

PROGETTO

TESSUTO DIGITALE METROPOLITANO



POR FESR 2014-2020, Azione 1.2.2

(Delibera 66/14 del 13.12.2016, Progetto Complesso area ICT della S3)

Deliverable

Deliverable D3.1 – PORTALE DI DISTRIBUZIONE OPEN DATA STANDARDIZZATO ED ACCESSIBILE

Data di consegna prevista: 12/2018

Data di consegna effettiva: 12/2018

Natura: Rapporto + Sito

Versione: 1.0

Livello di Disseminazione (PU)

Sommario

Il portale di distribuzione Open Data rappresenta il meccanismo primario di fruizione dei dati prodotti dal progetto TDM. Con questa release, Dicembre 2018, rendiamo disponibile una versione pienamente accessibile del portale permettendo quindi ad iniziative di trasferimento tecnologico di iniziare a sperimentare con i dati e le interfacce Open Data di TDM. Questa è una release con fini principalmente metodologici: con essa vengono accese le interfacce e fornito un insieme significativo di esempi concreti delle tipologie di dati prodotti ed esportati dal progetto. Questa dotazione di dati verrà via via ampliata con l'evoluzione del lavoro. Per descrivere i dati TDM usa i principali standard e best practice rilevanti nel contesto delle Smart Cities, basandosi, ad esempio, su formato dati NGSI utilizzato da OASC/FIWARE per i dati da sensori e dallo standard internazionale CF (Climate and Forecast) per la descrizione dei dati provenienti dalle simulazioni e dalle acquisizioni radar. Assieme al portale di accesso, disponibile alla URL <http://data.tdm-project.it> e alla relativa interfaccia rest <http://rest.tdm-project.it> viene reso disponibile un portale con applicazioni dimostrative che illustrano l'uso dei dati alla URL: <http://demo.tdm-project.it>

Redazione			
	Name	Partner	Data
Autore	F. Bettio / G. Busonera / M. Del Rio / M. Gaggero / E. Gobbetti / C. Impagliazzo / G. Lecca / G. Pusceddu / S. Manca / M. Marrocu / L. Massidda / G. Zanetti	CRS4	20/12/2018
Approvato da	G. Zanetti & E. Gobbetti	CRS4	21/12/2018

Storia e Contributi			
Vers.	Data	Commento	Autori
V0.5	01/10/2018	Primo draft con organizzazione contenuti	E. Gobbetti / G. Zanetti (CRS4)
V0.6	29/11/2018	Primo draft con tutte le sezioni	F. Bettio / G. Busonera / M. Del Rio / M. Gaggero / E. Gobbetti / C. Impagliazzo / G. Lecca / G. Pusceddu / S. Manca / M. Marrocu / L. Massidda / G. Zanetti (CRS4)
V0.6	05/12/2018	Prima versione completa	F. Bettio / G. Busonera / M. Del Rio / M. Gaggero / E. Gobbetti / C. Impagliazzo / G. Lecca / G. Pusceddu / S. Manca / M. Marrocu / L. Massidda / G. Zanetti (CRS4)
V0.7	19/12/2018	Revisione completa con esempi d'uso e catalogo eventi	F. Bettio / G. Busonera / M. Del Rio / M. Gaggero / E. Gobbetti / C. Impagliazzo / G. Lecca / G. Pusceddu / S. Manca / M. Marrocu / L. Massidda / G. Zanetti (CRS4)
V1.0	20/12/2018	Versione finale	E. Gobbetti / G. Zanetti (CRS4)

Sommario

Sommario.....	3
1. Introduzione	5
2. Soluzioni aperte per dati aperti	6
2.1 L'iniziativa OASC.....	6
2.2 OASC e TDM.....	7
3. Architettura del portale di distribuzione dati	8
3.1 Architettura generale TDM.....	8
3.2 Architettura del sotto-sistema di presentazione open-data	9
3.2.1 Portale CKAN.....	9
3.2.2 Accesso ai dati.....	10
4. Tipologie di dati trattati e relativi formati	13
4.1 Dati provenienti da misurazioni puntuali	13
4.1.1 Tipologie di eventi trattati	13
4.1.2 Standard NGSi.....	13
4.1.3 Estensioni proposte allo standard NGSi.....	14
4.1.4 Dati di telemetria (extra NGSi).....	15
4.2 Dati meteo e radar.....	16
4.2.1 Formati e convenzioni.....	16
4.2.2 Dati radar X-band.....	17
4.2.3 Dati della catena operativa di previsione meteo a scala limitata (LAM)	18
5. L'ecosistema dei portali TDM	22
5.1 Portale demo	22
5.2 Grafana	24

6.	Applicazioni di accesso visuale	27
6.1	Visualizzazione di dati da sensoristica	27
6.2	Visualizzazione di eventi meteo	30
6.2.1	Funzionalità.....	30
6.2.2	Implementazione	31
6.2.3	Interfaccia generale	33
6.2.4	Esempi di utilizzo	35
7.	Appendici.....	45
7.1	Ricerca dei dataset disponibili attraverso il front-end CKAN.....	45
7.2	Catalogo dati meteo disponibili	49
7.2.1	Evento del 2 maggio allerta rossa per rischio idraulico	49
7.2.2	Le precipitazioni intense del mese di ottobre.....	49
7.2.3	Evento del 10/10 rischio idrogeologico elevato e idraulico moderato per la zona di Cagliari.....	50
7.2.4	Evento del 18/10 rischio idrogeologico moderato/elevato per la zona di Cagliari.....	50
7.2.5	Evento del 03/11 rischio idrogeologico moderato per la zona di Cagliari	50
7.2.6	Alcune ondate di calore di luglio 2018 nella stazione di Cagliari.....	51
7.2.7	Evento del 13 luglio.....	51
7.2.8	Evento del 30 luglio.....	51
7.3	Esempi di utilizzo dei dati ed interfacce	52

1. INTRODUZIONE

Il portale di distribuzione Open Data rappresenta il meccanismo primario di fruizione dei dati prodotti dal progetto TDM. Con questa release, Dicembre 2018, rendiamo disponibile una versione pienamente accessibile del portale permettendo quindi ad iniziative di trasferimento tecnologico di iniziare a sperimentare con i dati e le interfacce Open Data di TDM.

Questa è una release con fini principalmente metodologici: con essa vengono accese le interfacce e fornito un insieme significativo di esempi concreti delle tipologie di dati prodotti ed esportati dal progetto. Questa dotazione di dati verrà via via ampliata con l'evoluzione del lavoro.

Tra le specificità del progetto TDM vi è l'esplorare come si possano integrare/interpolare in maniera sinergica informazioni digitali provenienti direttamente da sensori con quelle ottenute da simulazione per costruire un modello digitale multiscala di una zona metropolitana. Come conseguenza, i dati messi a disposizione da TDM corrispondono a diversi strati di informazione e sono organizzati, dal punto di vista logico, in varie forme, tra cui serie temporali di dati ottenuti da sensori "puntuali" e dati volumetrici rappresentanti campi continui su regioni dello spazio.

Per massimizzare la fruibilità dei dati esposti sia in termini della loro consultazione statica che di integrazione dinamica in applicazioni a valle, TDM fornisce un sistema che integra una indicizzazione estensiva dei dataset disponibili con due modalità di accesso: il classico download di interi dataset e l'accesso casuale mirato, e molto più veloce, a specifici pezzi di informazione come, ad esempio, nel caso di un dataset ottenuto da una simulazione meteo, il campo di velocità in una data sottoregione spaziale e temporale.

Per descrivere i dati TDM usa i principali standard e best practice rilevanti nel contesto delle Smart Cities, basandosi, ad esempio, su formato dati NGSI utilizzato da OASC/FIWARE per i dati da sensori e dallo standard internazionale CF (Climate and Forecast) per la descrizione dei dati provenienti dalle simulazioni e dalle acquisizioni radar.

Assieme al portale di accesso, disponibile alla URL <http://data.tdm-project.it> e alla relativa interfaccia rest <http://rest.tdm-project.it> viene reso disponibile un portale con applicazioni dimostrative che illustrano l'uso dei dati alla URL: <http://demo.tdm-project.it>



2. SOLUZIONI APERTE PER DATI APERTI

Uno degli obiettivi principali del progetto TDM è di realizzare un'architettura scalabile per l'acquisizione, l'integrazione e l'analisi di dati provenienti da sorgenti eterogenee in grado di gestire i dati generati da un'area metropolitana estesa. L'implementazione prevista nel quadro del progetto riguarda casi di studio nell'area metropolitana della città di Cagliari, ma l'obiettivo è quello di generare soluzioni scalabili generali che possano servire da *best practice* per future implementazioni di servizi in aree geografiche ampie e/o densamente popolate. Il progetto deve quindi combinare, sviluppare ed estendere soluzioni tecnologiche in vari campi, dalla sensoristica ai big data e dalla simulazione alla visualizzazione.

Rendere facilmente fruibili tutti i dati raccolti ed elaborati da organizzazioni pubbliche, o forniti da privati, ad esempio attraverso contributi volontari, è uno degli obiettivi da raggiungere. La piena fruizione dei dati non solo consente di aumentare la qualità di vita attraverso una migliore pianificazione e gestione del territorio, ma permette potenzialmente la nascita di nuovi servizi pubblici o privati a valore aggiunto. Il potenziale impatto delle soluzioni sviluppate è tanto ampio quanto più ampio è il bacino di utenza al quale i servizi si rivolgono, e a quanti più dati e più tipi si può accedere. Un servizio sviluppato sui dati di una sola città, per quanto grande, avrà come bacino di utenza e quindi potenziale clientela, solo la popolazione di quella città. Se però il servizio potesse accedere ai dati anche di altre città, il bacino di utenza sarebbe maggiore e con esso le opportunità commerciali ad collegate.

Per permettere questo è necessario creare soluzioni interoperabili basate su standard condivisi.

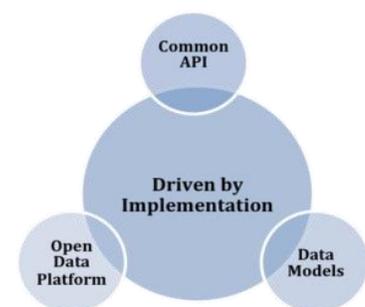
2.1 L'INIZIATIVA OASC

L'Open and Agile Smart Cities Initiative, OASC, è un'iniziativa promossa da diverse città in diversi paesi del mondo con lo scopo di creare un vasto e unico mercato di servizi per le Smart Cities basato sugli OpenData. Fondata nel gennaio 2015, la OASC vede attualmente 117 città aderenti in 24 nazioni tra Europa, America Latina e Asia-Pacifico con una crescita costante. Su invito del CRS4 in qualità di membro della Big Data Value Association di cui OASC è una iniziativa, il Comune di Cagliari ha aderito all'OASC col secondo gruppo di città, sottoscrivendo il Documento di Intenti già a settembre del 2015. Altre città italiane aderenti sono Ancona, Genova, Lecce, Messina, Milano, Palermo e Terni. A conferma dell'importanza dell'iniziativa OASC in Italia, il 4 Aprile 2017 si è tenuto a Milano il primo incontro OASC delle città italiane, alla quale hanno partecipato sia il comune di Cagliari che CRS4.

L'OASC invita le città ad adottare quattro semplici meccanismi come *standard de facto*:

- Un approccio *driven-by-implementation*, ossia la collaborazione tra le comunità, amministrazioni e gli sviluppatori nella creazione dei servizi;
- delle API standard;
- un set di modelli di dato;
- una piattaforma per gli open data.

Per quanto riguarda le API e la piattaforma OpenData, OASC ha deciso di adottare quanto sviluppato all'interno dell'ecosistema FIWARE. In particolare, per le API, OASC ha deciso di adottare le FIWARE NGSI API. Queste sono un set di API standard, aperte e



libere da royalty per la gestione di informazioni contestualizzate sul quanto succede in una città. Per quanto riguarda la piattaforma OpenData, si è invece scelto di adottare il portale OpenData dell'ecosistema FIWARE CKAN, in quanto strumento aperto, flessibile e facilmente distribuibile. CKAN inoltre è già integrato ed esteso come parte dell'architettura di riferimento di FIWARE. Dato che le NGS API sono agnostiche del data model i modelli di dato adottati da OASC sono basati inizialmente sui risultati ottenuti dal progetto CitySDK project. Tuttavia ulteriori correzioni ed estensioni di questo modelli sono previsti in futuro anche in base al feedback dell'uso corrente o di sperimentazioni.

2.2 OASC E TDM

Conformemente con questa visione, la parte tecnologica inerente gli Open Data e gli aspetti *Smart Cities* del progetto TDM saranno declinati in base alle indicazioni della OASC, e forniranno quindi un esempio di best-practice che potrà fornire utili indicazioni per implementazioni a larga scala da parte delle amministrazioni pubbliche.

In particolare, il progetto TDM adotta la piattaforma FIWARE per tutte le componenti dell'architettura riguardanti lo 'IoT', la gestione dei dispositivi remoti, la gestione del dato dal punto di vista logico e la pubblicazione degli OpenData relativi ai risultati delle elaborazioni scientifiche.

Per la rappresentazione delle informazioni, il progetto TDM ha scelto di adottare, ove possibile, i modelli di dato armonizzati suggeriti dalla piattaforma FIWARE, mentre laddove un particolare tipo di dato non abbia un modello armonizzato corrispondente, questo verrà studiato e creato in modo tale da poter poi essere sottoposto all'approvazione della stessa Fiware Foundation, l'organismo che coordina lo sviluppo e la diffusione della piattaforma FIWARE, per l'adozione come standard.

Ciò rappresenta un tipo di contributo che il TDM fa verso FIWARE, reso possibile e agevolato in special modo dall'adesione del CRS4 alla FIWARE Foundation come membro associato e quindi partecipante attivo ai tavoli tecnici.

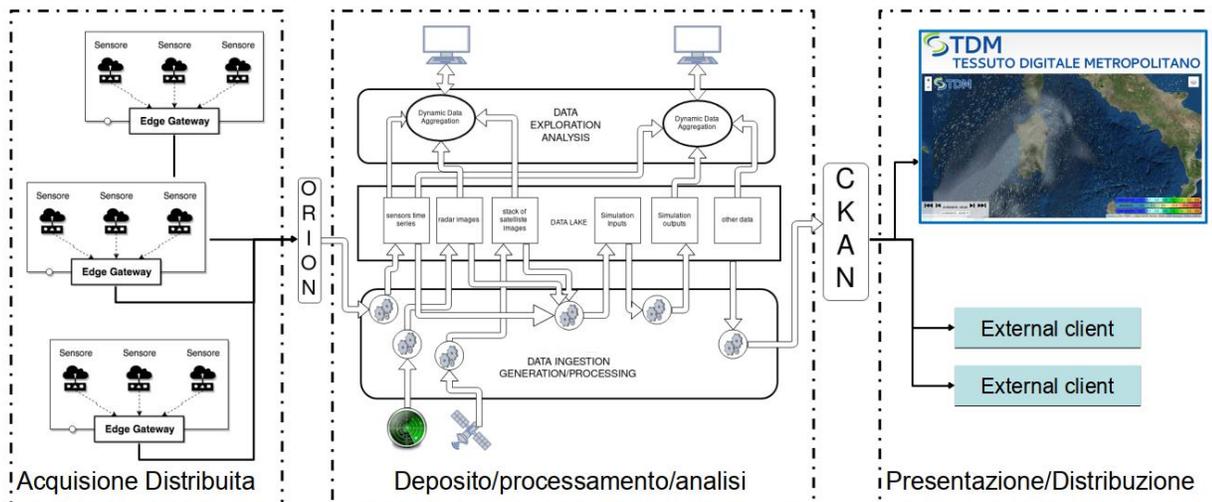
Per quanto riguarda l'infrastruttura, il progetto TDM adotta, sempre secondo le indicazioni di OASC, componenti standard della piattaforma FIWARE, quali, per fare un esempio, il Context Broker ORION, recentemente selezionato dalla Commissione Europea come uno dei componenti di riferimento per il CEF (Connecting Europe Facility), e il portale OpenData CKAN, adottato sempre dalla Commissione Europea e da molte altre pubbliche amministrazioni, come strumento per la pubblicazione di dati aperti.

Qui di seguito presenteremo lo stato attuale di sviluppo del portale di distribuzione dati, descrivendo l'architettura generale del portale, le tipologie di dati trattati e relativi formati, le modalità di accesso ai dati, l'integrazione del portale open data all'interno dell'ecosistema dei portali TDM e le prime applicazioni di accesso visuale sviluppate.

3. ARCHITETTURA DEL PORTALE DI DISTRIBUZIONE DATI

3.1 ARCHITETTURA GENERALE TDM

L'architettura generale di TDM è strutturata su tre grandi blocchi, dedicati, rispettivamente, all'acquisizione distribuita dei dati, al loro processamento e, infine, alla loro presentazione e distribuzione verso l'esterno.



Il primo blocco si preoccupa, sostanzialmente, di fornire gli strumenti necessari (edge device, standardizzazione su protocolli e formati dati FIWARE) per poter agganciare diverse tipologie di sensori (principalmente legati all'acquisizione di informazioni ambientali e consumi elettrici) e convogliarne i risultati in una forma standardizzata verso un context broker, ORION nel nostro caso specifico, che funge da punto di snodo per i flussi di dati acquisiti. Questo blocco è stato descritto in dettaglio dal punto di vista tecnico nel Deliverable D3.1.

