



[www.snap4city.org](http://www.snap4city.org)  
[www.snap4solutions.org](http://www.snap4solutions.org)



[www.km4city.org](http://www.km4city.org)

## IoT App. / Proc.Logic Server Side Business Logic

Nov. 2023, Course, Part 3

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

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

**DIGITAL TWIN SOLUTIONS TO SETUP SUSTAINABLE DECISION SUPPORT SYSTEMS AND BUSINESS INTELLIGENCE**

Paolo Nesi, [paolo.nesi@unifi.it](mailto:paolo.nesi@unifi.it)  
<https://www.Km4City.org>  
<https://www.disit.org>



*Be smart in a SNAP!*



7-9 November 2023, Barcelona, Spain

**SMARTCITY**  
EXPO WORLD CONGRESS

Visit Snap4City in Hall 1

IoT App. / Proc.Logic  
Server Side Business Logic

Nov. 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  
DIPARTIMENTO DI  
TECNOLOGIE DELL'INFORMAZIONE

DISIT  
DISTRIBUTED SYSTEMS  
AND IT INFRASTRUCTURES  
TECHNOLOGIES LAB







UNIVERSITÀ DEGLI STUDI  
FIRENZE

DINFO  
DIPARTIMENTO DELL'INFORMAZIONE

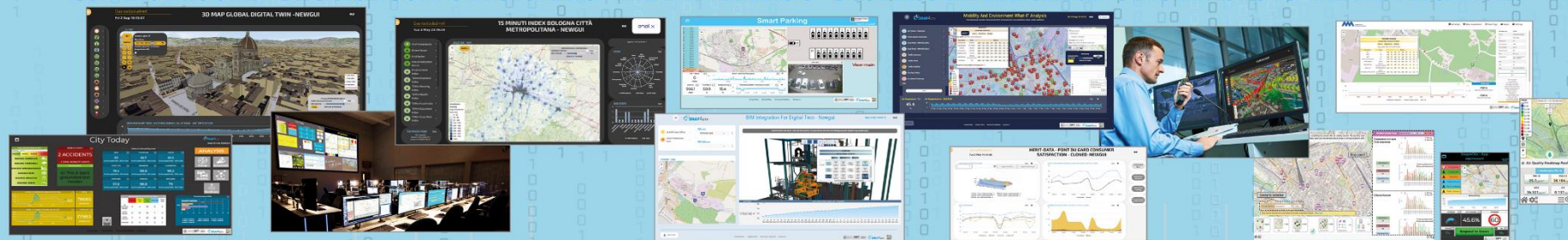
DISIT  
DIPARTIMENTO SISTEMI  
TECNOLOGIE LAB

SNAP4CITY



# 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  
EVENT DRIVEN

Native and External  
Smart Applications

Mobility & Transport

Light & Energy

Waste

Environment

Building

Tourism

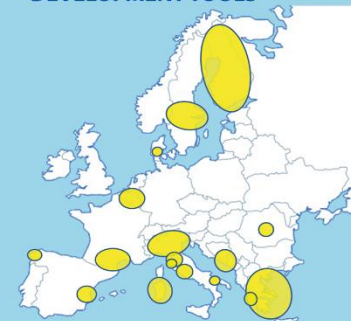
Asset Management

Security and Safety

Social Media



METHODOLOGIES  
LIVING LABS  
COURSES AND COMMUNITY  
DEVELOPMENT TOOLS



Powered by  
FIWARE

FREE  
TRIAL

PEN Test  
Passed

EU GDPR  
COMPLIANT

SNAP4  
Appliances and Dockers  
Installations

EUROPEAN OPEN  
SCIENCE CLOUD

Node-RED

JS Foundation

E015  
digital ecosystem

NVIDIA



























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

*On Line Training Material (free of charge)*



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







# Note on Training Material





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



# Tech Overview

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









## Snap4City Platform

### Technical Overview

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

Snap4City:

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

Contact Person: Paolo Nesi, [Paolo.nesi@unifi.it](mailto:Paolo.nesi@unifi.it)

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

1



# 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](mailto:Paolo.nesi@unifi.it)

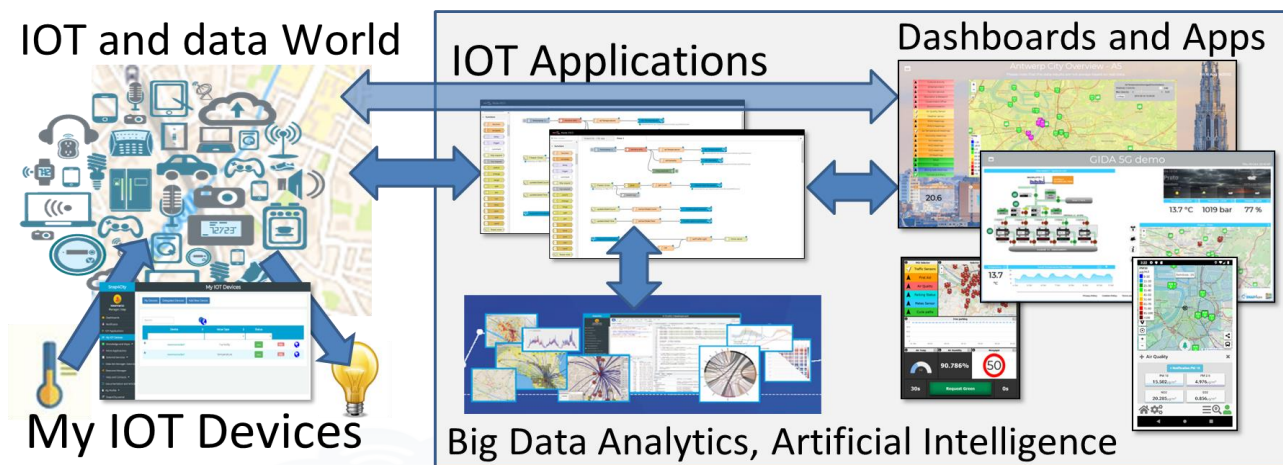
DISIT Lab, <https://www.disit.org>  
 DINFO dept of University of Florence,  
 Via S. Marta 3, 50139, Firenze, Italy  
 Phone: +39-335-5668674





# Free Trial

- Register on [WWW.snap4city.org](http://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.*

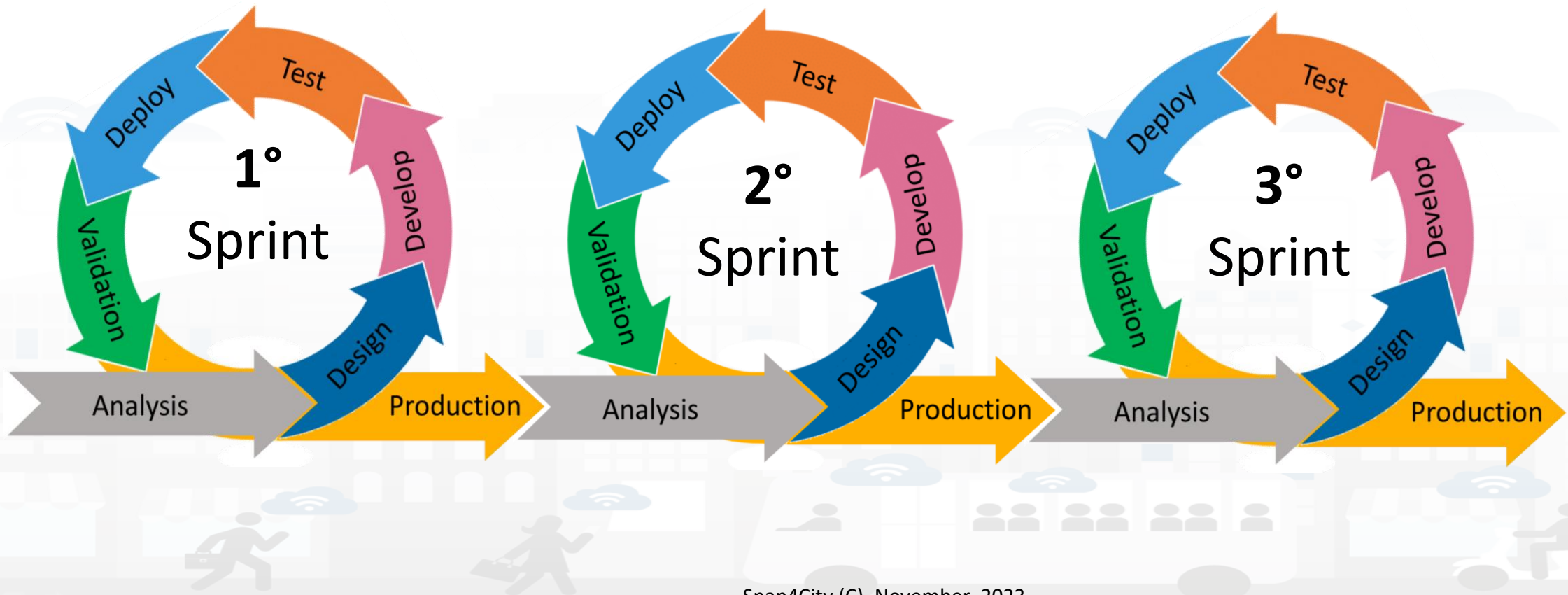


# Agenda of third part

- 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  $\leftrightarrow$  IoT App/Proc.Logic
- Dashboards  $\leftrightarrow$  IoT App/Proc.Logic
  - Server Side Business Logic

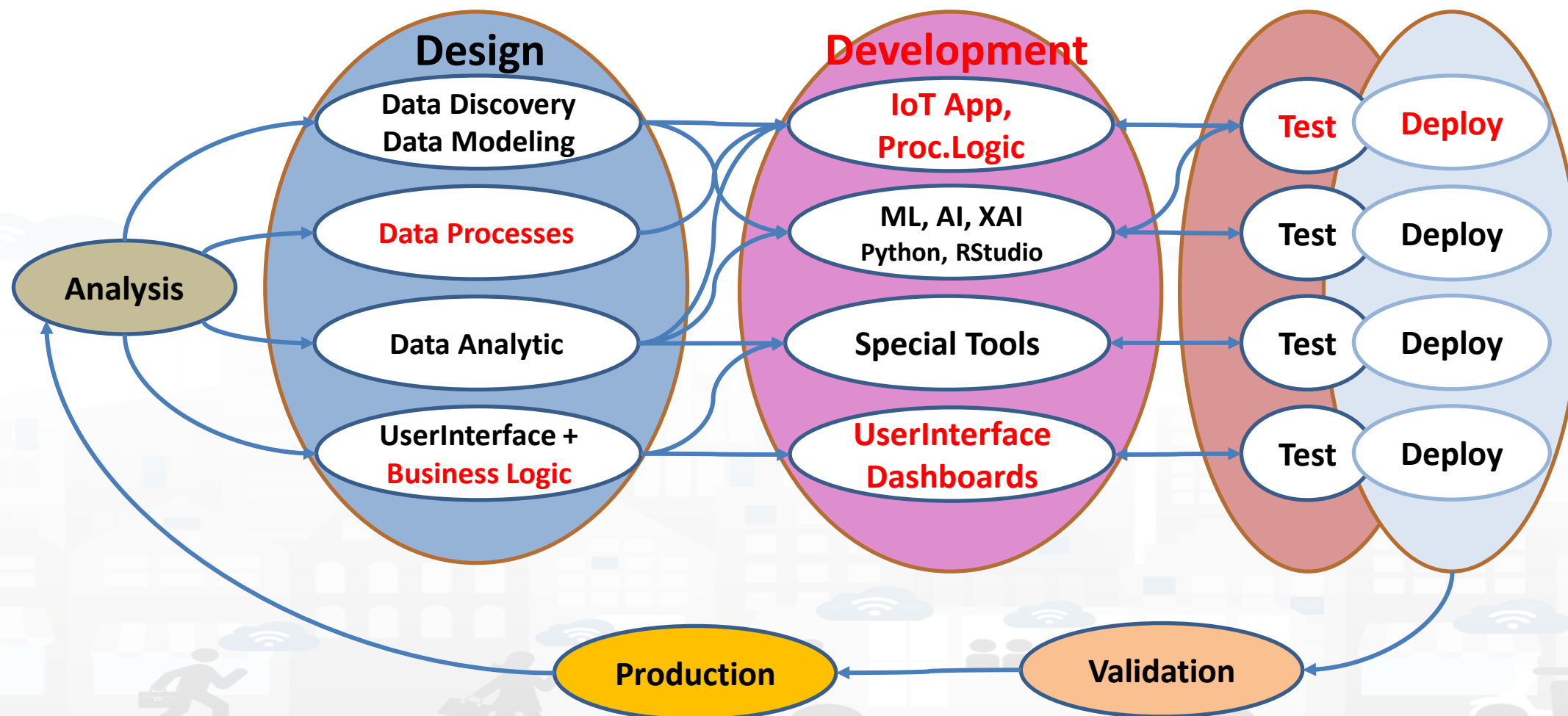


# Development Life Cycle Smart Solutions





# Development Life Cycle Smart Solutions



TOP

# Recall on Snap4City Architecture

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

APPLICATIONS THE LOGIC AND THE SMARTNESS

IC APPLICATIONS VISIT EVIDENCES

ADVANCED SMART CITY API, AND SERVICES SNAP4CITY API

SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK

SNAP4CITY FOR BEGINNERS

DATA ANALYTICS, BUSINESS INTELLIGENCE, WHAT-IF AND SIMULATION

CITY ARCHITECTURE AND ECOSYSTEM TYPE TO DEVELOPERS AND STAKEHOLDERS

TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS

DECISION SUPPORT SYSTEM AND CITY RESILIENCE

HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP

SNAP4CITY AND KM4CITY PROJECTS

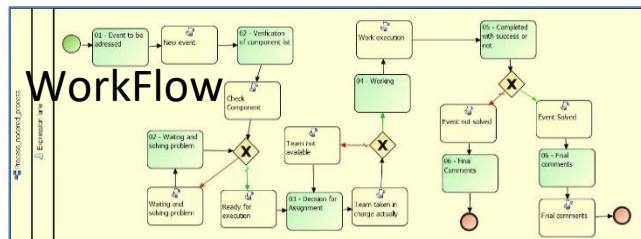
SNAP4CITY THE VIEW OF THE ADMINISTRATORS



# Concept



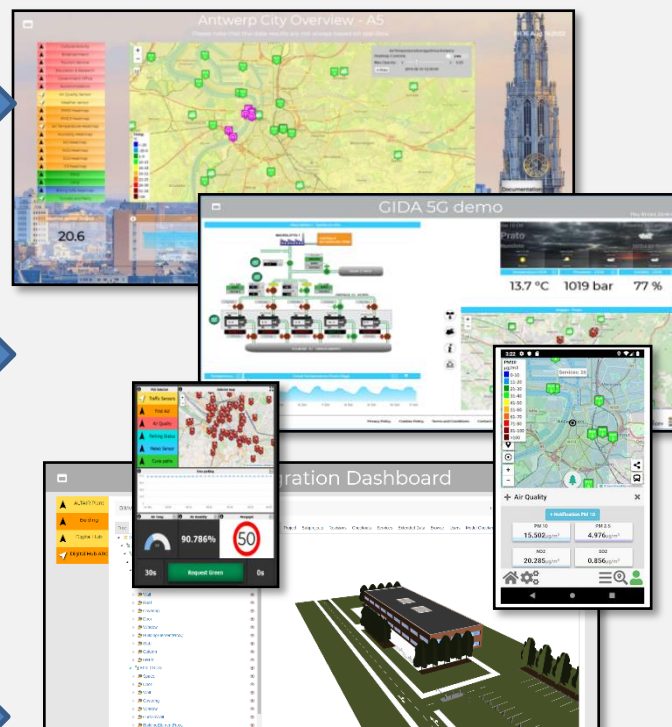
KPI, POI, MyKPI, ...  
API, External Services  
Web Scraping



IOT Apps



Data Analytics,  
Artificial Intelligence



IOT Brokers

IOT Broker

IOT Broker



GIS



Big Data



LD, LOD



ckan

Dashboards and Apps

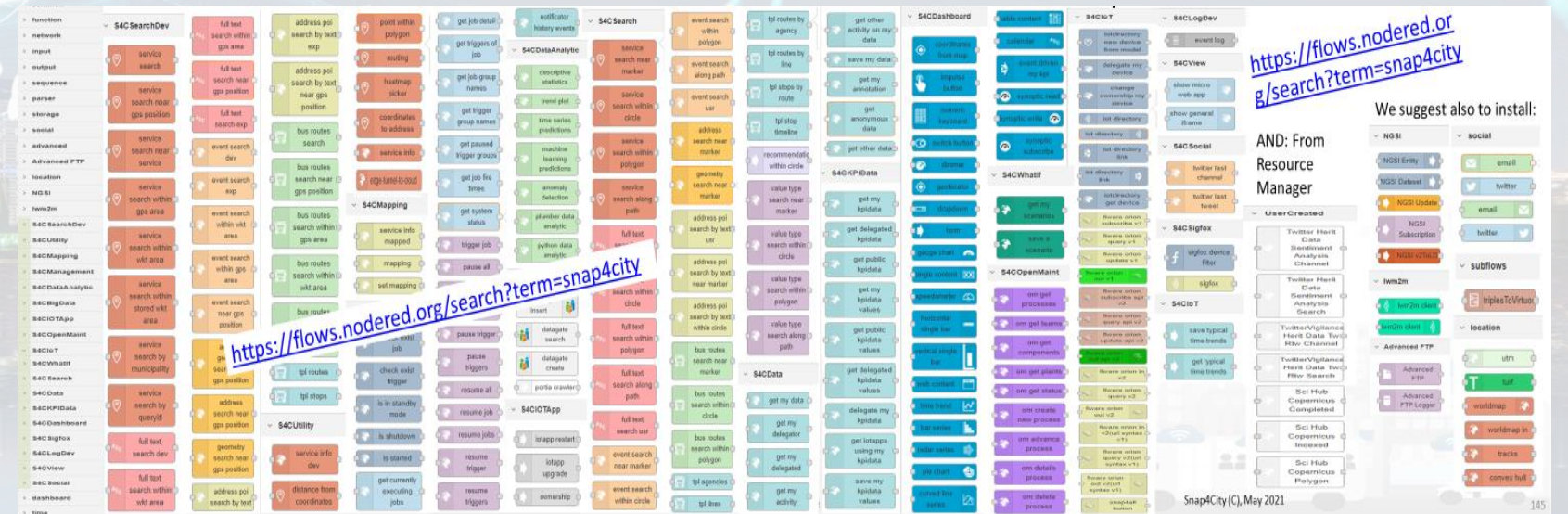
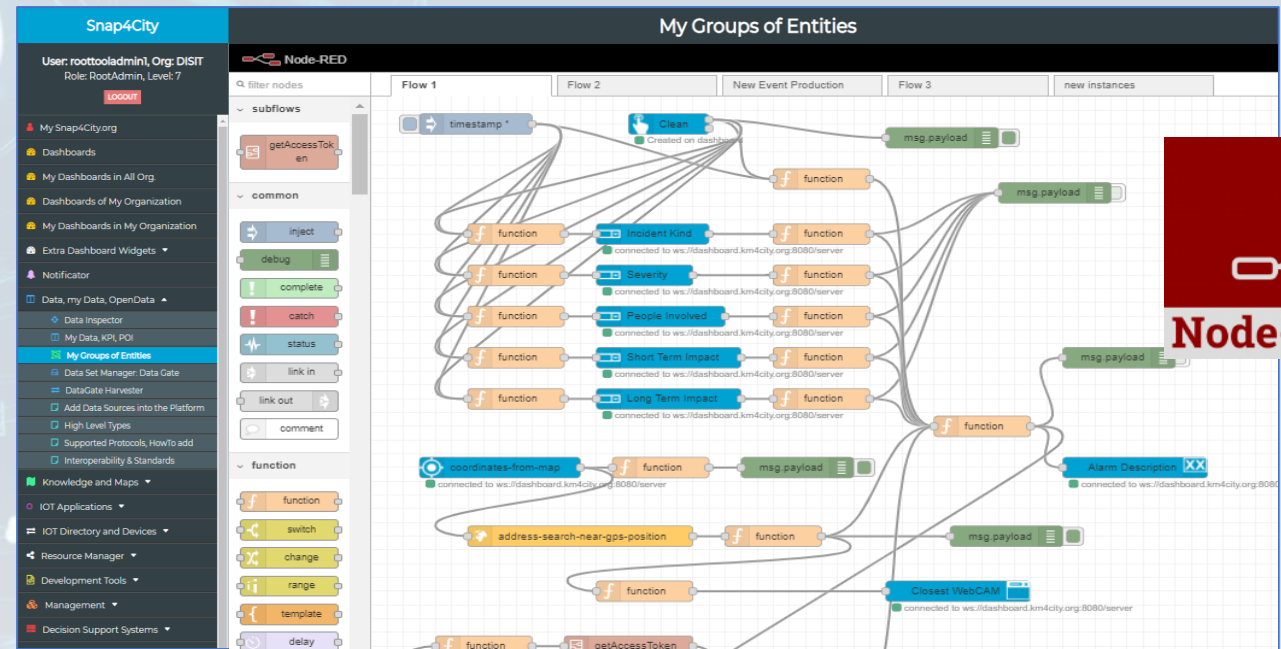




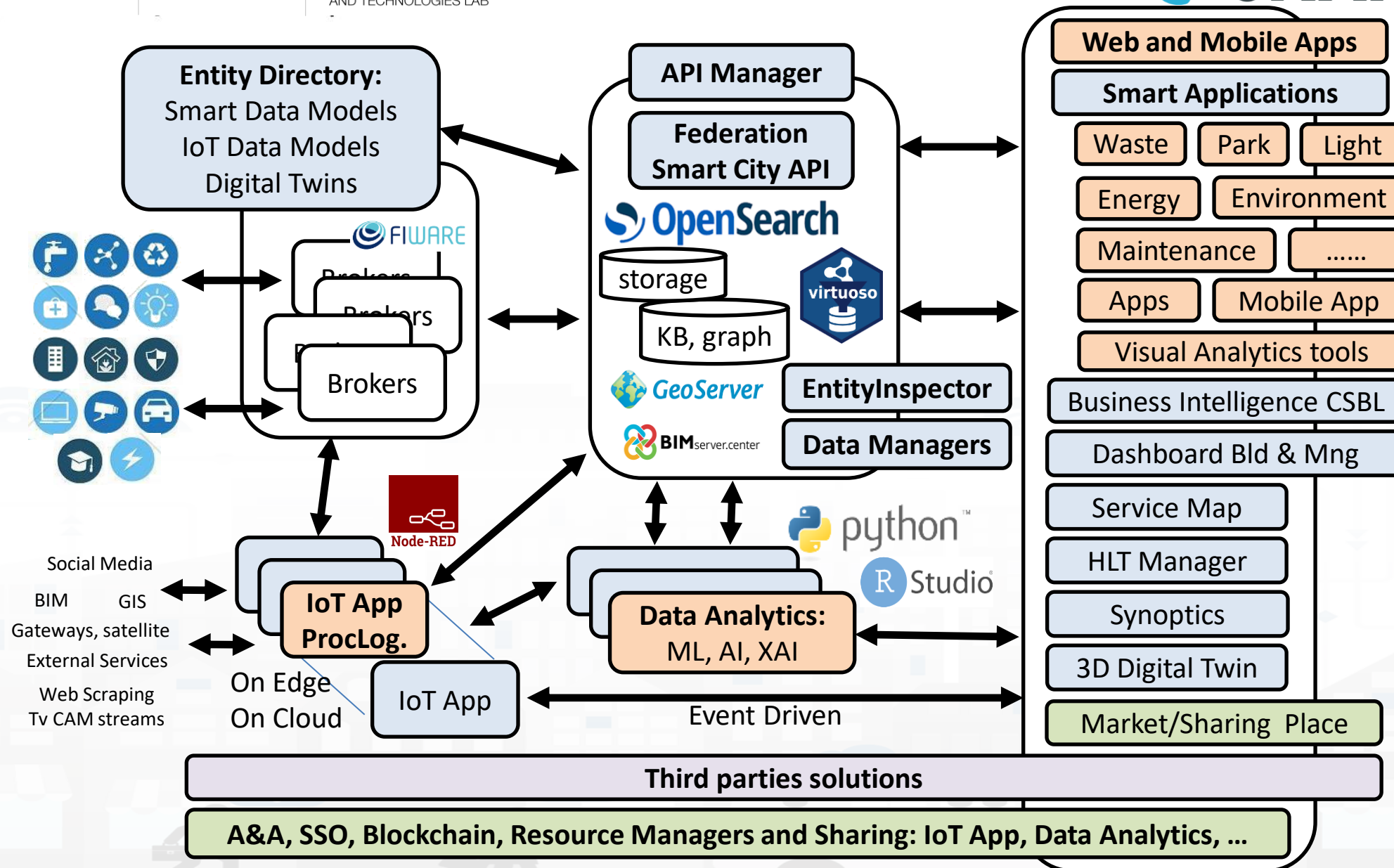
# Ingestion, aggreg. → exploitation

- 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

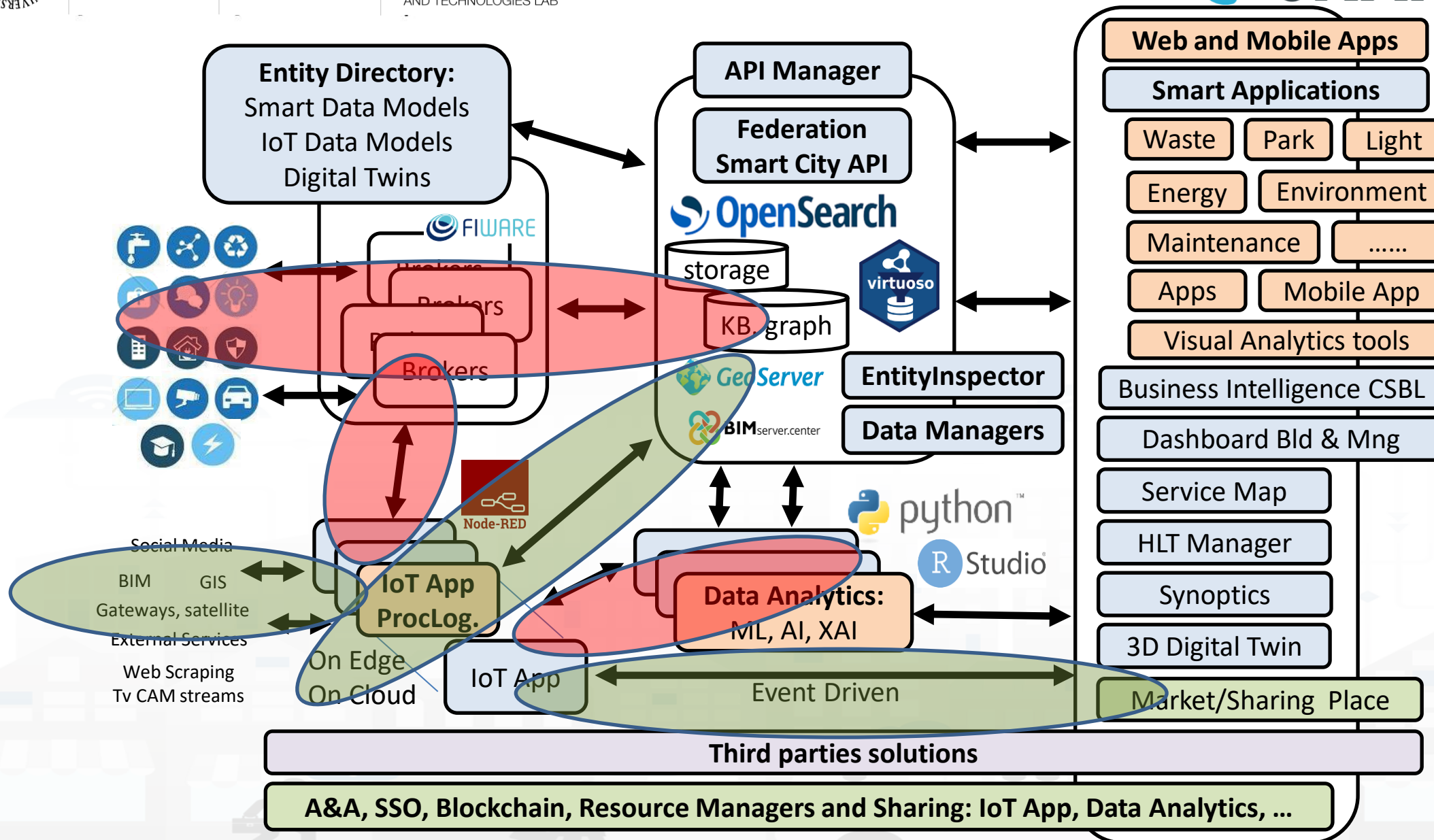


# Tech Arch





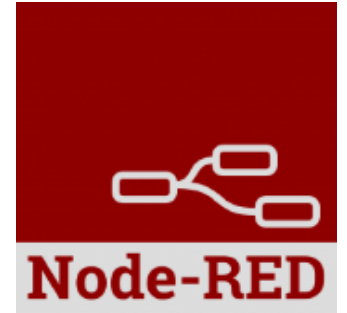
# Tech Arch





# IoT App / Proc.Logic

- Storage  $\rightarrow$  IoT App / Proc.Logic
- External Service  $\leftrightarrow$  IoT App / Proc.Logic
- Dashboards  $\leftrightarrow$  IoT App / Proc.Logic



- Data Analytics  $\leftrightarrow$  IoT App / Proc.Logic
- Broker  $\rightarrow$  Storage
- IoT App / Proc.Logic  $\rightarrow$  Broker
- Broker  $\rightarrow$  IoT App / Proc.Logic
- IoT App / Proc.Logic  $\rightarrow$  Storage

Part 4

Part 5

## FROM CITY DASHBOARD TO APPLICATIONS

## IOT APPLICATIONS VS IOT EDGE DEVICES

## DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT

## IOT/IOE DEVICES AND NETWORKS

## IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS

**ADVANCED  
SMART CITY API,  
MICROSERVICES,  
SNAP4CITY API**

**SNAP4CITY  
LIVING LAB FOR  
COLLABORATIVE  
WORK**

## SNAP4CITY FOR BEGINNERS

**SNAP4CITY  
ARCHITECTURE AND  
ECOSYSTEM. OPENED  
TO DEVELOPERS  
AND STAKEHOLDERS**

**TWITTER  
VIGILANCE: SOCIAL  
MEDIA ANALYSIS**

## SNAP4CITY AND KM4CITY PROJECTS

## HOW TO ADOPT SNAP4CITY, AND

AP4CITY THE  
W OF THE  
MINISTRATORS

# Node-RED

**DATA ANALYTICS,  
BUSINESS  
INTELLIGENCE,  
WHAT-IF AND  
SIMULATION**

# Node-RED

filter nodes

input

inject

catch

status

link

mqtt

http

websocket

tcp

udp

amqp

amqp2

output

debug

link

mqtt

http response

websocket

tcp

udp

amqp

amqp2

flow1

Flow 1

world map

point

service-search-near-marker

show micro web app

transform results

world map

event-log

popupopen

msg.payload

timestamp

service-info

vehicleFlow

vehicle flow (car/h)

worldmap

switch

sensor

msg.payload

timestamp

last temperature

Dashboard

get value

temperature

Temperature

event-log

info

debug

dashb

Flow

Name	flow1
ID	"49a71aa0.b297b4"
Status	Enabled

Information

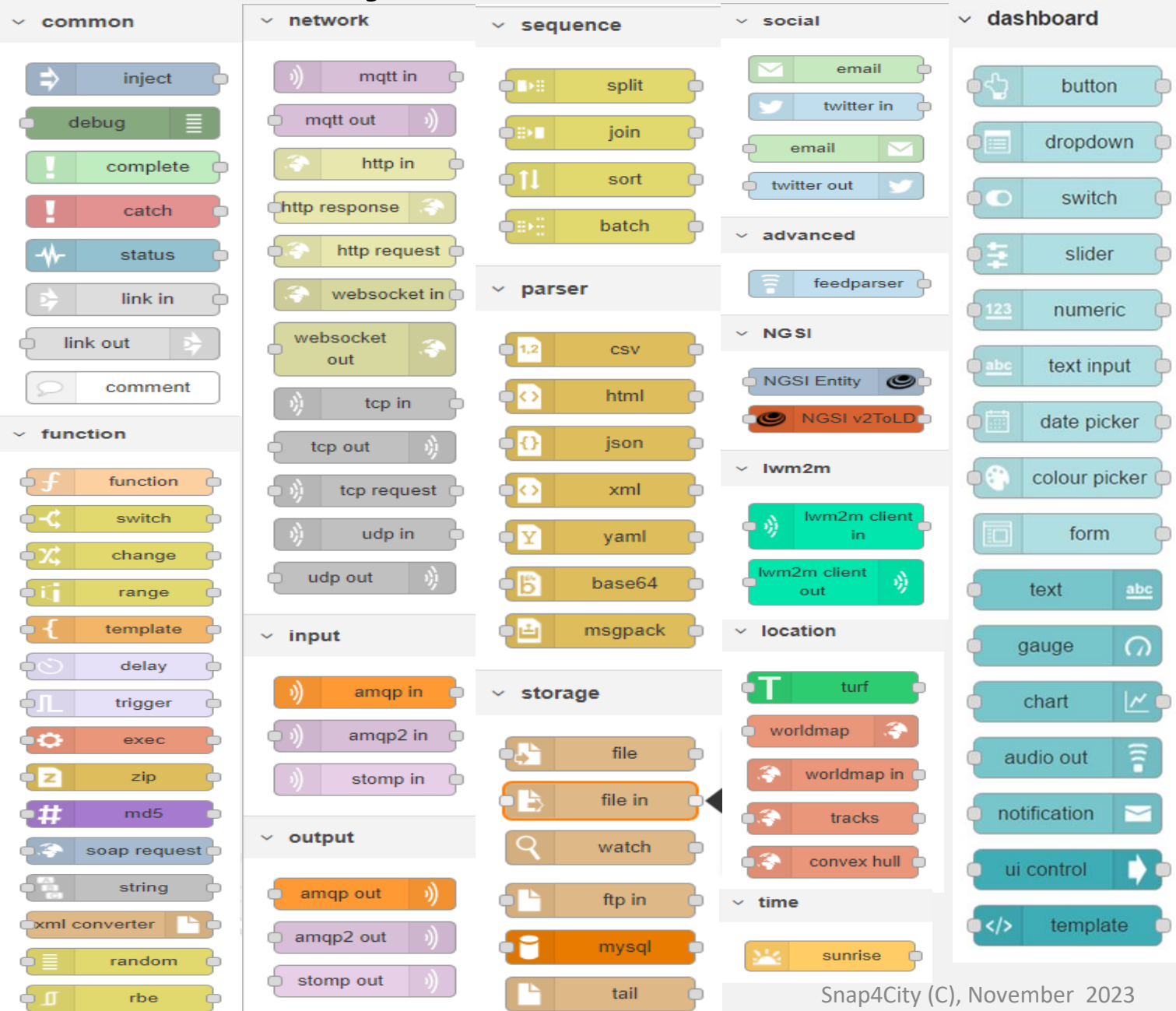
Node-RED

Search for nodes using

ctrl-J



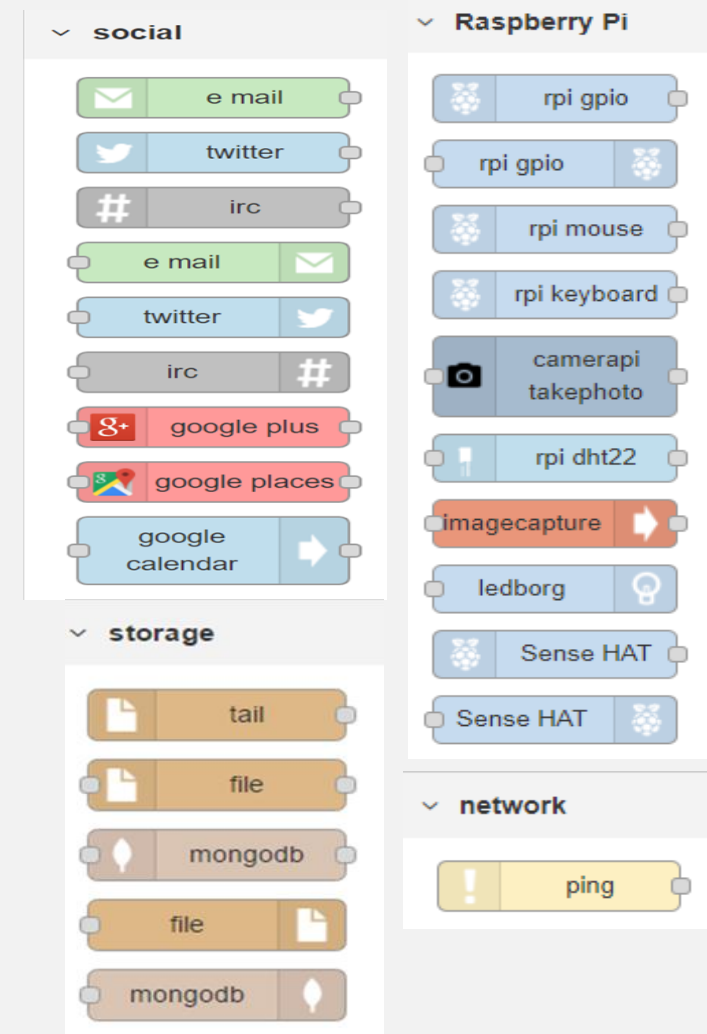
# Basic Node.js Blocks on NodeRed on our Advanced IOT Apps



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

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

+ on IOT Edge Raspberry



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

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

# Node-RED Basic Blocks

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

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

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

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

**+ on IOT Edge Raspberry**

Categories shown: common, network, sequence, social, dashboard, function, input, output, storage, time, Raspberry Pi.

Nodes visible include: inject, debug, complete, catch, status, link in, link out, comment, mqtt in, mqtt out, http in, http response, http request, websocket in, websocket out, tcp in, tcp out, tcp request, udp in, udp out, amqp in, amqp2 in, stomp in, amqp out, amqp2 out, stomp out, split, join, sort, batch, csv, html, json, yaml, base64, msgpack, file, file in, file out, watch, ftp in, ftp out, mysql, tail, email, twitter in, twitter out, feedparser, NGSI, NGSI v2toLD, Iwm2m, Iwm2m client in, Iwm2m client out, location, turf, worldmap, worldmap in, tracks, convex hull, time, sunrise, button, dropdown, switch, slider, numeric, text input, date picker, colour picker, form, text, gauge, chart, audio out, notification, ui control, template, e mail, twitter, irc, google plus, google places, google calendar, rpi gpio, rpi gpio, rpi mouse, rpi keyboard, camerapi takephoto, rpi dht22, imagecapture, ledborg, Sense HAT, Sense HAT, tail, file, mongodb, file, mongodb, ping.

Snap4City (C), November 2020

# Hello World of Node-RED

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

The screenshot shows the Node-RED web interface in a browser window. The address bar displays '127.0.0.1:1880/#'. The interface includes a left sidebar with a 'filter nodes' search bar and two categories: 'output' and 'function'. The 'output' category is expanded, showing nodes like 'debug', 'link', 'mqtt', 'http response', 'websocket', 'tcp', and 'udp'. The 'function' category shows a 'function' node. The main workspace, titled 'Flow 1', contains a flow with a 'Hello, world!' message box connected to a 'msg.payload' node. The right sidebar has tabs for 'info' and 'debug'. The 'info' tab is active, displaying a table with node details:

Node	
Type	debug
ID	2d930e35.482d92

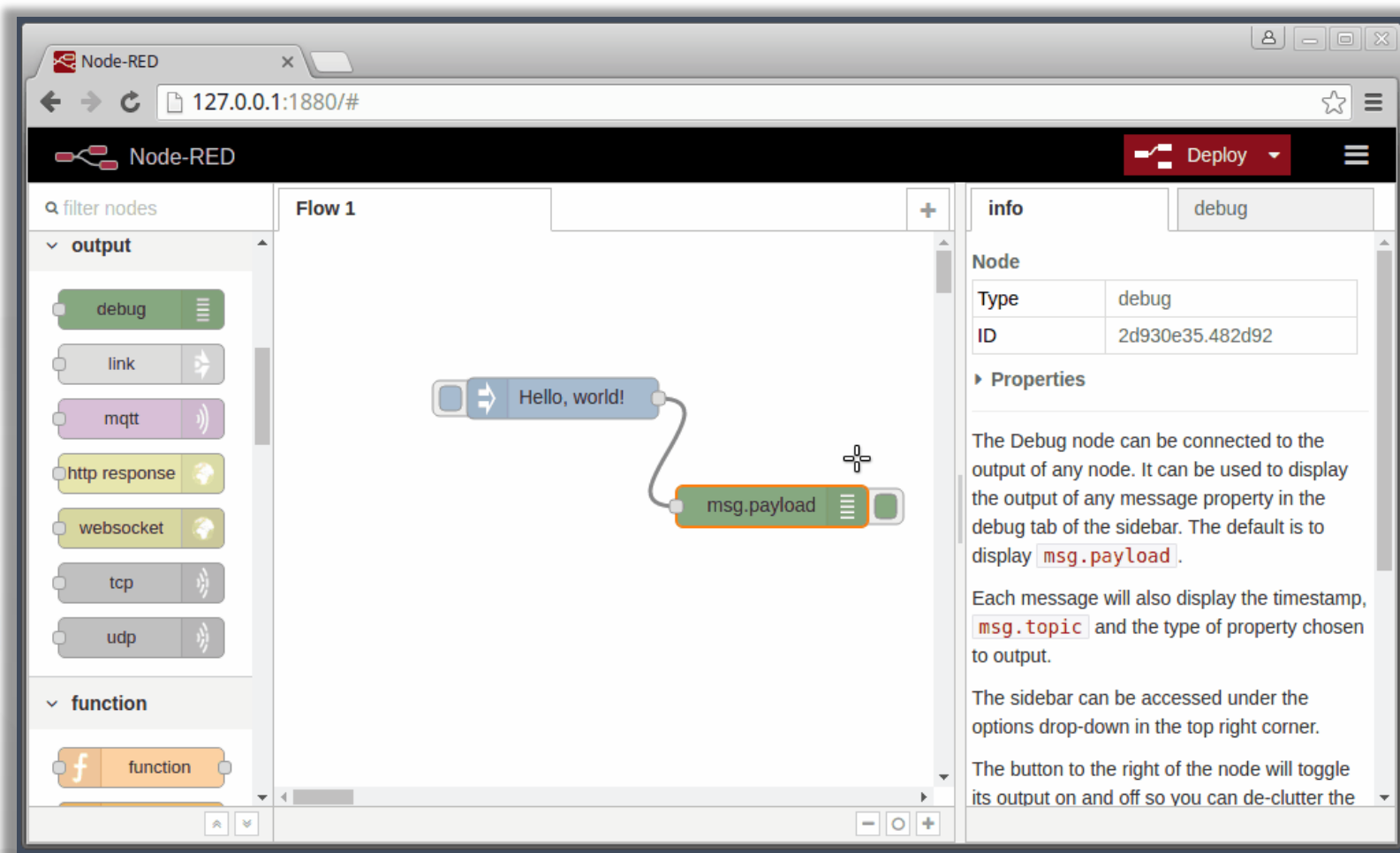
Below the table, the 'Properties' section explains the 'debug' node: 'The Debug node can be connected to the output of any node. It can be used to display the output of any message property in the debug tab of the sidebar. The default is to display `msg.payload`. Each message will also display the timestamp, `msg.topic` and the type of property chosen to output. The sidebar can be accessed under the options drop-down in the top right corner. The button to the right of the node will toggle its output on and off so you can de-clutter the'.



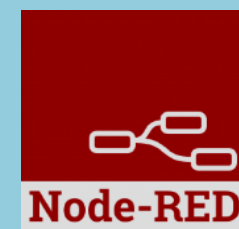


# Node-RED

- 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

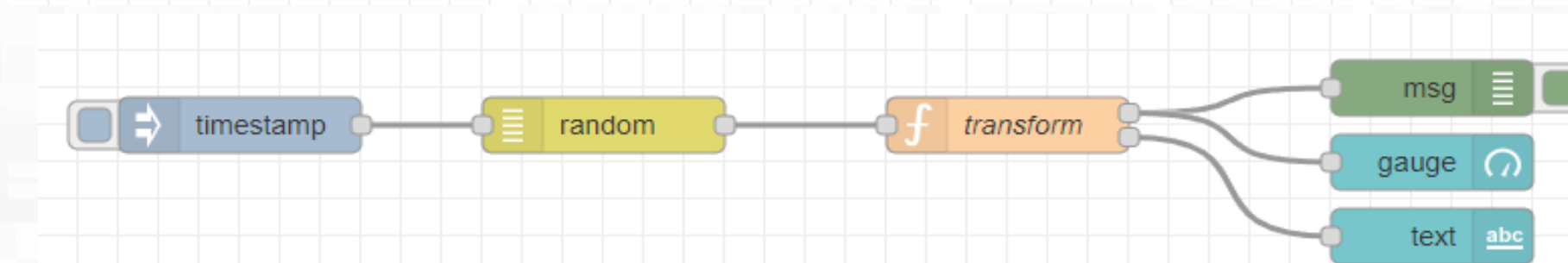
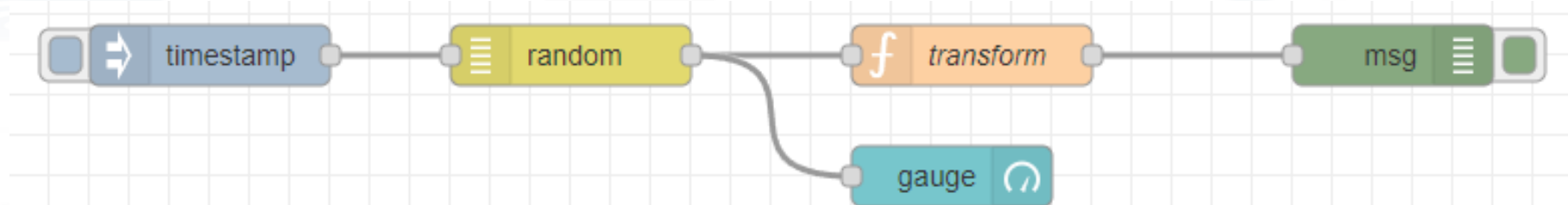


# Course 2023 IoT App / Proc.Logic

- Pattern:



- Pattern:



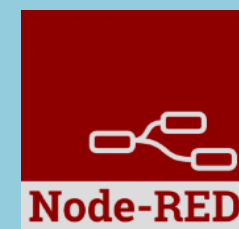


## recap

- 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.

TOP

# Node-RED Demo 1



# Example of simple IoT Application

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

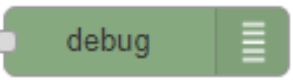
- generate a random value,
- the value is switched on the correct path
- the value is showed in the local dashboard of NodeRed



# Nodes for flow



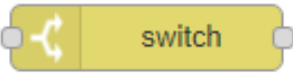
Generates an input for the other nodes. It can be repeated at predefined intervals, entered manually and of various types (timestamp, string, number, boolean, JSON etc)



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)



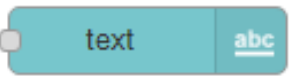
Generates a random number. You can configure the number generation interval and the type (integer or float).



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

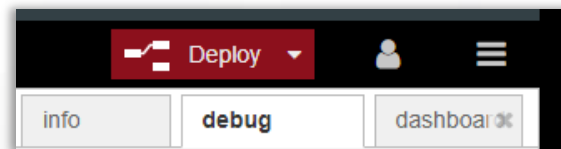


- Connect

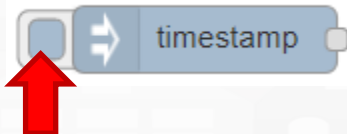
- Configure

☒ Payload timestamp  
  
☒ Repeat interval  
 every  minutes  
☒ Inject once at start?

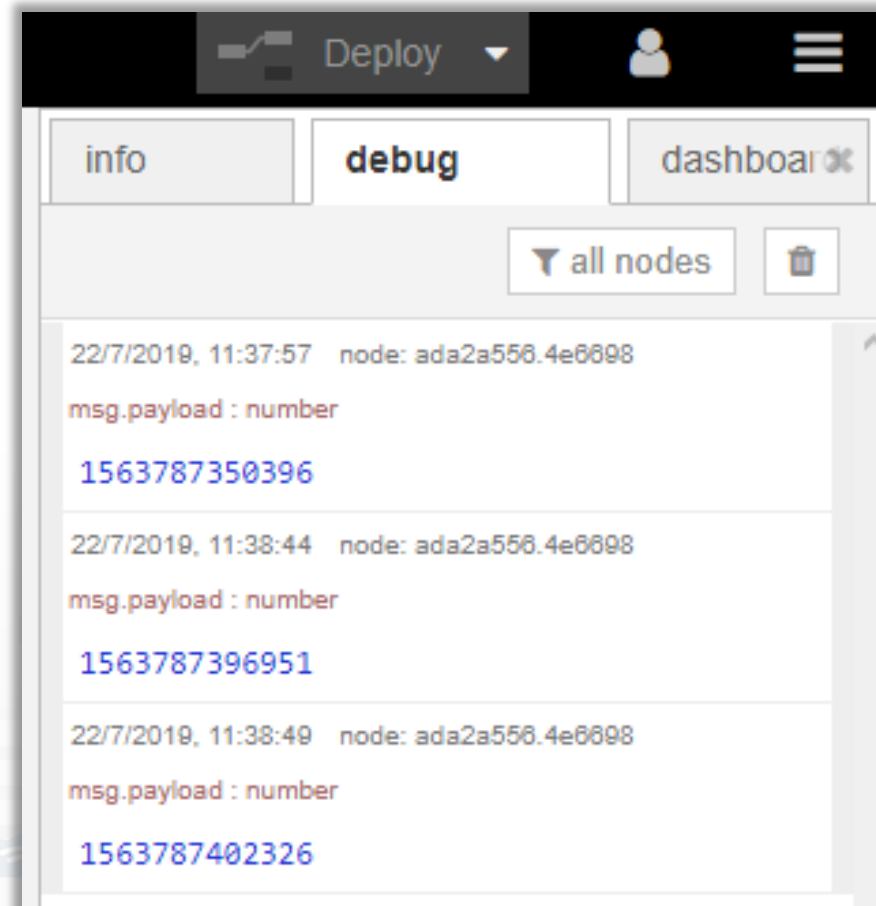
- Deploy



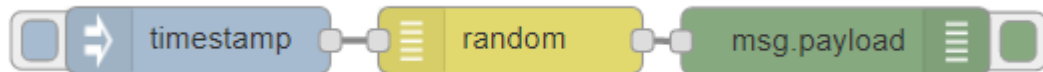
- Click



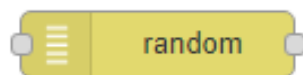
- Observe



# Step 2



- Random
- Connect
- Configure
- Deploy
- Click
- Observe



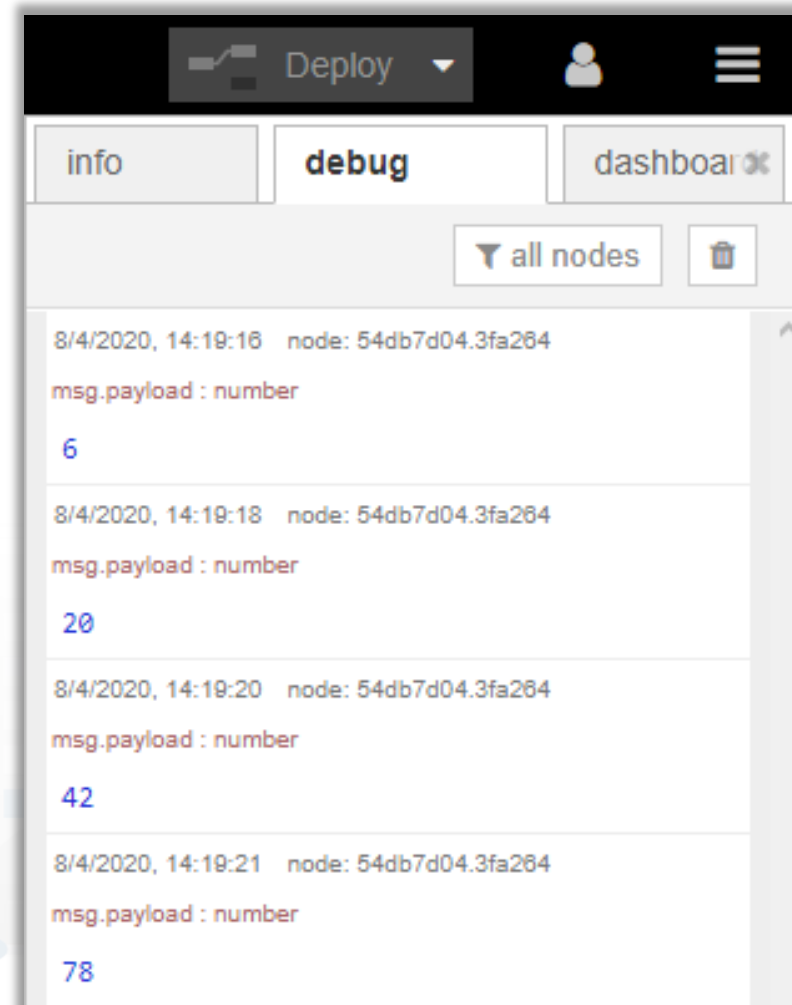
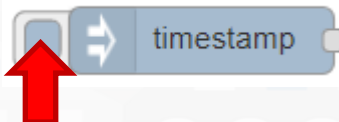
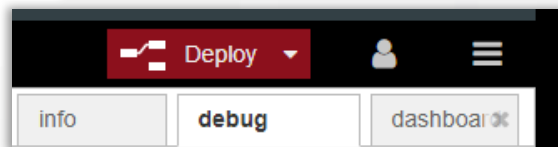
msg.payload

Generate: a whole number - integer

From: 1

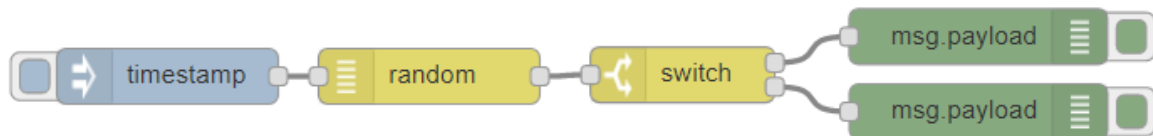
To: 100

Name: Name

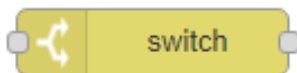




# Step 3



- Switch
- Connect
- Configure



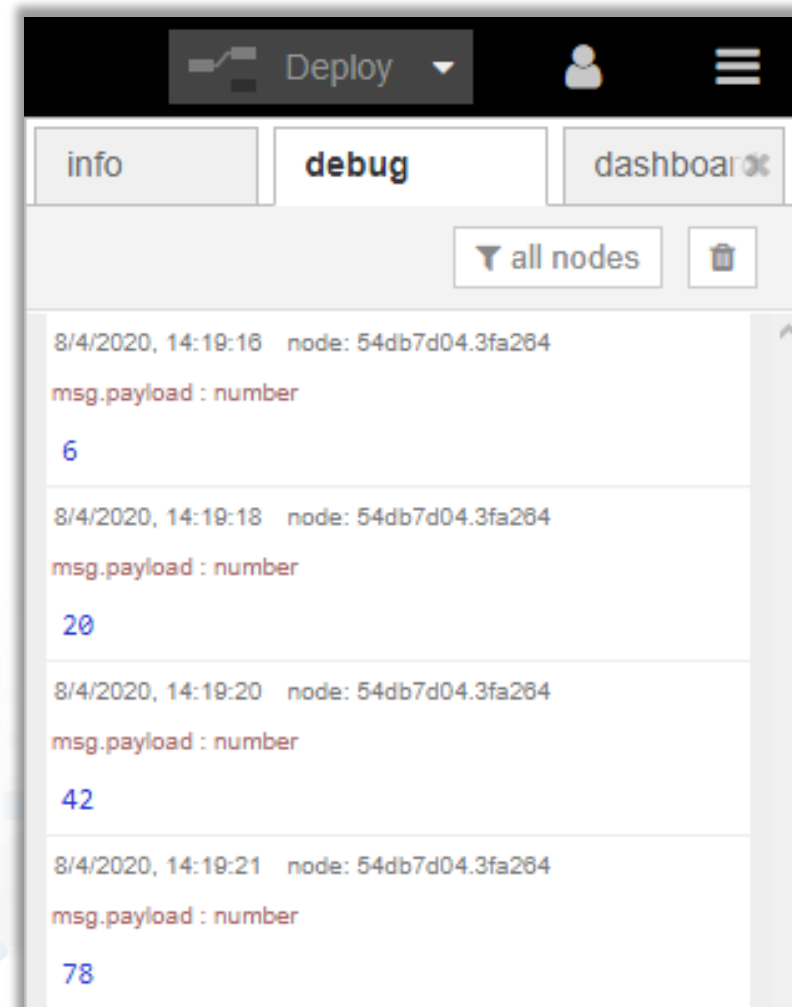
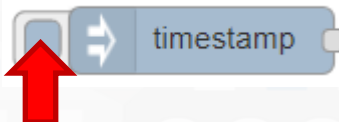
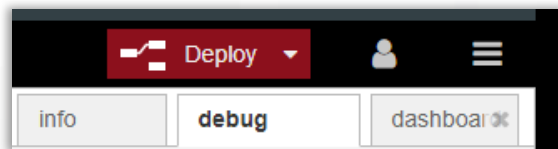
Name

Property

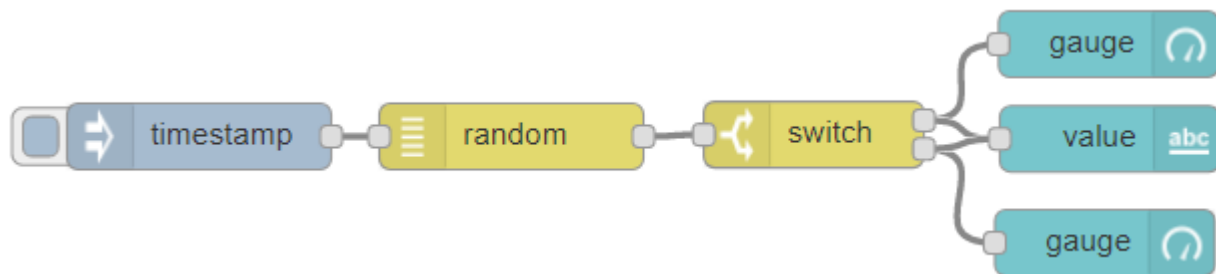
☐  → 1

☐ otherwise → 2

- Deploy
- Click
- Observe



# Step 4



- Gauge and text



- Connect

- Configure gauge

Group

[Home] Default

Size

auto

Type

Gauge

Label

gauge

Value format

{{value}}

Units

units

Range

min 0 max 100

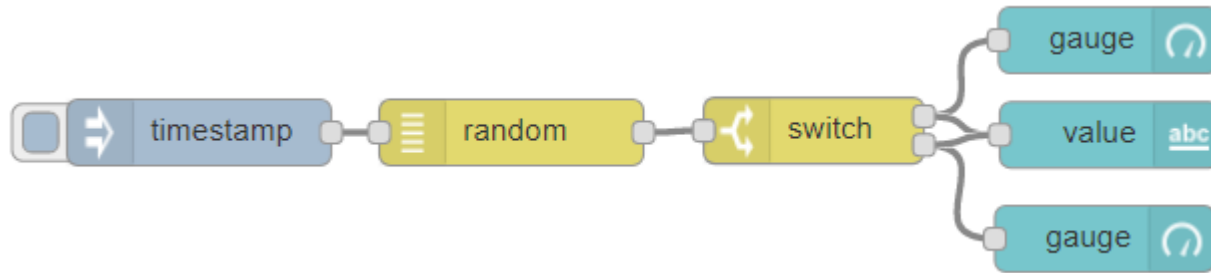
Colour gradient

Sectors

0 optional optional 100

Name

## Step 4 Bis



- Gauge and text
- Connect
- Configure text

Group

[Home] Default

Size

auto

Label

value

Value format

{{msg.payload}}

Layout

label value

label value

label value

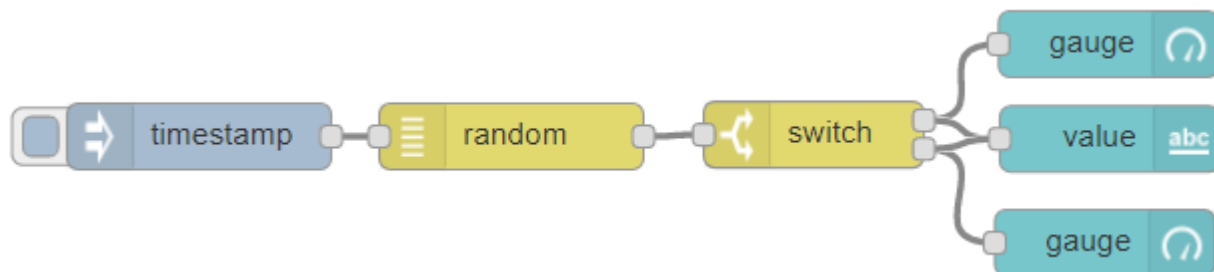
label value

label value

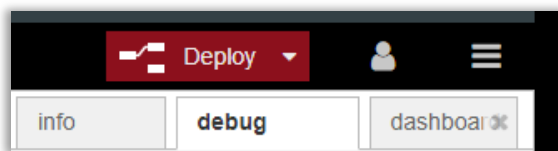
Name



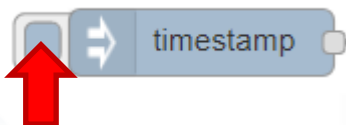
# Step 5



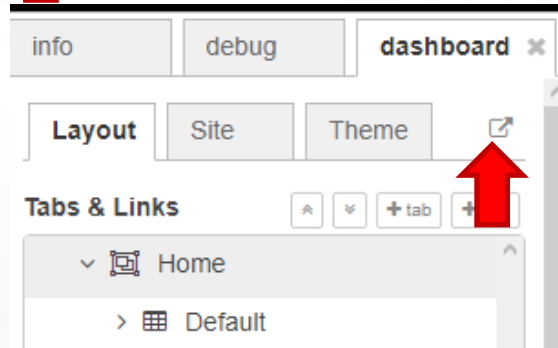
- Deploy



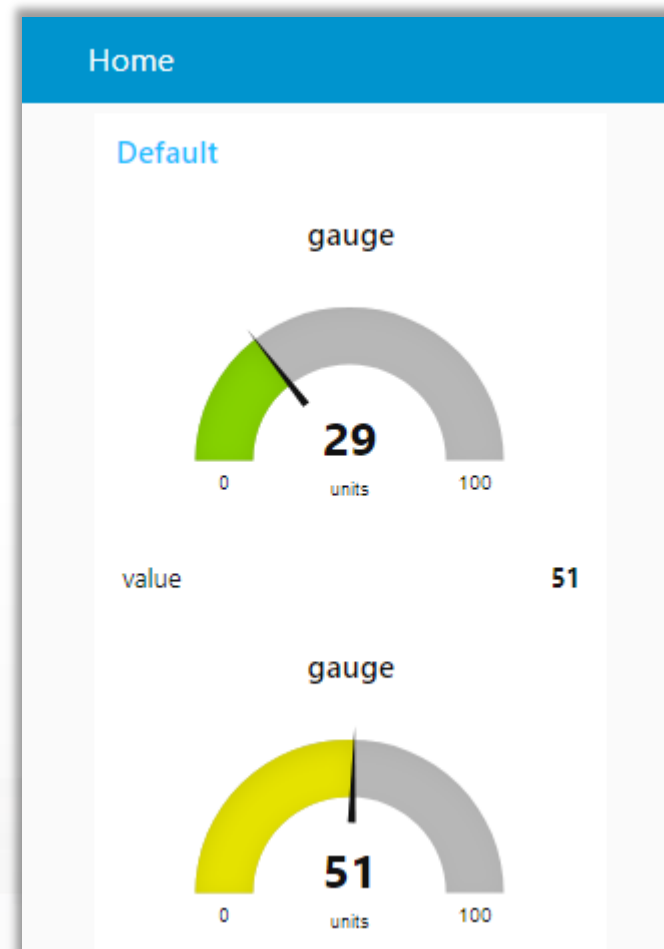
- Click



- Click



- Observe



# Nodes configuration 1/2

**inject**

**Payload**

**Topic**

**Repeat**

every

☒ Inject once at start?

