

# Urban Mobility Ontology

Lorgna Lorenzo 829776  
Marzorati Stefano 830272



# Contenuti

---

## Introduzione

Introduzione al progetto, obiettivi e ontologia di partenza

01

02

## Metodologia

Presentazione della metodologia utilizzata nel progetto

---

## Sviluppo e Query

Modellazione delle entità, regole, inferenze e query

03

04

## Conclusioni e Sviluppi

Conclusioni sul progetto e sviluppi futuri



**01**

# **Introduzione**

# Introduzione

---

## Dati non strutturati

L'informazione disponibile riguardante i sistemi di trasporto è spesso non strutturata e non organizzata al meglio.

## Dati non omogenei

Ogni organizzazione spesso opera in modo indipendente adottando i propri formati e standard.

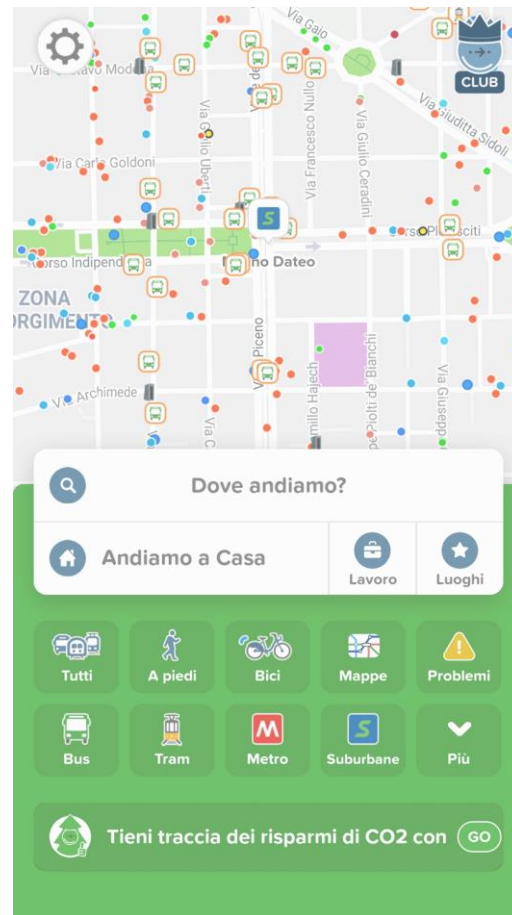
Avere questo tipo di frammentazione dei dati significa rendere più complessa per l'utente l'operazione di estrazione di informazioni interessanti e utili nel contesto della mobilità urbana.

# Obiettivo

Lo scopo di questo progetto è ottenere un'ontologia che sia di supporto all'utente modellando i dati necessari per muoversi all'interno di una grande città.

Gli aspetti modellati in questione riguardano sia la mobilità tradizionale che la sharing mobility.

Per perseguire tale obiettivo si è proceduto riadattando un'ontologia già esistente sulle medesime tematiche.



# Ontologia di Partenza

---

Si è partiti dalla suite di ontologie, **Ontologie per gli apparati IoT urbani (Urban-IoT)**, sviluppata in collaborazione con il comune di Milano.

Urban-IoT contiene ontologie sviluppate per supportare l'integrazione e la pubblicazione di dati dei diversi fornitori di servizi che operano nell'ambito urbano.

Comune-Milano/  
**ontologie-iot-urbani**

Ontologie per gli Apparati IoT Urbani



3 Contributors  
0 Issues  
4 Stars  
2 Forks



- **Modulo core**
- Modulo Sharing
- Modulo Electric Mobility

Il modulo Core è stato riadattato con un maggior focus sull'utente prendendo in considerazione anche gli aspetti della mobilità tradizionale.



**02**

# **metodologia**

## Metodologia utilizzata

La metodologia scelta è **SAMOD** di Silvio Peroni.