Il secondo blocco accumula, processa e analizza dati provenienti da varie sorgenti che, oltre a quelli provenienti da sensori distribuiti includono i risultati di simulazioni meteo, immagini satellitari, immagini dal radar meteorologico, mappe e altri dati statici georeferenziati. L'integrazione di tutti questi dati avviene all'interno di un'architettura scalabile di aggregazione, processamento e restituzione esemplificata nella parte centrale della figura. Essa è basata su tre distinti sottoblocchi, due computazionali, e uno di storage. Il sotto blocco di storage è costruito come un data lake scalabile in cui vengono riposti sia i dati direttamente acquisiti dai sensori/generati dalle simulazioni e dalle successive elaborazioni. Il primo sottoblocco computazionale è dedicato alla realizzazione di operazioni sostanzialmente "automatiche" legate all'acquisizione e generazione di dati. In particolare, in esso vengono fatte delle elaborazioni in *stream* di processamento quasi real-time dei dati che provengono dai sensori ed elaborazioni più lunghe, *batch*, che richiedono più dati e risultati meno tempestivi. Esempio di elaborazioni batch sono le previsioni meteorologiche a breve periodo (*Nowcasting*), le analisi storiche dei consumi energetici, la creazione di modelli per la simulazione e la visualizzazione. L'altro sottoblocco computazionale è dedicato alla esplorazione dei dati, principalmente attraverso l'utilizzo di servizi JupyterHub integrati

con il data repository. Per una discussione più dettagliata del blocco di processamento ed analisi dati come anche delle scelte tecnologiche prese per realizzarlo, rimandiamo al futuro deliverable D3.3 che sarà disponibile al mese 24.

Il terzo blocco, quello a destra nella figura, è dedicato alla presentazione dei dati accumulati verso i cittadini come dati aperti e liberamente fruibili, o *OpenData*. Esso è descritto in maggior dettaglio nel resto di questo documento.

3.2 ARCHITETTURA DEL SOTTO-SISTEMA DI PRESENTAZIONE OPEN-DATA

Il sotto sistema di presentazione/distribuzione open-data è il punto di accesso ai dati accumulate e generati dal progetto TDM. Esso è realizzato attraverso una serie di diversi componenti integrati tra di loro, ognuno con caratteristiche e compiti dedicati. Tra di essi i principali sono il portale CKAN, attraverso cui si forniscono, principalmente, informazioni di indicizzazione sui dati disponibili e server di dati veri e propri per cui è previsto un accesso secondo un protocollo di tipo REST. Di seguito descriviamo CKAN e l'interfaccia REST prevista, La sezione 4 presenta dettagli sulle tipologie di dati trattati, e le seguenti esempi su come utilizzare le interfacce e i dati.

3.2.1 PORTALE CKAN

La pubblicazione dei dati sotto forma di Open Data avviene tramite il portale web open source CKAN (*Comprehensive Knowledge Archive Network*). CKAN è uno dei software di riferimento per la pubblicazione di Open Data sia istituzionali, quali i dataset di Comuni e Regioni, sia governativi, sia privati, essendo largamente utilizzato anche da aziende che mettono a disposizione della propria utenza informazioni relative al funzionamento dei servizi forniti e del lavoro svolto (trasporti, servizi al cittadino, *utilities* in generale). La scelta di CKAN viene suggerita quindi sia dal requisito di presentare ai cittadini una interfaccia di accesso e consultazione familiare e comune ad altri portali, sia dall'aderenza da parte del progetto TDM alle best practice per le Smart Cities adottate dall'iniziativa Open and Agile Smart Cities (OASC), alla quale il Comune di Cagliari, territorio nel quale si svolge il progetto, ha aderito sin dal 2015. Le API di CKAN, assieme alle API FIWARE-NGSI, sono infatti suggerite da OASC per l'implementazione della piattaforma di pubblicazione dei dati condivisi tra le Smart Cities dell'iniziativa¹.

Il portale CKAN è strutturato in diverse componenti integrate tra loro:

- Il *backend* (Python), che gestisce la parte grafica, l'integrazione tra le varie componenti, l'autenticazione e la pubblicazione delle API;
- il *database* (PostgreSQL), che contiene i metadati, i riferimenti e i dataset pubblicati (ove presenti);
- il *motore di ricerca* (Apache Solr), che provvede a interrogare il database in base ai dataset richiesti dall'utente, sia attraverso identificativi predefiniti (*tag, gruppi, organizzazioni*) sia attraverso ricerche *full-text*.

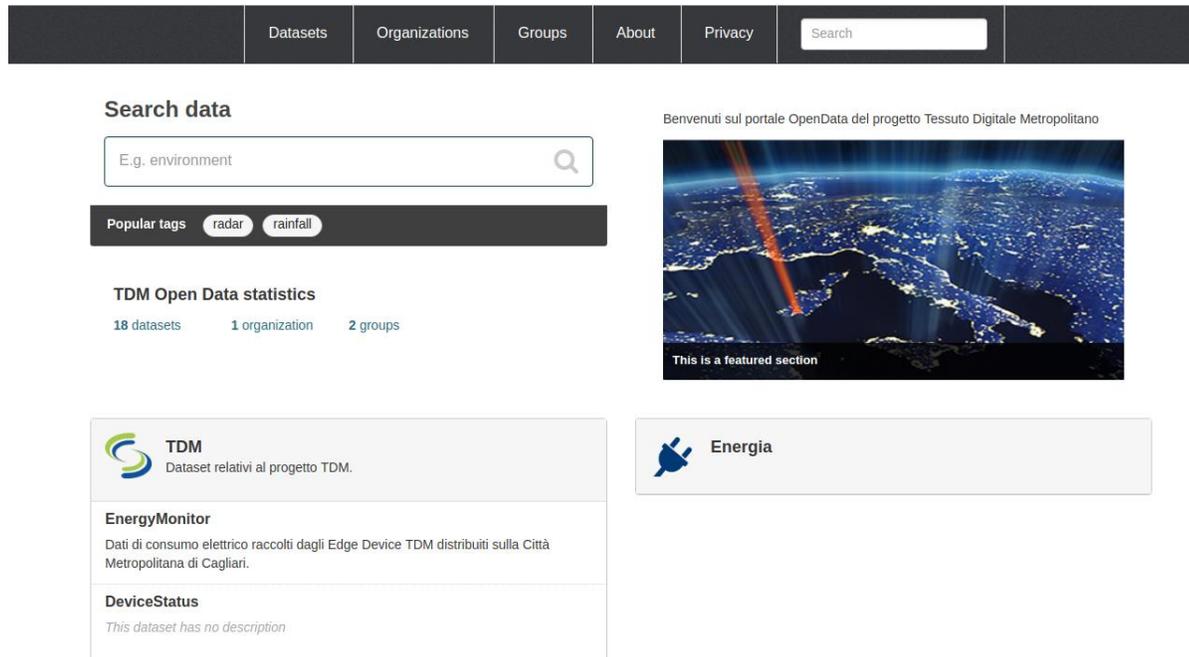
In particolare, il portale CKAN del progetto TDM è usato per indicizzare i dati raccolti e prodotti dal progetto TDM. Gran parte di questi non risiedono su CKAN stesso, essendo accessibili attraverso il portale *REST* specializzato nel fornire grossi moli di dati. CKAN contiene quindi i link a tali dati, raggruppati per *dataset* o *eventi* (nel caso dei dati delle simulazioni e

¹ "In addition, the cities will adopt Open Knowledge Foundation's CKAN as a shared open data and API publication platform. CKAN is already integrated and extended as part of the FIWARE Reference Architecture with add-ons easing the integration of the NGSI API to get access to context information.", Open & Agile Smart Cities, March 15, 2015, <https://oascities.org/open-agile-smart-cities/>

radar meteorologici) corredati di informazioni (metadati) utili al loro recupero e utilizzo, e permette di effettuare ricerche sul catalogo dei dataset da interfaccia web o tramite API.



TDM
TESSUTO DIGITALE METROPOLITANO



I dataset attualmente presenti sono relativi a:

- eventi meteo significativi - ciascun dataset contiene sia sequenze di immagini raccolte dal radar meteorologico che i risultati di simulazioni meteo dell'evento;
- sequenze di dati provenienti da sensori meteorologico/ambientali aggregati per ora e per ultimi 10 minuti;
- sequenze di dati energetico/elettrici aggregati per ora e per ultimi 10 minuti;
- sequenze di dati relativi al monitoring del funzionamento dei dispositivi Edge Gateway aggregati per ora e per ultimi 10 minuti.

3.2.2 ACCESSO AI DATI

In generale, ogni dataset pubblicato da TDM può essere scaricato direttamente, una volta individuato attraverso il portale CKAN, per la creazione di una copia locale a scopo di elaborazioni personalizzate. Mentre le sequenze di dati provenienti dalle misure di sensori sono, tipicamente, di piccole dimensioni e possono essere scaricati agevolmente, nel caso dei dataset di tipo volumetrico, come i risultati delle simulazioni meteo o le sequenze di acquisizione del radar meteorologico, il volume di dati da spostare è notevolmente superiore e, spesso, l'interesse è limitato solo ad una piccola parte dei dati

prodotti. Ad esempio, a fini di visualizzazione in un'applicazione esterna a TDM, può essere interessante accedere esclusivamente alle previsioni sulla pioggia in un data regione spaziale e un dato intervallo temporale senza doversi scaricare gigabytes di informazioni relativi a tutti gli altri campi prodotti dalla simulazione ma non rilevanti ai propri scopi.

Per poter supportare in maniera efficiente questa modalità di accesso ai dati abbiamo definito un protocollo REST che permetta, in maniera sistematica, un accesso casuale alle risorse.

Questo protocollo prevede che, una volta ottenuto l'identificativo della risorsa richiesta attraverso una query a CKAN, si possano accedere ai dati attraverso la costruzione di URI specifici a struttura gerarchica. Ad esempio, con l'URI seguente

<https://rest.tdm-project.it/tdm/odata/meteosim/moloch/2018101702/<uuid>/description.json>

si ha accesso ad una descrizione del dataset che può essere direttamente utilizzata dall'applicazione client per stabilire quali siano i campi disponibili, la regione spaziale e temporale in cui essi sono definiti e le relative risoluzioni. Con <uuid> indichiamo lo uuid che identifica la specifica istanza della simulazione.

La descrizione del dataset è sotto forma di un dato JSON con la seguente struttura:

```
{
  "description": {
    "group": "meteosim",
    "class": "moloch",
    "name": "2018101702",
    "uid": "61324998-ae0f-4a35-a966-0d7dab6ee512",
    "path": <URI to full dataset>,
    "history": [
      "Extracted by tdm_map_to_tree (V0.1)"
    ],
    "start_time": "2018-10-17_02:00:00",
    "end_time": "2018-10-19_00:00:00",
    "lon_range": "4.5:512:0.0226",
    "lat_range": "36.0:512:0.0226"
  },
  "result": {
    "resources": [
      {
        "name": "tcov",
        "description": "Total cloud coverage [percent]",

```

```
...  
},  
{  
  "name": "tprec",  
  "description": "Total precipitation [kg/m^2]",  
  ...  
}
```

...

Dalla consultazione di quest'ultima è possibile per un programma client di sapere quali sono i campi disponibili e in che range. L'accesso ai dati specifici viene poi fatto utilizzando URI più dettagliati. Ad esempio, con i seguenti URI si accede invece direttamente al campo di temperatura (scalare) a 2m dal suolo e di vento (componenti U e V della velocità del vento) a 10 metri dal suolo.

https://.../<uuid>/2018-10-17_02:00:00/4.5:512:0.0226_36.0:512:0.0226/temp2m.tif

https://...<uuid>/2018-10-17_02:00:00/4.5:512:0.0226_36.0:512:0.0226/temp2m.tif

Dove il primo selettore determina il timestamp richiesto (UTC) ed il secondo specifica la regione richiesta in coordinate GPS come una griglia regolare lon lat con i range definiti da <start>:<nsteps>:<step inc>.

La versione attuale dell'interfaccia ad accesso casuale è basata su ripartizioni dei dati generate automaticamente a priori, per opportuni sottoinsiemi di campi e risoluzioni spaziali e/o temporali. Nel corso del progetto è prevista un'evoluzione successiva caratterizzata da maggiore flessibilità, basata sulla generazione parzialmente o totalmente dinamica dei sottoinsiemi di dati.

4. TIPOLOGIE DI DATI TRATTATI E RELATIVI FORMATI

4.1 DATI PROVENIENTI DA MISURAZIONI PUNTUALI

Data la varietà dei dispositivi di misurazione che possono essere utilizzati per acquisire dati sul territorio, la *periferia* o *perimetro* del sistema, e l'inevitabile differenza tra formati di dato e messaggi adottati da produttori differenti, si pone il problema di una rappresentazione univoca dei dati esposti all'utente. Tale traduzione tra formati originali e formati standard è svolta dai software presenti nel *Edge Gateway*; questo, infatti, ridirige verso il Cloud TDM i dati raccolti sul campo dopo averli trasformati e accorpati.

Per i dati derivanti dai sensori, in ottemperanza alle linee guida OASC, sono stati adottati i modelli di dato armonizzati (Harmonized Data Models) FIWARE, ove presenti. Per i domini dei sensori per i quali non è presente un Data Model Armonizzato FIWARE o non è stato possibile identificarne uno adeguato tra quelli presenti, si è deciso di definirne di nuovi sul modello di quelli esistenti.

4.1.1 TIPOLOGIE DI EVENTI TRATTATI

I dati attualmente raccolti dalla periferia del sistema attraverso i sensori distribuiti appartengono principalmente a tre domini: il dominio Meteorologico / Ambientale; il dominio Energetico / Elettrico; il dominio del funzionamento di dispositivi embedded.

Nel primo rientrano, ad esempio, dati quali temperatura, umidità, pressione, quantità di particolato, intensità e direzione del vento; il dominio Energetico / Elettrico è rappresentato da dati di produzione e consumo di energia con parametri quali tensione, corrente e potenza usata o prodotta; l'ultimo comprende dati relativi al numero e funzionamento dei dispositivi di aggregazione periferica, gli *Edge Gateway*, utili a determinare lo stato complessivo della rete di misura e l'affidabilità delle misure stesse.

4.1.2 STANDARD NGSI

Facendo riferimento ai modelli di dato armonizzati definiti dalla piattaforma FIWARE-NGSI si è adottato, per quanto riguarda i dati di tipo Meteo/Ambientale trasmessi dagli Edge Gateway distribuiti, il modello *WeatherObserved*. Questo modello descrive le informazioni relative alla misurazione di parametri meteorologici osservati da una stazione meteo di uso comune

(<https://fiwaredatamodels.readthedocs.io/en/latest/Weather/WeatherObserved/doc/spec/index.html>).

Rappresenta quindi informazioni su:

- temperatura, umidità e pressione barometrica dell'aria;
- direzione e intensità del vento;
- quantità di pioggia.

Oltre a questi parametri, il progetto prevede l'acquisizione di informazioni sulla qualità dell'aria – quali percentuali di gas inquinanti e particolati presenti nell'aria – la cui rappresentazione non è prevista dalla versione standard del modello. La

soluzione adottata consiste nell'estensione di *WeatherObserved* con l'aggiunta di campi opzionali. Poiché l'assenza di uno o più campi opzionali non pregiudica la validità del messaggio, tale approccio consente di trattare anche le informazioni provenienti da stazioni meteo che non trasmettono tutti i dati previsti dal modello esteso.

Il modello *WeatherObserved* usato nel progetto contiene i seguenti campi (in **grassetto** i campi obbligatori, in *corsivo* i campi opzionali previsti nel modello originale, in sottolineato i campi aggiunti nella versione TDM):

- **id**: identificativo univoco della stazione;
- **type**: tipo del messaggio, deve essere "WeatherObserved";
- **location**: posizione della stazione meteo (GeoJSON), obbligatorio se address non è presente;
- **address**: indirizzo civico della stazione;
- **dateObserved**: data e ora dell'osservazione (ISO8601 UTC).
- *temperature*: temperatura dell'aria (numerico, gradi Celsius);
- *relativeHumidity*: umidità relativa dell'aria (numerico, percentuale);
- *precipitation*: quantità di pioggia (numerico, mm);
- *windDirection*: direzione del vento (numerico, gradi decimali Nord);
- *windSpeed*: velocità del vento (numerico, m/s);
- *atmosphericPressure*: pressione atmosferica (numerico, hPa);
- *illuminance*: illuminazione (numerico, lux o lumen/mq).
- timestamp: timestamp della creazione del messaggio (numerico, UNIX Epoch);
- infraredLight: luce infrarossa (numerico, misura relativa al sensore);
- CO: quantità di CO (numerico, misura relativa al sensore);
- NO: quantità di NO (numerico, misura relativa al sensore);
- NO2: quantità di NO₂ (numerico, misura relativa al sensore);
- NOx: quantità di NO_x (numerico, misura relativa al sensore);
- SO2: quantità di SO₂ (numerico, misura relativa al sensore);
- PM10: quantità di polveri PM 10 (numerico, misura relativa al sensore);
- PM2.5: quantità di polveri PM 2.5 (numerico, misura relativa al sensore).

4.1.3 ESTENSIONI PROPOSTE ALLO STANDARD NGS1

Per quanto riguarda i dati del dominio Energetico / Elettrico, non è stato possibile identificare un modello adeguato tra quelli inclusi nello standard. Si è quindi provveduto a creare un modello ad-hoc basato sulle linee guida FIWARE-NGSI, *EnergyMonitor*. Il modello è concepito per fornire una rappresentazione specifica per la produzione e il consumo di energia elettrica da parte di un utente, ma allo stesso tempo abbastanza flessibile da poter accogliere nuovi parametri che dovessero risultare necessari in futuro. Questo per far sì che, oltre a rappresentare in modo adeguato le informazioni pertinenti al progetto, il modello possa essere sottoposto alla comunità FIWARE per un'eventuale adozione.

Il modello *EnergyMonitor* è strutturato come segue (in **grassetto** i campi obbligatori, in *corsivo* i campi opzionali):

- **id**: identificativo univoco della stazione;
- **type**: tipo del messaggio, deve essere "EnergyMonitor";
- **location**: posizione della stazione meteo (GeoJSON), obbligatorio se address non è presente;

- **address**: indirizzo civico della stazione;
- **dateObserved**: data e ora dell'osservazione (ISO8601 UTC).
- **timestamp**: timestamp della creazione del messaggio (numerico, UNIX Epoch);
- **voltage**: tensione della linea elettrica (numerico, Volt);
- **current**: corrente che attraversa la linea elettrica (numerico, Ampere);
- **apparentPower**: potenza apparente assorbita dal carico (numerico, Volt-Ampere);
- **realPower**: potenza reale assorbita dal carico (numerico, Watt);
- **powerFactor**: fattore di potenza (numerico, tra 0 e 1);
- **consumedEnergy**: energia consumata (numerico, Wh);
- **frequency**: frequenza della linea elettrica (numerico, Hz).