**debug**

**Output**

**to**

**Name**

**switch**

**Name**

**Property**

**random**


**Generate**


**From**

**To**


**Name**

# Nodes configuration 2/2

**gauge** 

**Group** [Home] Default 

**Size** auto




**Type** Gauge 

**Label** gauge

**Value format** {{value}}

**Units** units


**Range** min 0 max 100

**Colour gradient**   

**Sectors** 0 ... optional ... optional ... 100

**Name**

**text** **abc**

**Group** [Home] Default 

**Size** auto

**Label** value

**Value format** {{msg.payload}}

**Layout**

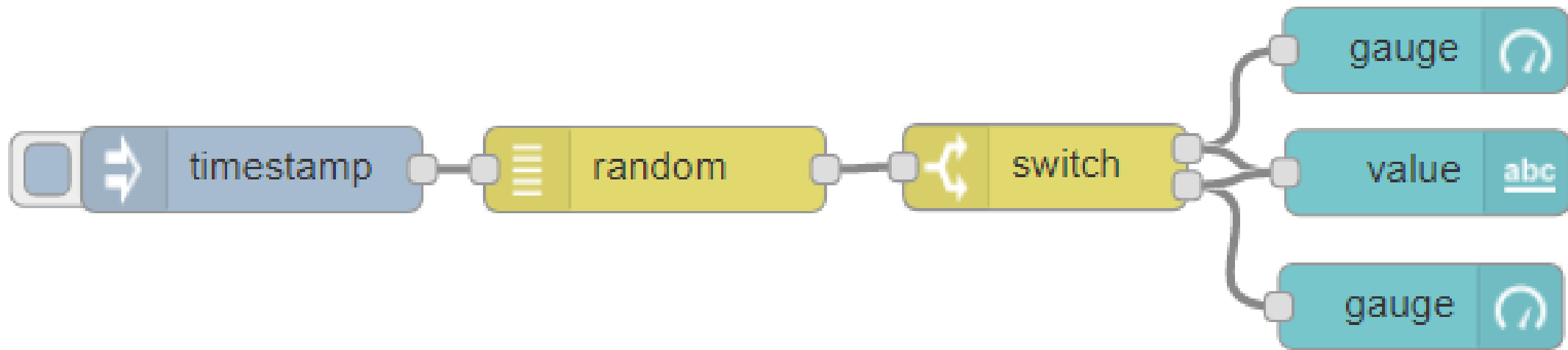
label value label value label value

label value label value

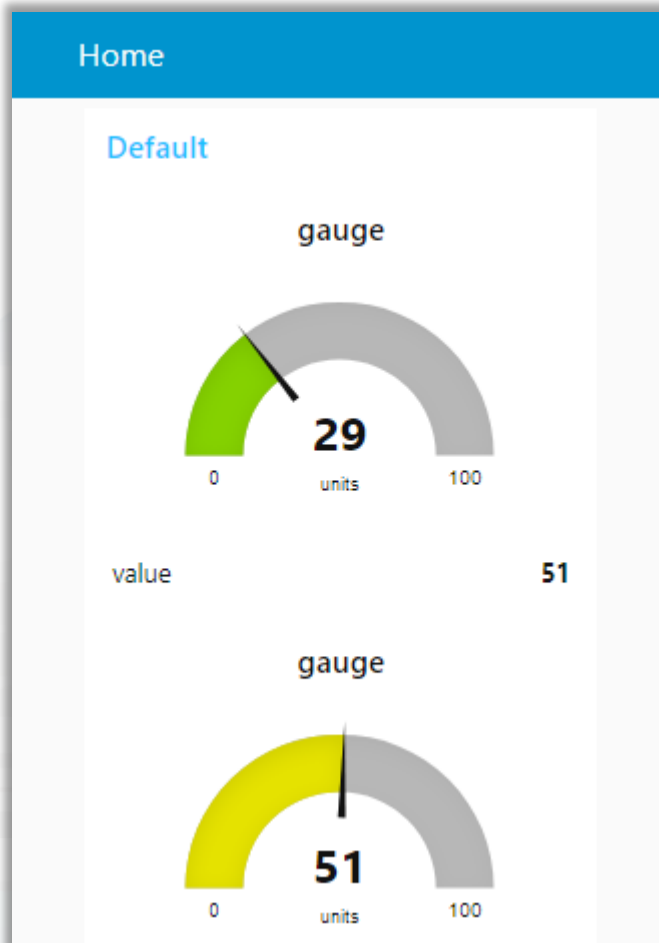
**Name**



# Nodes connections



# Resulting Dashboard



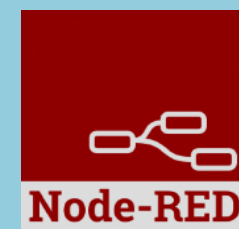
This is a local Node-RED dashboard.  
Simple to be created, very limited for many aspects.

Snap4city dashboards are more :

- Powerful
- Flexible
- Secure
- nicer

TOP

# Node-RED Libraries





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

**Node-RED**[home](#)[about](#)[blog](#)[documentation](#)[forum](#)[flows](#)[github](#)

[+](#)[Sign in with GitHub](#)

[nodes](#)[flows](#)[collections](#)

[recent](#)[downloads](#)[rating](#)

**node-red-contrib-websocket-header**  
Custom Websocket with Header  
v0.5.2 144 node

**node-red-contrib-websocket-header-acknowledge**  
Custom Websocket with Header  
v0.0.1 0 node

**node-red-contrib-zigbee2mqtt**  
Zigbee2mqtt connectivity nodes for node-red  
v2.0.9 1326 4.6 node

**node-red-contrib-nooperation**  
just do nothing.  
v1.0.6 6 node

**@nikolay\_kuropatkin/node-red-contrib-dynamic-file-path**  
A simple node that generate a file by dynamic file path  
v0.0.8 164 5.0 node

**node-red-contrib-mobilealerts**  
This provides a node for receiving Mobile Alerts status infos.  
v3.0.5 71 5.0 node

**node-red-contrib-websocket-header-subscriber**  
Custom Websocket with Header  
v0.0.1 0 node

**@mschaeffler/node-red-asterisk-ami-manager**  
Transfer Asterisk AMI events to json object string representation  
v1.1.2 6 node

**node-red-contrib-sun-position**  
NodeRED nodes to get sun and moon position  
v2.1.1 1259 4.8 node

**node-red-contrib-miio-localdevices**  
Node for Node-Red to control Mi Devices locally via node-mihome (Humidifiers, Purifiers, Heaters, Lights - list of devices to be enlarged).  
v0.4.1 270 1.9 node

**node-red-contrib-cx-alarm-log**  
A Node-RED industrial alarm parser for simple HMI applications.  
v1.1.0 16 5.0 node

**node-red-contrib-message-queue**  
Message queueing for Node-RED  
v1.1.4 11 node

**node-red-contrib-sendmail**  
send emails with help of a local sendmail command.  
v1.0.5 16 node

**node-red-contrib-websocket-header-test**  
Custom Websocket with Header  
v0.0.1 0 node

**node-red-contrib-daylight-rgbw**  
Daylight RGBW Color control for Node RED  
v2.1.3 128 node

1 of 429 [Next](#)

# Load Library from Palette

<https://flows.nodered.org/>

The screenshot shows the Snap4City interface with the Node-RED palette open. The 'Manage palette' button is highlighted with a red circle and a red arrow pointing to it. The palette lists various nodes, including 'node-red', 'node-red-contrib-amqp', 'node-red-contrib-amqp2', 'node-red-contrib-firmware\_official', 'node-red-contrib-ftp', 'node-red-contrib-lwm2m', 'node-red-contrib-md5', 'node-red-contrib-snap4city-developer', and 'node-red-contrib-snap4city-user'.

The top screenshot shows the Node-RED Library website with a search bar and a list of recent nodes and flows. The bottom screenshot shows the 'node-red-contrib-heatweb' node page with a red circle highlighting the 'node info' section, which includes details about the node's version, license, and installation instructions.

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

**Node-RED**[home](#)[about](#)[blog](#)[documentation](#)[forum](#)[flows](#)[github](#)

+[Sign in with GitHub](#)

nodes



flows



collections



recent



downloads


rating


**node-red-contrib-snap4city-milestone**  
Node-Red integration to communicate with Milestone XProtect VMS  
v0.0.3  34  5.0 node

**node-red-contrib-snap4city-developer**  
A description of the available nodes can be found [here](https://www.km4city.org/iot-micro-  
v0.5.13  7  5.0 node

**node-red-contrib-snap4city-user**  
Nodes for Snap4city project, targeted to standard user (no developer)  
v0.9.45  21  4.0 node

**node-red-contrib-snap4city-d3-dashboard-widgets**  
Nodes for Snap4city project for D3 Dashboard Widgets  
v0.0.13  5  5.0 node

**node-red-contrib-snap4city-tunnel**  
Nodes for Snap4city project, targeted to tunneling edge device  
v0.0.3  2 node

**Snap4City module for tunneling on IOT Edge**  
Snap4City module for tunneling on IOT Edge  
disit  NaN collection

1 of 1

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

# Load an IOT application of example



aaa

The screenshot shows the Node-RED web interface. On the left, a 'msg.payload' node is connected to a 'Import s4c' dialog box. The dialog box contains a list of public flows, including 'RecommendationsForYou2', 'SuggestionsForYou', and several 'TC2.7' and 'TC9.2' flows. A red arrow points from the 'Import S4C' option in the 'Library' menu to the 'Import s4c' dialog box. The 'Library' menu is open, showing options like 'Clipboard', 'Library', 'Import S4C', and 'Examples'. The 'Import' option in the top right menu is also circled in red. The 'Import s4c' dialog box has a 'msg.payload' node connected to it, and a 'new flow' button is visible at the bottom.

Import s4c

Public flow: RecommendationsForYou2  
Public flow: SuggestionsForYou  
Public flow: TC2.7 (b) - IOT protocol Telemetry  
Public flow: TC2.7 (a) - IOT protocol Telemetry  
Public flow: TC2.5 - IOT application; IOT Discovery of sen  
Public flow: TC9.2 (JSON) - Managing heterogeneous  
Public flow: TC9.2 (XML) - Managing heterogeneous  
Public flow: TC9.2 (RDF) - Managing heterogeneous  
Public flow: TC9.2 (HTML) - Managing heterogeneous  
Public flow: TC9.2 (CSV) - Managing heterogeneous

msg.payload

Import to

Cancel

Clipboard  
Library  
**Import S4C**  
Examples

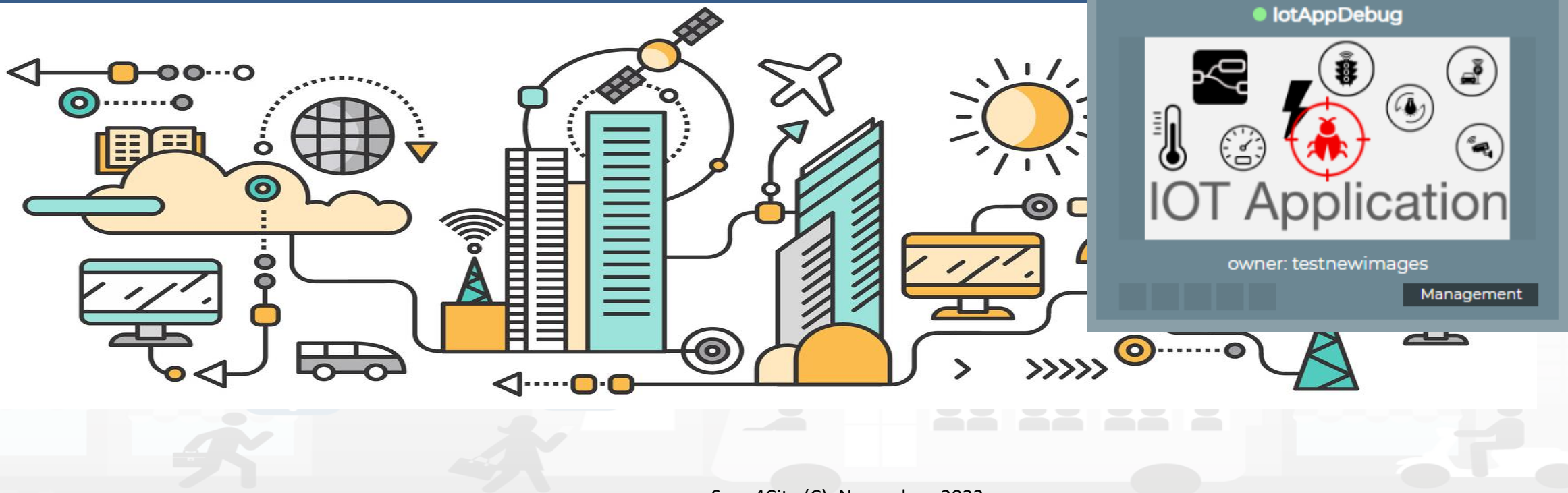
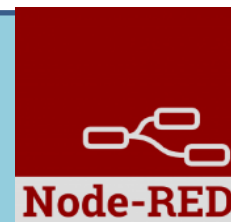
View  
**Import**  
Export  
Search flows  
Configuration nodes  
Flows  
Subflows  
Manage palette  
Settings  
Keyboard shortcuts  
Node-RED website



TOP

# *Snap4City Node-RED*

## *Debug extending Cauldron*



App Maps Google Gmail Snap4City Snap4 Calendar Translate Google Scholar Cita... DISIT DISIT old Facebook DataCenter Trello Km4City major tools Impostazioni YouTube Google Forms News Tutti i preferiti

## Snap4City

Switch To New Layout (Beta)

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

LOGOUT

- My Snap4City.org
- Tour Again
- www.snap4solutions.org
- ダッシュボード
- Dashboards (Public)
- My Dashboards in All Org.
- Dashboards of My Organization
- My Dashboards in My Organization
- My Data Dashboard Dev Kibana
- My Data Dashboard Kibana
- Extra Dashboard Widgets
- Notifier
- Data Management, HLT
- Knowledge and Maps
- Processing Logics / IOT App
  - Processing Logics / IOT App
  - MicroServices for Proc.Logic/IOT App
  - MicroServices from DataAnalytic
  - IOT MicroServices for Final Users
  - IOT MicroServices for Developers
  - DOC: Processing Logic/IOT App
  - How to Develop Proc.Logic / IOT App
  - Create A MicroService from RestCall
- Entity Directory and Devices
- Resource Manager
- Development Tools

### advanced3debug

Node-RED

filter nodes

Flow 4 Flow 3 Flow 1 R - Python

subflows

- getAccessTok en
- getAccessTok en (2)
- getAccessTok en (3)
- getAccessTok en (2) (2)

common

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

function

- function
- switch
- change

timestamp

debug 6

function 4

function 5

chart

debug 7

info

Search flows

Flows

- Flow 4
- Flow 3
- Flow 1
- R - Python

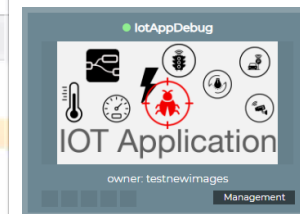
Subflows

Global Configuration Nodes

Flow 3

Flow "ff58664fe453ec66"

Import a flow by dragging its JSON into the editor, or with `ctrl-i`





TOP

***IoT App = Node-RED + Snap4City***  
***IoT App === Proc.Logic***



DATA GATHERING  
AND CITY DATA  
KNOWLEDGE  
MANAGEMENT



IOT/IOE DEVICES  
AND NETWORKS

IOT APPLICATIONS,  
THE LOGIC AND  
THE SMARTNESS

ADVANCED  
SMART CITY API,  
MICROSERVICES,  
SNAP4CITY API

SNAP4CITY  
LIVING LAB FOR  
COLLABORATIVE  
WORK

DATA ANALYTICS,  
BUSINESS  
INTELLIGENCE,  
WHAT AND  
SIMULATION

TWITTER  
VIGILANCE: SOCIAL  
MEDIA ANALYSIS

SNAP4CITY  
ARCHITECTURE AND  
ECOSYSTEM. OPENED  
TO DEVELOPERS  
AND CITIES

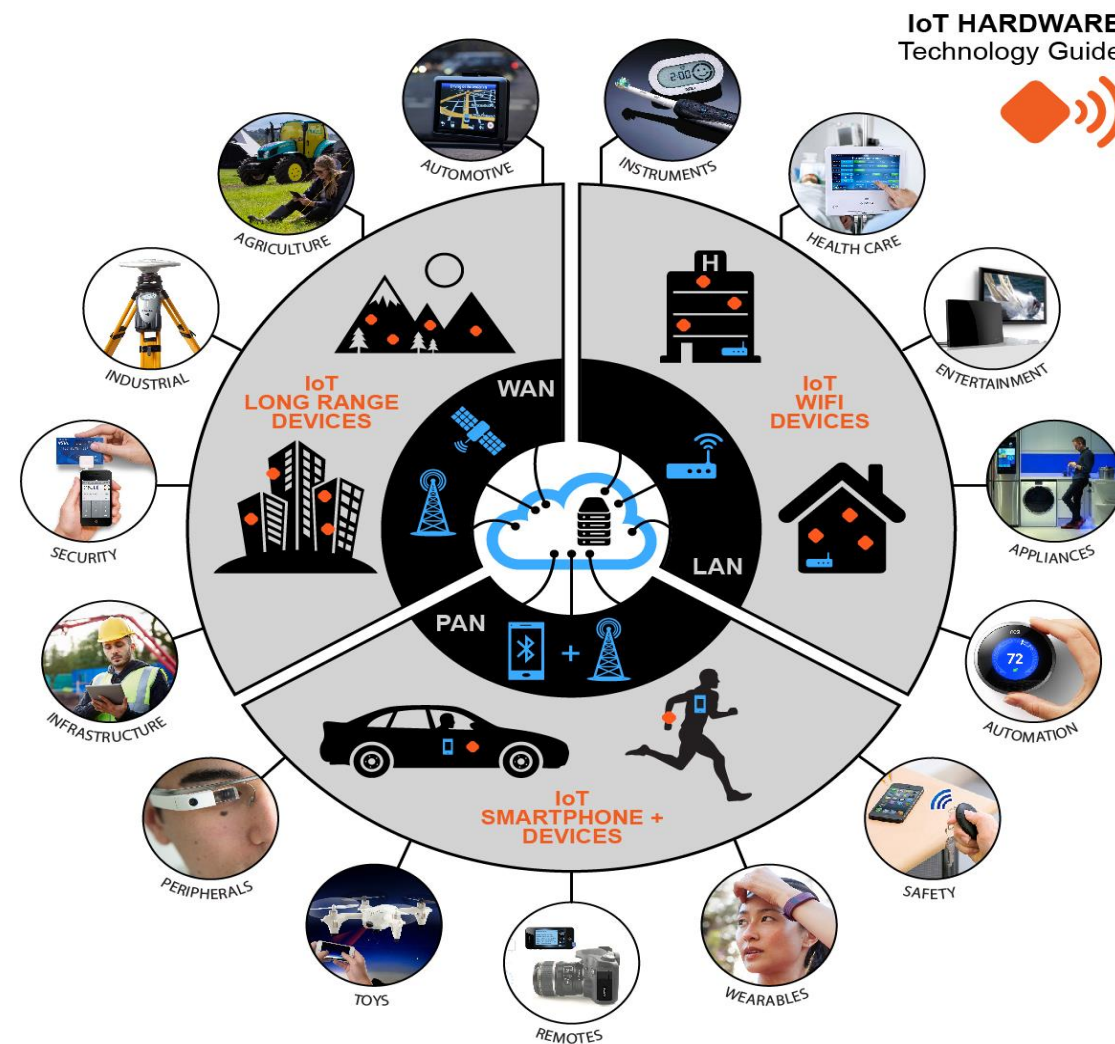
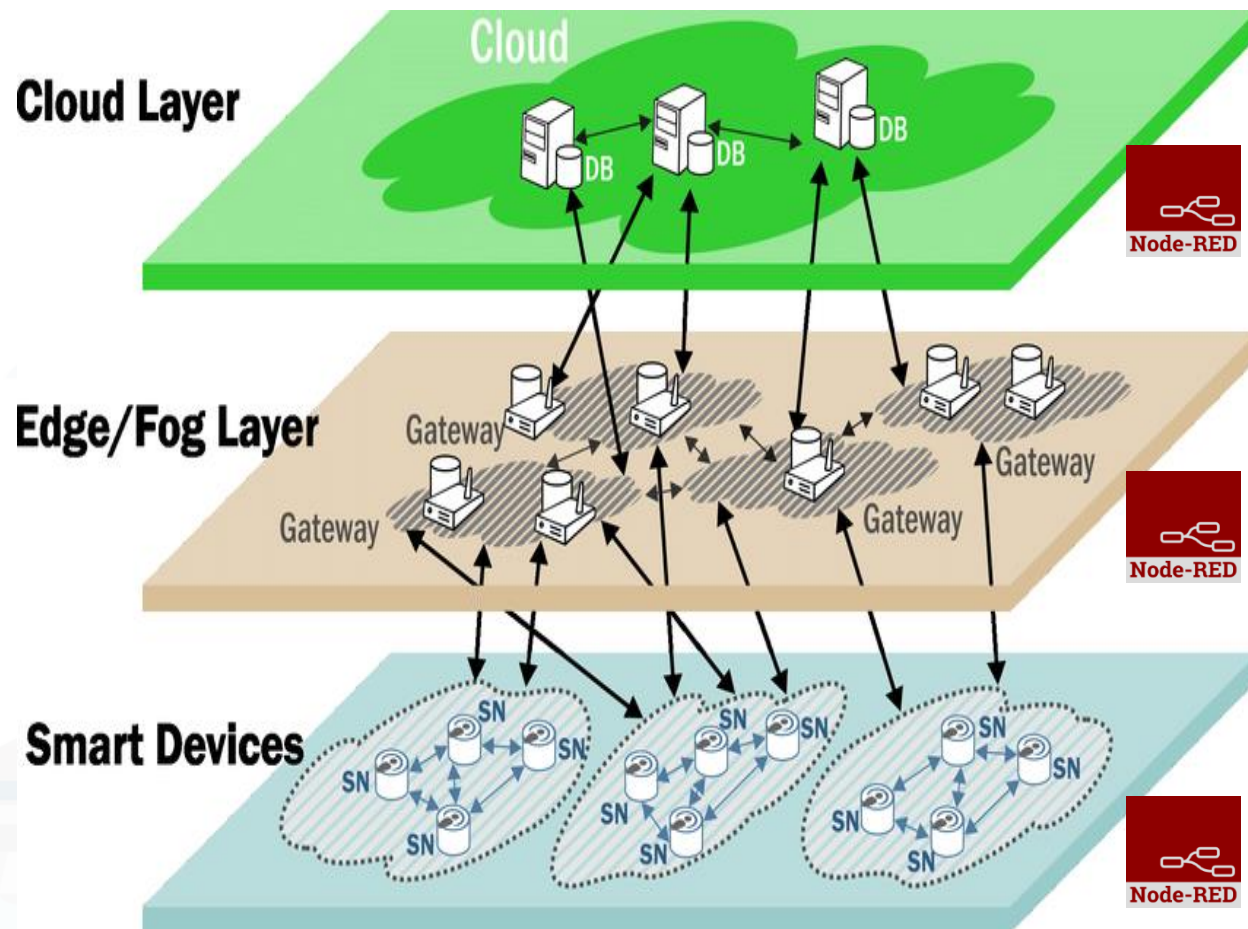
SNAP4CITY  
AND KM4CITY  
PROJECT

HOW TO ADOPT  
SNAP4CITY, AND  
OUR ROADMAP





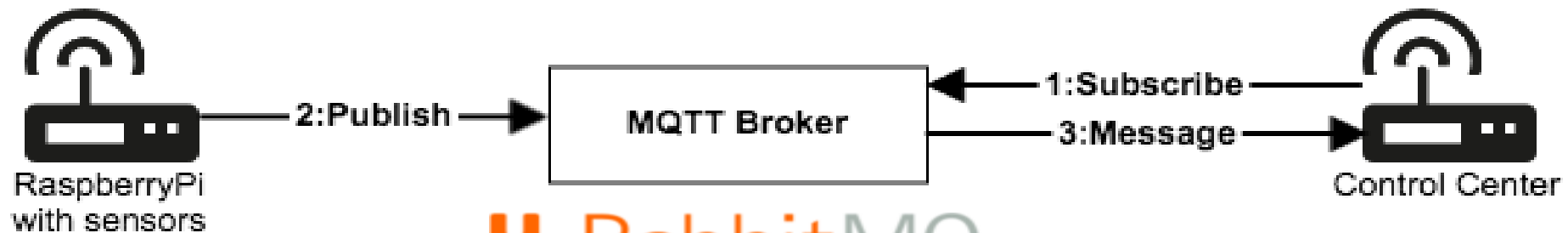
# WoT: Cloud vs Fog/Edge Computing





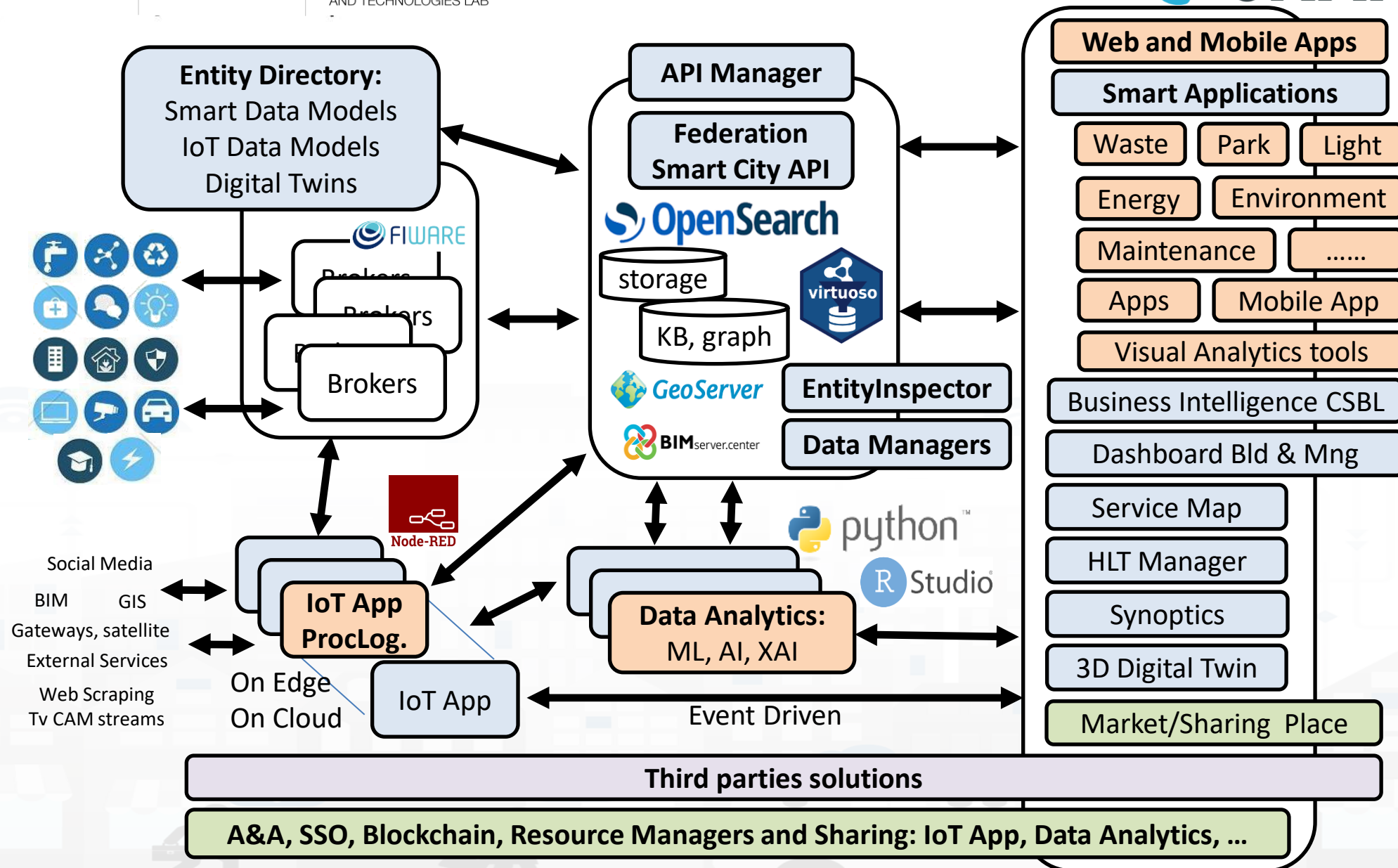
# Brokers

	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



IOT Device

# What About IoT Devices, Time Series



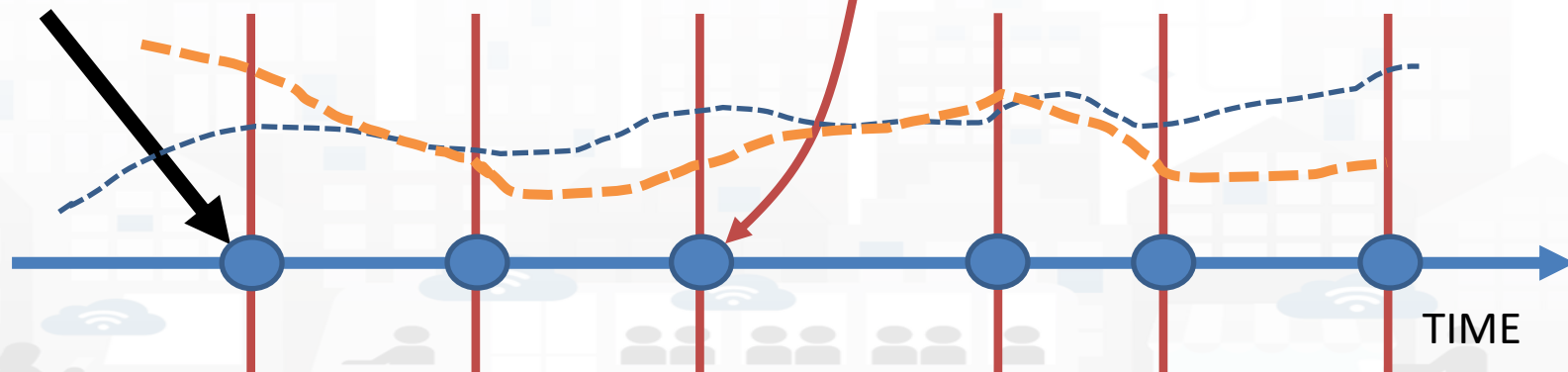
Entity: IOT  
Device

Sends a  
message

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

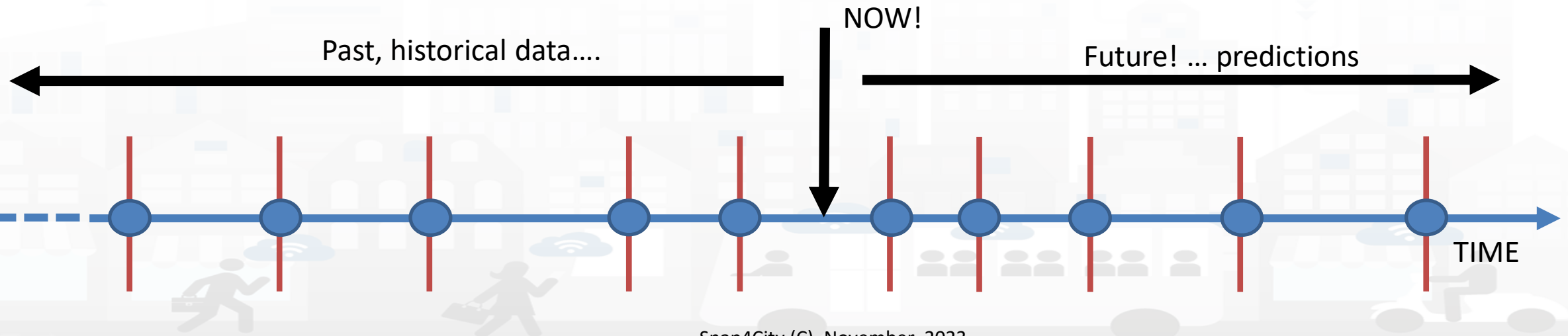
- A set of data coming from an IoT Device with multiple sensor become a time series of values for devices.
  - For example: taking a new measure every 10 minutes (**Red Lines**)
  - Non regular rates can be valid data as well.
- Each new measure in Snap4City is conventionally time located in «**dateObserved**»,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

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



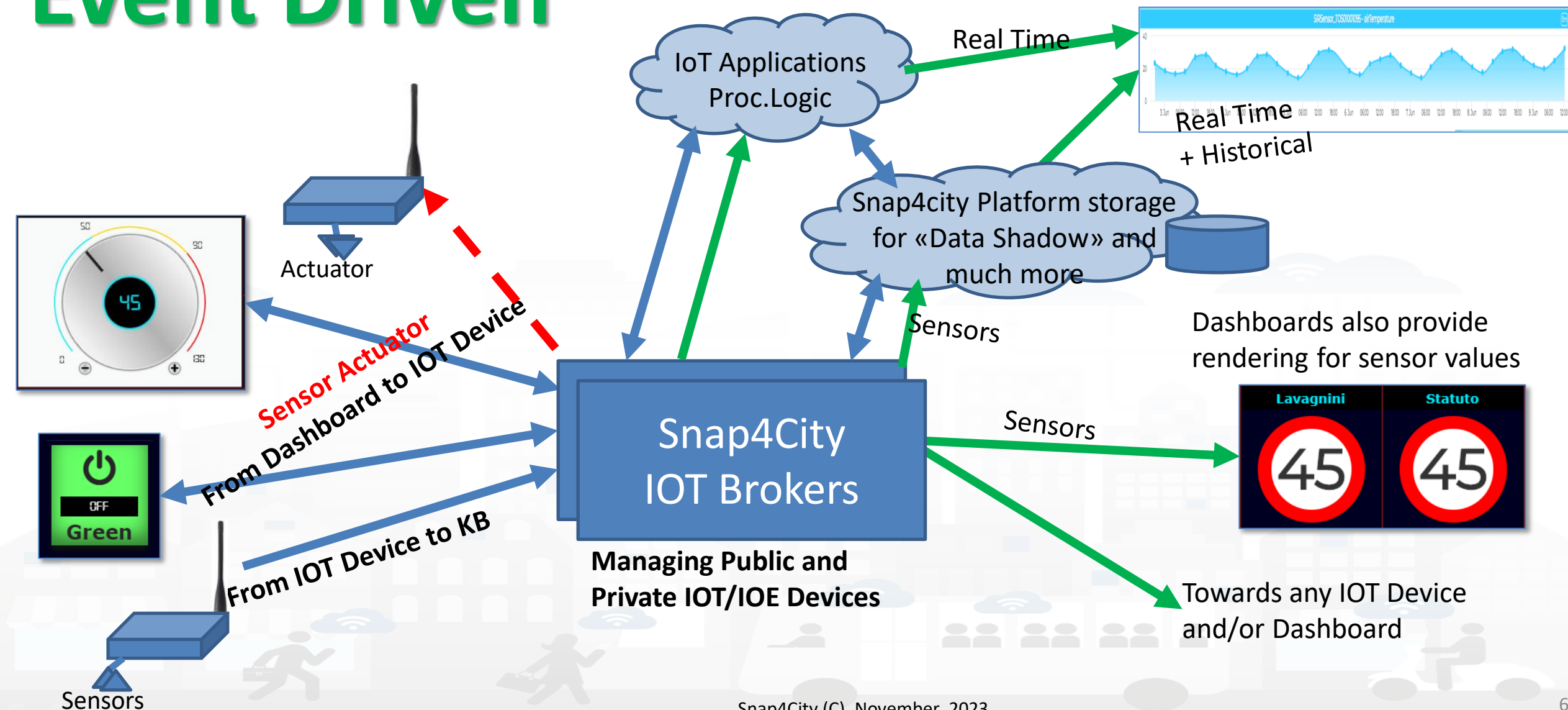
# Time Series: they are data streams

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

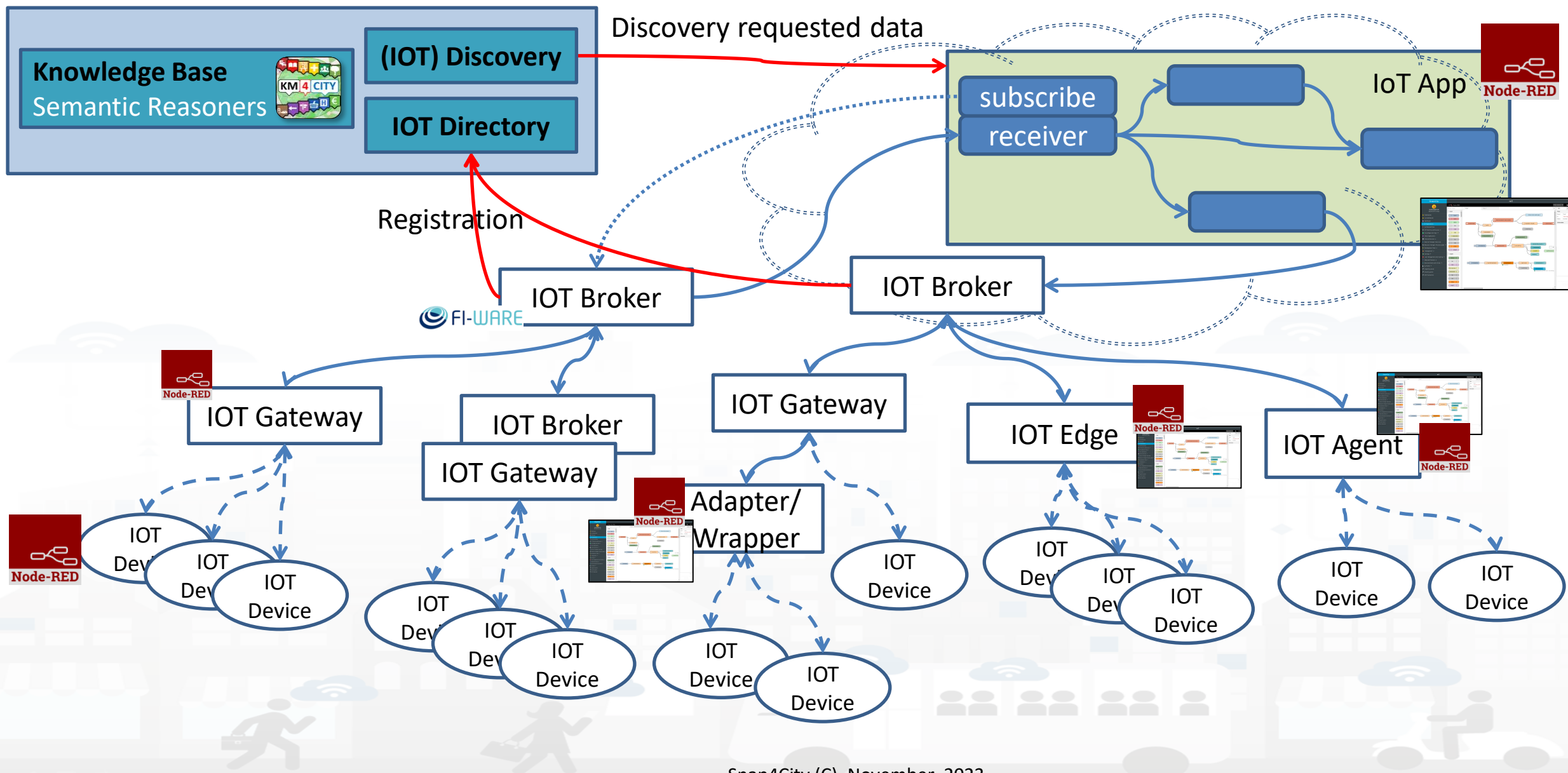




# Event Driven



# IoT Network



# IoT/IIoE Protocols



## Communication Patterns



### Discovery

Discover, register and "thrust" new devices on the network

Registration



### Telemetry

Information Flows From device to another system for conveying status changes in the device

Push



### Inquiries

Requests from devices looking to gather required information or asking to initiate activities



### Commands

Commands from other systems to a device or a group of devices to perform specific activities

Bulk action



### Notifications

Information flows from other systems to a device or a group for conveying status changes in the world

- MQTT
- HTTP(s)
- AMQP
- COAP
- NGSI
- OneM2M
- WebSocket
- S
- .....
- Etc.

## Note on Communication patterns

- Not all Communication Patterns are supported by all Protocols
- Not all Communication Patterns are supported by all Platforms
- Protocols implement Patterns, + formats, + sequences, etc.
- They are referred at level of communications
  - IoT Device  $\leftarrow \rightarrow$  IoT Gateway  $\leftarrow \rightarrow$  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,





## 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

[+ CREATE NEW](#)

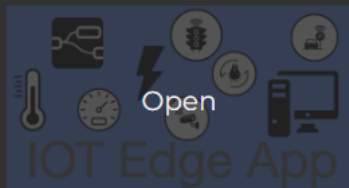


Prev 1 2 3 Next

Filter



● 2020-07-28T10:20

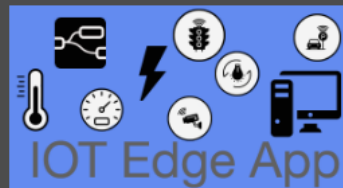


My own



Management

● 2020-07-28T12:32

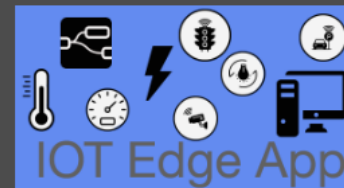


My own



Management

● 2020-08-18T08:38

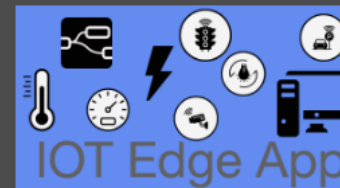


My own



Management

● 2021-01-19T16:25

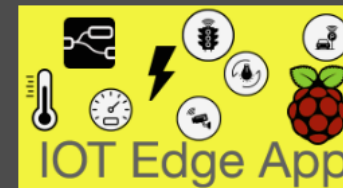


My own



Management

● 2021-08-21T13:26

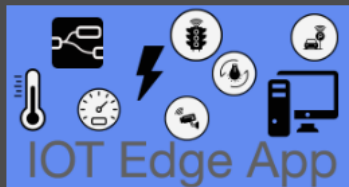


My own



Management

● 2022-05-28T14:50



My own



Management

● actionurltest



My own



Management

● Alarm Management



My own



Management

● corona1

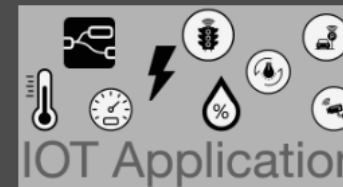


My own



Management

● coronaR



My own



Management

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

Dashboards of My Organization

My Dashboards in My Organization

My Data Dashboard Dev Kibana

Extra Dashboard Widgets

Data Management, HLT

Knowledge and Maps

Processing Logics / IOT App

Processing Logics / IOT App

MicroServices for Proc.Logic/IOT Apps

MicroServices from DataAnalytic

IOT MicroServices for Final Users

IOT MicroServices for Developers

DOC: Processing Logic/IOT App

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



Prev 1 2 3 ... 9 Next

Filter



Create new

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

2018-09-14T04:44

IOT Edge App

owner: badii

Management

2018-09-21T03:19

IOT Edge App

owner: panesi

Management

2018-10-19T16:07

IOT Edge App

owner: pb3

Management

2018-10-19T17:17

IOT Edge App

owner: pb3

Management

2018-10-22T11:57

IOT Edge App

owner: semolarudy

Management

application

IOT Application

owner: tester5

Management

Bib APP

IOT Application

owner: semolarudy

Management

ChargingStations

IOT Application

owner: comunedashres

Management

Deprecated - SiiMobilityControlRoom

IOT Application

owner: badii

Management

SamsungGalaxyS4Barcode

IOT Edge App

owner: badii

Management

esercitazione

IOT Application

owner: tester2

Management

lot-App

IOT Application

owner: tester14

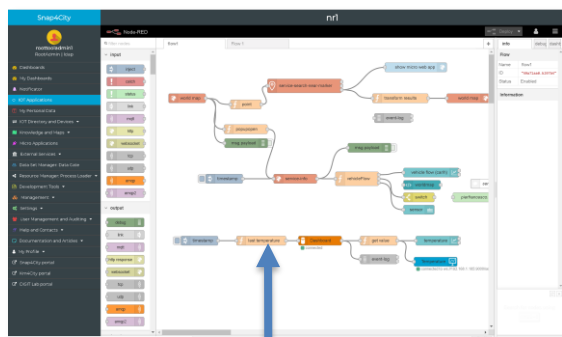
Management

# IOT Application Listing, they can be

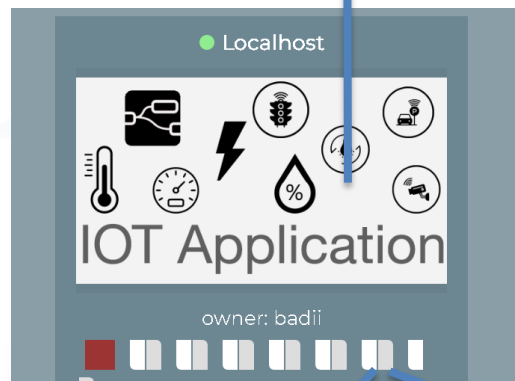
- Basic (white)
- Advanced (red)
- IOT Edge
  - Raspberry Pi
  - Android
  - Win/Linux
- Data Analytic (Plumber)
- Web Scraper (Portia)

# IOT Applications Listing

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

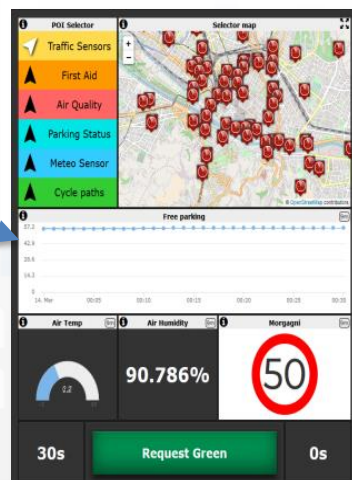
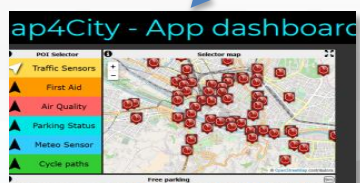


EDIT IOT APP



VIEW

EDIT



IOT App title

Ownership of the IOT App

Click the icon to edit the IOT App

Click to open the Node-RED IOT App dashboard

Click to edit IOT App properties

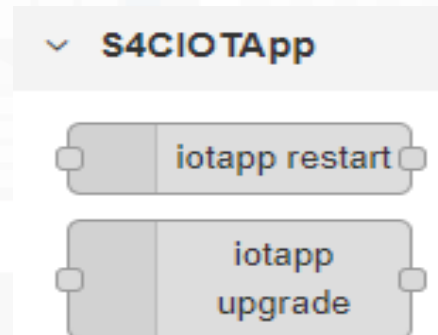
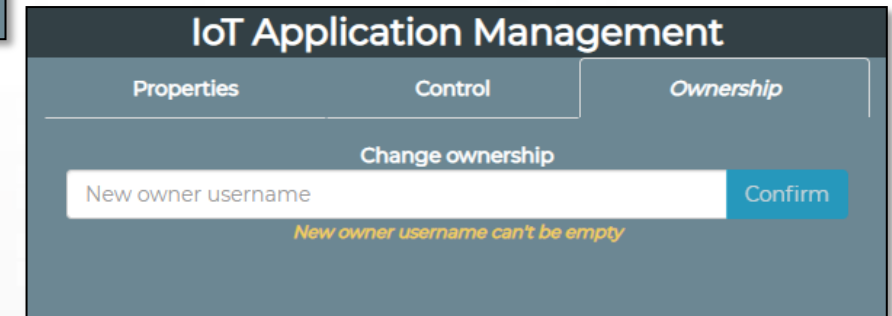
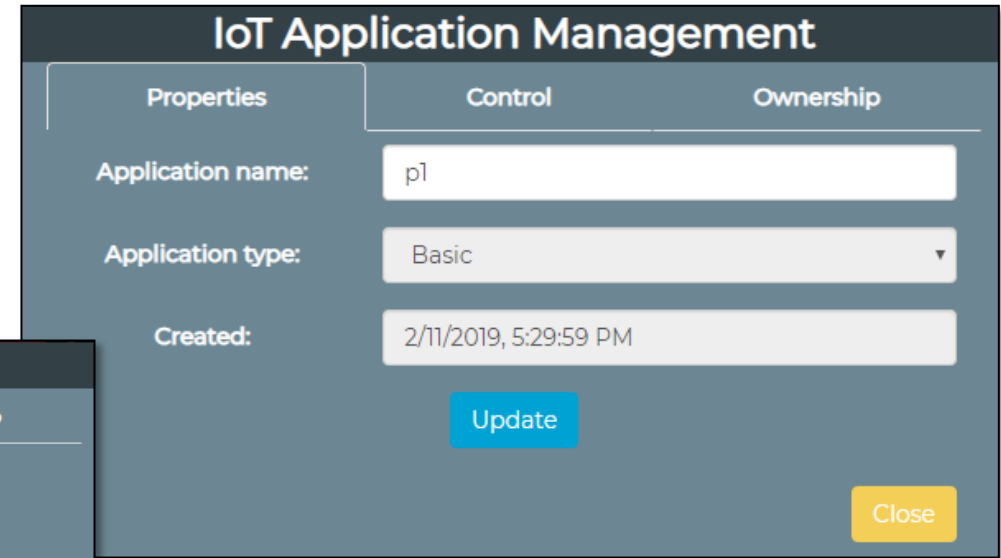
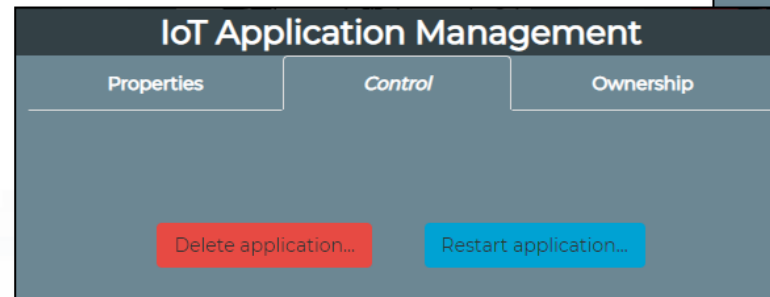
Click to view the Snap4City Dashboard

Click to edit the Snap4City Dashboard



# 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

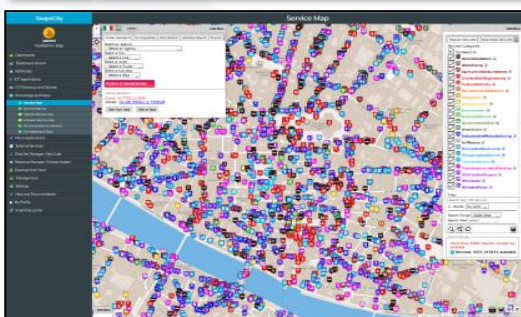


Automating restart  
and update

# IOT Applications Development

IOT Discovering

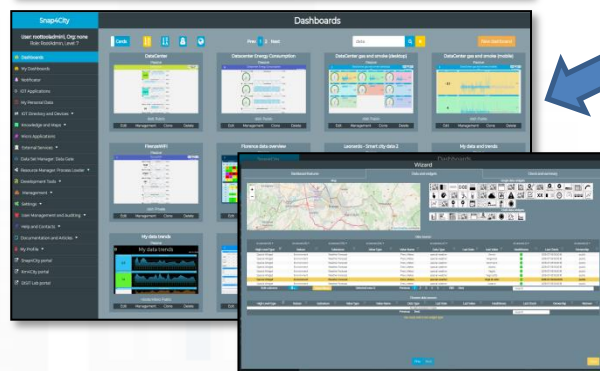
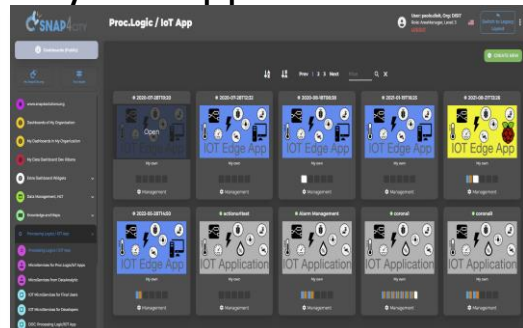
MicroServices collections



ServiceMap Discovery  
Knowledge Base, Km4City

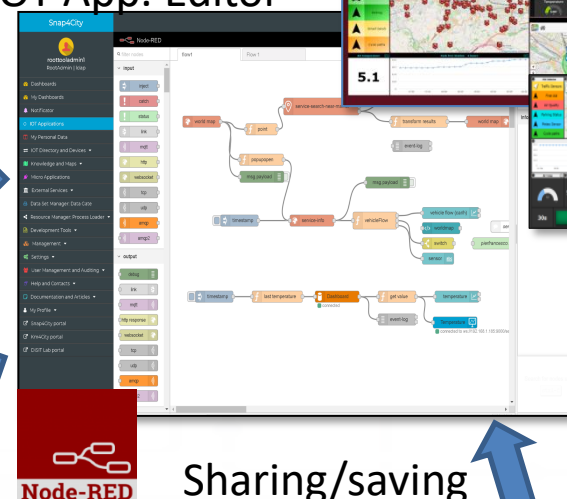


My IOT Applications



Dashboard Collection,  
Editor and Wizard

IOT App. Editor



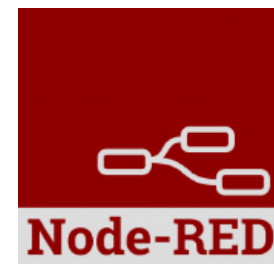
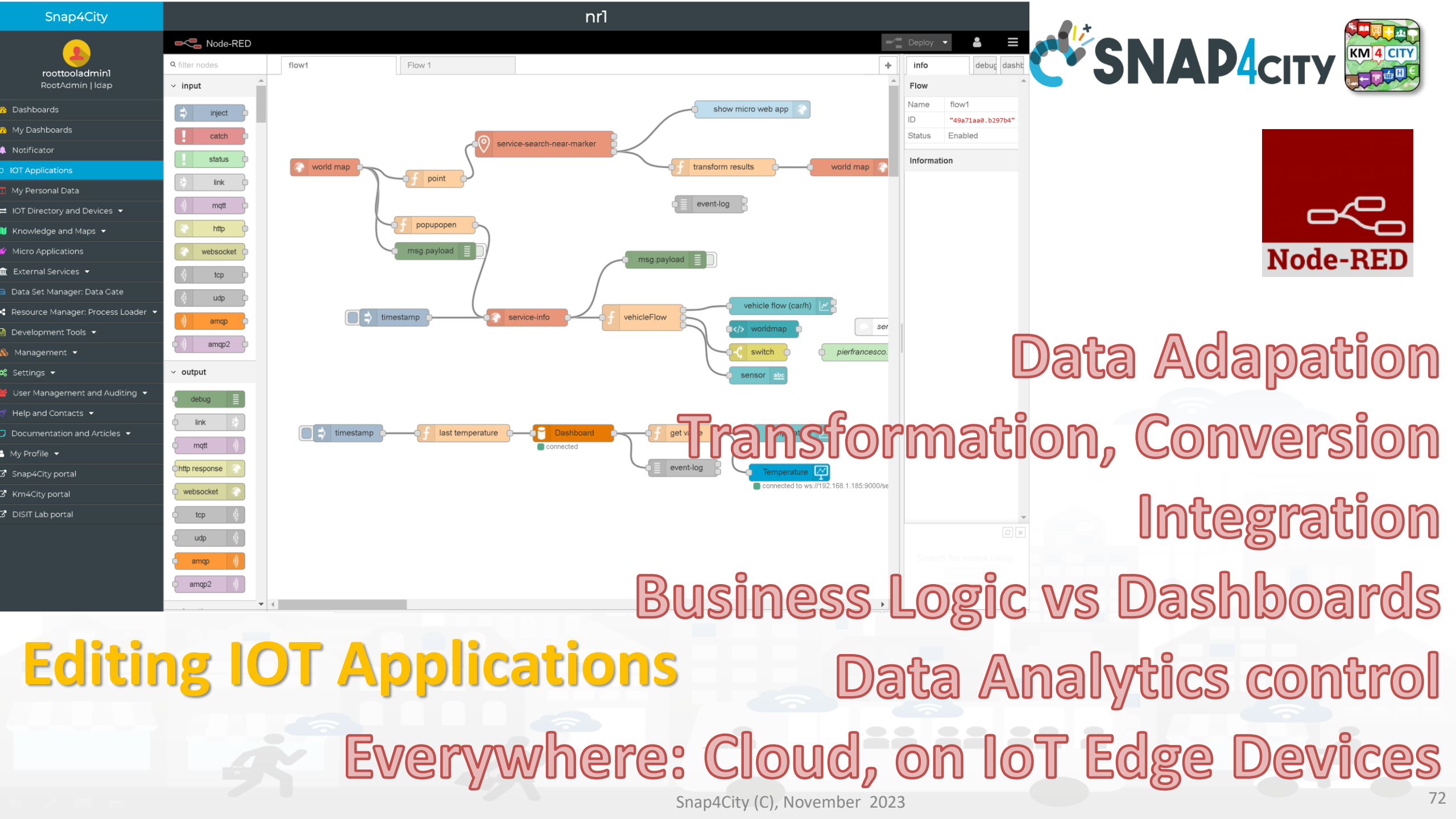
Sharing/saving  
reusing IOT App



