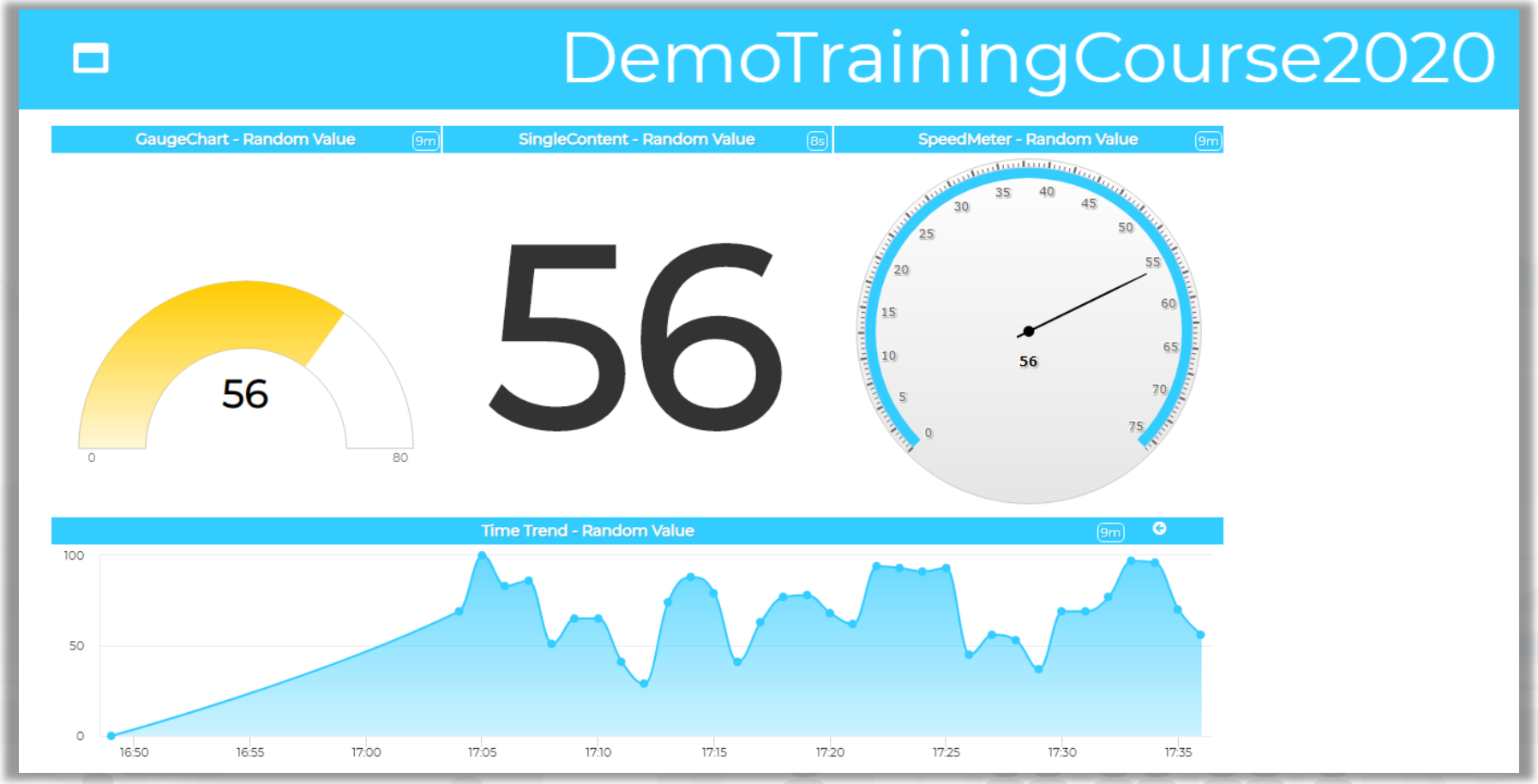
Nodes connections



Explaining: IOT Application Flow

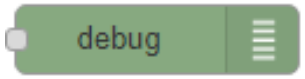
- On Click or Every 15 minutes the ***timestamp*** node sends a message to the ***random*** node.
- When the message arrives, the ***random*** node generates a random number as output message.
- The Number can be sent to Different kinds of nodes to show it on NodeRed Dashboard.

Resulting Dashboard

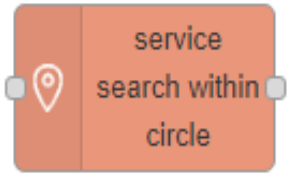


Nodes for flow

Generates an input for the other nodes. It can be repeated at predefined intervals, entered manually and of various types (timestamp, string, number, boolean, JSONetc)

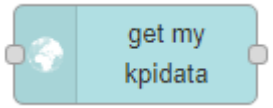


Each message that enters the debug node is shown in the "debug" tab on the right of nodered (you can choose which part of the message to show)



Search in around a certain point of the indicated service. It returns:

- servicesUri of all the services found,
- a GeoJSON containing a minimum of information about the services found, including the coordinates and the name of the service.



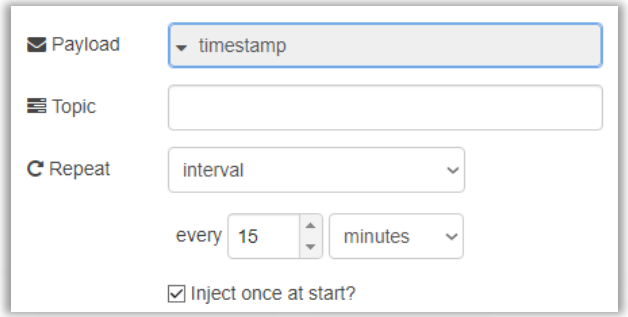
Retrieve the information about My KPIData saved on the Snap4city platform

Display values in different modes on a dashboard. Check info of the node in the Node-RED tab.

