

# **Smartcity, quali tecnologie per la città dei dati**

**Esperienze del Comune di Cagliari**



**Pierangelo Orofino**

**webinar ForumPa - 14 novembre 2017**

# LE NUOVE SFIDE PER UNA PUBBLICA AMMINISTRAZIONE DIGITALE E INNOVATIVA

## *case studies*

ELEZIONI  
COMUNALI 2016



SPERIMENTAZIONE  
SMART PARKING



I NOSTRI PROGETTI  
*SMART CITY*



# 1

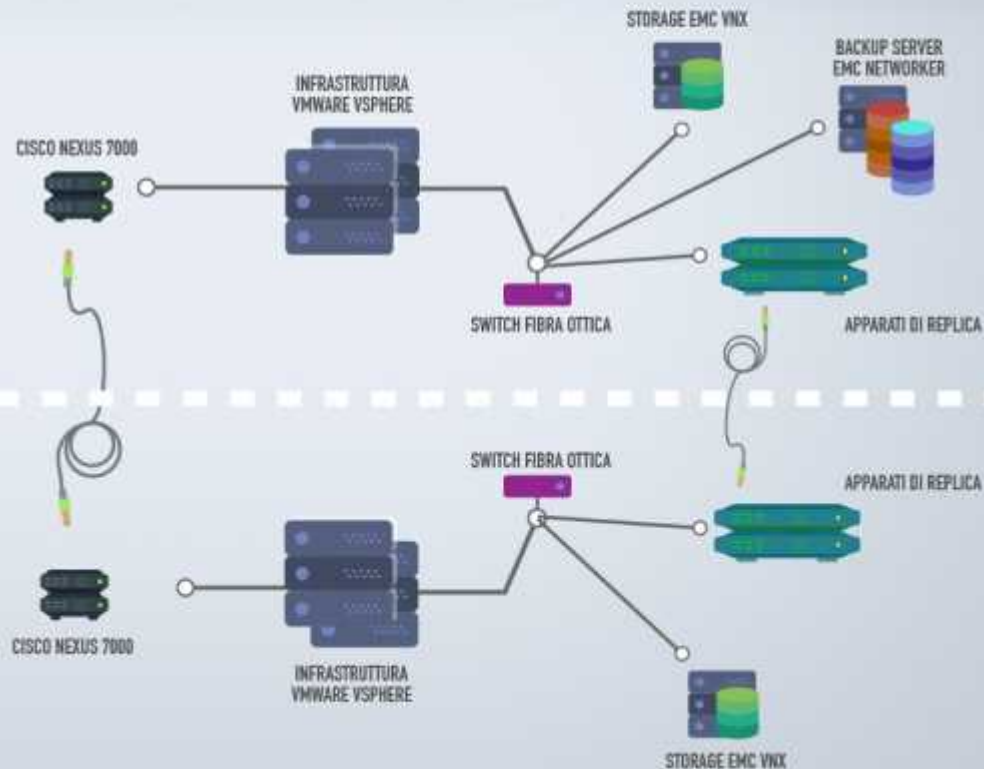
## IL SITO INFORMATIVO PER LE ELEZIONI COMUNALI 2016