4.1.4 DATI DI TELEMETRIA (EXTRA NGS)

Oltre a dati provenienti da sensoristica distribuita, per gli scopi del progetto TDM si è reso necessario avere dagli Edge Gateway ulteriori dati riguardo il funzionamento dei dispositivi stessi. Anche in questo caso, data la specificità del dispositivo Edge, tra i modelli di dato suggeriti da Fiware non si è potuto identificare un modello adeguato al caso e si è quindi provveduto a creare un modello ad-hoc basato sulle linee guida FIWARE-NGSI, il modello *DeviceStatus*. Tale modello è concepito per fornire una fotografia dello stato del sistema Edge Gateway dal punto di vista del funzionamento, dello stato di aggiornamento e dell'utilizzo di risorse di calcolo. Inoltre, potendo l'Edge Gateway ospitare dei sensori, sono presenti nel modello anche parametri quali *temperatura*, *umidità*, *illuminazione* e *pressione barometrica*. Trattandosi in questa fase del progetto di un prototipo soggetto a evoluzioni future, il modello *DeviceStatus* è stato definito lasciando aperta la possibilità di aggiungere ulteriori parametri. Come per il modello *EnergyMonitor*, il modello *DeviceStatus* potrebbe essere proposto in futuro alla comunità Fiware per la valutazione come precursore o modello dato armonizzato.

Il modello *DeviceStatus* è strutturato come segue (in **grassetto** i campi obbligatori, in *corsivo* i campi opzionali):

- **id**: identificativo univoco della stazione;
- **type**: tipo del messaggio, deve essere "Device Status";
- **location**: posizione della stazione meteo (GeoJSON), obbligatorio se **address** non è presente;
- **address**: indirizzo civico della stazione;
- **dateObserved**: data e ora dell'osservazione (ISO8601 UTC).
- **timestamp**: timestamp della creazione del messaggio (numerico, UNIX Epoch);
- *temperature*: temperatura dell'aria (numerico, gradi Celsius);
- *humidity*: umidità relativa dell'aria (numerico, percentuale);
- *illuminance*: illuminazione (numerico, Lumen);
- *atmosphericPressure*: pressione atmosferica (numerico, hPa);
- *lastBoot*: data dell'ultimo avvio (stringa);
- *operatingSystem*: sistema operativo (string, normalmente 'Linux');
- *kernelRelease*: release del kernel in uso (stringa);
- *kernelVersion*: build del kernel in uso (stringa);
- *systemArchitecture*: architettura del processore in uso (stringa, normalmente 'armv7l');
- *cpuCount*: numero di CPU/core in uso (numerico);

- *diskTotal*: capacità totale del disco (numerico, MB);
- *diskFree*: capacità disponibile del disco (numerico, MB);
- *memoryTotal*: capacità totale della RAM (numerico, MB);
- *memoryFree*: capacità disponibile della RAM (numerico, MB);
- *swapTotal*: capacità totale della Swap (numerico, MB);
- *swapFree*: capacità disponibile della Swap (numerico, MB).

4.2 DATI METEO E RADAR

Il progetto TDM produce e rilascia come dati aperti informazioni meteo-ambientali provenienti sia da sensoristica che da simulazioni. In particolare sono resi disponibili sul portale i dati provenienti dalla catena operativa di previsione meteo a scala limitata (LAM) ed i dati che provengono da un radar X-band che copre l'area metropolitana di Cagliari.

4.2.1 FORMATI E CONVENZIONI

i dati meteo e radar sono resi disponibili in due formati principali:

- NetCDF², allineati alle convenzioni CF³, per il batch download;
- GeoTIFF, per quelli destinati all'analisi di singole immagini o alla visualizzazione interattiva.

La griglia spaziale dei dati radar è disponibile sia nel sistema di coordinate originali (EPSG:3003 - Monte Mario / Italy zone 1) che in longitudine-latitudine (EPSG:4326 - WGS 84). Nella sezione 3.2.2 è già stato descritto come accedere ai dati. Qui viene discusso il formato dei dati stessi.

Le convenzioni CF⁴ definiscono uno standard per la strutturazione dei metadati nei file NetCDF contenenti dati meteorologici. Lo scopo principale è quello di localizzare i dati nello spazio e nel tempo e in funzione di altre variabili, facilitando l'elaborazione, la visualizzazione e il confronto tra variabili in dataset differenti. I metadati CF forniscono una descrizione autocontenuta (non dipendente da tabelle esterne) di ciò che i dati rappresentano, e sono strutturati in modo da essere facilmente leggibili sia da parte di un operatore umano che da un parser automatizzato. Le convenzioni CF sono state adottate da svariati gruppi di lavoro e sono riconosciute da numerosi software per l'elaborazione dei dati climatici.

I data set NetCDF messi a disposizione nell'ambito di TDM, sia quelli derivanti dal radar che le simulazioni meteo, seguono le convenzioni CF. Ad esempio, la variabile tempo è espressa in "seconds since yyyy-mm-dd HH:MM:SS", unità facilmente leggibile da un operatore umano e riconosciuta dai software che supportano le convenzioni. Per esempi più dettagliati, si rimanda all'appendice "esempi di utilizzo dei dati ed interfacce".

² <https://www.unidata.ucar.edu/software/netcdf/>

³ <http://cfconventions.org>

⁴ <http://cfconventions.org>



4.2.2 DATI RADAR X-BAND

Il progetto TDM rende disponibili come dati aperti le misure effettuate da un radar X-band posizionato presso il Dipartimento di Ingegneria Civile, Ambientale ed Architettura (Piazza d'Armi). Le principali caratteristiche del radar in oggetto sono descritte nella tabella seguente.

Table 2.1. X-band mini weather radar characteristics

Transmitter	
<i>Peak power</i>	10 kW
<i>Pulse length</i>	600 ns
<i>Transmitted frequency</i>	(9410 ± 30) MHz
<i>Pulse Repetition Frequency (PRF)</i>	800 Hz
<i>Modulator</i>	Solid state
Receiver	
<i>Receiver type</i>	Logarithmic, solid state
<i>Dynamic range</i>	From 0 to -100 dBm
<i>Intermediate Frequency (IF)</i>	60 MHz
<i>Receiver filter bandwidth</i>	(20 ± 2) MHz
Other specifications	
<i>Front end</i>	Microwave integrated circuit
<i>Noise figure</i>	< 3.5 dB
<i>Duplexer</i>	Ferrite circulator with state solid diode limiter
<i>Antenna rotation speed</i>	(22 ± 2) r. p. m.
<i>Power</i>	24 Vdc

Per scopi di nowcasting, il radar meteorologico è uno strumento fondamentale perché è in grado di monitorare lo spostamento delle idrometeore e quindi di facilitare la previsione della loro evoluzione a brevissima scadenza. Si vuole inserire questo strumento nel sistema di nowcasting di progetto utilizzando le infrastrutture di rete per la gestione in tempo reale dei dati misurati e la modellistica per assimilare e validare, anche mediante confronto con i dati pluviometrici misurati dalla rete fiduciaria della Regione Sardegna.

I data set derivanti dalle immagini radar contengono le misure effettuate in un raggio di ca. 30 Km centrato sulla Facoltà di Ingegneria dell'università di Cagliari (Piazza d'Armi). Il radar acquisisce un'immagine ogni minuto, e il dato, l'intensità della pioggia in mm/h, è rappresentato da una matrice 1024x1024 su una griglia regolare di circa 60Km di lato centrata sul radar stesso. La griglia spaziale è disponibile sia nel sistema di coordinate originali (EPSG:3003 - Monte Mario / Italy zone 1) che in longitudine-latitudine (EPSG:4326 - WGS 84).

Le immagini sono acquisite in maniera sistematica in caso di pioggia, e, nel caso di eventi significativi, sono raccolte in sequenze temporali in corrispondenza dell'evento. I dati delle sequenze sono poi messe a disposizione sia come dataset netCDF4, convenzioni CF, per permetterne il download batch che come GeoTIFF attraverso il protocollo REST descritto sopra.

4.2.3 DATI DELLA CATENA OPERATIVA DI PREVISIONE METEO A SCALA LIMITATA (LAM)

I dati di previsione meteo a scala limitata resi disponibili sul portale sono il risultato di una catena operativa che effettua successivi raffinamenti a partire da condizioni di contorno calcolate a scala globale.

Dati iniziali: modello globale GFS e scaricamento

Il Global Forecast System (GFS) è il modello di previsione meteorologica del National Centre for Environment Protection (NCEP, USA). Grazie alla simulazione effettuata, da NCEP, su base continuativa ogni 6h, questo modello rende disponibili a scala globale dati relativi a molte variabili atmosferiche e del suolo. L'intero globo è coperto con una risoluzione orizzontale di base di 28 chilometri per le previsioni fino a 8 giorni. La risoluzione orizzontale scende a 70 km per previsioni comprese tra una settimana e due settimane.

I dati in griglia sono disponibili per il download attraverso il NOAA National Operational Model Archive and Distribution System (NOMADS).

I dati del modello globale GFS sono la base per il funzionamento dei modelli di previsione meteorologica ad area limitata (LAM), come il BOLAM, perché forniscono le condizioni iniziali e al contorno necessarie per poter effettuare la simulazione. Con i LAM si possono così ottenere previsioni più dettagliate su porzioni limitate del globo.

Modelli: BOLAM e MOLOCH

BOLAM è un modello meteorologico idrostatico, che opera su un'area limitata del globo, sviluppato dal ISAC-CNR di Bologna. Le variabili prognostiche sono le componenti del vento, la temperatura, la pressione sulla superficie, l'umidità specifica e l'energia cinetica turbolenta. Il ciclo dell'acqua è descritto per mezzo di cinque ulteriori variabili prognostiche: le nubi di ghiaccio, quelle d'acqua, pioggia, neve e la grandine. Le variabili prognostiche definite nella griglia verticale hanno risoluzione maggiore nello strato limite atmosferico vicino alla superficie. La discretizzazione verticale è basata su un sistema ibrido di coordinate verticale, in cui le coordinate sigma terrain-following gradualmente tendono a una sistema di coordinate di pressione puro al crescere dell'altitudine. La discretizzazione orizzontale si basa su una griglia Arakawa-C sfalsata (Arakawa e Lamb, 1977), in coordinate geografiche (latitudine-longitudine). Lo schema di advezione attualmente implementato è WAF - Billet e Toro, 1997. Il regime di diffusione orizzontale è del secondo ordine per tutte le variabili prognostiche, tranne la pressione superficiale. Le condizioni al contorno laterali sono applicate su un numero limitato (tipicamente 8) di righe di punti di griglia, utilizzando uno schema di rilassamento (Leheman, 1993) che assorbe efficacemente l'energia delle onde, contribuendo a ridurre la riflessione spuria dai confini laterali. BOLAM ha la capacità di effettuare simulazioni annidate, cioè con condizioni iniziali e al contorno ottenute mediante integrazione dello stesso modello (chiamato convenzionalmente "run padre") ad una risoluzione più bassa. La massima risoluzione è limitata dalla approssimazione idrostatica e dalla parametrizzazione della convezione, tipicamente circa 6-8 km. Per scale spaziali più piccole, viene utilizzato il modello MOLOCH che non utilizza l'approssimazione idrostatica. L'intero codice di BOLAM è scritto in Fortran 90. È stato parallelizzato, applicando la tecnica "splitting domain", ed è compatibile con ambienti di elaborazione in parallelo mpich2 e OpenMP. Il modello viene utilizzato operativamente da varie agenzie nazionali italiane



e servizi meteorologici regionali, ed è stato impiegato nelle previsioni in tempo reale per conto del Dipartimento Nazionale della Protezione Civile

MOLOCH è stato sviluppato dall' ISAC-CNR di Bologna per fornire previsioni spazialmente dettagliate, rappresentando esplicitamente i fenomeni convettivi. Il modello MOLOCH integra il set completo delle equazioni per un'atmosfera non idrostatica e comprimibile utilizzando come variabili prognostiche la pressione, la temperatura, l'umidità specifica, le componenti della velocità verticale e orizzontale del vento e cinque specie di acqua condensata. La dinamica del modello è integrata nel tempo con uno schema implicito per la propagazione verticale delle onde sonore, mentre sono espliciti e time-split gli schemi per i restanti termini. L'avvezione è calcolata utilizzando lo schema euleriano WAF (Billet e Toro, 1997). La griglia verticale è distanziata esponenzialmente con l'altezza. La microfisica è trattata sulla base della parametrizzazione proposta da Drofa e Malguzzi (2004). I processi fisici che determinano la tendenza nel tempo dell'umidità specifica, le nubi acqua/ghiaccio e il precipitato acqua/ghiaccio sono divisi in "veloci" e "lenti". I processi veloci coinvolgono trasformazioni tra umidità specifica e nube, mentre quelli lenti coinvolgono la produzione e la caduta di pioggia/neve/grandine. La temperatura è aggiornata imponendo l'esatta conservazione dell'entalpia a pressione costante. La precipitazione viene calcolata con lo schema stabile e dispersivo backward/upstream in cui la velocità è funzione della concentrazione.

La catena operativa

BOLAM. A partire dai dati forniti dal modello globale GFS (Global Forecast System) a 0.25° di risoluzione, abbiamo definito un esperimento giornaliero di previsione dello stato dell'atmosfera attraverso l'uso del modello meteo ad aria limitata BOLAM. La catena è costituita da uno strumento per lo scaricamento quotidiano dei dati del modello globale GFS e da un esperimento in cui è stato usato un dominio centrato sull'Italia di 578 x 418 punti con passo di 0.075° e 60 livelli verticali. Può essere usata l'analisi delle 00, 06, 12, 18 per l'inizializzazione del modello, mentre le condizioni al contorno vengono fornite ogni ora per un totale di 48h. I tempi di esecuzione sono in media di 1h30m su un cluster linux in cui abbiamo usato 4 task nella medesima cpu su 2 cpu. La previsione può essere effettuata giornalmente alle 4 scadenze sinottiche.

MOLOCH. A partire dalla previsione fornita dal modello ad area limitata BOLAM a 0.075° di risoluzione abbiamo definito un esperimento giornaliero di previsione dello stato dell'atmosfera attraverso l'uso del modello non idrostatico MOLOCH. Il dominio è centrato sull'Italia con 1156 x 1154 punti con passo di 0.0114°, 60 livelli verticali e 7 livelli nel suolo. L'analisi è basata sull'output di BOLAM a 3h dall'inizio della previsione principale (ossia quella effettuata con BOLAM) mentre le condizioni al contorno sono state fornite ogni ora per un totale di 24h.

Dati ottenuti da simulazioni meteo

L'output post-processato delle simulazioni con il modello idrostatico BOLAM è costituito da 62 variabili che sono di seguito elencate con: l'acronimo utilizzato, la denominazione standard e la relativa unità di misura.

ACPCPsfc surface Convective Precipitation [kg/m²]
APCPsfc surface Total Precipitation [kg/m²]
ASNOWsfc surface Total Snowfall [m]
CAPEsfc surface Convective Available Potential Energy [J/kg]
CINsfc surface Convective Inhibition [J/kg]
DIST0C isotherm Geometric Height [m]
EPOT950mb 950 mb Pseudo-Adiabatic Potential Temperature (or Equivalent Potential Temperature) [K]
EPOT850mb 850 mb Pseudo-Adiabatic Potential Temperature (or Equivalent Potential Temperature) [K]
GUSTsfc surface Wind Speed (Gust) [m/s]
HCDCsfc surface High Cloud Cover [%]



HGT850m 850 mb Geopotential Height [gpm]
 HGT700mb 700 mb Geopotential Height [gpm]
 HGT500mb 500 mb Geopotential Height [gpm]
 HGT300mb 300 mb Geopotential Height [gpm]
 HPBLsfc surface Planetary Boundary Layer Height [m]
 ICECsfc surface Ice Cover [Proportion]
 ICETKsfc surface Ice Thickness [m]
 LAPRsfc surface Lapse Rate [K/m]
 LCDCsfc surface Low Cloud Cover [%]
 LFTXsfc surface Surface Lifted Index [K]
 LHFTLsfc surface Latent Heat Net Flux [W/m²]
 MCDCsfc surface Medium Cloud Cover [%]
 MTERHsfc surface Model Terrain Height [m]
 PRMSlmsl mean sea level Pressure Reduced to MSL [Pa]
 PVORT850mb 850 mb Potential Vorticity [Km²/kg/s]
 PVORT300mb 300 mb Potential Vorticity [Km²/kg/s]
 PVORT325K 325 K isentropic level Potential Vorticity [Km²/kg/s]
 RELV950mb 950 mb Relative Vorticity [1/s]
 RELV850mb 850 mb Relative Vorticity [1/s]
 RELV500mb 500 mb Relative Vorticity [1/s]
 RH700mb 700 mb Relative Humidity [%]
 RH500mb 500 mb Relative Humidity [%]
 RH2m 2 m above ground Relative Humidity [%]
 SHTFLsfc surface Sensible Heat Net Flux [W/m²]
 SOILW3cm 0.03 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW9cm 0.09 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW18cm 0.18 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW33cm 0.33 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW57cm 0.57 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW96cm 0.96 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW159cm 1.59 m underground Volumetric Soil Moisture Content [Proportion]
 TCDCsfc surface Total Cloud Cover [%]
 TCIVWsfc surface Total Column Integrated Water Vapour [kg/m²]
 TMPsfc surface Temperature [K]
 TMPprs (950 850 700 500 300) Temperature [K]
 TMP2m 2 m above ground Temperature [K]
 TMPhy60 60 hybrid level Temperature [K]
 TSOIL3cm 0.03 m underground Soil Temperature [K]
 TSOIL9cm 0.09 m underground Soil Temperature [K]
 TSOIL18cm 0.18 m underground Soil Temperature [K]
 TSOIL33cm 0.33 m underground Soil Temperature [K]
 TSOIL57cm 0.57 m underground Soil Temperature [K]
 TSOIL96cm 0.96 m underground Soil Temperature [K]
 TSOIL159cm 1.59 m underground Soil Temperature [K]
 UGRDprs (950 850 700 500 300) U-Component of Wind [m/s]
 UGRD10m 10 m above ground U-Component of Wind [m/s]
 UGRDhy60 60 hybrid level U-Component of Wind [m/s]
 VGRDprs (950 850 700 500 300) V-Component of Wind [m/s]
 VGRD10m 10 m above ground V-Component of Wind [m/s]
 VGRDhy60 60 hybrid level V-Component of Wind [m/s]
 VVEL700mb 700 mb Vertical Velocity (Pressure) [Pa/s]
 WEASDsfc surface Water Equivalent of Accumulated Snow Depth [kg/m²]

L'output post-processato delle simulazioni con il modello non idrostatico MOLOCH è costituito da 53 variabili che sono di seguito elencate con: l'acronimo utilizzato, la denominazione standard e la relativa unità di misura.