Resource Manager

Generating IOT App  
With Dashboard





Data Adapation  
Transformation, Conversion  
Integration

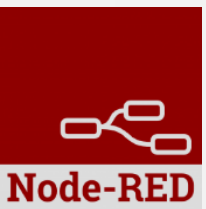
Business Logic vs Dashboards

Editing IOT Applications

Data Analytics control

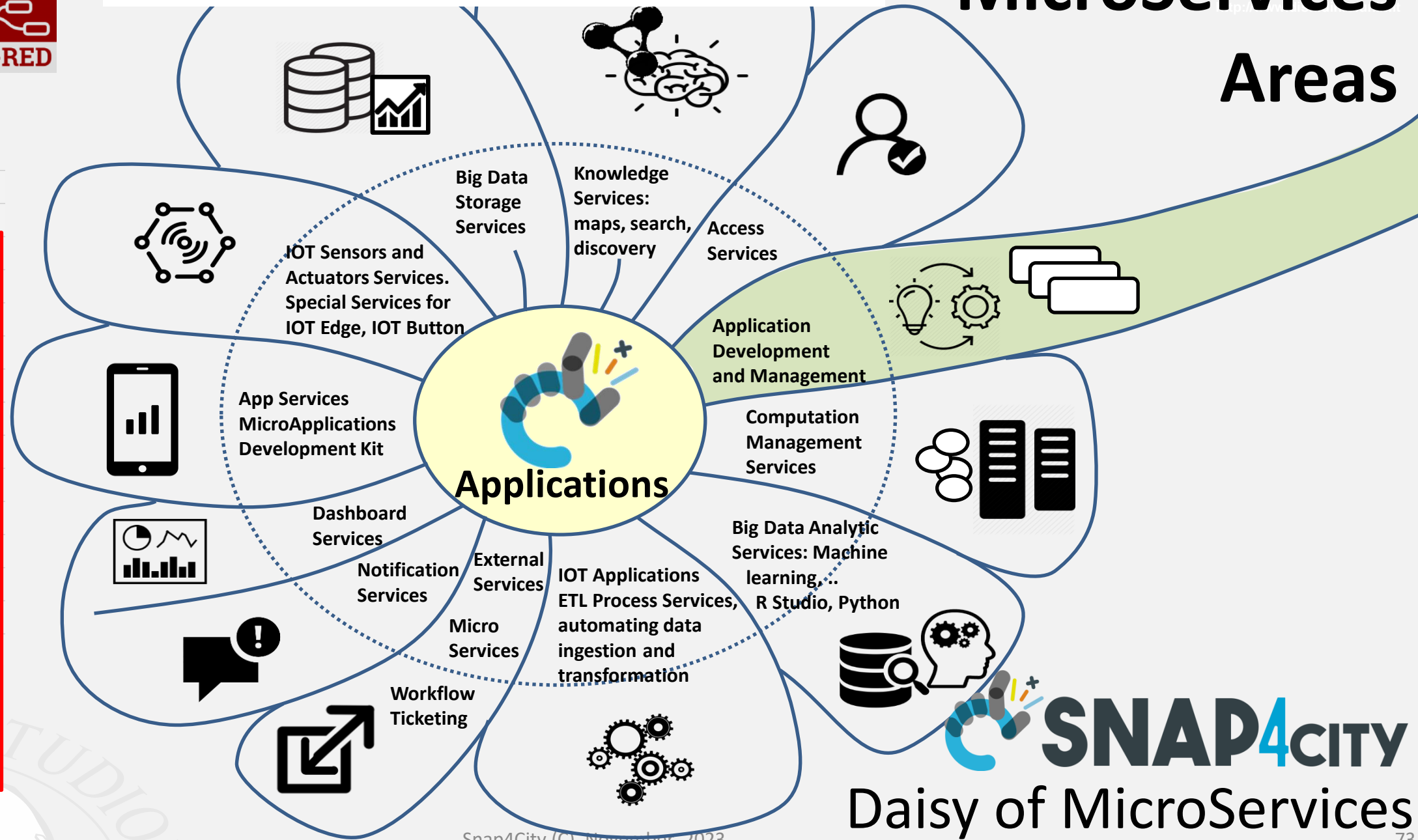
Everywhere: Cloud, on IoT Edge Devices

- > 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**



common

function

network

input

output

sequence

parser

storage

social

advanced

Advanced FTP

location

NGSI

Iwm2m

S4C SearchDev

S4C Utility

S4C Mapping

S4C Management

S4C DataAnalytic

S4C BigData

S4C IoT App

S4C OpenMaint

S4C IoT

S4C WhatIf

S4C Search

S4C Data

S4C KPIData

S4C Dashboard

S4C Sigfox

S4C LogDev

S4C View

S4C Social

dashboard

time

**S4C SearchDev**

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

full text search within gps area

full text search near gps position

full text search exp

event search dev

event search exp

event search within wkt area

event search within gps area

event search near gps position

address search near gps position

geometry search near gps position

address poi search by text

address poi search by text exp

address poi search by text near gps position

bus routes search

bus routes search near gps position

bus routes search within gps area

bus routes search within wkt area

bus routes

point within polygon

routing

heatmap picker

coordinates to address

service info

edge-tunnel-to-cloud

**S4C Mapping**

- service info mapped
- mapping
- set mapping

tpl routes

tpl stops

**S4C Utility**

- service info dev
- distance from coordinates

get job detail

get triggers of job

get job group names

get trigger group names

get paused trigger groups

get job fire times

get system status

trigger job

pause all

pause trigger

pause triggers

resume all

resume job

resume jobs

resume trigger

resume triggers

notifier history events

**S4C DataAnalytic**

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

**S4C Search**

- service search near marker
- service search within circle
- service search within polygon
- service search along path
- full text search within circle
- full text search within polygon
- full text search along path
- full text search usr
- event search near marker
- event search within circle
- event search within polygon
- event search along path
- event search usr
- address search near marker
- geometry search near marker
- address poi search by text usr
- address poi search by text near marker
- address poi search by text within circle
- address poi search by text within polygon
- value type search near marker
- value type search within circle
- value type search within polygon
- value type search along path

event search within polygon

event search along path

event search usr

address search near marker

geometry search near marker

address poi search by text usr

address poi search by text near marker

address poi search by text within circle

address poi search by text within polygon

value type search near marker

value type search within circle

value type search within polygon

value type search along path

tpl routes by agency

tpl routes by line

tpl stops by route

tpl stop timeline

recommendation within circle

value type search near marker

value type search within circle

value type search within polygon

value type search along path

**S4C Data**

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

get my data

get my delegator

get my delegated

get my activity

insert

datagate search

datagate create

portia crawler

**S4C IoT App**

- iotapp restart
- iotapp upgrade
- ownership

event search near marker

event search within circle

bus routes search near marker

bus routes search within circle

bus routes search within polygon

tpl agencies

tpl lines

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

## AND: From Resource Manager

We suggest also to install:

Snap4City (C), November 2023



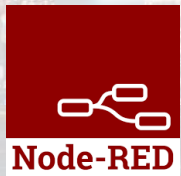
# Standards and Interoperability (6/2023)



## Compliant with:

- **IoT:** NGSI V2/LD, LoRa, LoRaWan, MQTT, AMQP, COAP, OneM2M, TheThingsNetwork, SigFOX, Libelium, IBIMET/IBE, EnOcean, Zigbee, DALI, ISEMC, Alexa, Sonoff, HUE Philips, Tplink, BACnet, TALQ, Protocol Buffer, KNX, OBD2, Proximus, ..
- **IoT model:** FIWARE Smart Data Model, Snap4City IoT Device Models
- **General:** HTTP, HTTPS, TLS, Rest Call, SNMP, TCP, UDP, SOAP, WSDL, FTP, FTPS, WebSocket, WebSocket Secure, GML, WFS, WMS, RTSP, ONVIF, AXIS TVCam, CISCO Meraki, OSM, Copernicus, The Weather Channel, Open Weather, OLAP, VMS Milestone, ....
- **Formats:** JSON, GeoJSON, XML, CSV, GeoTIFF, OWL, WKT, KML, SHP, db, XLS, XLSX, TXT, HTML, CSS, SVG, IFC, XPD, OSM, Enfuser FMI, Lidar, 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>

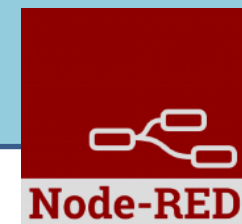




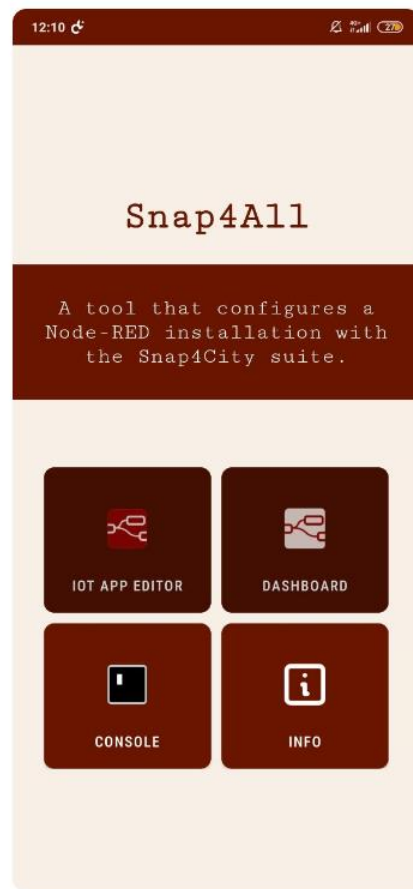
TOP

# Snap4All Mobile App

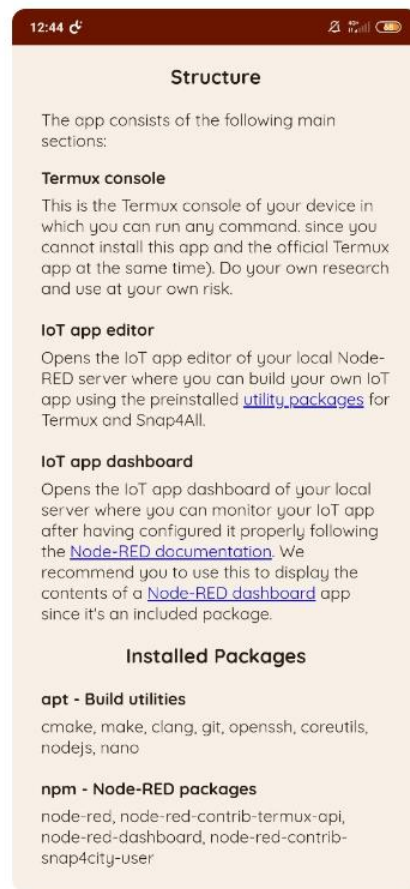
## Node-RED on Android



# Snap4All mobile app for Android



(a) Home



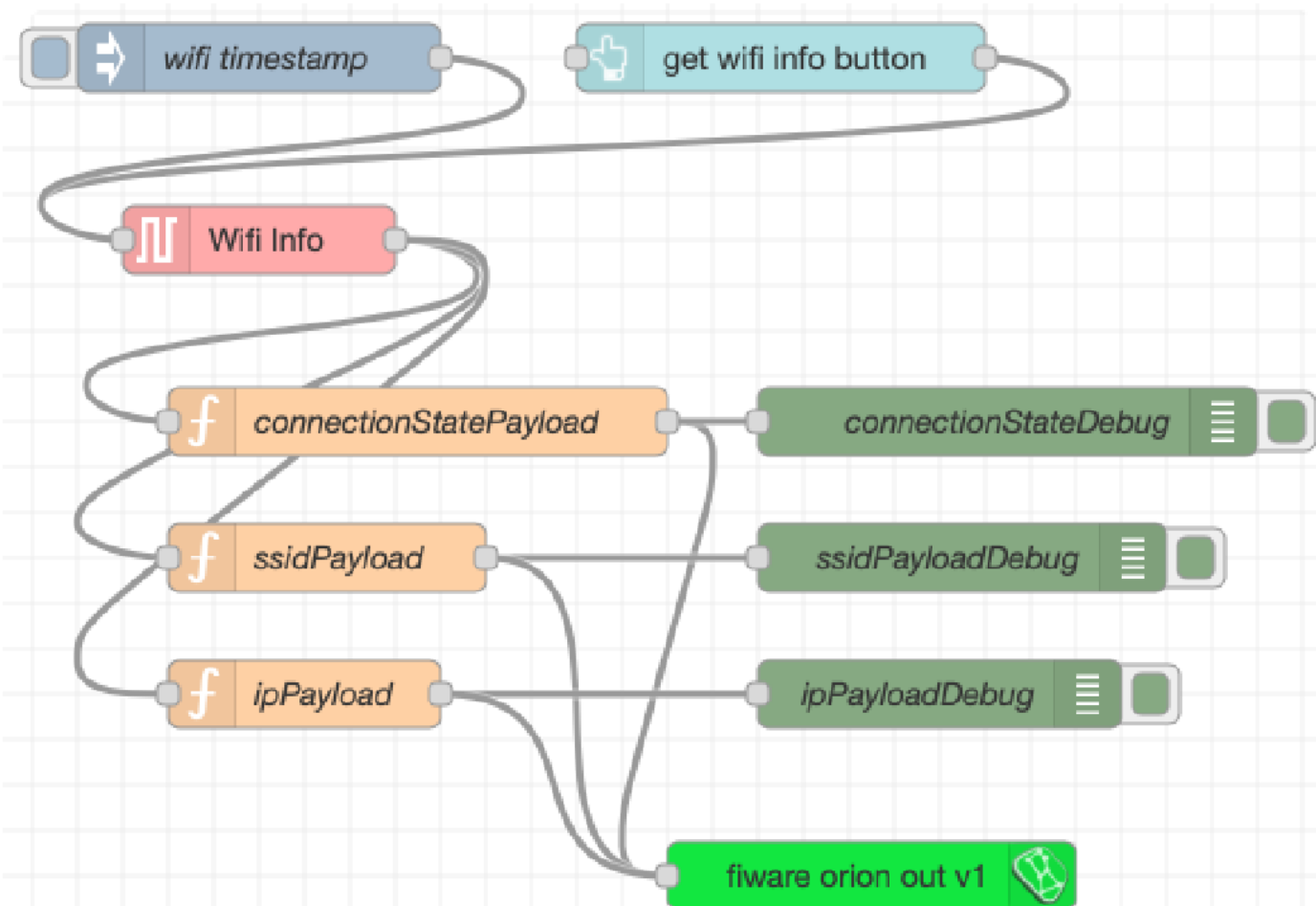
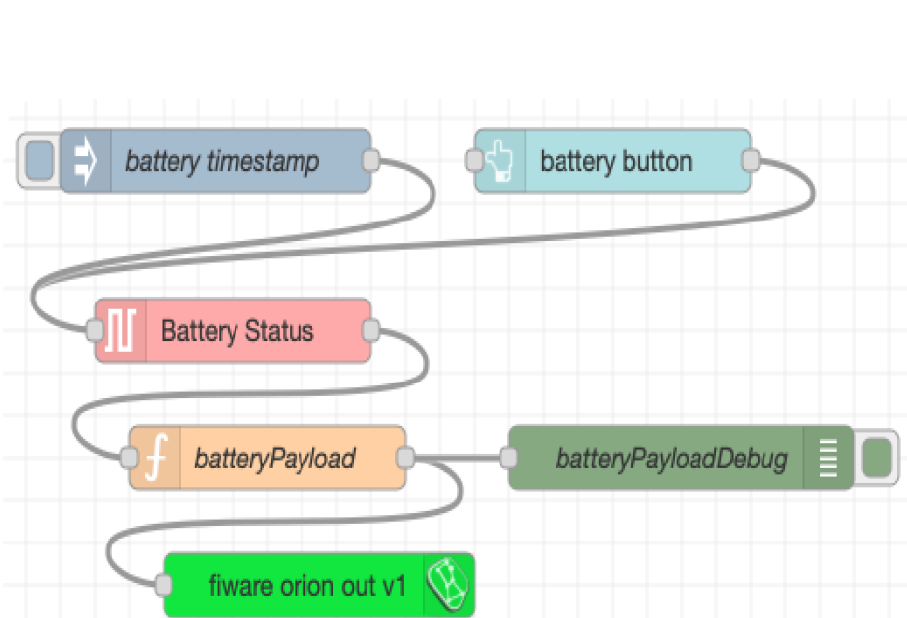
(b) Info



(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>



TOP

# Examples of IoT App/Proc.Logic for Smartening Solutions



DATA GATHERING  
AND CITY DATA  
KNOWLEDGE  
MANAGEMENT

FORGING &  
MANAGING OPEN  
AND FLEXIBLE WEB  
AND MOBILE APPS

IOT APPLICATIONS  
VS IOT ED  
VICES

IOT/IOE DEVICES  
AND NETWORKS

IOT APPLICATIONS,  
THE LOGIC AND  
THE SMARTNESS

ADVANCED  
SMART CITY API,  
MICROSERVICES,  
SNAP4CITY API

SNAP4CITY  
LIVING LAB FOR  
COLLABORATIVE  
WORK

SNAP4CITY FOR  
BEGINNERS

SNAP4CITY  
ARCHITECTURE AND  
ECOSYSTEM. OPENED  
TO DEVELOPERS  
AND FUNDHOLDERS

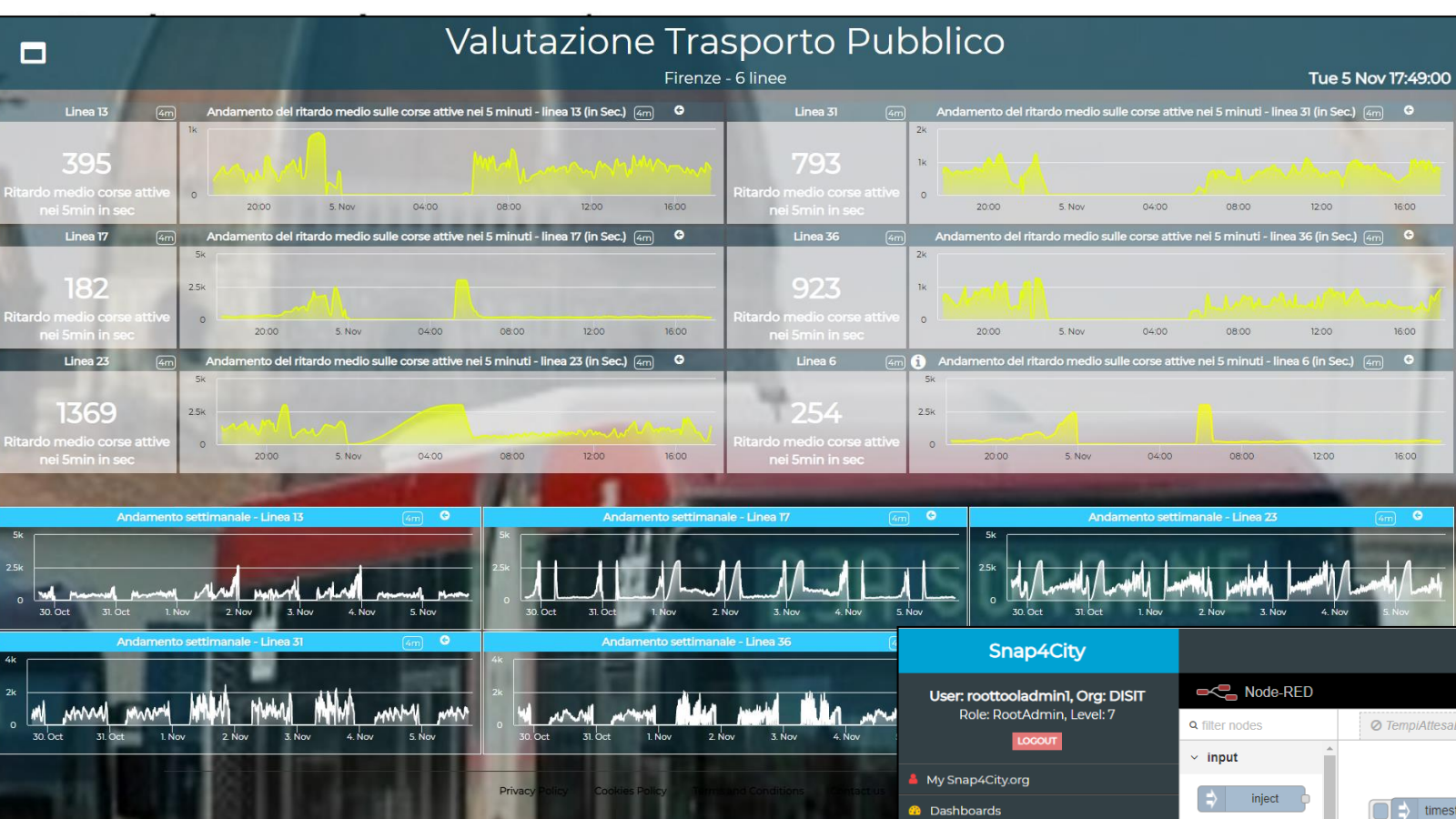
TWITTER  
VIGILANCE: SOCIAL  
MEDIA ANALYSIS

SNAP4CITY  
AND KM4CITY  
PROJECTS

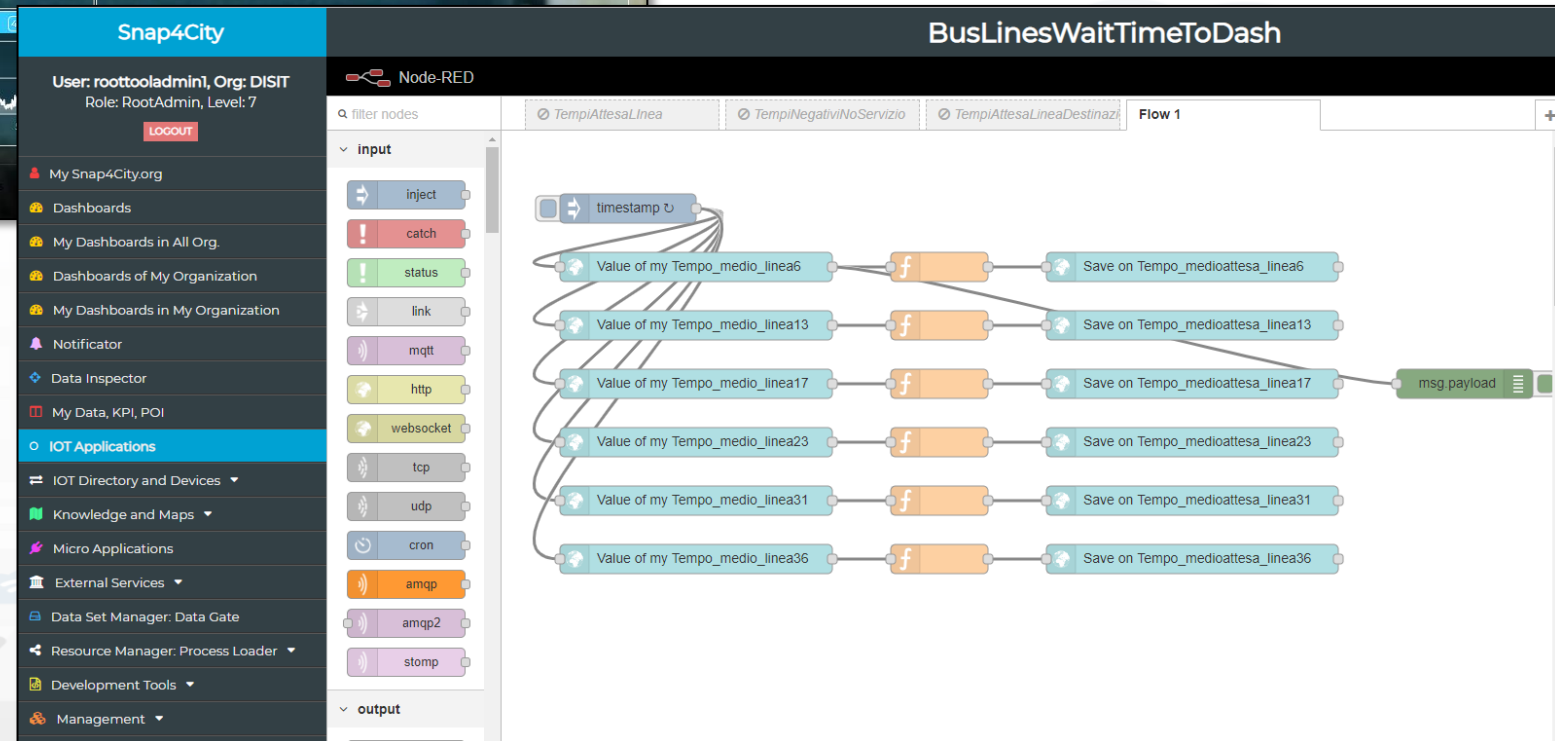
HOW TO ADOPT  
SNAP4CITY  
OUR ROAD







## Estimation of the mean waiting time at bus stops



TOP

# *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

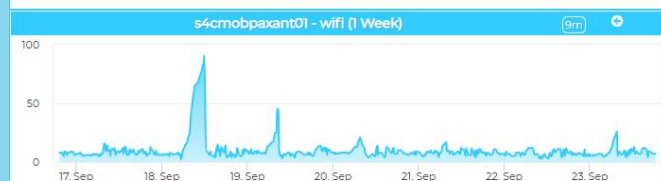
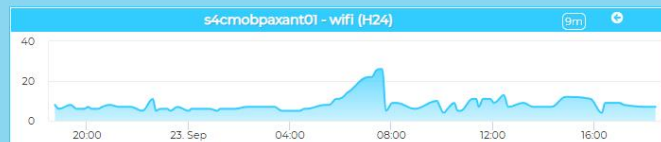


<https://www.snap4city.org/456>

# Programmable PAX counting

## Mobile PAXCounter 01 in Antwerp

Mon 23 Sep 18:39:46



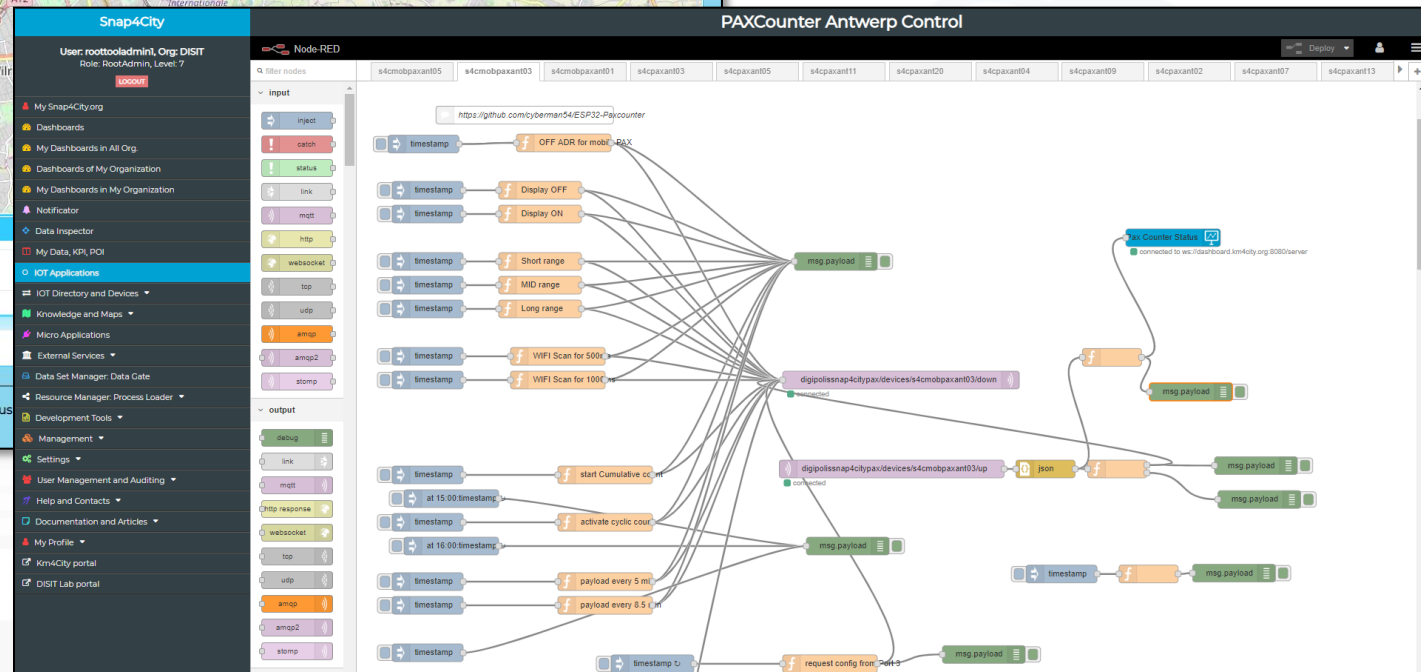
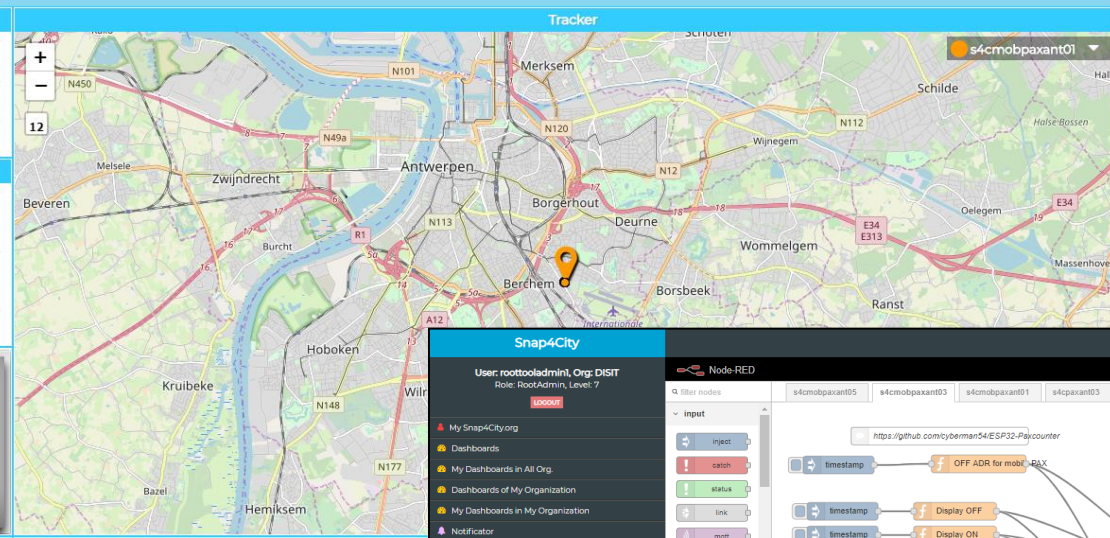
Begin 3:00  
Finish 5:30

**Activate**

CUMULATIVE MODE OFF

Cumulative Mode Active from 2019-09-23T03:00:00.000Z and 2019-09-23T05:30:00.000Z

Device in Cumulative Mode OFF



Antwerp



TOP

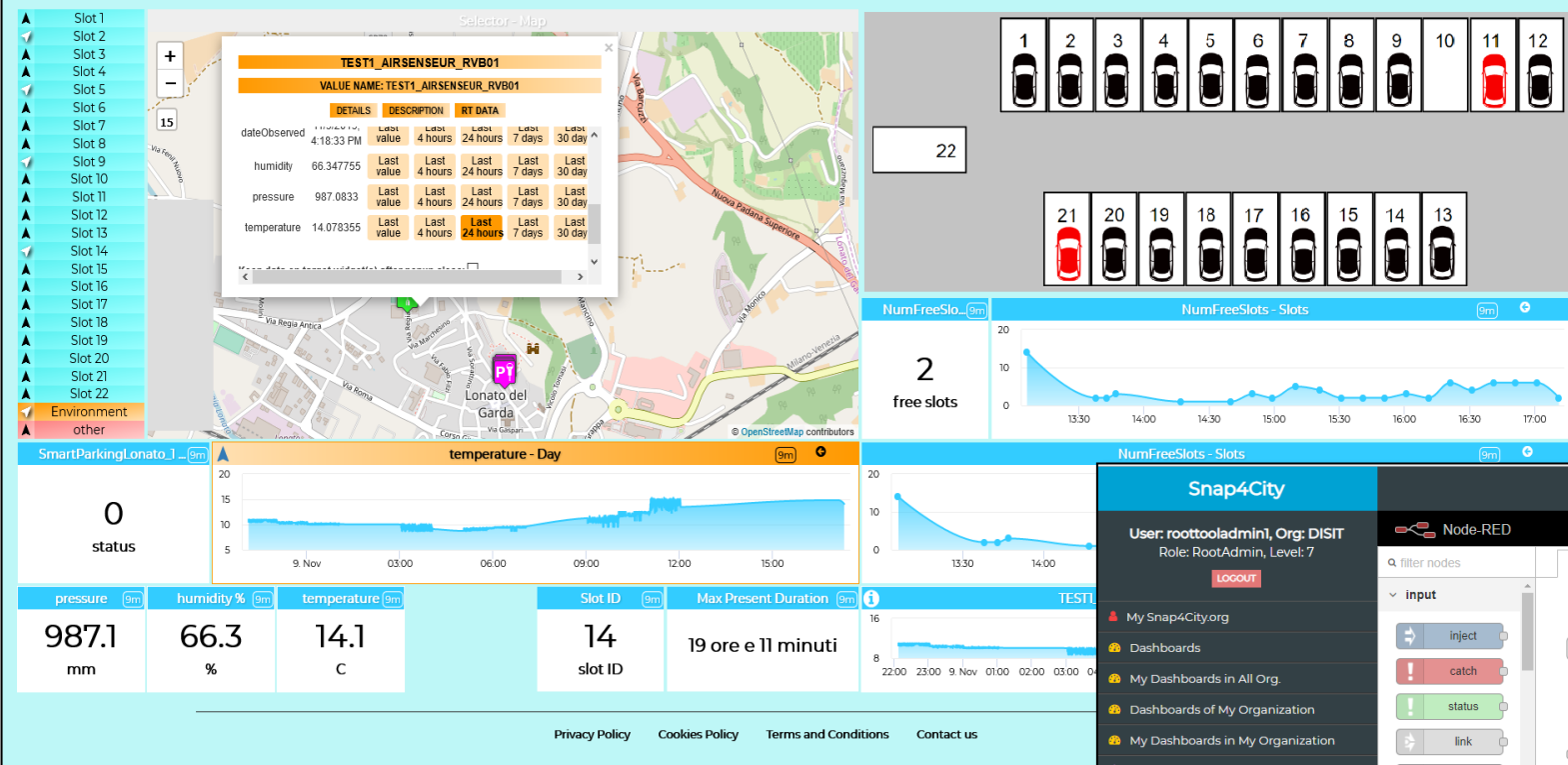
# *IoT App Smart Parking*





## Smart Lonato del Garda

Sat 9 Nov 17:20:59



# Smart Parking Monitoring (SVG, statistic, overparking)



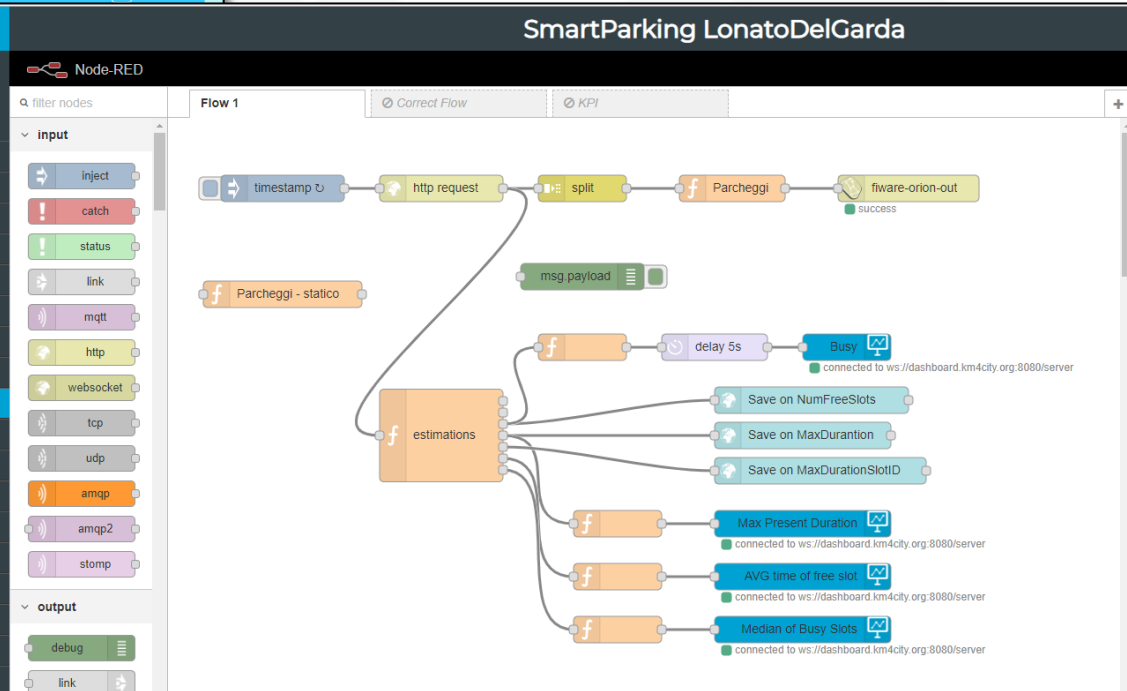
*Lonato del Garda*

**Snap4City**

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

[Logout](#)

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



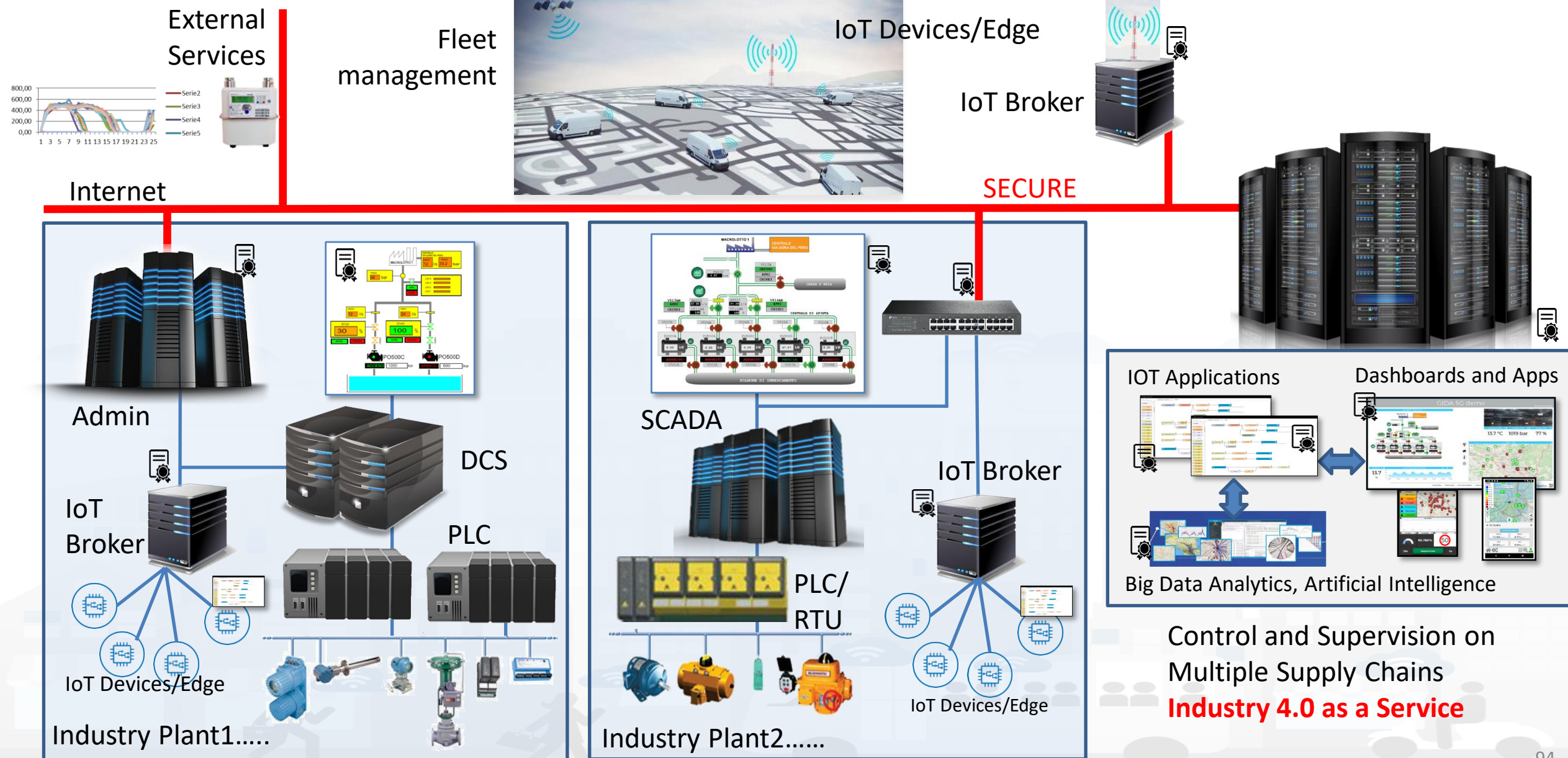
TOP

# *IoT App Smart Industry 4.0*

## *Snap4Industry*











*Prato*

<https://www.snap4city.org/758>

*Smart City vs Industry 4.0*

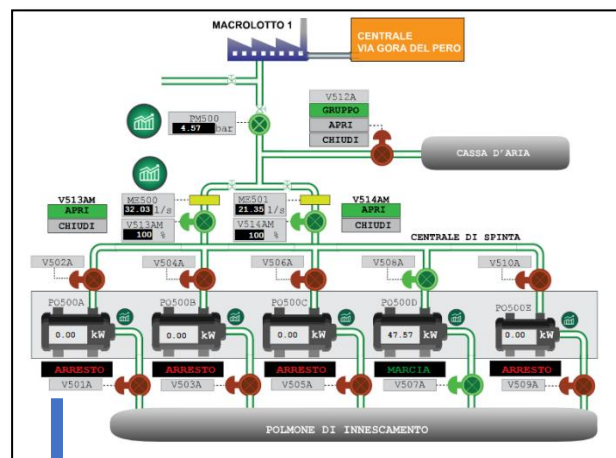


# GIDA set up



GESTIONE  
IMPIANTI  
DEPURAZIONE  
ACQUE S.p.A.

Smart City  
data from  
many  
sources

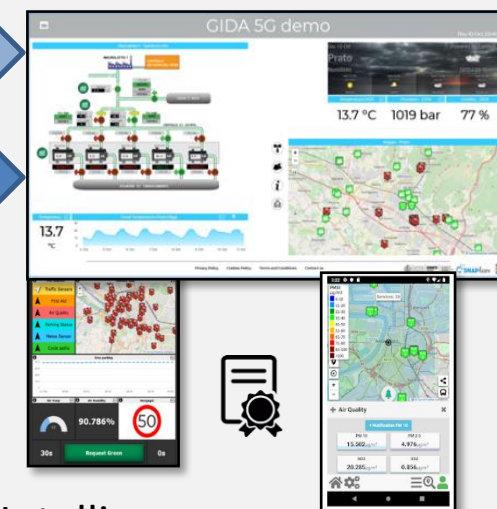


GESTIONE  
IMPIANTI  
DEPURAZIONE  
ACQUE S.p.A.

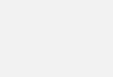
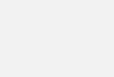
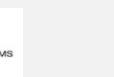
IOT Data  
Shadow  
Snap4City

IOT Applications

Dashboards and Apps



Big Data Analytics, Artificial Intelligence



ModBus to  
Snap4City  
Gateway Edge

5G network  
devices

## Telemonitoring Telecontrol

<https://www.snap4city.org/369>

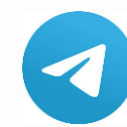
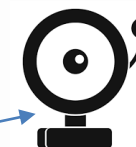
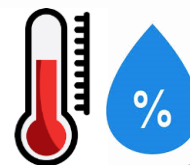
5G



Measuring any kind of sensors values

Controlling Energy Power

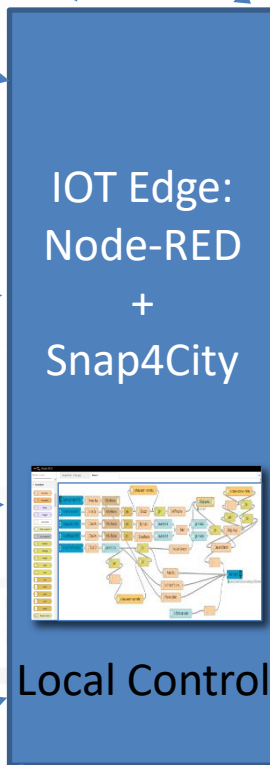
Measuring  
Energy Consumption



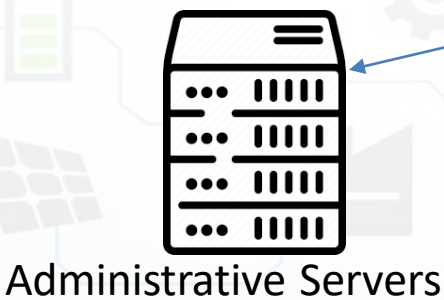
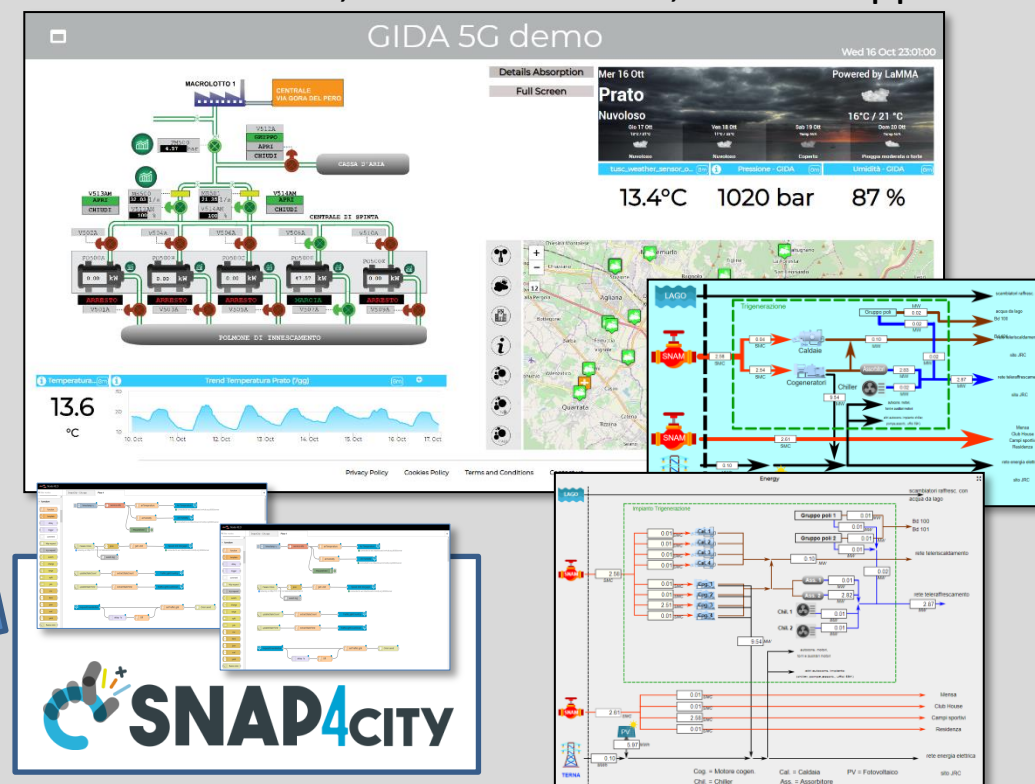
Any kind of notification channel



DCS



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



Administrative Servers



ODBC

Alexa: Voice Commands

Snap4City (C), November 2023





*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

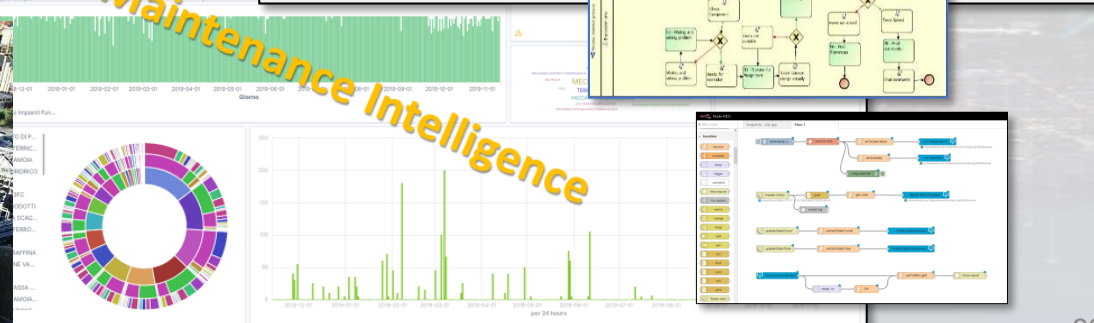
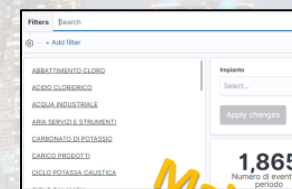
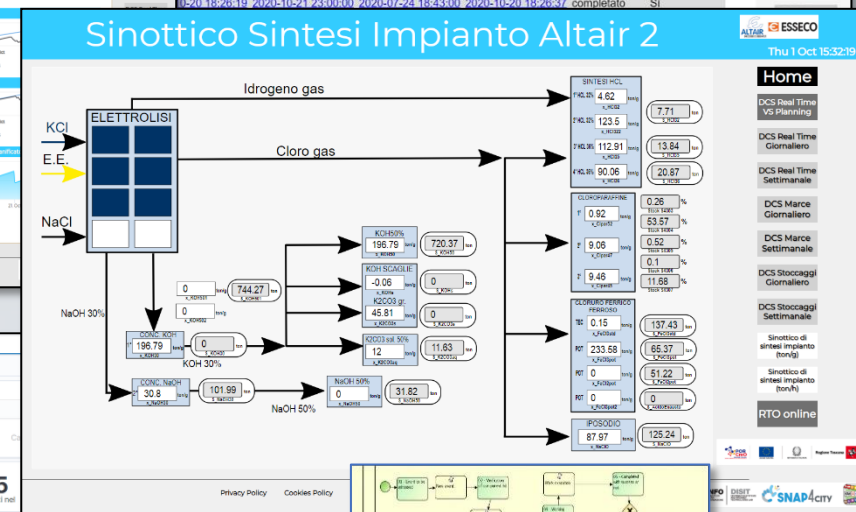
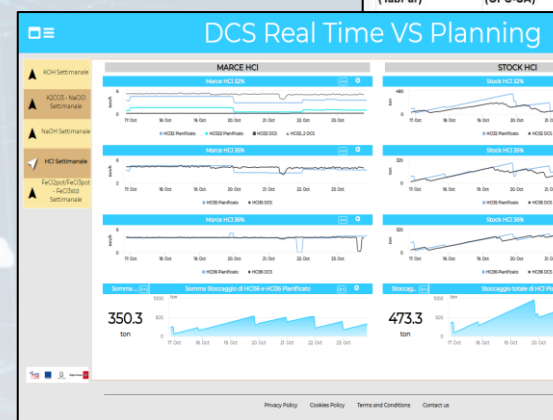
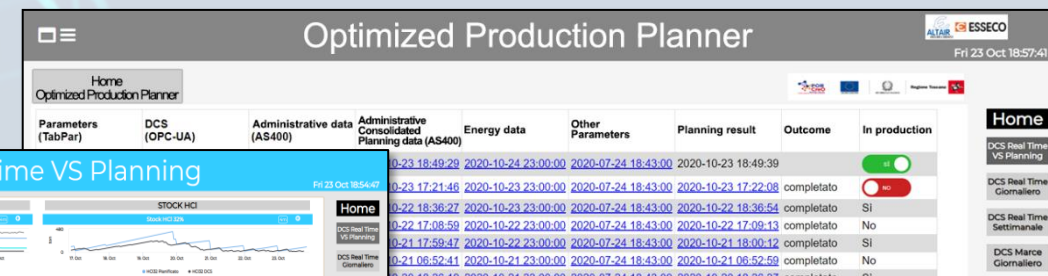
- **Historical and Real Time data**

- Billions of Data

- **Services Exploited on:**

- Multiple Levels, Mobile Apps, API

- **Since 2020**





# Industry Plant Supervision and Maintenance

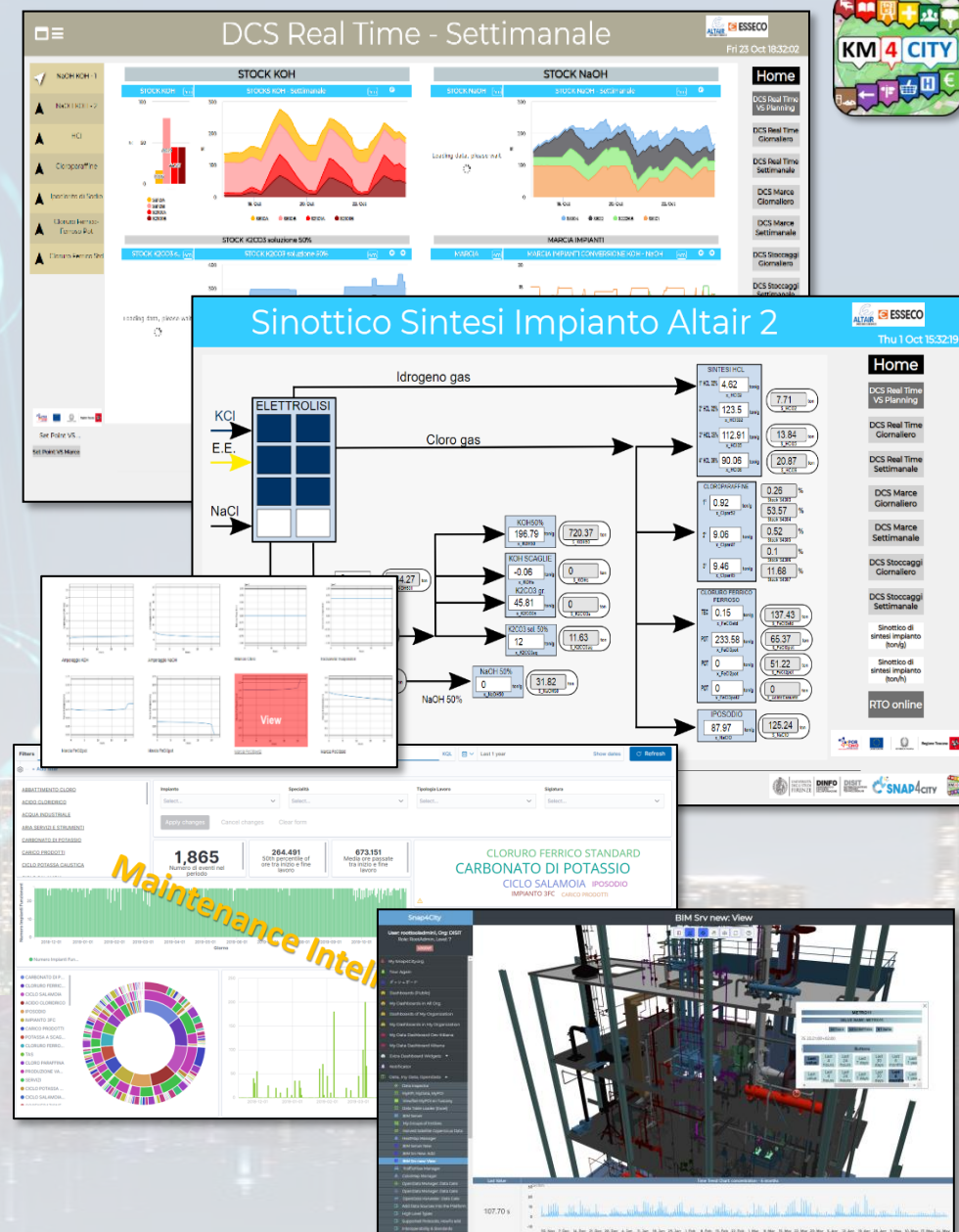
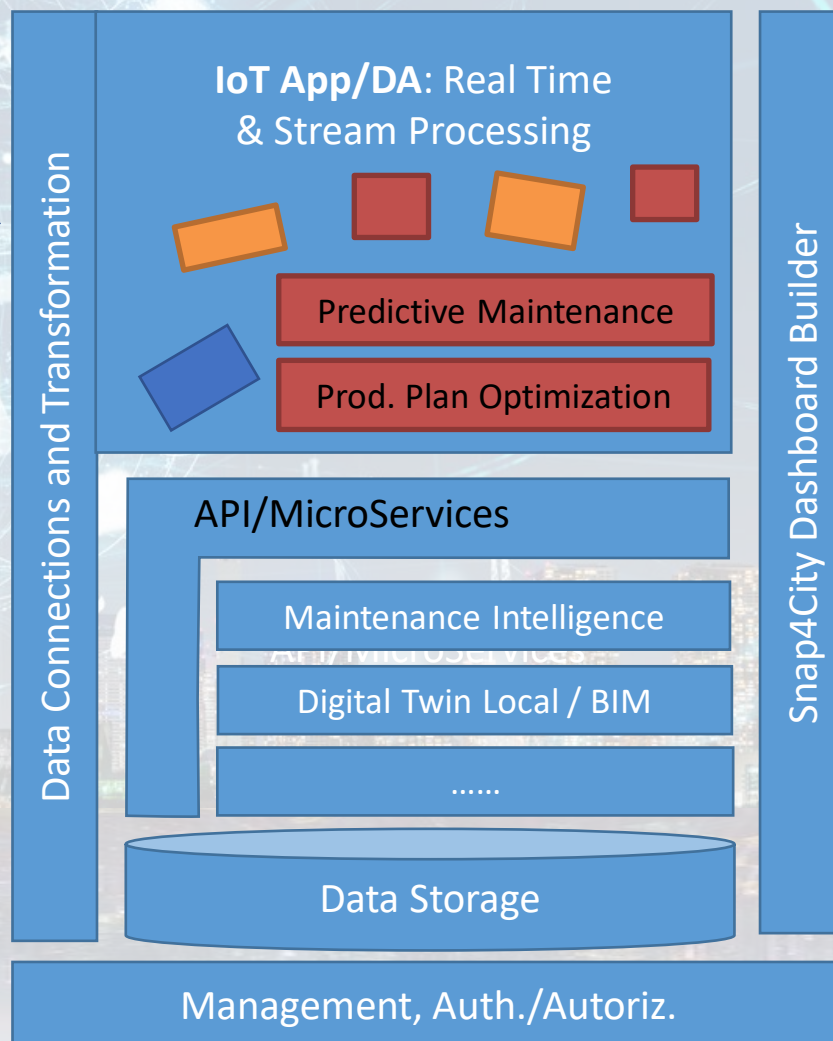


## Aims

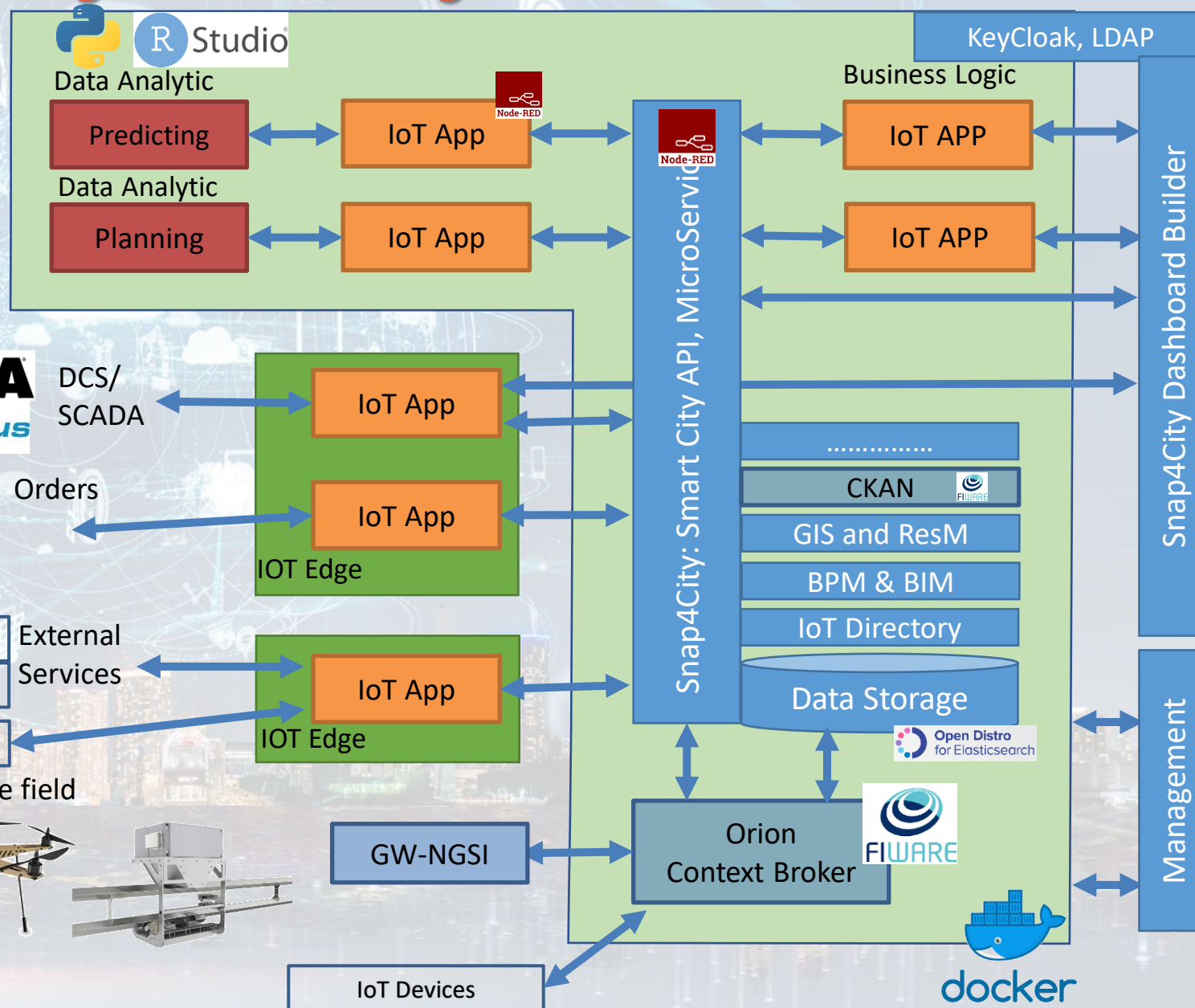
- **Control Room:** Higher level supervision and monitoring (since 2020)
  - Management of Production Plan *Optimization*
  - Control of Perimeter with drone and sensors
- **Maintenance ticketing** (since 2017)
  - *predictive* (in development)
  - 3D Digital Twin (in development)



# MicroService Architecture

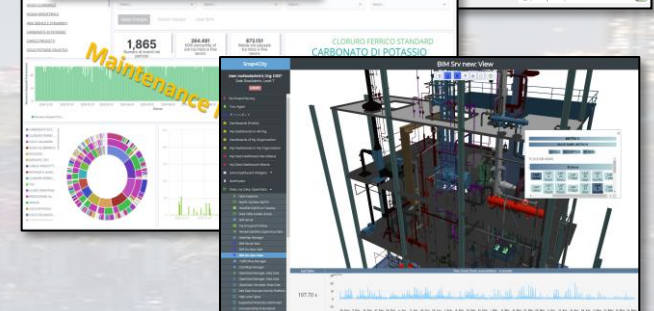
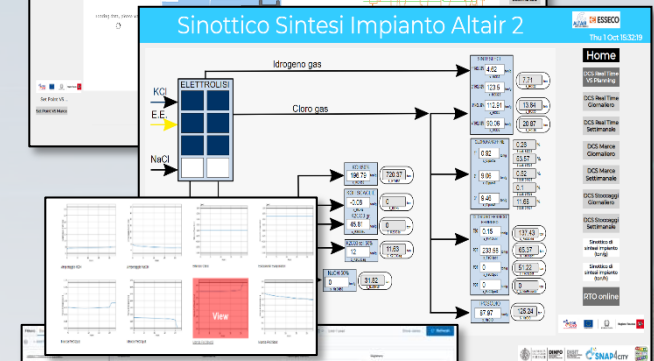
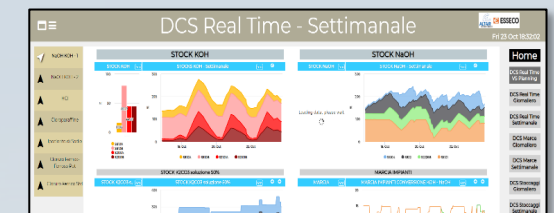