*IL DATA CENTER  
DEL COMUNE  
È PROGETTATO SU  
MODERNI STANDARD  
DI VIRTUALIZZAZIONE  
E REPLICA*

SITO PRIMARIO  
VIA SONNINO



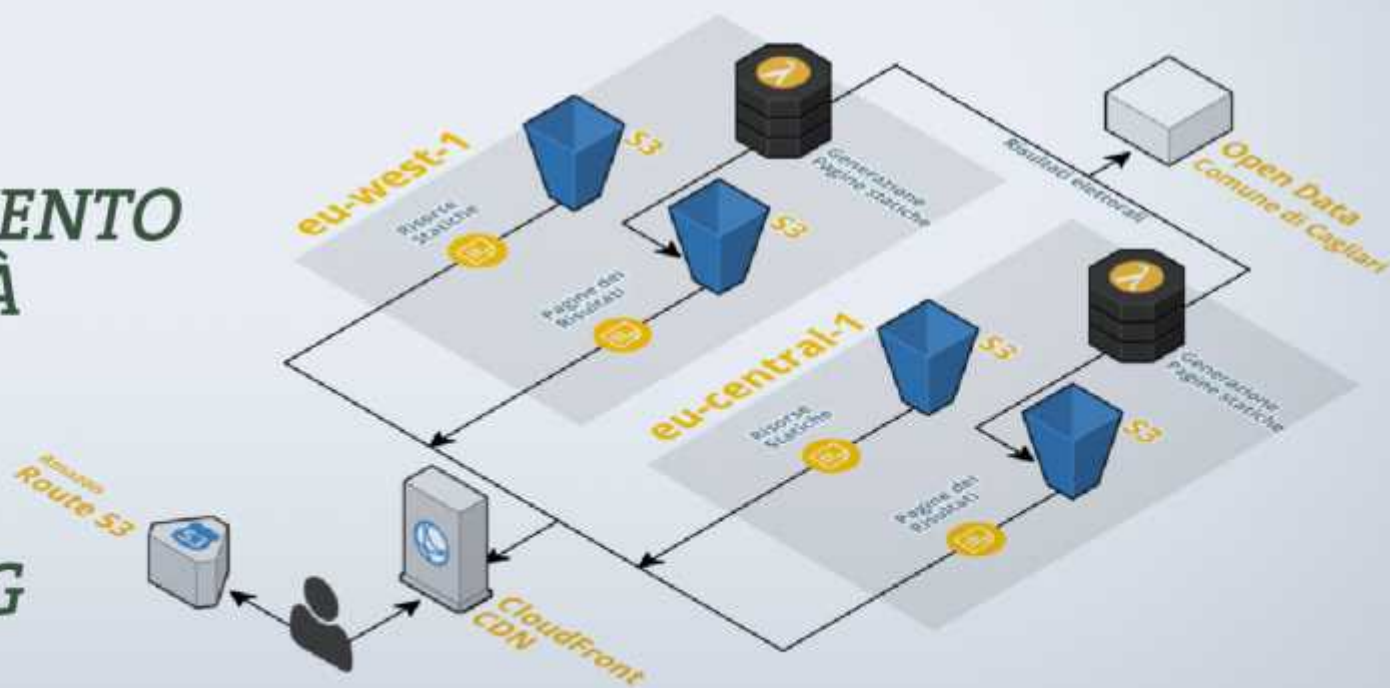
SITO SECONDARIO  
VIA CREPELLANI



# 1

## IL SITO INFORMATIVO PER LE ELEZIONI COMUNALI 2016

PER GESTIRE UN EVENTO  
AD ALTA INTENSITÀ  
DI TRAFFICO  
UTILIZZIAMO  
SISTEMI  
CLOUD COMPUTING  
AMAZON AWS



# 1

## IL SITO INFORMATIVO PER LE ELEZIONI COMUNALI 2016



*SERVIZIO  
EFFICIENTE  
SOTTO STRESS*

*CON SCALABILITÀ  
DELLE RISORSE  
E COSTI MINIMI*



**COSTI TOTALI  
PER TEST  
E PRODUZIONE**  
**125 €**

**Average: 94,394.4359**

**Total: 7,362,766**

**Maximum: 701,673**

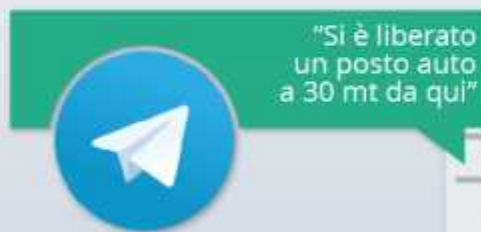
**Minimum: 53**



# 2

## PROGETTO SMART PARKING

*UNA NUOVA INFRASTRUTTURA  
INTERNET OF THINGS  
PER IL COMUNE DI CAGLIARI  
PER INVENTARE UN NUOVO MODO  
DI FRUIRE LA CITTÀ IN DIGITALE*



# 3 SMART ANCHE NEL TRAFFICO



**1** I dati raw sul traffico sono prelevati ad intervalli di tempo prefissati dal database sorgente (SQL Server del CTM) tramite una funzione Lambda.