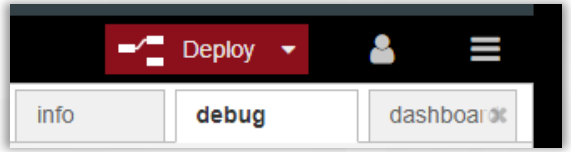
Step 1

- Inject and Debug

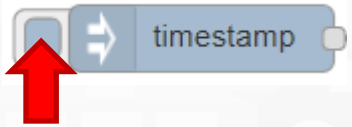
- Connect
- Configure



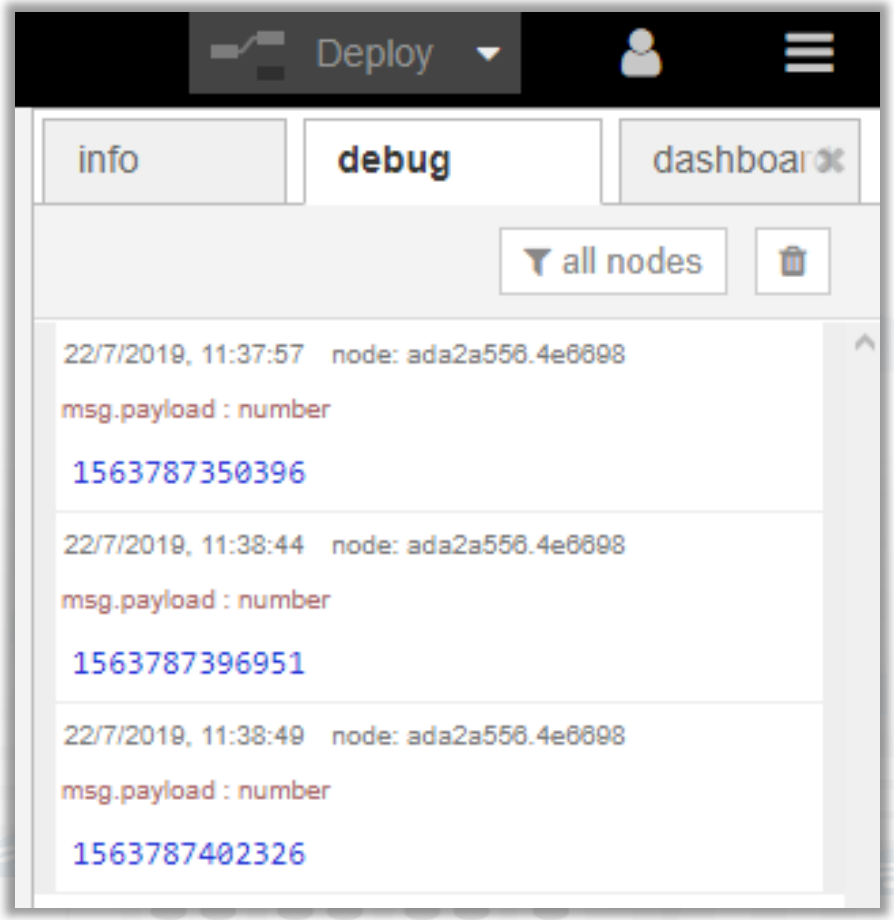
- Deploy



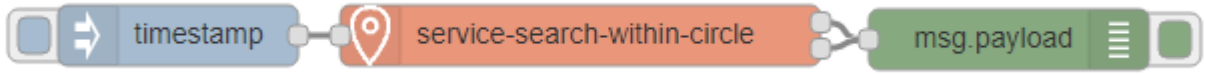
- Click



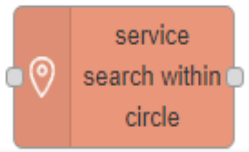
- Observe



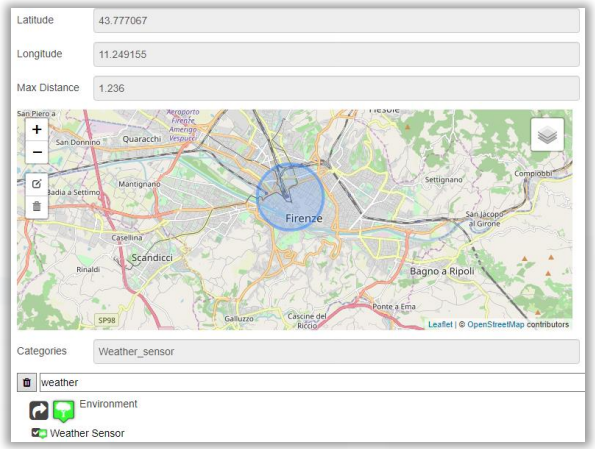
Step 2



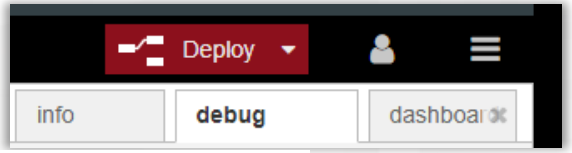
- Service Search Within Circle



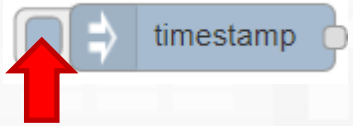
- Connect
- Configure



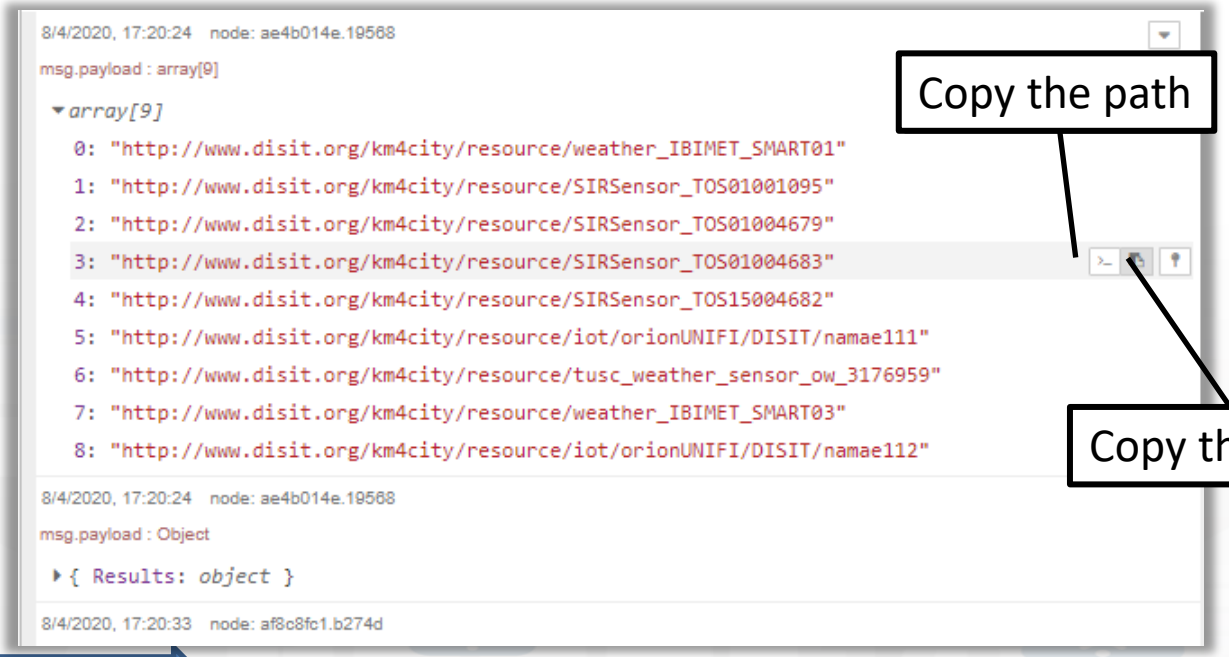
- Deploy



- Click



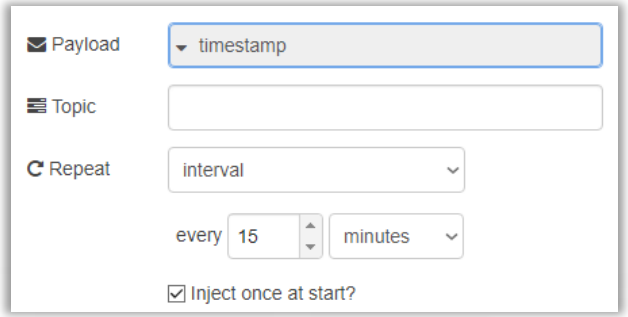
- Observe



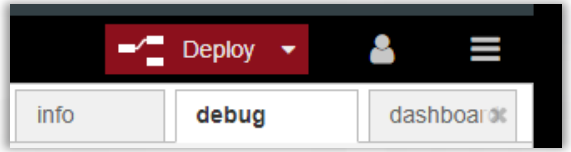
Step 1 Bis

- Inject and Debug

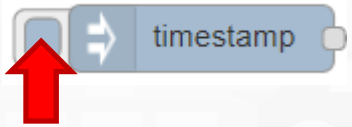
- Connect
- Configure



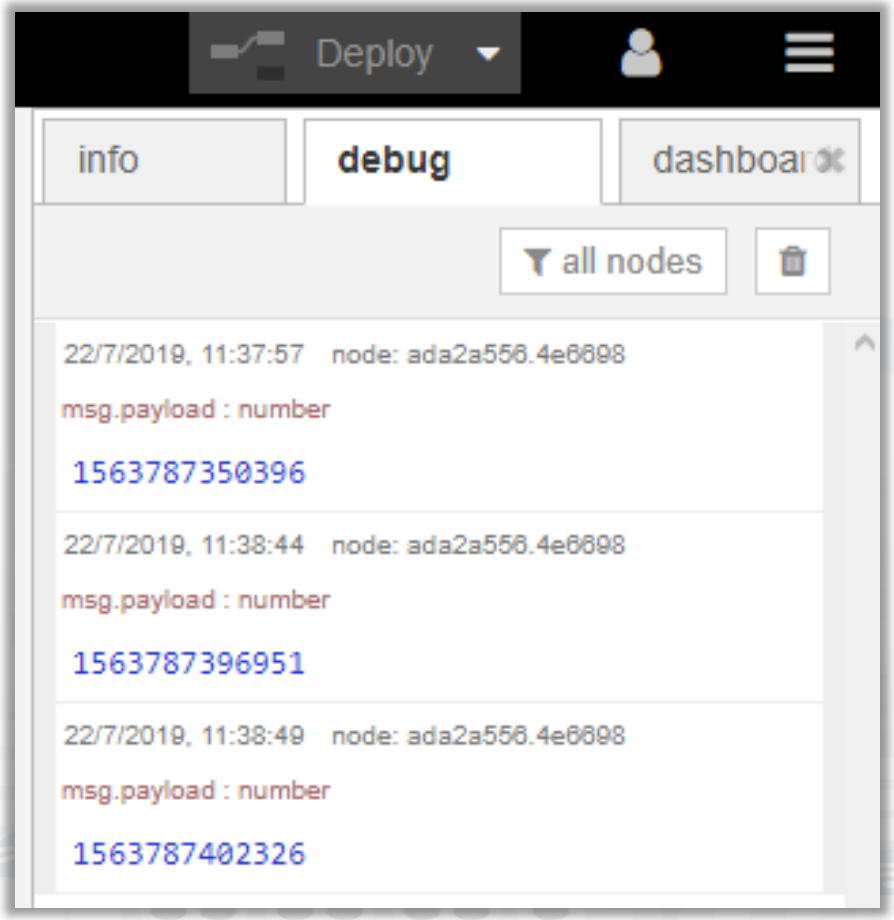
- Deploy



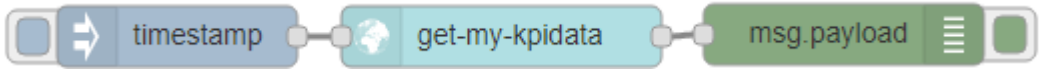
- Click

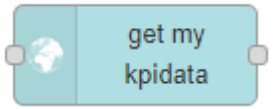


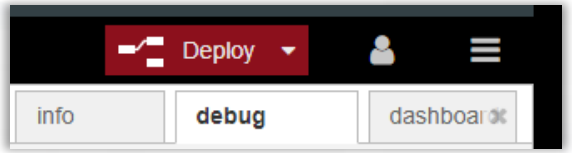
- Observe

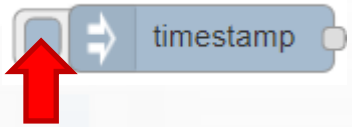


Step 2 Bis



- Get My KPIData 

- Connect
- Deploy 

- Click 

- Observe

Step 3

- Inject Node 

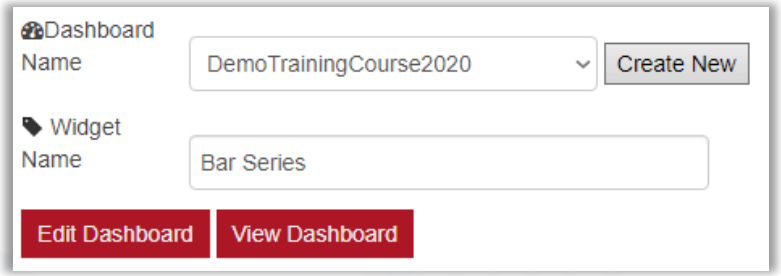
- Configure with data of Weather Sensors and MyKPI retrieved at the previous steps

```
[
    {
        "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
        "metricHighLevelType": "Sensor",
        "metricName": "tusc_weather_sensor_ow_3166540",
        "metricType": "airTemperature"
    },
    {
        "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3182522",
        "metricHighLevelType": "Sensor",
        "metricName": "tusc_weather_sensor_ow_3182522",
        "metricType": "airTemperature"
    },
    {
        "metricId": "17057458",
        "metricHighLevelType": "MyKPI",
        "metricName": "Room 1",
        "metricType": "Temperature"
    },
    {
        "metricId": "17057459",
        "metricHighLevelType": "MyKPI",
        "metricName": "Room 2",
        "metricType": "Room Temperature"
    }
]
```