Tale metodologia si basa su **3 step iterativi**:

- Studio di uno specifico dominio e creazione di test case
- Unione dei nuovi test case al modello già esistente
- Miglioramento del modello ottenuto

## Modelizzazione

Tramite approcci **bottom up** e **top down** sono state definite entità e relazioni.

In questa fase sono stati utilizzati:

- Schema.org
- Protégé
- OWL
- GeoSPARQL

## A-BOX

Non avendo a disposizione dati reali è stata creata un **A-BOX** al fine di validare l'ontologia rispetto alla capacità di modellare le informazioni sottostanti.

## Query

Ai fini di simulare dei casi d'uso reali dell'applicazione sono state definite delle **query SPARQL**.





**03**

## **Sviluppo e Query**

# Modellazione

---



**Veicolo**



**Servizio**



**Sessione utente**



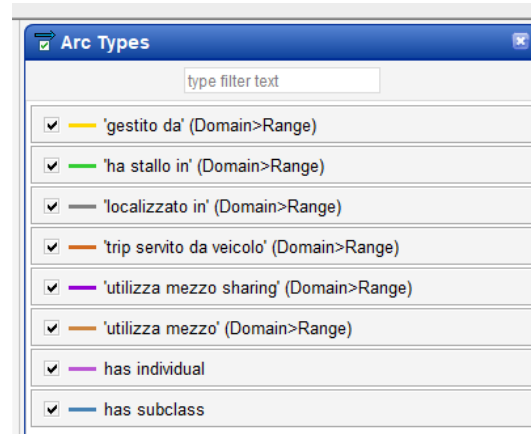
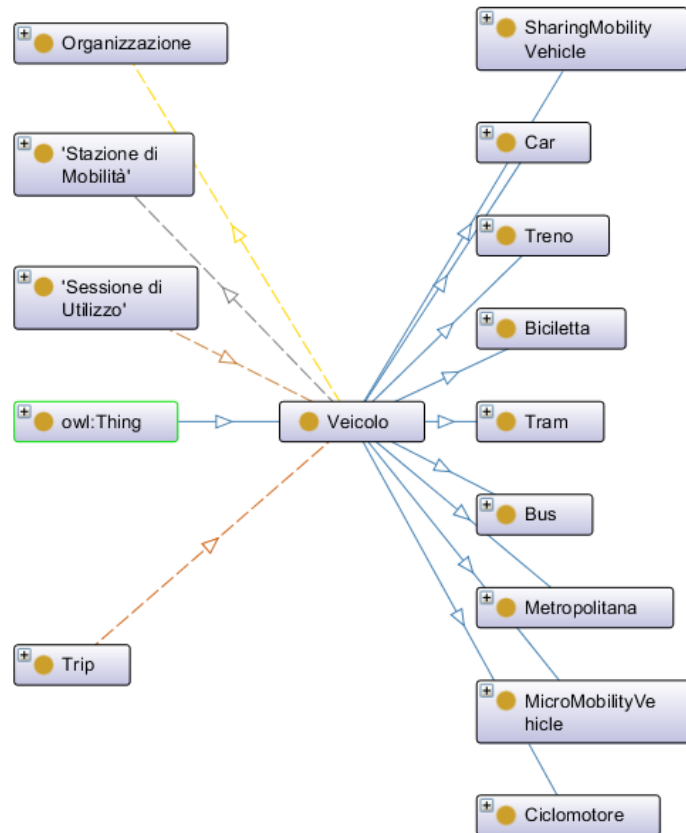
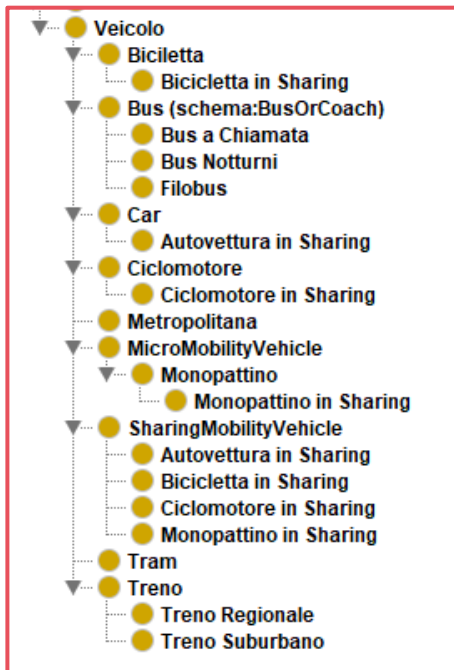
**Stazione di  
mobilità**



