Dipartimento di Ingegneria e Scienza dell'Informazione

– KnowDive Group –

KGE 2023 - Trentino Territory and Transportation

Document Data:	Reference Persons:
January 7, 2024	Rubens Rissi Onzi, Ferrari Eugenio

KnowDive (internal) reports are for internal only use within the KnowDive Group. They describe preliminary or instrumental work which should not be disclosed outside the group. KnowDive reports cannot be mentioned or cited by documents which are not KnowDive reports. KnowDive reports are the result of the collaborative work of members of the KnowDive group. The people whose names are in this page cannot be taken to be the authors of this report, but only the people who can better provide detailed information about its contents. Official, citable material produced by the KnowDive group may take any of the official Academic forms, for instance: Master and PhD theses, DISI technical reports, papers in conferences and journals, or books.

Index:

Revision History:

Revision	Date	Author	Description of Changes
0.1	October 16, 2023	Rubens Rissi Onzi, Ferrari Eugenio	Phase 1 - Purpose Definition
0.2	October 25, 2023	Rubens Rissi Onzi, Ferrari Eugenio	Phase 2 - Information Gathering
0.3	November 8, 2023	Rubens Rissi Onzi, Ferrari Eugenio	Phase 3 - Language Definition
0.4	November 20, 2023	Rubens Rissi Onzi, Ferrari Eugenio	Phase 4 - Knowledge Definition
0.5	January 7, 2024	Rubens Rissi Onzi, Ferrari Eugenio	Phase 5 - Data Definition

1 Introduction

Reusability is one of the main principles in the Knowledge Graph Engineering (KGE) process defined by iTelos. The KGE project documentation plays an important role to enhance the reusability of the resources handled and produced during the process. A clear description of the resources as well as of the process (and sub processes) developed, provides a clear understanding of the project, thus serving such an information to external readers for the future exploitations of the project's outcomes.

This project aims at producing useful data for applications that intend to tell it's users about possible delays using the data we provide. The data will feature details mainly about bus stops locations using it's territory data. With that benefiting Trento inhabitants of urban areas.
The current document has the objective provide a detailed report of the project developed following the iTelos methodology. The report is structured, to describe:

- Section 2: Definition of the project's purpose and its domain of interest.
- Section 3: Description of the project development, based on the two main sub process considered by iTelos: producer and consumer.
- Section 4: Purpose formalization step of the iTelos methodology. This step aims to formalize the purpose, by extracting the functional requirement.
- Section 5: Information gathering step of the iTelos methodology. This step aims to formalize the sources, and data.
- Section 6: Language Definition step of the iTelos methodology. This step aims to identify and formalize the language concepts used to represent the information to be included in the final KG.
- Section 7: Knowledge Definition step of the iTelos methodology. This step aims to unify the representation of the information.
- Section 8: Data definition step of the iTelos methodology. this step aims to merge the knowledge with the data in one structure.
- Section 9: Evaluation of the final structure by analysing how it complies with the project purpose.
- Section 10: The description of the metadata produced for the resources handled and generated by the iTelos process, while executing the project.
- Section 11: Conclusion and open issues.

2 Purpose and Domain of Interest (DoI)

2.1 Project Purpose

The goal of this project is to provide data to applications and services that need information suitable for predicting bus delays. To do so, we want to incorporate data that could affect transportation delays in urban areas, more in detail. We will consider supermarkets, catering stores, education facilities, tourism destinations, population density, number of right and left turns and number of traffic signs [1]. The data shall be integrated in knowledge graphs (KGs), using as base bus transportation and territorial data available. This release is intended to enable applications to use our KGs, created by the available data, to predict possible transportation delays in Trento, city of Italy.

2.2 Domain of Interest

The project will focus on bus transportation and will utilize data from the following time frames:

- Bus transportation data in Trento urban areas, covering the period from 1 September 2022, to 1 October 2023.
- Territorial and facilities data from *Trentino* OSM Places, collected up to 28 February 2023.
- Population data for the year 2018, sourced from the most recent survey conducted by the *Comune di Trento*.

3 Project Development

3.1 Data Production

To fulfill our purpose, we need to produce some resources. These resources either do not currently exist or are of insufficient quality.

The first resource the delay out group could not find any data of it as the Trentino Trasporti doesn't release any data on the subject. For further works as data becomes available can be added to this dataset. For its properties were chosen the predicted delay and actual delay, as its a good way to predict actual delays and compare them to what was chosen.

The second resource we require regards the Trento city areas, we want data about their names, boundaries, and population density. The Municipality of Trento is divided into 12 official areas, the density of population in each of these areas can be a factor for predicting delays, as it can bring too many people in one bus stop making the bus wait more for it.

The third resource we need is about the number right and left turns, and the length of specific bus routes. We will integrate this new information on the bus routes, adding information regarding it‘s name in a matter to identify the lines path. With the path gathered we will construct the entities properties: length, left_turns, right_turns. As mentioned in the paper of 2.2, the length

[1] Relevant features to predict delays: `https://journals.sagepub.com/doi/abs/10.3141/1666-12`

of a route can be a factor in the delays of buses as the left turns case the right turn case was added as it could be used in countries where people use the left-hand traffic system. We plan to create this data by using the coordinates of the routes and checking changes on its axis that do not correspond to roundabouts.

The fourth resource we need is bus_trip, some of his properties, seats and bus_id, can not be found in any of the datasets currently available.