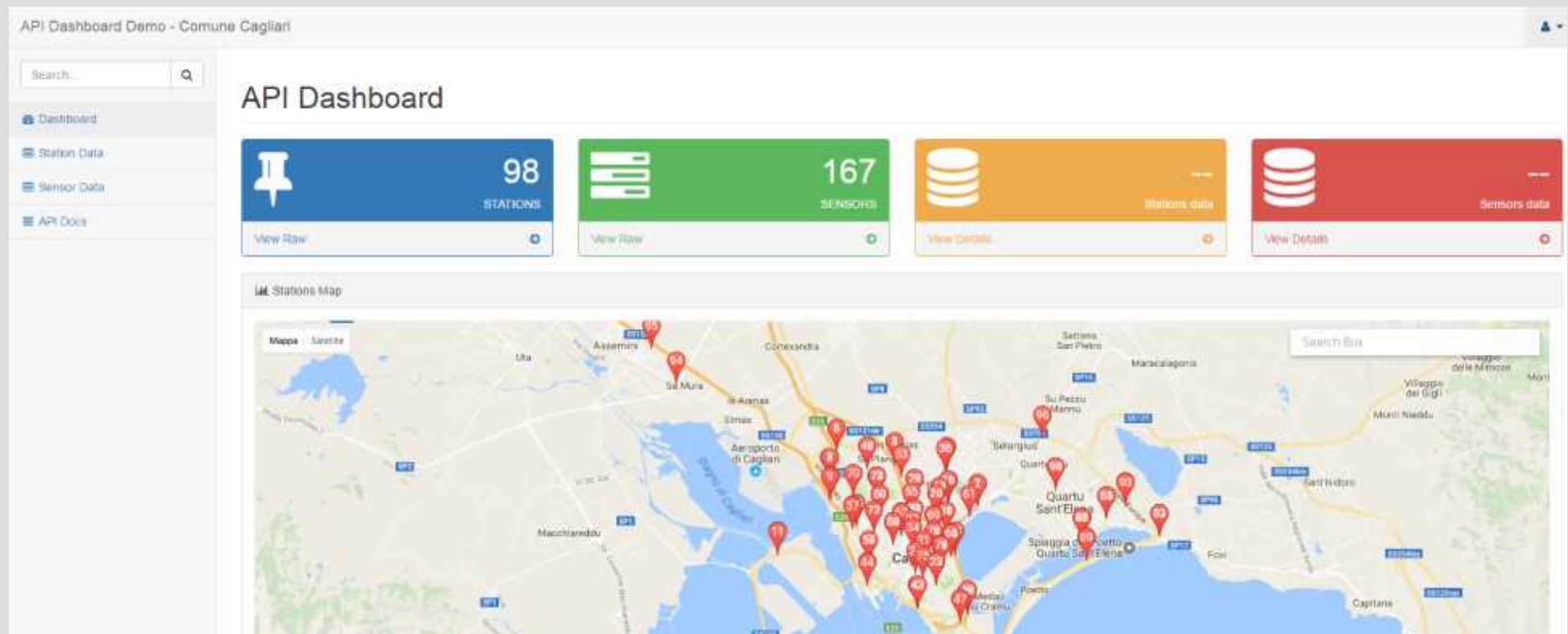
**2** I dati prelevati vengono copiati in apposite tabelle del database nosql DynamoDB.