# Snap4City/Industry Detailed Architecture

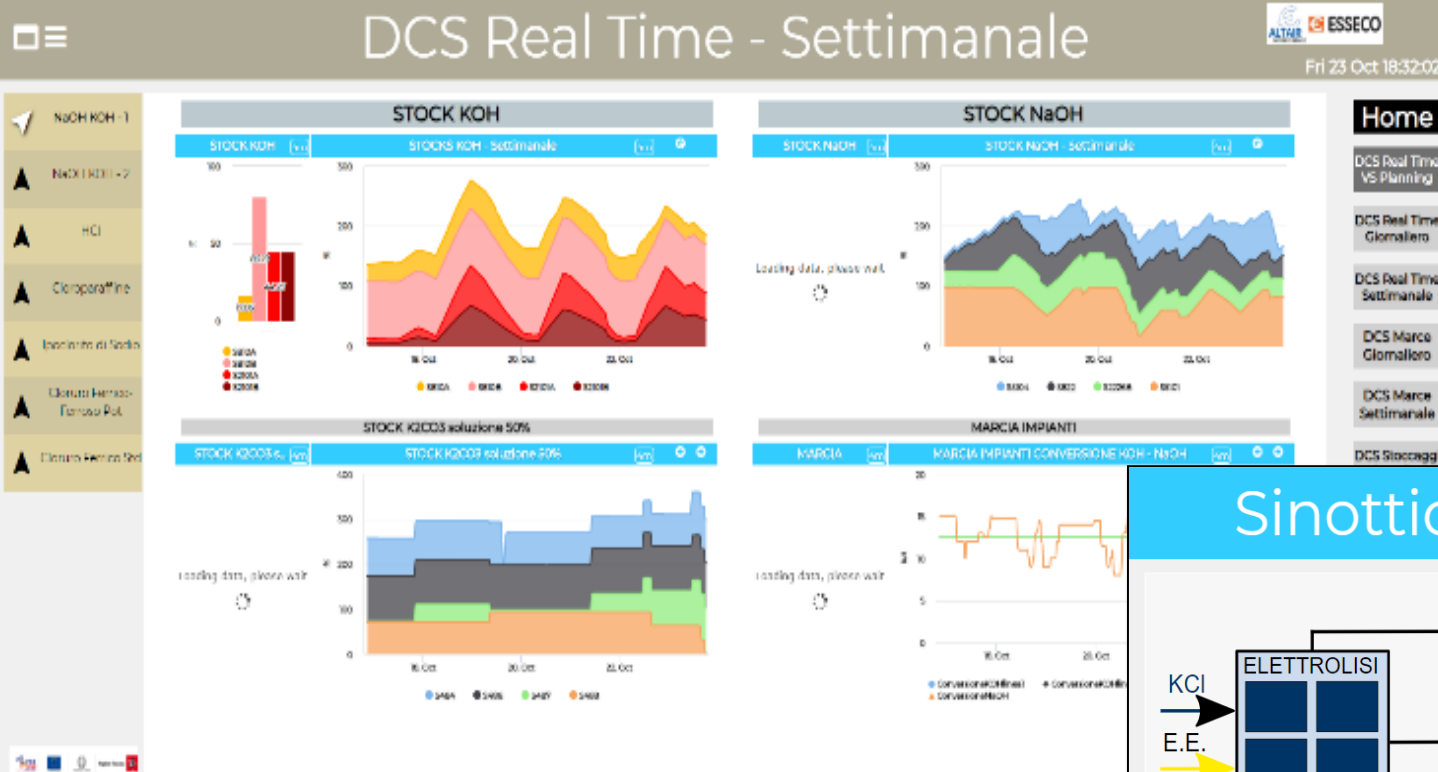


Production Parameters

Dashboards, Visual Analytics, Synoptics, 3D, Maps







## RTO online

Thu 1 Oct 15:34:29

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

Home

DCS Real Time VS Planning

DCS Real Time Giornaliero

DCS Real Time Settimanale

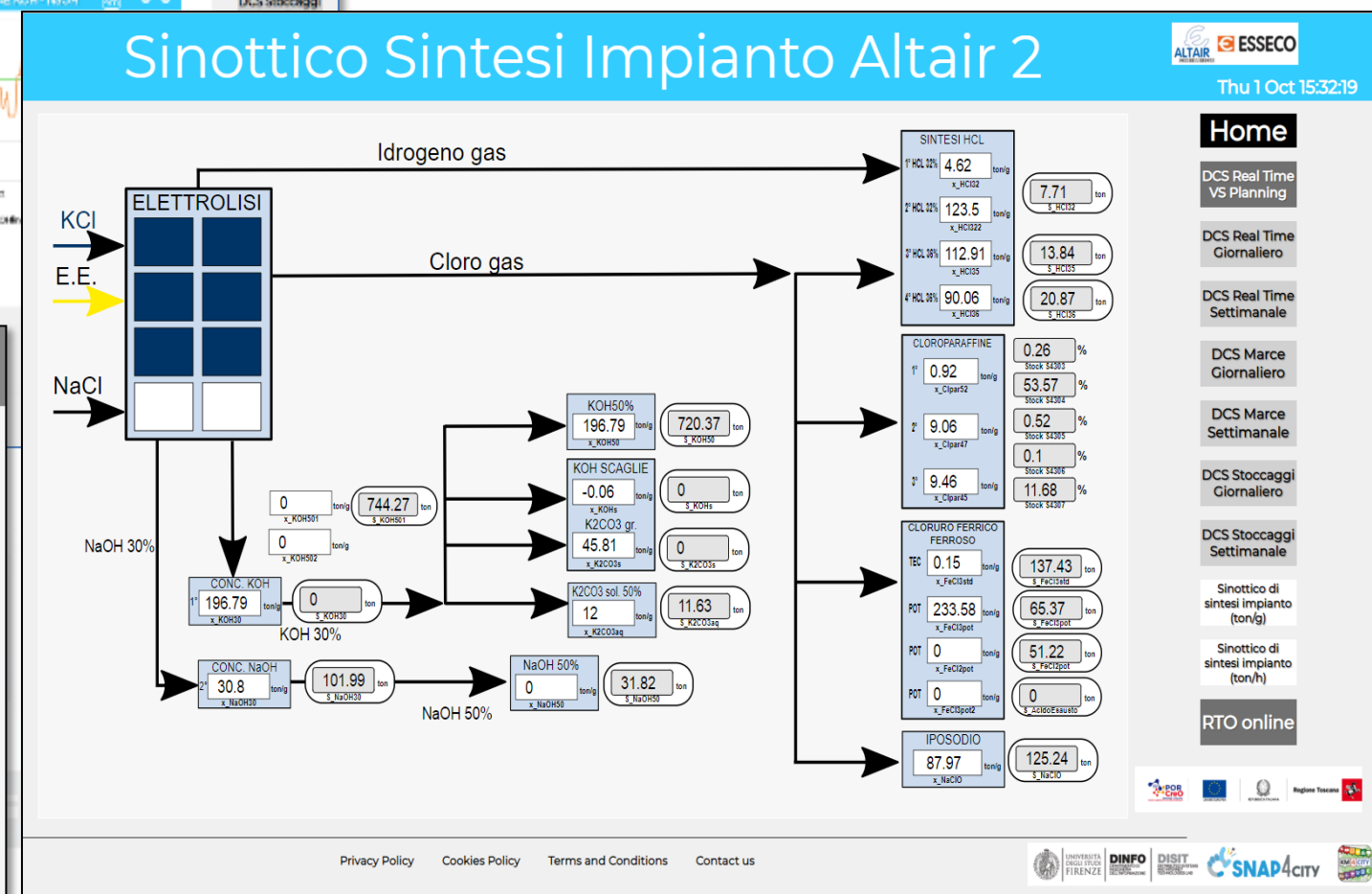
DCS Marce Giornaliero

DCS Marce Settimanale

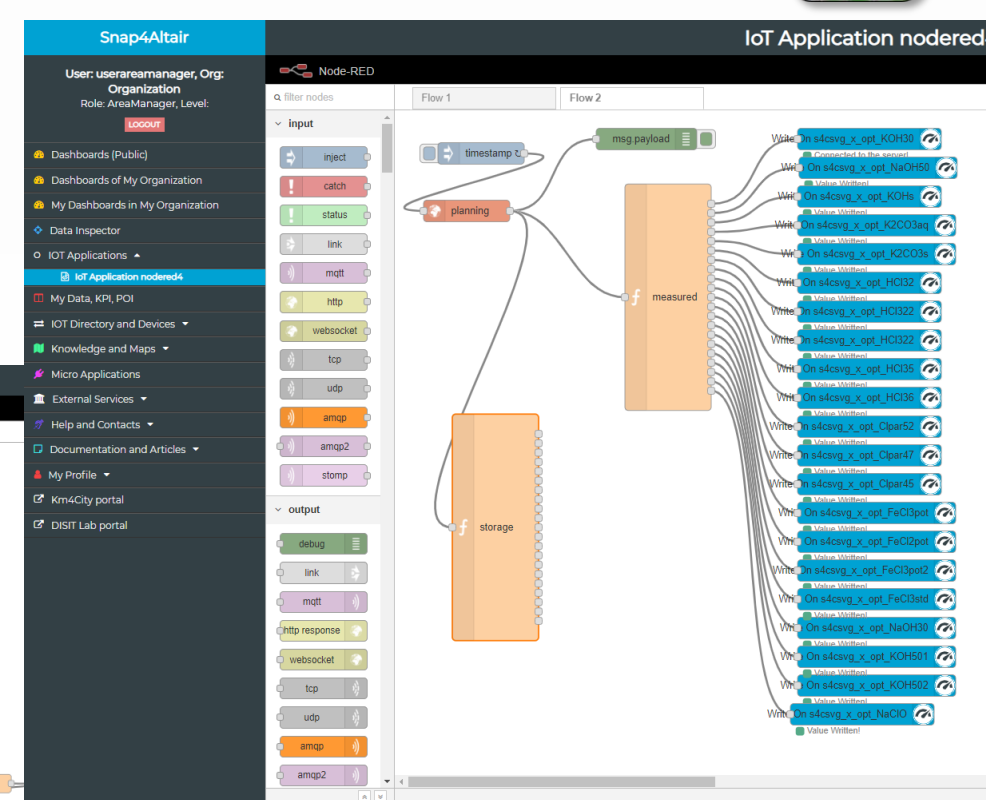
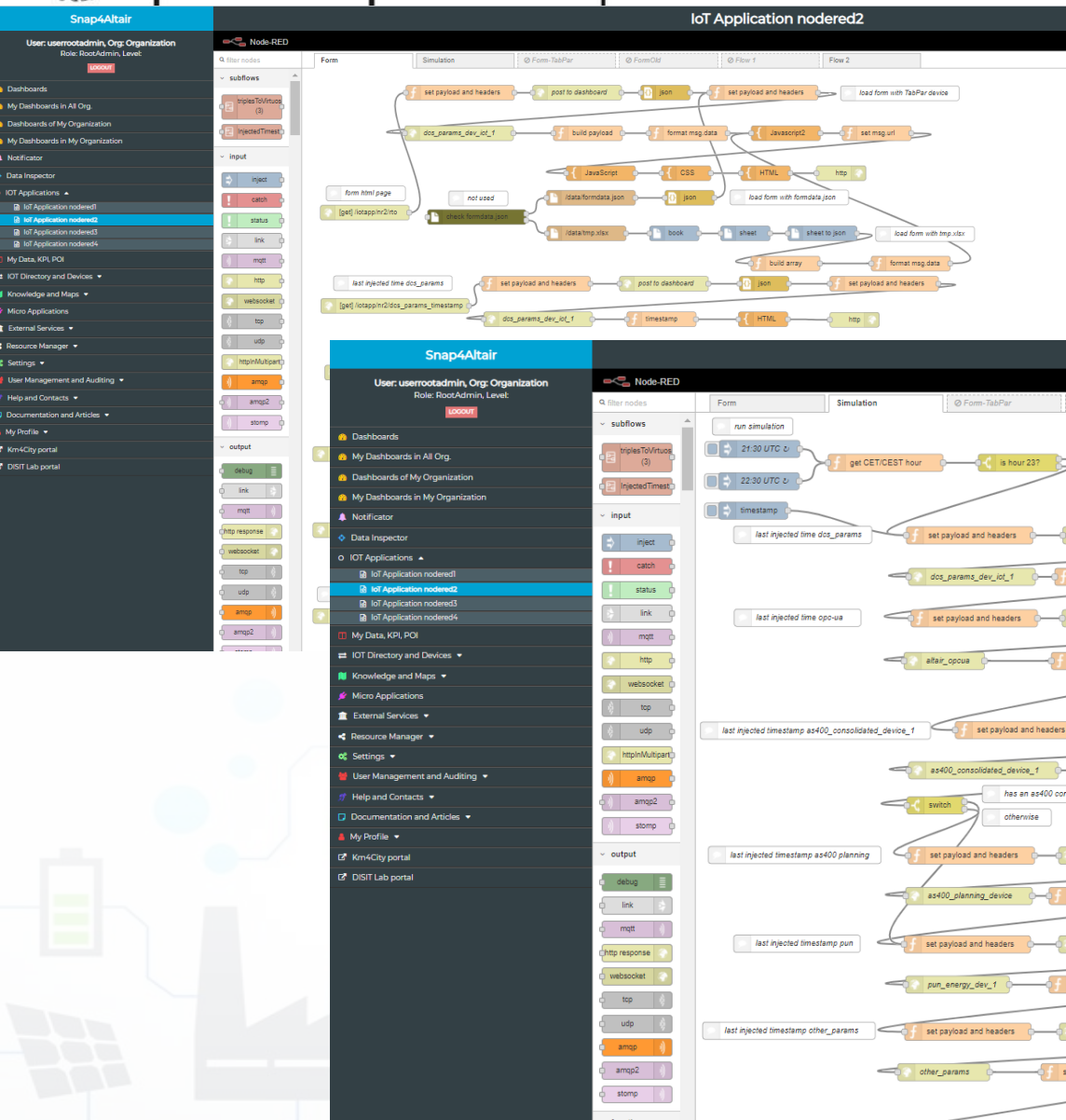
DCS Stoccaggi Giornaliero

DCS Stoccaggi Settimanale

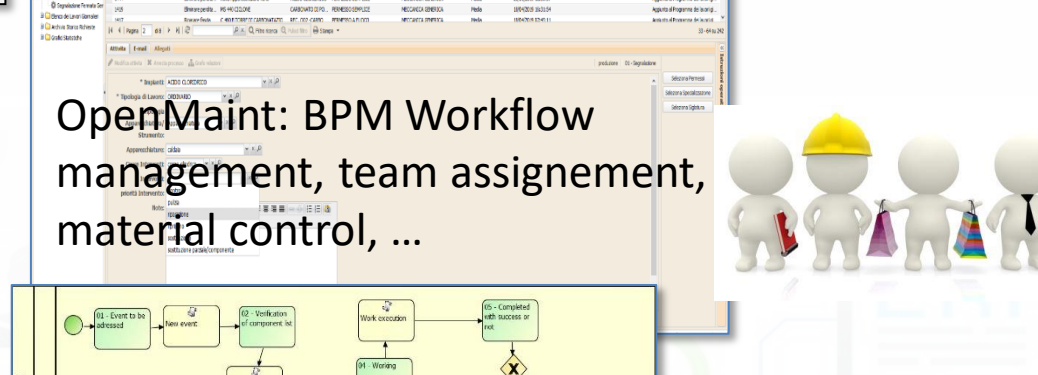
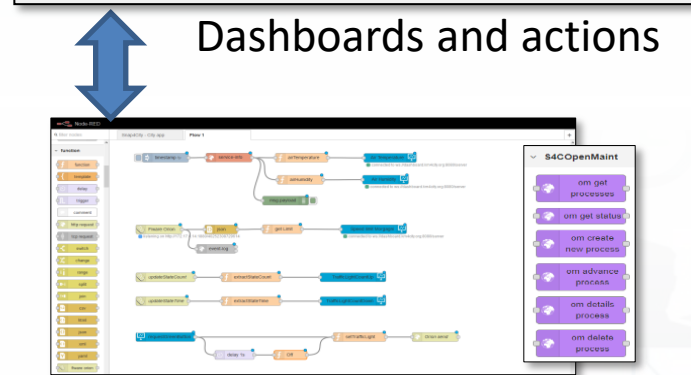
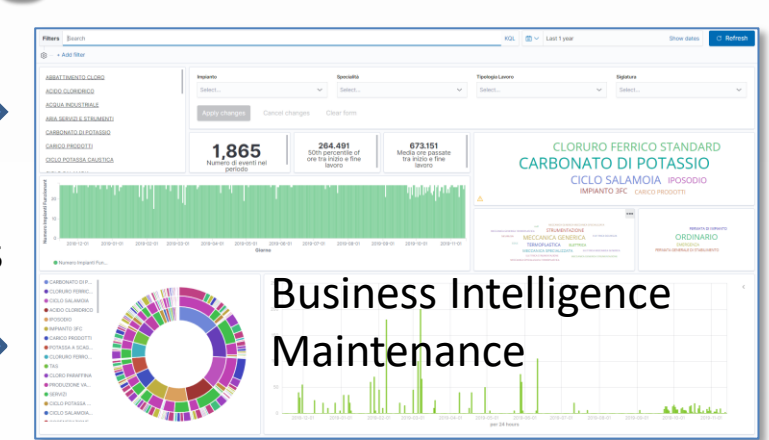
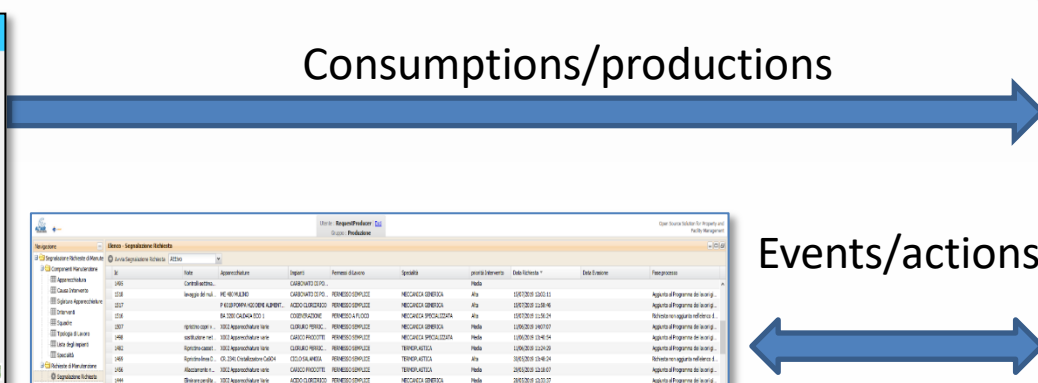
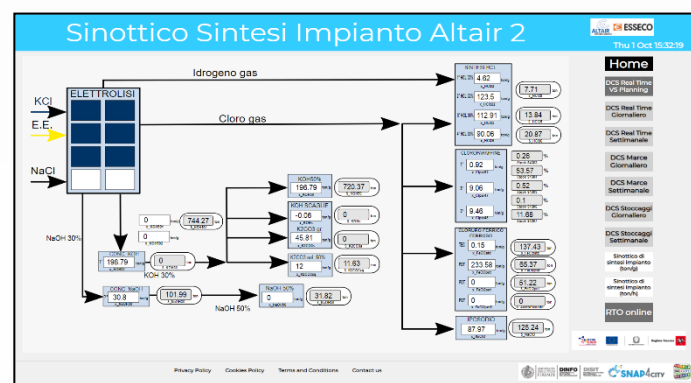
Sinottico di sintesi impianto



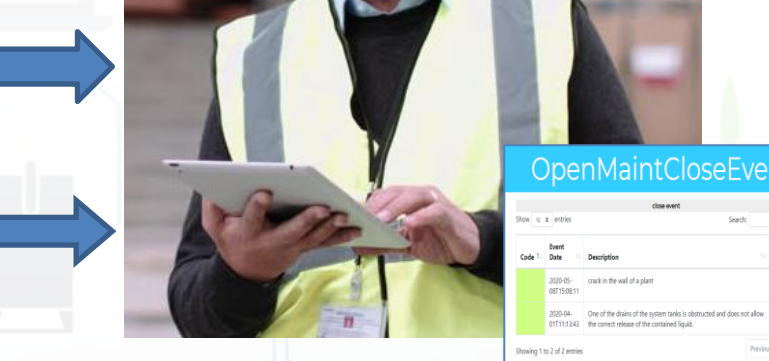
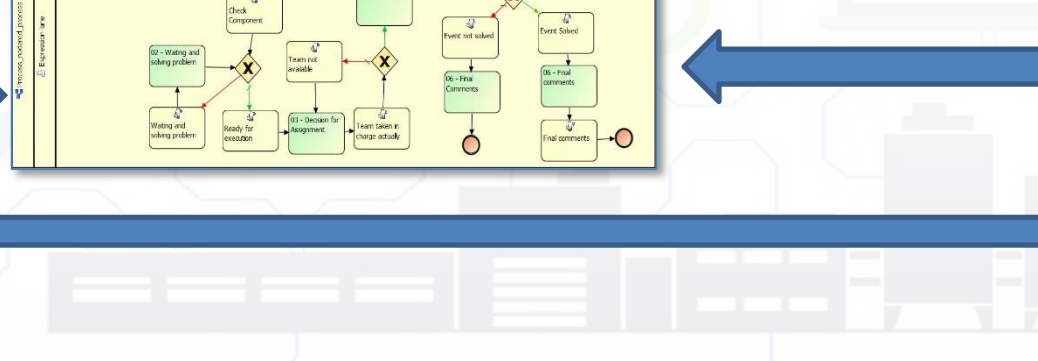




# Workflow for Ticket management



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







# CAPELON

*Västerås, Sweden*



reference

- Capelon Test Lights - Cloned - Cloned2**

Tue 26 Jan 17:40:35

**Selector**

  - ▲ DOCSEFFFEBA9010
  - ▲ DOCSEFFFEBA8FA7
  - ▼ DOCSEFFFEBA8CB8
  - ▲ 90FD9FFFEBD5A7F
  - ▲ DOCSEFFFEBA8E56
  - ▼ DOCSEFFFEBA8B82
  - ▲ DOCSEFFFEBA8B1T
  - ▲ DOCSEFFFEBA8F68
  - ▲ DOCSEFFFEBA8FAS
  - ▲ DOCSEFFFEBA8F87
  - ▲ DOCSEFFFEBA8A02
  - ▲ 90FD9FFFEBEES0T
  - ▲ DOCSEFFFEBA8B73
  - ▲ DOCSEFFFEBA8F02
  - ▼ DOCSEFFFEBA8DEA
  - ▲ DOCSEFFFEBA8E22
  - ▲ DOCSEFFFEBA8EBA
  - ▼ DOCSEFFFEBA8FA4
  - ▲ DOCSEFFFEBA8F0D
  - ▲ DOCSEFFFEBA8E10
  - all
  - all bulb
  - Arby101
  - DOCSEFFFEBA9...

**Selector - Map**

Map showing the location of the lights (marked with yellow and green icons) in the area of Dorset and Andover.

**DOCSEFFFEBA9010 - OLtemperature**

50 °C

**Lights ON**

63

**Lights Off**

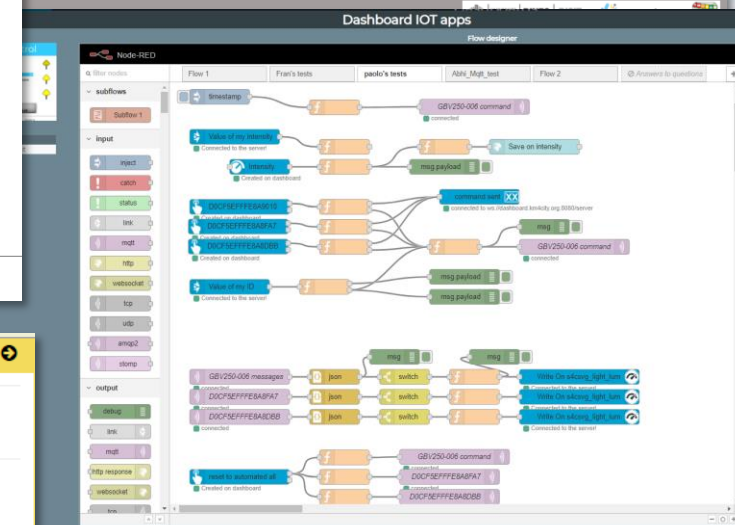
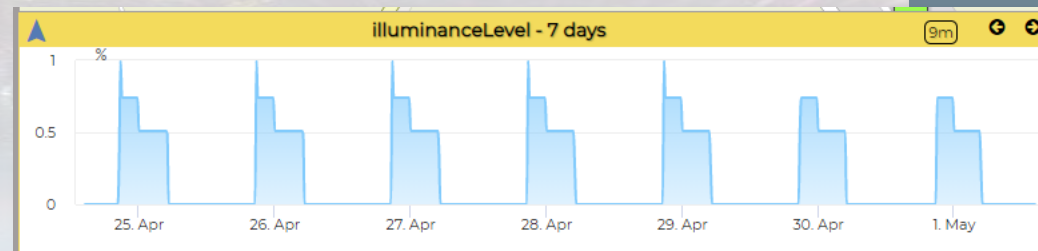
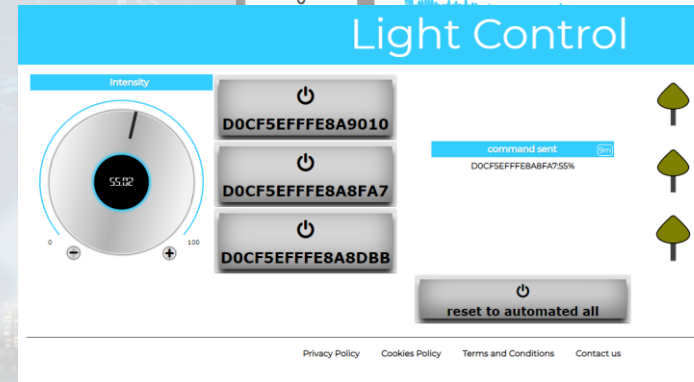
63

**Active Power Spend...**

0 15 30 45

**Bar Series**

Bar chart showing the relative intensity (0 to 1) of 20 different light fixtures.



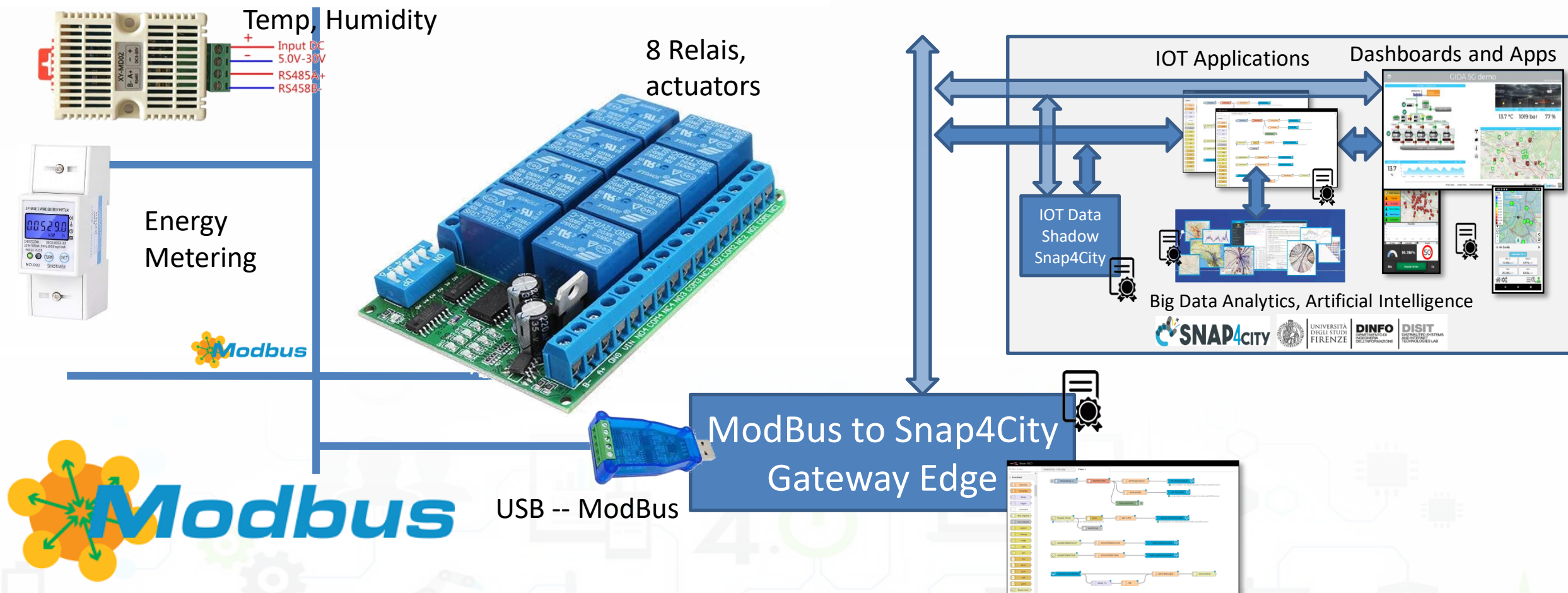
TOP

# *IoT App Smart Industry 4.0 ModBus Integration*



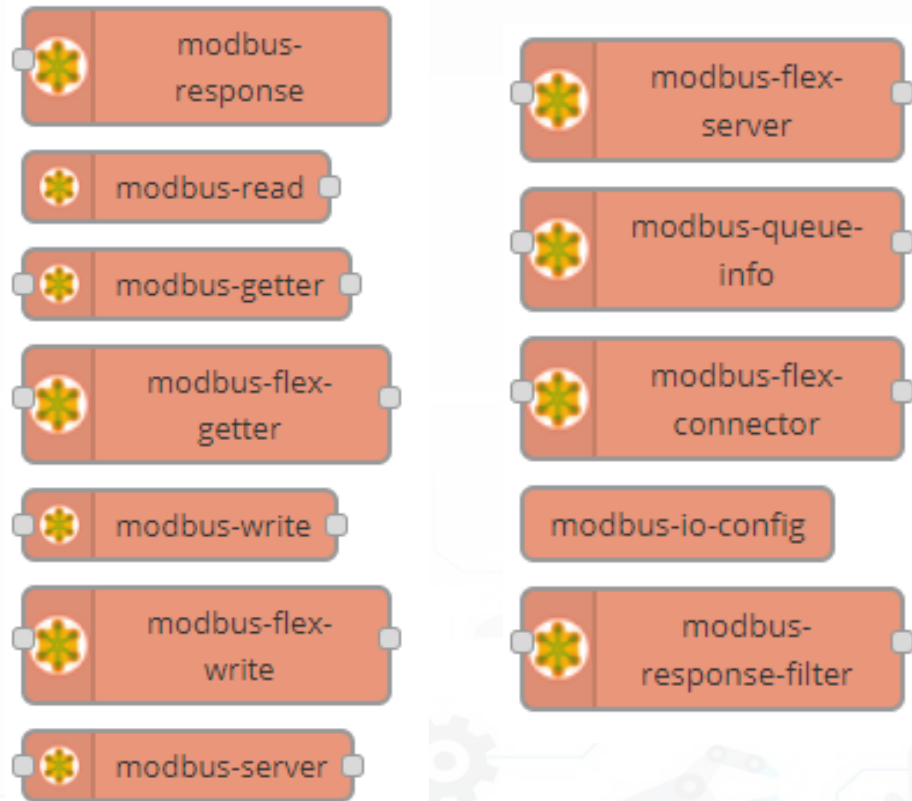


# Devices

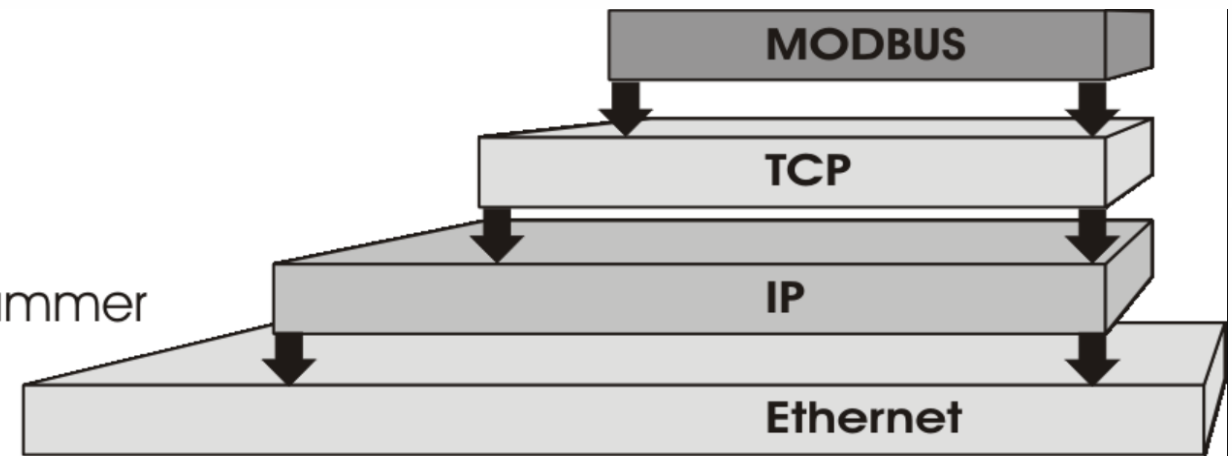


- A large range of devices: sensors and actuators
- Over serial as RS485 and/or IP





UNIT  
PORT  
TCP/IP Nummer  
MAC ID

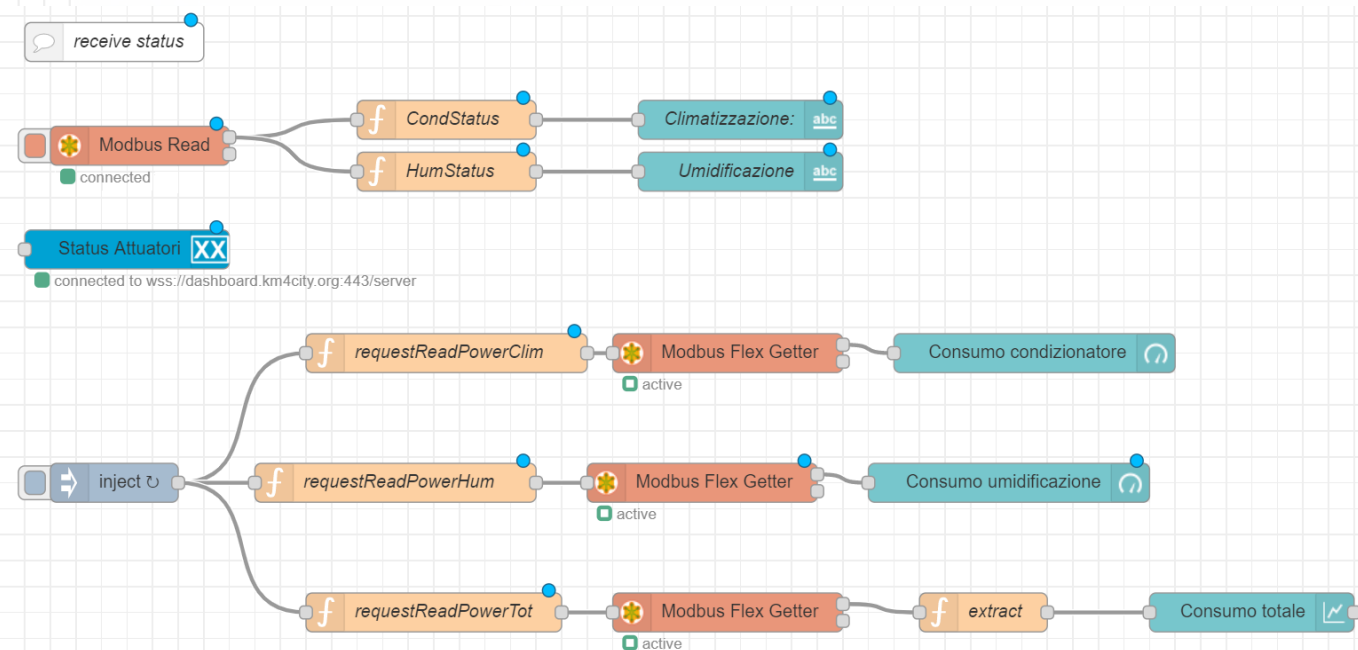
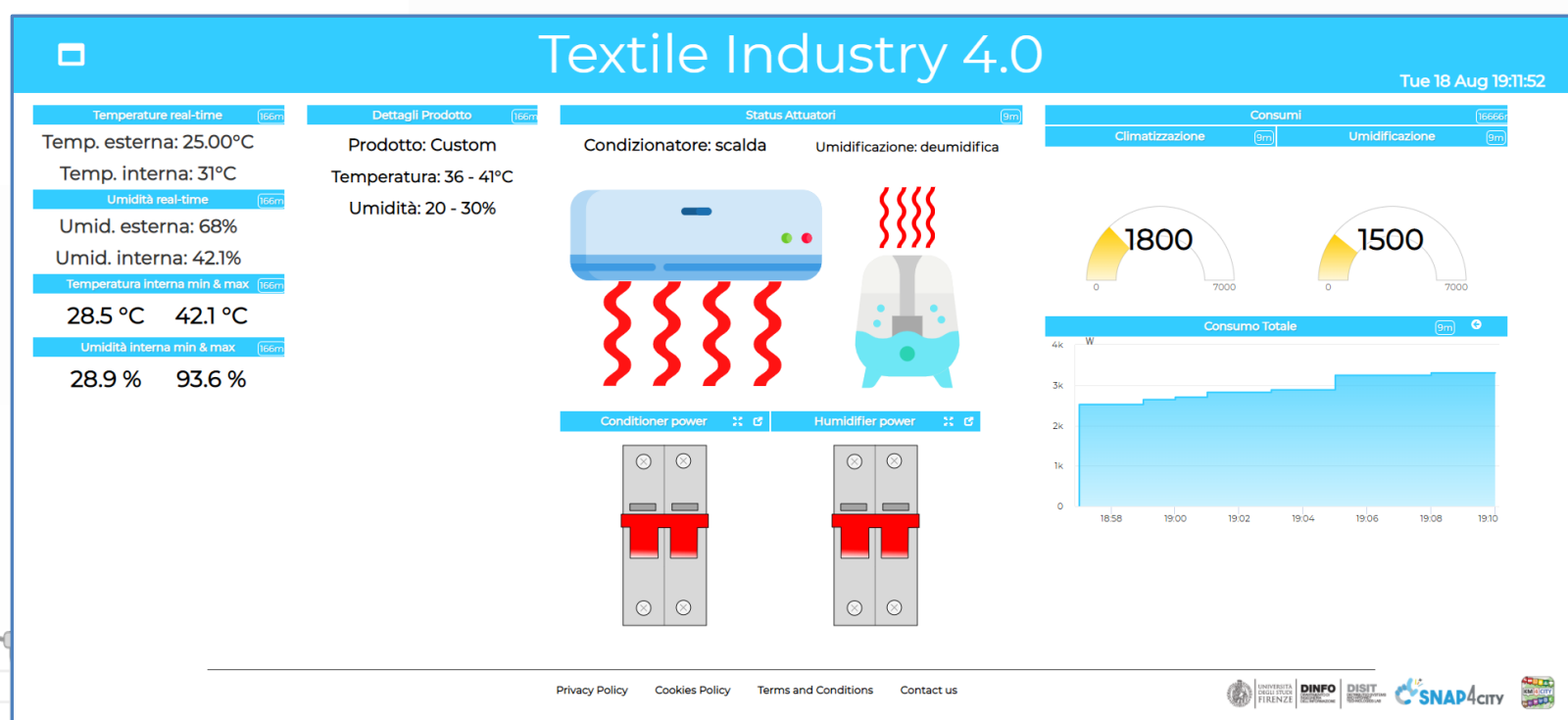
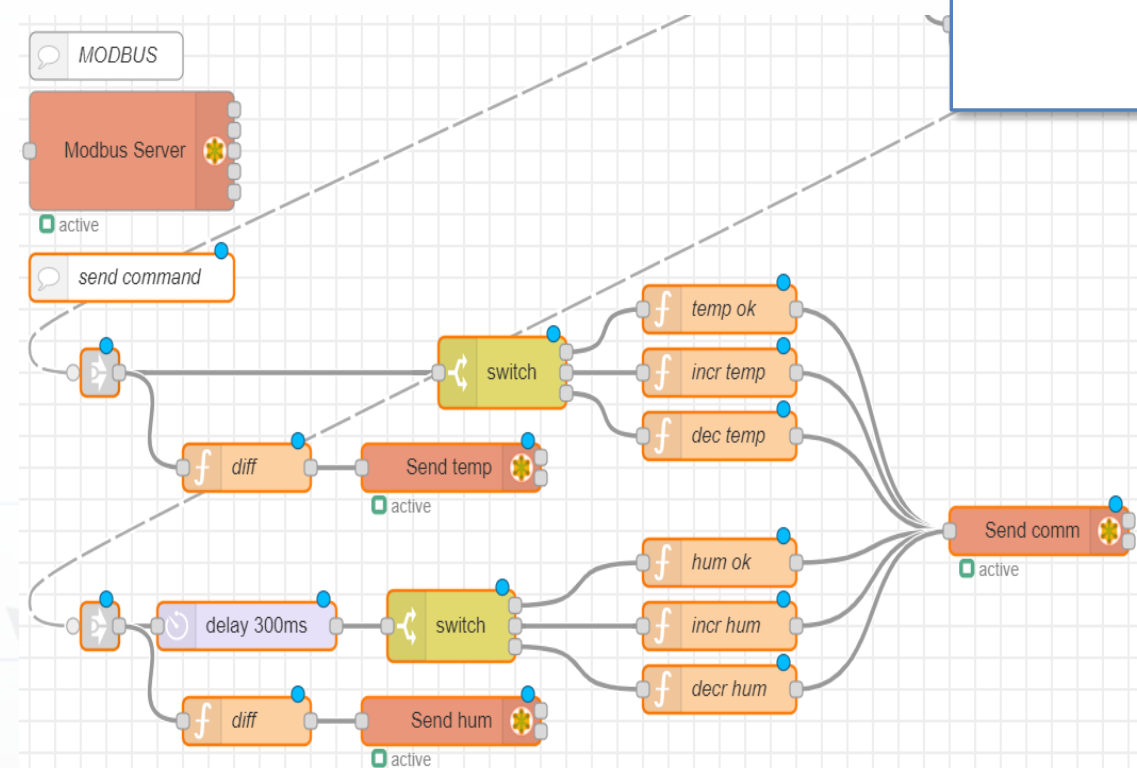




UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

**DINFO**  
DIPARTIMENTO DI  
INGEGNERIA  
DELL'INFORMAZIONE

**DISIT**  
DISTRIBUTED SYSTEMS  
AND INTERNET  
TECHNOLOGIES LAB



TOP

# *IoT App vs Smart Home* *Snap4Home*





Sonoff: Controlling Energy Power



Philips Hue: Controlling Lights



Hue: Motion Control / Alarm



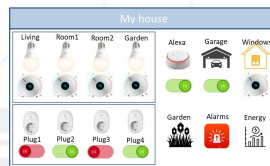
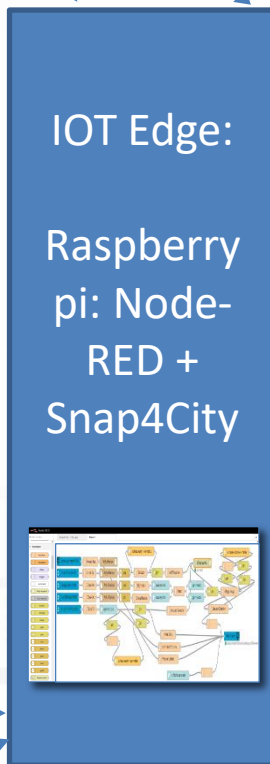
Measuring  
Energy Consumption



TP Link: Controlling / Measuring Energy Plugs

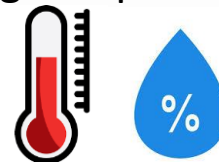


Alexa: Voice Control



Local Control

Measuring Temperature and Humidity



Controlling Motors



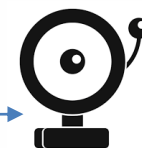
Controlling  
Irrigators



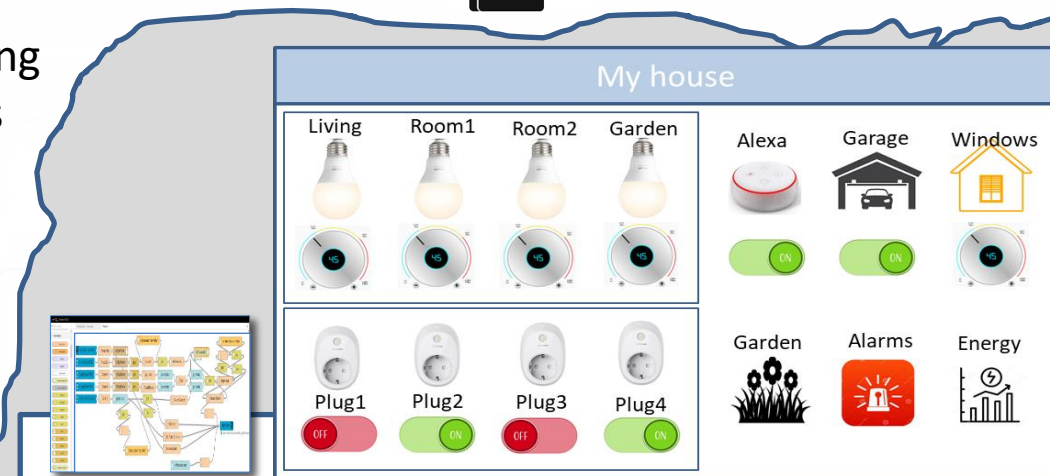
Garage Door



Window  
Roller Shutters



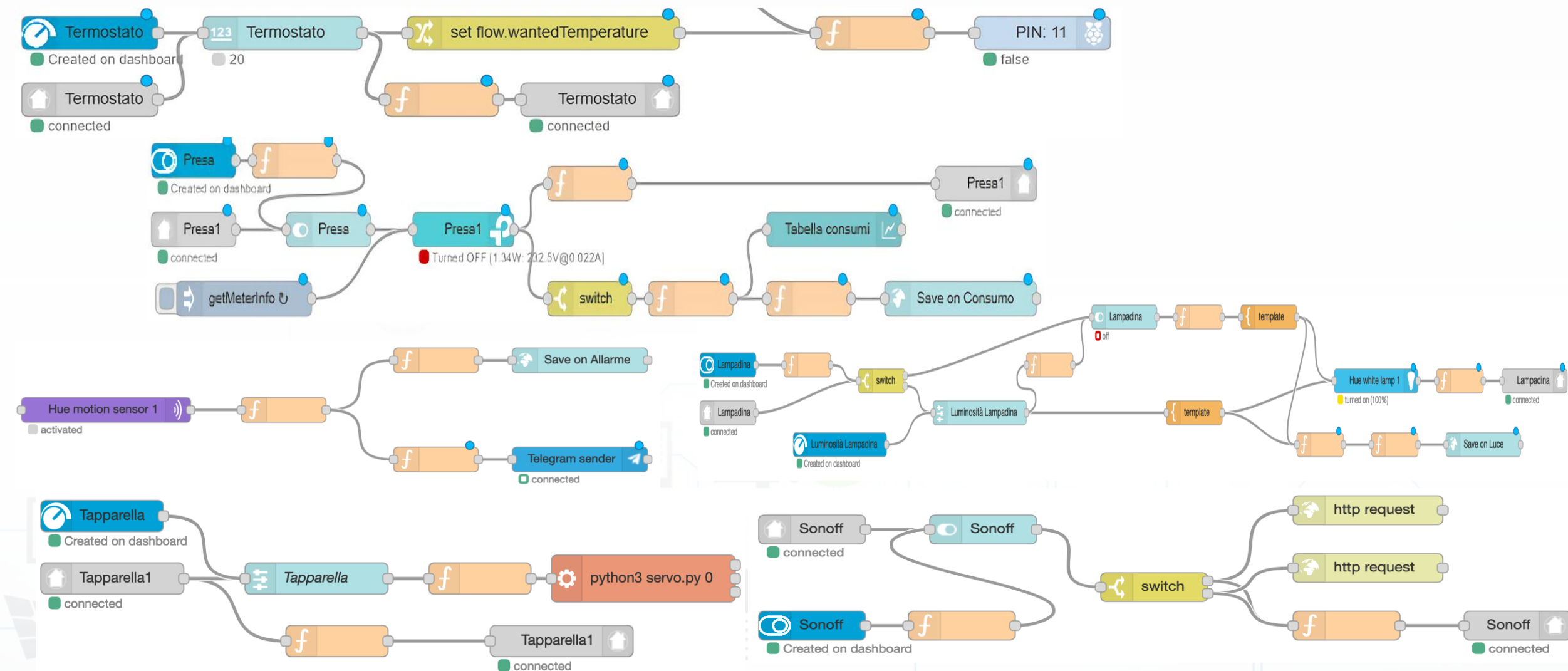
Alarm sound  
and light



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



# Example: IOT App on Snap4Home



Hue Hub



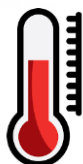
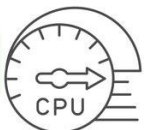
Motion Control / Alarm



TP Link  
plugs:  
meter



Alexa: Voice Control

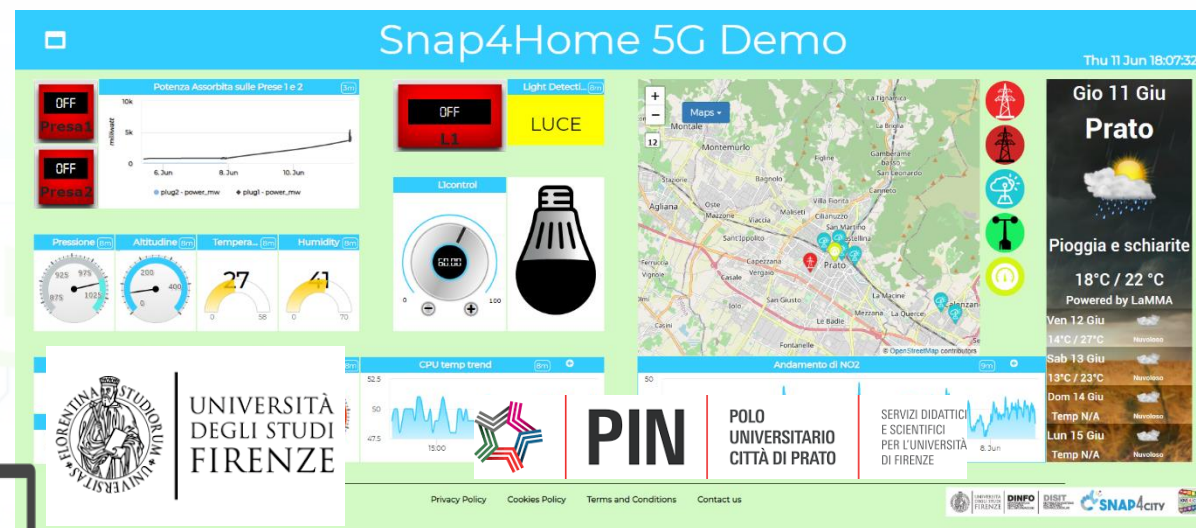
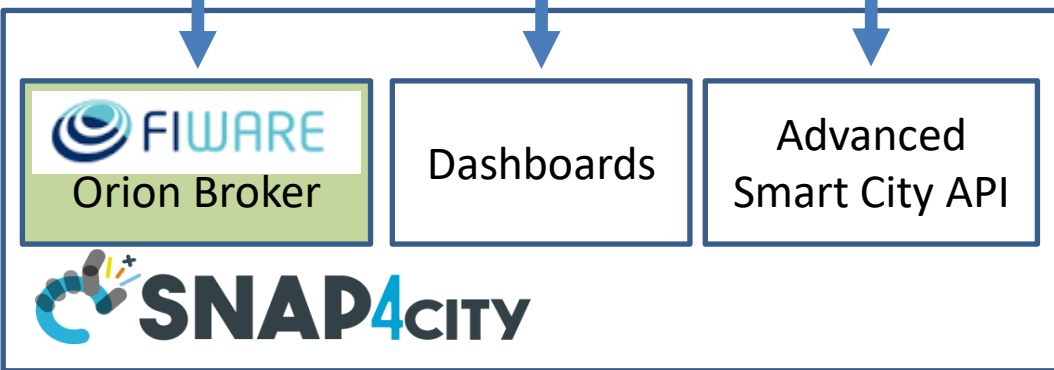


**IOT Edge:**

**Raspberry  
pi:  
Node-RED  
+  
Snap4City  
MicroServ  
ice Library**

5G gateway

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





Philips Hue: Controlling Lights



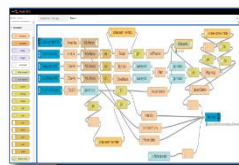
Hue: Motion Control / Alarm



TP Link: Controlling / Measuring Energy Plugs



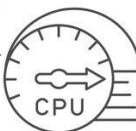
Alexa: Voice Control



IOT Edge:

Raspberry  
pi: Node-  
RED +  
Snap4City

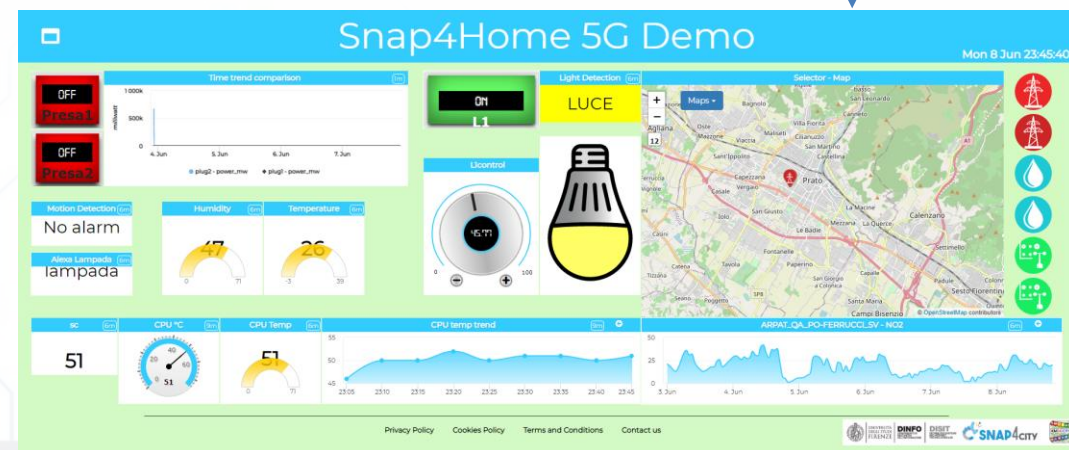
Measuring: Temperature, Humidity, light in the room



Monitoring: CPU clock, status

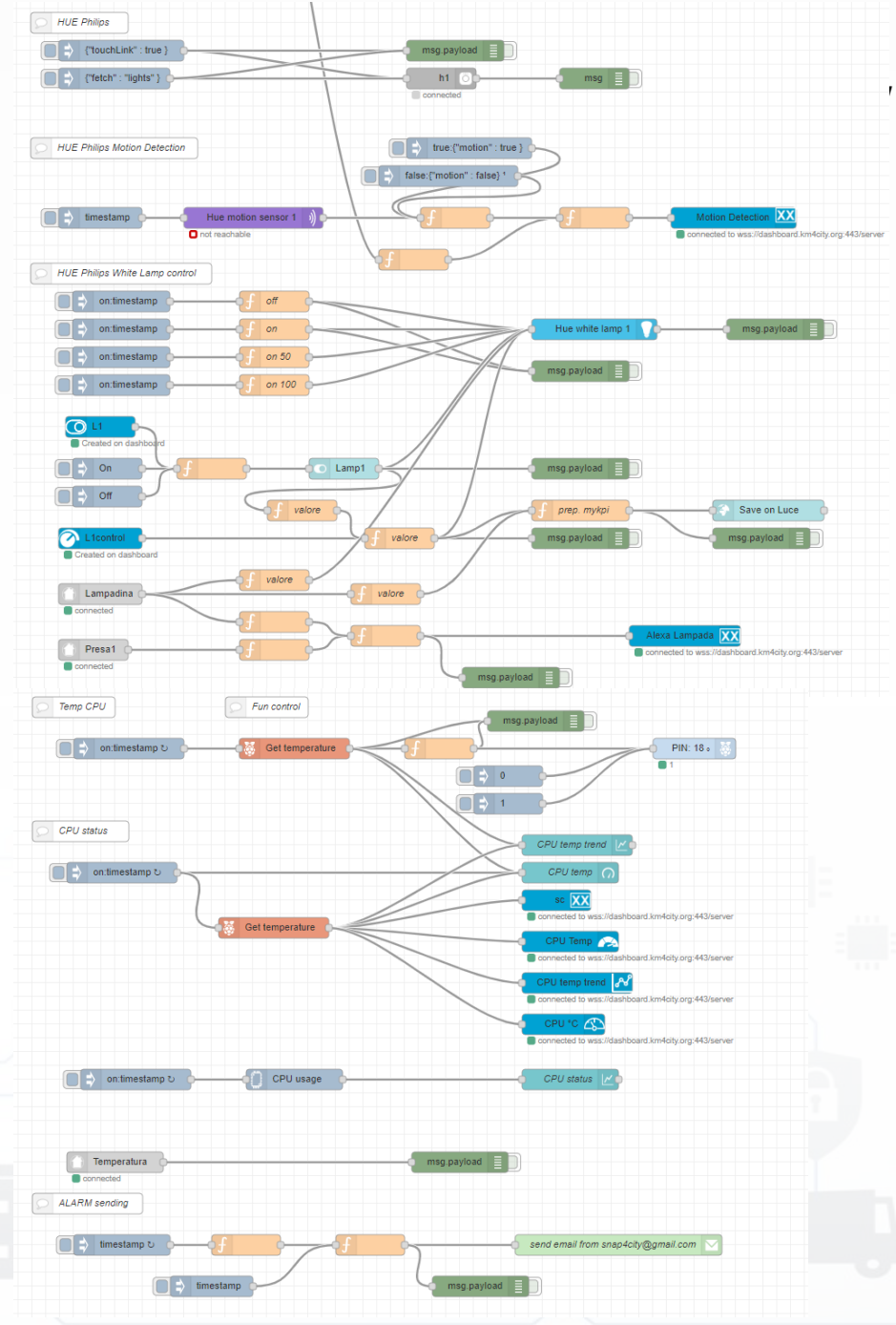
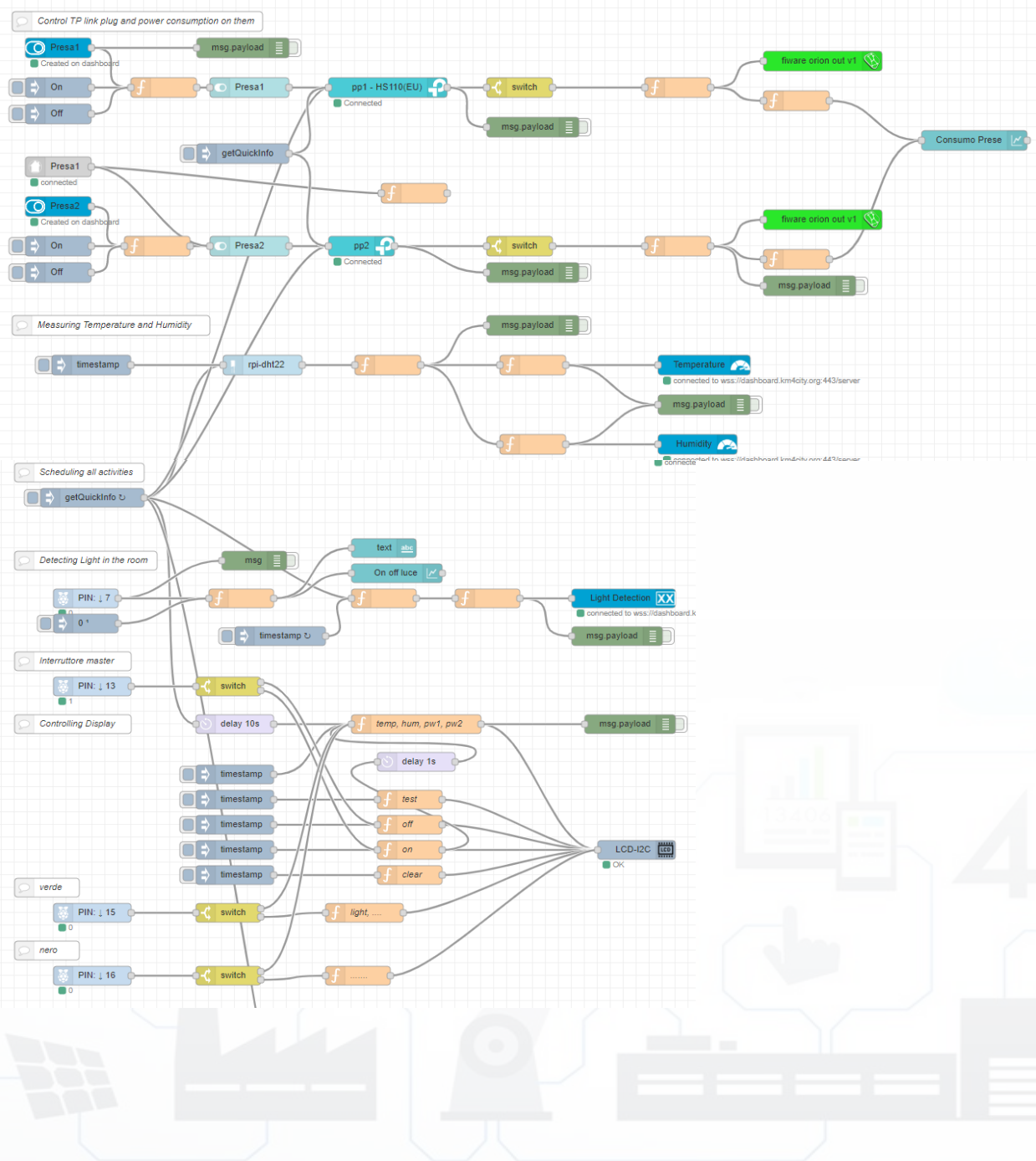
5G gateway

Internet



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





# *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

**My Data, KPI, POI**

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

10 entries

No. +	High Level Type	Nature	Sub Nature	Value Name
17057634	MyKPI	Environment	Weather_sensor	slider
17057633	MyKPI	Environment	Weather_sensor	maxslider
17057632	MyKPI	UtilitiesAndSupply	Energy_supply	minslider
17057631	MyKPI	UtilitiesAndSupply	Energy_supply	button
17057453	MyKPI	UtilitiesAndSupply	Agents	S4CTuscanyApp
17057452	MyKPI	UtilitiesAndSupply	Agents	S4CTuscanyTra
17057448	MyKPI	HealthCare	Health_district	corkpim2liguria

**KPI Data Details**

High Level Type: MyKPI  
Nature: Environment  
Sub Nature: Weather\_sensor  
Value Name: slider  
Value Type: temperature  
Value Unit: °C  
Data Type: integer  
Last Date: 27/10/2020, 09:49:25  
Last Value: 43.18572617038263  
Last Check: 27/10/2020, 09:49:25  
Username: paolo.disit  
Organizations: [ou=DISIT,dc=ldap,dc=disit,dc=org]  
Healthiness: false  
Ownership: public  
Description: Info  
Latitude: 43.769625  
Longitude: 11.255811  
Insert Time: 27/10/2020, 09:49:25