APCPsfc surface Total Precipitation [kg/m²]
 ASNOWsfc surface Total Snowfall [m]
 CAPEsfc surface Convective Available Potential Energy [J/kg]
 CINSfc surface Convective Inhibition [J/kg]
 DZDT850mb 850 mb Vertical Velocity (Geometric) [m/s]
 DZDT700mb 700 mb Vertical Velocity (Geometric) [m/s]
 DZDT500mb 500 mb Vertical Velocity (Geometric) [m/s]
 EPOTprs (950 850 700) Pseudo-Adiabatic Potential Temperature (or Equivalent Potential Temperature) [K]
 GUSTsfc surface Wind Speed (Gust) [m/s]
 HCDCsfc surface High Cloud Cover [%]
 HGT500mb 500 mb Geopotential Height [gpm]
 LAPRsfc surface Lapse Rate [K/m]
 LCDCsfc surface Low Cloud Cover [%]
 LFTXsfc surface Surface Lifted Index [K]
 LHFFLsfc surface Latent Heat Net Flux [W/m²]
 MCDCsfc surface Medium Cloud Cover [%]
 MTERHsfc surface Model Terrain Height [m]
 PRMSLmsl mean sea level Pressure Reduced to MSL [Pa]
 RHprs (950 850 700 500 300) Relative Humidity [%]
 RH2m 2 m above ground Relative Humidity [%]
 SHFFLsfc surface Sensible Heat Net Flux [W/m²]
 SOILW3cm 0.03 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW9cm 0.09 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW18cm 0.18 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW33cm 0.33 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW57cm 0.57 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW96cm 0.96 m underground Volumetric Soil Moisture Content [Proportion]
 SOILW159cm 1.59 m underground Volumetric Soil Moisture Content [Proportion]
 SPFHhy1 1 hybrid level Specific Humidity [kg/kg]
 TCDCsfc surface Total Cloud Cover [%]
 TCIWVsfc surface Total Column Integrated Water Vapour [kg/m²]
 TCONDold850mb 850 mb Total Condensate [kg/kg]
 TCONDold700mb 700 mb Total Condensate [kg/kg]
 TCONDold500mb 500 mb Total Condensate [kg/kg]
 TCONDold300mb 300 mb Total Condensate [kg/kg]
 TMPsfc surface Temperature [K]
 TMPprs (950 850 700 500 300) Temperature [K]
 TMP2m 2 m above ground Temperature [K]
 TMPhy1 1 hybrid level Temperature [K]
 TSOIL3cm 0.03 m underground Soil Temperature [K]
 TSOIL9cm 0.09 m underground Soil Temperature [K]
 TSOIL18cm 0.18 m underground Soil Temperature [K]
 TSOIL33cm 0.33 m underground Soil Temperature [K]
 TSOIL57cm 0.57 m underground Soil Temperature [K]
 TSOIL96cm 0.96 m underground Soil Temperature [K]
 TSOIL159cm 1.59 m underground Soil Temperature [K]
 UGRDprs (950 850 700 500 300) U-Component of Wind [m/s]
 UGRD10m 10 m above ground U-Component of Wind [m/s]
 UGRDhy1 1 hybrid level U-Component of Wind [m/s]
 VGRDprs (950 850 700 500 300) V-Component of Wind [m/s]
 VGRD10m 10 m above ground V-Component of Wind [m/s]
 VGRDhy1 1 hybrid level V-Component of Wind [m/s]
 WEASDsfc surface Water Equivalent of Accumulated Snow Depth [kg/m²]

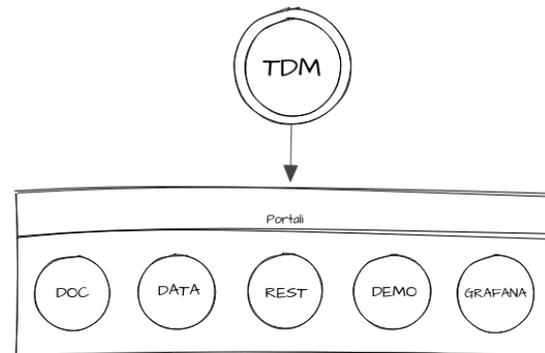
Tutte queste informazioni sono rese disponibili da TDM come open data accessibili attraverso la modalità di accesso per eventi. In appendice a questo documento è incluso un catalogo di eventi rilevanti attualmente disponibili.

5. L'ECOSISTEMA DEI PORTALI TDM

Il progetto TDM richiede una vetrina espositiva articolata su più fronti per soddisfare le diverse necessità di progetto. Pur essendo interfacce differenti e con modalità di fruizione distinte, possiamo definirle “portali” in quanto attraverso essi vengono veicolate i dati in varie modalità. Il portale open data descritto in questo documento è pienamente integrato, con si vedrà, in questo ecosistema.

L'ecosistema dei portali ha la seguente struttura:

- Il portale **DOC** è il landing site del progetto, più comunemente noto come **WWW**. Contiene aggiornamenti, descrizioni ed approfondimenti sugli aspetti cardine del progetto, documentazione e deliverable.
- **DATA** è il data source degli Open Data. Qui sono esposti i vari file che vengono prodotti dall'infrastruttura interna di calcolo del TDM o dalle varie sorgenti che vi confluiscono. Questo portale può essere usato come riferimento per gli indici verso i dati esposti dal portale seguente.
- **REST** è il data source dei dati processati. Da qui è possibile scaricare file e descrittori che definiscono un catalogo di dati
- **DEMO** è la vetrina del progetto. Qui sono ospitate le informazioni sui dati, varie demo, il visualizzatore e piccole guide su come accedere direttamente ai dati
- **GRAFANA** è il visualizzatore di dati realtime della sensoristica che alimenta il progetto



Tutti i portali citati sono raggiungibili abbinando il nome ad uno dei domini di progetto, ossia:

- tdm-project.it
- tdm-project.com
- tdm-project.org

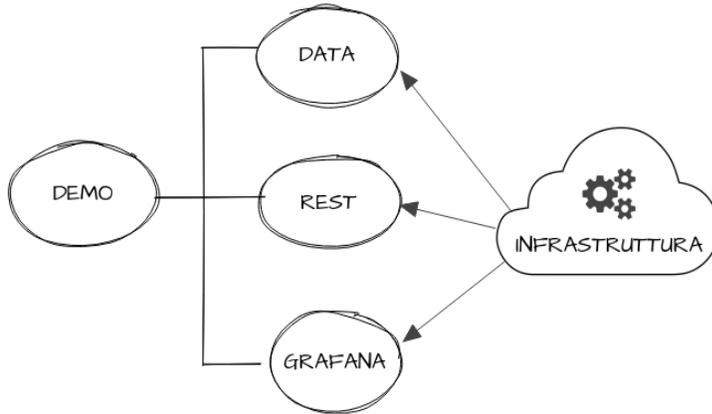
Partendo dal portale **DOC** è possibile raggiungere gli altri portali passando dalla sezione :

“Risultati” sotto la voce “Demo e software”.

5.1 PORTALE DEMO

Il portale demo è indipendente dagli altri portali ed è collocato a valle degli altri rispetto all'architettura interna, in questo modo i dati di cui usufruisce per i visualizzatori possono arrivare in maniera indipendente dall'infrastruttura interna.

Lo scopo del portale è quello di fare da contenitore per degli esempi di utilizzo e fornisce esso stesso un esempio di utilizzo dell'infrastruttura sottostante in quanto si alimenta con dati degli altri portali.



La fruizione del portale e' pubblica così come i pezzi di codice ospitati sono liberamente usabili.

Dal punto di vista estetico l'ossatura del portale **DEMO** ricalca quella del portale **DOC** già esistente, principalmente quindi con un header composto da logo di progetto seguito da una barra orizzontale con menu.

Sotto l'header lo schema prevede il container del tipo di informazione che occorre mostrare nella pagina.

Il footer e' costruito nel rispetto delle linee guida in tema di estetica per i progetti POR-FESR 2014-2020

In questa fase di progetto sono previste le seguenti voci di menu:

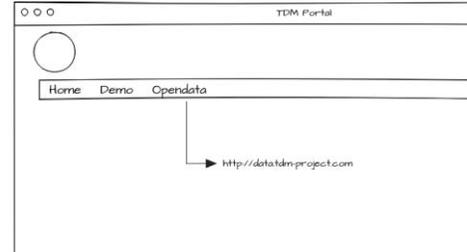
- Home
- Demo
- Opendata



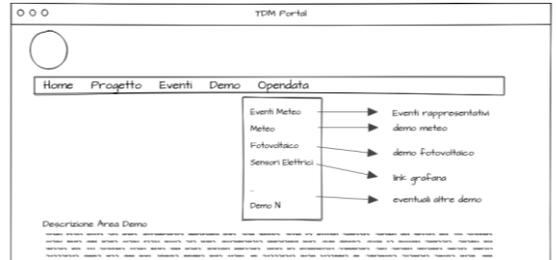
Nella pagina *HOME* il contenuto è descrittivo del portale, spiega come è articolato lo stesso e cita gli esempi che esso contiene.

Le informazioni di questa pagina sono di natura documentativa e non strettamente tecnica, può essere considerata come la landing page di presentazione.

Il menu contiene la voce Opendata che e' un rimando ad una pagina intermedia di presentazione per il portale **DATA**, dentro questa pagina c'e' una parte descrittiva su cosa siano gli Open Data ed il link al portale.



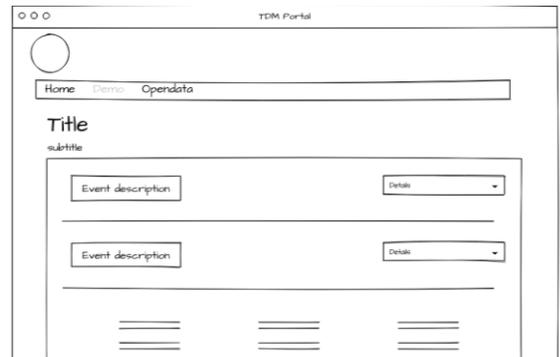
Sempre nel menu superiore si trova la voce "Demo", questo è un link verso una pagina in cui vengono elencati i principali casi dimostrativi sia con dei contenuti descrittivi per gli stessi.



Il menu è dropdown e presenta anche la tendina da cui è possibile raggiungere velocemente i singoli casi d'esempio, siano questi interni al portale o link ad applicazioni residenti esternamente.

In questa pagina ci sono anche i riferimenti a alcuni script che mostrano un possibile accesso ed uso dei contenuti del portale **DATA**.

La voce "Eventi meteo", più genericamente rappresentabile anche come "Eventi", apre una pagina con la lista degli eventi meteo di un certo rilievo che hanno interessato l'area di studio per la sperimentazione.



La pagina in questione è caricata direttamente usando come alimentazione i dati provenienti dal portale DATA ed effettuando una scrematura delle informazioni sulle risorse che ogni singolo dataset per evento mette a disposizione.

Da questa sezione è possibile passare al visualizzatore dell'evento correlato.

5.2 GRAFANA

Grafana è una piattaforma Web per la creazione di pannelli di indicatori di vario tipo per la visualizzazione grafica di parametri e grafici basati su serie storiche.

Grafana permette facilmente di rappresentare i dati provenienti dai sensori distribuiti attraverso heatmap, istogrammi, indicatori, grafici di vario tipo,

È possibile definire visivamente le regole di allarme per le metriche più importanti, queste vengono valutate continuamente ed il sistema Grafana potrà, se necessario, inviare delle notifiche via e-mail o tramite Slack, PagerDuty, VictorOps, OpsGenie, o via webhook.

Grafana vuole rappresentare una interfaccia unificata per la visualizzazione di dati da diverse sorgenti, come Graphite, Influxdb, Prometheus, Elasticsearch, AWS CloudWatch ed altri. E' altresì possibile realizzare visualizzazioni congiunte di dati provenienti da diverse sorgenti.

Il software è OpenSource ed è utilizzato, oltre che per il portale dei dati, anche all'interno degli Edge Gateway per la visualizzazione locale dei dati in essi memorizzati.

È inoltre possibile estendere i moduli di visualizzazione offerti attualmente con altri moduli dedicati, partendo dai moduli già presenti in formato aperto e modificandoli per meglio adeguarsi alle esigenze specifiche del progetto.

La sezione web Grafana del portale di progetto è pensata per due scopi. Il primo è quello di fornire ai visitatori un esempio delle possibilità di rappresentazione dei dati raccolti attraverso i sensori. Il secondo è quello di mostrare in maniera veloce e intuitiva lo stato del progetto, il funzionamento e la diffusione della rete di sensori e i dati da questi raccolti in versione grafica (a differenza di CKAN dai quali possono essere estratti anche in forma tabellare).

A regime gli indicatori presenti su CKAN mostreranno dati sul funzionamento della rete di sensori come:

- numero di *Edge Gateway* attivi sul numero totale installato
- numero di *stazioni meteo* sul totale installato
- numero di stazioni di monitoraggio elettrico sul totale installato
- disposizione come *heatmap* sul territorio
- valori medi, massimi e medie dei parametri misurati sul territorio
- Serie temporali delle misurazioni aggregate di consumi elettrici in ambito domestico
- Serie temporali delle misurazioni aggregate dei consumi elettrici di edifici
- Serie temporali della generazione elettrica tramite sistemi PV
- Serie aggregate dei dati di consumo e di generazione elettrica da fotovoltaico

La visualizzazione in Grafana è basata sui concetti di Dashboard e di Panel. Il Panel è il singolo strumento di rappresentazione dei dati, attraverso cruscotti, grafici, heatmap, o anche tabella. La dashboard costituita da un insieme di panel e da una visione di insieme dei dati analizzati. Sia le dashboard che i panels sono ampiamente configurabili dall'utente. Le dashboard sono condivisibili, e permettono agli utenti di condividere una rappresentazione comune dei dati.

Sul portale di progetto verranno messe a disposizione delle dashboard preconfigurate, a disposizione degli utenti per la visualizzazione dei propri dati sull'edge device.

A titolo di esempio, e suscettibile di sviluppi nel corso del progetto, nel portale è disponibile una dashboard con la visualizzazione dei consumi energetici della sede del progetto TDM. Si tratta di un impianto trifase che alimenta gli uffici con sia per il circuito di illuminazione, per i computer e le altre apparecchiature elettroniche, per la computer room del progetto ed il relativo impianto di raffreddamento.

La dashboard si compone di:

- una sezione cruscotti, in cui è mostrato il valore più recente rilevato per ciascuna fase dell'impianto, con una frequenza di campionamento di un minuto;
- una sezione con la potenza misurata per ciascuna delle linee nelle ultime ore, rappresentata in forma di line plot;
- una sezione con i consumi elettrici giornalieri cumulati per le ultime 2 settimane di misurazione;
- una sezione con un carpet plot della potenza delle tre linee per le ultime due settimane di misurazione (si tratta di un istogramma 2D, il colore rappresenta la potenza media nel quarto d'ora, l'asse x è diviso per giornate, l'asse y in funzione dell'ora del giorno);
- una sezione con la heat map dei consumi per le tre linee con bucket temporali pari ad un quarto d'ora, che evidenzia la variabilità dei consumi;
- una sezione con la rilevazione dei profili medi di consumo giornaliero nell'ultima settimana con risoluzione oraria.

Si veda la sezione seguente per ulteriori dettagli.



6. APPLICAZIONI DI ACCESSO VISUALE

Il portale TDM, nella versione attuale, include anche alcune applicazioni che permettono l'accesso ai dati aperti in maniera visuale. Queste applicazioni sono pensate sia per rendere direttamente fruibili i dati aperti disponibili sul portale, che per fornire esempi pratici di come sviluppare applicazioni attraverso il nostro accesso standardizzato ai dati. Queste applicazioni saranno aggiornate ed integrate con altri esempi nel corso del progetto.

Qui di seguito illustriamo l'architettura delle applicazioni di visualizzazione di dati provenienti da sensoristica e delle applicazioni di visualizzazione di eventi meteo.

6.1 VISUALIZZAZIONE DI DATI DA SENSORISTICA

I dati provenienti dalla sensoristica distribuita sul territorio attraverso gli *Edge Gateway* entrano nel cloud TDM attraverso la piattaforma per l'IoT messa a disposizione dall'ecosistema Fiware. Questa consiste nel broker MQTT, il componente che mantiene il canale di comunicazione tra Edge Gateway e Cloud TDM, l'*IoTAgent* Fiware che gestisce i tipi di dato in arrivo, e il *context broker Orion* che mantiene l'informazione su parametri ricevuti, dispositivi e contesto degli oggetti che scambiano dati. Tutte le componenti fin qui enunciate tuttavia non mantengono informazione storica sui dati che trattano, a meno di informazioni sulla loro configurazione, dato trattato, collegamento e, nel caso del context broker, l'ultimo dato trasmesso.