**3** Una batteria di funzioni Lambda implementano la business logic delle API, che vengono erogate attraverso il servizio API Gateway che, opportunamente configurato, ne gestisce le logiche d'accesso. Grazie alla filosofia di provisioning di DynamoDB e alla configurazione di API Gateway è possibile calibrare il numero di chiamate al secondo supportate.

**4** Le applicazioni per il cittadino, o per i servizi interni al Comune, sono costruite sulla base di queste API (ad es. su EC2 in autoscaling)

**5** Le stesse API, erogate tramite https su internet, possono essere usate da sviluppatori di terze parti per costruire applicazioni, portali, ecc.

# Cruscotto delle API





# Documentazione API

The screenshot shows the Swagger UI for the 'Comune di Cagliari - opendata' API. The page title is 'Comune di Cagliari - opendata' and the description is 'The Cagliari Datastore is a free and open data sharing portal where anyone can access data relating to the city.' The API is titled 'Traffic : API for accessing data of the sensors for traffic monitoring'. There are six API endpoints listed:

- GET** /stations: Retrieve all stations definitions.
- GET** /station/{station\_id}/readings: Retrieve specific station data.
- GET** /station/{station\_id}: Retrieve specific station definition.
- GET** /sensors: Retrieve all sensors definitions.
- GET** /sensor/{sensor\_id}/readings: Retrieve specific sensor data.
- GET** /sensor/{sensor\_id}: Retrieve specific sensor definition.

The 'Response Class (Status 200)' section shows the following JSON structure:

```
{
  "stations": 8,
  "sensors": 7,
  "data": {
    "id": "00000000000000000000000000000000",
    "location": "Cagliari",
    "flow": "Outgoing",
    "efficiency": "Normal",
    "type": "Traffic",
    "accuracy": "High"
  }
}
```

The response content type is set to 'application/json'.



# Opendata e OpenService



# API Rest - dettaglio rimozioni

The screenshot shows the API documentation for the resource `/cagliari/elencoRimozioni`. The page includes a navigation menu with links like HOME, OPEN DATA, DATASET, OPEN SERVICE, METODI SVILUPPATORI, INTERFACCE A.P.I., and NOTIZIE. The main content area displays the resource name, a brief description, and details for the GET method. A table lists the XML schema elements: `motivo`, `dataA`, and `dataB`, with their descriptions and whether they are required.

**Risorsa /cagliari/elencoRimozioni**  
Fornisce una lista di informazioni circa le rimozioni

**Dettaglio risorsa**  
Nome: `elencoRimozioni`  
Namespace: `cagliari`  
Sorgente: Comune Cagliari  
URI: `http://www.comune.cagliari.it/portale/api/30/cagliari/elencoRimozioni`  
Estensione: `xml` per la risposta in formato xml o `json` per la risposta in formato json

**GET**

Metodo Disponibile  
Descrizione: Fornisce una lista di informazioni circa le rimozioni  
Autenticazione: Nessuna, questo metodo è pubblico