**IOT Device Models**

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

55 MODELS

Show 10 entries

Device Model	Device	Device Type	Device Category	Device Subcategory	Device Location	Device Status	Device Owner	Device Description
Raspberry snap4city 1	Raspberry Pi 3 Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM	sensor	Raspberry Pi	Ambiental	DELEGATED	DISIT	sensor	Raspberry Pi
Raspberry snap4city 2	Raspberry Pi 3 Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM	sensor	Raspberry Pi	Ambiental	DELEGATED	DISIT	sensor	Raspberry Pi
Arduino Uno	Arduino Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM	sensor	Arduino	Ambiental	DELEGATED	DISIT	sensor	Arduino
Arduino uno-bis	Arduino Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM	sensor	Arduino	Ambiental	DELEGATED	DISIT	sensor	Arduino
sigfox	SigFox Model B Scheda madre CPU 1.2 GHz Quad Core, 1 GB RAM	sensor	SigFox	Ambiental	DELEGATED	DISIT	sensor	SigFox

**Add New Model**

General Info | IOT Broker | Static Attributes | Values

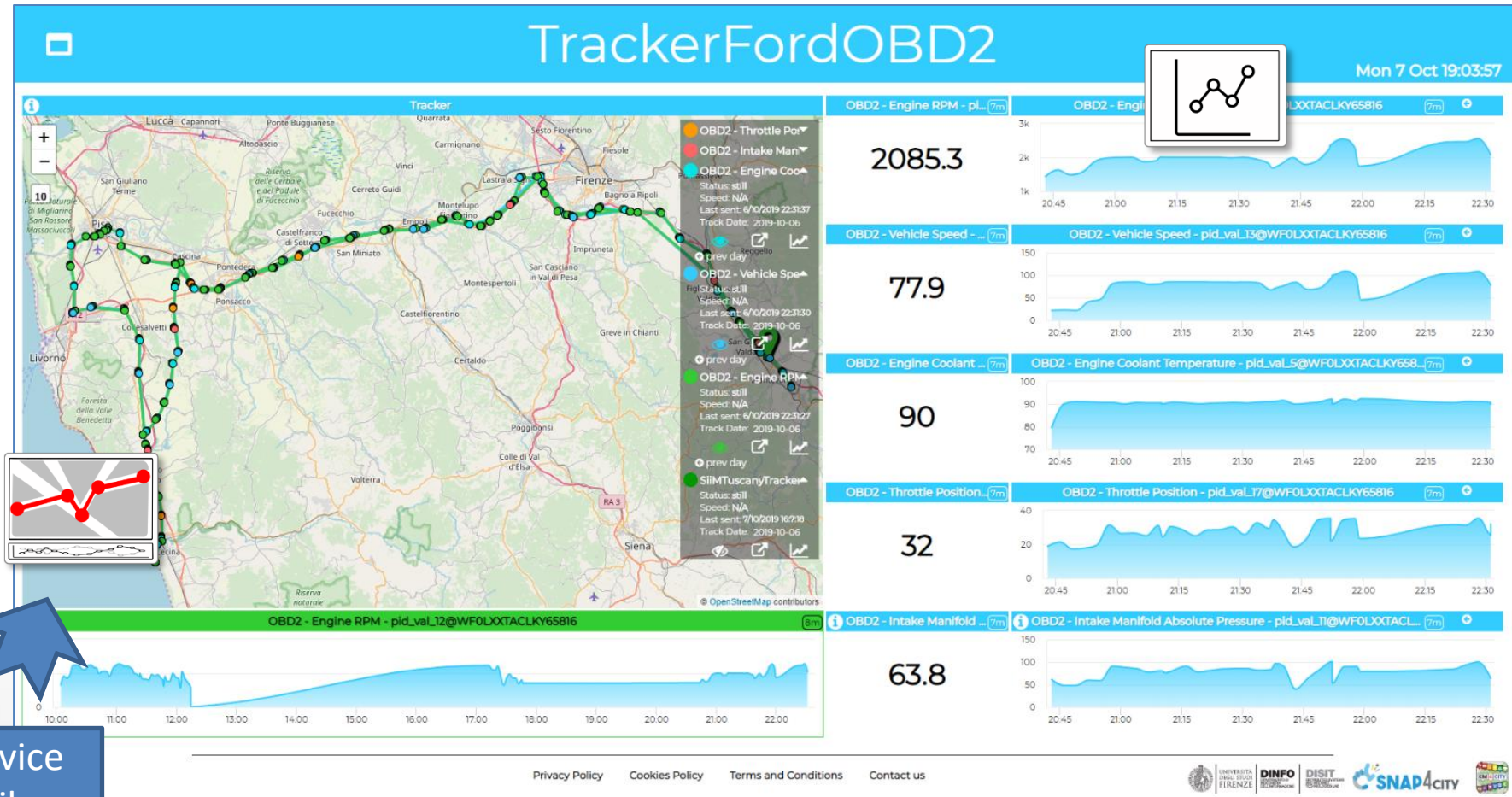
☒ Device in Mobility

Subnature: Select an option

Cancel Confirm

# 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

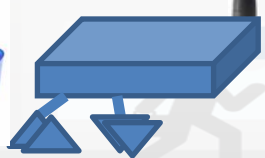


Mobile  
PAX Counter



Apps

OBU



OBD2



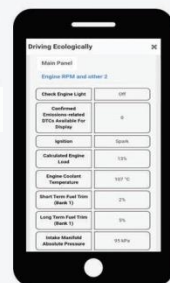
IOT Device  
MOBILE



# IOE – Vehicle Monitoring

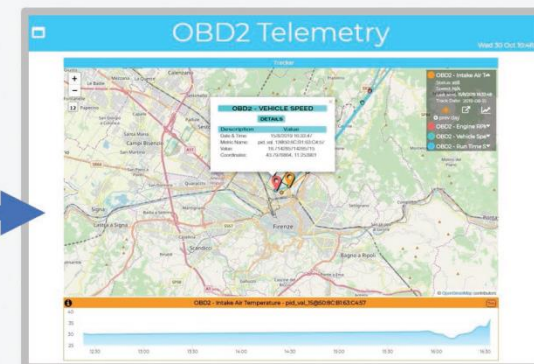


CANBUS  
sniffer

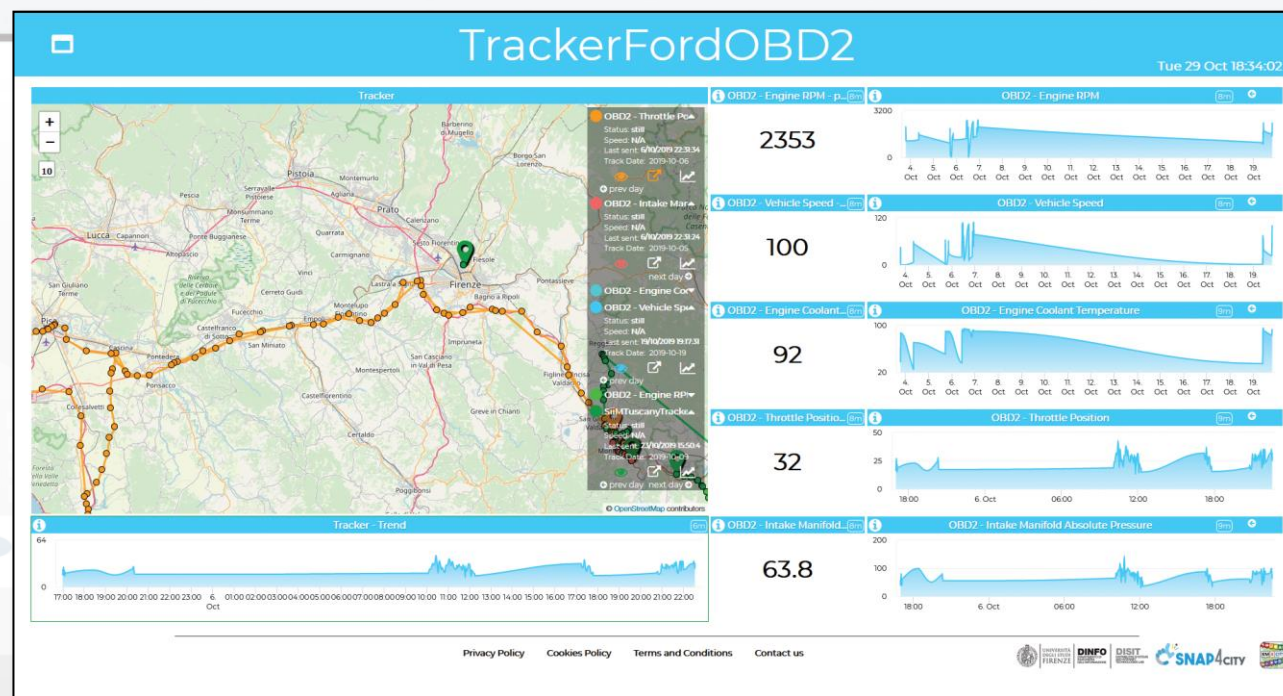
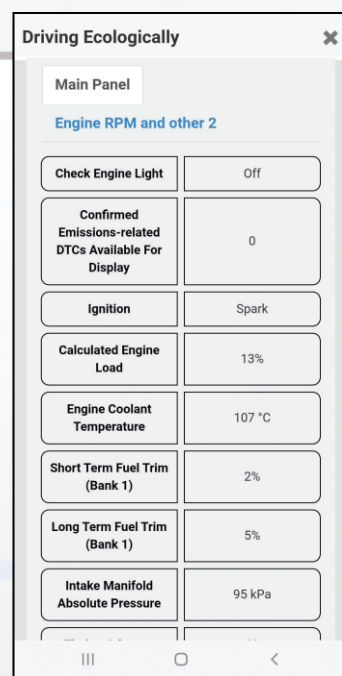


### My Data, KPI, POI

No.	High Level	Nature	Sub Nature	Value Name	Value Type	Data Type	Last Date	Last Value	Ownership	Username	Control	Data	Visibility
17057177	MyKPI	TransferServiceAndRenting	SensorSite	OBD2 - Vehicle Speed	pid_13(I3C)5444725267	integer	27/10/2019 15:26:00	0	private	badianover	OK	OK	OK
17057156	MyKPI	TransferServiceAndRenting	SensorSite	OBD2 - Vehicle Speed	pid_13(I3C)5444725267	integer	27/10/2019 12:58:55	0	private	badihelinski	OK	OK	OK
17057137	MyKPI	TransferServiceAndRenting	SensorSite	OBD2 - Vehicle Speed	pid_13(I3C)5444725267	integer	23/10/2019 15:49:04	126	private	badi toscana	OK	OK	OK
17056990	MyKPI	TransferServiceAndRenting	SensorSite	OBD2 - Vehicle Speed	pid_val_13(I3C)5444725267	integer	5/10/2019 15:36:02	10,75	private	paolotto2	OK	OK	OK
17056968	MyKPI	TransferServiceAndRenting	SensorSite	OBD2 - Vehicle Speed	pid_13(I3C)5444725267	integer	19/10/2019 19:17:31	100	public	badi toscana	OK	OK	OK



Tuscany in a  
Snap Mobile  
App on  
Android





TOP

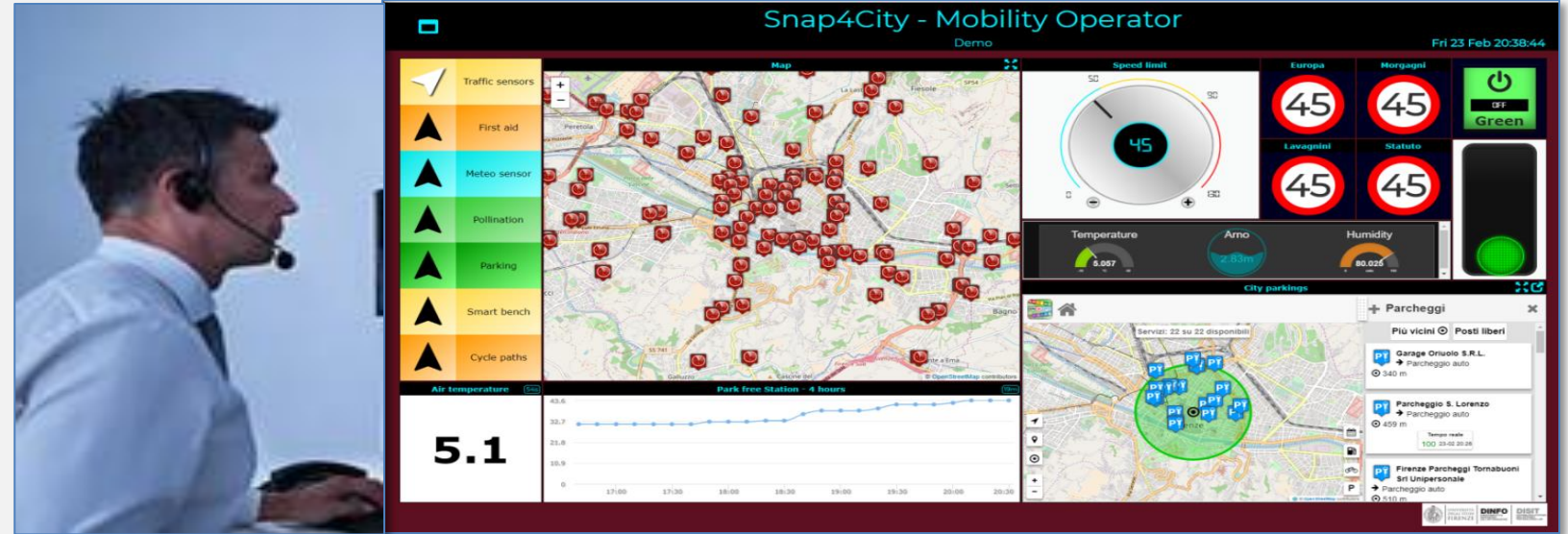
# *IOT App vs Smart City Solutions*



## 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



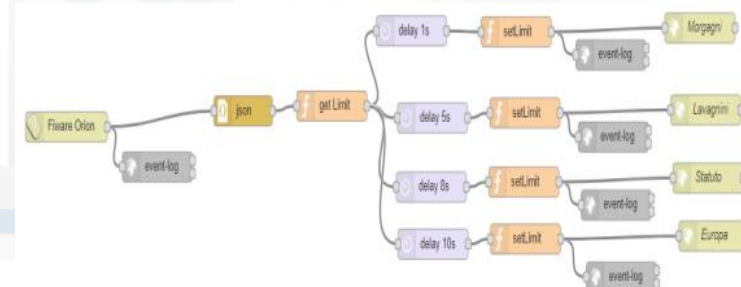
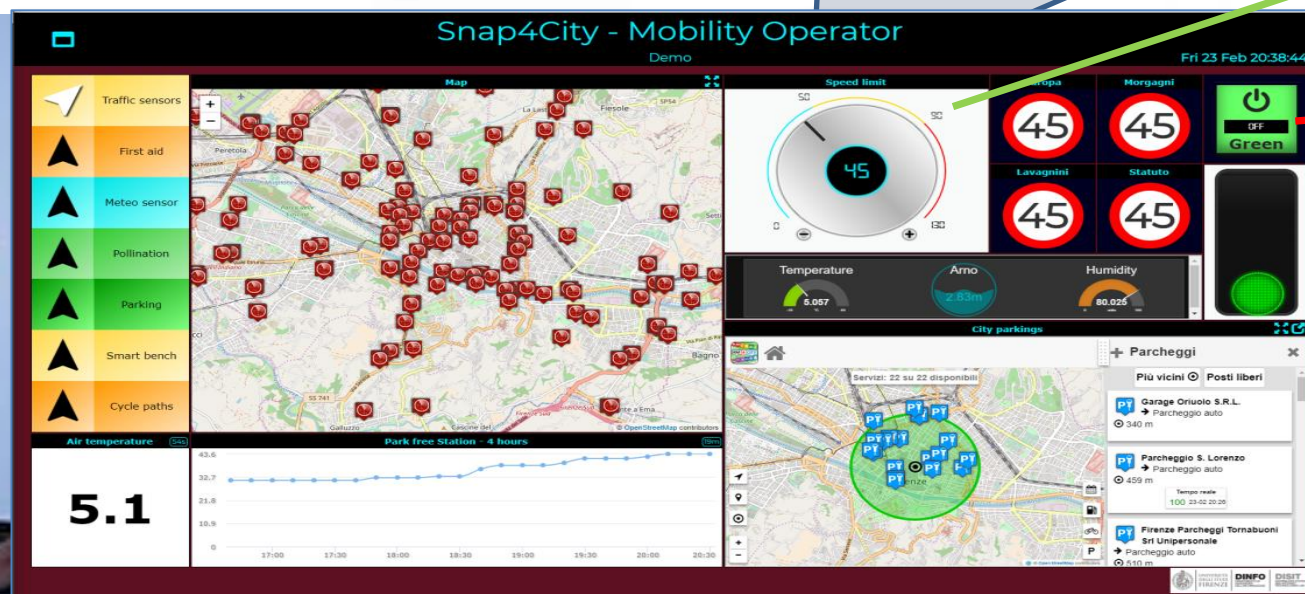
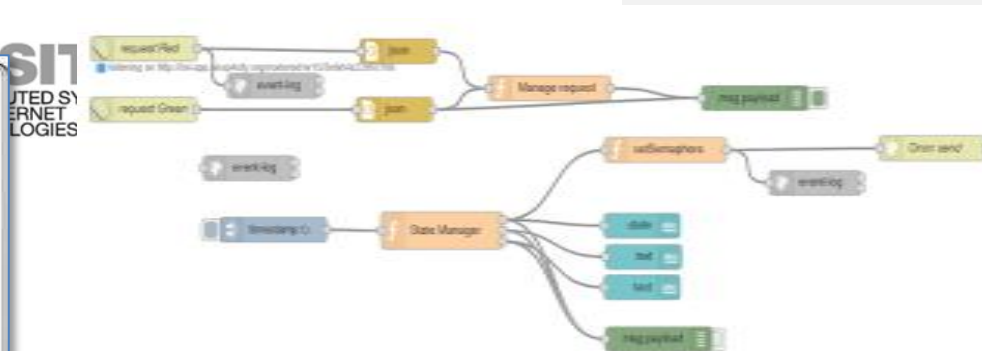
## Driver, Policeman

Would like to:

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





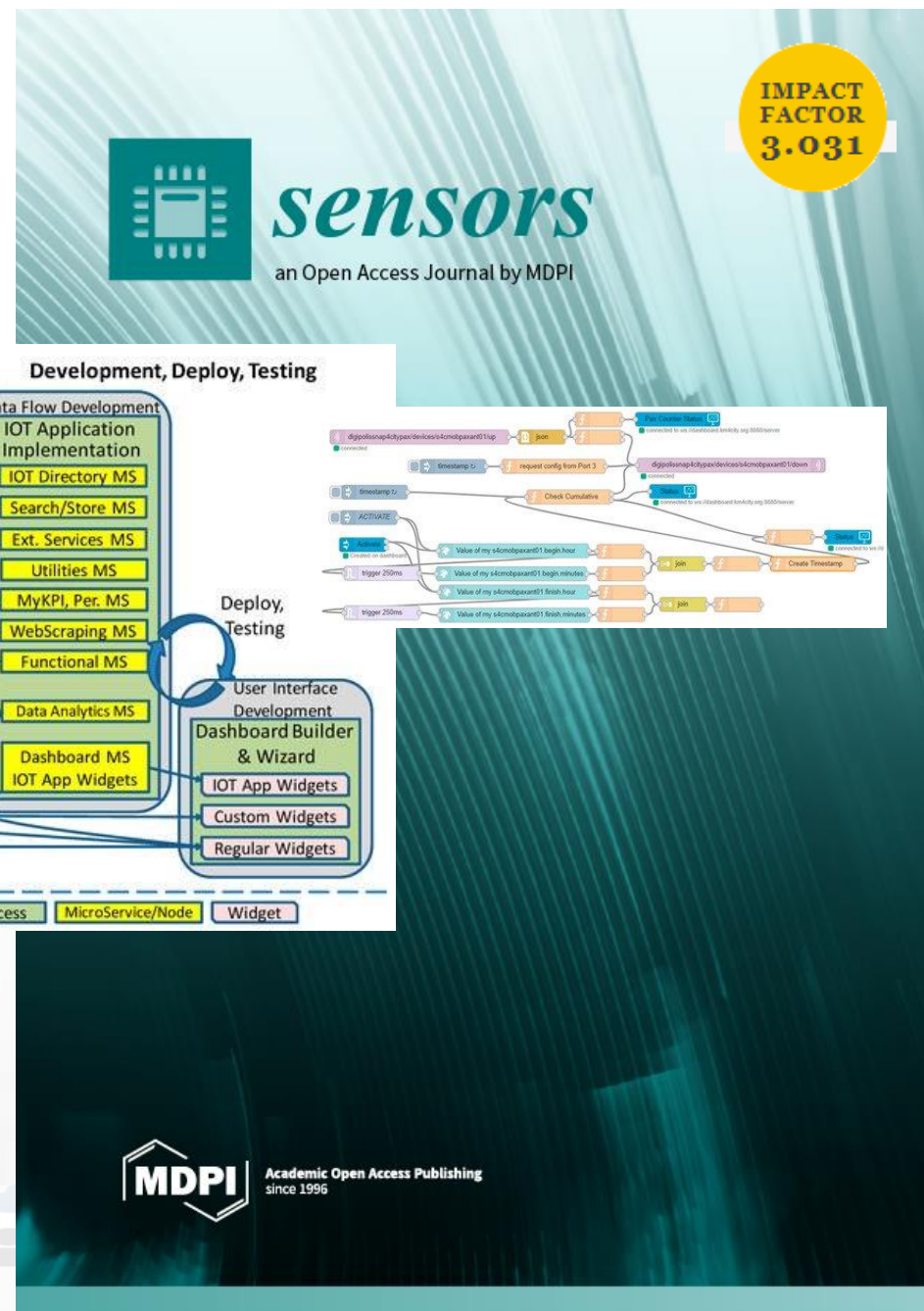
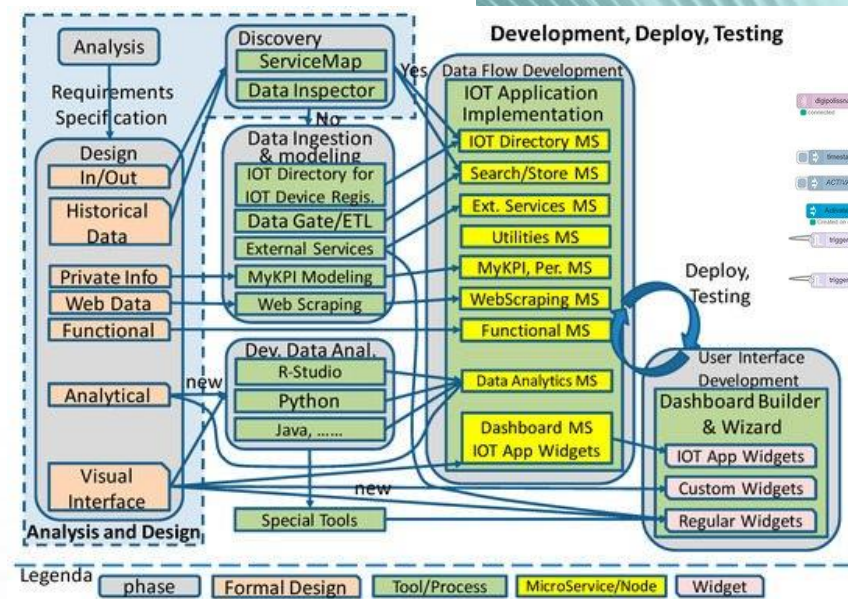




# MicroServices Suite for Smart City

- Badii, C.; Bellini, P.; Difino, A.; Nesi, P.; Pantaleo, G.; Paolucci, M. MicroServices Suite for Smart City Applications.

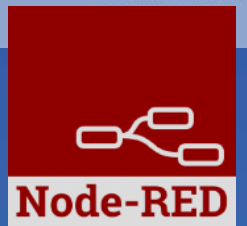
- *Sensors* **2019**, *19*, 4798.
- <https://www.mdpi.com/1424-8220/19/21/4798/pdf>





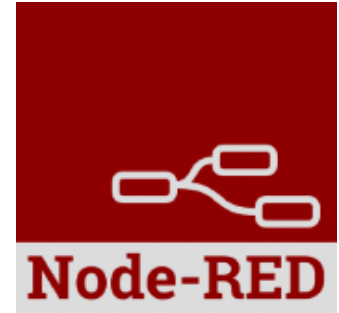
TOP

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



# IoT App / Proc.Logic

- Storage → IoT App / Proc.Logic
- External Service  $\leftrightarrow$  IoT App / Proc.Logic
- Dashboards  $\leftrightarrow$  IoT App / Proc.Logic



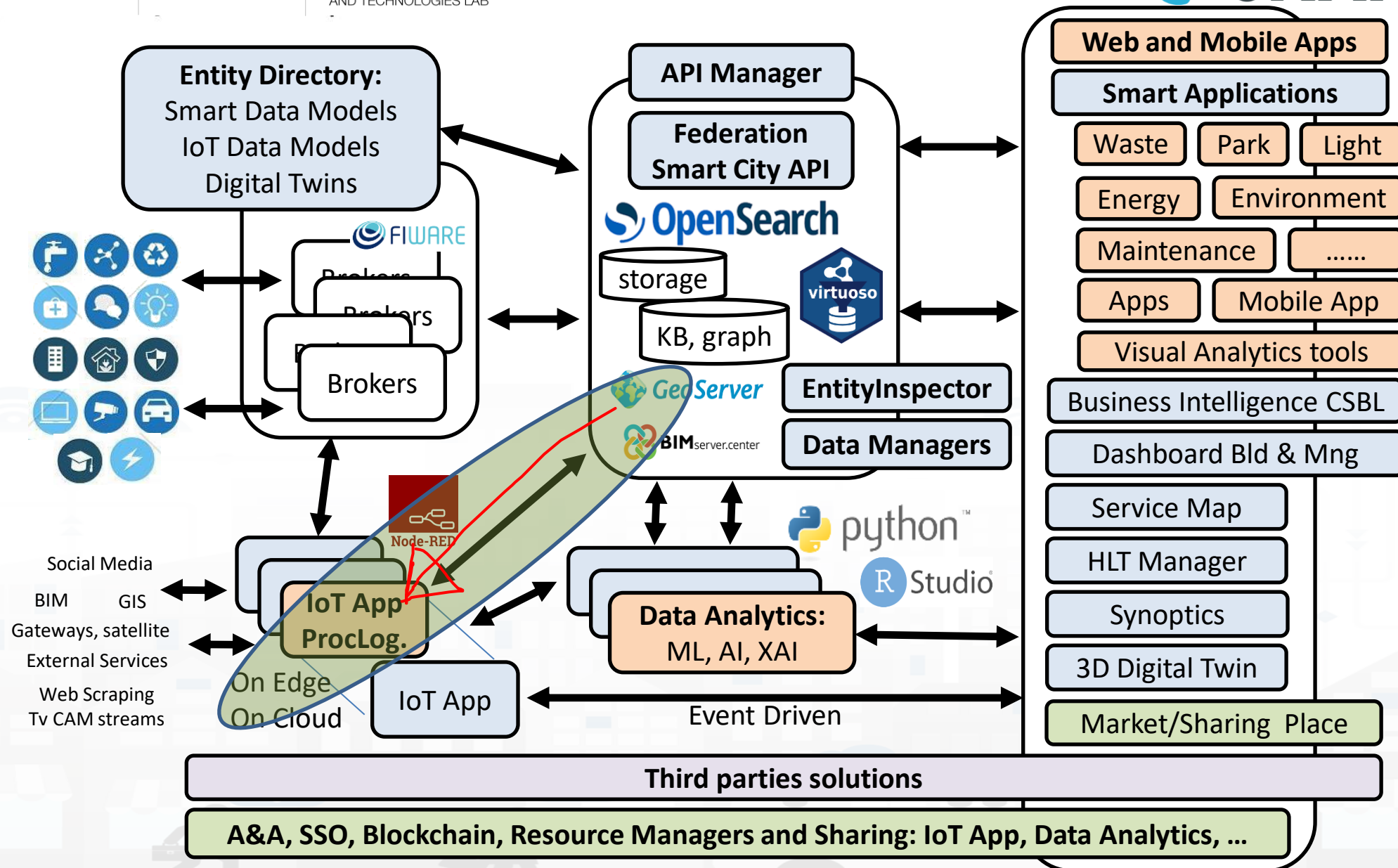
- Data Analytics  $\leftrightarrow$  IoT App / Proc.Logic
- Broker → Storage
- IoT App / Proc.Logic → Broker
- Broker → IoT App / Proc.Logic
- IoT App / Proc.Logic → Storage

Part 4

Part 5





# Tech Arch

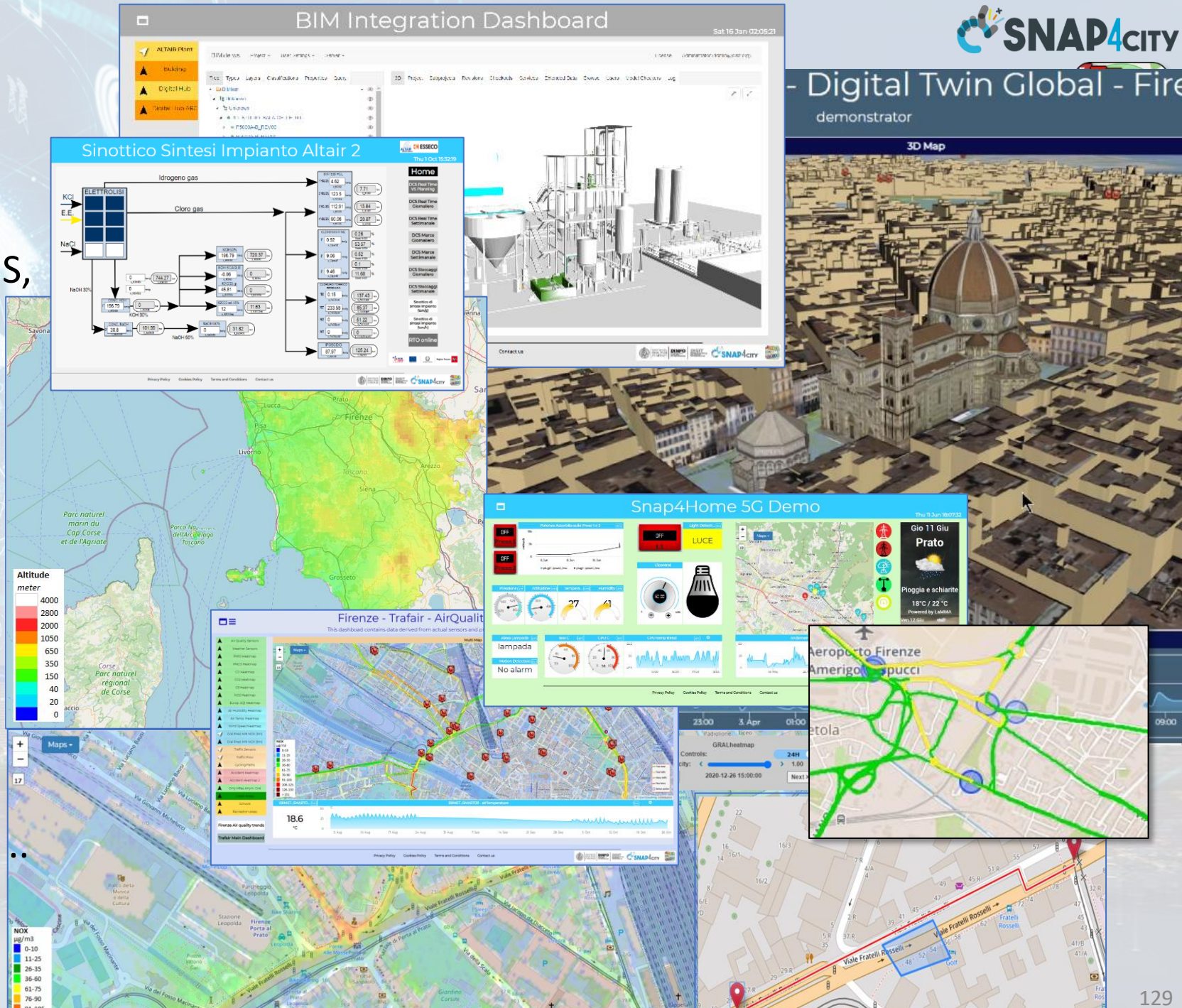




# High Level Types

Snap4City (C), November 2023

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





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

**Service Map (Toscana)**

User: roottooladmin1, Org: DISIT  
Role: RootAdmin, Level: 7  
[Logout](#)

My Snap4City.org  
Dashboards  
My Dashboards in All Org.  
Dashboards of My Organization  
My Dashboards in My Organization  
Extra Dashboard Widgets  
Notificator  
Data, my Data, OpenData  
Knowledge and Maps

**Service Map (Toscana)**

Service Map 3D (Firenze)  
Helsinki Service Map  
Antwerp Service Map  
Garda Lake Service Map  
Cagliari Service Map  
Lonato Del Garda Service Map  
Valencia Service Map  
Pont Du Gard Service Map  
Dubrovnik Service Map  
WestGreece Service Map  
Mostar-Bosnia Service Map  
Svealand Service Map  
Roma Service Map  
Pisa Service Map  
Creating WKT  
Service Map 3D (Antwerp)  
Service Map 3D (Helsinki)  
Producing POI triples for KB  
Load WKT on ServiceMap (Helsinki)  
Load WKT on ServiceMap (Toscana)  
Load WKT on ServiceMap (Antwerp)

Public transport Municipalities Text Search Address Search Events

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

[Position of selected Busses](#)

Actual Selection  
Service: IBIMET Air Quality Sensor - BORGO SAN LORENZO

**IBIMET Air Quality Sensor - BORGO SAN LORENZO**

Serviceuri: [http://www.disit.org/km4city/resource/IBIMET\\_SMART01](http://www.disit.org/km4city/resource/IBIMET_SMART01)  
Name: IBIMET\_SMART01  
Nature: Environment  
Subnature: Air\_quality\_monitoring\_station  
Address: BORGO SAN LORENZO  
City: FIRENZE

Property/Value Type	Value
PM10	2.4131048386898826
PM2_5	19.236197270630925
CO	0.22832953110492907
CO2	391.00
NO	
NO2	25.268744995957327
O3	128.39966613043157
airTemperature	18.60
airHumidity	73.60

Latest Update: 2020-10-26T17:46:50+02:00

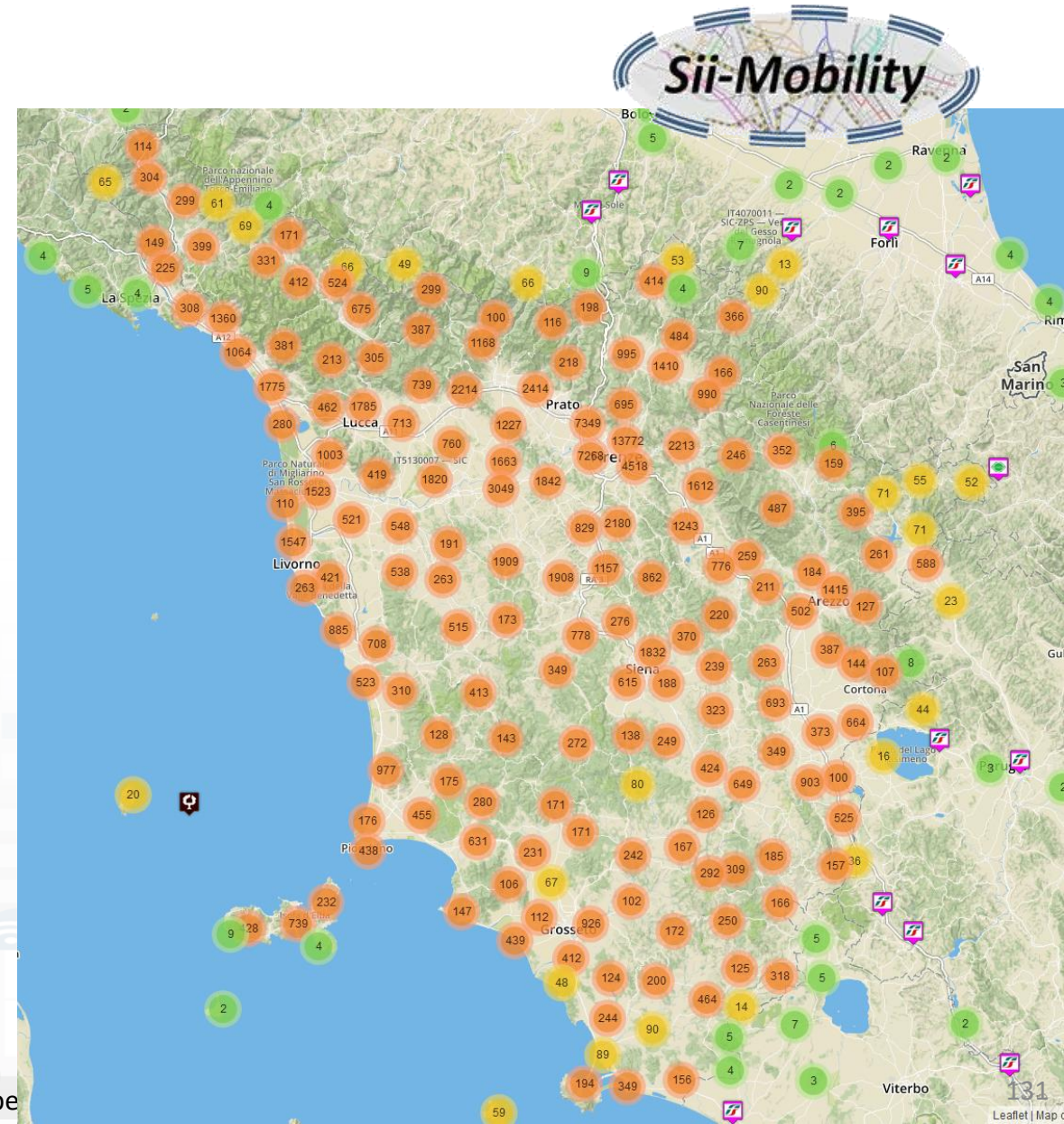
Regular Services  
Services Categories  
☐ De/Select All  
☒ Area  
☐ DigitalLocation  
☐ HappeningNow  
☐ IoTDevice  
☒ Path  
☐ Fresh Place  
☐ Public Transport  
☐ Road Sensors  
☐ Bus Stops  
☐ Tram stops  
☐ Subway station  
☐ Train station  
☐ Ferry stop  
☐ Car park  
☐ Bike sharing  
☐ RTZgate  
☐ Fuel station  
☐ Charging station  
☐ Underpass  
☒ Air quality monitoring  
☒ Pollen monitoring  
☒ Smart waste management  
☐ Smart irrigation  
☐ Weather sensor  
☐ Noise level sensor  
☐ People counter  
☐ Smart bench  
☐ First aid  
☐ Police headquarters

Filter:  
search text into service  
N. results for each: [No]  
Search Range [visible area]  
Search Results

Serviceuri: [http://www.disit.org/km4city/resource/IBIMET\\_SMART01](http://www.disit.org/km4city/resource/IBIMET_SMART01)

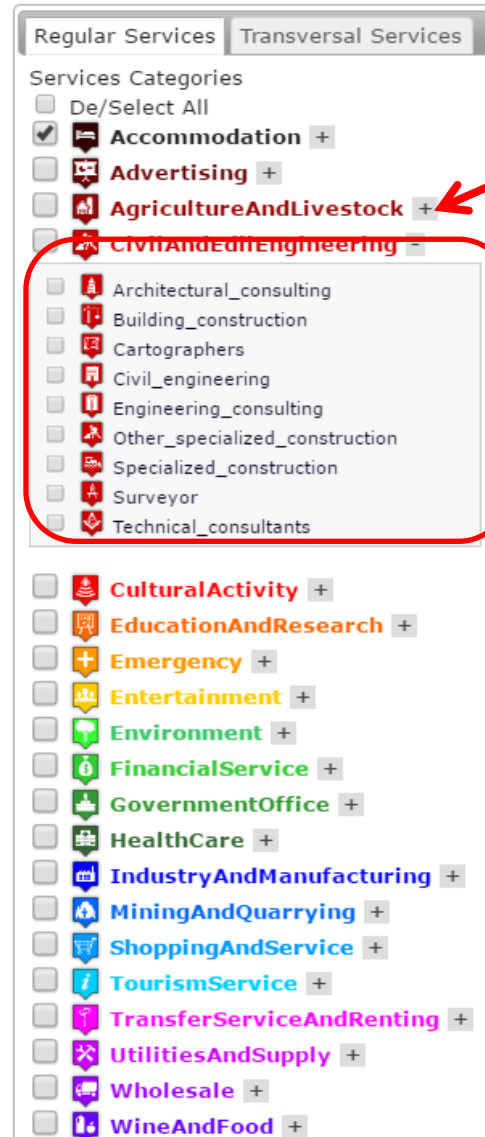


- **Street and geoinformation of the territory and details for routing, navigation, ...**
- **GeoResolution, Environmental data**
- **Mobility and Transport:** public and private, public transport, parking status, fuel stations prices, traffic sensors, etc.
- **Culture and Tourism:** POI, churches, museum, 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, ...



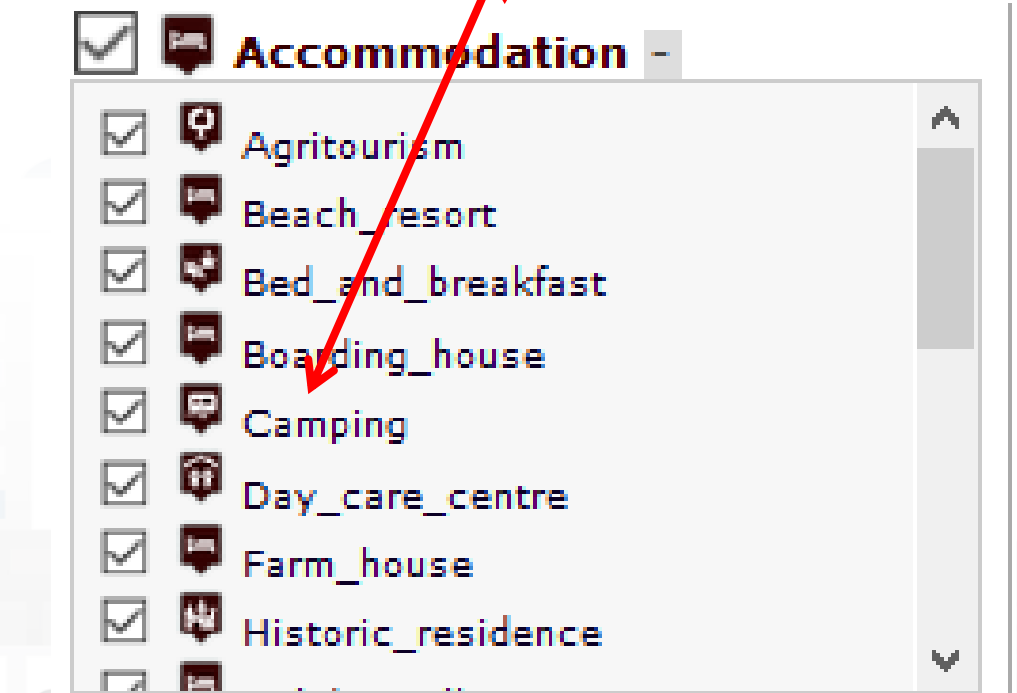
# Concepts of Services: Macro and subcategory

A SKOS area into  
the Km4City  
Ontology and  
Knowledge base  
for modeling POI  
and any element  
on map



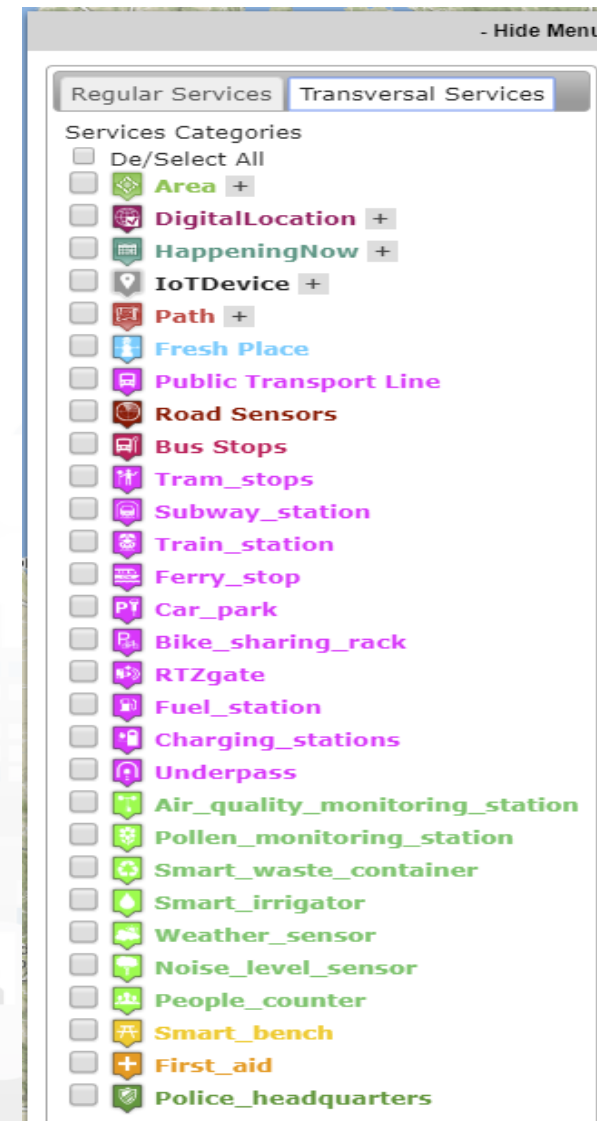
20 Service Macro Classes (The Nature)

Service subClasses (the SubNature)



# Access to Entities

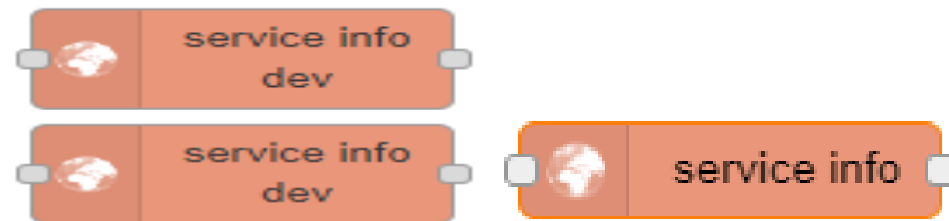
- IoT, POI, etc.
- **Classif**: macro (nature) and subcategories (subnature)
- **Position**: GPS, address, telephone, fax, email, URL, ...
- **Description**: textual, multilingual, with images, ...
- **Link** to dbPedia, Linked Open Data
- **Links to other services**
- **ActionURL**: links to actions 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





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

## S4CUtility



### Loggia San Paolo

[LINKED OPEN GRAPH](#)

Tipology: CulturalActivity - Monument\_location

Digital Location

Address: VIA DELLA SCALA, 3

Cap: 50123

City: FIRENZE

Prov.: FI

Photos:



Description: The rounded arches, the stone skeleton and the glazed terracotta medallions recall the model of the Loggiato degli Innocenti. The medallions in glazed terracotta by Andrea della Robbia and his sons Marco and Luca contain seven polychrome figures of Santi Francescani and two works of mercy Cristo conforta un Giovane and Cristo conforta un Anziano. Beneath the portico can be admired the expressive embrace between San Domenico Guzman and San Francesco d Assisi by Andrea della Robbia

### TPL STOP : Piazza Stazione (Fr. Cc)

Vaubus

[LINKED OPEN GRAPH](#)

Lines:

FI-LU FI-VG

No available routes

Display 50 Bus per page

Search:

Time	Line	Direction
06:46:00 2017-03-20	FI-LU	Piazzale Verdi
08:16:00 2017-03-20	FI-LU	Piazzale Verdi
10:09:00 2017-03-20	FI-LU	Piazzale Verdi
11:09:00 2017-03-20	FI-LU	Piazzale Verdi
12:16:00 2017-03-20	FI-LU	Piazzale Verdi
13:16:00 2017-03-20	FI-LU	Piazzale Verdi

Showing page 1 of 1

Real-time data currently not available

### AURORA

[LINKED OPEN GRAPH](#)

Tipology: Accommodation - Hotel

Email: [info@hotelauroa.info](mailto:info@hotelauroa.info)

Website: [www.hotelauroa.info](http://www.hotelauroa.info)

Phone: 055210283

Address: VIA L. ALAMANNI, 5

Cap: 50100

City: FIRENZE

Prov.: FI

### Giardino di piazza dell'Indipendenza

[LINKED OPEN GRAPH](#)

Tipology: Entertainment - Green\_areas

Digital Location

Address: PIAZZA DELLA INDIPENDENZA, 15

Cap: 50129

City: FIRENZE

Prov.: FI

Note: areeverdi238

Remove from map

### ZCS\_1\_D

[LINKED OPEN GRAPH](#)

Tipology: TransferServiceAndRenting - Controlled\_parking\_zone

Digital Location

Address: VIA GUSCIANA

Cap: 50124

City: FIRENZE

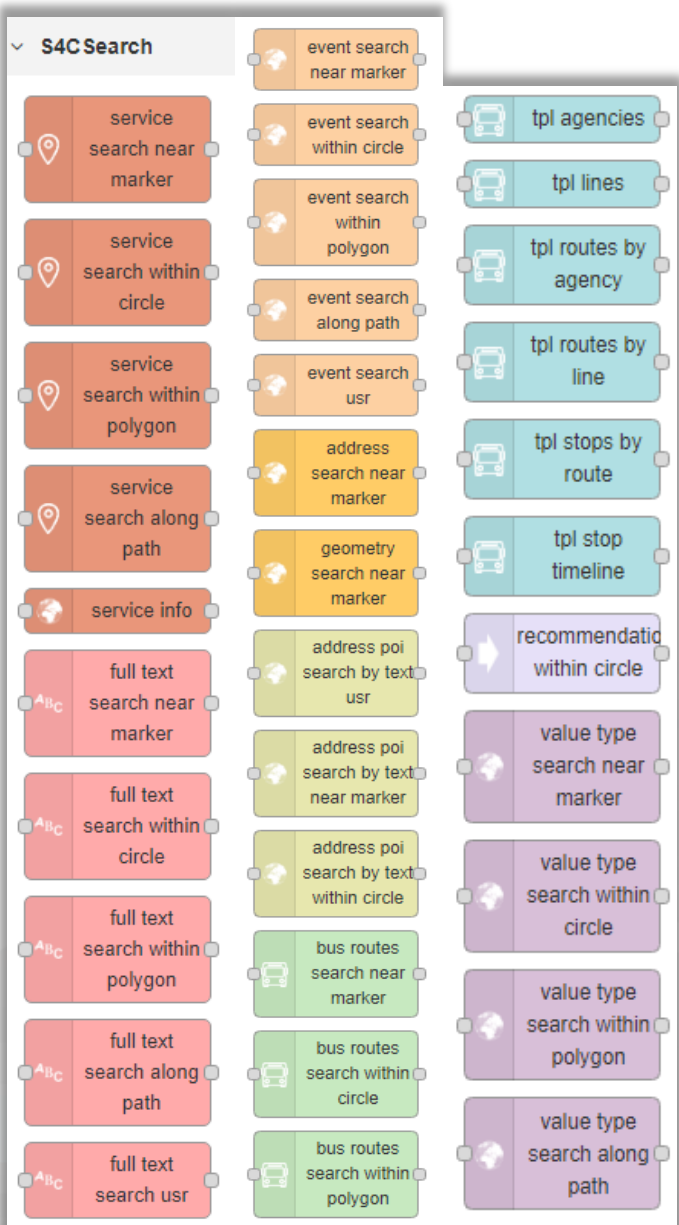
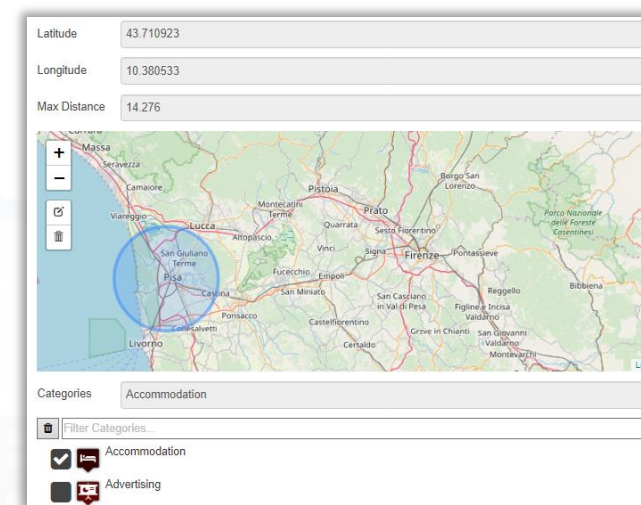
Prov.: FI

Remove from map

## 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



The screenshot shows the Node-RED interface with the 'Edit device-registration node' dialog open. The dialog has a 'Delete' button at the top right. Below it is the 'node properties' section, which includes a 'Device Name' text input field and a 'Model' dropdown menu currently set to 'Raspberry snapclity 1'. A map of the region around Florence, Italy, is displayed below the properties. At the bottom of the dialog, there are two text input fields for 'Key 1' and 'Key 2', each containing a long alphanumeric string. A green 'Check!' button is located at the bottom left of the dialog. The background shows the Node-RED flow editor with various nodes like 'feedparser', 'exec', 'S4CLogDev', 'event.log', 'S4CIoT', 'device-registration', and 'iot directory'.