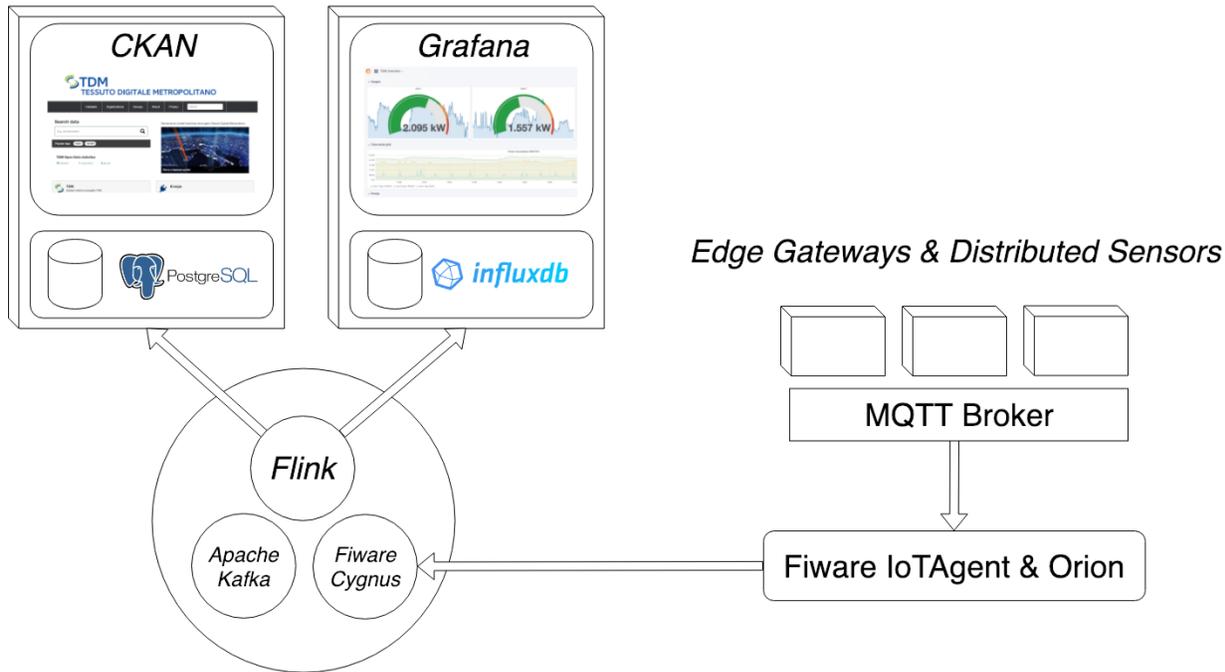
Per poter archiviare ed elaborare l'informazione, il context broker comunica i dati ricevuti alla sezione *Big Data* dell'infrastruttura che provvede ad archiviare ed elaborare i dati ricevuti. Questa sezione è anche la sorgente dei dati che vanno ad alimentare le due componenti di pubblicazione dei dati da sensore e Open Data: Grafana e CKAN.

L'elaborazione dei dati in arrivo dai sensori è implementata su Apache Flink⁵. Si tratta di un framework scalabile di ultima generazione, che consente il processing distribuito di dati sia di tipo batch che streaming, garantendo alto throughput e bassa latenza. Più task Flink in parallelo elaborano continuamente i dati in arrivo, ognuno con un tipo di messaggio sorgente, elaborazione specifica, e dataset di uscita.

A seconda del componente di visualizzazione questi dati diventano poi:

- per la pubblicazione OpenData, su CKAN, dataset giornalieri con dati aggregati quali medie orarie e medie dell'ultima ora;
- per la visualizzazione aggregata, record sul database di appoggio locale di Grafana, *InfluxDB*, contenenti dati riassuntivi pre-elaborati per la generazione di grafici, istogrammi e mappe.

⁵ <https://flink.apache.org>



L'interfaccia web di Grafana è modulare sia nei componenti da mostrare sia con il livello di permessi ed autorizzazioni per l'utente, in pratica si può scegliere la visualizzazione più idonea per la tipologia di utente specifico.

Di base l'interfaccia di Grafana risulta chiusa per gli utenti non autenticati quindi per gli scopi del progetto e' stata affidata alla landing page una dashboard complessiva in grado di dare una visione d'insieme dei dati dei sensori.

In questa fase si è scelto di mostrare i livelli di carico puntuale su tre linee elettriche, un componente per monitorare il carico sul medio periodo delle stesse linee, un componente per monitorare il carico storico complessivo delle linee e tre componenti per evidenziare le medie di carico per fasce orarie nelle singole linee.



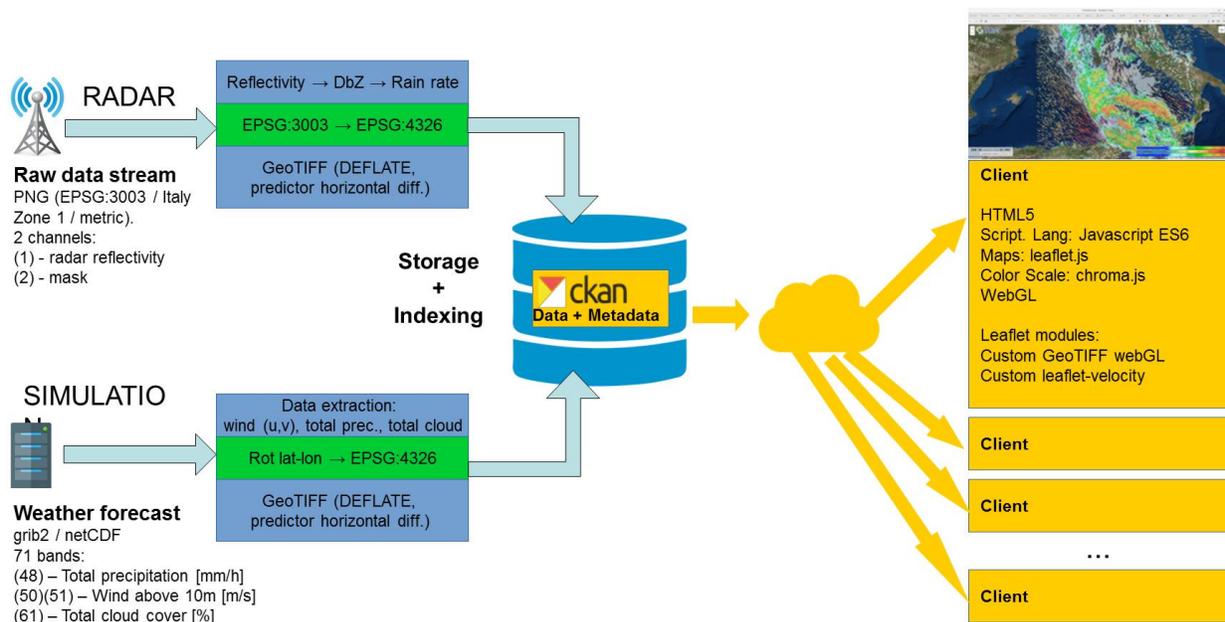
6.2 VISUALIZZAZIONE DI EVENTI METEO

Il visualizzatore integrato di eventi meteorologici è un esempio operativo di utilizzo dell'intera pipeline TDM che, attraverso le varie fasi a partire dalla produzione del dato, alla sua elaborazione e all'inserimento nella piattaforma open-source, consente la visualizzazione e il confronto di dati georeferenziati meteo-ambientali derivanti da simulazioni di previsione meteorologica o da misure da sensore (radar) relative a precipitazioni atmosferiche nella zona di Cagliari. I dati originari di simulazioni e misure sono automaticamente elaborati dal sistema TDM e resi disponibili sulla rete, per i fini della visualizzazione, in formati grafici georeferenziati di tipo standard.

Data l'attuale disponibilità di efficienti risorse grafiche nei moderni dispositivi mobili e personal computer, la scelta del framework di visualizzazione per la pipeline TDM si è naturalmente orientata verso il rendering client-side implementato in Javascript. Questa scelta, in linea con le tendenze attuali, consente di sfruttare moltissime librerie grafiche già realizzate ad-hoc per questi scopi e sviluppare soluzioni specifiche per i dati del sistema TDM.

6.2.1 FUNZIONALITÀ

La struttura della pipeline è illustrata alla figura seguente



La pipeline integra i dati da simulatore o da sensoristica passando attraverso l'interfaccia fornita da TDM CKAN descritta nelle sezioni precedenti.

Tutte le variabili misurate o simulate sono georeferenziate e tempo-varianti.

Le componenti del sistema di visualizzazione sono state progettate per le due tipologie di dati: campi scalari e campi vettoriali 2D. L'approccio scelto, basato su piattaforme altamente customizzabili, consente comunque di estendere in futuro le visualizzazioni ad altri tipi di dati.

Il progetto delle componenti di visualizzazione tiene quindi conto del tipo di grandezze (scalari o vettori), della loro localizzazione geografica e della loro variabilità nel tempo.

I dati georeferenziati sono rappresentati su mappe geografiche disponibili in formato aperto e multiscala sulla rete (tiled web map). Il sistema di visualizzazione è quindi a strati (layer): ogni layer rappresenta un tipo di dato da visualizzare, e tutti sono sovrapposti (con un certo ordine di disegno e livello di trasparenza) alla mappa geografica di base (es. OpenStreetMap). Il risultato è quindi una cartina geografica su cui sono rappresentate, con le tecniche opportune di sciviz, le varie grandezze da visualizzare.

I campi scalari sono rappresentati tramite colormap, ad ogni dato scalare posizionato sulle sue coordinate geografiche, compreso tra un valore massimo e minimo, è associato quindi un colore preso da una scala di colori continua. Tale mapping genera quindi delle strutture visuali utilizzate per migliorare la percezione dei dati e la loro interpretazione efficace. Sono uno dei modi più comuni per vedere i dati, e tra i più studiati in letteratura.

I campi vettoriali sono rappresentati tramite animated particle tracing: delle particelle, posizionate in modo casuale ed uniforme sul campo vettoriale, sono “trasportate” dal campo stesso lungo quindi le linee di flusso del campo. Ogni particella traccia una scia evanescente di lunghezza proporzionale al modulo del campo vettoriale. Anche il colore di ogni particella rappresenta il modulo del campo vettoriale (l’associazione tra colore e valore è definita da una colormap). Il risultato grafico è molto efficace e consente di valutare velocemente la presenza di perturbazioni del campo del campo vettoriale (come turbolenze o vortici) o valori di attenzione delle grandezze visualizzate.

6.2.2 IMPLEMENTAZIONE

L’implementazione del visualizzatore client-side, basata su Javascript, si appoggia su alcune librerie open-source standard:

- **Leaflet:**
Libreria Javascript per la gestione di mappe cartografiche interattive mobile-friendly.
URL: <https://leafletjs.com/>
Release: 1.3.1
- **geotiff.js**
Libreria Javascript per la lettura di numerosi tipi di file (Geo)TIFF e gestione di metadati geospaziali e raw data array
URL: <https://geotiffjs.github.io/geotiff.js/>
Release: 1.0.0-beta.3
- **twgl.js**
Libreria Javascript di utility per WebGL orientata alla programmazione a basso livello
URL: <https://twgljs.org/>
Release: 1.9.0
- **chroma.js**
Libreria Javascript per la gestione efficiente di colormap e conversioni tra spazi colore
URL: <https://github.com/gka/chroma.js/>
Release: 2.0.2

- **L.CanvasLayer.js**
Libreria Javascript di utility per la gestione di layer generici Leaflet in overlay (rel. > 1.0.0)
URL: <https://github.com/Sumbera/gLayers.Leaflet>
Release: 1.0.4
- **leaflet-velocity.js**
Leaflet plugin per l'implementazione di un canvas per il disegno di campi vettoriali
URL: <https://github.com/danwild/leaflet-velocity>
Release: 1.2.4

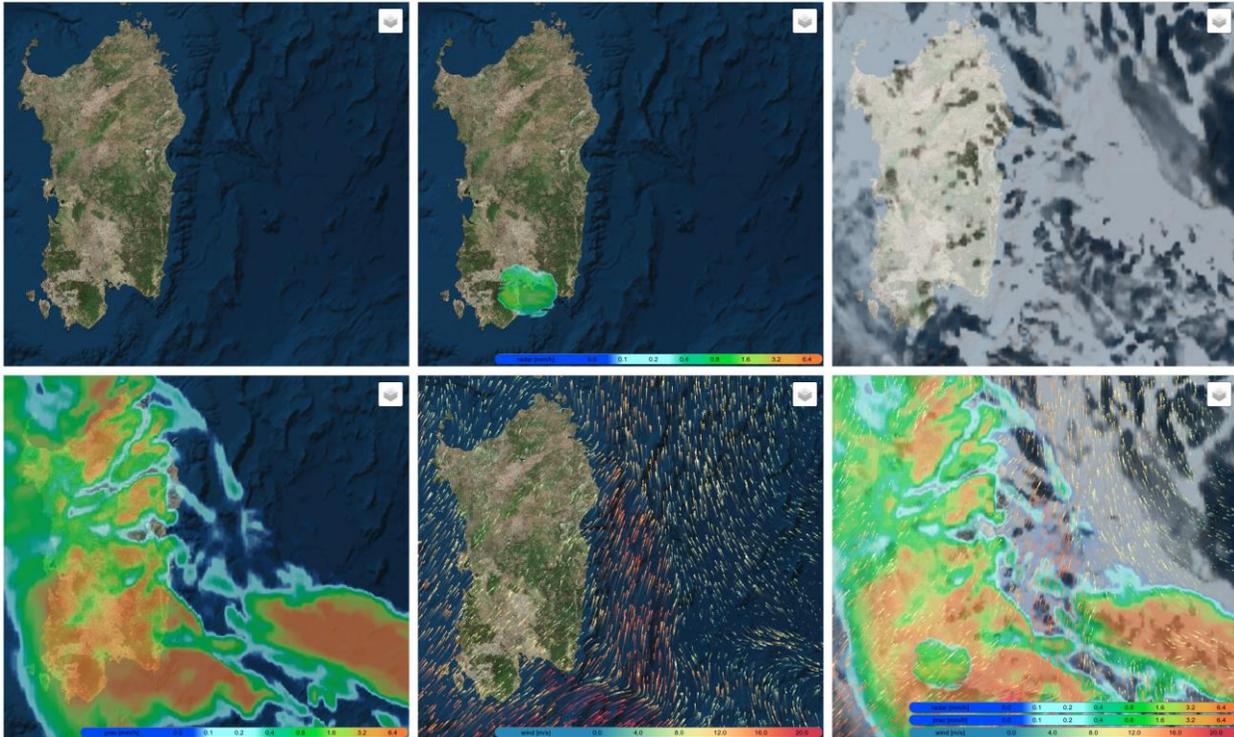
Le librerie Javascript sopra elencate sono state impiegate per lo sviluppo di moduli specifici di visualizzazione. In particolare sono stati realizzati:

- **tdm-leaflet-velocity**
Una versione modificata ed estesa della libreria leaflet-velocity.js in grado in particolare di trattare direttamente dati di campi vettoriali in formato GeoTIFF
- **L.TDMCanvasLayer.js**
Una variante di L.CanvasLayer.js integrata nella struttura del visualizzatore per la costruzione di layer leaflet GeoTIFF
- **tdm-basic-geotiff-layer.js**
Un'estensione di L.TDMCanvasLayer.js per la visualizzazione di campi scalari encodati in immagini GeoTIFF. Il layer supporta la definizione di valori soglia, coordinate geografiche, colormap, e vari livelli di opacità. La visualizzazione avviene tramite WebGL, utilizzando texturing e shaders.
- **Mutex.js**
Versione di un mutex per garantire la mutua esclusione di chiamate parallele. E' utilizzato nel codice per sfruttare al meglio il caricamento asincrono dei dati.

Sulla base di queste librerie è stato realizzato il visualizzatore dimostrativo di TDM. I dati meteo-ambientali attualmente gestiti da questo visualizzatore sono:

- **Dati da sensore (radar)**
Intensità di precipitazione [mm/h] (scalare)
- **Dati da simulazione**
Intensità di precipitazione totale [mm/h] (scalare)
Copertura nuvolosa totale [%] (scalare)
Temperatura a 2m [°C] (scalare)
Velocità del vento a 10m [m/s] (vettore 2D)

Nell'immagine seguente sono raffigurati dall'alto al basso, da sinistra a destra: il layer di base, il layer intensità di precipitazione derivata da dati radar, la copertura nuvolosa totale da previsione, l'intensità di precipitazione totale da previsione, la velocità del vento da previsione e la sovrapposizione di tutti questi layer in semitrasparenza.



Visualizzazione di alcuni layer meteo disponibili nel visualizzatore TDM.

Il visualizzatore, pienamente integrato nell'infrastruttura open data di TDM, è inizializzato fornendo il descrittore di evento oppure direttamente le due chiamate REST per l'accesso ai descrittori dei dati derivanti da simulazioni di previsioni meteorologiche e/o radar.