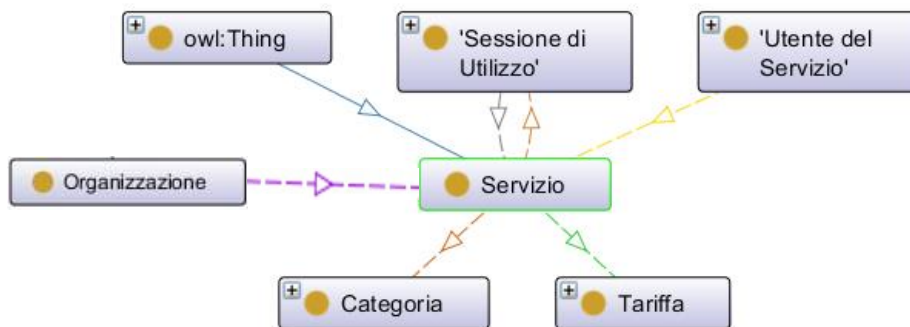
**Linea di  
trasporto**



# Veicolo



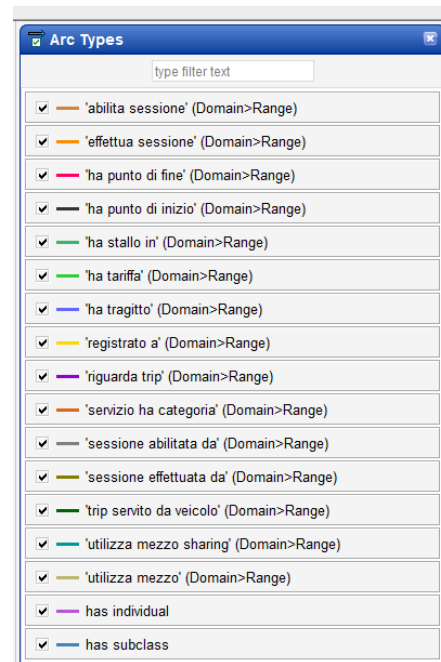
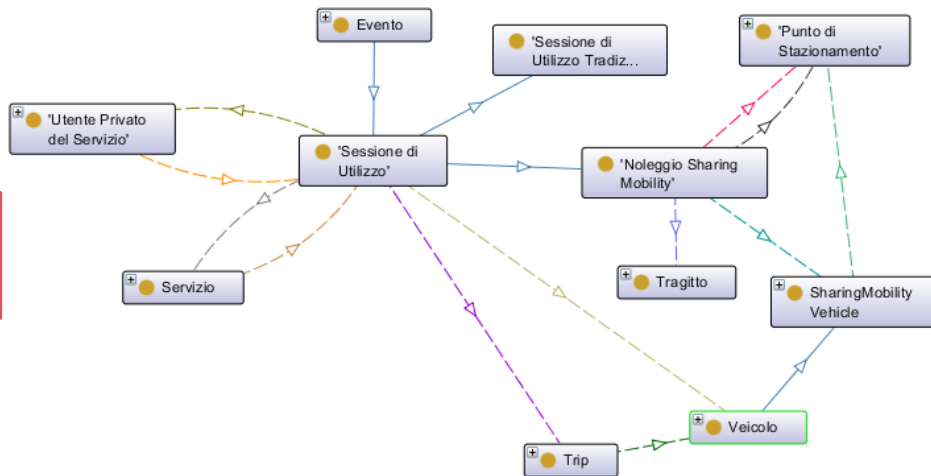
..... Servizio