# Smart City Entities Advanced Search

Flexibility

- 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

Latitude

Longitude

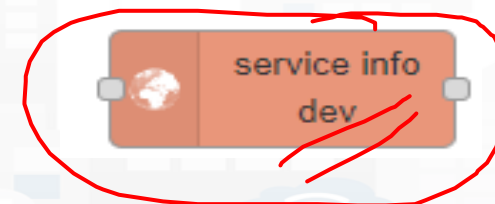
Categories

Max Distance (in km)

Max Results (0 for all Results)

Geometry ☐

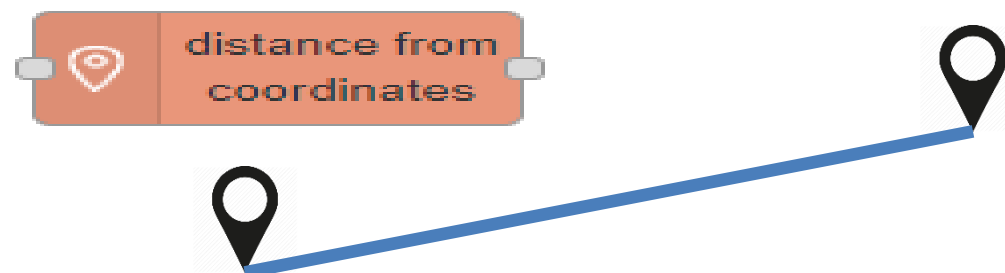
Language



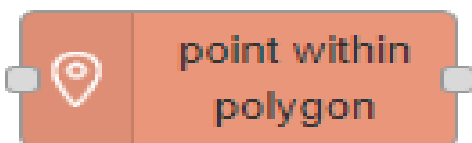
**S4CSearchDev**

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

- Distance from GPS point



- Point  is in Polygon ?  
– Polyline as WKT



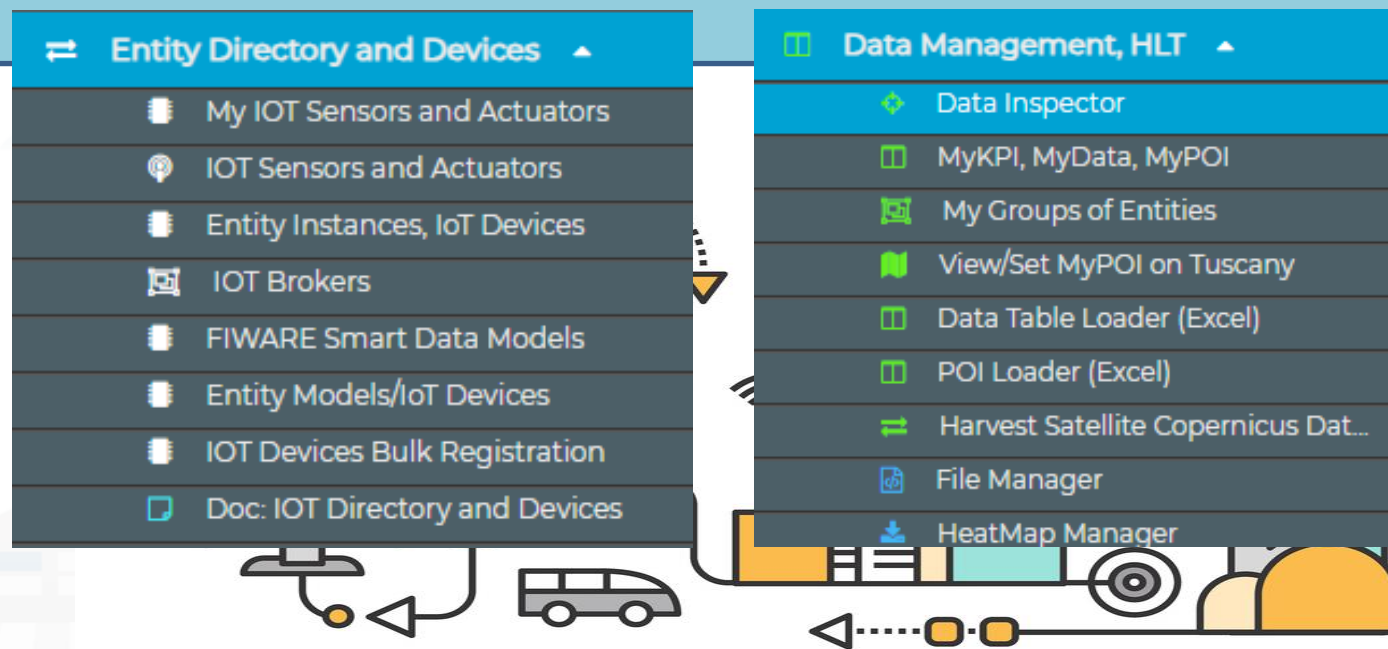
## On video

- 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



TOP

# *search vs services, the ServiceURI*



# Understanding / Testing an Entity/ IoT Device

	AdminDevice001	orionUNIFI	Ambiental		MYOWNPRIVATE	active				
Broker URI: <a href="https://broker1.snap4city.org">https://broker1.snap4city.org</a>					Broker Port: 8080					
Kind: sensor					Visibility: MyOwnPrivate					
Device Type: Ambiental					Format: json					
Protocol: ngsi					MAC:					
Model:					Producer: Raspberry PI					
Longitude: 9.228193					Latitude: 45.499369					
Device Uri: <a href="http://www.disit.org/km4city/source/iot/orionUNIFI/AdminDevice001">http://www.disit.org/km4city/source/iot/orionUNIFI/AdminDevice001</a>										
Organization: DISIT										
K1: b7c4115-f25c-4cb6-95eb-e4b363222bef					K2: 441ffb6c-dc8a-4fc9-a415-7f6564d656f5					
Created at: 2018-05-24 21:54:03										



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

The Broker

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

See IoT Device on ServiceMap

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



### Snap4City

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

LOGOUT

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

### Data Inspector

Map

Single data widgets

Multi data widgets

Map Controls:

FilterMap GPSUser GPSOrg

Sensor

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

Hide columns

Last Value

14.9

Data sources Details

Device	Values	Healthiness	Process	Image	Licensing	User
GPS Coordinates:	42.642033, 18.1122					
High-Level Type:	Sensor					
Nature:	From IOT Device to KB					
Subnature:	IoT Sensor					
Value Name:	Dubrovnik:orionDubrovnik-UNIFI:camera_Dubrovnik_1_Place					
Device ServiceURI or Data ID:	http://www.disit.org/km4city/resource/iot/orionDubrovnik-UNIFI/Dubrovnik/camera_Du					
Sensor ServiceURI or Data ID:	http://www.disit.org/km4city/resource/iot/orionDubrovnik-UNIFI/Dubrovnik/camera_Du					

Datasource: IoT

Ownership: private

Organizations: Dubrovnik

Link to Service Map Link to IoT Device

Healthiness

Check

Ownership

Healthiness	Check	Ownership
●	2019-08-13 07:18:30	public
●	2019-08-13 07:18:30	public
●	2019-08-13 07:18:30	public
●	2019-08-13 07:18:30	public
●	2019-08-13 07:18:30	public
●	2019-08-13 07:18:30	public
●	2019-08-13 07:17:27	public
●	2019-08-13 07:17:27	public

Search

# MAP4CITY

- Click with the mouse on it

## Data Inspector Wizard

Knowledge Base view

Link to Service Map

Link to IoT Device

### IOT Devices

Show: 1 entries

IoT Device	IoT Device	Device Type	Model	Ownership	Status	Edit	Delete	Location
AccessPoint_FamiasSuperstore	orionLoratoCelCardie-UNIFI	AccessPointSensor	AccessPointLorato	DELEGATED	active	EDIT	DELETE	
AccessPoint2_T16	orionLoratoCelCardie-UNIFI	AccessPointSensor	AccessPointLorato	DELEGATED	active	EDIT	DELETE	
AccessPoint3_Palestina	orionLoratoCelCardie-UNIFI	AccessPointSensor	AccessPointLorato	DELEGATED	active	EDIT	DELETE	
AdminDev1	orionUNIFI	Ambiental	MYOWNPRIVATE	active	EDIT	DELETE		
AdminDevice001	orionUNIFI	Ambiental	MYOWNPRIVATE	active	EDIT	DELETE		
AdminDevice002	orionUNIFI	Ambiental	MYOWNPRIVATE	active	EDIT	DELETE		
AdminDevice004	orionUNIFI	Ambiental	MYOWNPRIVATE	active	EDIT	DELETE		
AdminDevice009	orionUNIFI	Ambiental	MYOWNPRIVATE	active	EDIT	DELETE		
AdminDevice010	orionUNIFI	Ambiental	MYOWNPRIVATE	active	EDIT	DELETE		
AdminTest005	orionUNIFI	Ambiental	MYOWNPRIVATE	active	EDIT	DELETE		

Showing 1 of 370 entries

Previous 1 2 3 4 5 ... 37 Next

Some functionalities are limited to certain roles

Snap4City (C), November 2023

142



# Notation Terminology

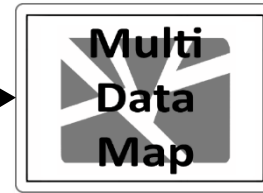
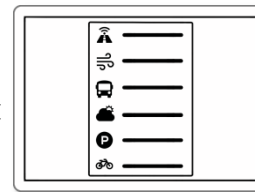
WHERE	Are synonymous at level of service which can be <b>IoT device or entity</b> with data and references to	Are synonymous at level of the single <b>attribute of the entity</b> , device, service, etc.
IoT Directory, Entity Directory	IoT Device, Entity Instance, Device URI	Sensor, Actuator, Attributes,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

## ServiceURI, SURI of a sensor device:

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

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

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



# Dashboard Usage and recipe: Event map target

- **Selector to Show on Map a**
  - **category of Map positioned elements**
    - [https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.08694333811321;8.791809082031252;44.93758500391093;14.065246582031252&categories=Traffic\\_sensor&maxResults=0&maxDists=0.1&text=&model=&value\\_type=&format=json](https://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.08694333811321;8.791809082031252;44.93758500391093;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](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=Firenze\\_PM10](https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=Firenze_PM10)
  - **Traffic flow**
    - <https://wmsserver.snap4city.org/geoserver/Snap4City/wms?service=WMS&layers=FirenzeFIPILITrafficRealtime&trafficflowmanager=true>
    - <https://firenzetraffic.km4city.org/trafficRTDetails/roads/read.php>
  - **Origin Destination Map**
    - [https://odmm.snap4city.org/api/get?precision=communes&from\\_date=&organization=Toscana&inflow=True&longitude=11.255751&latitude=43.769710&od\\_id=mobile\\_Toscana\\_1000&perc=True](https://odmm.snap4city.org/api/get?precision=communes&from_date=&organization=Toscana&inflow=True&longitude=11.255751&latitude=43.769710&od_id=mobile_Toscana_1000&perc=True)
- **Events which are also PIN on map**
- **Il Service URI as the unique identifier of the Entity**
  - <http://www.disit.org/km4city/resource/iot/orionUNIFI/DISIT/METRO632>

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

**Snap4City**

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

LOGOUT

- My Snap4City.org
- Tour Again
- ダッシュボード
- Dashboards (Public)
- My Dashboards in All Org.
- Dashboards of My Organization
- My Dashboards in My Organization
- My Data Dashboard Dev Kibana
- My Data Dashboard Kibana
- Extra Dashboard Widgets
- Notifier
- Data, my Data, OpenData
- Knowledge and Maps
- IOT Applications
- IOT Directory and Devices
  - My IOT Sensors and Actuators
  - IOT Sensors and Actuators
  - IOT Devices**
  - IOT Devices Management
  - IOT Device Discovery
  - IOT Brokers
  - IOT Device Models
  - IOT Devices Bulk Registration
  - IOT Broker Periodic Update setting
  - IOT Orion Broker Mapping Rules
  - Doc: IOT Directory and Devices
  - Create an IOT Device Instance
  - Create an IOT Device Model
  - Add an IOT Device into Snap4City
- Resource Manager

**IOT Devices**

Show  entries

Search:  [New Device](#)

Device Identifier	IOT Broker	Device Type	Model	Ownership	Status	Edit	Delete	Location	View
+ 15EP22T2AA1S000022	orionFirenze-UNIFI	ChargingStation	ChargingStationModel	PUBLIC	active	EDIT	DELETE		VIEW
+ AccessPoint1_FamilaSuperstore	orionLonatoDelGarda-UNIFI	AccessPointSensor	AccessPointLonato	DELEGATED	active	EDIT	DELETE		VIEW
+ AccessPoint2_ITIS	orionLonatoDelGarda-UNIFI	AccessPointSensor	AccessPointLonato	DELEGATED	active	EDIT	DELETE		VIEW
+ AccessPoint3_Palasport	orionLonatoDelGarda-UNIFI	AccessPointSensor	AccessPointLonato	DELEGATED	active	EDIT	DELETE		VIEW
+ adminDev1	orionUNIFI	Ambiental		MYOWNPUBLIC	active	EDIT	DELETE		VIEW
- AdminDevice001	orionUNIFI	Ambiental		MYOWNPRIVATE	active	EDIT	DELETE		VIEW

Broker URI: <https://broker1.snap4city.org>

Kind: sensor

Device Type: Ambiental

Protocol: ngsi

Model:

Longitude: 9.228193

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

Organization: DISIT

PAYLOAD NGSI v1

K1: b7c4c15-f25c-4cb6-95eb-e4b363222bef

Created on: 2018-05-24 21:54:03

Broker Port: 8080

Visibility: MyOwnPrivate

Format: json

MAC:

Producer: Raspberry PI

Latitude: 45.499369

VIEW IN SERVICE MAP

NEW DATA IN AdminDevice001

PAYLOAD NGSI v2

K2: 441ffb6c-dc8a-4fc9-a415-7f6564d656f5

+ AdminDevice002	orionUNIFI	Ambiental		MYOWNPRIVATE	active	EDIT	DELETE		VIEW
+ AdminDevice004	orionUNIFI	Ambiental		MYOWNPRIVATE	active	EDIT	DELETE		VIEW
+ AdminDevice005	orionUNIFI	Ambiental		MYOWNPRIVATE	active	EDIT	DELETE		VIEW
+ AdminDevice1	orionUNIFI	Ambiental		MYOWNPRIVATE	active	EDIT	DELETE		VIEW

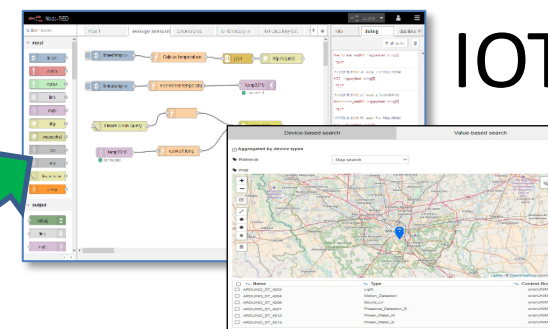
Showing 1 to 10 of 462 entries

Previous 1 2 3 4 5 ... 47 Next

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



# IOT Apps

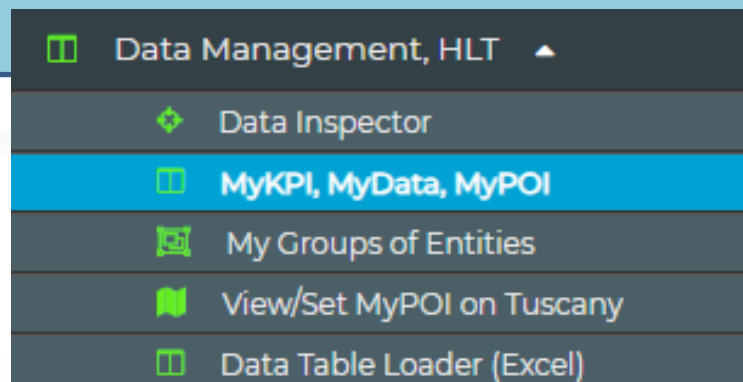


# DataInspector Dashboard Wizard



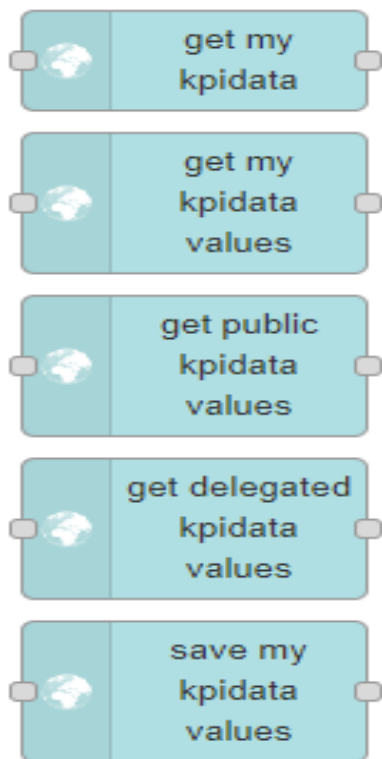
TOP

# *MyKPI Nodes*



- Save and retrieve MyKPI into the safe personal data storage

## ▼ S4CKPIData



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



TOP

## External Service ↔ IoT App / Proc.Logic

FROM CITY  
DASHBOARD TO  
APPLICATIONS

DATA GATHERING  
AND CITY DATA  
KNOWLEDGE  
MANAGEMENT

FORGING &  
MANAGING OPEN  
AND FLEXIBLE WEB

SNAP4CITY FOR

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

Processing Logics / IoT App

Processing Logics / IoT App

MicroServices for Proc.Logic/IoT App

MicroServices from DataAnalytic

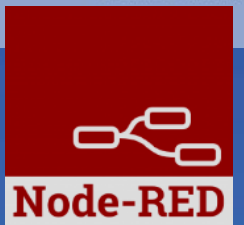
IOT MicroServices for Final Users

IOT MicroServices for Developers

DOC: Processing Logic/IoT App

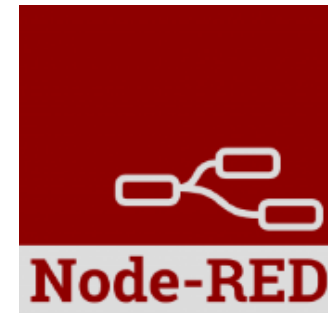
How to Develop Proc.Logic / IoT App

Create A MicroService from RestCall



# IoT App / Proc.Logic

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

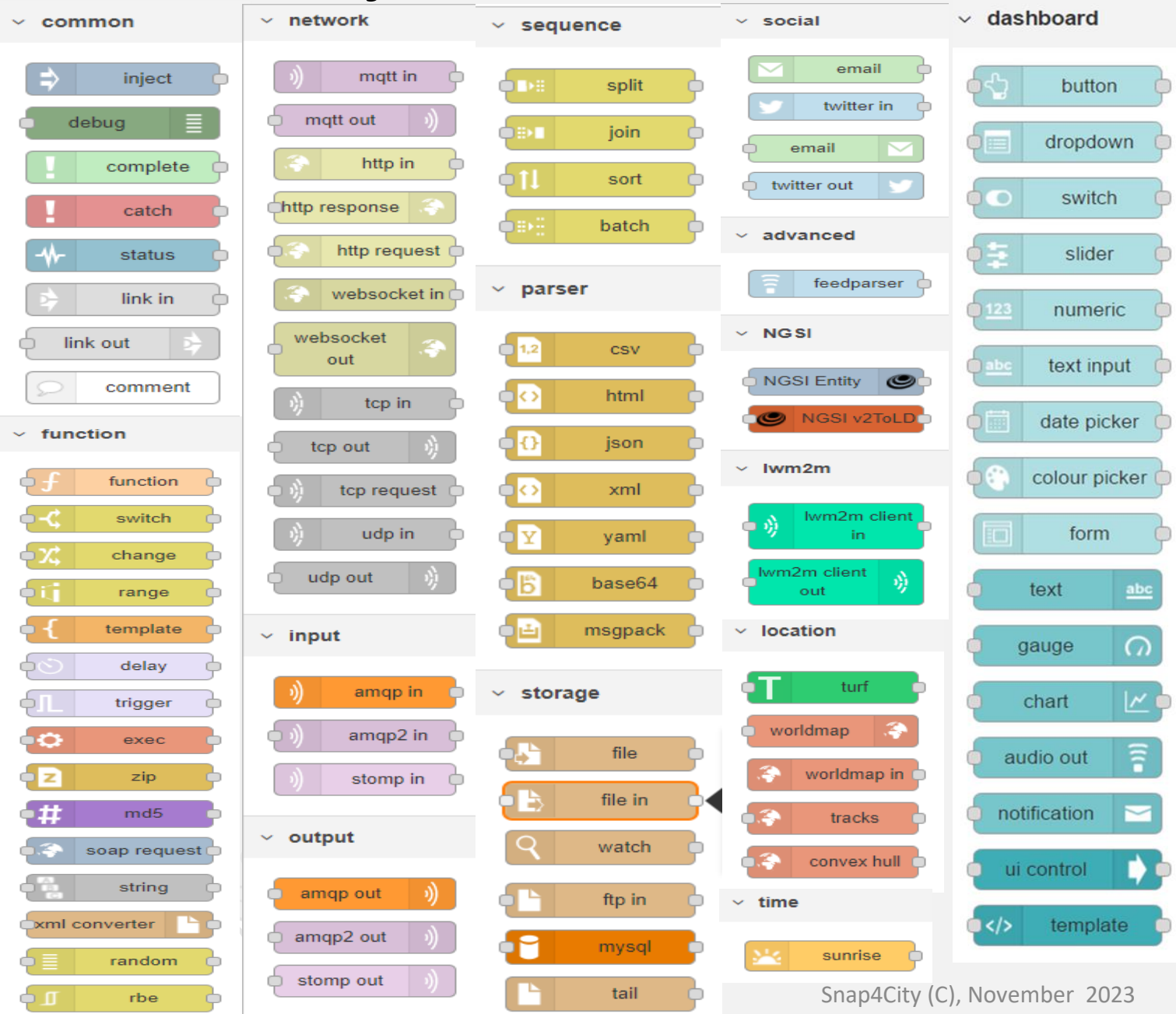


- Data Analytics ↔ IoT App / Proc.Logic
- Broker → Storage
- IoT App / Proc.Logic → Broker
- Broker → IoT App / Proc.Logic
- IoT App / Proc.Logic → Storage

Part 4

Part 5

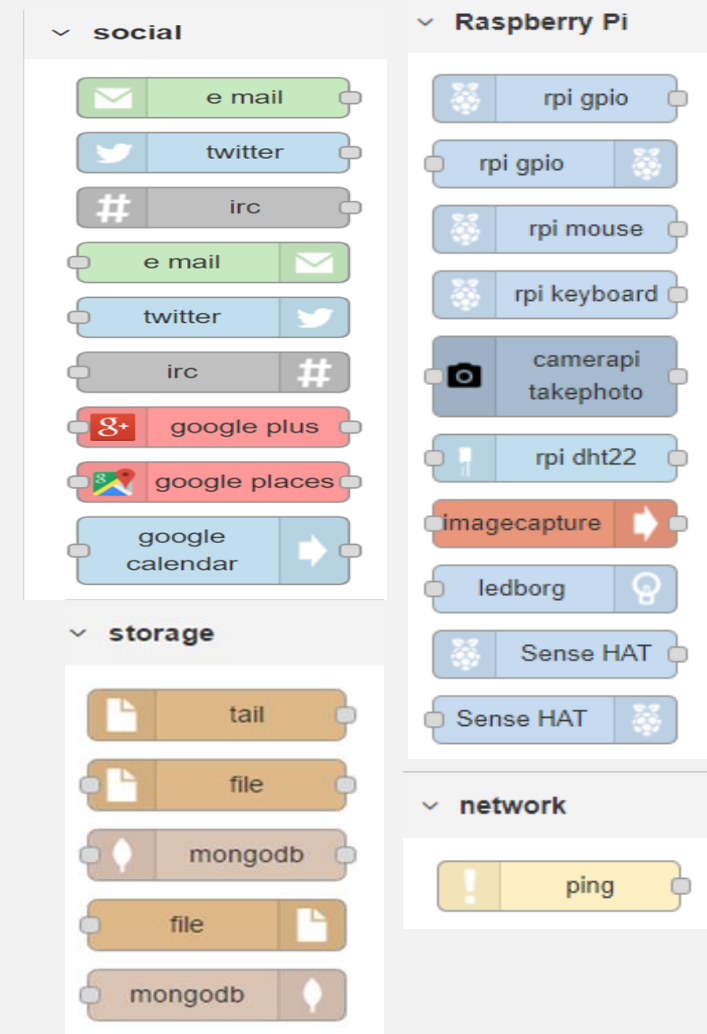
# Basic Node.js Blocks on NodeRed on our Advanced IOT Apps



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

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

+ on IOT Edge Raspberry

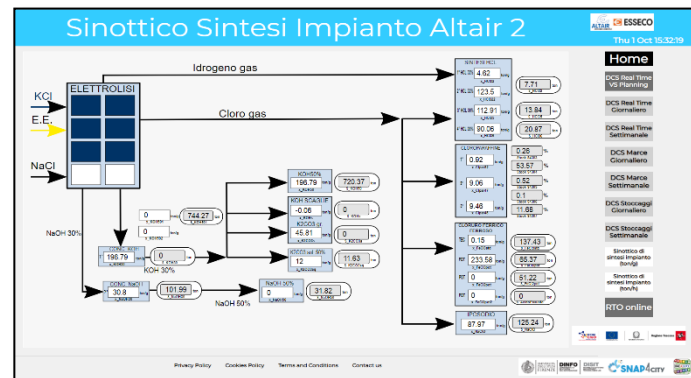


The screenshot displays the Node-RED block palette with the following categories and blocks:

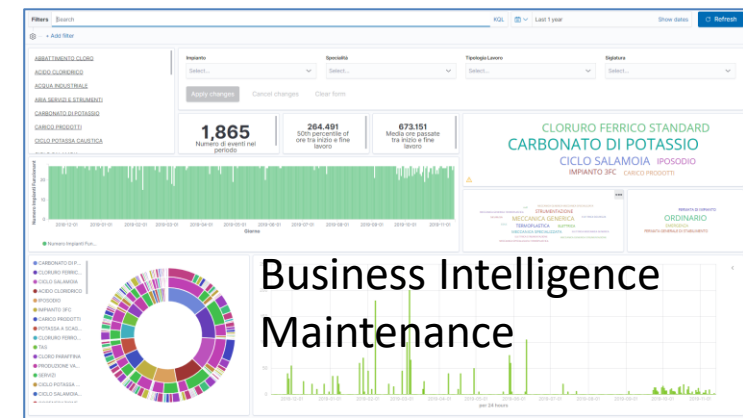
- social**: e mail, twitter, irc, e mail, twitter, irc, google plus, google places, google calendar.
- storage**: tail, file, mongodb, file, mongodb.
- Raspberry Pi**: rpi gpio, rpi gpio, rpi mouse, rpi keyboard, camerapi takephoto, rpi dht22, imagecapture, ledborg, Sense HAT, Sense HAT.
- network**: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



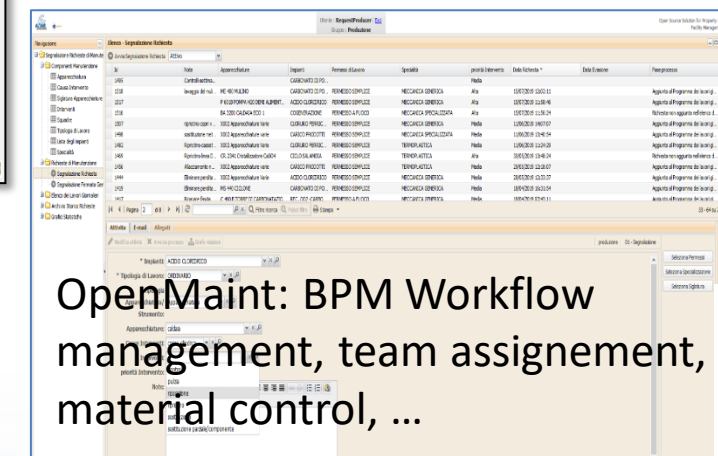
# Example of Integrated workflow



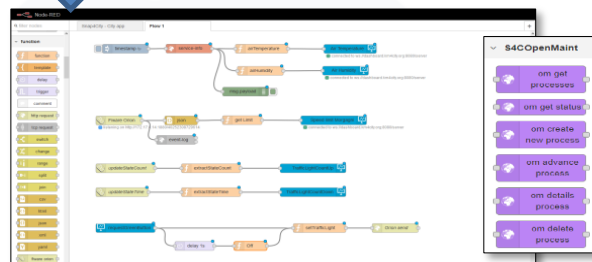
Consumptions/productions



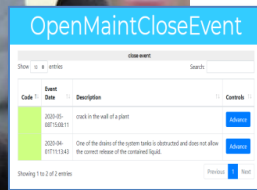
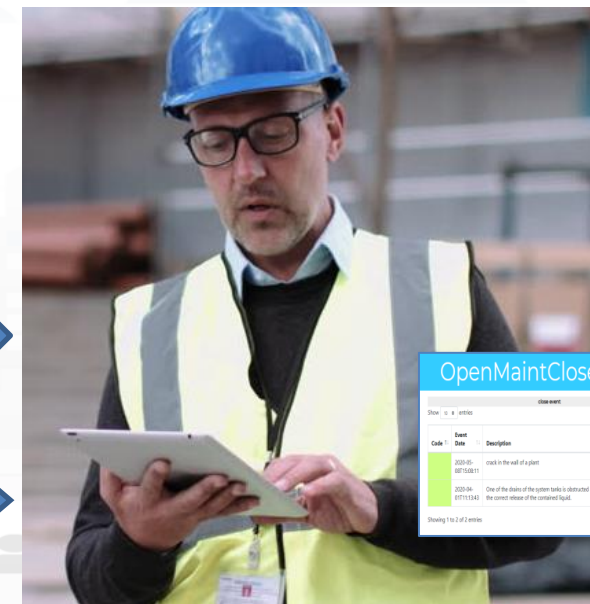
Events/actions



Dashboards and actions



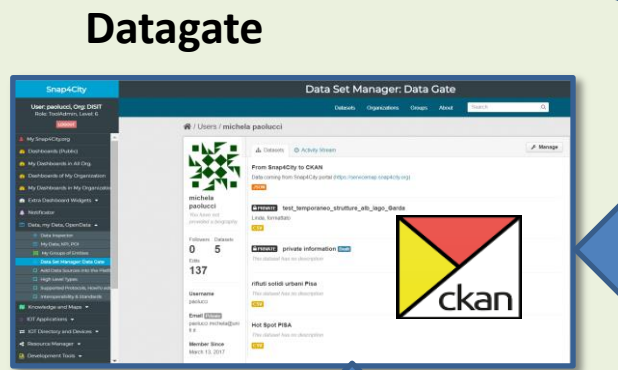
IOT App, Data event firing, event detection and firing Critical event management



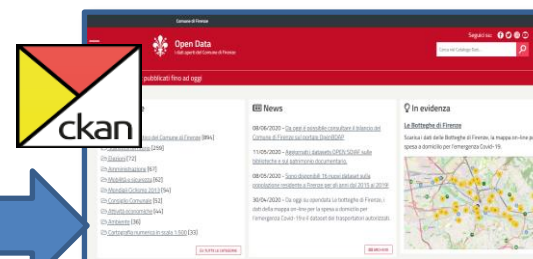
# Snap4City vs CKAN

**Snap4City Portal and Integrated tools**

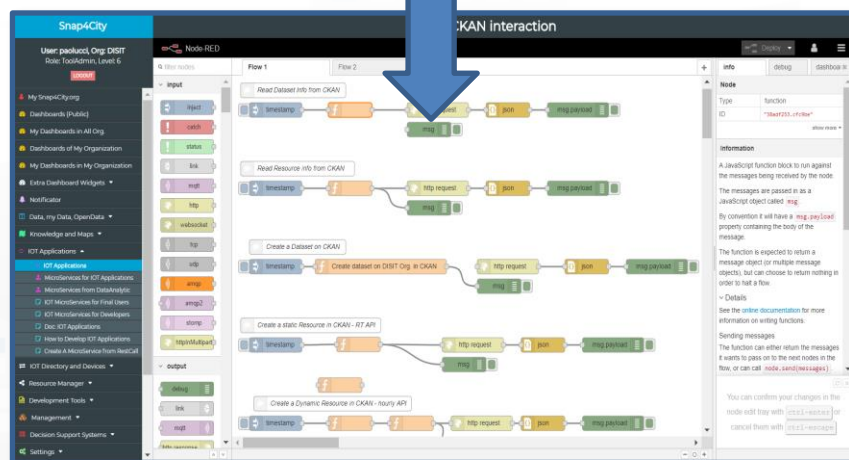
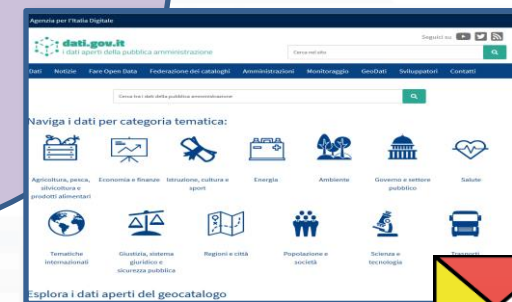
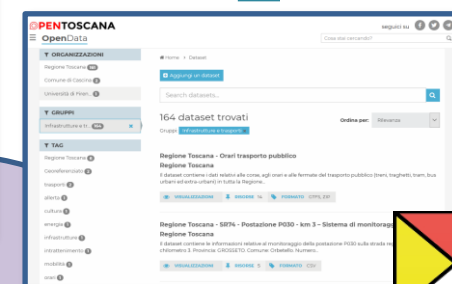
**Advanced Snap4City APIs and Micro Services**



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 automatically transformed into a MicroService for the IOT Applications

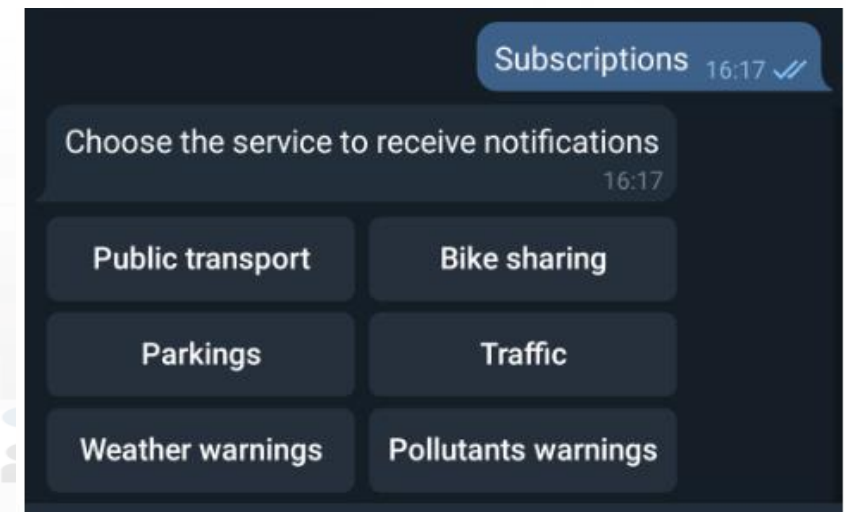
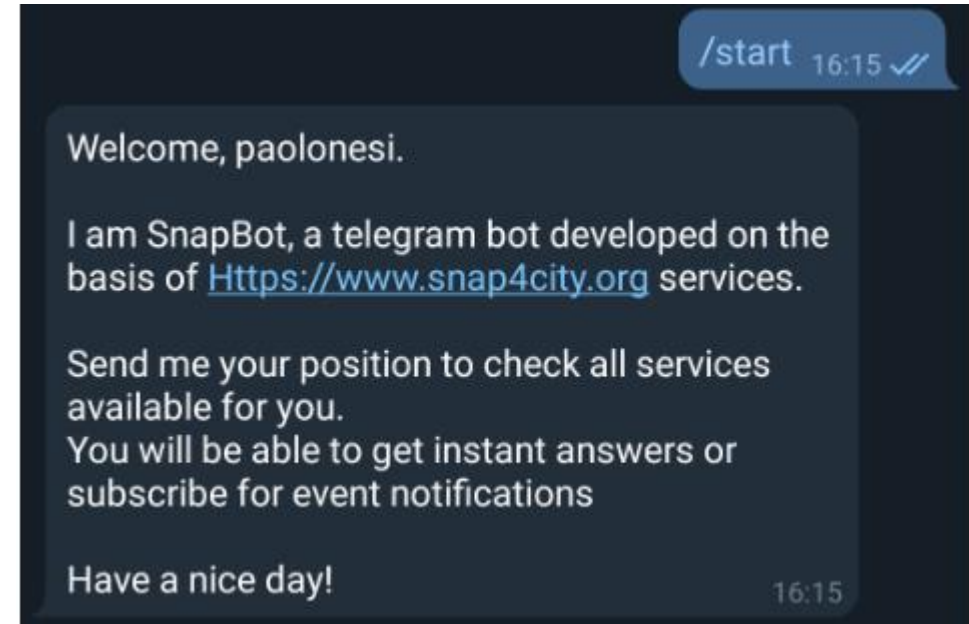
<https://www.snap4city.org/129>

File Name	Upload Date	Description	Control Status	View	Metadata	Published	Delete
Air quality.zip	2018-05-25 13:10:35	Air quality Microservice	OK - 2018-05-25 13:10:35	VIEW	EDIT	NO	DEL
Antwerp cameras location.zip	2019-01-13 17:22:06	Antwerp cameras location from A Open Data	OK - 2019-01-13 17:22:06	VIEW	EDIT	YES	DEL
Antwerp museum.zip	2019-01-13 17:27:08	Antwerp museum (data coming from the A Open Data API)	OK - 2019-01-13 17:27:08	VIEW	EDIT	NO	DEL
Antwerp velo stations.zip	2019-01-13 17:32:17	Antwerp velo stations location (data coming from A Open Data API)	OK - 2019-01-13 17:32:17	VIEW	EDIT	NO	DEL
Car Park Prediction.zip	2018-06-21 16:55:28	Free Parking Lots Prediction	OK - 2018-06-21 16:55:28	VIEW	EDIT	NO	DEL
Current UV in Antwerp.zip	2019-01-13 15:38:13	Current UV in Antwerp (data coming from the openweather API)	OK - 2019-01-13 15:38:13	VIEW	EDIT	YES	DEL
Current weather in Antwerp.zip	2019-01-13 15:25:55	Current weather in Antwerp (Openweather API)	OK - 2019-01-13 15:25:55	VIEW	EDIT	YES	DEL
Events in Finland.zip	2019-01-07 17:43:47	Cultural and educational events (Frequently updated events from multiple cultural event organizers including concerts, sports events, museum exhibitions and many more.) only in Finnish	OK - 2019-01-07 17:43:47	VIEW	EDIT	YES	DEL
Finland Ceticos.zip	2019-02-13 12:33:31	Statistiche	OK - 2019-02-13 12:33:31	VIEW	EDIT	NO	DEL
Finland_ceticos_interni.zip	2019-02-12 13:00:30	Ticket Ceticos interni	OK - 2019-02-12 13:00:30	VIEW	EDIT	NO	DEL





- 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.





Tap on the hour you prefer to receive 3 notification everyday for the Bike Sharing service 16:18

00:00	01:00	02:00	03:00	04:00	05:00
06:00	07:00	08:00	09:00	10:00	11:00
12:00	13:00	14:00	15:00	16:00	17:00
18:00	19:00	20:00	21:00	22:00	23:00

Qualità dell'aria 02:22 ✓

Qualità dell'aria rilevata dal sensore più vicino alla posizione:

- Temperatura: 8.10 °C
- Umidità: 97.50%
- CO: 0.3 µg/m3
- CO2: 499.0 µg/m3
- NO: NaN µg/m3
- NO2: 56.1 µg/m3
- O3: 20.9 µg/m3
- PM10: 13.8 µg/m3
- PM2.5: 12.2 µg/m3

Public transport 16:41 ✓

Choose a bus stop: 16:42

Giorgini	Giorgini
Vittorio Emanuele	Montelatici

Giorgini - FM0256

- 17:12 - [55] → Cappuccini
- 17:29 - [55] → Cappuccini
- 17:45 - [55] → Cappuccini
- 18:01 - [55] → Cappuccini
- 18:17 - [55] → Cappuccini
- 18:33 - [55] → Cappuccini

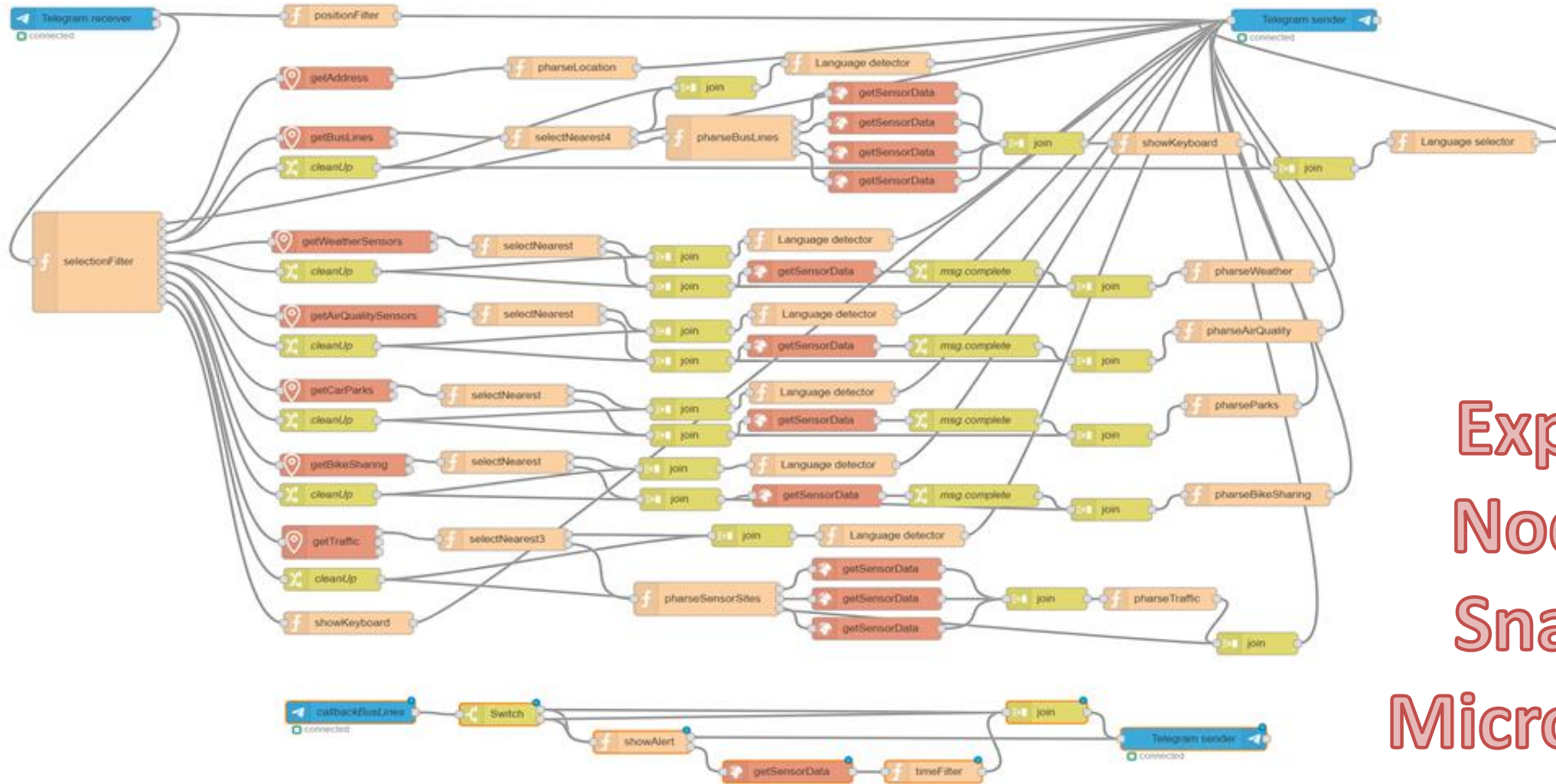
16:43

Trasporti pubblici 14:53 ✓

Ho trovato 6 linee vicino a te:

- 24 - ATAF&LINEA  
Grassina → Bagno A Ripoli Robinson
- 49 - ATAF&LINEA  
Grassina 02 → Bagno A Ripoli Robinson
- 48 - ATAF&LINEA  
Il Roseto 01 → Bagno A Ripoli Robinson

# IOT App of SnapBot: OneShot Services



Exploiting  
Node-RED  
Snap4City  
MicroServices



TOP

## Dashboards ↔ IoT App / Proc.Logic Server Side Business Logic

FORGING &  
MANAGING OPEN  
AND FLEXIBLE WEB  
AND MOBILE APPS

SNAP4CITY FOR  
BEGINNERS

SNAP4CITY

TWITTER  
VIGILANCE: SOCIAL  
MEDIA ANALYSIS

SNAP4CITY  
AND KM4CITY  
PROJECTS

DATA GATHERING  
AND CITY DATA  
KNOWLEDGE  
MANAGEMENT

HOW TO ADOPT  
SNAP4CITY, AND  
OUR ROADMAP

MISSION SUPPORT  
TEAM AND CITY  
SEQUENCE

SNAP4CITY THE  
VIEW OF THE  
ADMINISTRATORS

IOT APPLICATIONS,  
THE LOGIC AND  
THE SMARTNESS

SMART CITY API,  
MICROSERVICES,  
SNAP4CITY API

SNAP4CITY  
LIVING LAB FOR  
COLLABORATIVE  
WORK

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

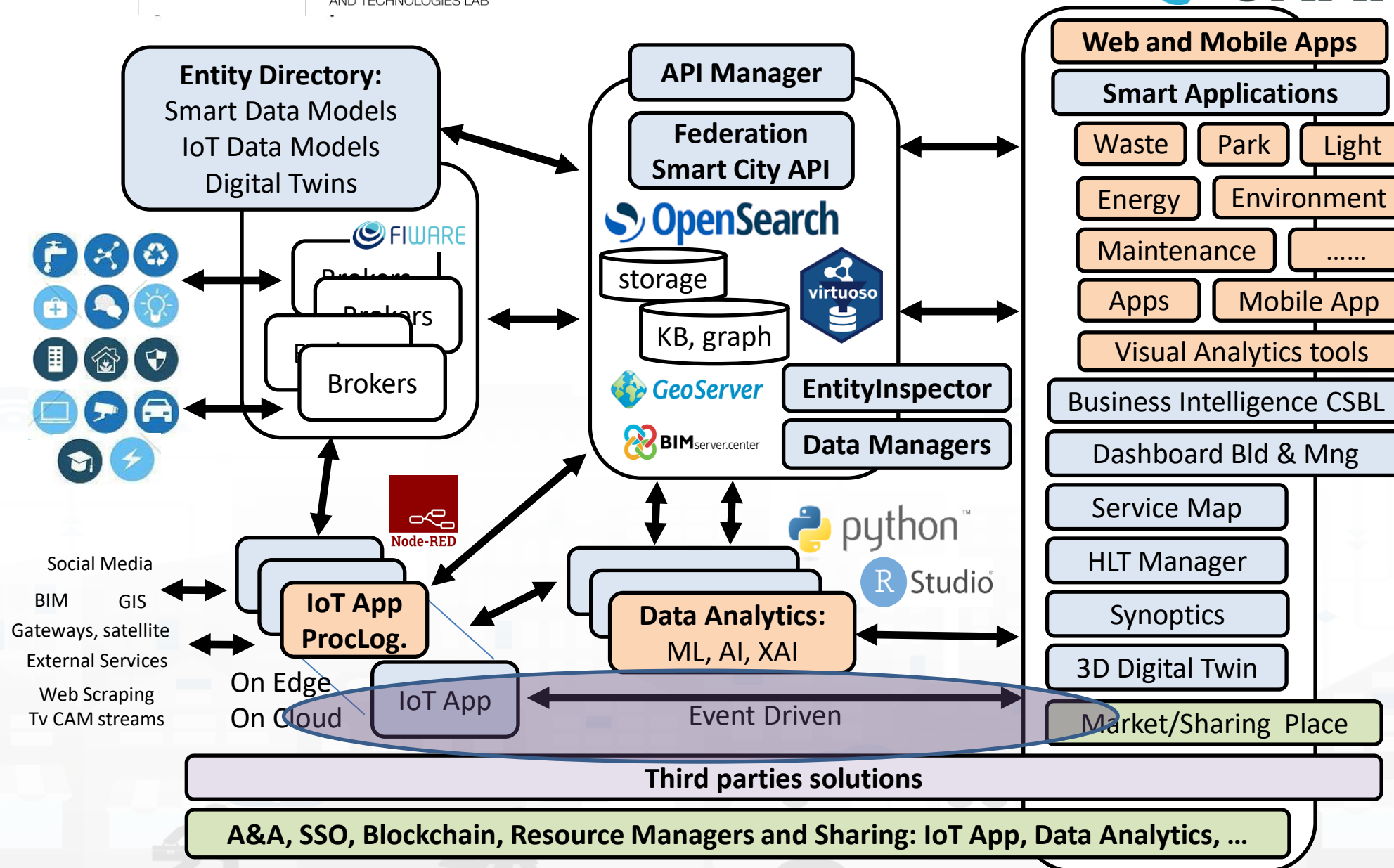
DOC: Processing Logic/IoT App

How to Develop Proc.Logic / IoT App

Create A MicroService from RestCall

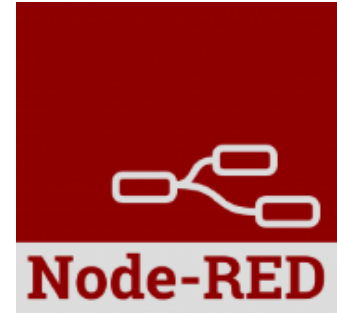
Node-RED

# Tech Arch



# IoT App / Proc.Logic

- Storage  $\rightarrow$  IoT App / Proc.Logic
- External Service  $\leftrightarrow$  IoT App / Proc.Logic
- **Dashboards  $\leftrightarrow$  IoT App / Proc.Logic**



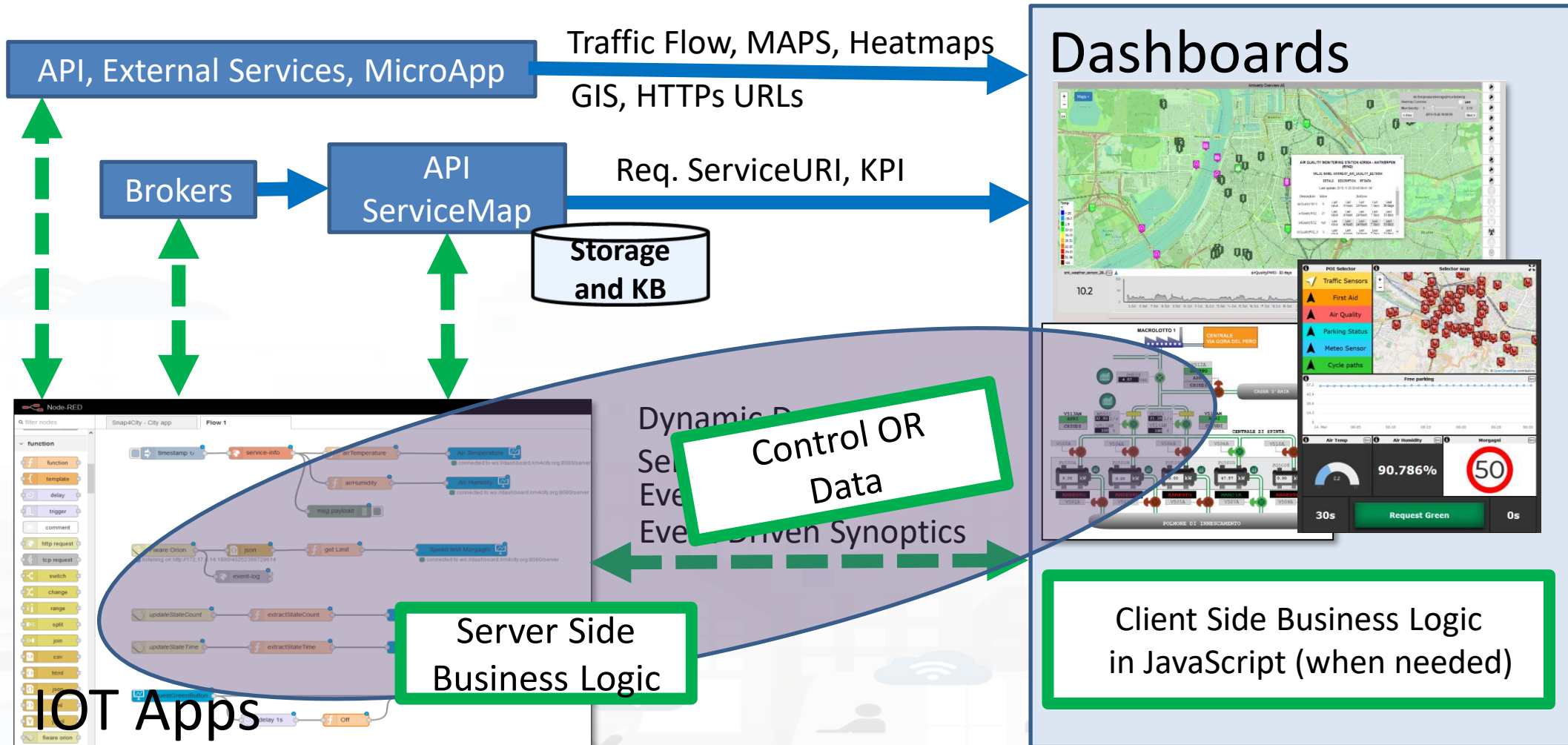
- Data Analytics  $\leftrightarrow$  IoT App / Proc.Logic
- Broker  $\rightarrow$  Storage
- IoT App / Proc.Logic  $\rightarrow$  Broker
- Broker  $\rightarrow$  IoT App / Proc.Logic
- IoT App / Proc.Logic  $\rightarrow$  Storage

Part 4

Part 5



# How the Dashboards exchange data



# Snap4City Dashboards main concepts



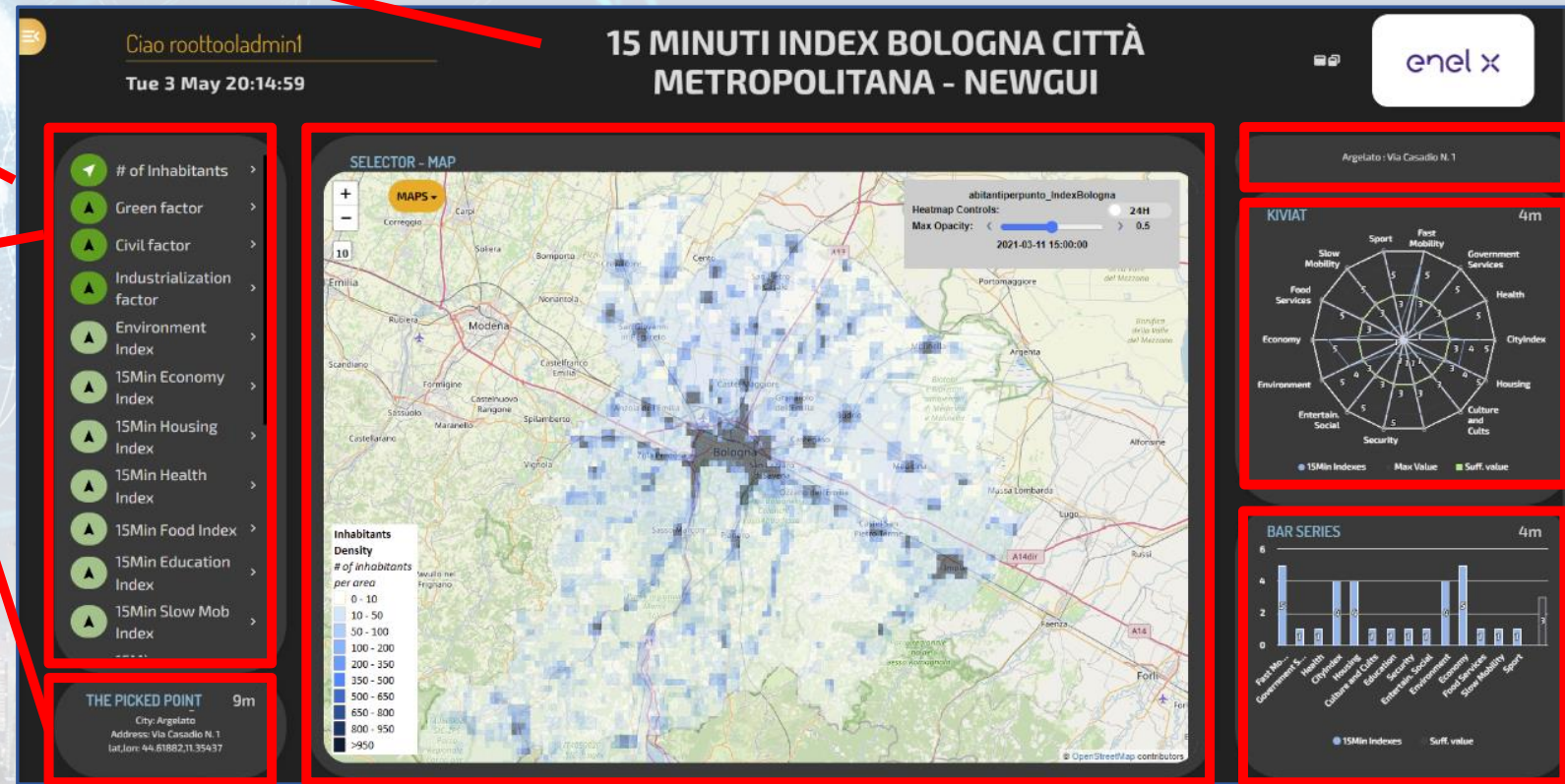
Header

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**



▼ S4CDashboard

- coordinates from map
- impulse button
- numeric keyboard
- switch button
- dimmer
- geolocator
- dropdown
- 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
- event driven my kpi
- synoptic read
- synoptic write
- synoptic subscribe
- form

## Native Local

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

- Local on IOT Edge

## or

## Snap4City

- 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

▼ dashboard

- button
- dropdown
- switch
- slider
- numeric
- text input
- date picker
- colour picker
- form
- text
- gauge
- chart
- audio out
- notification
- ui control
- template

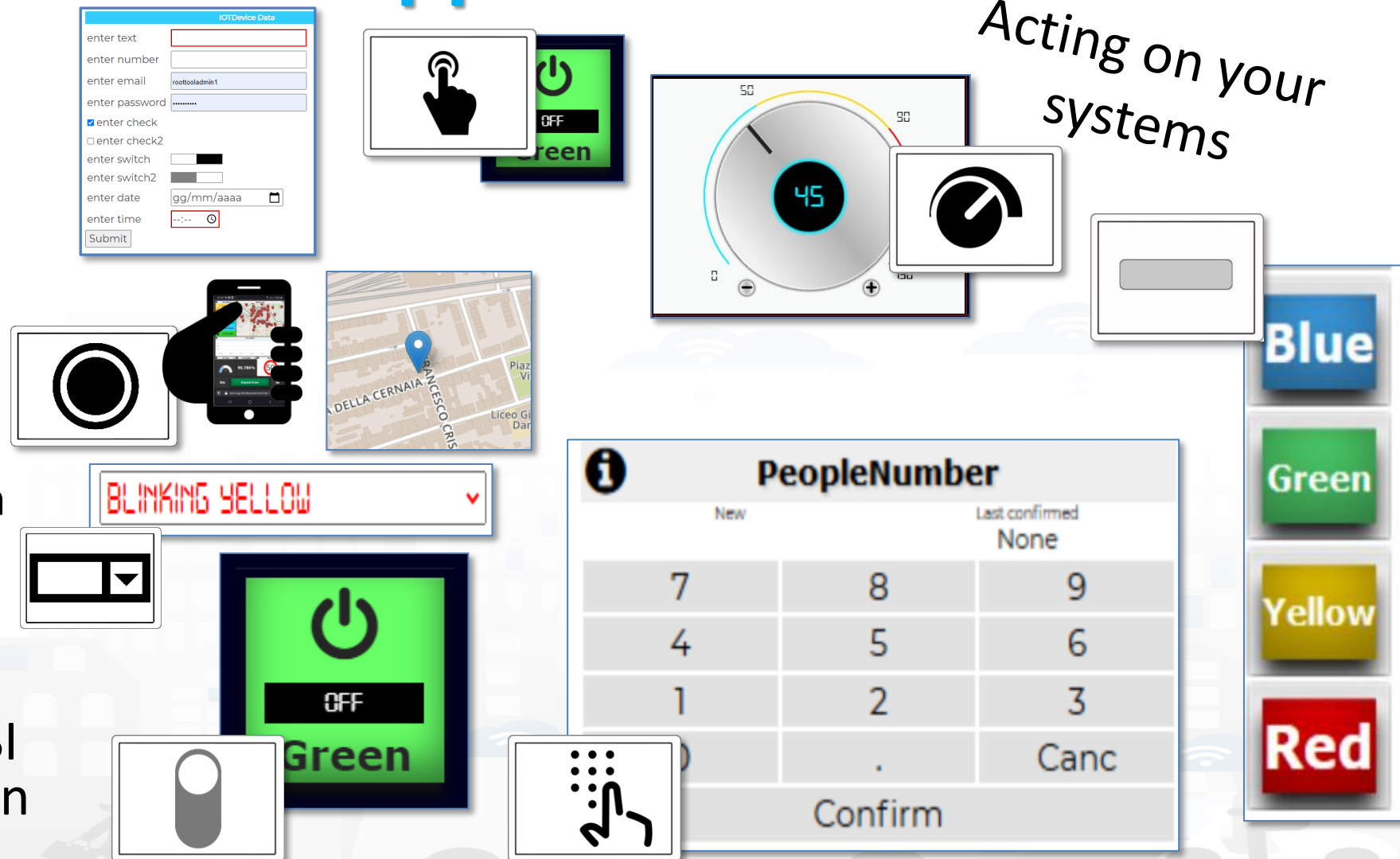


# From Dashboard to IoT App

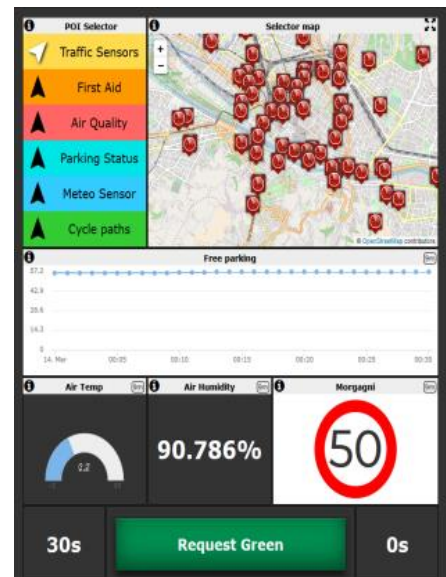
- **Widgets:**

- Impulse Button
- Button
- Switch
- Dimer/Knowb
- KeyPad
- Geolocator
- Selection/Dropdown
- Form
- Map Picking

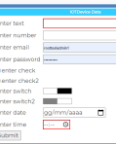
- **Registered** on some IOT brokers with NGSI mutual authentication



# Dashboard-IoT App

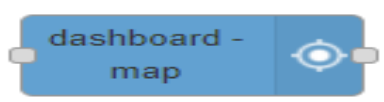


PeopleNumber		
time	Last confirmed	
7	8	9
4	5	6
1	2	3
0	.	Cancel
Confirm		

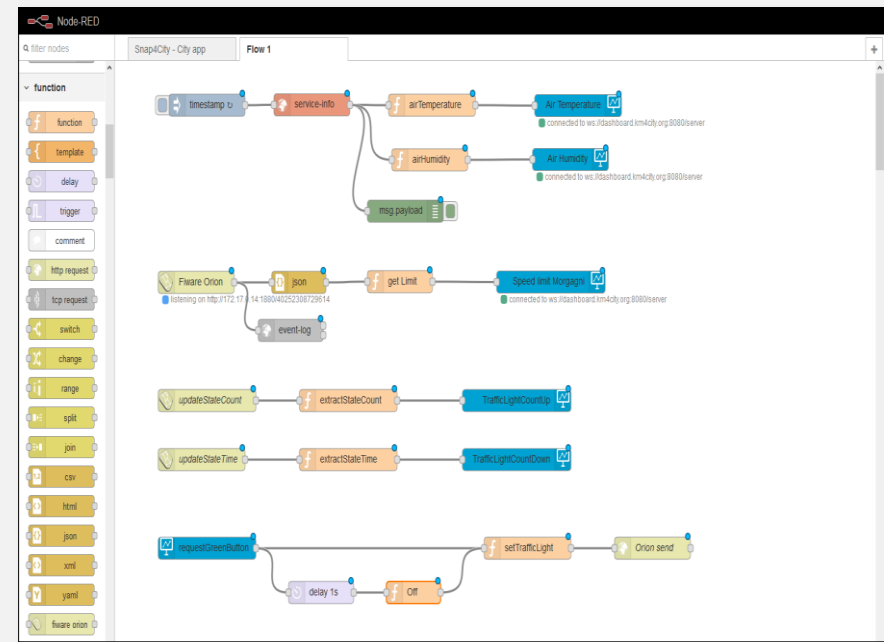


MapClick  
MyKPI variable onchange  
Synoptics

- impulse button
- numeric keyboard
- switch button
- dimmer
- geolocator
- dropdown
- form
- coordinates from map
- event driven my kpi
- synoptic read
- synoptic subscribe



## From Dashboard to IOT App



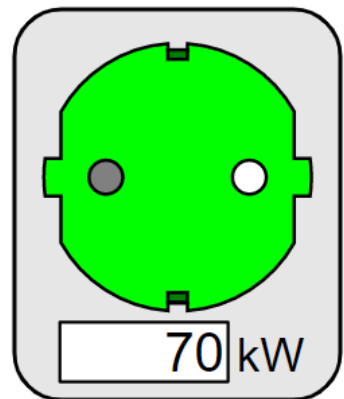
## IOT Application



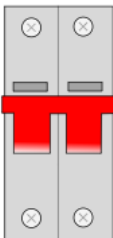
# SVG Custom Widgets Examples 2

Tue 17 Nov 18:46:47

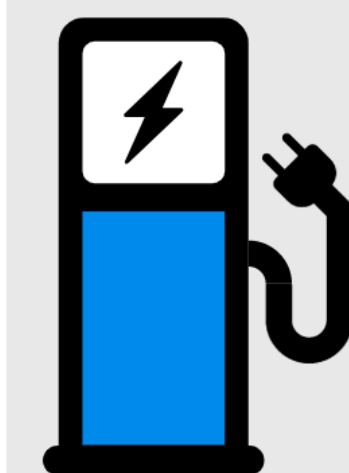
SVG shucko plug



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 status set

YELLOW LIGHT

Traffic 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 Explanation

### Speed Limit Custom Widget example

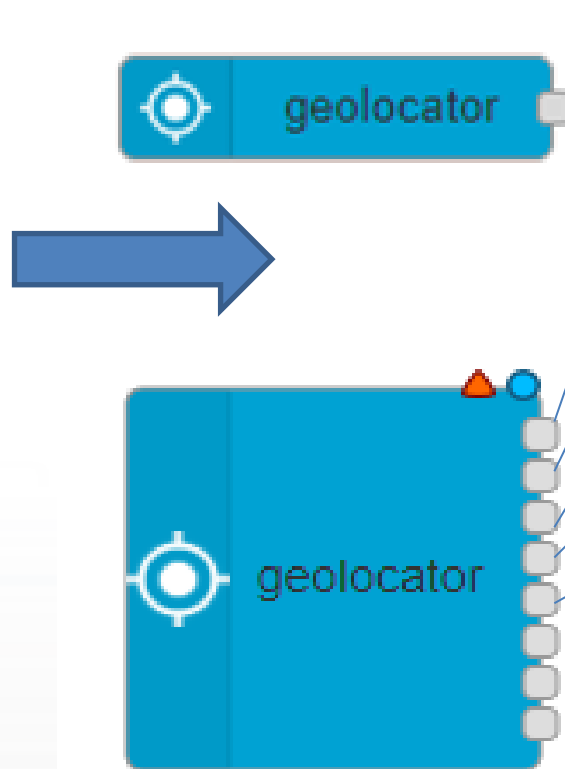
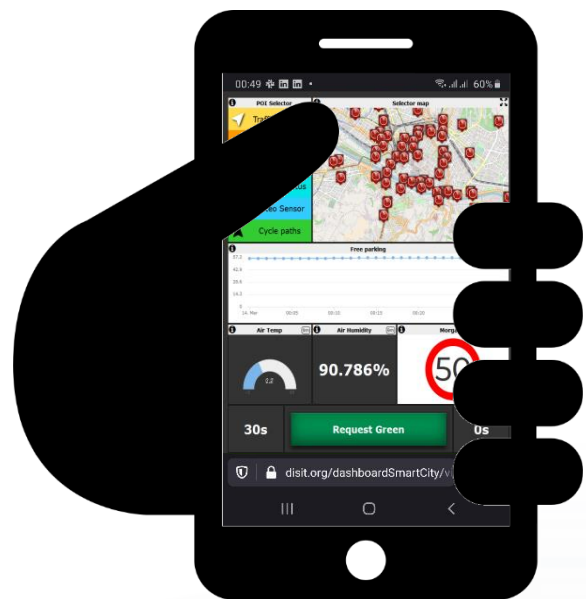
Write the speed limit by using the keypad and click CONFIRM.

9999 = white sign.