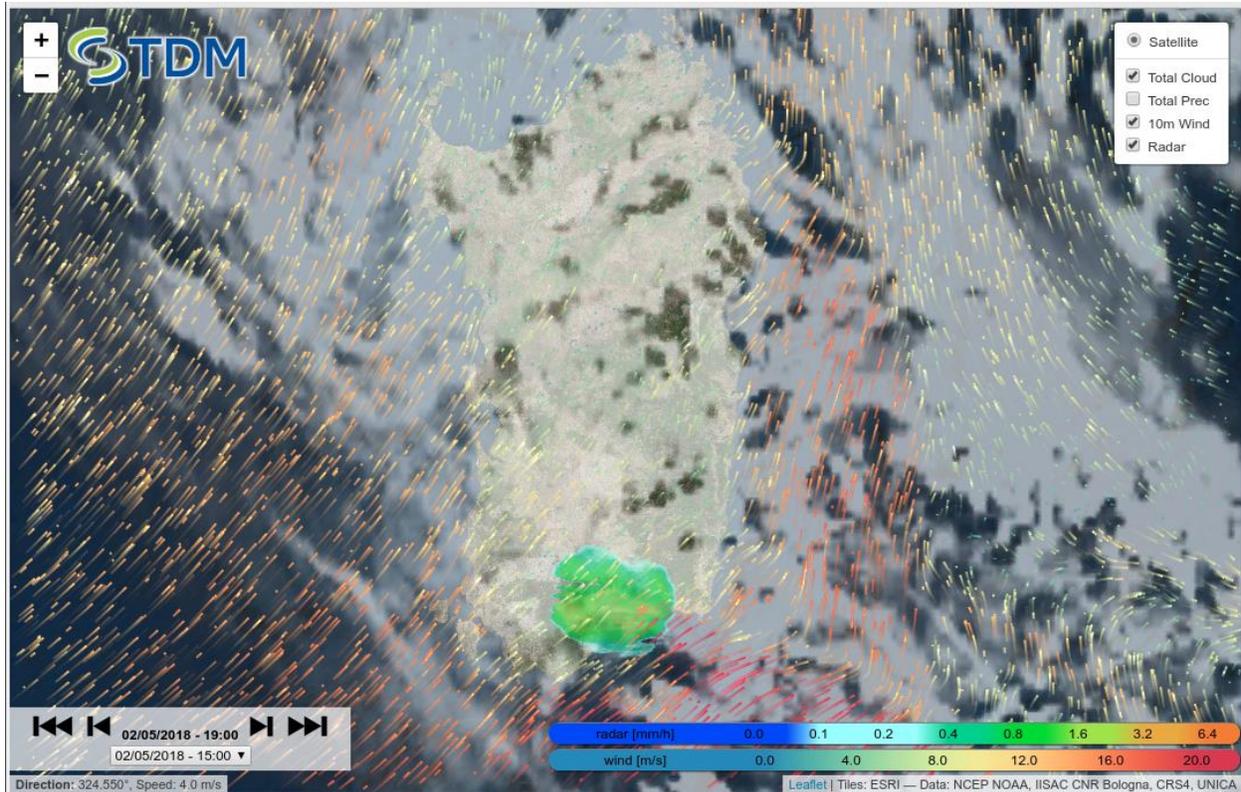
All'inizializzazione il visualizzatore analizza le informazioni ricevute e crea tutto quanto è necessario per la visualizzazione. In particolare:

- La descrizione dei singoli timestep con associati metadati (data, ora, tipologia di dato e URL per l'accesso diretto ai dati)
- La mappa cartografica di base (per default elaborazione ESRI da dati NCEP NOAA)
- L'interfaccia grafica per il controllo di Pan/Zoom, l'attivazione e disattivazione dei singoli layer, l'accesso temporale ai singoli timestep, la legenda relativa a tutti i layer attivi.

Per il timestep corrente, tutti i layer sono caricati in asincrono e visualizzati appena disponibili, in un definito ordine di priorità, in overlay sulla mappa di base.

6.2.3 INTERFACCIA GENERALE

Il visualizzatore di dati meteo-ambientali di TDM è un'applicazione web client-side che si presenta con l'interfaccia grafica raffigurata in figura



Visualizzazione di un evento meteo nel visualizzatore TDM

La finestra del visualizzatore presenta la mappa cartografica di base su cui sono disegnati i layer in overlay relativi a dati scalari (rappresentazione colormap) e vettoriali (particle tracing).

Il visualizzatore, progettato come responsive website, consente all'utente di interagire utilizzando il mouse o il touch screen di un mobile device. La mappa cartografica e i layer di dati possono essere traslati alla posizione desiderata tenendo premuto il tasto sinistro del mouse o muovendo un dito sul touch screen. Mappa e dati possono essere zoomati tramite la rotellina del mouse o il movimento di indice e pollice (pinch-to-zoom) tipico dei dispositivi mobili multitouch.

In alto a destra è posizionato il *selettore dei layer* diviso in due parti: i layer di base e i layer dei dati. E' possibile configurare il visualizzatore con più layer di base (mutualmente esclusivi) e con numerosi layer overlay (ognuno singolarmente attivabile dal selettore). Nell'esempio in figura è presente un solo layer di base (Satellite) e 4 layer dati (*Total Cloud*, *Total Prec*, *10m Wind*, *Radar*) di cui 1 (*Total Prec*) non attivato.

La *legenda delle colormap* è posizionata in basso a destra. Per ogni layer dati attivo vengono indicati la grandezza rappresentata, l'unità di misura e la scala di colori graduata impiegata per il mapping dei dati. Sotto la legenda delle

colormap sono indicate alcune informazioni sulle fonti dei dati. In basso a sinistra sono presenti altri due elementi grafici: il *selettore del timestep* e informazioni puntuali su direzione e velocità del vento nella posizione del cursore (mouse).

Il *selettore del timestep* consente di selezionare data e ora dei dati da visualizzare (scelti tra quelli di un evento definito dal descrittore della chiamata REST di inizializzazione del sistema).

Nel campo centrale del selettore del timestep è indicata la data e ora corrente a cui si riferiscono i dati visualizzati. Sono presenti quattro pulsanti (da sinistra a destra) per selezionare il primo step temporale della sequenza (evento), selezionare lo step precedente (rispetto all'attuale), selezionare lo step seguente (rispetto all'attuale) e selezionare l'ultimo step temporale. Per facilitare la selezione di un determinato step temporale è possibile scegliere direttamente un elemento da una drop-down list cliccabile nella parte in basso del selettore del timestep.

In alto a sinistra è presente un selettore di scala.

6.2.4 ESEMPI DI UTILIZZO

Le componenti del visualizzatore sono state sviluppate in modo da integrarsi completamente e semplicemente all'interno del framework Leaflet. Di seguito saranno illustrati alcuni esempi operativi di come un utente, accedendo alle risorse messe a disposizione dalla piattaforma di open data di TDM, possa implementare il proprio visualizzatore.

Tutti gli esempi di seguito descritti sono disponibili nel repository demo in: <https://github.com/tdm-project/tdm-tools>

I codici HTML e CSS di base sono comuni per tutti gli esempi.

HTML e CSS di base

Il file index.html deve contenere nella sua parte `head` gli include alle librerie JavaScript e agli stylesheet (CSS) locali o su rete (via CDN) utilizzati dal visualizzatore. In particolare: `leaflet.js`, `geotiff.js`, `chroma.js`, `twgl.js`, `tdm-leaflet-velocity.js`, `L.TDMCanvasLayer.js`, `tdm-basic-geotiff-layer.js`. Servono anche i riferimenti agli stylesheet `leaflet.css` e `tdm-demo.css` (il CSS dell'esempio corrente).

```
<head>
  <title>TDM SNIPPET</title>
  <meta charset="utf-8">
  <!-- ##### CDNs -->
  <link rel="stylesheet"
        href="https://unpkg.com/leaflet@1.3.1/dist/leaflet.css"/>
  <script src="https://unpkg.com/leaflet@1.3.1/dist/leaflet.js" >
  <script src="https://unpkg.com/geotiff@1.0.0-beta.3/dist/geotiff.bundle.js"/>
  <!-- ##### Plugins -->
  <script src="js/chroma.min.js" />
  <script src="js/twgl-full.js" />
```

```
<script src="js/tdm-leaflet-velocity.js" />

<script src="js/L.TDMCanvasLayer.js" />
<script src="js/tdm-basic-geotiff-layer.js" />
<!-- ##### CSS -->
<link rel="stylesheet" href="css/tdm-demo.css" />
</head>
```

La parte `body`, in accordo alle specifiche di Leaflet, include un riferimento al contenitore della mappa (nell'esempio il tag `<div>` con il selettore `id="map"`). Al fine di velocizzare le operazioni di disegno di tutti gli elementi HTML, è preferibile mettere l'include specifico del codice JavaScript dell'esempio nella riga finale, prima del tag di chiusura `</body>`.

```
<body>
  <div id="map"></div>
  <!--TDM Snippet-->
  <script src="js/tdm-demo.js" />
</body>
```

Come previsto nelle specifiche Leaflet, alcune caratteristiche del selettore `map` devono essere definite da uno `stylesheet`. Per questo esempio, queste definizioni sono contenute nel file `tdm-snippet.css`:

```
#map {
  position: absolute;
  z-index: 1;
  height: 100%;
  width: 100%;
  background-color: #333;
}
```

In particolare il selettore `map` deve indicare le dimensioni della finestra grafica (nell'esempio, l'intera pagina del browser al 100%) e sul colore di `background` (usato da Leaflet per colorare le `tile` temporaneamente mancanti durante gli aggiornamenti delle mappe, ad esempio durante le operazioni di `unzoom`).

Esempio 1: Visualizzatore di un dato scalare

Di seguito viene dettagliato quanto serve per visualizzare una mappa cartografica standard a cui è aggiunta in `overlay` una mappa dati (nell'esempio la "Precipitazione Totale" del 3/11/2018 ore 6:00 derivata da previsione meteorologica). I codici HTML e CSS sono quelli di base sopra descritti. Il visualizzatore `client-side` è essenzialmente caratterizzato dal codice Javascript in `tdm-demo-tprec.js`

JavaScript

Nell'esempio le istruzioni di inizializzazione sono impartite da una funzione `main()` eseguita da una chiamata `window.onload` che si attiva dopo che il browser ha caricato completamente tutti gli elementi grafici (HTML e CSS) e verificato l'assenza di errori. In un'applicazione pratica il `main` è alimentato da una chiamata REST alla struttura `open data`

TDM. Per facilitare la lettura dell'esempio, in questo caso il main ha come input l'URL di uno specifico GeoTIFF (estratto da una chiamata REST). Il main chiama quindi in sequenza una funzione di costruzione della mappa cartografica di base e degli elementi di controllo (`initBaseMap`), e una funzione sincrona di costruzione della mappa dati di overlay (`setupTiffLayer`). La variabile globale `mapStuff` è un oggetto che contiene un riferimento alla mappa Leaflet e alle strutture di controllo.

```
// Main
async function main(geotiff_url) {
    mapStuff=initBaseMap();
    await setupTiffLayer(geotiff_url);
}

// TPREC
window.onload = function() {
    main("https://rest.tdm-project.it/tdm/odata/product/meteosim/moloch/AAABBBBCCC3090/2018-11-03_06:00:00/4.5:512:0.0226_36.0:512:0.0226/tprec.tif");
};
```

A livello di variabili globali vengono definiti l'ordine con cui i layer di dati devono comparire nel selettore dei layer (nel visualizzatore). L'array `layer_order` e la funzione `compare_layers` servono a questo scopo. Per il dato sono definite l'ordine di disegno sulla mappa cartografica di base `tiff_layer_zindex` e le caratteristiche per il colormapping: valore minimo e massimo del range dei dati, la scala dei valori `tiff_domain`, la scala dei colori `tiff_scale`, il livello di trasparenza globale del layer `tiff_opacity`.

```
// GLOBAL VARIABLES

let mapStuff = null;
const base_satellite_name='Satellite';

const tiff_layer_name='Total Prec';
const tiff_layer_zindex=450;

const layer_order=[base_satellite_name,
    tiff_layer_name];

function compare_layers(A,B,nameA,nameB) {
    let indexA=layer_order.indexOf(nameA);
    let indexB=layer_order.indexOf(nameB);
    if (indexA < indexB) return -1;
    if (indexA > indexB) return 1;
    return 0;
}
```

```
}  
  
const tiff_min = 0;  
const tiff_max = 6.4;  
const tiff_domain = [0, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4];  
const tiff_scale = chroma.scale(['rgba(30,60,255,0.0) ',  
                                'rgba(160,255,255,1.0) ',  
                                'rgba(160,255,255,1.0) ',  
                                'rgba(0,210,140,1.0) ',  
                                'rgba(0,220,0,1.0) ',  
                                'rgba(160,230,50,1.0) ',  
                                'rgba(230,175,45,1.0) ',  
                                'rgba(240,130,40,1.0) ']).domain(tiff_domain);  
  
const tiff_opacity=0.6;
```

La funzione `setupTiffLayer` legge il file in formato GeoTIFF e assegna tutte le variabili necessarie al visualizzatore TDM `TDMBasicGeotiffLayer`. Tale operazione è incapsulata in una `Promise`, utile nel caso si vogliono visualizzare più layer di dati (le chiamate sono così eseguite in parallelo e la sincronizzazione viene fatta con un `await` di tutte le `Promise`).

La classe rappresentante il layer di dati delle precipitazioni `tiffLayer` viene aggiunta alla map di leaflet e alle strutture di controllo.

```
async function setupTiffLayer(tiff_url) {  
  let map = mapStuff.map;  
  let layerControl = mapStuff.layerControl;  
  return new Promise(async function(resolve, reject) {  
    if(tiff_url) {  
      const tiff_info= await getGeoTiff(tiff_url);  
      if(tiff_info) {  
        tiffLayer= new TDMBasicGeotiffLayer("TDM_TIFF", {  
          width: tiff_info.width,  
          height: tiff_info.height,  
          geobounds: tiff_info.bounds,  
          channels: tiff_info.data,  
          colorscale: tiff_scale,  
          opacity: tiff_opacity,  
          min_value: tiff_min,  
          max_value: tiff_max
```

```
});  
layerControl.addOverlay(tiffLayer, tiff_layer_name);  
    tiffLayer.addTo(map);  
    tiffLayer._canvas.style.zIndex = tiff_layer_zindex;  
    resolve("OK");  
} else {  
    reject(Error("CANNOT LOAD "+tiff_url));  
}  
} else {  
    resolve("OK");  
}  
});  
}
```

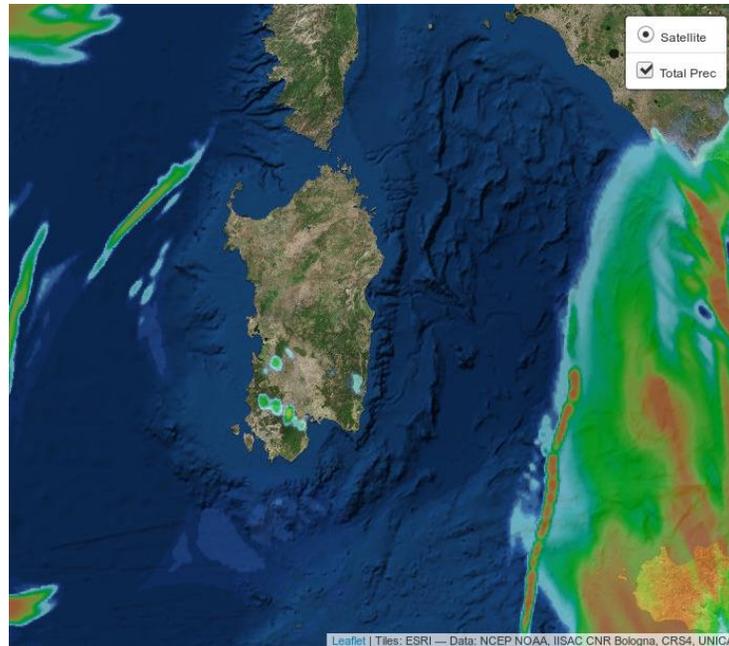
La funzione `getGeoTiff` è un'utility che chiama il loader della libreria `geotiff.js` e ritorna una struct con dati e informazioni accessorie.

```
async function getGeoTiff(url) {  
    let tiffInfo=null;  
    const response = await fetch(url);  
    const arrayBuffer = await response.arrayBuffer();  
    const tiff = await GeoTIFF.fromArrayBuffer(arrayBuffer);  
  
    let image = await tiff.getImage();  
    let rasters = await image.readRasters();  
  
    let tiffWidth = image.getWidth();  
    let tiffHeight = image.getHeight();  
    let tiepoint = image.getTiePoints()[0];  
  
    let pixelScale = image.getFileDirectory().ModelPixelScale;  
    let geoTransform = [tiepoint.x, pixelScale[0], 0, tiepoint.y, 0, -1*pixelScale[1]];  
    let imageBounds = [[geoTransform[3],  
                        geoTransform[0]],  
                      [geoTransform[3] + tiffHeight*geoTransform[5],  
                        geoTransform[0] + tiffWidth*geoTransform[1]]];  
  
    tiffInfo = {  
        width: tiffWidth,  
        height: tiffHeight,  

```

```
        bounds: imageBounds,  
        data: rasters  
    }  
    return tiffInfo;  
}
```

Il risultato finale di questo esempio dimostrativo è illustrato nella figura sottostante.



Visualizzazione di campo scalare (precipitazioni totali) nel visualizzatore TDM

Questo esempio è valido per la visualizzazione di tutti i layer dati scalari float che sono salvati in formato GeoTIFF nella struttura open data di TDM. L'esempio sopra descritto, relativo alle precipitazioni totali, è valido anche per la visualizzazione dei dati delle precipitazioni misurati da sensore (radar), basta semplicemente sostituire l'URL del GeoTIFF.

Esempio 2: Visualizzatore di un dato vettoriale

Di seguito viene dettagliato quanto serve per visualizzare una mappa cartografica standard a cui è aggiunta in overlay una mappa dati vettoriali (nell'esempio la "Velocità del vento a 10m" del 3/11/2018 ore 6:00 derivata da previsione meteorologica). I codici HTML e CSS sono quelli di base sopra descritti. Il visualizzatore client-side è caratterizzato dal codice Javascript in `tdm-demo-wind.js`

JavaScript

Il codice JavaScript mantiene la stessa filosofia di quello descritto nel paragrafo precedente inerente la visualizzazione di dati scalari. Si differenzia solo nella definizione delle colormap e nella chiamata del costruttore del layer di dati vettoriali `setupWindLayer`.