Arc Types	
	type filter text
<input checked="" type="checkbox"/>	'abilita sessione' (Domain>Range)
<input checked="" type="checkbox"/>	'ha tariffa' (Domain>Range)
<input checked="" type="checkbox"/>	'registrato a' (Domain>Range)
<input checked="" type="checkbox"/>	'servizio ha categoria' (Domain>Range)
<input checked="" type="checkbox"/>	'sessione abilitata da' (Domain>Range)
<input checked="" type="checkbox"/>	has subclass

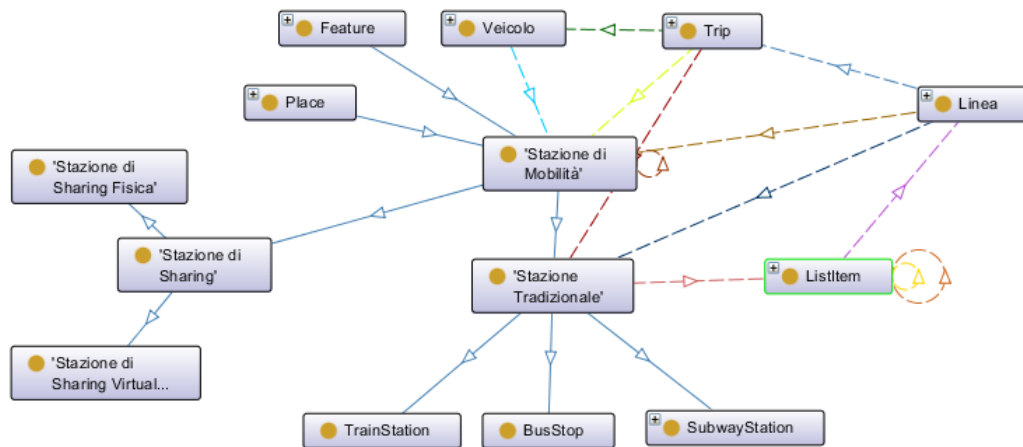


# Sessione di utilizzo





# Stazione di mobilità



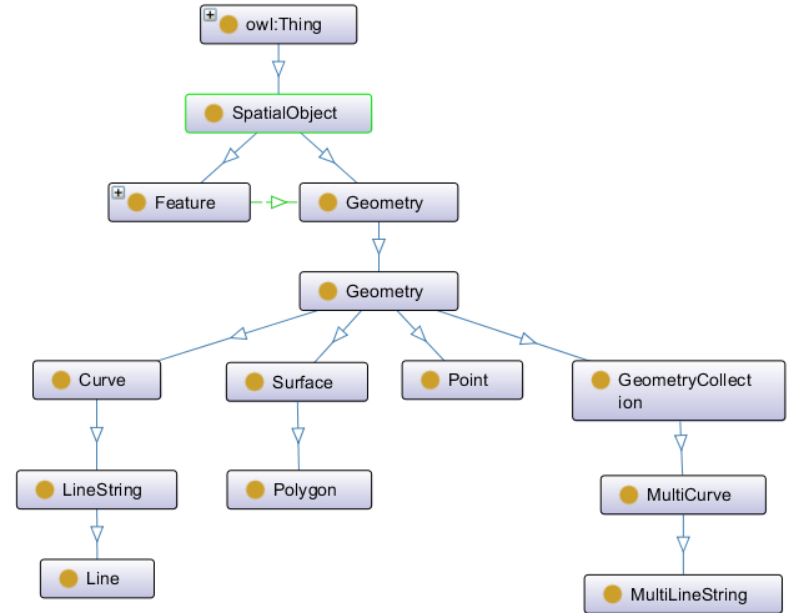
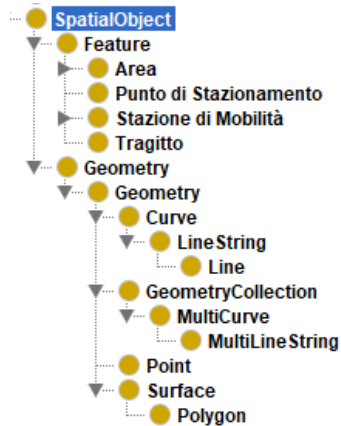
Arc Types	
type filter text	
<input checked="" type="checkbox"/>	'abilita sessione' (Domain>Range)
<input checked="" type="checkbox"/>	'contenuta in linea' (Domain>Range)
<input checked="" type="checkbox"/>	'corrisponde a fermata' (Domain>Range)
<input checked="" type="checkbox"/>	'corrisponde a Trip' (Domain>Range)
<input checked="" type="checkbox"/>	'effettua sessione' (Domain>Range)
<input checked="" type="checkbox"/>	'ha fermata precedente' (Domain>Range)
<input checked="" type="checkbox"/>	'ha fermata successiva' (Domain>Range)
<input checked="" type="checkbox"/>	'ha punto di fine' (Domain>Range)
<input checked="" type="checkbox"/>	'ha punto di inizio' (Domain>Range)
<input checked="" type="checkbox"/>	'ha stallo in' (Domain>Range)
<input checked="" type="checkbox"/>	'ha tariffa' (Domain>Range)
<input checked="" type="checkbox"/>	'ha tragitto' (Domain>Range)
<input checked="" type="checkbox"/>	'inclusa nella stazione di mobilità' (Domain>Range)
<input checked="" type="checkbox"/>	'localizzato in' (Domain>Range)
<input checked="" type="checkbox"/>	'passa per stazione di mobilità' (Domain>Range)
<input checked="" type="checkbox"/>	'passa per stazione tradizionale' (Domain>Range)
<input checked="" type="checkbox"/>	'registrato a' (Domain>Range)
<input checked="" type="checkbox"/>	'riguarda trip' (Domain>Range)