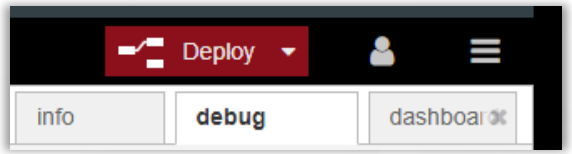
Step 4

- Bar Series

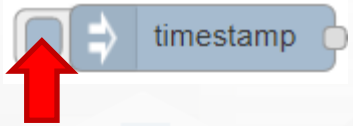
- Connect
- Configure



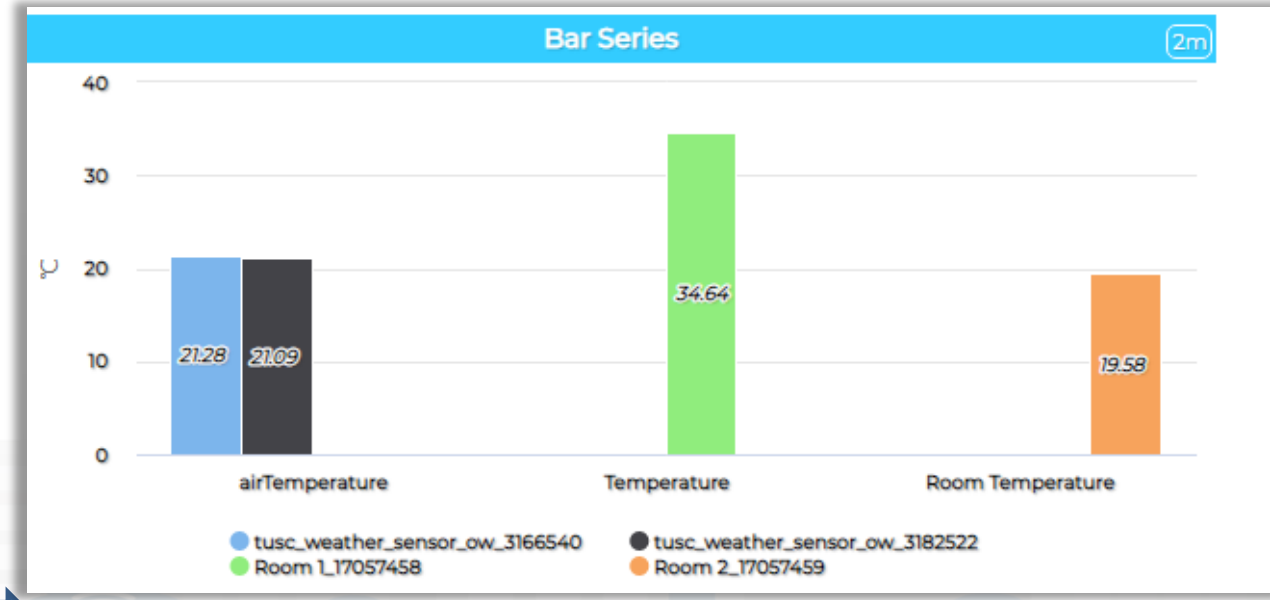
- Deploy



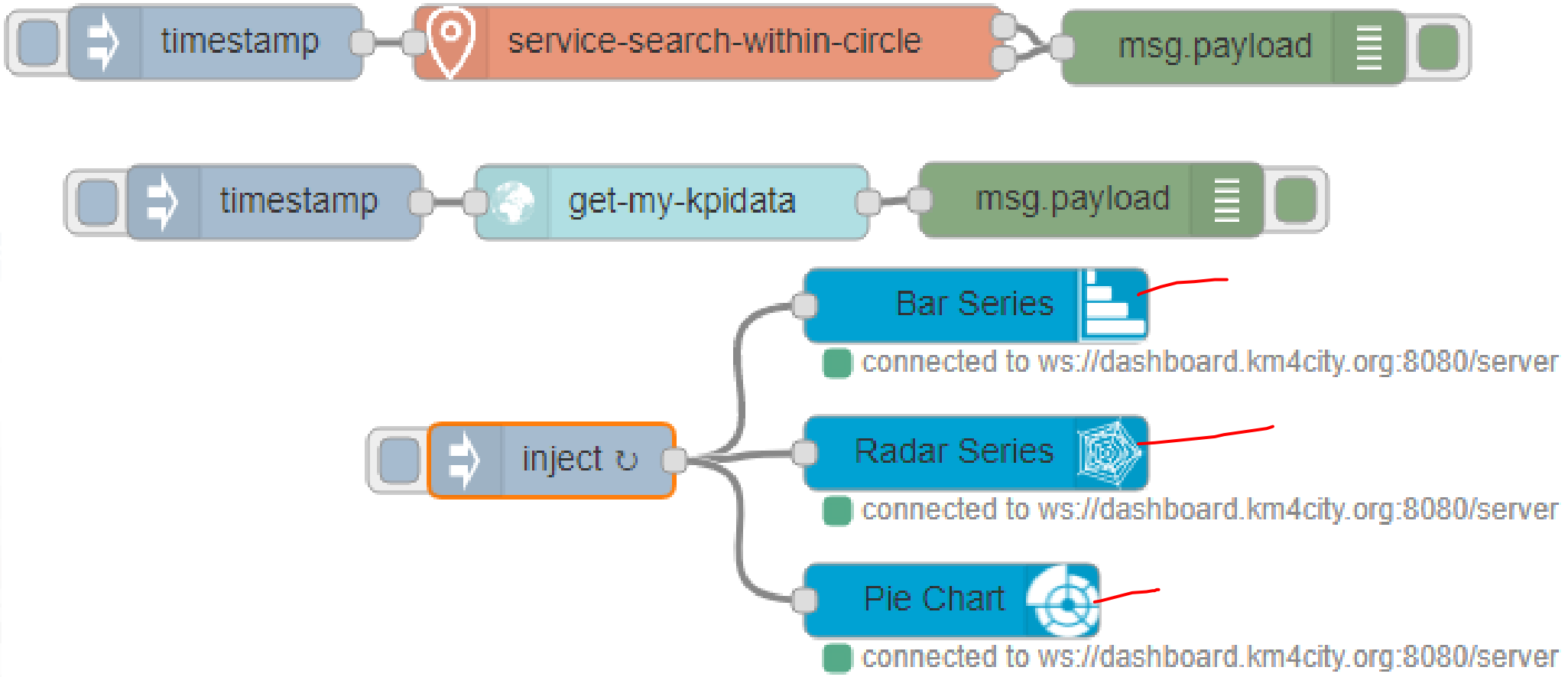
- Click



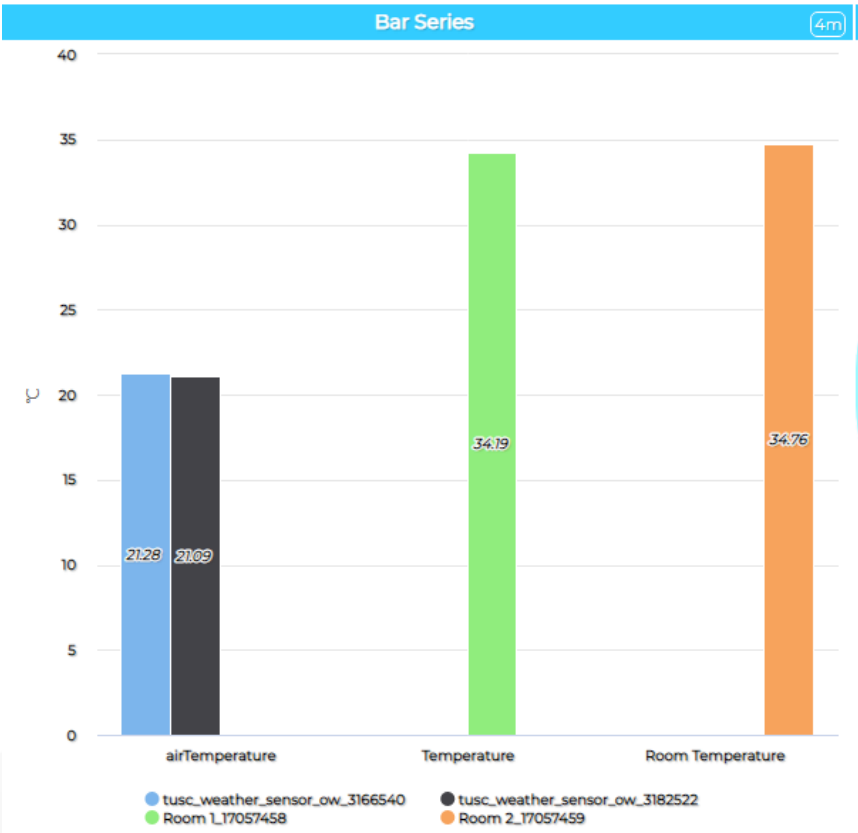
- Observe

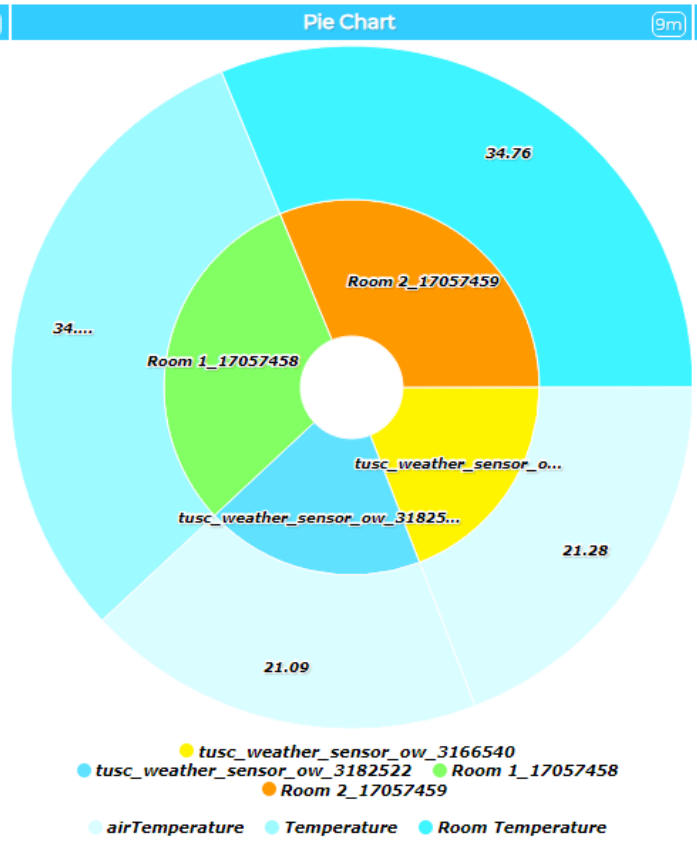


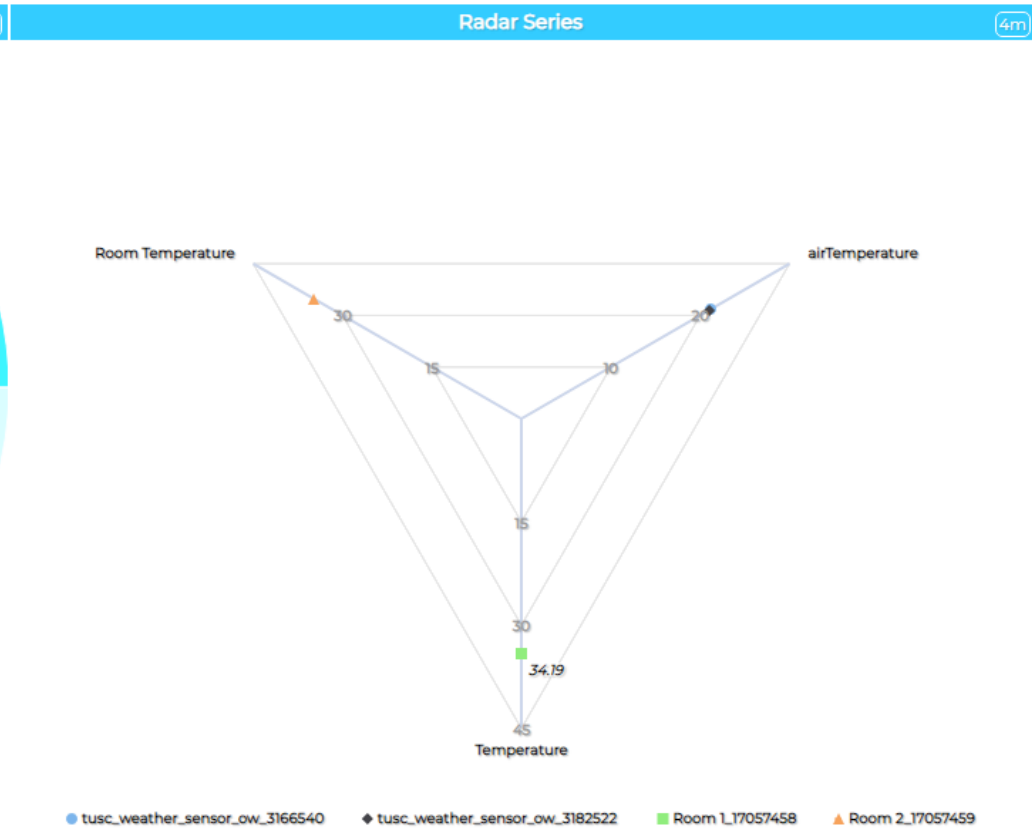
Nodes connections



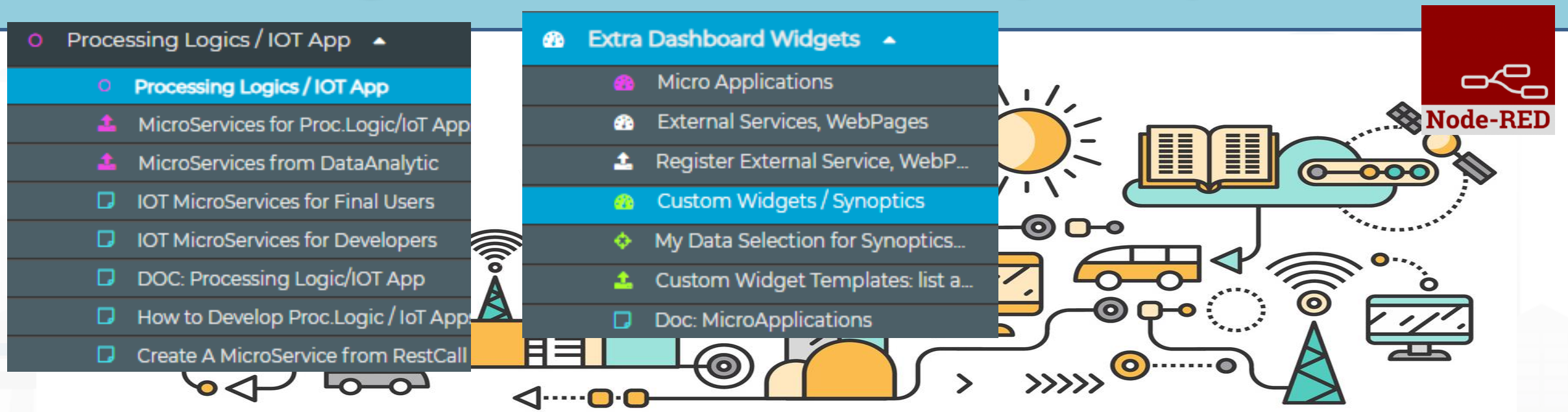
Resulting Dashboard







IoT Application Integration with Synoptics



From-To Custom Widgets / Synoptics to Storage in WS

MyKPI

Sensors

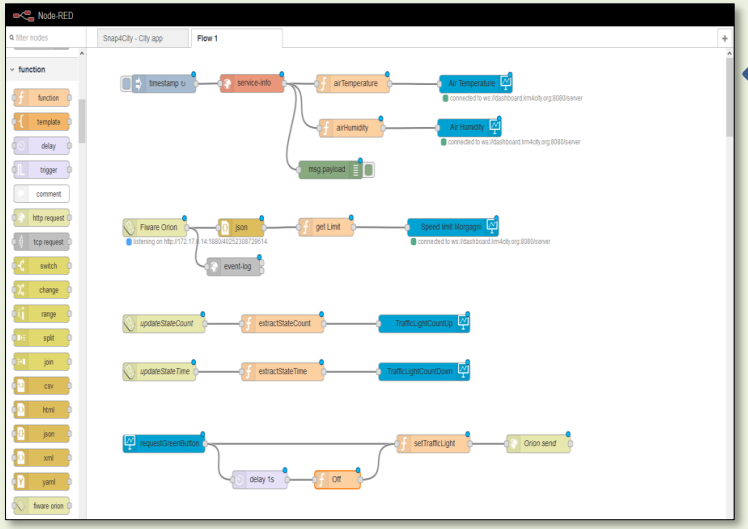
MyKPI

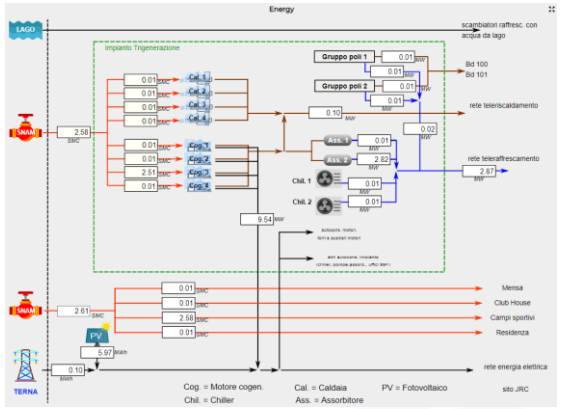
Sensor

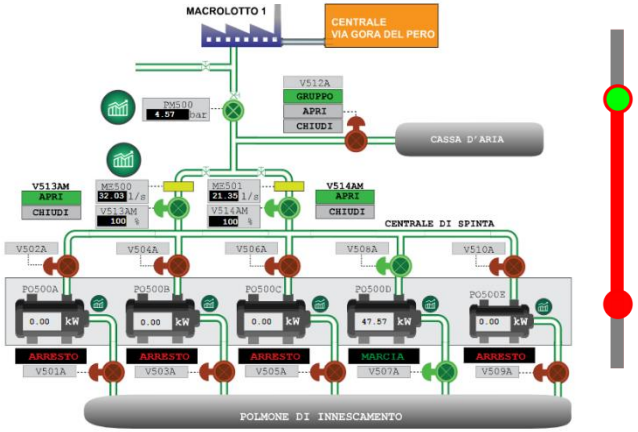
[ ]