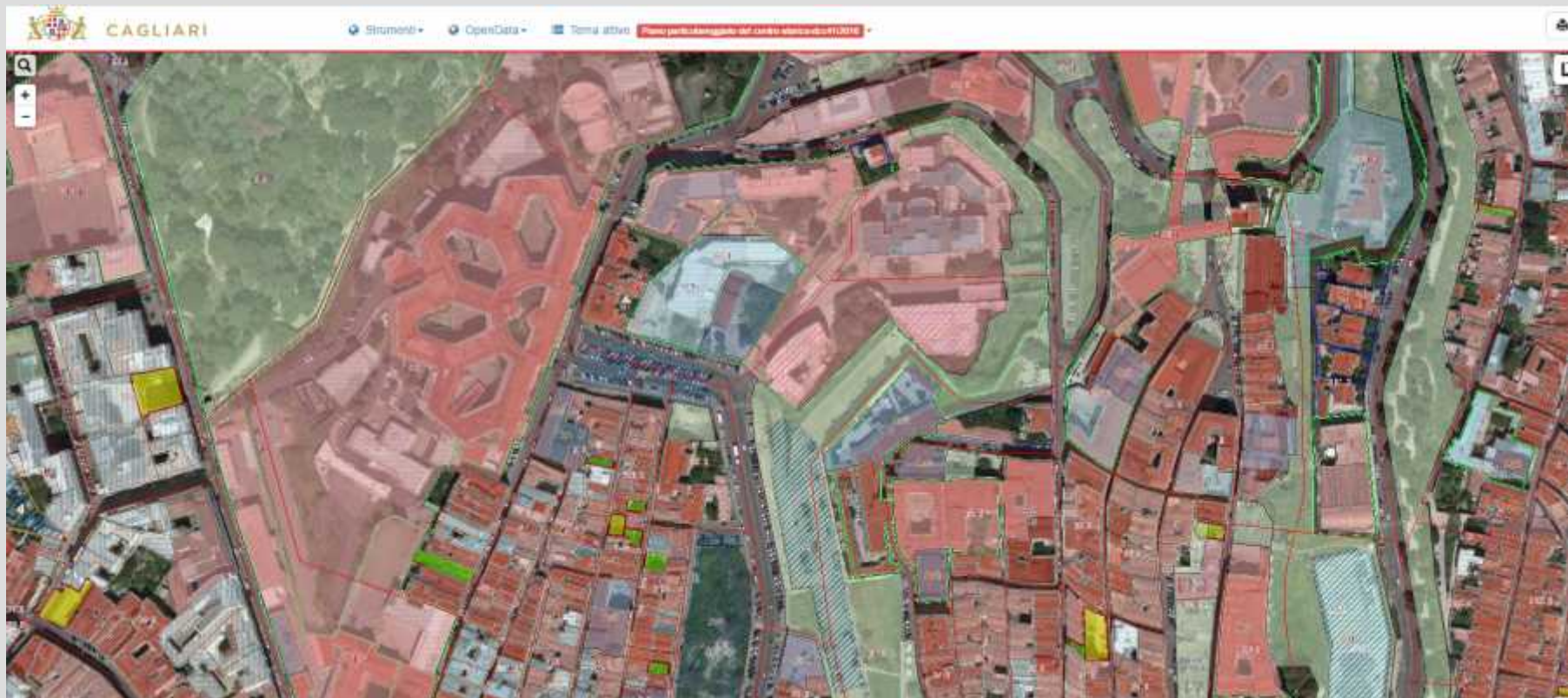
Schema XML: `Risorsa`

Nome	Descrizione	Obbligatorio
<code>motivo</code>	Motivo della rimozione	Si
<code>dataA</code>	Data data(es/NTA/YYYY)	Si
<code>dataB</code>	Data data(es/PM/YYYY)	Si

POST  
PUT  
DELETE



# il nuovo Sistema Informativo Territoriale



# SIT – sezione OpenData

**Tematismi disponibili in OpenData**

Inquadramento    RetiTematiche    ProtezioneCivile    EfficienzaEnergetica    Urbanistica    PoliziaMunicipale

**Rilevazione degli incidenti stradali (anno2014)**  
Formato dei dati: [GeoJSON](#)  
Aggiornamento: **Febbraio 2017**  
Tipo di primitiva: Poligono  
Sistema di coordinate: WGS84-EPSG-3857