<input checked="" type="checkbox"/>	'servizio ha categoria' (Domain>Range)
<input checked="" type="checkbox"/>	'sessione abilitata da' (Domain>Range)
<input checked="" type="checkbox"/>	'sessione effettuata da' (Domain>Range)
<input checked="" type="checkbox"/>	'trip passa per stazione di mobilità' (Domain>Range)
<input checked="" type="checkbox"/>	'trip passa per stazione tradizionale' (Domain>Range)
<input checked="" type="checkbox"/>	'trip servito da veicolo' (Domain>Range)
<input checked="" type="checkbox"/>	'utilizza mezzo sharing' (Domain>Range)
<input checked="" type="checkbox"/>	'utilizza mezzo' (Domain>Range)
<input checked="" type="checkbox"/>	has individual
<input checked="" type="checkbox"/>	has subclass



Arc Types	
type filter text	
✓	'abilità sessione' (Domain>Range)
✓	'contenuta in linea' (Domain>Range)
✓	'corrisponde a fermata' (Domain>Range)
✓	'corrisponde a Trip' (Domain>Range)
✓	'effettua sessione' (Domain>Range)
✓	'gestito da' (Domain>Range)
✓	'ha fermata precedente' (Domain>Range)
✓	'ha fermata successiva' (Domain>Range)
✓	'ha punto di fine' (Domain>Range)
✓	'ha punto di inizio' (Domain>Range)
✓	'ha stallo in' (Domain>Range)
✓	'ha tariffa' (Domain>Range)
✓	'ha tragitto' (Domain>Range)
✓	'inclusa nella stazione di mobilità' (Domain>Range)
✓	'linea gestita da' (Domain>Range)
✓	'localizzato in' (Domain>Range)
✓	'passa per stazione di mobilità' (Domain>Range)
✓	'passa per stazione tradizionale' (Domain>Range)
✓	'registrato a' (Domain>Range)
✓	'riguarda trip' (Domain>Range)
✓	'servizio ha categoria' (Domain>Range)
✓	'sessione abilitata da' (Domain>Range)
✓	'sessione effettuata da' (Domain>Range)
✓	'trip passa per stazione di mobilità' (Domain>Range)
✓	'trip passa per stazione tradizionale' (Domain>Range)
✓	'trip servito da veicolo' (Domain>Range)
✓	'utilizza mezzo sharing' (Domain>Range)
✓	'utilizza mezzo' (Domain>Range)
✓	has individual
✓	has subclass