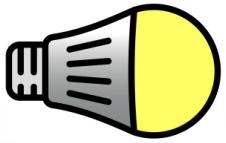
New Shared Variables

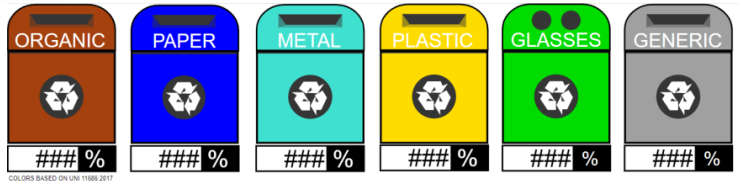
Constant Values

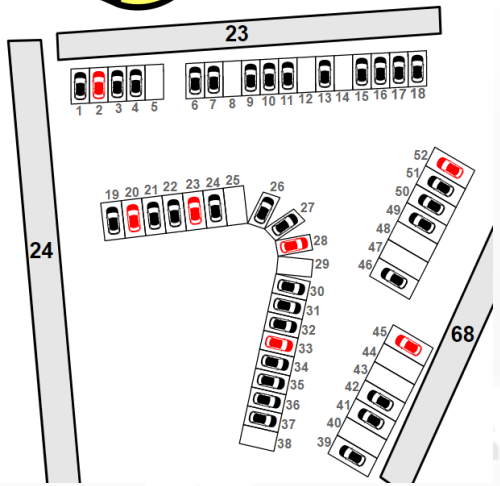


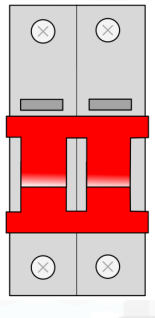


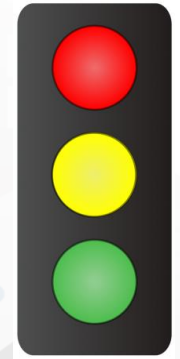






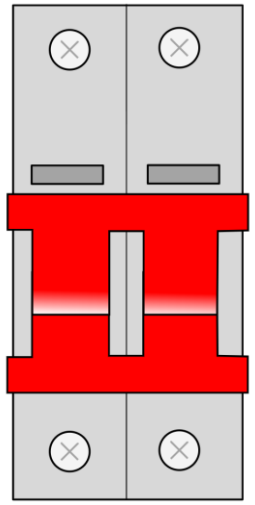




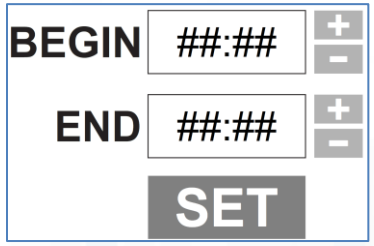


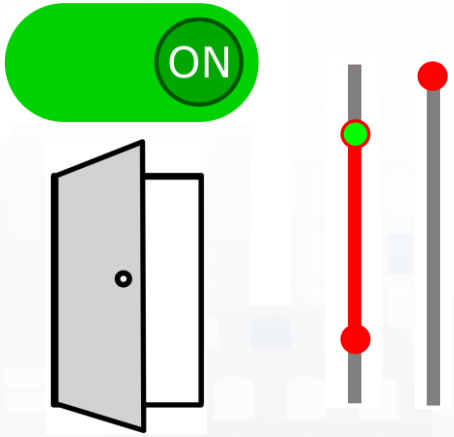
Web Socket Secure

Other examples

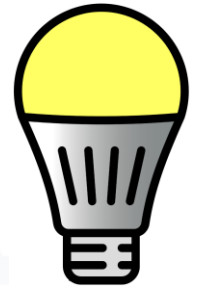


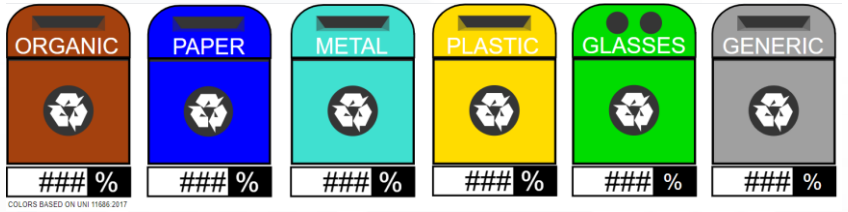
- **Virtual Actuators (sensor-actuator)**
  - From: Dashboard
  - To: IOT App, MyKPI, other Synoptics

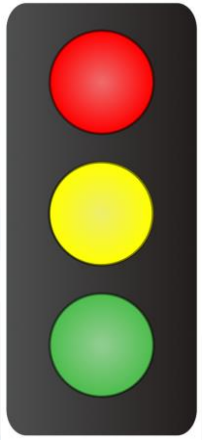


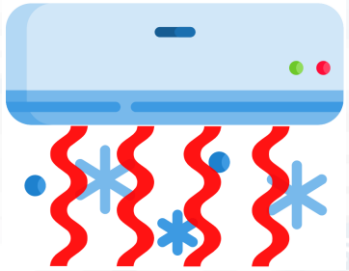


- **Virtual Sensors**
  - From: MyKPI, Sensors, IOT App, other Synoptics
  - To: Dashboards