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



# Geolocation of Mobile Device



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

- 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

# Dashboard-IOT App

## From IoT App to Dashboard

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

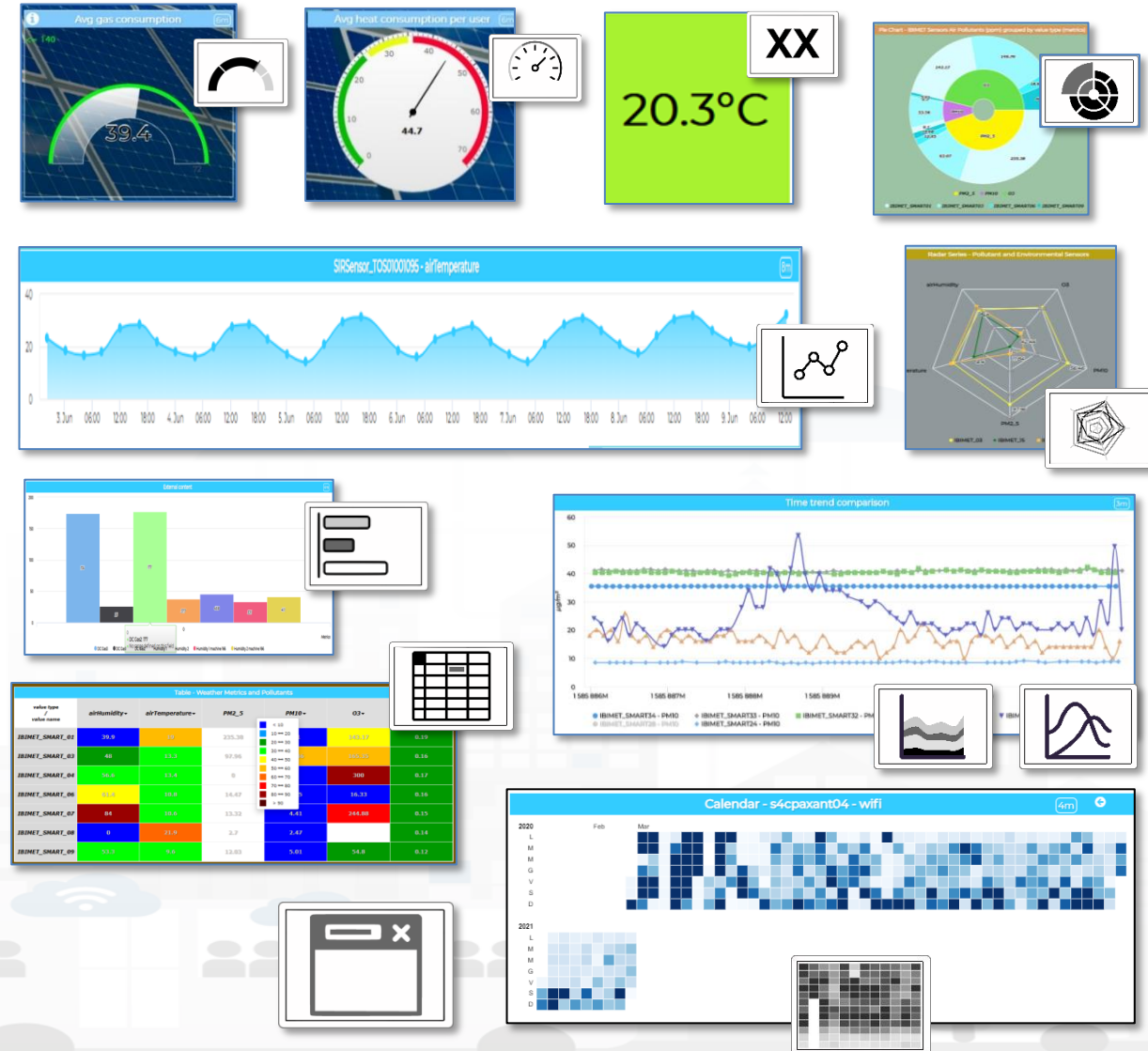
Snap4D3

dashboard - map

event table

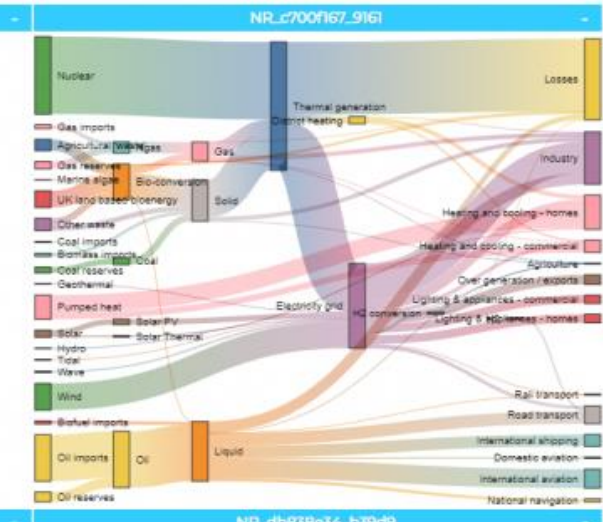
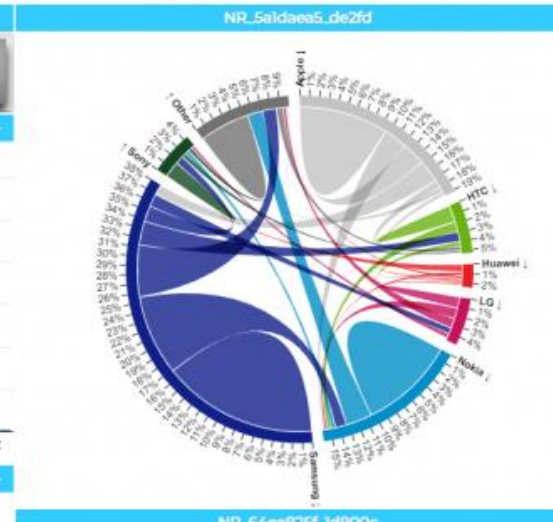
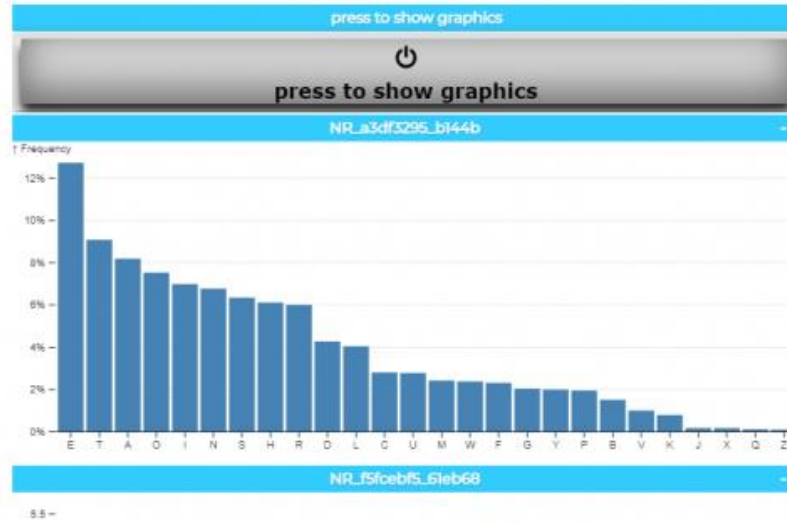
device table

## IOT Application

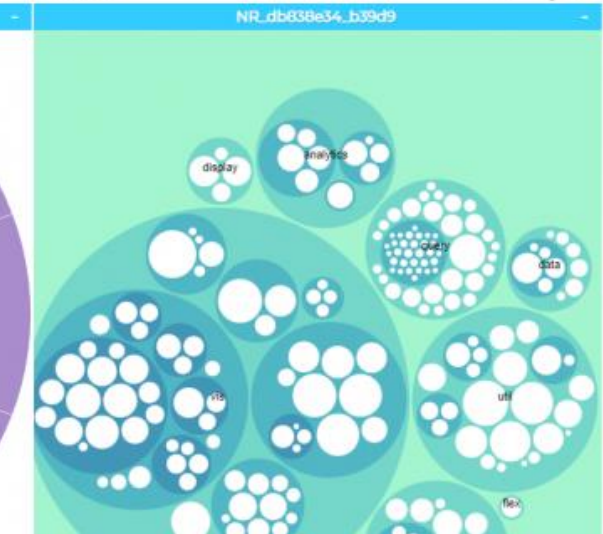
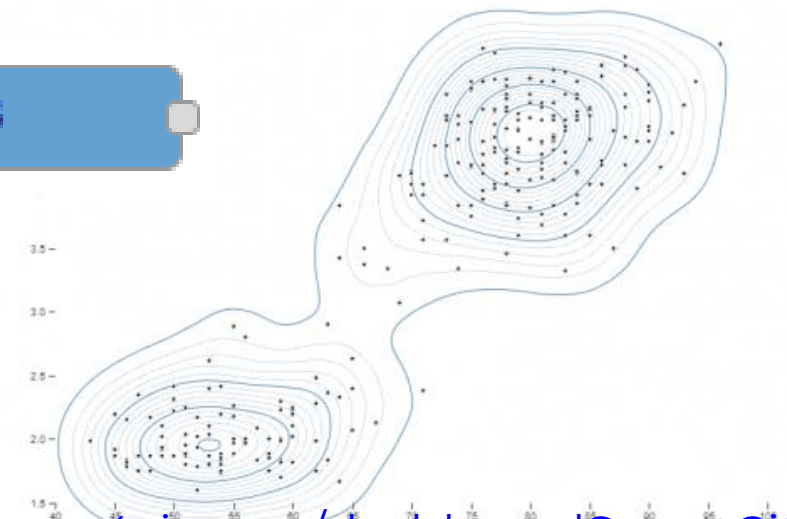


## D3 library Example

Fri 10 Jun 19:46:06



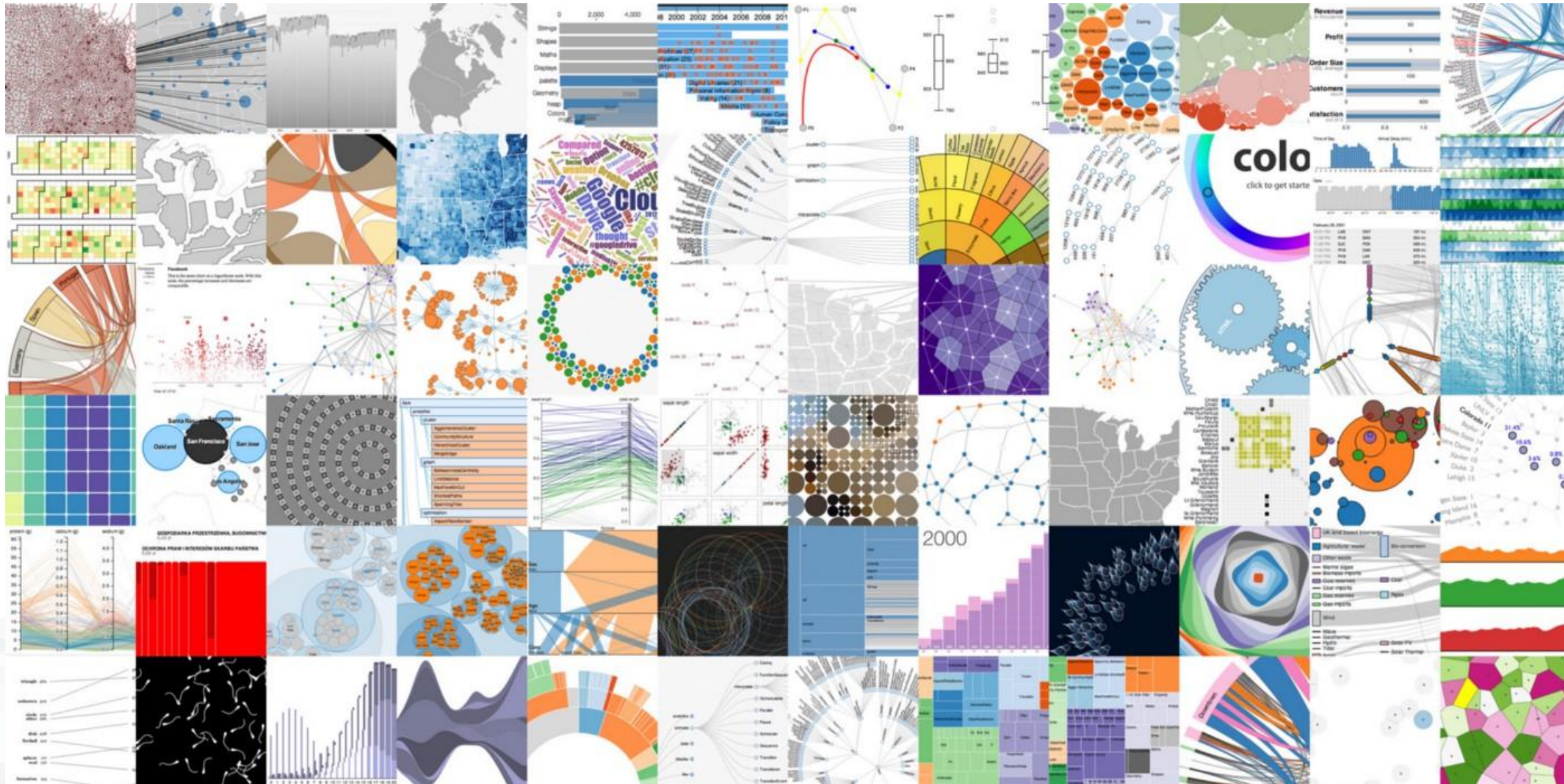
Snap4D3



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



# D3.js graphs



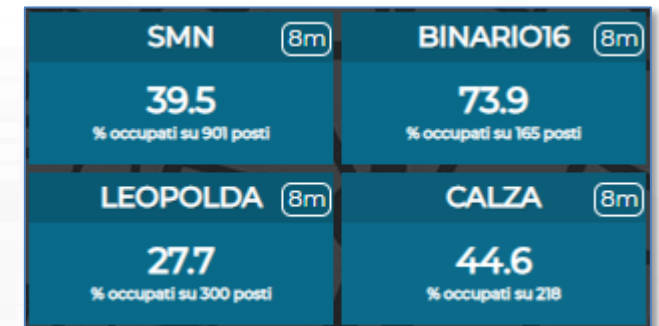
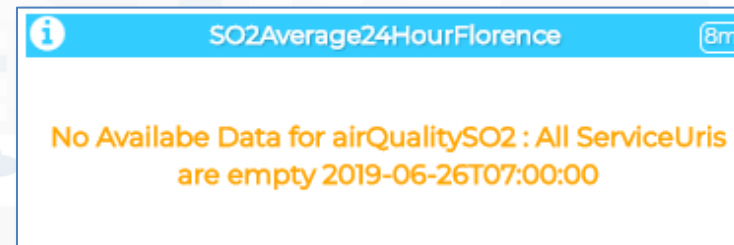
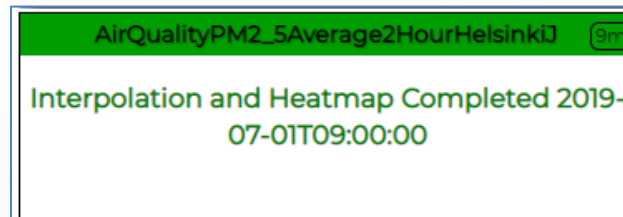
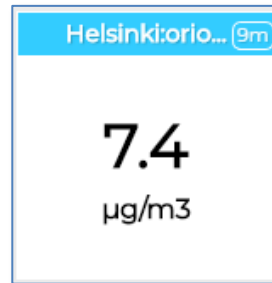
# Single Content Widget (flexibility)

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]
}
```

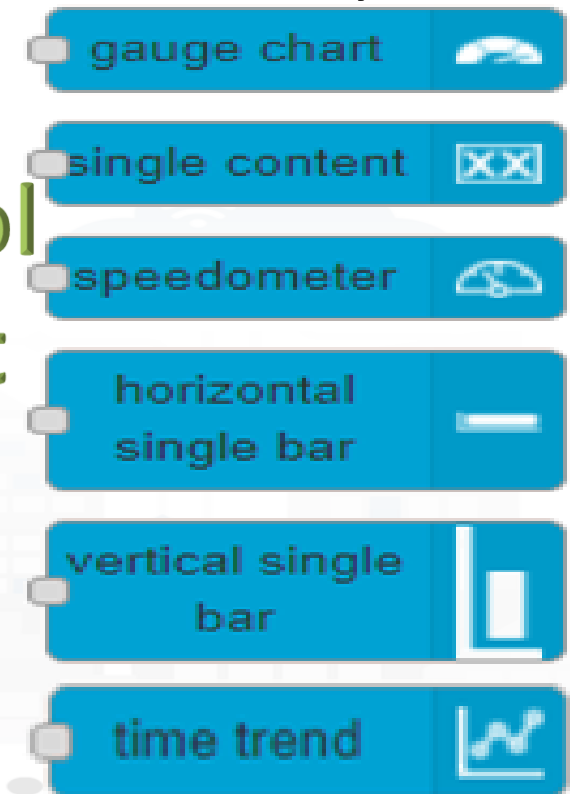
## Two examples:

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

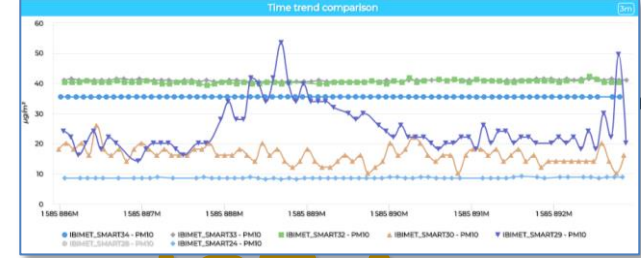
{
  "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
  "metricHighLevelType": "Sensor",
  "metricName": "tusc_weather_sensor_ow_3166540",
  "metricType": "airTemperature"
}
```

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]
  }
]
```



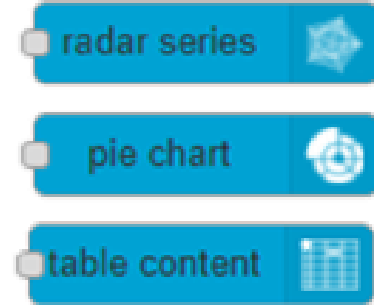
```
[
  {
    "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
    "metricHighLevelType": "Sensor",
    "metricName": "tusc_weather_sensor_ow_3166540",
    "metricType": "airTemperature"
  },
  {
    "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3182522",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



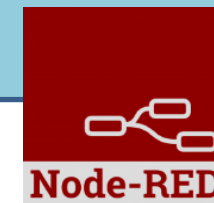
```
[
  {"metricId":"17056320",
    "metricHighLevelType":"MyKPI",
    "metricName":"SiiMTuscanyTrackerLocation",
    "metricType":"Velocity"},
  {"metricId":"http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
    "metricHighLevelType":"Sensor",
    "metricName":"tusc_weather_sensor_ow_3166540",
    "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}
]
```

TOP

# Device Tables vs IoT App data

## Getting data from Dashboards

- 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



<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 category, 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 ",



# 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
<div>+</div> CarParkStazioneFirenzeS.M.N.	877	2022-09-01T11:33:01.681Z	<div>📍</div>
<div>+</div> CarParkStazioneFortezzaFiera	530	2022-09-01T11:33:01.681Z	<div>📍</div>
<div>+</div> CarParkS.Ambrogio	379	2022-09-01T11:33:01.681Z	<div>📍</div>
<div>+</div> CarParkAlberti	313	2022-09-01T11:33:01.681Z	<div>📍</div>
<div>+</div> CarParkPieracciniMeyer		2022-09-01T11:33:01.681Z	<div>📍</div>

DT1

43s

Show

5

▼

Search:

First

<< Prev

1

2

3

Next >>

Last

device	capacity	dateObserved	Actions
<div>-</div> CarParkStazioneFortezzaFiera	530	2022-09-01T11:33:01.681Z	<div>📍</div>

freeParkingLots 277

occupancy

occupiedParkingLots 253

status

<div>+</div> CarParkStazioneFirenzeS.M.N.	877	2022-09-01T11:33:01.681Z	<div>📍</div>
<div>+</div> CarParkS.Ambrogio	379	2022-09-01T11:33:01.681Z	<div>📍</div>
<div>+</div> CarParkParterre	1006	2022-09-01T11:33:01.681Z	<div>📍</div>
<div>+</div> CarParkCareggi	514	2022-09-01T11:33:01.681Z	<div>📍</div>

- 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



```
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	capacity	dateObserved	Actions
CarParkStazioneFortezzaFiera	530	2022-09-01T11:33:01.681Z	
freeParkingLots 277			
occupancy			
occupiedParkingLots 253			
status			
CarParkStazioneFirenzeS.M.N.	877	2022-09-01T11:33:01.681Z	
CarParkS.Ambrogio	379	2022-09-01T11:33:01.681Z	

- **Output:**
  - The Action clicked by the user with the name of the SURI and ID

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



# *Widgets MAP with business intelligence on IoT Apps*

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



# Maps Business Logic vs IOT Apps

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

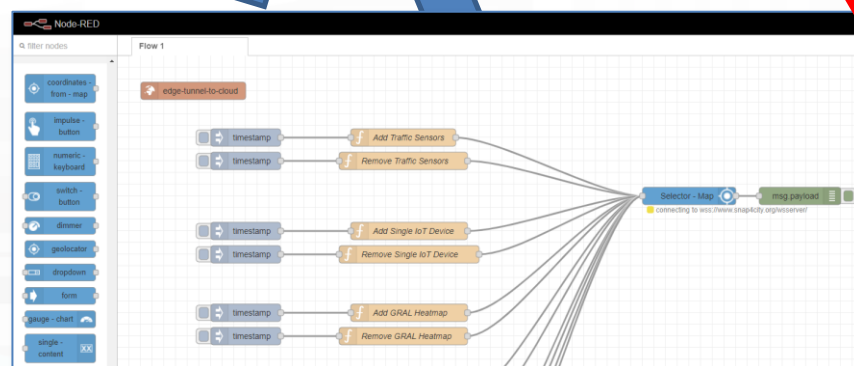
data

Selector

Data, changed data  
References, commands,  
selectives, selections



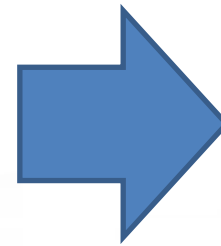
data



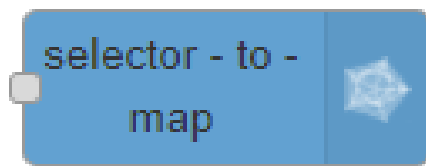
selections, positions, ServiceURI

# IoT App, Node-RED nodes: IoT App vs Dashboards

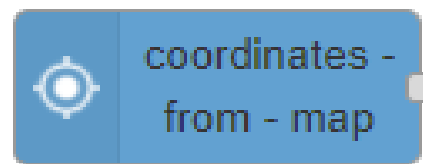
- **Coordinates From Map:**
  - Get GPS click from Map
  - Get Selected ServiceURI from Map
- **SelectorMap:**
  - Send commands, references, data to Maps



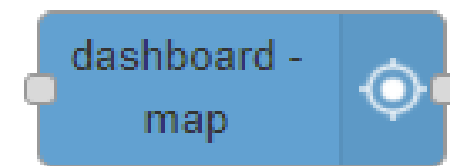
- **Dashboard Map provides:**
  - **All Features of**
    - Coordinate From Map:
    - SelectorMap:



+



= =

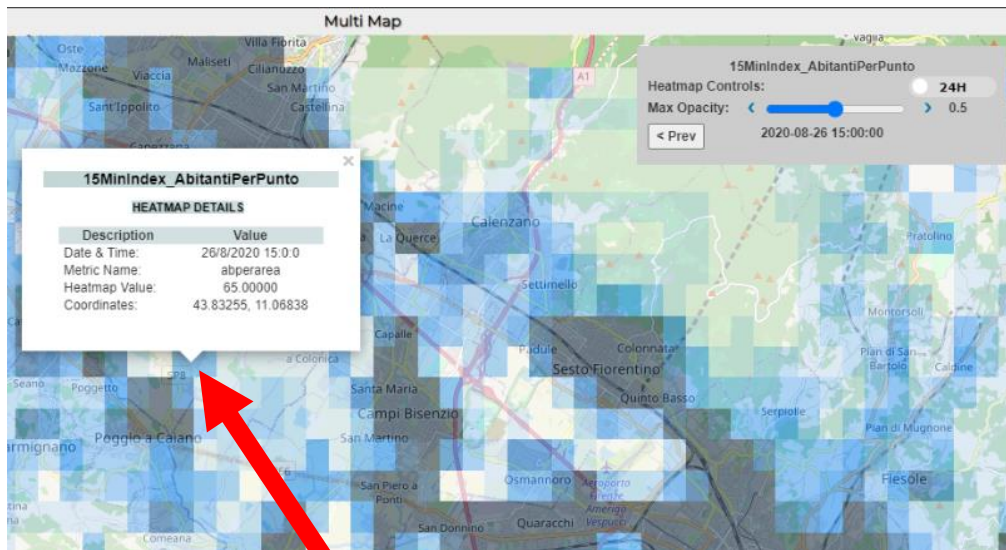


*SelectorMap and Coordinates From Map will be Deprecated from Snap4City Library in late 2022*

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



# Multi Data Map GPS Location Picking vs IOT App



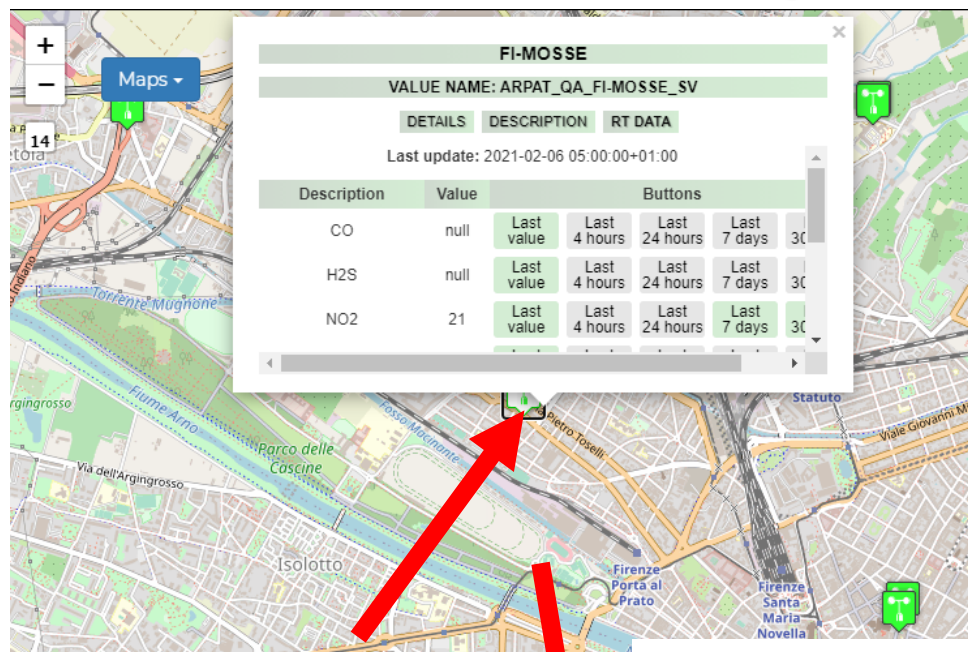
1) Click

2) GET event with:

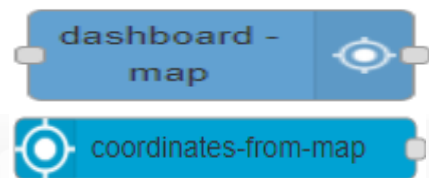
- Lat,Long
- ServiceURI

- 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:

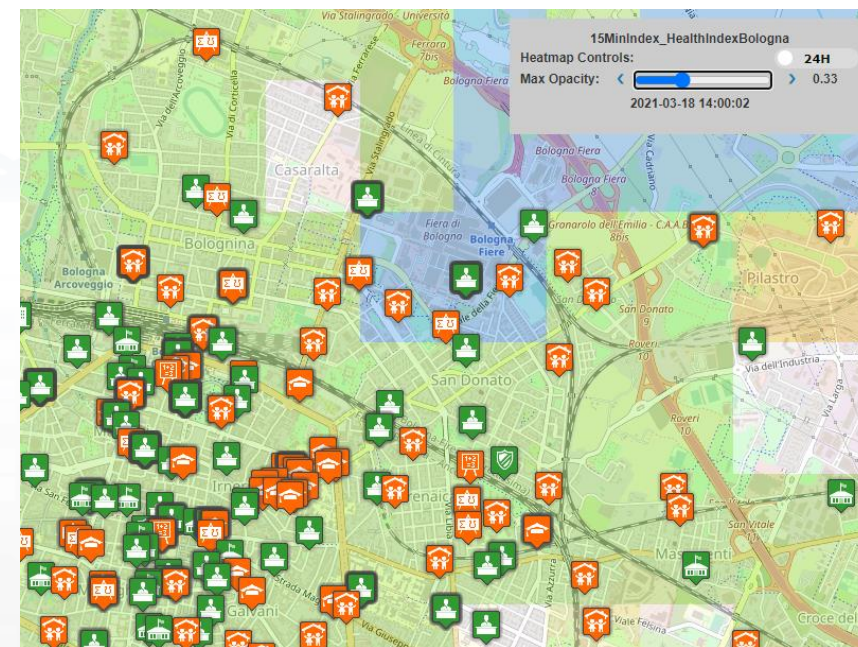
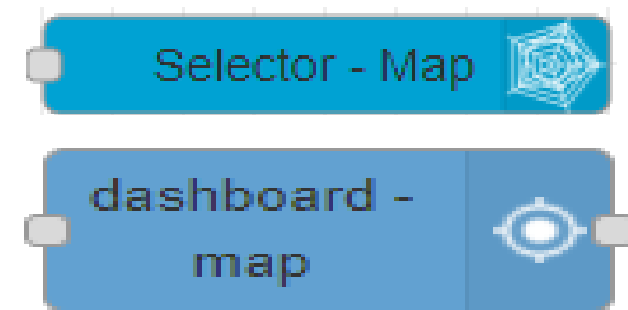
- Lat,Long
- **ServiceURI**

- 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.

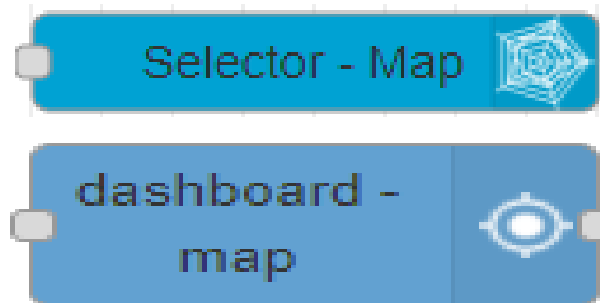


## Controlling Maps from IOT Apps

- 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

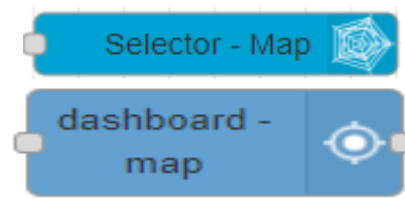






```
msg.payload = {
  "type": "addSelectorPin",
  "target": "w_Map_956_widgetMap32131",
  "passedData": {
    "desc": "my studio temperature - GP -",
    "query": "datamanager/api/v1/poidata/17055853",
    "color1": "#ffdb4d",
    "color2": "#fff5cc",
    "display": "pins",
    "queryType": "MyPOI",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

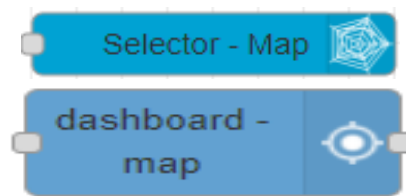
**Add MyKPI**  
**Add MyPOI**



```
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  
}
```



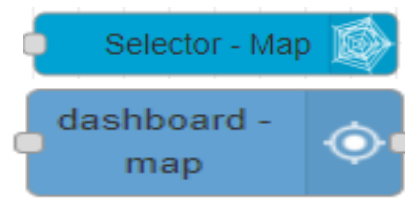
```
payload : {  
  "type": "addOD",  
  "target": "w_Map_956_widgetMap32131",  
  "passedData" : "https://odmm.snap4city.org/api/get?precision=communes&from_date=2017-10-19%2000:00:00&organization=Tuscany&inflow=True&longitude=11.257123947143556&latitude=43.771837562821375",  
  "passedParams": {  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

## Add OD Matrix

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

## Remove OD Matrix





```
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"  }, ... ]
}
```

## Add GeoInfoPin set of

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

## Remove GeoInfoPin set of

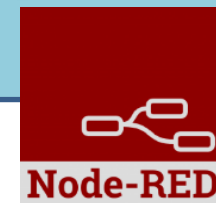


TOP

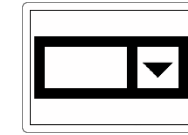
# Selector and Forms vs IOT App data

## Getting data from Dashboards

- 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



# DropDown Selector



Dashboard  
Name SVG Custom Widgets Examples 2

Widget Name Traffic Light status set

OFF	0	x
RED LIGHT	1	x
RED and YELLOW LIGHT	2	x
YELLOW LIGHT	3	x
YELLOW and GREEN LIK	4	x
GREEN LIGHT	5	x

+ add

Edit Dashboard View Dashboard

Traffic Light status set

RED LIGHT

Traffic Light status set

RED LIGHT

OFF

RED LIGHT

RED AND YELLOW LIGHTS

YELLOW LIGHT

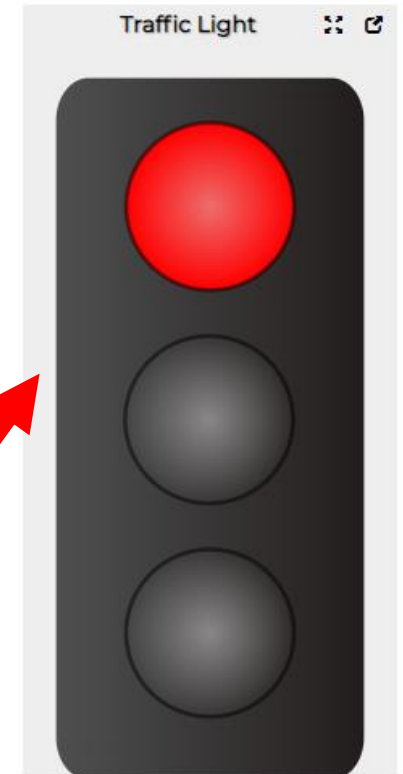
YELLOW AND GREEN LIGHTS

GREEN LIGHT

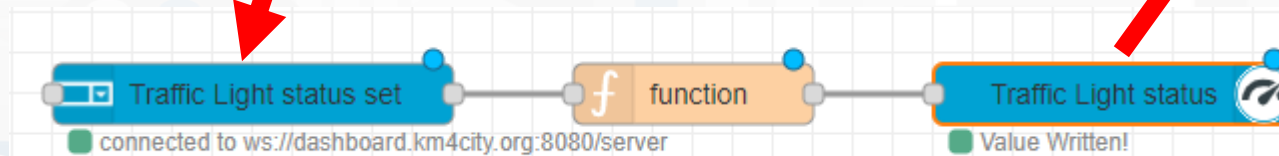
RED, YELLOW AND RED LIGHTS

BLINKING YELLOW

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



HTML



Traffic Light status set

RED LIGHT

```

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



IoTDevice Data

enter text

enter number

enter email

enter password

☒ enter check

☐ enter check2

enter switch ☐

enter switch2 ☐

enter date

enter time

# 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



IOTDevice Data

enter text	a text
enter number	123
enter email	paolo.nesi@unifi.it
enter password	.....
<input checked="" type="checkbox"/> enter check	
<input type="checkbox"/> enter check2	
enter switch	<div style="width: 50%; background-color: black; height: 10px;"></div>
enter switch2	<div style="width: 20%; background-color: #888; height: 10px;"></div>
enter date	19/03/2021
enter time	09:38
<div style="border: 1px solid #ccc; padding: 5px 15px; display: inline-block;">Submit</div>	

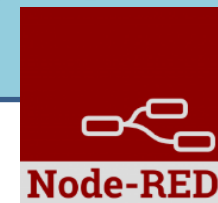
"selected": ["a text", "123", "paolo.nesi@unifi.it", "aaaaaa", "checked", "", "on", "", "2021-03-19", "09:38"]

TOP

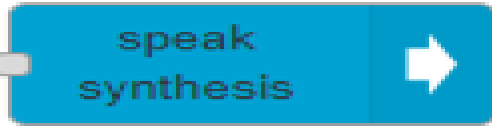
# *Talk to your users*

## *Producing voice on Dashboards*

- 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







# 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
  - ..

**Edit speak-synthesis node**

Delete Cancel Done

**Properties**

Authentication: Add new snap4city-authentication...

text: insert text

rate: insert rate. 1 is the default

pitch: insert pitch. 1 is the default

lang:

Dashboard Name: New Dashboard

Widget Name:

Edit Dashboard View Dashboard

You must have an account with Snap4city to use this node. You can register for one [here](#).

**help**

Search help

**speak-synthesis**

With this node you can send a voice message to an existing dashboard or a new one created by the node.

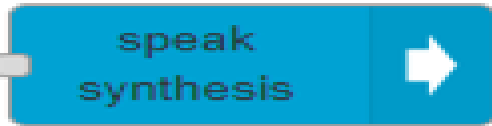
Configuration

**text** string  
Text of the message to be sent

**rate** string  
Speed of execution. 1 is the default value

**pitch** string  
Running tone. 1 is the default value

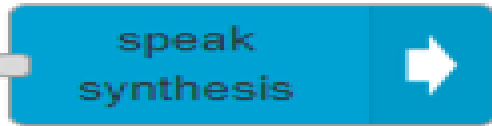
**lang** string  
String to select language and voice. The possible choices are:  
'engF' for English woman  
'engM' for English man  
'itF' for Italian woman  
'itM' for Italian man  
'frF' for French woman  
'frM' for French man  
'esF' for Spanish woman  
'esM' for Spanish man



# 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
  - ..





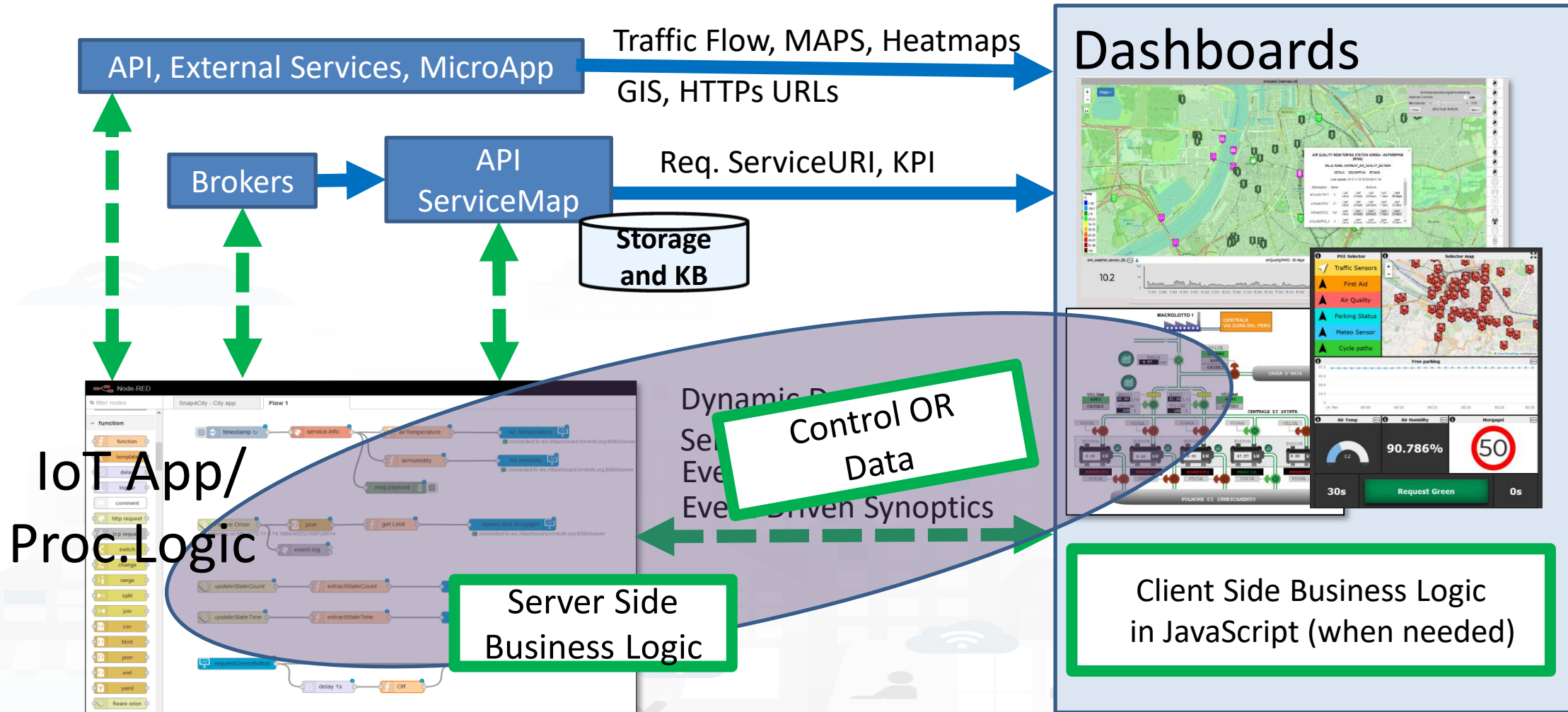
TOP

# Dynamic Widgets data on Dashboard from IOT Applications

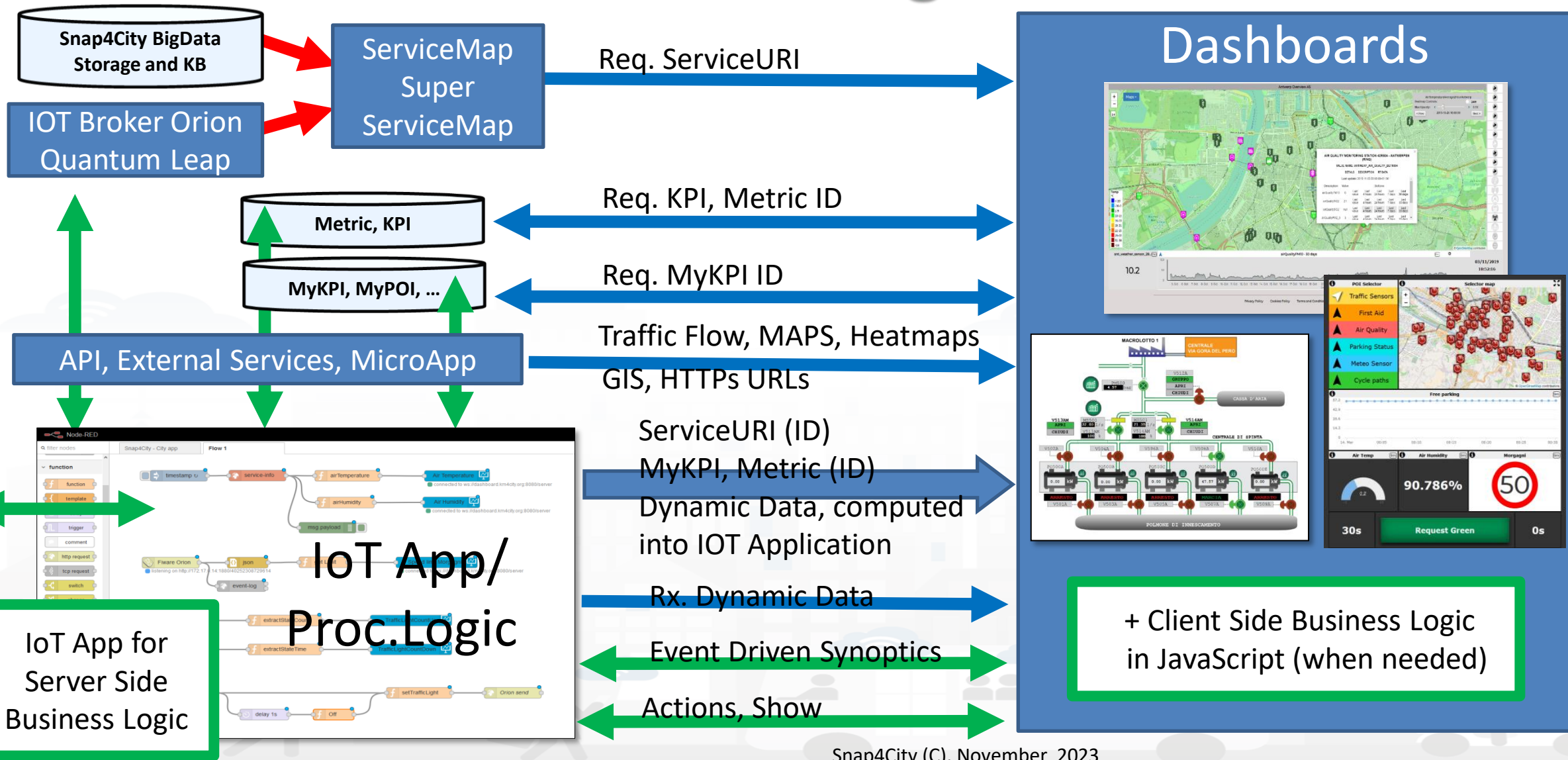
- Processing Logics / IOT App ▲
- Processing Logics / IOT App
- ↑ MicroServices for Proc.Logic/IoT App
- ↑ MicroServices from DataAnalytic
- ☐ IOT MicroServices for Final Users
- ☐ IOT MicroServices for Developers
- ☐ DOC: Processing Logic/IoT App
- ☐ How to Develop Proc.Logic / IoT App
- ☐ Create A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



# How the Dashboards exchange data



# How the Dashboards exchange data





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

DD: Event Driven

- **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)

TOP

# Demo IoT Application exploiting Snap4City Dashboard

- 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



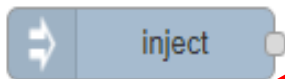


# Example of complex IOT Application

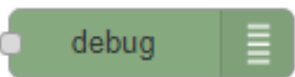
In this demo let's create an IoT Application that:

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

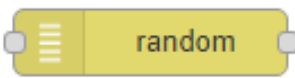
# Nodes for flow



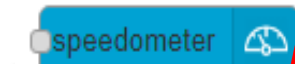
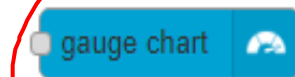
Generates an input for the other nodes. It can be repeated at predefined intervals, entered manually and of various types (timestamp, string, number, boolean, JSONetc)



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

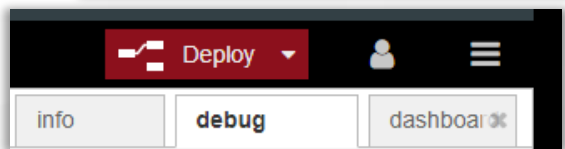


- Connect

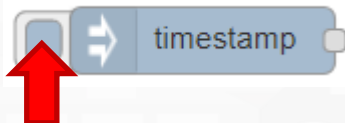
- Configure

☒ Payload timestamp  
  
☒ Repeat interval  
 every  minutes  
☒ Inject once at start?

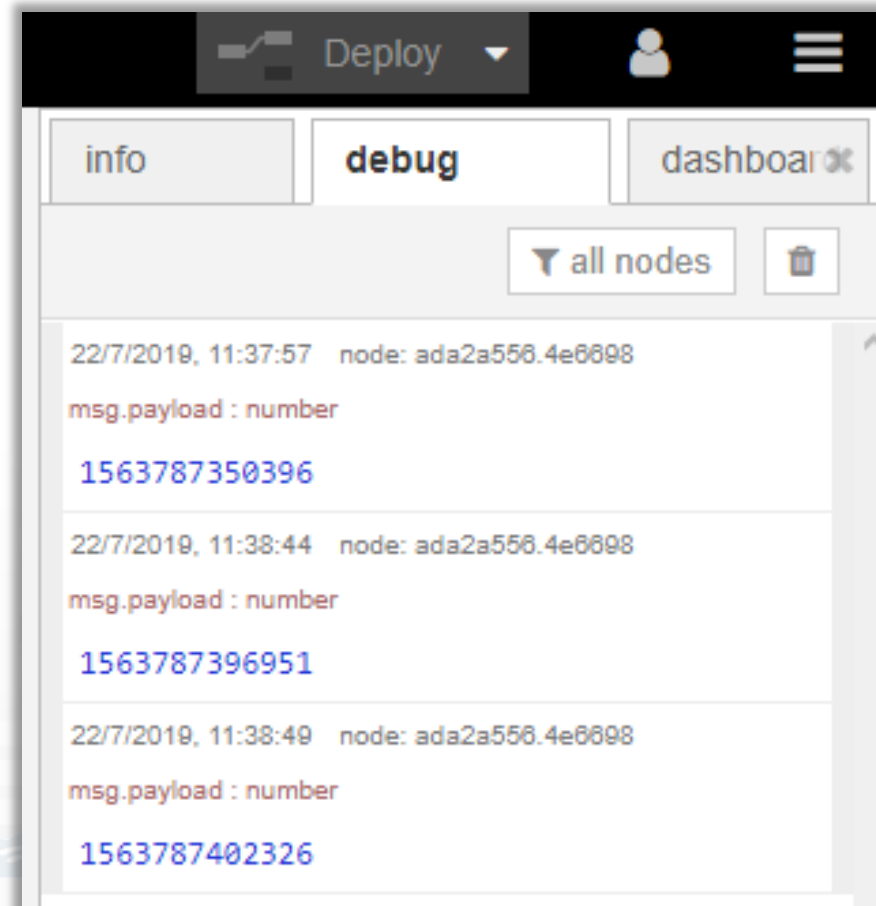
- Deploy



- Click

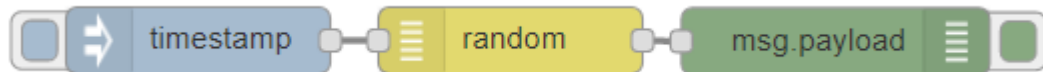


- Observe





# Step 2



- Random
- Connect
- Configure

random

msg.payload

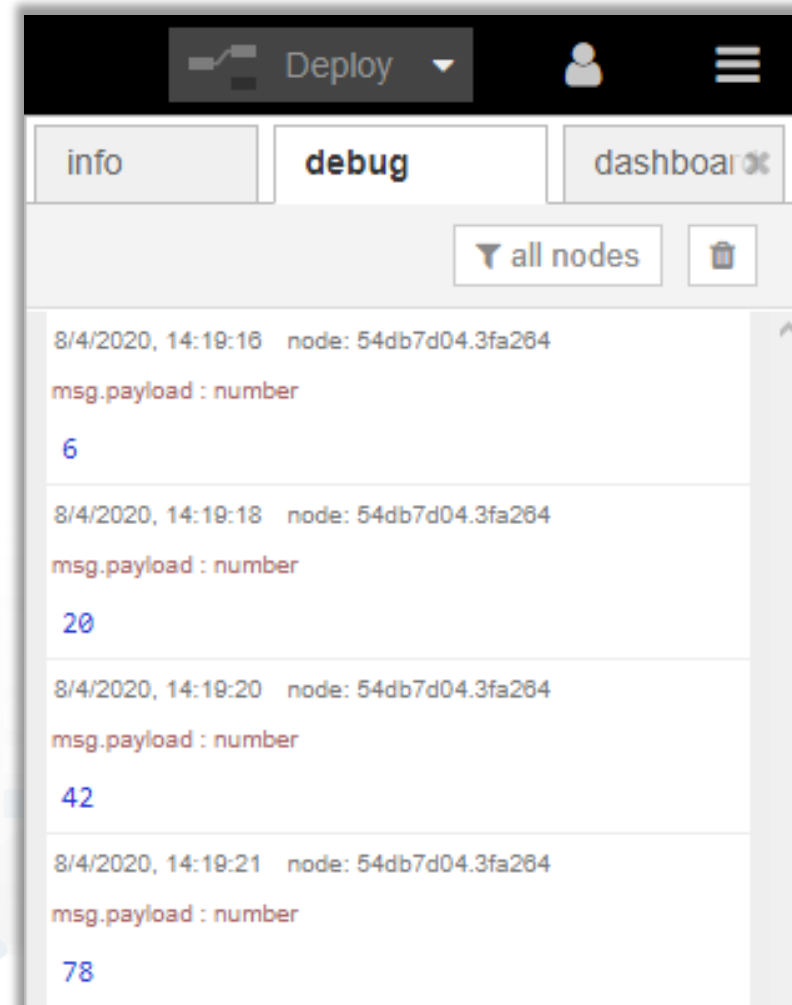
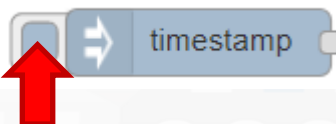
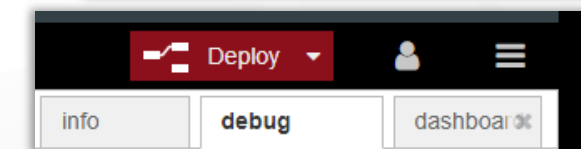
Generate: a whole number - integer

From: 1

To: 100

Name: Name

- Deploy
- Click
- Observe



# Step 3

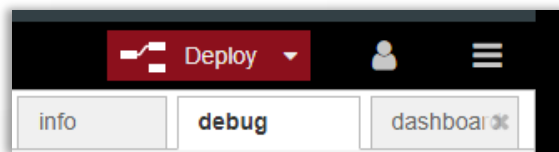


- Single content

- Connect

- Configure

- Deploy



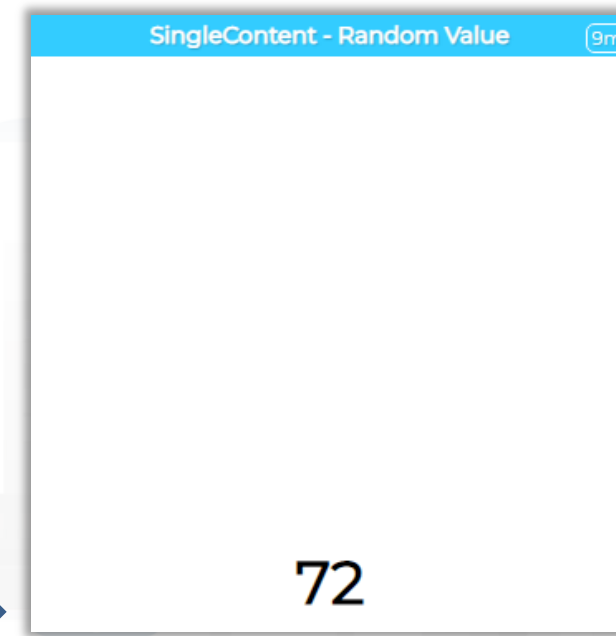
- Click



- Click



- Observe



# Nodes configuration

**inject**

**Payload** ▼ timestamp

**Topic**

**Repeat** interval

every 15 minutes

☒ Inject once at start?

**debug**

**Output** ▼ msg. payload

**to** debug tab

**Name** Name

**gauge chart**

**single content**

**speedometer**

**time trend**

**Dashboard Name** DemoTrainingCourse2020 **Create New**

**Widget Name** SingleContent - Random Value

**Edit Dashboard** **View Dashboard**

**random**

**msg. payload**

**Generate** a whole number - integer

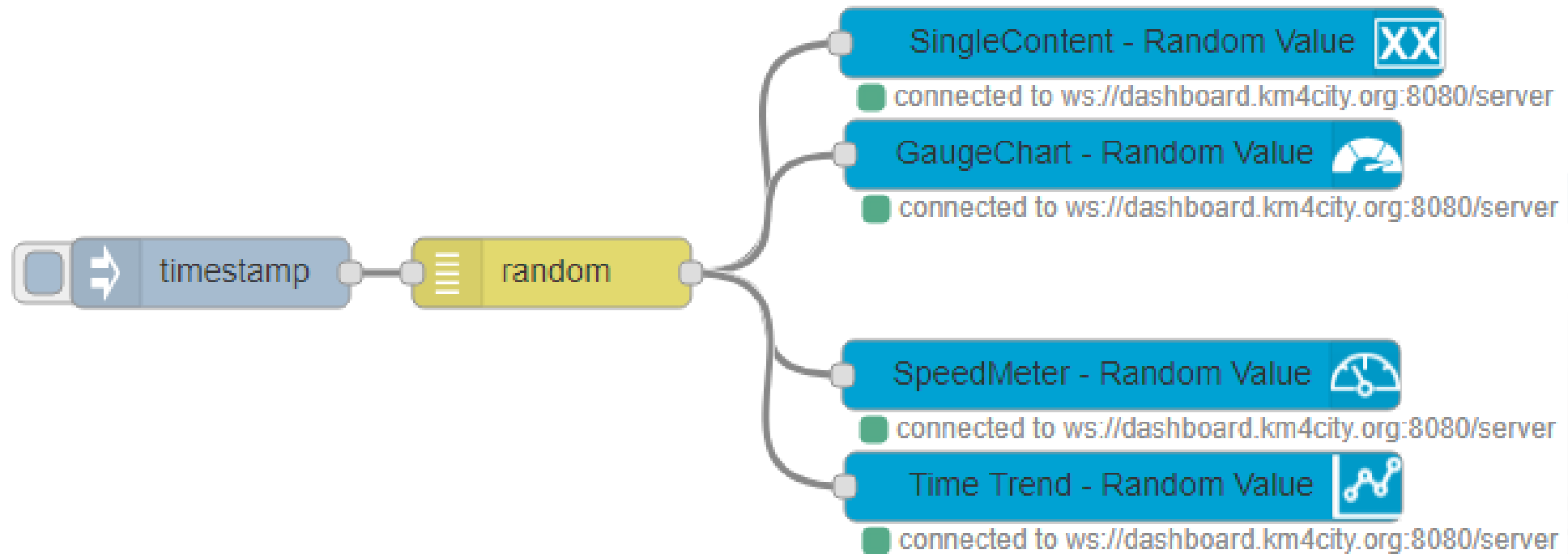
**From** 1

**To** 100

**Name** Name



# Nodes connections

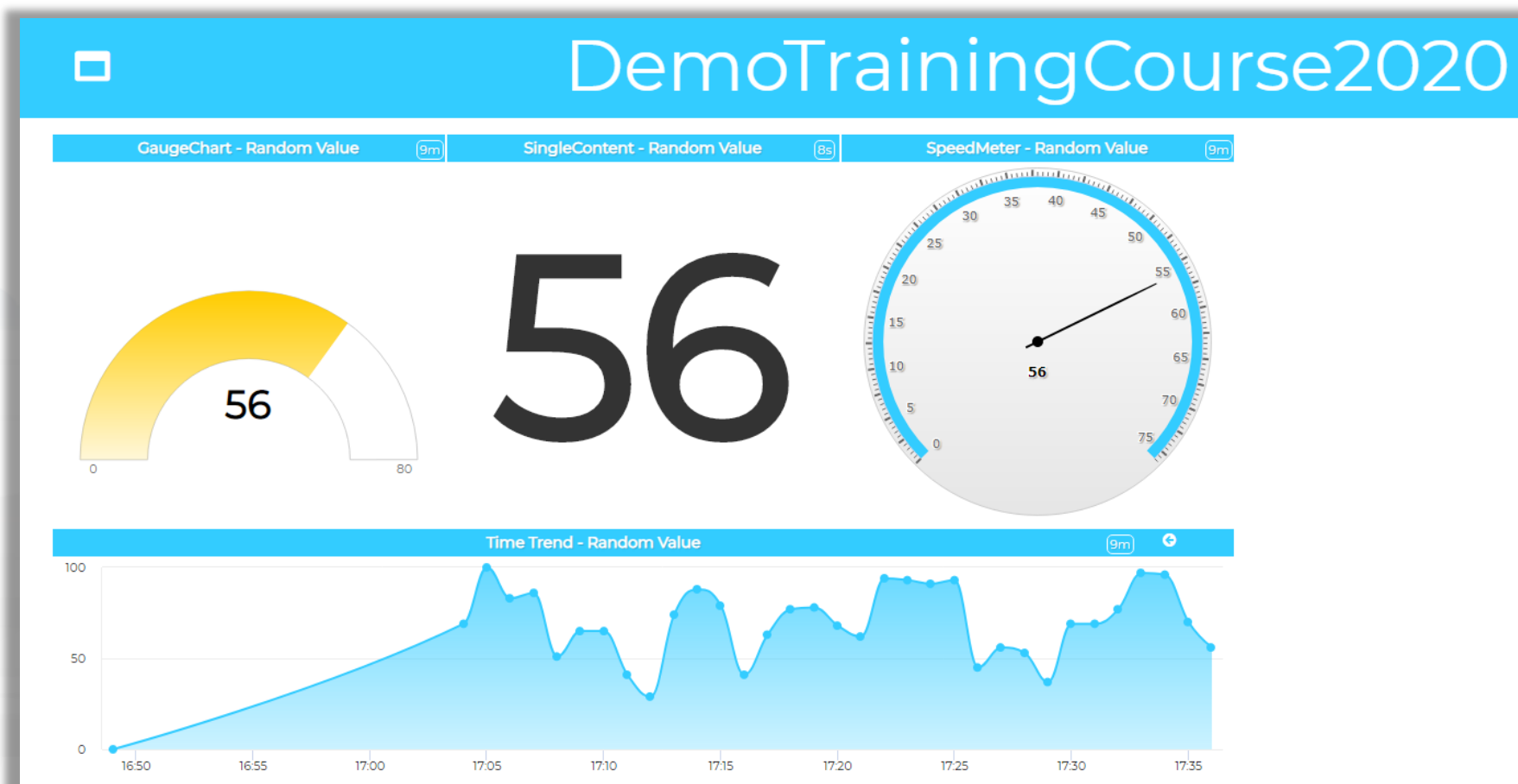


# Explaining: IOT Application Flow

- 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.

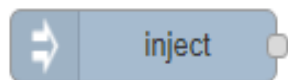


# Resulting Dashboard

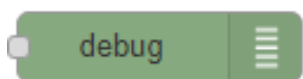




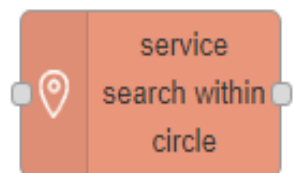
# Nodes for flow



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)

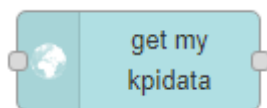


Each message that enters the debug node is shown in the "debug" tab on the right of nodered (you can choose which part of the message to show)



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

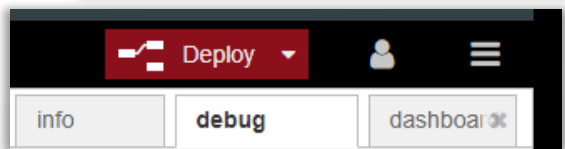


- Connect

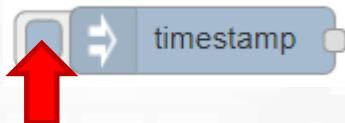
- Configure

☒ Payload timestamp  
  
☒ Repeat interval  
 every  minutes  
☒ Inject once at start?