# GeoSPARQL

GeoSPARQL (Geographic Query Language for RDF Data) è uno standard che definisce un set di funzioni SPARQL e un vocabolario RDF/OWL per le informazioni geografiche, basato sul General Feature Model.





# Regole

---

Al fine di completare in maniera più dettagliata le proprietà degli individui sono state definite delle regole che permettono di aumentare il numero di inferenze generate di conseguenza la conoscenza generata.

'inclusa nella stazione di mobilità'(?z, ?y), 'localizzato in stazione'(?x, ?z) -> 'localizzato in'(?x, ?y)

'ha linea'(?a, ?x), 'contiene fermata'(?x, ?z), 'associata a stazione'(?z, ?p), 'inclusa nella stazione di mobilità'(?p, ?y) -> 'trip passa per stazione di mobilità'(?a, ?y)

'trip servito da veicolo'(?z, ?y), 'riguarda trip'(?x, ?z) -> 'utilizza mezzo'(?x, ?y)

'corrisponde a fermata'(?x, ?z), 'contenuta in linea'(?z, ?p), 'corrisponde a Trip'(?p, ?y) -> 'è sul trip'(?x, ?y)

'contiene fermata'(?x, ?z), 'associata a stazione'(?z, ?p), 'inclusa nella stazione di mobilità'(?p, ?y) -> 'passa per stazione di mobilità'(?x, ?y)

'contenuta in linea'(?z, ?y), 'corrisponde a fermata'(?x, ?z) -> 'è sulla linea'(?x, ?y)

# Inferenze

Per la generazione delle inferenze è stato utilizzato il reasoner HermiT che ha permesso di verificare la coerenza dell'ontologia e di generare nuova conoscenza a partire dai dati inseriti.

Alcuni esempi di inferenze ottenute:

Property assertions: Mobike2

Object property assertions

- 'gestito da' Mobike
- 'localizzato in stazione' Mobike\_Stazione\_Virtuale\_Sharing\_2
- 'ha stato' Non\_Disponibile
- 'localizzato in' Cairoli\_Stazione\_di\_Mobilita
- 'utilizzato da sessione' Noleggio1
- 'utilizzato da sessione' Noleggio3

Data property assertions

- emissionsCO2 0
- numero seriale "MB002"

Property assertions: Non\_Disponibile

Object property assertions

- 'è lo stato di' Metropolitana\_002
- 'è lo stato di' Mobike1
- 'è lo stato di' Mobike2
- 'è lo stato di' Share\_Now2
- 'è lo stato di' ATM\_Stazione\_Fisica\_Sharing\_2

Data property assertions

- valoreStato "Non Disponibile"^^xsd:string

Property assertions: Cadorna\_Stazione\_di\_Mobilita

Object property assertions

- 'include stazione specifica' Subway\_Station\_Cadorna\_FN
- 'include stazione specifica' Bus\_Stop\_Cadorna\_M1\_M2
- 'include stazione specifica' Mobike\_Stazione\_Virtuale\_Sharing\_1
- 'stazione di mobilità è sul trip' Trip\_Bus\_50\_01
- 'stazione di mobilità è sul trip' Trip\_Bus\_50\_03
- 'stazione di mobilità è sul trip' Trip\_Bus\_50\_02
- 'stazione di mobilità è sul trip' Trip\_Metropolitana\_M1
- 'stazione mobilità contiene mezzo' Mobike1
- 'stazione mobilità contiene mezzo' Mobike3
- 'stazione di mobilità è sulla linea' Linea\_Bus\_50
- 'stazione di mobilità è sulla linea' Linea\_Metropolitana\_M1\_Bisceglie

Data property assertions

- nome "Cadorna Stazione di Mobilità"

# Query

Con Snap SPARQL Query, plugin di Protégé, sono state recuperate informazioni di interesse dall'ontologia sviluppata, simulando dei casi d'uso reali lato utente, che potrebbero rappresentare le funzionalità di un'applicazione.