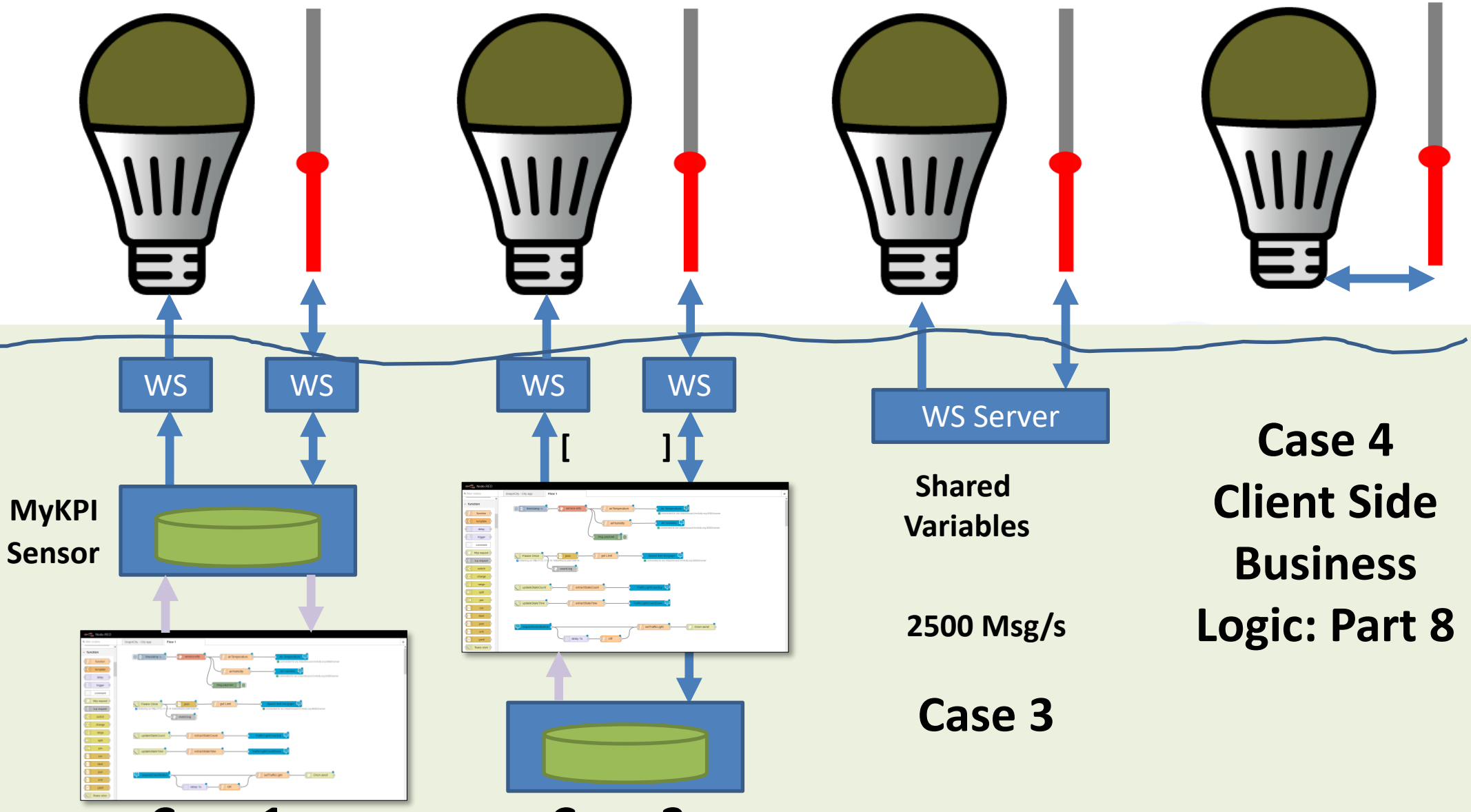


https://www.snap4city.org/663

Dashboard on Browser

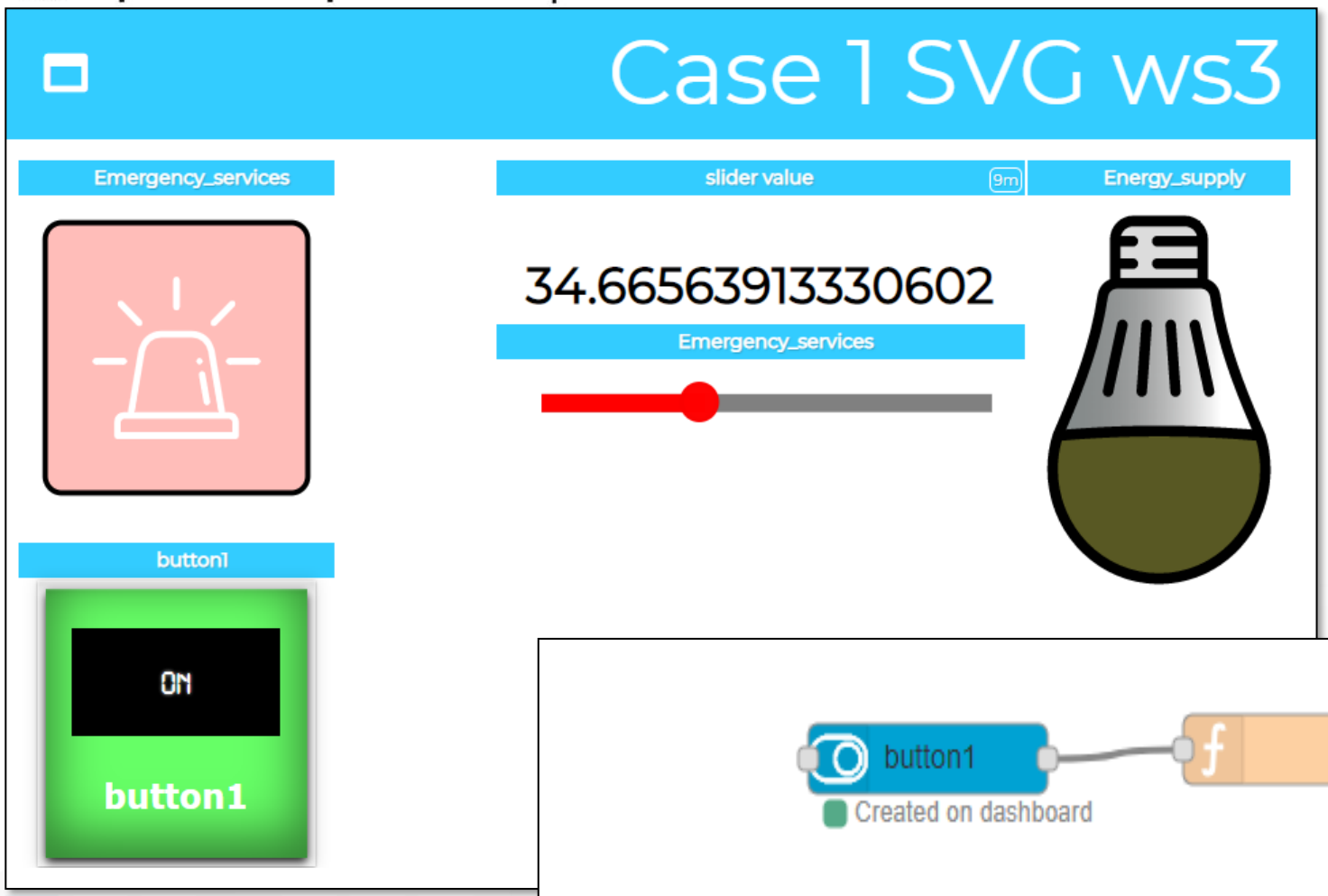
Internet

Storage and IOT App on cloud or on Premise



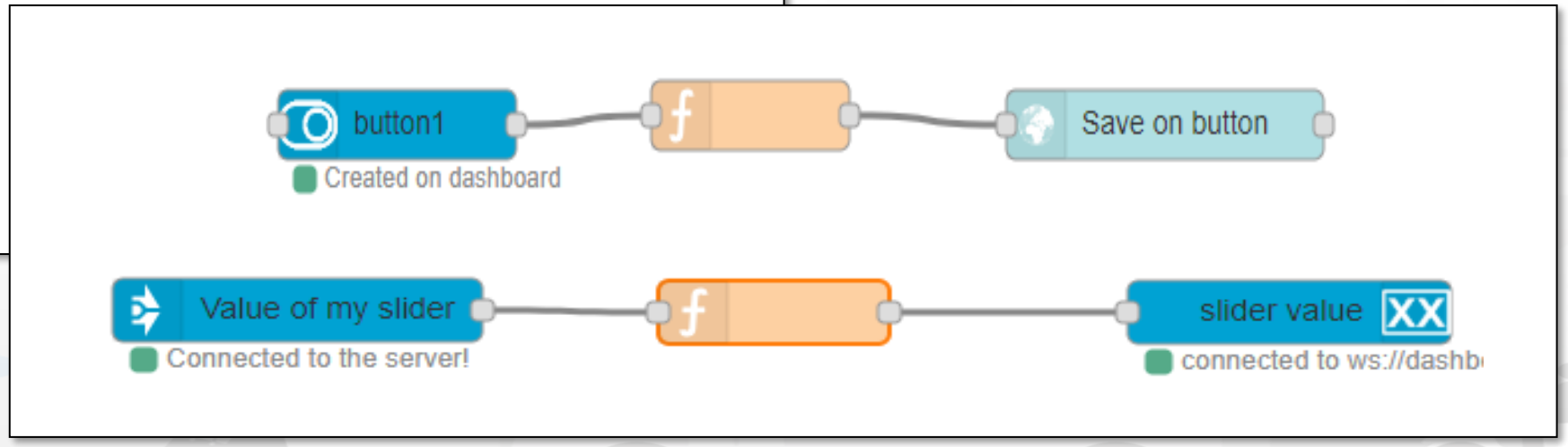
Case 4 Client Side Business Logic: Part 8

Case 1

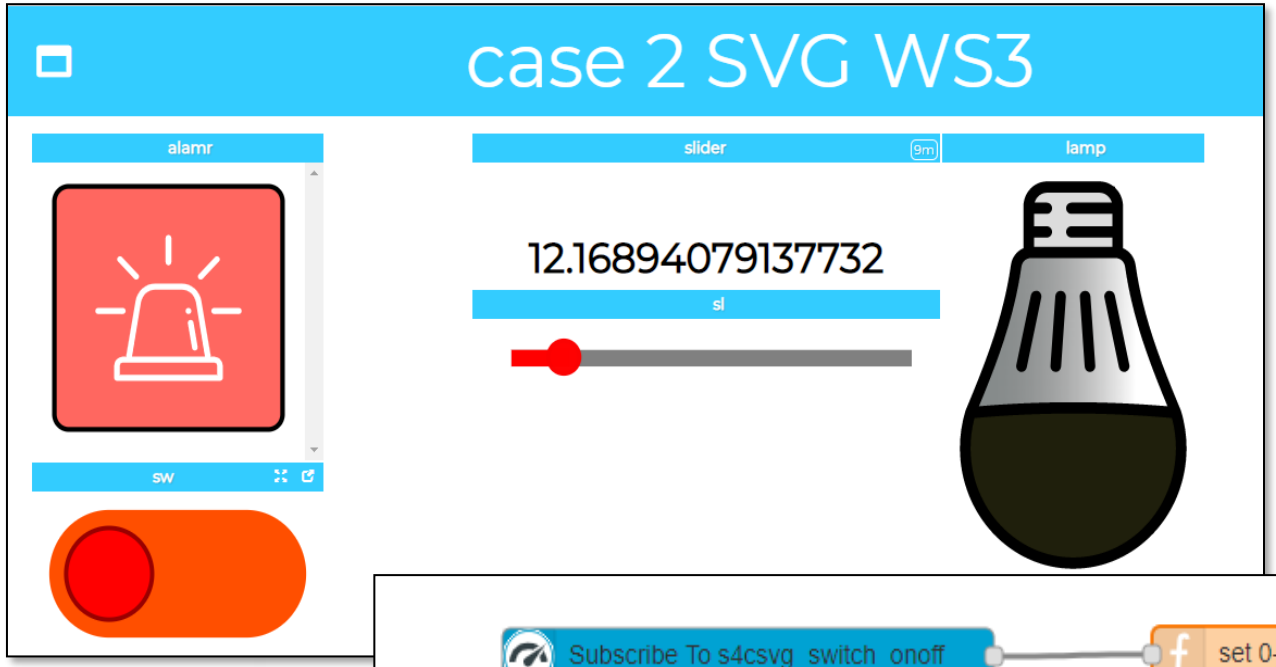


https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=Mjc4NA==

10 WS messages per second

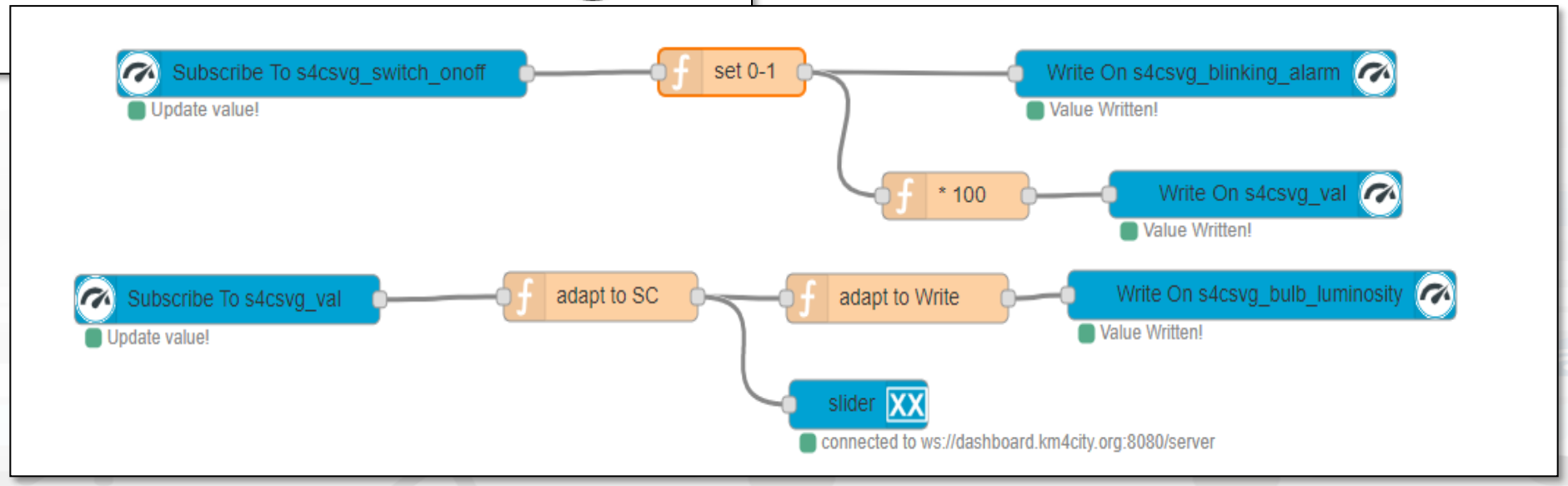


Case 2: Event Driven 100%



https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=Mjc4NQ==

40 messages per second



Read more on

- TC9.19: Custom Widgets / Synoptics controlled by IOT Applications
- Custom Synoptics and Widgets for Dashboards
- Scenario: 5G Enabled Water Cleaning Control
- Snap4Industry: Snap4City for Industry 4.0
- TC1.22: Create and configure a Snap4City SVG Custom Widget for real-time interaction

HTML on Dashboards: the last choice if none of the solutions satisfy your needs

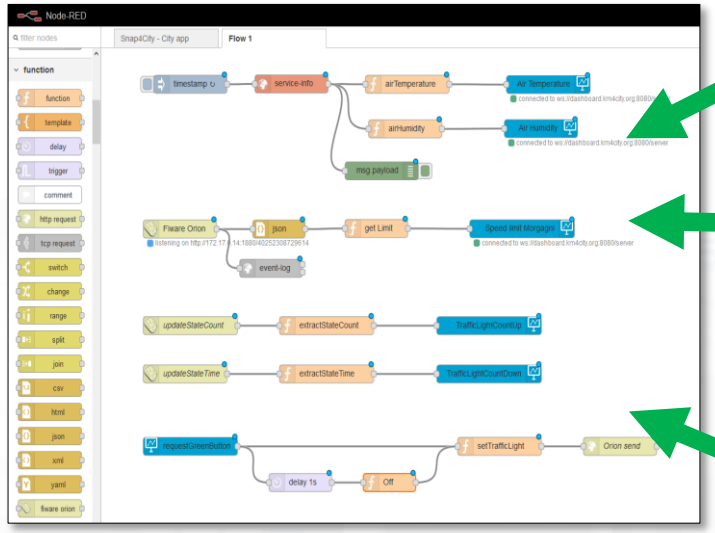
others

- You can send to the Web Content Widget an URL, a link to the web page to be loaded into an External Content Widget

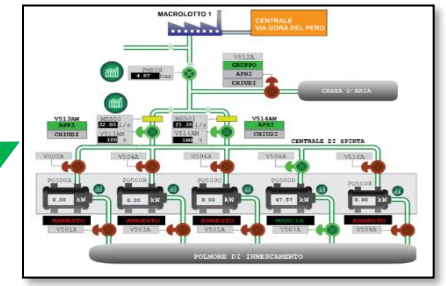
- You can create an HTML page by using the IoT App HTTP nodes
  - and then also send of the WebContent the URL to ask at the dashboard to show the HTML you created ☺

Advanced IOT Applications

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

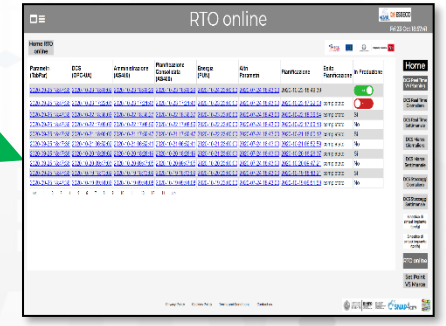


Synoptics
Custom
Widgets



Widgets
Maps
Buttons
Keypads
Controls
..

HTML pages
HTML Forms
Tables
..