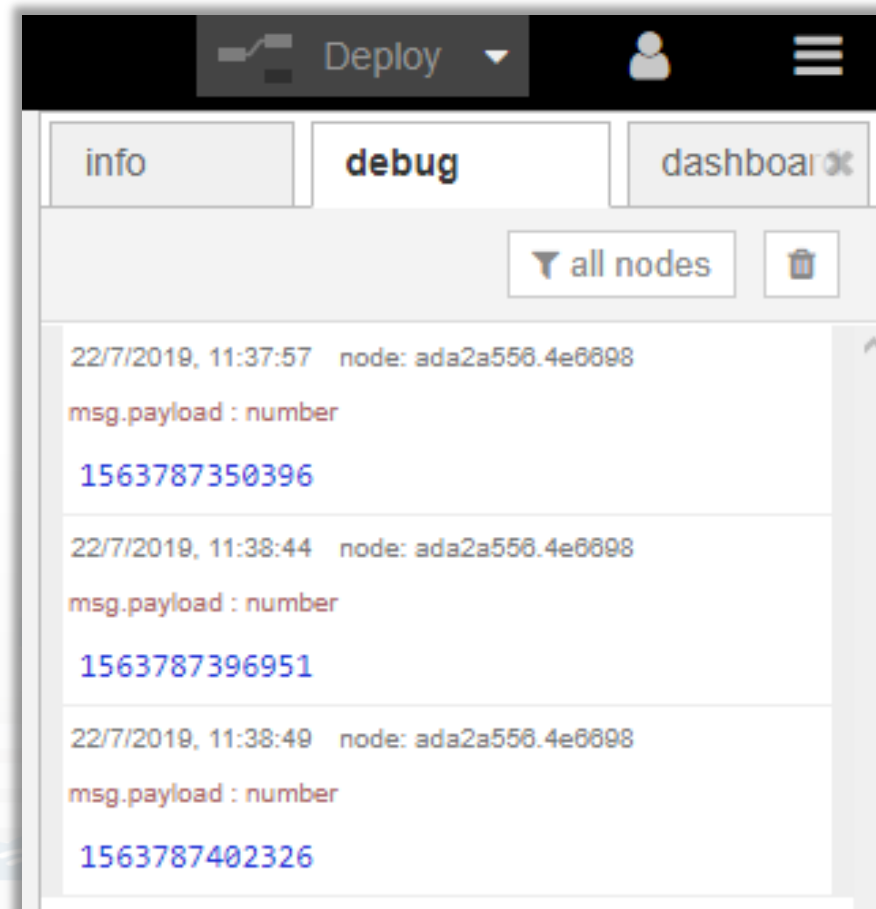
- Deploy



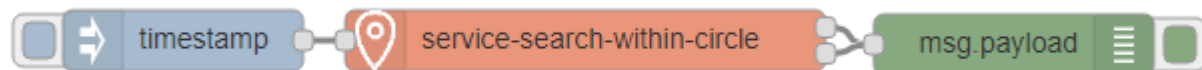
- Click



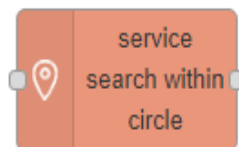
- Observe



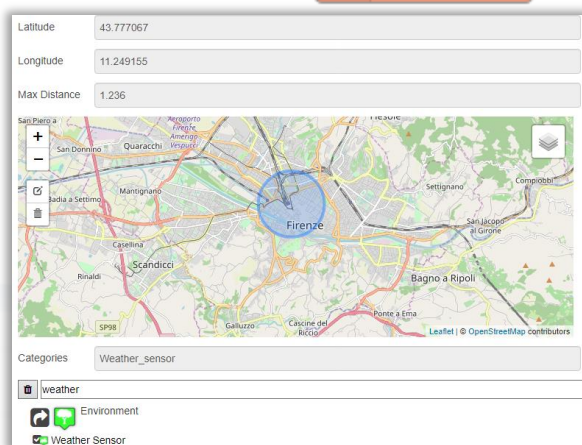
# Step 2



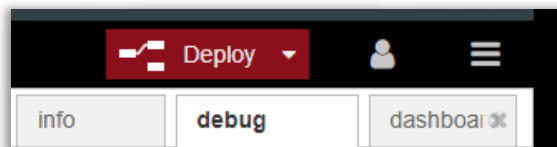
- Service Search Within Circle



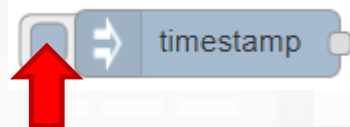
- Connect
- Configure



- Deploy



- Click



- Observe



Copy the path

Copy the value





# Step 1 Bis



- Inject and Debug

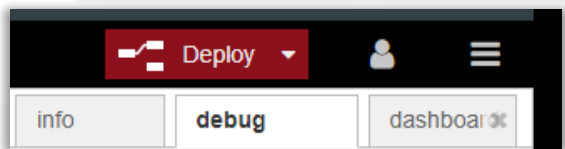


- Connect

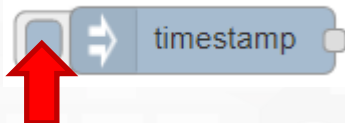
- Configure

☒ Payload timestamp  
  
☒ Repeat interval  
 every  minutes  
☒ Inject once at start?

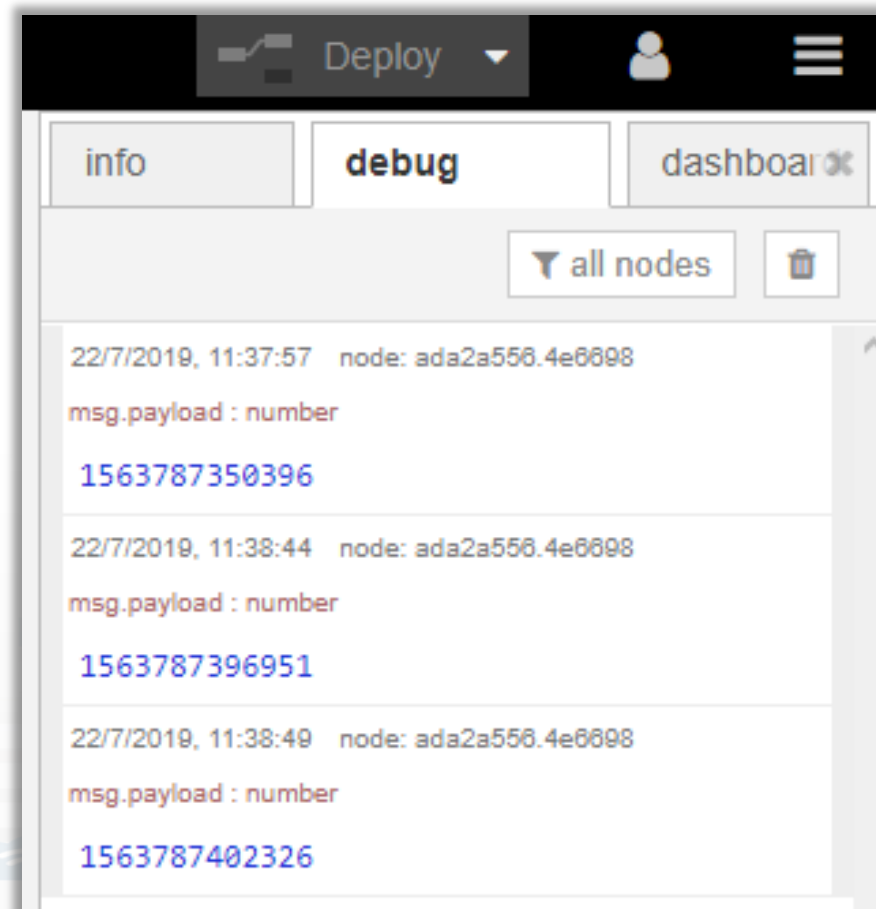
- Deploy



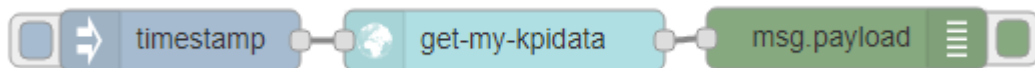
- Click



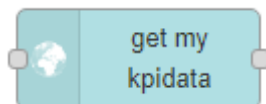
- Observe



# Step 2 Bis

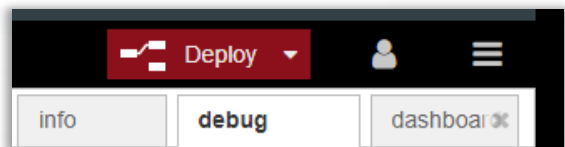


- Get My KPIData

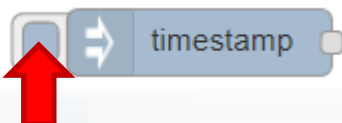


- Connect

- Deploy



- Click



- Observe



```

id: 17057458
description: ""
healthiness: "false"
highLevelType: "MyKPI"
info: ""
insertTime: 1586359858000
valueName: "Room 1"
lastDate: 1586359858000
lastValue: "0.054644625420117166"
latitude: ""
longitude: ""
valueType: "Temperature"
valueUnit: "°C"
nature: "Environment"
organizations: "[ou=DISIT,dc=ldap,dc=disit,dc=unifi]"
ownership: "private"
subNature: "Weather_sensor"
dataType: "float"
username: "badii"
  
```

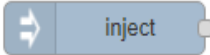
▼ [40 ... 40]

```

40: object
  id: 17057459
  description: ""
  healthiness: "false"
  highLevelType: "MyKPI"
  info: ""
  insertTime: 1586359858000
  valueName: "Room 2"
  lastDate: 1586359858000
  lastValue: "1.3839476707239307"
  
```

## Step 3



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

```
1 [
2   {
3     "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3166540",
4     "metricHighLevelType": "Sensor",
5     "metricName": "tusc_weather_sensor_ow_3166540",
6     "metricType": "airTemperature"
7   },
8   {
9     "metricId": "http://www.disit.org/km4city/resource/tusc_weather_sensor_ow_3182522",
10    "metricHighLevelType": "Sensor",
11    "metricName": "tusc_weather_sensor_ow_3182522",
12    "metricType": "airTemperature"
13  },
14  {
15    "metricId": "17057458",
16    "metricHighLevelType": "MyKPI",
17    "metricName": "Room 1",
18    "metricType": "Temperature"
19  },
20  {
21    "metricId": "17057459",
22    "metricHighLevelType": "MyKPI",
23    "metricName": "Room 2",
24    "metricType": "Room Temperature"
25  }
26 ]
```



# Step 4



- Bar Series



- Connect

Dashboard

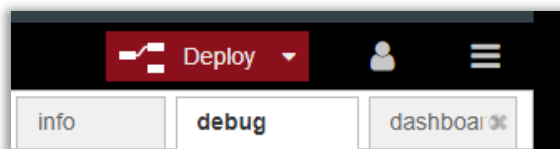
Name: DemoTrainingCourse2020 Create New

Widget Name: Bar Series

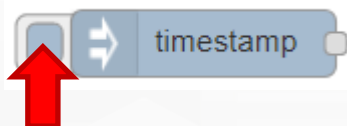
Edit Dashboard View Dashboard

- Configure

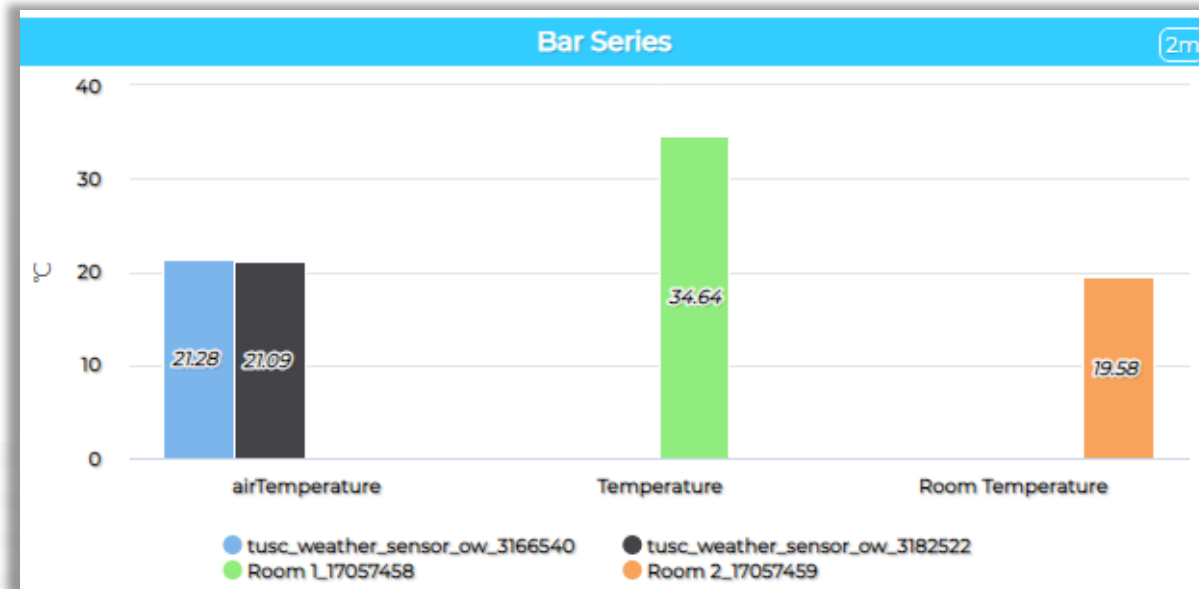
- Deploy



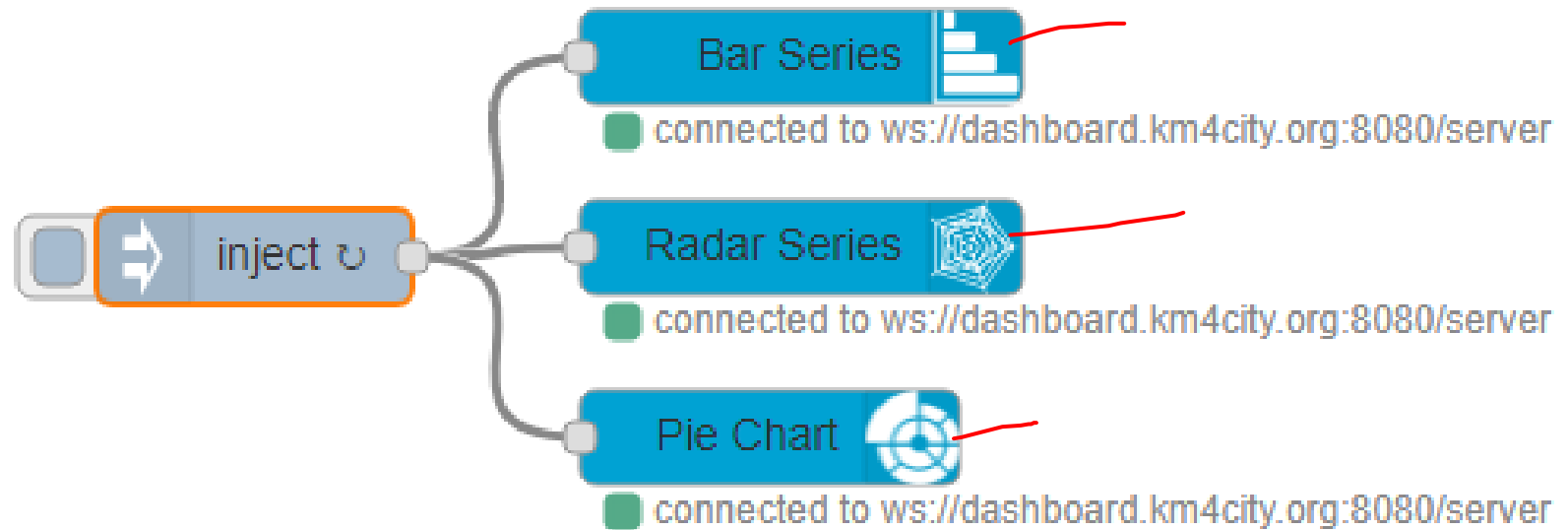
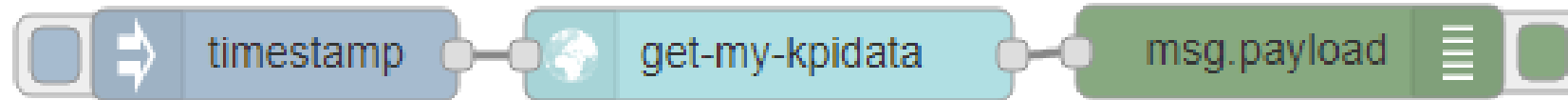
- Click



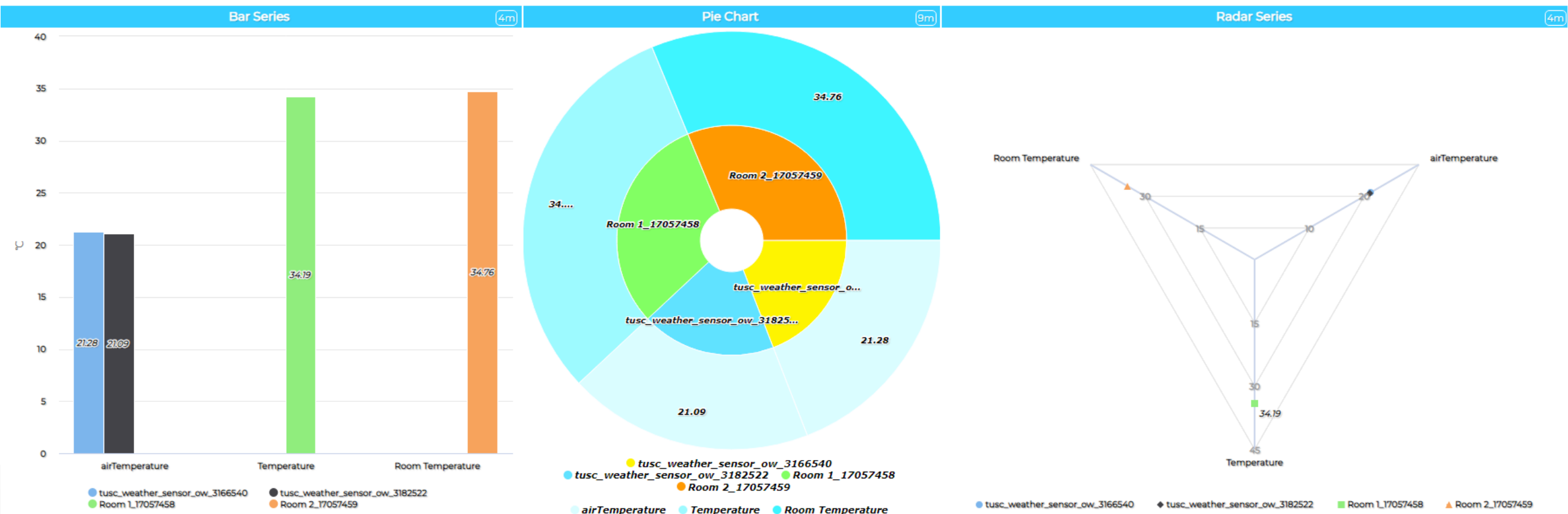
- Observe



# Nodes connections



# Resulting Dashboard



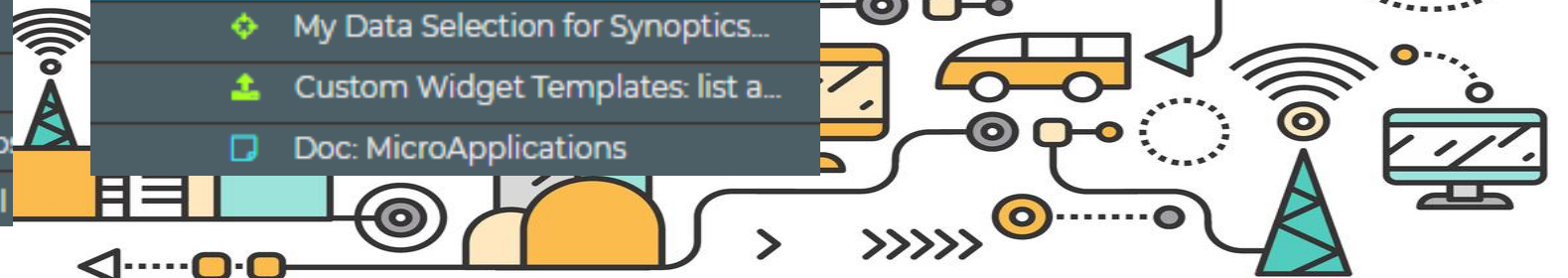
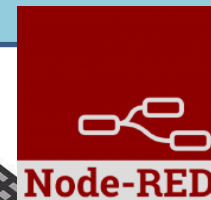


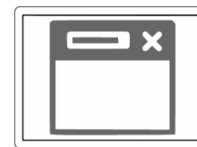
TOP

# IoT Application Integration with Synoptics

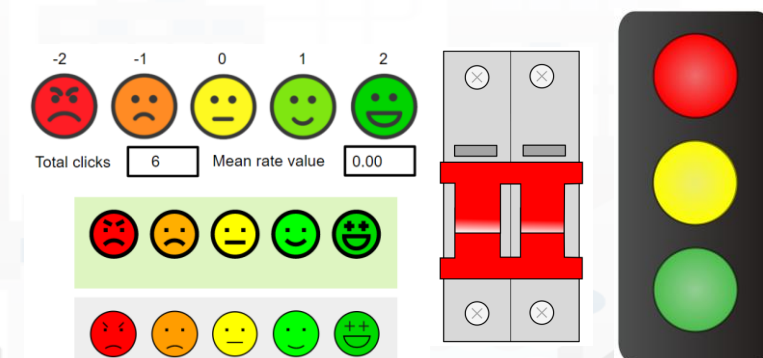
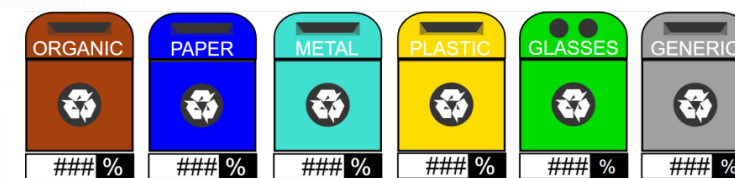
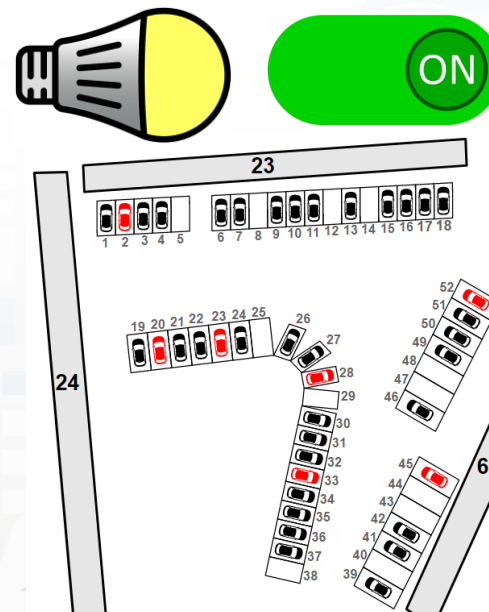
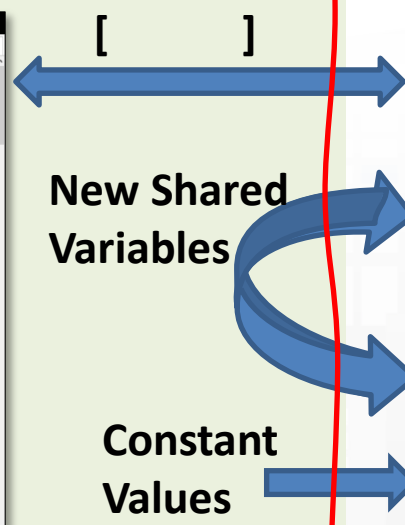
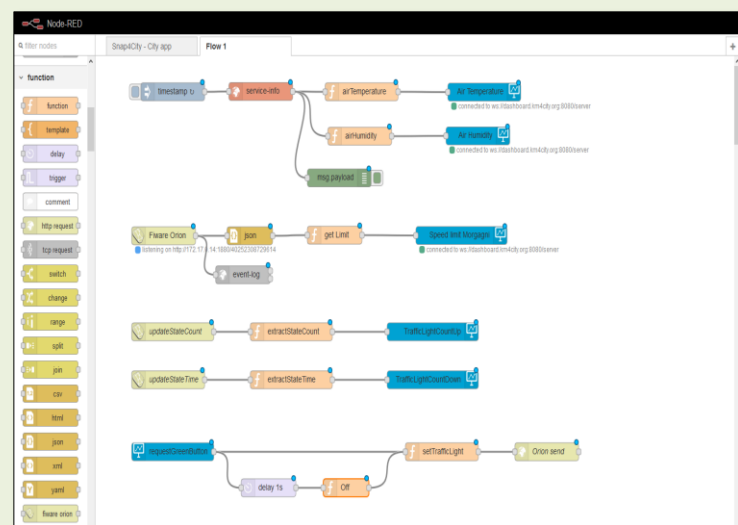
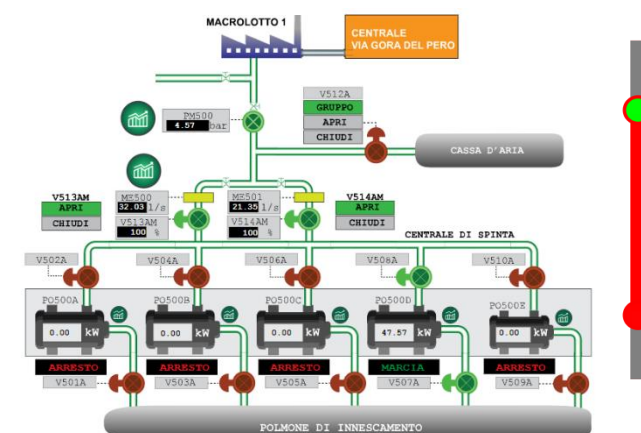
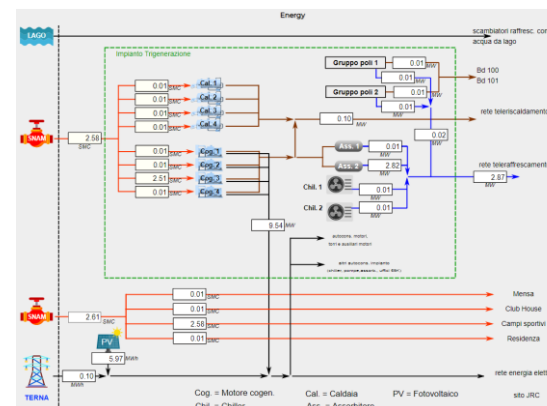
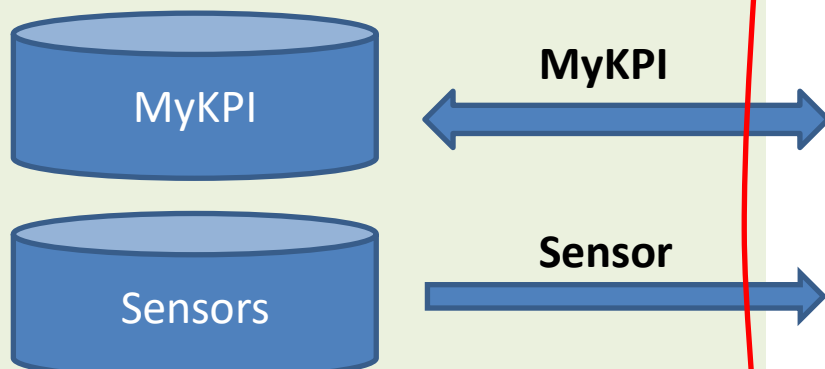
- Processing Logics / IOT App
- Processing Logics / IOT App
- MicroServices for Proc.Logic/IOT App
- MicroServices from DataAnalytic
- IOT MicroServices for Final Users
- IOT MicroServices for Developers
- DOC: Processing Logic/IOT App
- How to Develop Proc.Logic / IoT App
- Create A MicroService from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

- Extra Dashboard Widgets
- Micro Applications
- External Services, WebPages
- Register External Service, WebP...
- Custom Widgets / Synoptics
- My Data Selection for Synoptics...
- Custom Widget Templates: list a...
- Doc: MicroApplications





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

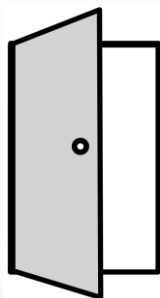
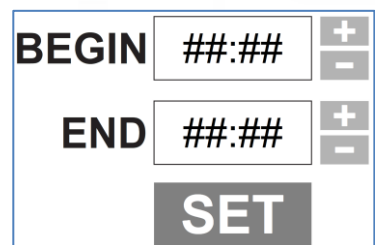
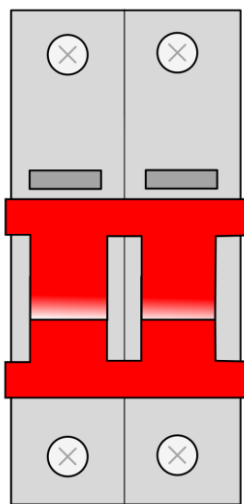


**Web Socket Secure**

# Other examples

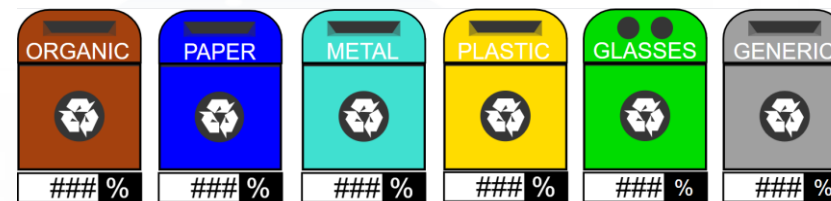
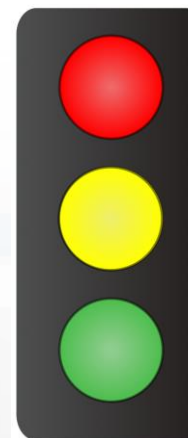
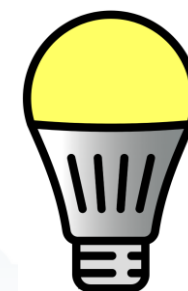
## • Virtual Actuators (sensor-actuator)

- From: Dashboard
- To: IOT App, MyKPI, other Synoptics

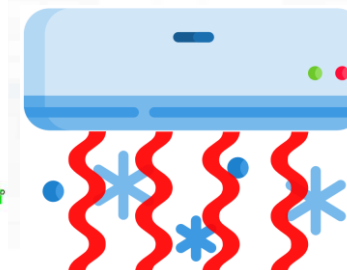
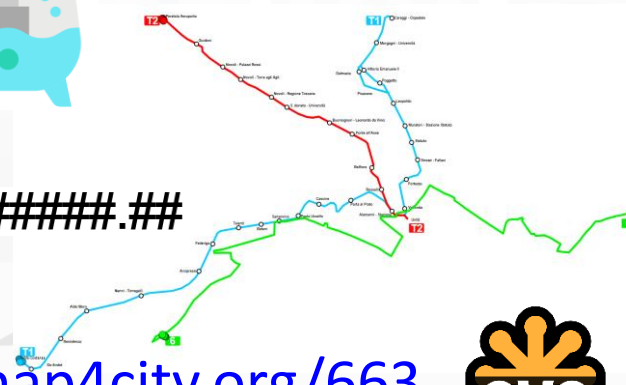


## • Virtual Sensors

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



#####.##



<https://www.snap4city.org/663>

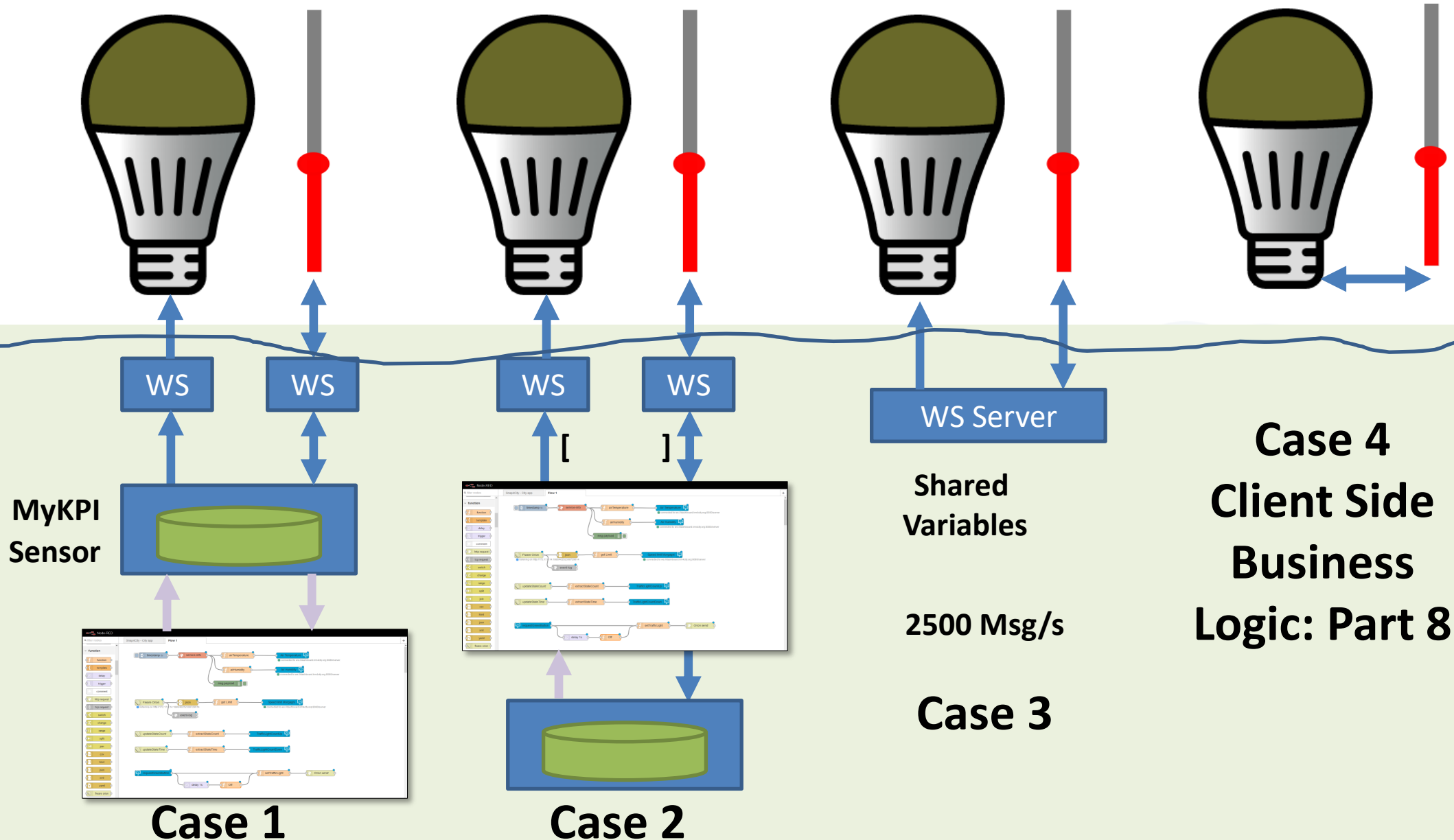




Dashboard  
on Browser

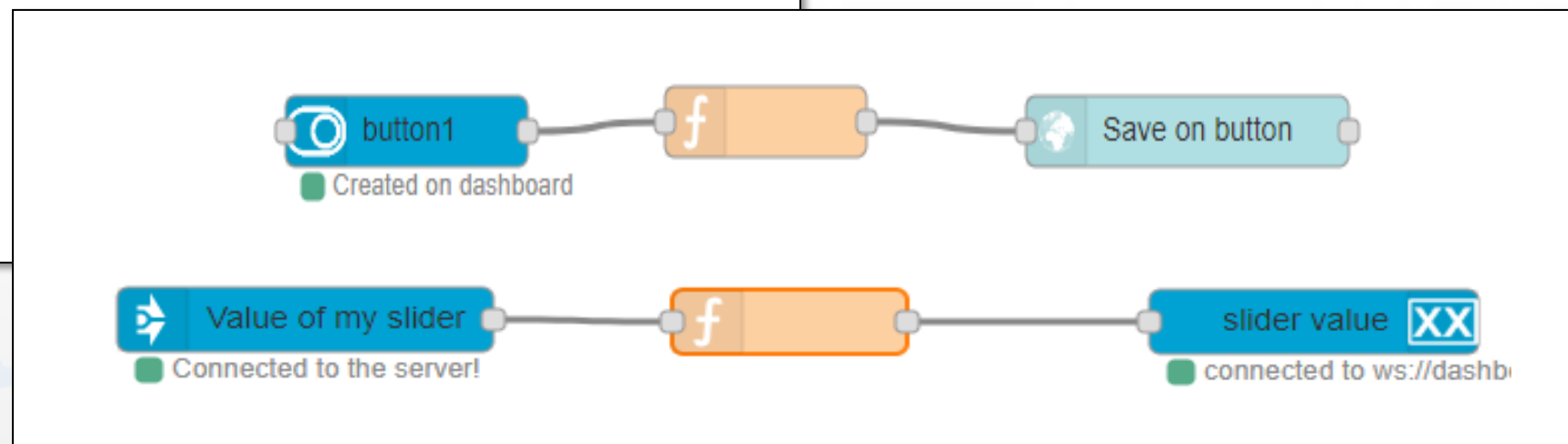
Internet

Storage and  
IOT App on  
cloud or on  
Premise



<https://www.snap4city.org/dashboardSmartCity/view/index.php?iddashboard=Mjc4NA==>

# 10 WS messages per second

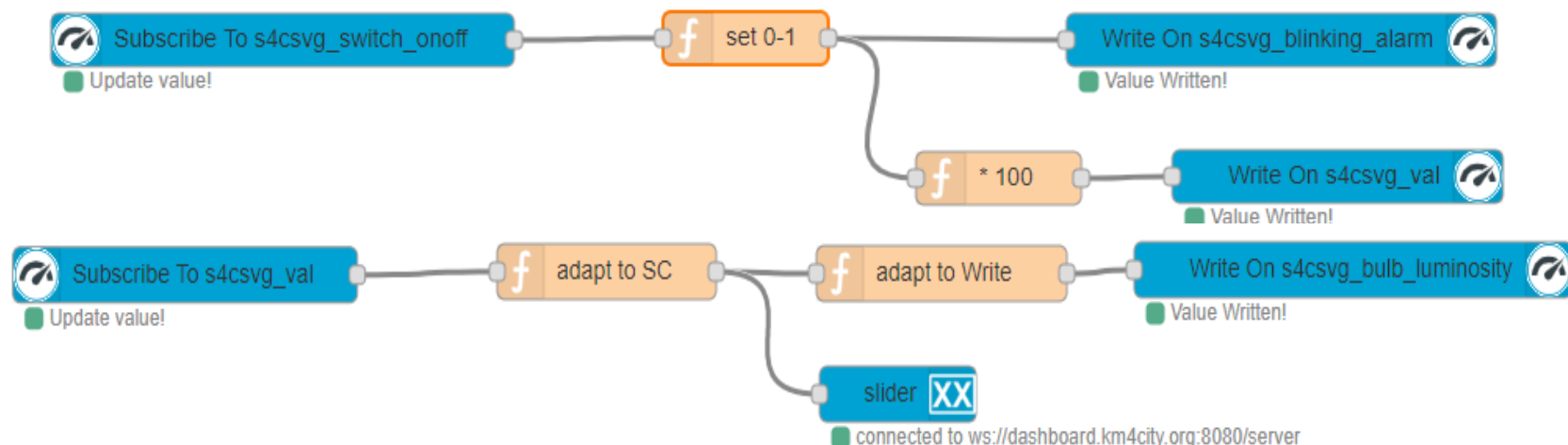
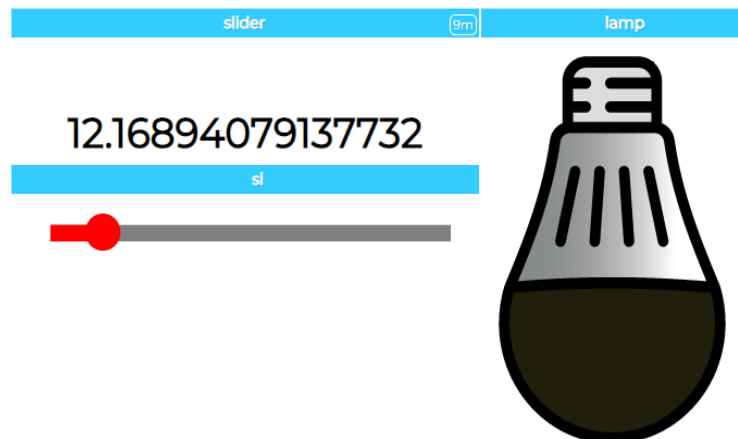


# Case 2: Event Driven 100%

<https://www.snap4city.org/dashboardSmartCity/view/index.php?iddashboard=Mjc4NQ==>

40 messages per second

case 2 SVG WS3





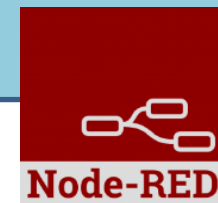
## Read more on

- [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](#)

TOP

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

- 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...



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

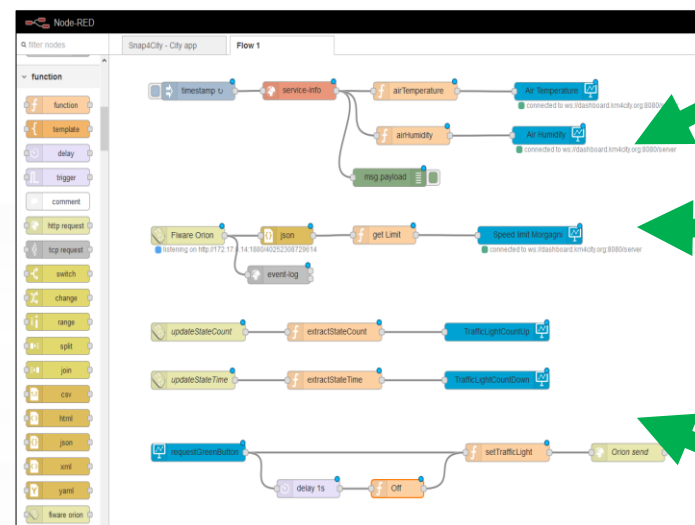


- You can create an HTML page by using the IoT App HTTP nodes
  - and then also send of the WebContent the URL to ask at the dashboard to show the HTML you created 😊

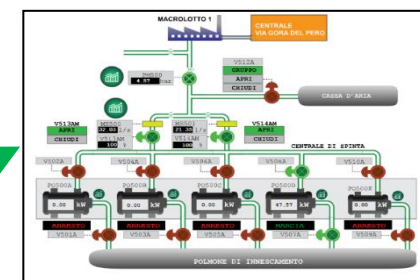


# Advanced IOT Applications

- **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



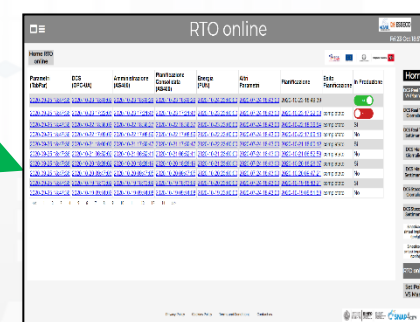
Synoptics  
Custom  
Widgets



Widgets  
Maps  
Buttons  
Keypads  
Controls



HTML pages  
HTML Forms  
Tables



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

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

## IOT App with Dynamic Web Pages

RTO online

Home RTO online

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

Home

DCS Real Time VS Planning

DCS Real Time Giornaliero

DCS Real Time Settimanale

DCS Marce Giornaliero

DCS Marce Settimanale

DCS Stoccaggi Giornaliero

DCS Stoccaggi Settimanale

RTO online

Set Point VS Marce

RTO online

Home RTO online

Visualizza ed Edita altri parametri

Visualizza e produci Pianificazione Consolidata da Pianificazione Ipotetica del 01-10-2020 15:32:05

Non c'è una Pianificazione Consolidata attiva

Elenco esecuzioni pianificazione

Elenco esecuzioni pianificazione in produzione

Avvia Pianificazione

Ultima data di aggiornamento parametri (TabPar): 25-09-2020 18:47:36

Ultima data di aggiornamento dati da DCS (OPC-UA): 01-10-2020 15:33:02

Ultima data di aggiornamento dati da amministrazione (AS400): 23-09-2020 14:51:06

Ultima data di aggiornamento dati da pianificazione vendite ipotetica (AS400): 01-10-2020 15:32:05

Ultima data di aggiornamento dati da pianificazione vendite consolidata (AS400): 01-10-2020 09:32:54

Ultima data di aggiornamento costo giornaliero energia h24 (PUN): 01-10-2020 23:00:00

Ultima data di aggiornamento altri parametri: 24-07-2020 18:43:00

Salva Parametri

Elenco Parametri Iniziali Algoritmo RTO SODA4.0

(effettuare cambiamenti che saranno utilizzati dalla prossima esecuzione)

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

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

0.31 C12\_HCI32 (C12 consumato per ton di prodotto) [ton C12/ton HCI32]

297.54 MAX\_HCI32\_s (Massimo stock HCI32) [ton]

Home

DCS Real Time VS Planning

DCS Real Time Giornaliero

DCS Real Time Settimanale

DCS Marce Giornaliero

DCS Marce Settimanale

DCS Stoccaggi Giornaliero

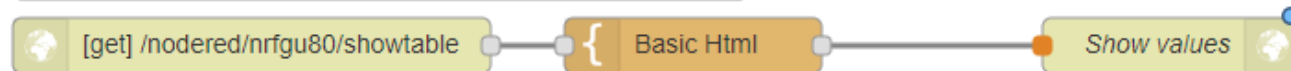
DCS Stoccaggi Settimanale

RTO online

- **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

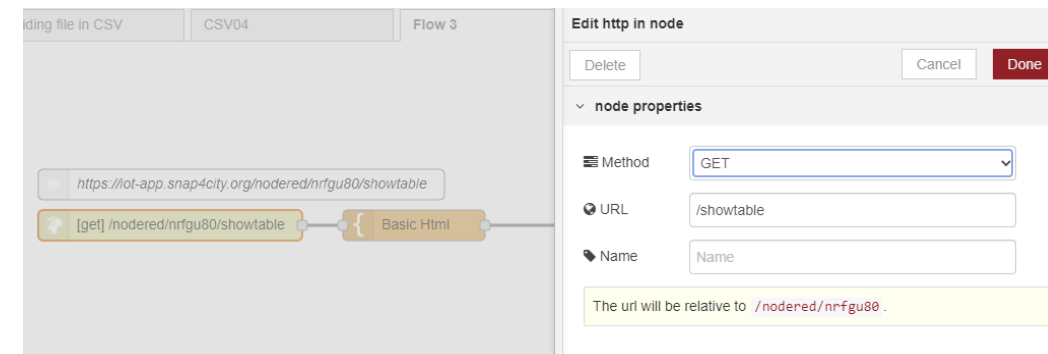
## From IoT App to HTML pages

<https://iot-app.snap4city.org/nodered/nrfgu80/showtable>



```

<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

**This is a table**

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



TOP

# Proc.Logic / IoT App Programming Patterns

FORGING & MANAGING OPEN AND FLEXIBLE WEB AND MOBILE APPS

FROM CITY DASHBOARD TO APPLICATIONS

DATA GATHERING AND CITY DATA KNOWLEDGE MANAGEMENT

IOT/IOE DEVICES AND NETWORKS

IOT APPLICATIONS, THE LOGIC AND THE SMARTNESS

DEVICE SMART CITY API, MICROSERVICES, SNAP4CITY API

SNAP4CITY LIVING LAB FOR COLLABORATIVE WORK

SNAP4CITY FOR BEGINNERS

SNAP4CITY ARCHITECTURE AND ECOSYSTEM, OPENED TO NEW/ELDER AND OLD

TWITTER VIGILANCE: SOCIAL MEDIA ANALYSIS

SNAP4CITY AND KM4CITY PROJECTS

HOW TO ADOPT SNAP4CITY, AND OUR ROADMAP

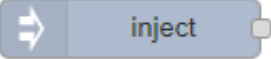
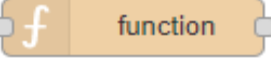
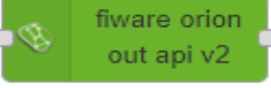
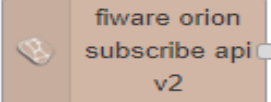
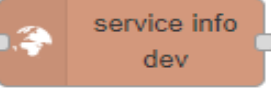
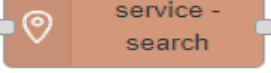
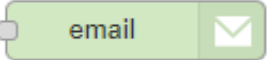
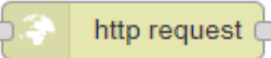
SNAP4CITY THE

## 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...



# examples

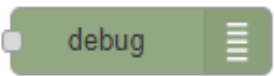
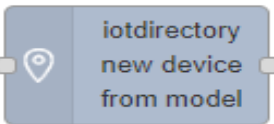
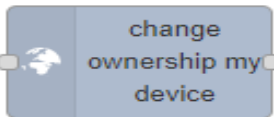
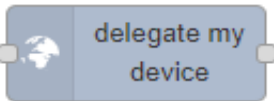
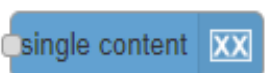
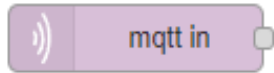
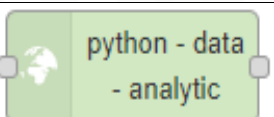
Node shape	Description	Snap4City or standard
	To generate injection messages into a flow, scheduled or on manual demand by click it on left.	standard
	A java script function, from a JSON input to one or more JSON outputs, which can be produced by setting it.	standard
	To send an Entity Message of an 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

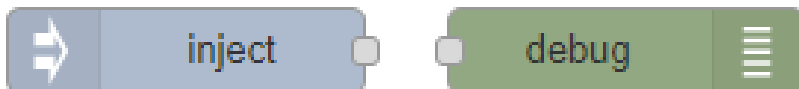
Part 5

Part 4

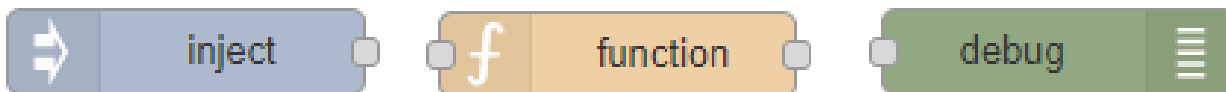


# Some patterns

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



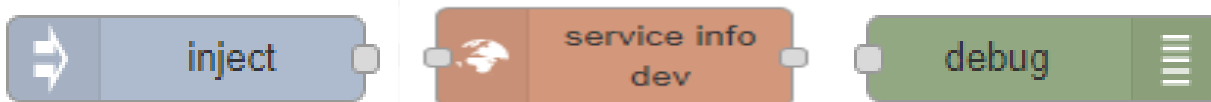
- 2) Hello world of node-red at two steps, the inject provides a push while a JSON is created into the function as *msg.payload = {.....}* and sent/shown to/by the debug.



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

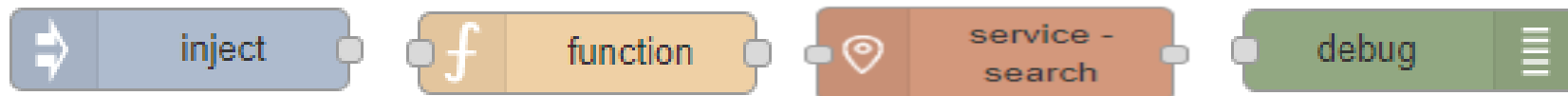


- 4) request on inject of a SURJ to the storage to see data on debug.



## Part 5

- 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

*Any other  
Request  
?*

- 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

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

On Line Training Material (free of charge)



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


# Note on Training Material

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



# HOW ARE YOU GOING TO BUILD THE FUTURE?

Snap4City: a framework for rapid implementation of Decision Support Systems and Smart Applications.



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

## Snap4City: Smart aNalytic APp builder for sentient Cities and IOT

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

Username: paolo.disit

### Search




Training on Tools and Platform

Powered by [www.km4city.org](http://www.km4city.org)



### Organization Groups

DISIT  
• Developer  
• Operative

Updates on

- TECHNICAL OVERVIEW: <https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf>
- Development Life Cycle: <https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf>
- Client-Side Business Logic Widget Manual: <https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf>
- Booklet Data Analytics: Snap4Solutions: [https://www.snap4city.org/download/video/DBL\\_SNAP4SOLUTION.pdf](https://www.snap4city.org/download/video/DBL_SNAP4SOLUTION.pdf)



# 2023 booklets



- Smart City



[https://www.snap4city.org/download/video/DPL\\_SNAP4CITY.pdf](https://www.snap4city.org/download/video/DPL_SNAP4CITY.pdf)

- Industry



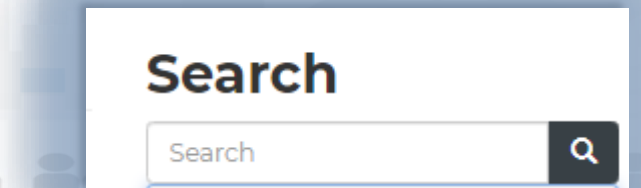
[https://www.snap4city.org/download/video/DPL\\_SNAP4INDUSTRY.pdf](https://www.snap4city.org/download/video/DPL_SNAP4INDUSTRY.pdf)

- Artificial Intelligence



[https://www.snap4city.org/download/video/DPL\\_SNAP4SOLU.pdf](https://www.snap4city.org/download/video/DPL_SNAP4SOLU.pdf)

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















# Tech Overview

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









## Snap4City Platform

### Technical Overview

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

Snap4City:

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

Contact Person: Paolo Nesi, [Paolo.nesi@unifi.it](mailto:Paolo.nesi@unifi.it)

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

1

# Development

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



## Development Life-Cycle

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

### From Snap4City:

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

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

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

# Client Side Business Logic

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

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



## Client-Side Business Logic Widget Manual

### From Snap4City:

- We suggest you read <https://www.snap4city.org/download/video/Snap4Tech-Development-Life-Cycle.pdf>
- We suggest you read the TECHNICAL OVERVIEW:
  - <https://www.snap4city.org/download/video/Snap4City-PlatformOverview.pdf>
- slides go to <https://www.snap4city.org/577>
- <https://www.snap4city.org>
- <https://www.snap4solutions.org>
- <https://www.snap4industry.org>
- <https://twitter.com/snap4city>
- <https://www.facebook.com/snap4city>
- <https://www.youtube.com/channel/UC3tAQ09EbNba8f2-u4vanda>

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



# Overview



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



TOP

# Acknowledgements

FROM CITY  
DASHBOARD TO  
APPLICATIONS

DATA GATHERING  
AND CITY DATA  
KNOWLEDGE  
MANAGEMENT

FORGING &  
MANAGING OPEN  
AND FLEXIBLE WEB  
AND MOBILE APPS

IOT APPLICATIONS  
VS IOT EDGE  
DEVICES

IOT APPLICATIONS,  
THE LOGIC AND  
THE SMARTNESS

ADVANCED  
SMART CITY API,  
MICROSERVICES,  
SNAP4CITY API

SNAP4CITY  
LIVING LAB FOR  
COLLABORATIVE  
WORK

SNAP4CITY FOR  
BEGINNERS

AT BUSINESS  
INTELLIGENCE,  
WHAT-IF AND  
SIMULATION

SNAP4CITY  
ARCHITECTURE AND  
ECOSYSTEM. OPENED  
TO DEVELOPERS  
AND STAKEHOLDERS

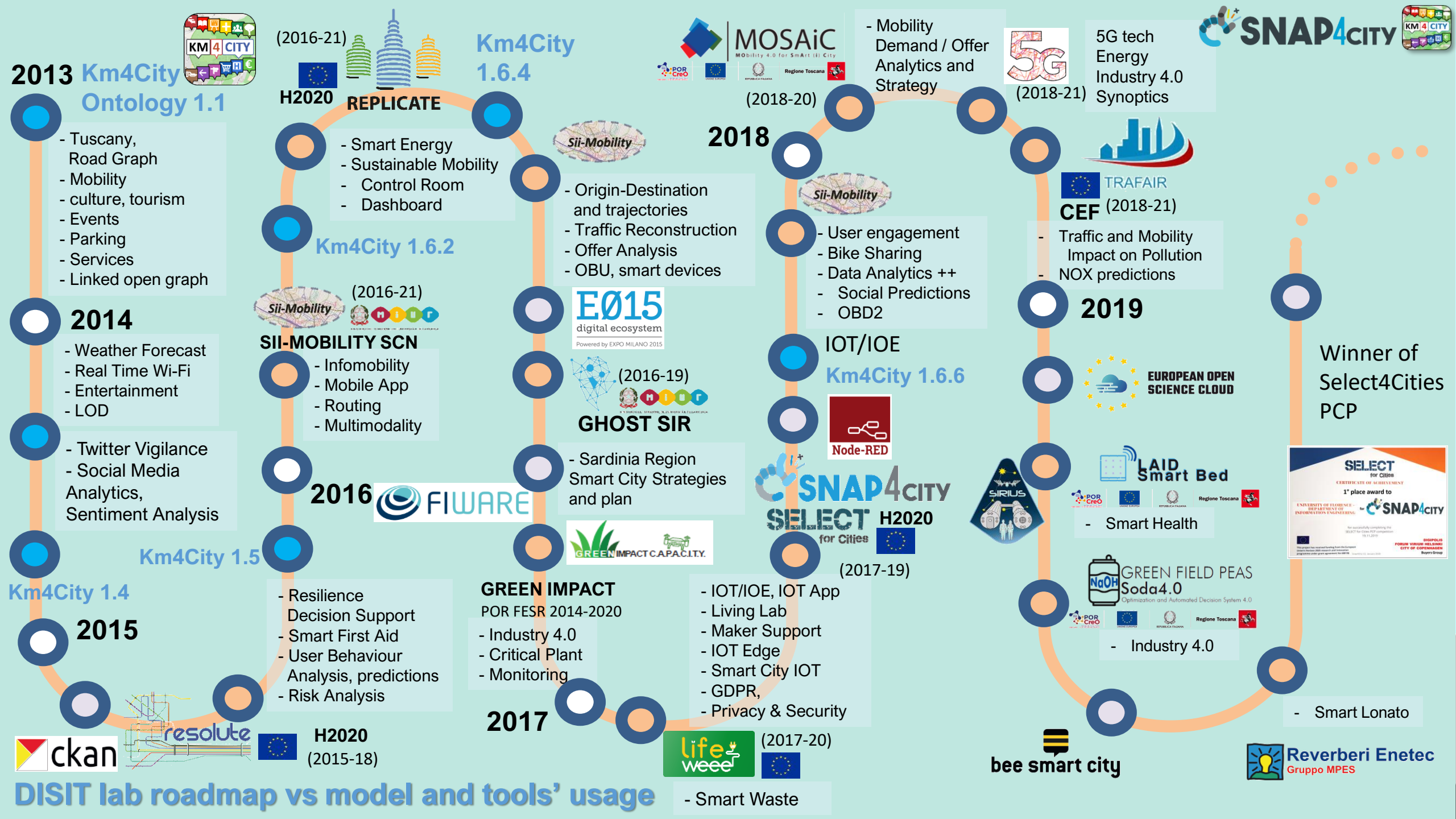
TWITTER  
VIGILANCE: SOCIAL  
MEDIA ANALYSIS

DECISION SUPPORT  
SYSTEM AND CITY  
RESILIENCE

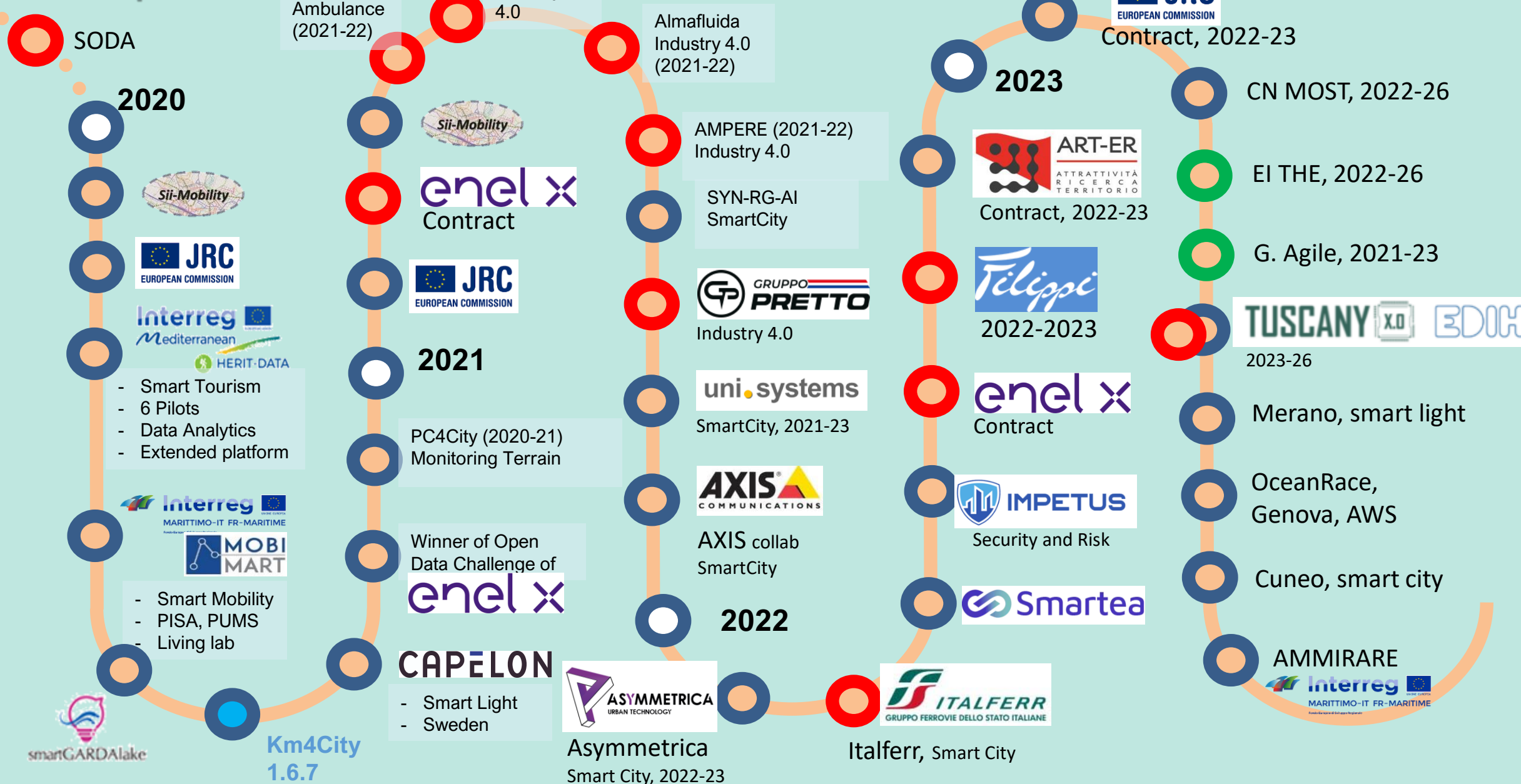
HOW TO ADOPT  
SNAP4CITY, AND  
OUR ROADMAP

SNAP4CITY  
AND KM4CITY  
PROJECTS

SNAP4CITY THE  
VIEW OF THE  
ADMINISTRATORS







TOP



*Be smart in a SNAP!*



**SMARTCITY**  
EXPO WORLD CONGRESS

7-9 November 2023, Barcelona, Spain

Visit Snap4City in Hall 1

## CONTACT

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

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

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



Appliances and Dockers  
**Installations**

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

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



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

**DINFO**  
DIPARTIMENTO DI  
INGEGNERIA  
DELL'INFORMAZIONE

**DISIT**  
DISTRIBUTED SYSTEMS  
AND INTERNET  
TECHNOLOGIES LAB