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
```

```
SELECT ?stazione_mobilità (count(?veicolo) as ?conteggio_veicoli)
```

```
WHERE {
  ?stazione_mobilità uon:stazione_mobilita_contiene_mezzo ?veicolo.
```

```
}
GROUP BY ?stazione_mobilità
```

?stazione_mobilità	?conteggio_veicoli
uon:Cairolì_Stazione_di_Mobilità	1
uon:Cadorna_Stazione_di_Mobilità	2

Vengono restituite le stazioni di mobilità e il rispettivo numero di veicoli presenti in esse

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
```

```
SELECT ?istanza
```

```
WHERE {
  ?istanza uon:ha_stato ?stato.
  ?stato uon:valoreStato "Disponibile" .
}
```

?istanza
uon:ATM_Bus_1
uon:ATM_Stazione_Fisica_Sharing_1
uon:Metropolitana_001
uon:Share_Now1
uon:Tram001
uon:Trenord01

Vengono riportate tutte le stazioni di mobilità e tutti i veicoli il cui stato è disponibile

# Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
PREFIX schema: <http://schema.org/>
```

```
SELECT DISTINCT ?stato_stallo
```

```
WHERE {
  ?stallo schema:name "ATM_Stazione_Fisica_Sharing_1".
  ?stallo uon:ha_stato ?stato .
  ?stato uon:valoreStato ?stato_stallo
}
```

?stato_stallo
---------------

Disponibile^^xsd:string

Dato uno stallo specifico viene restituito lo stato di esso

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
```

```
SELECT ?utente (count(?sessione) as ?numero_sessioni)
```

```
WHERE {
  ?utente iot:performsSession ?sessione.
  ?sessione rdf:type uon:Noleggio_Sharing_Mobility.
}
GROUP BY ?utente
```

?utente	?numero_sessioni
---------	------------------

uon:Utente1	2
-------------	---

Dato un utente viene restituito il numero di sessioni ad esso associato

# Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
PREFIX schema: <http://schema.org/>
```

```
SELECT (SAMPLE(?tipologia_mezzo) AS ?tipologia_mezzo) (COUNT(?tipologia_mezzo) as
?numero_sessioni)
```

```
WHERE {
    ?utente schema:name "Mario Rossi".
    ?utente iot:performsSession ?sessione.
    ?sessione uon:utilizza_veicolo ?mezzo.
    ?mezzo rdf:type ?tipologia_mezzo.

    MINUS { ?mezzo rdf:type ?other.
             ?other rdfs:subClassOf ?tipologia_mezzo.
             FILTER (?other != ?tipologia_mezzo)
    }
}
GROUP BY ?tipologia_mezzo
ORDER BY DESC(?tipologia_mezzo)
```

?tipologia_mezzo	?numero_sessioni
uon:Metropolitana	1
uon:Bicicletta_in_Sharing	2

Dato un utente specifico viene restituito il numero di sessioni effettuate con le diverse tipologie di mezzo ad esso associate

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
PREFIX schema: <https://schema.org/>
```

```
SELECT ?corsa
WHERE {
    ?corsa rdf:type schema:ItemList.
```

```
}
```

?corsa
uon:Linea_Tram_1
uon:Linea_Bus_50
uon:Linea_Treno_S4
uon:Linea_Metropolitana_M1_Bisceglie

