NOI A.G. / S.p.A.
Roberto Cavaliere
r.cavaliere@noi.bz.it
T +39 0471 066 676

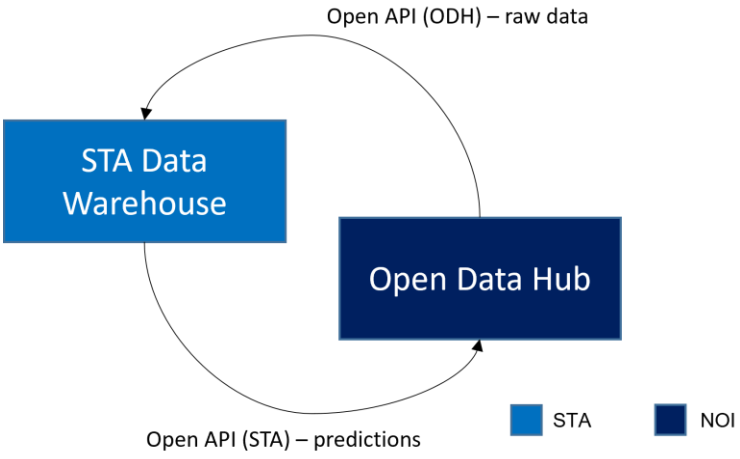
Parking Prediction STA Data Collector

v1.2, 28.02.2022

Note preliminari	1
Dettagli del flusso dati messo a disposizione dal data provider	2
End-point e meccanismo di autenticazione	2
Metadati stazioni	2
Metadati tipi	2
Dati	3
Frequenza di interrogazione del web-service	4

Note preliminari

Nell'ambito dei lavori di sviluppo di un proprio Data Warehouse STA ha realizzato un modello basato su logiche AI di previsione di occupazione dello stato dei parcheggi. Il modello prende i dati dall'Open Data Hub, ne elabora le previsioni e le restituisce in output all'Open Data Hub stesso con una propria API.



Dettagli del flusso dati messo a disposizione dal data provider

I dati sono messi a disposizione dell'Open Data Hub via REST API, con formato JSON:

- **publish_timestamp [timestamp]**: è il timestamp di pubblicazione della previsione
- **forecast_start_timestamp [timestamp]**: è il timestamp relativo alla prima previsione della serie che viene prodotta
- **forecast_period_seconds [integer]**: è il periodo tra due istanti consecutivi per i quali è disponibile una previsione, espresso in [secondi]. Attualmente settato a 300 (5 minuti).
- **forecast_duration_hours [integer]**: è la durata totale della previsione, espressa in [ore]. Attualmente settata a 48 (2 giorni). Questo significa che è disponibile fino ad una previsione a 2 giorni dall'istante iniziale di elaborazione.
- **timeseries [array]**: è il tracciato dati che contiene le elaborazioni, secondo questo schema ad albero
    - **stationcode**
        - **nr. elaborazione**
            - **ts**: istante di riferimento della previsione:
            - **lo**: valore conservativo ("low") di previsione
            - **mean**: valore medio di previsione
            - **hi**: valore ottimistico ("high") di previsione
            - **rmse**: errore quadratico medio della previsione (indicatore di accuratezza)

End-point e meccanismo di autenticazione

L'end-point da cui ottenere le previsioni è il seguente: https://web01.sta.bz.it/parking-forecast/result.json

Non sono previsti attualmente meccanismi di autenticazione.

Metadati stazioni

Non applicabile in questo caso, visto che si tratta di aggiungere tipi di "misure" a stazioni già esistenti.