Le istruzioni di inizializzazione sono impartite da una funzione `main()` eseguita da una chiamata `window.onload` che si attiva dopo che il browser ha caricato completamente tutti gli elementi grafici (HTML e CSS) e verificato l'assenza di errori.

```
// Main
async function main(geotiff_url) {
    mapStuff=initBaseMap();
    await setupWindLayer(geotiff_url);
}

// WIND
window.onload = function() {
    main("https://rest.tdm-project.it/tdm/odata/product/meteosim/moloch/AAABBBBCCC3090/2018-11-03_06:00:00/4.5:512:0.0226_36.0:512:0.0226/uv10.tif");
};
```

Le variabili globali definiscono l'ordine con cui i layer di dati devono comparire nel selettore dei layer (nel visualizzatore). L'array `layer_order` e la funzione `compare_layers` servono a questo scopo. Per il dato sono definite l'ordine di disegno sulla mappa cartografica di base `wind_layer_zindex` e le caratteristiche per il colormapping: valore minimo e massimo del range dei dati e la scala dei colori `wind_scale`.

```
// GLOBAL VARIABLES
let mapStuff = null;
const base_satellite_name='Satellite';

const wind_layer_name='10m Wind';
const wind_layer_zindex=450;

const layer_order=[base_satellite_name,
                    wind_layer_name];

function compare_layers(A,B,nameA,nameB) {
    let indexA=layer_order.indexOf(nameA);
    let indexB=layer_order.indexOf(nameB);
    if (indexA < indexB) return -1;
    if (indexA > indexB) return 1;
    return 0;
}
```

```
const wind_min_speed = 0;
const wind_max_speed = 20; // 30 m/s right value
const wind_scale = chroma.scale(['#3288bd',
    '#66c2a5',
    '#abdda4',
    '#e6f598',
    '#fee08b',
    '#fdae61',
    '#f46d43',
    '#d53e4f']).domain([wind_min_speed, wind_max_speed]);
```

La funzione `setupWindLayer` legge il file in formato GeoTIFF e assegna tutte le variabili necessarie al visualizzatore TDM `L.velocityLayer`. Tale operazione è incapsulata in una `Promise`, utile nel caso si vogliono visualizzare più layer di dati (le chiamate sono così eseguite in parallelo e la sincronizzazione viene fatta con un `await` di tutte le `Promise`).

La classe rappresentante il layer di dati delle precipitazioni `velocityLayer` viene aggiunta alla map di leaflet e alle strutture di controllo.

```
async function setupWindLayer(wind_url) {
    let map = mapStuff.map;
    let layerControl = mapStuff.layerControl;

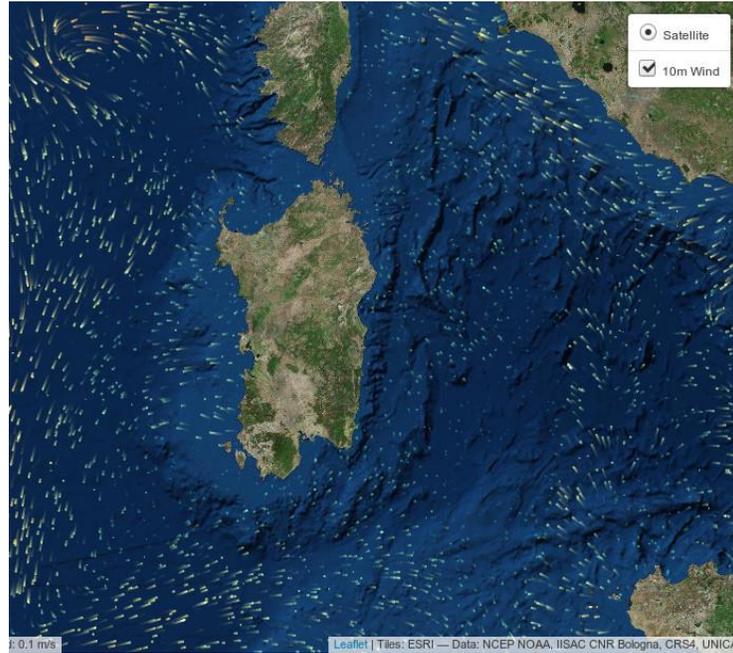
    return new Promise(async function(resolve, reject) {
    if(wind_url) {
        const wind_info= await getGeoTiff(wind_url);
        if(wind_info) {
            let cscale = [];
            for (let i = wind_min_speed; i <= wind_max_speed; i++) {
                cscale.push(wind_scale(i).hex());
            }

            let velocityLayer = L.velocityLayer({
                displayValues: true,
                displayOptions: {
                    velocityType: '10m Wind',
                    position: 'bottomleft', //REQUIRED !
                    emptyString: 'No velocity data', //REQUIRED !
                    angleConvention: 'bearingCW', //REQUIRED !
                    displayPosition: 'bottomleft',
```

```
        displayEmptyString: 'No velocity data',
        speedUnit: 'm/s'
    },
    width: wind_info.width,
    height: wind_info.height,
    geobounds: wind_info.bounds,
    data: wind_info.data,
    minVelocity: wind_min_speed,          // used to align color scale
    maxVelocity: wind_max_speed,          // used to align color scale
    velocityScale: 0.005,                 // modifier for particle animations
    colorScale: cscale, // define your own function of hex/rgb colors
    particleAge: 16,                       // default 64
    particleMultiplier: 1/200, // default 1/300 (particles/pixels);
    lineWidth: 2,                          // default 1
});
layerControl.addOverlay(velocityLayer, wind_layer_name);
velocityLayer.addTo(map);
velocityLayer._canvasLayer._canvas.style.zIndex = wind_layer_zindex;
resolve("OK");
} else {
    reject(Error("CANNOT LOAD "+wind_url));
}
} else {
    resolve("OK");
}
});
}
```

Analogamente all'esempio 01, la funzione `getGeoTiff` è un'utility che chiama il loader della libreria `geotiff.js` e ritorna una struct con dati e informazioni accessorie.

Il risultato finale di questo esempio dimostrativo è illustrato nella figura sottostante.



Visualizzazione di campo vettoriale (vento a 10m) nel visualizzatore TDM

7. APPENDICI

7.1 RICERCA DEI DATASET DISPONIBILI ATTRAVERSO IL FRONT-END CKAN

Tramite Il portale CKAN è possibile accedere ai contenuti disponibili secondo due modalità: con la navigazione libera dei dataset, eventualmente raggruppati per organizzazione o gruppo; tramite ricerca, sia full-text che basata su tag. La ricerca è raffinata filtrando tramite una collezione di tag, formati desiderati, tipi di licenza, gruppi e organizzazioni.

Le stesse funzionalità sono esposte tramite una REST API. I principali URL sono i seguenti:

- http://data.tdm-project.it/api/3/action/package_list produce la lista dei dataset
- http://data.tdm-project.it/api/3/action/package_search?q=QUERY effettua una ricerca full-text e restituisce i dataset pertinenti
- http://data.tdm-project.it/api/3/action/package_show?id=DATASET_ID restituisce le informazioni (metadati e risorse) relative ad un dataset
- http://data.tdm-project.it/api/3/action/resource_show?id=RESOURCE_ID restituisce le informazioni relative ad una risorsa
- http://data.tdm-project.it/api/3/action/resource_search?q=QUERY effettua una ricerca full-text e restituisce i dataset pertinenti

Ulteriori informazioni sulla REST API di CKAN sono reperibili alla pagina <https://docs.ckan.org/en/latest/api/index.html>.

Nel portale dimostrativo DEMO sono presenti due mini guide orientate ad utenti Python ed utenti PHP con degli esempi su come accedere facilmente ai dati esposti da CKAN.

Le guide mostrano come ottenere informazioni legate ai gruppi, ai tag ed ai dataset.

Per ragioni storiche legate allo sviluppo di CKAN nelle API REST i dataset sono chiamati packages, dal punto di vista concettuale i due termini coincidono.

Le guide sono accessibili dalla pagine riepilogativa denominata “Demo”, i link diretti sono

Python: http://demo.tdm-project.it/demo/esempio_notebook_python

PHP: http://demo.tdm-project.it/demo/esempio_notebook_php

Relativamente ai dati provenienti dai sensori, questi sono raggruppati per tipologia e memorizzati come risorse in dataset dedicati (uno per i sensori ambientali, uno per quelli relativi ai consumi energetici e uno per i dati relativi allo stato dei device). Le risorse sono un insieme di rilevazioni puntuali (o una loro media nel tempo) provenienti da sensori dello stesso tipo. Nello specifico, per ciascuna tipologia di sensore vengono create due tipi di risorse: quelle con i dati del giorno, di cui è stata calcolata la media per ogni ora; quelle con i dati relativi all'ultima ora, aggregati e mediati ogni dieci minuti.

Le risorse possono essere visualizzate in maniera tabellare, oppure su una mappa o ancora posso essere graficate come serie temporali.

Source: WeatherObserved

Data Explorer

Fullscreen Embed

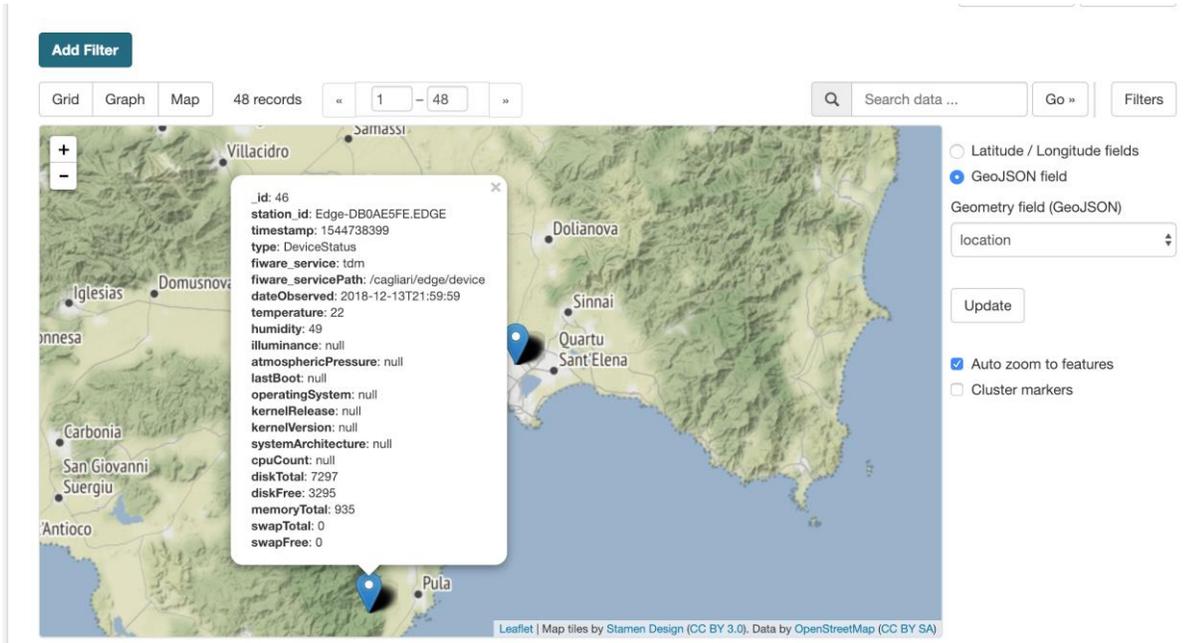
Add Filter

Grid Graph Map 32 records « 1 - 32 »

Q Search data ... Go » Filters

station_id	timestamp	location	dateObserved	PM10	PM2_5	barometricPre...	relativeHumidity	temperature
Edge-CFA703F4.esp8266-8955757	1544687...	39.2477...	2018-12-13...	1999.9	191.715	101473	39	24
Edge-CFA703F4.esp8266-8955757	1544713...	39.2477...	2018-12-13...	1879.2286	134.66492	100980	41	26
Edge-CFA703F4.esp8266-8955757	1544698...	39.2478...	2018-12-13...	1999.899	272.38962	101264	39	26
Edge-CFA703F4.esp8266-8955757	1544716...	39.2477...	2018-12-13...	1362.4625	60.888966	100937	41	26
Edge-CFA703F4.esp8266-8955757	1544702...	39.2477...	2018-12-13...	1999.899	234.03355	101159	39	26
Edge-CFA703F4.esp8266-8955757	1544720...	39.2478...	2018-12-13...	1309.0154	48.581085	100906	41	25
Edge-CFA703F4.esp8266-8955757	1544691...	39.2477...	2018-12-13...	1999.899	236.50299	101428	39	25
Edge-CFA703F4.esp8266-8955757	1544724...	39.2477...	2018-12-13...	1221.9902	45.46595	100886	41	25
Edge-CFA703F4.esp8266-8955757	1544727...	39.2477...	2018-12-13...	1093.6775	41.822727	100887	41	25
Edge-CFA703F4.esp8266-8955757	1544688...	39.2477...	2018-12-13...	1999.8992	215.07457	101470	39	25
Edge-CFA703F4.esp8266-8955757	1544731...	39.2477...	2018-12-13...	1076.1184	40.490818	100855	41	25
Edge-CFA703F4.esp8266-8955757	1544706...	39.2477...	2018-12-13...	1999.899	216.13298	101097	39	26
Edge-CFA703F4.esp8266-8955757	1544734...	39.2477...	2018-12-13...	997.41284	37.803112	100855	41	25
Edge-CFA703F4.esp8266-8955757	1544709...	39.2477...	2018-12-13...	1959.0028	136.1951	101036	40	26

Visualizzazione tabellare di una risorsa relativa ai sensori ambientali



Visualizzazione sulla mappa dei dati relativi allo stato di device

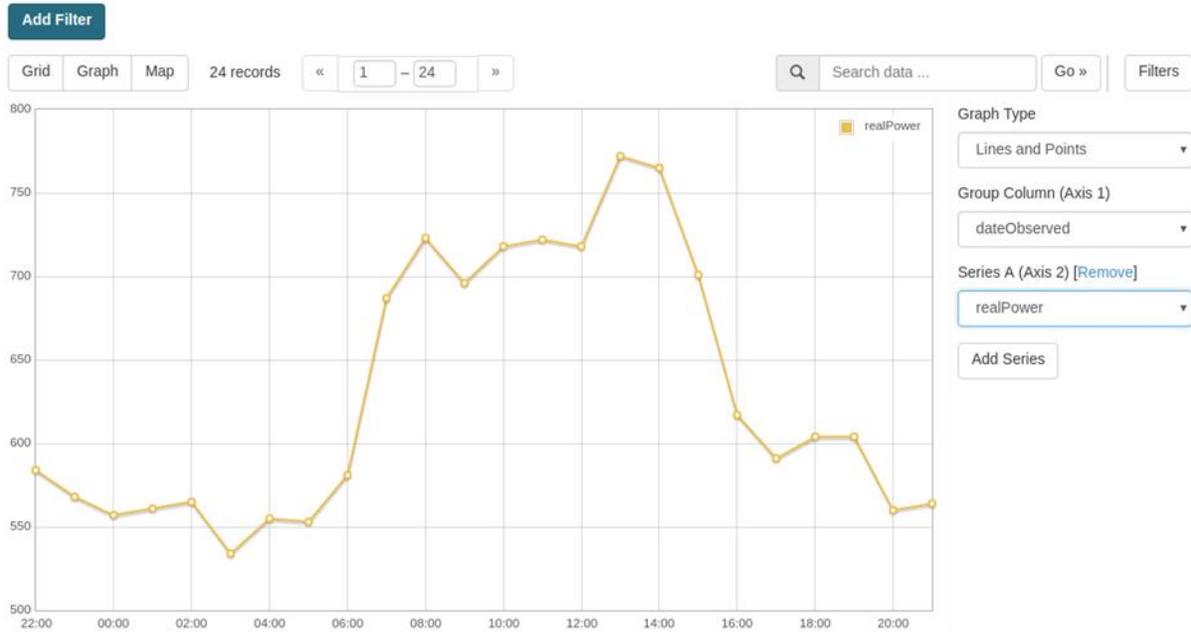


Grafico di una serie temporale relativi ad un sensore per la rilevazione dei consumi energetici

7.2 CATALOGO DATI METEO DISPONIBILI

In questa sezione diamo una descrizione di dettaglio di alcuni eventi inclusi nella release del portale open data alla data di redazione di questo deliverable. I dati sono quelli della situazione meteorologica relativa agli eventi di precipitazione intensa che hanno interessato l'area metropolitana vasta di Cagliari nel periodo maggio-novembre 2018. Per gli stessi eventi sono state eseguite le previsioni meteorologiche con la catena operativa descritta in precedenza e per i quali si dispone anche dei dati misurati dal radar meteorologico. Questo in prospettiva dell'utilizzo congiunto dei dati del radar e dei modelli meteorologici con l'intenzione di indagare la possibilità di un miglioramento della qualità della previsione a breve (~ora) scadenza.

Inoltre, sono descritti anche due eventi di ondata di calore, e per i quali sono state eseguite le relative simulazioni meteo, che si ritengono di interesse per gli scopi del progetto. I dati sono rilasciati in formato aperto e sono pienamente disponibili per l'utilizzo.

7.2.1 EVENTO DEL 2 MAGGIO ALLERTA ROSSA PER RISCHIO IDRAULICO

Nei primi giorni di maggio il Mediterraneo è stato attraversato da un ciclone che dal giorno 1, posizionandosi a ovest della Sardegna ha fatto affluire aria caldo umida da sud su tutta la verticale. Il ciclone il giorno 2 si è spostato a sud est e poi a nord continuando a generare instabilità sulla Sardegna. Questa configurazione ha generato precipitazioni, con cumulati anomali per il periodo (soprattutto il 2 e 3 maggio) e che sono proseguite nei giorni successivi anche se con cumulati decisamente inferiori. Le precipitazioni hanno colpito uniformemente tutta la Sardegna con cumulati decisamente inferiore nel Sud-Est. Il giorno 1 l'accumulo massimo di precipitazione in 24h si è avuto a Siniscola mentre il giorno 2 è stato il Sulcis ad essere colpito maggiormente con accumuli a Sant'Anna Arresi di 96 mm giornalieri. Dal giorno 3 assistiamo ad una decisa diminuzione della precipitazione su tutta l'isola. Per quanto riguarda la stazione di Cagliari il primo maggio la precipitazione risulta contenuta a 23 mm, quasi tutta tra le 15 e le 21. Il giorno 2 su Cagliari in 24 h cadono circa 52 mm distribuiti su tutta la giornata con un intensità massima di 3.4 mm in 15 min alle 7.