Vengono restituite tutte le linee delle diverse tipologie di mezzi

# Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
PREFIX schema: <http://schema.org/>
```

**SELECT ?trip ?orario\_partenza**

```
WHERE {
  ?linea schema:name "Linea Bus 50".
  ?linea uon:corrisponde_a_Trip ?trip.
  ?trip schema:departureTime ?orario_partenza.
}
```

?trip	?orario_partenza
uon:Trip_Bus_50_01	2022-01-04T12:30:00 <sup>xx</sup> xsd:dateTime
uon:Trip_Bus_50_03	2022-01-04T14:30:00 <sup>xx</sup> xsd:dateTime
uon:Trip_Bus_50_02	2022-01-04T13:30:00 <sup>xx</sup> xsd:dateTime

Data una linea si restituiscono le diverse corse e l'orario di partenza delle stesse

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
PREFIX schema: <http://schema.org/>
```

**SELECT ?servizio ?tariffa\_offerta**

```
WHERE {
  ?organizzazione schema:name "ATM".
  ?organizzazione uon:eroga_servizio ?servizio.
  ?servizio uon:ha_tariffa ?offerta.
  ?offerta schema:price ?tariffa_offerta.
}
```

?servizio	?tariffa_offerta
iot:ATM_Servizio_Abbonamento_Annuale	330
iot:ATM_Servizio_Abbonamento_Mensile	39
iot:ATM_Servizio_Biglietto_Carnet	18
uon:ATM_Servizio_Abbonamento_Settimanale_BikeMi	9
iot:ATM_Servizio_Biglietto_Giornaliero	7
uon:ATM_Servizio_Biglietto_90_Min	2
iot:ATM_Servizio_Abbonamento_Annuale_Under_27	200

Data un'organizzazione si restituiscono tutti i servizi che essa offre ed il relativo prezzo

# Query

---

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX iot: <http://www.w3id.org/urban-iot/core#>
PREFIX uon: <http://www.inside.disco.unimib.it/ds/urban_ontology#>
PREFIX schema: <https://schema.org/>
```

```
SELECT ?emissioniCO2
```

```
WHERE {
  ?mezzo iot:serialNumber "ATMBUS001".
  ?mezzo schema:emissionsCO2 ?emissioniCO2.
}
```

40	?emissioniCO2
----	---------------

Dato un mezzo si restituisce la quantità di CO<sub>2</sub> che esso produce



**04**

## **Conclusioni e Sviluppi futuri**



# Conclusioni

---

## Conclusioni

L'ontologia finale riesce a modellare diversi aspetti della mobilità urbana, soddisfacendo i requisiti inizialmente definiti e rispondendo adeguatamente alle domande formulate in un primo momento. Tuttavia trattandosi di un fenomeno complesso e variegato risulterebbe utile il supporto di esperti del dominio al fine di ottenere una modellazione più accurata e attenta, anche in base alle esigenze degli utenti.

## Sviluppi futuri

Alcuni aspetti che potrebbero essere estesi riguardano una migliore e una più ricca gestione dei dati relativi all'utente. In secondo luogo, si potrebbe pensare ad una più precisa suddivisione delle aree geografiche tenendo in considerazione aspetti caratteristici dei diversi fornitori del settore. In conclusione, per far sì che l'ontologia possa essere utilizzata in contesti reali sarebbe necessario implementare una procedura automatizzata per l'importazione dati.

# Riferimenti

---

- Scrocca, M., Baroni, I., & Celino, I. *Urban IoT Ontologies for Sharing and Electric Mobility*.
  - Nandini, D., & Shahi, G. K. (2019). *An Ontology for Transportation System*. Kalpa Publications in Computing, 10, 32-37.
  - <https://comune-milano.github.io/ontologie-iot-urbani/docs/core/index-en.html>
  - Peroni, S. (2016). *A simplified agile methodology for ontology development*. In *OWL: Experiences and Directions—Reasoner Evaluation* (pp. 55-69). Springer, Cham.
-

**Grazie per  
l'attenzione!**