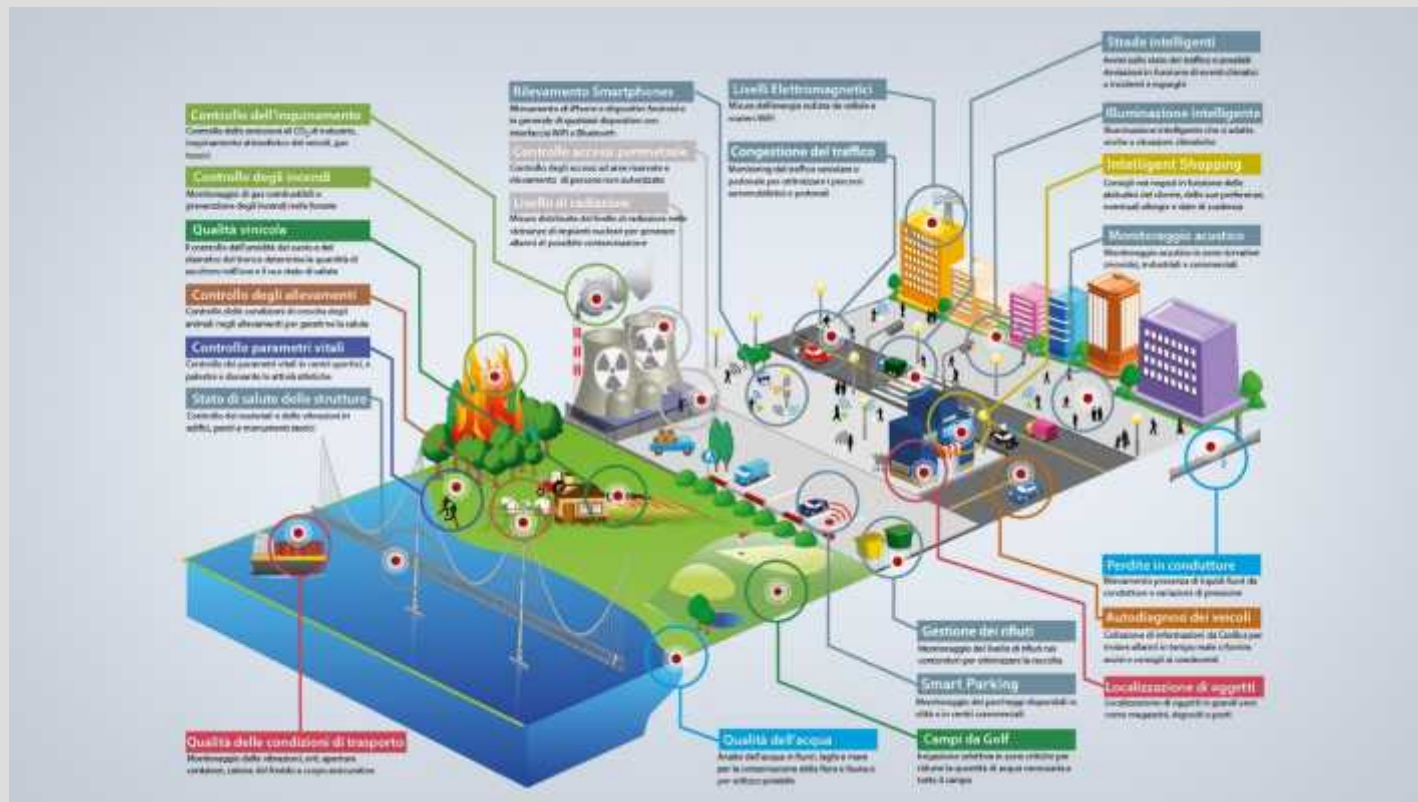
📄 scarica

**Rilevazione delle situazioni di pronto intervento (anno2014)**  
Formato dei dati: [GeoJSON](#)  
Aggiornamento: **Febbraio 2017**  
Tipo di primitiva: Poligono  
Sistema di coordinate: WGS84-EPSG-3857

📄 scarica

**Rilevazione degli interventi di rimozione (anno2014)**  
Formato dei dati: [GeoJSON](#)  
Aggiornamento: **Febbraio 2017**  
Tipo di primitiva: Poligono  
Sistema di coordinate: WGS84-EPSG-3857

# Smart city on the Cloud (PON-Metro)



# Grazie per l'attenzione!

## Domande?

Piero Orofino  
Servizio Innovazione Tecnologica e Sistemi Informatici  
Comune di Cagliari  
[piero.orofino@comune.cagliari.it](mailto:piero.orofino@comune.cagliari.it)