Metadati tipi

L'Open Data Hub gestisce già tipi per le previsioni di parcheggio, in particolare:

- **parking-forecast-30**: previsione di occupazione a 30 minuti
- **parking-forecast-60**: previsione di occupazione a 1 ora
- **parking-forecast-90**: previsione di occupazione a 1 ora e mezza
- **parking-forecast-120**: previsione di occupazione a 2 ore
- **parking-forecast-150**: previsione di occupazione a 2 ore e mezza
- **parking-forecast-180**: previsione di occupazione a 3 ore
- **parking-forecast-210**: previsione di occupazione a 3 ore e mezza
- **parking-forecast-240**: previsione di occupazione a 4 ore

Si propone di salvare soltanto le previsioni a questi istanti, e non tutti quelli forniti dal web-service, per contenere il numero di tipi da gestire a livello di Open Data Hub. In altre parole, saranno da considerare solo le previsioni fornite rispetto a certe posizioni dell'array che è disponibile.

D'altro canto, il modello fornisce tre valori di previsione anziché uno. E' importante acquisire tutti e tre i valori perché può dare un'indicazione del grado di incertezza del modello. Per qui si propone di usare i tipi esistenti per la previsione media ("mean"), e di generare nuovi tipi per i due nuovi parametri di previsione, nello specifico:

- **parking-forecast-low-30**: previsione conservativa di occupazione a 30 minuti
- **parking-forecast- low-60**: previsione conservativa di occupazione a 1 ora
- **parking-forecast- low-90**: previsione conservativa di occupazione a 1 ora e mezza
- **parking-forecast- low-120**: previsione conservativa di occupazione a 2 ore
- **parking-forecast- low-150**: previsione conservativa di occupazione a 2 ore e mezza
- **parking-forecast- low-180**: previsione conservativa di occupazione a 3 ore
- **parking-forecast- low-210**: previsione conservativa di occupazione a 3 ore e mezza
- **parking-forecast- low-240**: previsione conservativa di occupazione a 4 ore
- **parking-forecast-high-30**: previsione ottimistica di occupazione a 30 minuti
- **parking-forecast- high -60**: previsione ottimistica di occupazione a 1 ora
- **parking-forecast- high -90**: previsione ottimistica di occupazione a 1 ora e mezza
- **parking-forecast- high -120**: previsione ottimistica di occupazione a 2 ore
- **parking-forecast- high -150**: previsione ottimistica di occupazione a 2 ore e mezza
- **parking-forecast- high -180**: previsione ottimistica di occupazione a 3 ore
- **parking-forecast- high -210**: previsione ottimistica di occupazione a 3 ore e mezza
- **parking-forecast- high -240**: previsione ottimistica di occupazione a 4 ore
- **parking-forecast-rmse-30**: previsione ottimistica di occupazione a 30 minuti
- **parking-forecast- rmse -60**: errore quadratico medio delle previsioni a 1 ora
- **parking-forecast- rmse -90**: errore quadratico medio delle previsioni a 1 ora e mezza
- **parking-forecast- rmse -120**: errore quadratico medio delle previsioni a 2 ore
- **parking-forecast- rmse -150**: errore quadratico medio delle previsioni a 2 ore e mezza
- **parking-forecast- rmse -180**: errore quadratico medio delle previsioni a 3 ore
- **parking-forecast- rmse -210**: errore quadratico medio delle previsioni a 3 ore e mezza
- **parking-forecast- rmse -240**: errore quadratico medio delle previsioni a 4 ore

Dati

I valori delle previsioni sono da salvare nelle tabelle **measurement** (in cui vengono salvati solo i valori più recenti) e **measurementhistory**.

Campi web-service	Colonne tabella measurement del database
ts	**timestamp**
lo / mean / hi	**double_value**

Tabella 1: Mapping tra gli attributi del tracciato dati fornito e la tabella measurement (measurementhistory) dell'Open Data Hub.

Frequenza di interrogazione del web-service

Le previsioni sono elaborate su base oraria. Per evitare latenze nell'integrazione dei dati nell'Open Data Hub, si propone di interrogare il servizio ogni **10 minuti**, controllando il valore del campo publish_timestamp che quindi deve essere salvato lato Data Collector per capire se c'è stato un aggiornamento del modello. Il processo di aggiornamento dei dati avviene quindi se la data fornita da questo campo è cambiata.