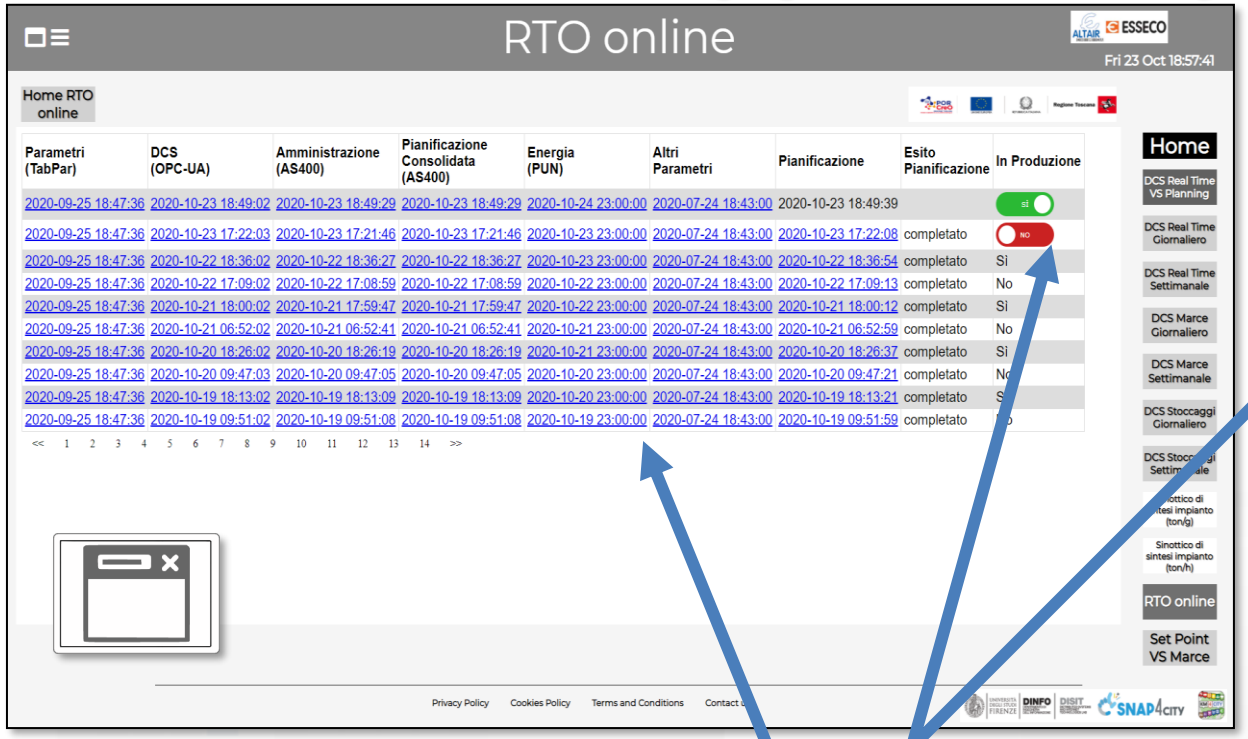


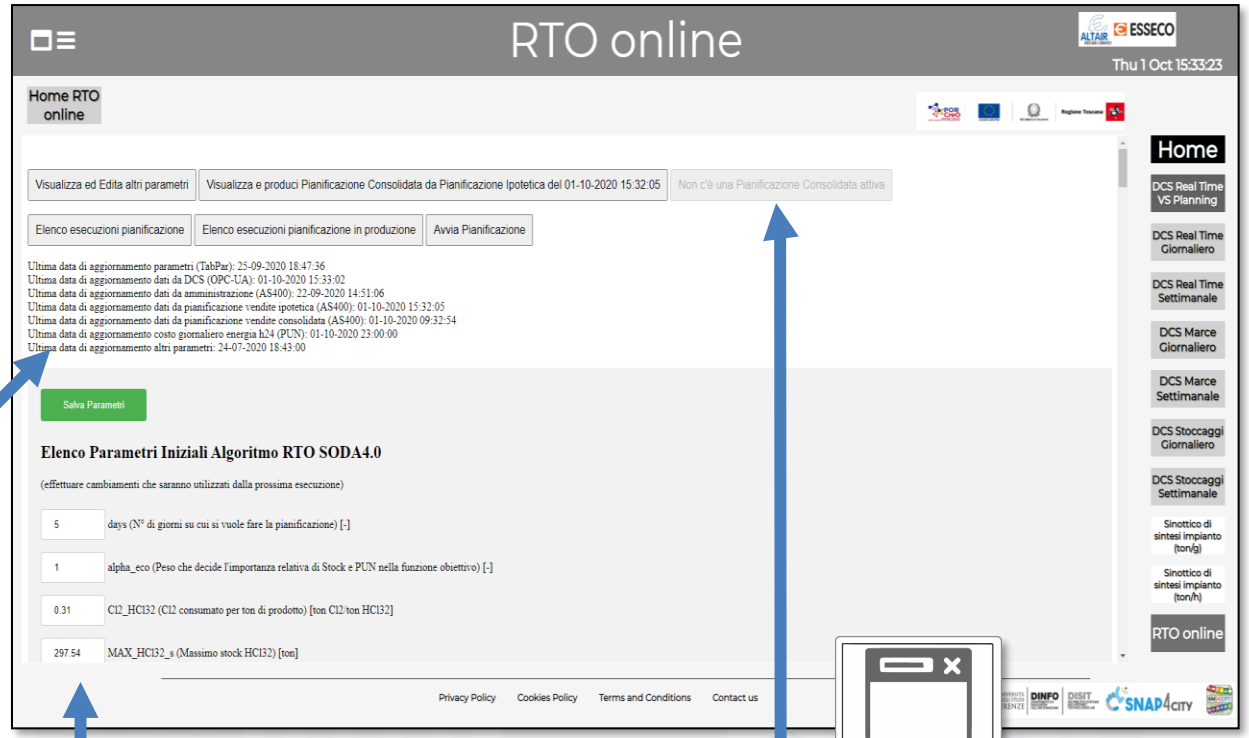
HTML

https://www.snap4city.org/394

https://www.snap4city.org/596

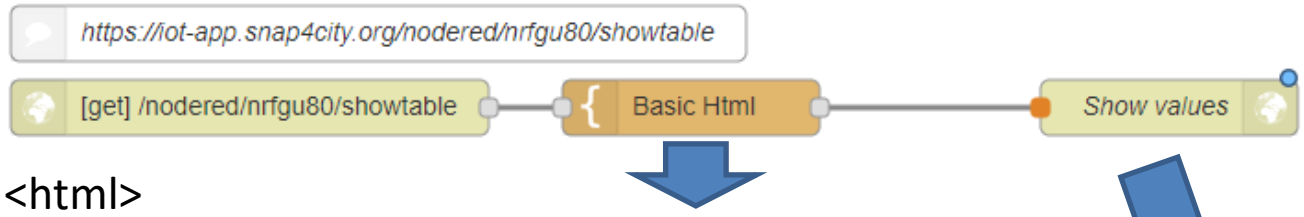
IOT App with Dynamic Web Pages



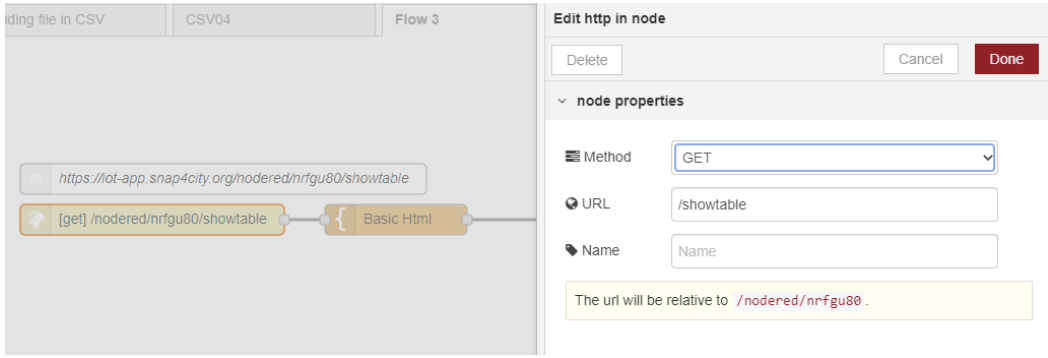


- **HTML pages** can
  - be dynamically generated from the IOT App
  - provide forms to produce data to the IOT App, also including interactive elements
  - collect file from users, and produce files to web and to the system
  - have CSS and AJAX controls