7.2.2 LE PRECIPITAZIONI INTENSE DEL MESE DI OTTOBRE

L'analisi delle serie storiche di precipitazione in Sardegna mostra che ottobre è il mese che presenta il numero massimo di eventi estremi spesso causa di disastri e perdite di vite umane. In particolare gli accumuli di precipitazione giornaliera dell'ottobre del 2018 nelle stazioni della zona Sud della Sardegna sono stati tra i più abbondanti degli ultimi quaranta anni e se consideriamo questo mese in termini di numero di giorni di pioggia un giorno su due dell'intero periodo è stato piovoso, almeno nel Sud.

Assolutamente eccezionali sono state, inoltre, le precipitazioni del 10 ottobre per l'effetto combinato dell'alluvione che ha colpito l'area vasta di Cagliari e in particolare Capoterra e dei nubifragi che hanno investito il Sud-Est.

La scelta sugli eventi, sui quali approfondire l'analisi con simulazioni a bassa ed alta risoluzione è quindi caduta su tre eventi verificatisi tra ottobre e gli inizi del mese di novembre, rispettivamente il 10, 18 ottobre e 3 novembre. Tutti questi eventi sono stati considerati dalla protezione civile di allerta: di grado massimo cioè rossa quella del 10 ottobre per rischio idrogeologico e di grado inferiore cioè arancione gli eventi del 18 ottobre e del 3 novembre. Per quanto riguarda l'evento

del 18 che è stato il meno importante dei 3, l'allerta è stata emanata anche in considerazione dello stato di imbibizione dei suoli che risentivano ancora fortemente dell'evento del 10

7.2.3 EVENTO DEL 10/10 RISCHIO IDROGEOLOGICO ELEVATO E IDRAULICO MODERATO PER LA ZONA DI CAGLIARI

Tra 9 e 11 ottobre si approfondisce un minimo barico sulle Baleari, che si muove molto lentamente verso la Penisola Iberica, frenato, a nord e ad est, da un campo di alta pressione che coinvolge Europa centrale e centro-orientale. La Sardegna viene interessata da una ventilazione calda ed umida da sud-est che produce precipitazione a carattere locale già dalla giornata del 9. Le giornate del 10 e 11 sono quelle in cui le precipitazioni raggiungono livelli elevati o molto elevati soprattutto nel Sulcis-Iglesiente, nella Quirra e nel Sarrabus, zone maggiormente esposte agli effetti della ventilazione umida di Scirocco.

La stazione pluviometrica di Santa Lucia di Capoterra misura 437 mm di pioggia in 24h di cui 82 mm in 1h, quella di Capoterra 218 mm di cui 86 in 1h, quella di Decimomannu 136 mm di cui 46 in 1h, quella di Villaspeciosa nell'Iglesiente 183 mm di cui 56 in 1h.

Per la stazione di Santa Lucia di Capoterra è il record dalla sua installazione. Ma non dobbiamo dimenticare che spesso in questa zona della Sardegna sono stati registrati cumulati confrontabili con precipitazioni orarie che il 22 ottobre del 2008 hanno raggiunto i 150 mm/h. Per quanto riguarda la stazione di Cagliari i cumulati sono stati di 34 mm per il giorno 10 e di 58 mm per il giorno 11 con un'intensità max di 9 mm in 15 minuti alle 7 di mattina.

7.2.4 EVENTO DEL 18/10 RISCHIO IDROGEOLOGICO MODERATO/ELEVATO PER LA ZONA DI CAGLIARI

A partire dal 13 ottobre una nuova perturbazione atlantica si muove verso la penisola Iberica. Questa viene costretta a dirigersi verso il Mediterraneo Occidentale dal contemporaneo espandersi verso nord-est dell'Anticiclone delle Azzorre sull'Europa occidentale e centrale. La circolazione depressionaria sul Mediterraneo produce sulla Sardegna precipitazioni moderate o intense tra il 14 e il 19 ottobre. La stazione con cumulato più elevato in 24 h è Tempio Pausania con 89 mm di cui 44 in 3h per il giorno 18, mentre il 19 la stazione più piovosa è stata Gairo con 23 mm

Le stazioni pluviometriche di Cagliari misurano 14 mm il 18 con un'intensità massima di quasi 7 mm in 15 mm alle 6 e non misurano alcuna precipitazione il giorno 19.

7.2.5 EVENTO DEL 03/11 RISCHIO IDROGEOLOGICO MODERATO PER LA ZONA DI CAGLIARI

Il primo novembre un minimo barico sull'Islanda che si estende fino al Mediterraneo occidentale è in lento movimento verso est producendo tra il 2 e il 3 un minimo barico sul mar di Sardegna nord occidentale. Lo spostamento verso est di tale perturbazione ha prodotto tra la giornata del 3 novembre e quella del 4 precipitazioni intense su tutta la Sardegna che localmente sono state di livello intenso o molto intenso. La stazione con cumulato di precipitazione più alto per il giorno 3 è stata Urzulei con 73 mm, con 22 mm in 1 h, mentre il 4 è stata Santu Lussurgiu con 78 mm in 24 h e 51 mm in 3h. Le stazioni pluviometriche di Cagliari hanno misurato 13 mm il 3 e 39 mm il 4 novembre con circa 30 mm di precipitazione tra le 7:45 e le 10:45 e un'intensità massima di 5.8 mm in 15 minuti dalle 9:30.



7.2.6 ALCUNE ONDATE DI CALORE DI LUGLIO 2018 NELLA STAZIONE DI CAGLIARI

Il mese di luglio 2018 è stato piuttosto caldo con temperature medie che hanno superato i 30°C su tutte le stazioni non montane della Sardegna. Le giornate più calde sono state il 13 e il 30 luglio. Usando come indice biometeorologico il THI (Temperature Humidity Index) che valuta il disagio per l'uomo con una scala che va da lieve ad alta si è verificato che per tutta la Sardegna i giorni del 13 e 30 luglio, tranne le zone montane, hanno avuto un THI tra medio e alto. Queste condizioni se permangono per almeno 14 ore giornaliere per 3 giorni consecutivi danno luogo a ciò che viene definito ondata di calore. Quindi, secondo questa definizione, sulla Sardegna sono arrivate in luglio almeno 2 ondate di calore una il 13 e l'altra il 30. Per quanto riguarda la stazione di Cagliari su 31 giorni di luglio: 7 sono stati di disagio lieve per 15/17 h, 18 sono stati di disagio medio per 18/20 h, 6 sono state di disagio alti per 24h.

7.2.7 EVENTO DEL 13 LUGLIO.

A partire dal 12 luglio una perturbazione si è estesa dal Nord sino alla penisola Iberica, spostando l'anticiclone delle Azzorre e creando un flusso di aria calda meridionale verso la Sardegna; la situazione è rimasta stabile fino al 17. In questo evento l'ondata di calore è iniziata il 13 luglio con THI medio nella stazione di Cagliari che è durato 18 h, ed è continuata per 3 giorni fino al 16 con THI alto per tutte le 24h

7.2.8 EVENTO DEL 30 LUGLIO.

Dal 24 luglio l'anticiclone delle Azzorre si è esteso sul Mediterraneo causando sulla Sardegna un lento afflusso di aria calda che ha raggiunto il suo apice il 30 luglio. In questo evento l'ondata di calore nella stazione di Cagliari è iniziata il 30 luglio con una durata di 20 h con THI medio ed è continuata THI alto per 4 giorni interi.



7.3 ESEMPI DI UTILIZZO DEI DATI ED INTERFACCE

Come descritto in precedenza, i dati meteo rilasciati come dati aperti sono esposti in due formati principali: [NetCDF4](#), [convenzioni CF](#) per i data set batch download e [GeoTIFF](#) per quelli destinati alla visualizzazione interattiva.

I data set derivanti dalle immagini radar contengono una stima delle precipitazioni in un raggio di 30 Km centrato sulla Facoltà di Ingegneria dell'università di Cagliari (Piazza d'armi). Per ogni istante (o intervallo) di tempo, il dataset contiene una matrice 1024 x 1024 che rappresenta l'intensità della pioggia in mm/h. L'esempio che segue illustra l'accesso a un dataset di questo tipo in Python.

CODICE

```
import os
from netCDF4 import Dataset

def dump_ncattrs(obj, indent=0):
    for k in obj.ncattrs():
        print("%s%s: %s" % (" " * indent, k, obj.getncattr(k)))

ds = Dataset("2018-05-01_23:00:04.nc", "r")
dump_ncattrs(ds)
```

OUTPUT

```
Conventions: CF-1.7
title: Rainfall
institution: TDM
source: Radar
references: http://www.tdm-project.it
history: Estimated with https://github.com/tdm-project/tdm-tools
```

La griglia spaziale è disponibile sia nel sistema di coordinate originali (EPSG:3003, variabili $x-y$) che in longitudine-latitudine (EPSG:4326, variabili $lon-lat$). Le informazioni dettagliate sulla proiezione originale sono disponibili attraverso la variabile di appoggio `crs`.

CODICE

```
for k, v in ds.variables.items():
    print("%s:" % k)
    dump_ncattrs(v, indent=2)
```

OUTPUT

```
time:
  long_name: time
  standard_name: time
  units: seconds since 2018-05-01 23:00:04
```



```
x:
  long_name: x coordinate of projection
  standard_name: projection_x_coordinate
  units: m
y:
  long_name: y coordinate of projection
  standard_name: projection_y_coordinate
  units: m
lat:
  long_name: latitude coordinate
  standard_name: latitude
  units: degrees_north
lon:
  long_name: longitude coordinate
  standard_name: longitude
  units: degrees_east
rainfall_rate:
  FillValue: -1.0
  least_significant_digit: 4
  long_name: estimated rainfall rate
  standard_name: rainfall_rate
  coordinates: lat lon
  grid_mapping: crs
  units: mm/hour
crs:
  grid_mapping_name: transverse_mercator
  longitude_of_central_meridian: 9.0
  latitude_of_projection_origin: 0.0
  false_easting: 1500000.0
  false_northing: 0.0
  scale_factor_at_central_meridian: 0.9996
  semi_major_axis: 6378388.0
  inverse_flattening: 297
  projected_coordinate_system_name: EPSG:3003 Monte Mario / Italy zone 1
  geographic_coordinate_system_name: Monte Mario
  horizontal_datum_name: Monte Mario
  reference_ellipsoid_name: International 1924
  prime_meridian_name: Greenwich
  towgs84: [-104.1 -49.1 -9.9 0.971 -2.917 0.714 -11.68 ]
  crs_wkt: PROJCS["Monte Mario / Italy zone 1",GEOGCS["Monte
Mario",DATUM["Monte Mario",SPHEROID["International
1924",6378388,297,AUTHORITY["EPSG","7022"]],TOWGS84[-104.1,-49.1,-9.9,0.971,-2.917,0.714,-
11.68],AUTHORITY["EPSG","6265"]],PRIMEM["Greenwich",0,AUTHORITY["EPSG","8901"]],UNIT["degree",0.0174532
925199433,AUTHORITY["EPSG","9122"]],AUTHORITY["EPSG","4265"]],PROJECTION["Transverse Mercator"],PARAMET
ER["latitude_of_origin",0],PARAMETER["central_meridian",9],PARAMETER["scale_factor",0.9996],PARAMETER["
false_easting",1500000],PARAMETER["false_northing",0],UNIT["metre",1,AUTHORITY["EPSG","9001"]],AXIS["X
,EAST],AXIS["Y",NORTH],AUTHORITY["EPSG","3003"]]
```

Il tempo, come previsto dalle convenzioni CF, è rappresentato dal numero di istanti (in questo caso secondi) a partire da un valore iniziale specificato negli attributi della variabile. Nel nostro caso, tale valore coincide con il primo istante memorizzato nel data set.

CODICE

```
t = ds.variables["time"]
t
```

OUTPUT

```
<class 'netCDF4._netCDF4.Variable'>
float32 time(time)
```



```
long_name: time
standard_name: time
units: seconds since 2018-05-01 23:00:04
unlimited dimensions:
current shape = (60,)
filling off
```

CODICE

```
import cf_units
tu = cf_units.Unit(t.units)
dts = [tu.num2date(_) for _ in t]
print("time range: from %s to %s" % (dts[0], dts[-1]))
```

OUTPUT

```
time range: from 2018-05-01 23:00:04 to 2018-05-01 23:59:05
```

La stima delle precipitazioni è contenuta in `rainfall_rate`, un piano per ogni istante temporale:

CODICE

```
rr = ds.variables["rainfall_rate"]
rr
```

OUTPUT

```
<class 'netCDF4._netCDF4.Variable'>
float32 rainfall_rate(time, x, y)
  _FillValue: -1.0
  least_significant_digit: 4
  long_name: estimated rainfall rate
  standard_name: rainfall_rate
  coordinates: lat lon
  grid_mapping: crs
  units: mm/hour
unlimited dimensions:
current shape = (60, 1024, 1024)
filling on
```

Si noti che i dati sono rappresentati da [masked arrays](#) (contengono dati non validi, in particolare all'esterno del raggio d'azione del radar).

Esempio di visualizzazione:

CODICE

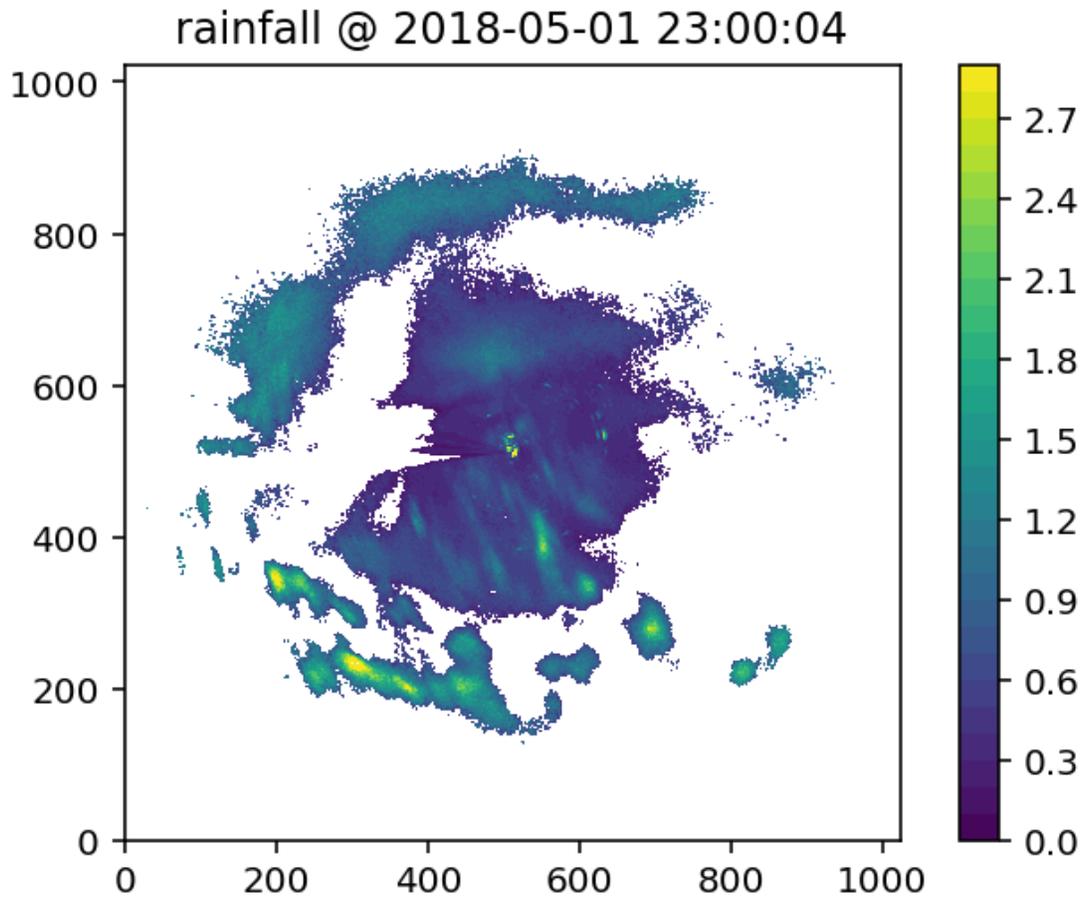
```
import matplotlib.pyplot as plt
import numpy as np

def show_rain(rain, dt):
    plt.cla()
    plt.clf()
    plt.figure(dpi=144)
    c = plt.contourf(rain, levels=np.arange(0, 3, 0.1))
    cbar = plt.colorbar(c)
```

```
plt.title("rainfall @ %s" % dt)  
plt.gca().set_aspect("equal")
```

```
show_rain(rr[0], dts[0])
```

OUTPUT



I dati GeoTIFF contengono le precipitazioni relative a singoli istanti di tempo, riproiettate in EPSG:4326. Un esempio di accesso e visualizzazione è mostrato di seguito.

CODICE

```
import gdal
import matplotlib.pyplot as plt
import numpy as np

def band_to_ma(band):
    assert band.GetMaskFlags() == gdal.GMF_NODATA
    m_band = band.GetMaskBand()
    return np.ma.masked_array(
        band.ReadAsArray(),
        mask=(m_band.ReadAsArray() == 0),
        fill_value=band.GetNoDataValue()
    )

def read_rain(fn):
    dataset = gdal.Open(fn)
    return band_to_ma(dataset.GetRasterBand(1))

def show_rain(rain):
    plt.cla()
    plt.clf()
    plt.figure(dpi=144)
    c = plt.contourf(rain, levels=np.arange(0, 3, 0.1))
    cbar = plt.colorbar(c)
    plt.gca().set_aspect("equal")

def show_rain_from_file(fn):
    show_rain(read_rain(fn))

show_rain_from_file("2018-05-03_11:00:00.tif")
```

OUTPUT