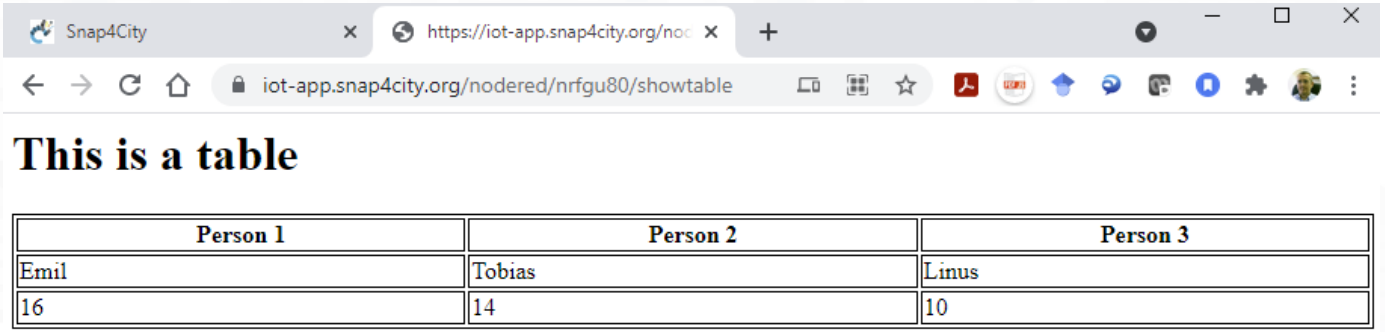
From IoT App to HTML pages




```
<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



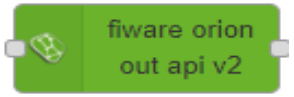
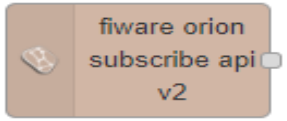
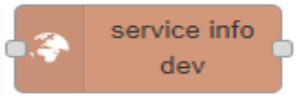
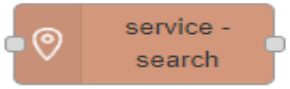
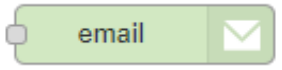
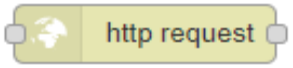
TOP

Proc.Logic / IoT App Programming Pattners

- IOT Applications
  - IOT Applications
  - MicroServices for IOT Applications
  - MicroServices from DataAnalytic
  - IOT MicroServices for Final Users
  - IOT MicroServices for Developers
  - Doc: IOT Applications
  - How to Develop IOT Applications
  - Create A MicroService from RestCal...

Node-RED

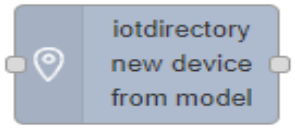
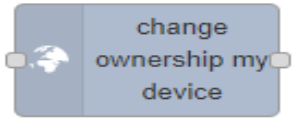
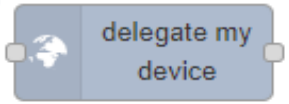
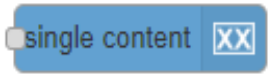
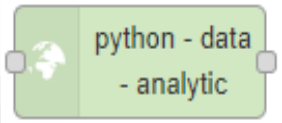
examples

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

Part 5

Part 5

examples

Node shape	Description	Snap4City or standard
	A block which is printing on debug view the data JSON passed in its input. Please note that the node can be tuned to provide only msg.payload or the full JSON message.	standard
	To create a Entity Instance (device instance) from a model prepared on Entity Directory (IoT Directory).	Snap4city
	To change the ownership of an Entity Instance (IoT Device).	Snap4city
	To delegate a certain Entity Instance (IoT Device) to some other user for which you have to know the Nickname. Delegations can be: Read_access, Read_write, Modify (to modify the Entity Instance structure).	Snap4city
	To show something on Snap4City dashboard with a simple widget. A large set of dashboard nodes to send and retrieve data to/from dashboards. This specific Nodes allows to send on dashboard HTML formatted messages with some limitations. Full HTTP widget is also accessible.	Snap4city
	MQTT broker listener, to receive messages from the Broker. Another similar node can be used to send MQTT messages to the MQTT broker. This node allows to perform a subscription to a topic of the MQTT broker.	standard
	Request performed on a Container including a Python data analytics, which is loaded into the node and the container is created at the first Deploy of the Processing Logic. Similar Approach is performed for RStudio Data Analytics	Snap4city

Part 5

Part 5

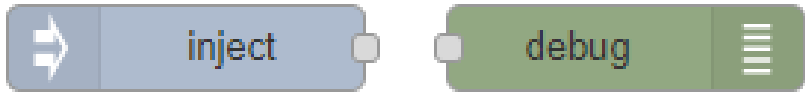
Part 5

Part 5

Part 4

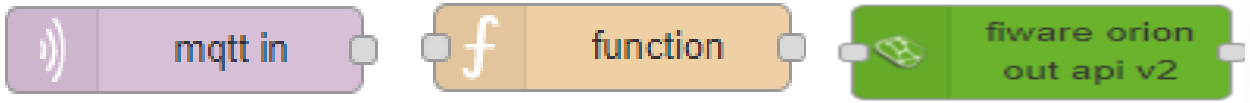
Some patterns

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.

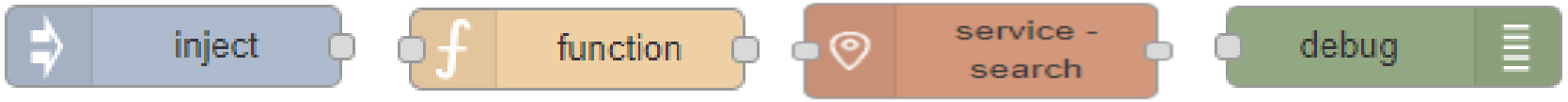
3) Event data reception from an MQTT broker, transformation and send it to the storage pushing data into the Orion Broker V2.



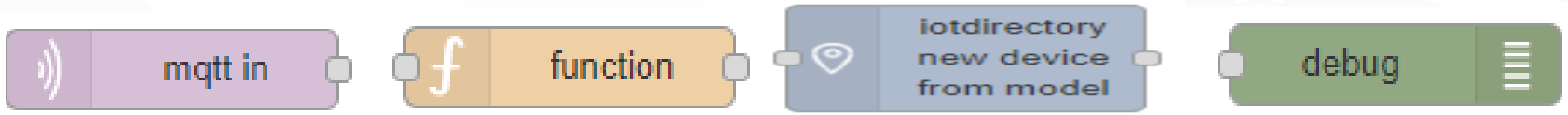
Part 5

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

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

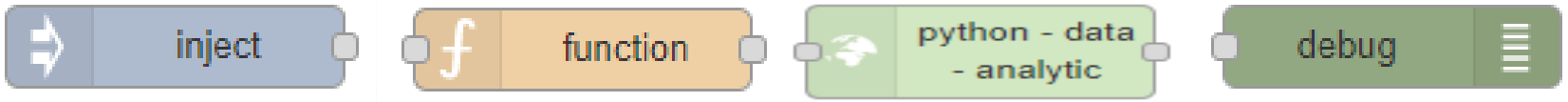


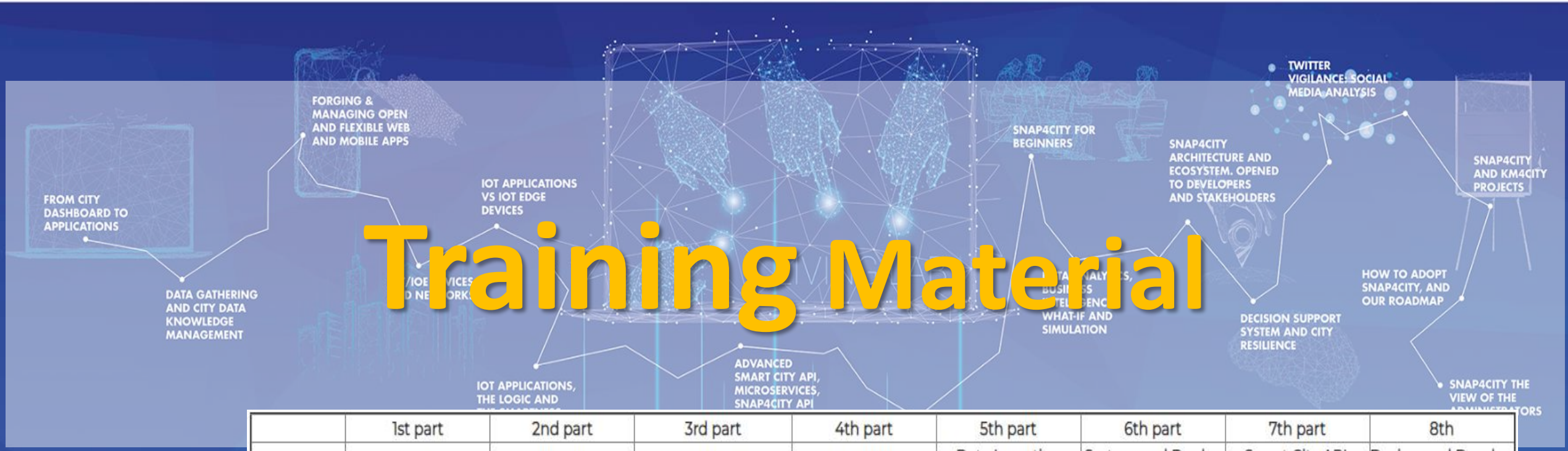
2) Event data reception from an MQTT broker, transformation to create an Entity Instance from a known Entity Model, debug to see eventual errors, for example if the device is already present (to avoid production of error, one may verify if the Entity Instance is already present by posing a query on the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



Part 5

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





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								

In Part 5 and 6 we have

- Broker → Storage
- IoT App / Proc.Logic → Broker
- Broker → IoT App / Proc.Logic

- IOT Network Management and Control
- IOT Devices hardware-software integration
- Using Data Models:
  - FIWARE Smart Data Models, Snap4City IoT Device Models
- Data Exchange and Distributed,
  - computing on multiple Snap4City Domains
- IOT end-2-end Secure Stack, IOT ←→ Dashboards

Any other Request ?

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

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

Dashboards (Public)

My Snap4City.org | Tour Again

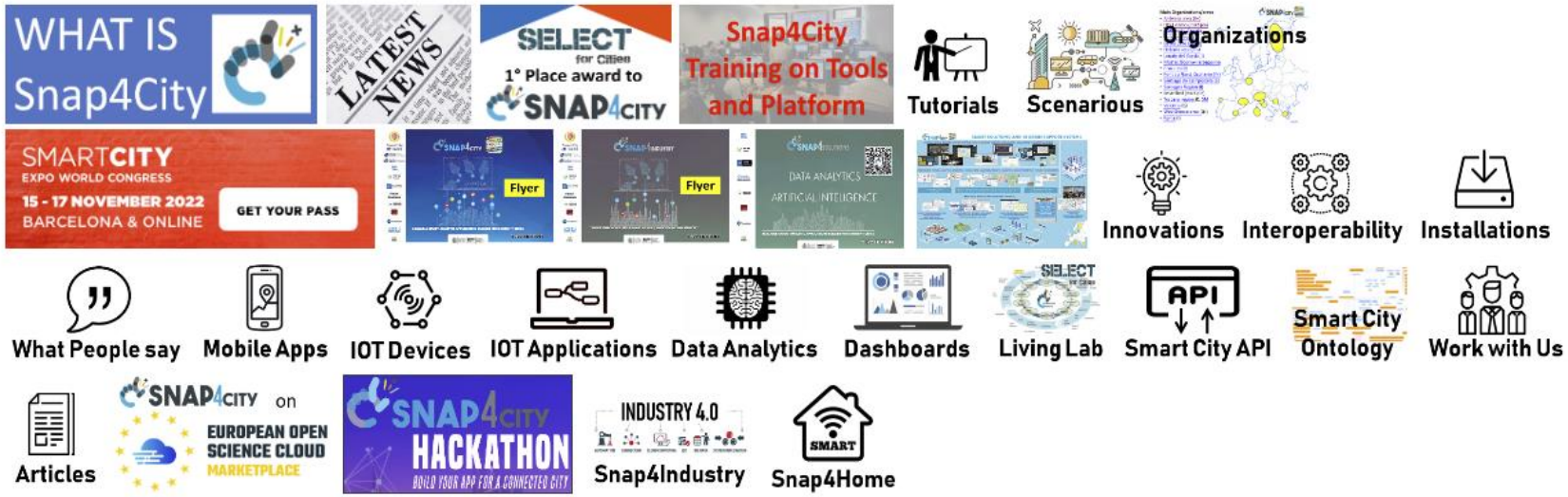
www.snap4solutions.org
Dashboards of My Organization
My Dashboards in My Organization
My Data Dashboard Dev Kibana
Extra Dashboard Widgets
Data Management, HLT
Knowledge and Maps
Processing Logics / IOT App
Entity Directory and Devices
Resource Manager
Development Tools
Management
Decision Support Systems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

Privacy Policy | Cookies Policy

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

Snap4City: Smart aNalytic APp builder for sentient Cities and IOT

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



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

Username: paolo.disit

Search

Search

-Any-

Powered by www.km4city.org

FIWARE | Node-RED

Sii-Mobility

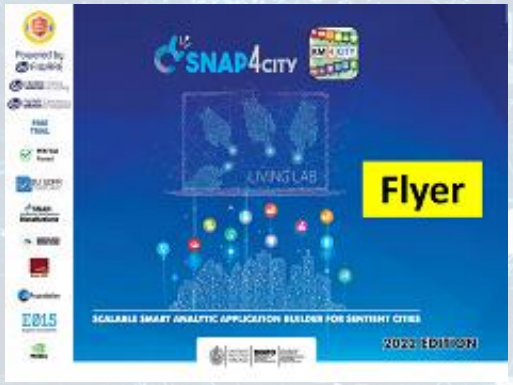
Organization Groups

DISIT
- Developer
- Operativo

Updates on

2022 booklets

- Snap4City



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

- Snap4Industry

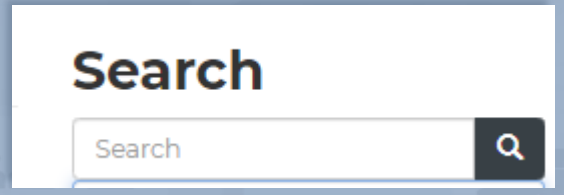
https://www.snap4city.org/download/video/DPL_SNAP4INDUSTRY_2022-v03.pdf

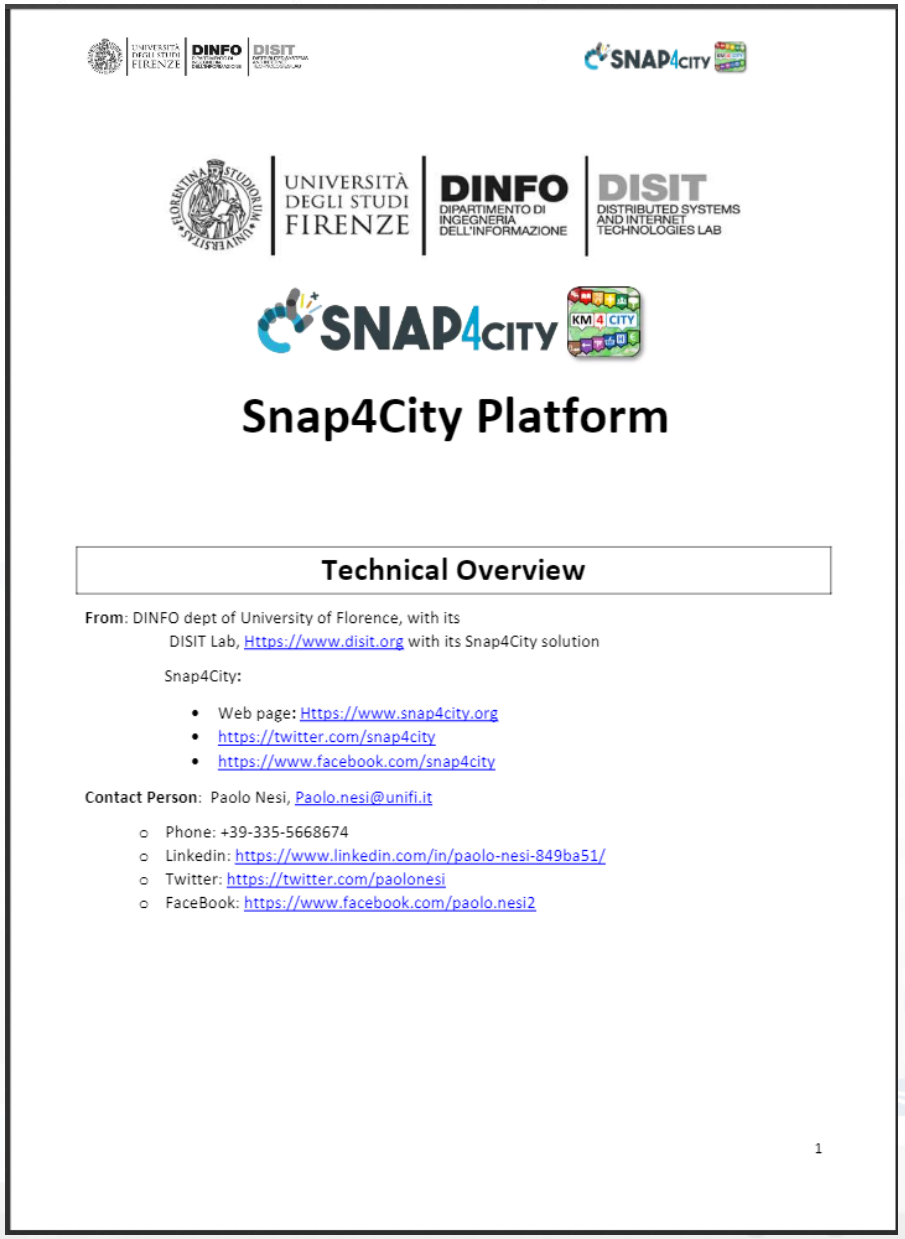
- Solutions
- Data Analytics

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

The Platform

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





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

SNAP4CITY KM4CITY

Snap4City Platform

Technical Overview

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

Snap4City:

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

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

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

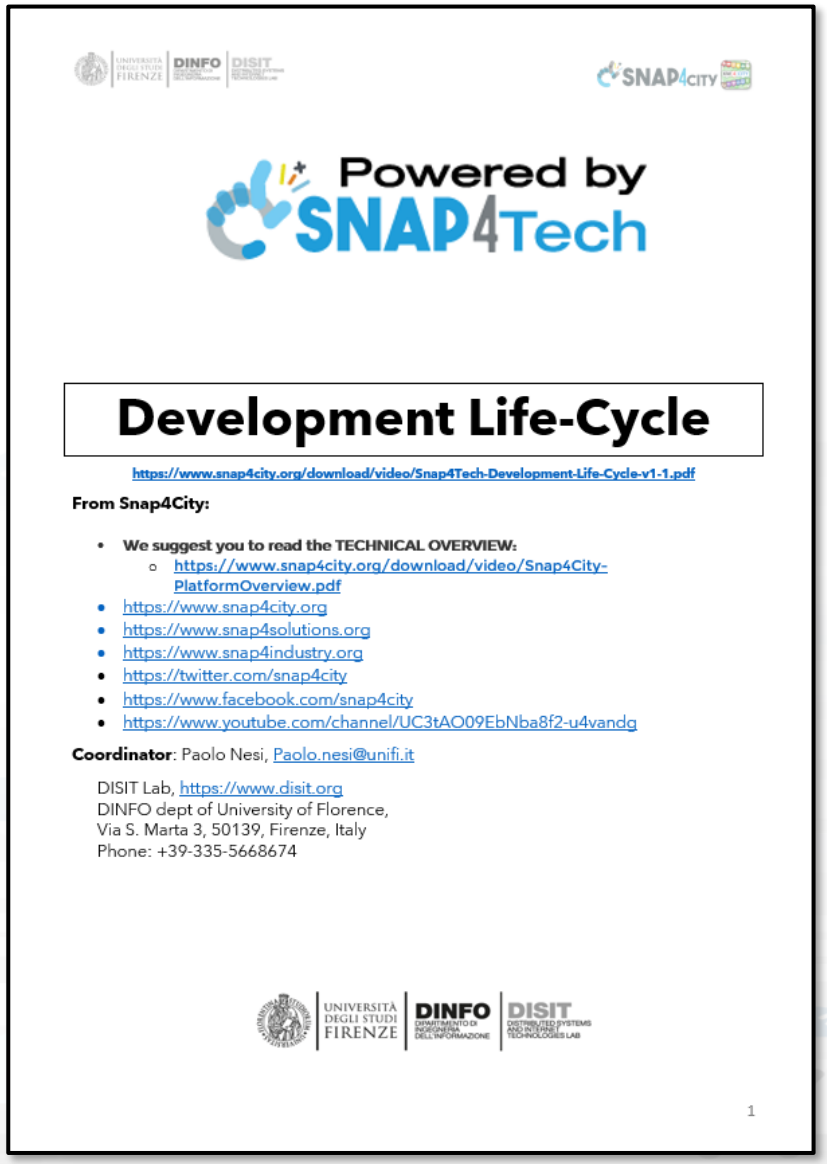
1

Tech Overview

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

Development

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



Powered by SNAP4Tech

Development Life-Cycle

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

From Snap4City:

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

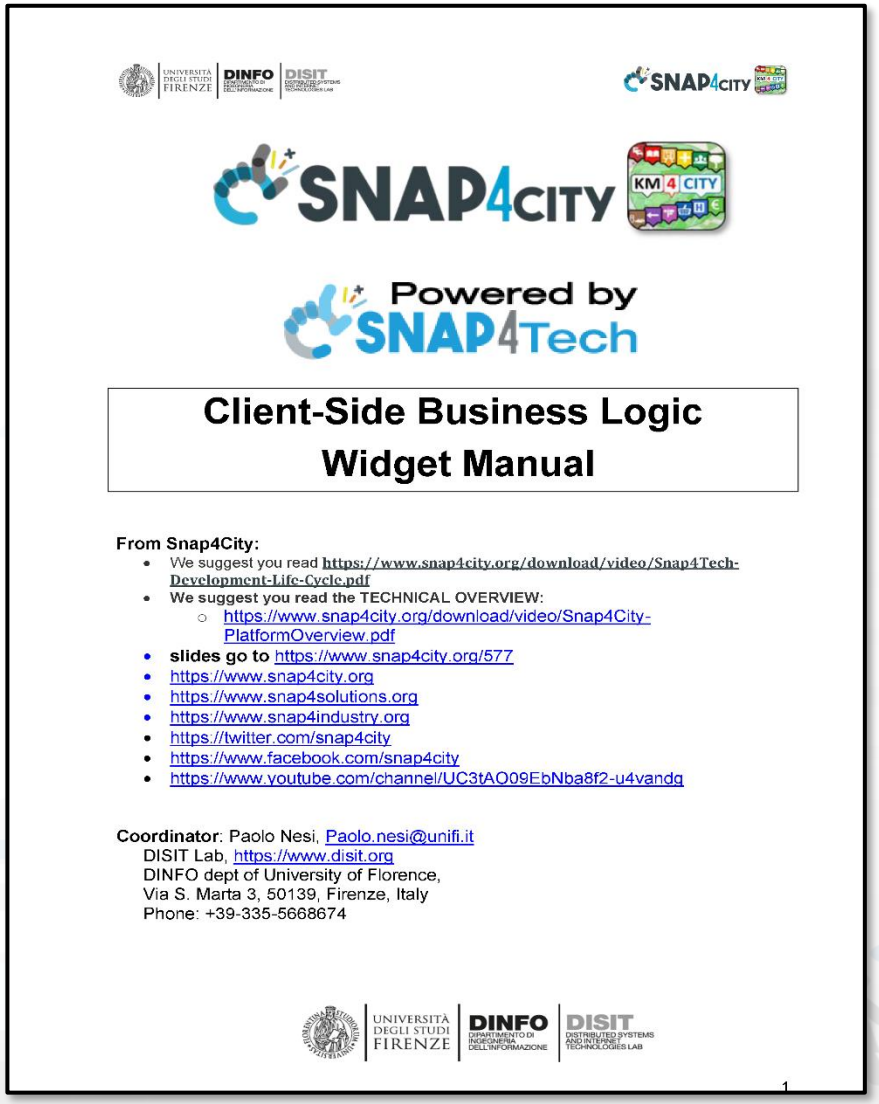
Coordinator: Paolo Nesi, Paolo.nesi@unifi.it

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

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

1

Client Side Business Logic



SNAP4CITY KM4CITY

Powered by SNAP4Tech

Client-Side Business Logic Widget Manual

From Snap4City:

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

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

UNIVERSITÀ DEGLI STUDI FIRENZE | DINFO | DISIT

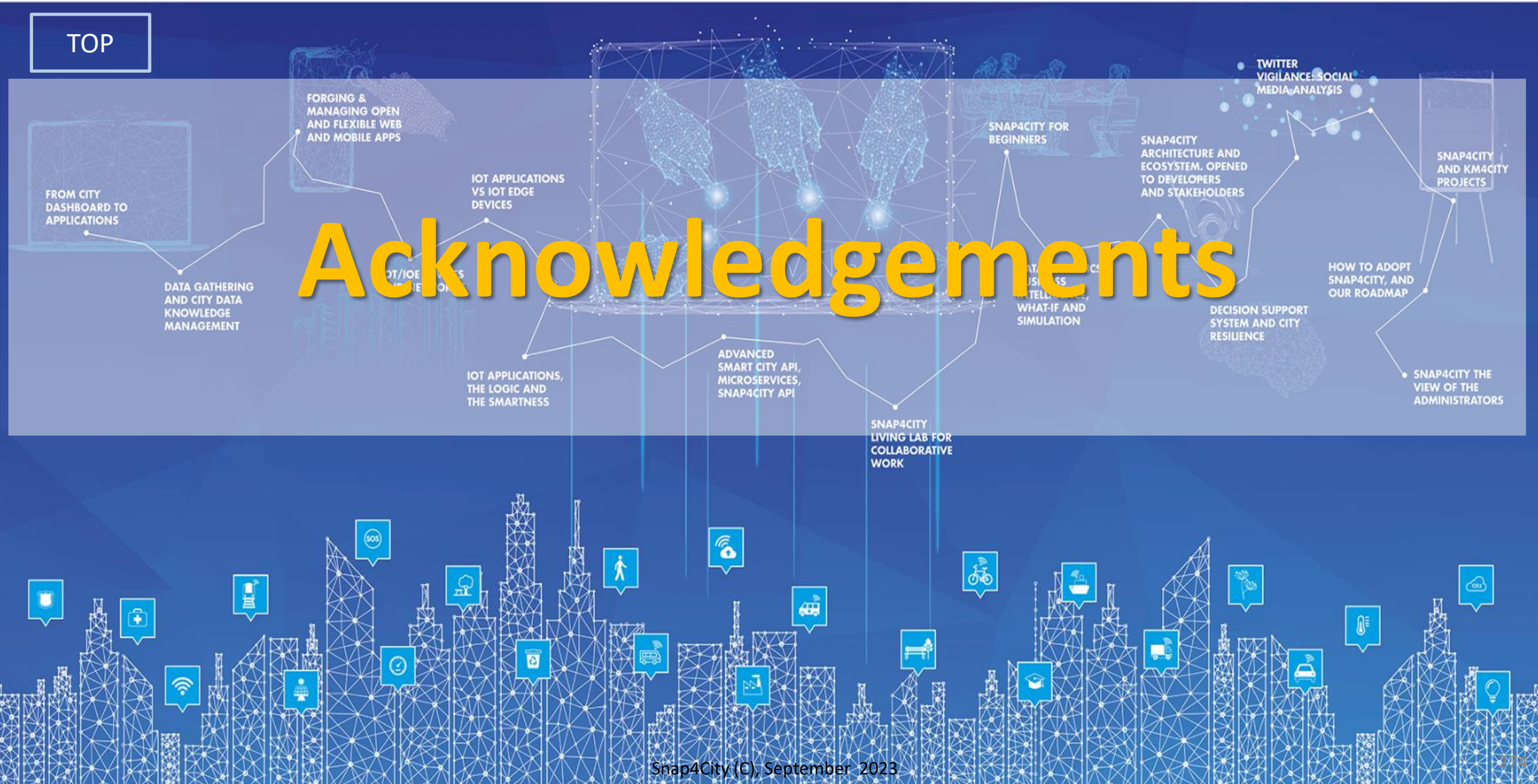
1

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

Overview

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

TOP



Acknowledgements

DISIT lab roadmap vs model and tools' usage

2013
Km4City Ontology 1.1

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

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

- Twitter Vigilance
- Social Media Analytics, Sentiment Analysis

Km4City 1.4

2015
Km4City 1.5

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

ckan, resolute, H2020 (2015-18)

2016
(2016-21) H2020 REPLICATE

Km4City 1.6.4

- Smart Energy
- Sustainable Mobility
- Control Room
- Dashboard

Km4City 1.6.2

Sii-Mobility (2016-21) MIUR

SII-MOBILITY SCN

- Infomobility
- Mobile App
- Routing
- Multimodality

FIWARE

Km4City 1.5

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

2017
Sii-Mobility

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

E0I5 digital ecosystem (Powered by EXPO MILANO 2015)

(2016-19) MIUR GHOST SIR

- Sardinia Region Smart City Strategies and plan

GREEN IMPACT C.A.P.A.C.I.T.Y.

GREEN IMPACT
POR FESR 2014-2020

- Industry 4.0
- Critical Plant
- Monitoring

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

life weee (2017-20)

- Smart Waste

2018
MOSAiC Mobility 4.0 for SmArt (i) City (2018-20)

- Mobility Demand / Offer Analytics and Strategy

Sii-Mobility

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

IOT/IOE

Km4City 1.6.6

Node-RED

SNAP4CITY SELECT for Cities H2020 (2017-19)

2019
5G (2018-21)

- 5G tech
- Energy
- Industry 4.0
- Synoptics

TRAFAIR CEF (2018-21)

- Traffic and Mobility Impact on Pollution
- NOX predictions

EUROPEAN OPEN SCIENCE CLOUD

SIRIUS

LAID Smart Bed

- Smart Health

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

- Industry 4.0

bee smart city

- Smart Lonato

Reverberi Enetec Gruppo MPES

SNAP4CITY

Winner of Select4Cities PCP

SNAP4CITY roadmap

SODA

2020

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

- Smart Mobility
- PISA, PUMS
- Living lab

smartGARDAlake

Km4City 1.6.7

Smart Ambulance (2021-22)

Enterprise (2021-22) Industry 4.0

Contract

2021

PC4City (2020-21) Monitoring Terrain

Winner of Open Data Challenge of

- Smart Light
- Sweden

Almafluida Industry 4.0 (2021-22)

AMPERE (2021-22) Industry 4.0

SYN-RG-AI SmartCity

Industry 4.0

SmartCity, 2021-23

AXIS collab SmartCity

2022

Asymmetrica Smart City, 2022-23

Contract, 2022-23

2023

Contract, 2022-23

2022-2023

Contract

Security and Risk

Italferr, Smart City

Contract, 2022-23

CN MOST, 2022-26

EI THE, 2022-26

G. Agile, 2021-23

2023-26

Merano, smart light

OceanRace, Genova, AWS

Cuneo, smart city

TOP

SNAP4CITY

KM4CITY

100% OPEN SOURCE

Be smart in a SNAP!

SMARTCITY EXPO WORLD CONGRESS

7-9 November 2023, Barcelona, Spain

Visit Snap4City in Hall 1

SNAP4 Appliances and Dockers **Installations**

CONTACT

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

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

www.snap4city.org

Email: snap4city@disit.org

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

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