3.2 Data Composition

For our project we are using for the territorial data, *Trentino* OSM places, that contain data for urban transportation that will be used both buses entities, as the name of bus stops. This dataset will provide the information about the facilities in the city as well, such as, name, category and coordinates. Within the information gathered we have chosen some sub classes of facilities as more important than others. Then important facilities were chosen as probable cause on many people trying to access the transport service at same time or in case that it causes business in the area they are located. The chosen facilities were: catering, supermarkets, tourism locations and education facilities [2]. With the KGE22 - *Trentino* Urban Transportation we will gather bus route schedule, their coordinates, names and paths. These selected resources will be composed together with the ones created in the data production step. As mentioned before, we want to integrate the produced data regarding bus route characteristics with the KGE 22 dataset such as: turns and length. This integration will provide valuable information for predicting delays. We also want to link each bus stop within the KGE22 dataset with its respective city area, allow us to relate the bus stops with the population density in that area which may contribute to delays. Last step is to connect each bus stop with the facilities in that area (that we get from Trentino OSM Places dataset), services and facilities in an area can serve as an indicator of the volume of people visiting, potentially leading to delays.

4 Purpose Formalization

In the step of purpose formalization the producer have the goal of creating resources of one or more field of it‘s goal domain and increase the data availability. The role of the consumer is to follow the leads of the purpose. Our Purpose can be formalized by the scenarios, personas and competency questions. For this Project they are as it follows:

4.1 Scenarios

1. A day in Trento on a weekday.
2. A day in Trento on a weekend.
3. A day in Trento on rush hours.
4. A day in Trento on nighttime.

[2] During the information gathering phase we realized that we could not find specific information on this subclasses, so we decided to use more generally facilities not only from these subclasses.

4.2 Personas

1. Giovanni, 19, is a college student that lives in the city centre, even though he studies far, in Mesiano.

2. Isabella, 83, is a senior citizen and lives in the outskirts of Trento, Cassoti di Povo, she often goes with her husband for groceries in weekdays.

3. Lily, 21, is a waiting staff worker in a hotel in the city center, she lives with roommates in a flat in Madonna Bianca. She also likes to go to parties and events in the city.

4. Giosepina, 45, is a wildlife biologist that lives in Mattarello, she has to spend time both in the gathering of samples and behaviours in the wild. She also need to record reports about the samples gathered to the biology department in Povo.

4.3 Competency Questions

1. Isabella, after lunch, wants to reach the city center, where she can find lot of shops to buy groceries to prepare sweets to her daughter. She wants to know how much time it is gonna take to reach the center, and arrive home for dinner.

2. On Tuesday, Giovanni ends his lectures at 19:30 in Mesiano university department. He's curious about his arrival time at the city center, considering that 19:30 falls during rush hour when many people are heading home from work.

3. Isabella wants to visit her daughter in the weekend. As her daughter is available only on Sunday mornings, she have to be aware at witch time she should take the bus, as in the weekend there aren't many available in her part of the city. Her daughter also lives in the other side of the city, Gardolo. She also needs to change bus lines in between.

4. Lily don't want to go too early to work, but as she works sometimes in the night shift starting at 18:30, or in the day shift starting at 7:00. She usually have to go early to not get late. The bus that she rides in the afternoon is always full, by the time that she needs to leave. The hotel she works is well placed, having many amenities like museum and attractions nearby.

5. After enjoying a dinner with his friends, Giovanni decide to head to one of his friend's houses in Martignano. They are fortunate to be right on schedule for the last bus. The buses to Martignano are usually punctual, as the area had fewer residents compared to the city center, resulting in less traffic. They want to confirm if the bus will run on time so that they can arrive at their destination on time.

6. Giosepina needs to present the data gathered in the province of Trento, but as it is raining she prefer not to go by motorcycle, her preferred method, as it would drench her clothes and risk getting sick. Then she decides to take a bus from her house in Mattarello to go to Povo, but she has to take multiple lines for it, she then needs to check the delay of the second line in order to arrive in time in Povo, as her bus stop is the first in the bus ride.

4.4 Concepts Identification and Categorization

From the scenarios, personas and CQs we extract the following entities with their properties:

Table 1: CQs Extraction

Scenarios	Personas	Competency Questions	Entities	Properties	Focus	Popularity
1 - 2 - 3	2	1	bus_stop	(id: int, name: string, coordinates: location, arrival_times: schedule)	Contextual	Common
1 - 2 - 3	2	1	delay	(id: int, predicted_delay_time: string, actual_delay_time: string)	Contextual	Contextual
1 - 2 - 3	2	1	supermarket	(id: int, name: string, coordinates: string,)	Common	Common
1 - 3 - 4	1	2	education_facilities	(id: int, name: string, coordinates: string,)	Common	Common
1 - 3 - 4	2	3	bus_route	(id: int, name: string, length: float, left_turns: int, right_turns: int,)	Contextual	Contextual
1 - 3 - 4	2	3	city_area	(id: int, name: string, boundary: string, population_density: float)	Core	Contextual
1 - 2 - 3 - 4	3	3	tourism_destinations	(id: int, name: string, coordinates: string,)	Common	Common
1 - 2 - 4	1	4	catering	(id: int, name: string, coordinates: string,)	Common	Common
1 - 3	4	5	bus_trip	(id: int, seats: int, bus_id: int, weekdays: string,)	Contextual	Core

4.5 ER Modeling

Given the entities and property identified in the step above, we can design the purpose ER model as in Figure 1. As it can be seen in Table 1 the ETypes supermarket, education_facilities, tourism_destinations and catering, have common properties, initially we opted to treat them as a subclass of the facility EType and use only those as factors that contribute to delays, but during the next phase of information gathering we realized we could not get specific data on just those, so we decided to just consider more generally the facility EType. In the following we will consider and describe the new version, the one with just the facility EType.

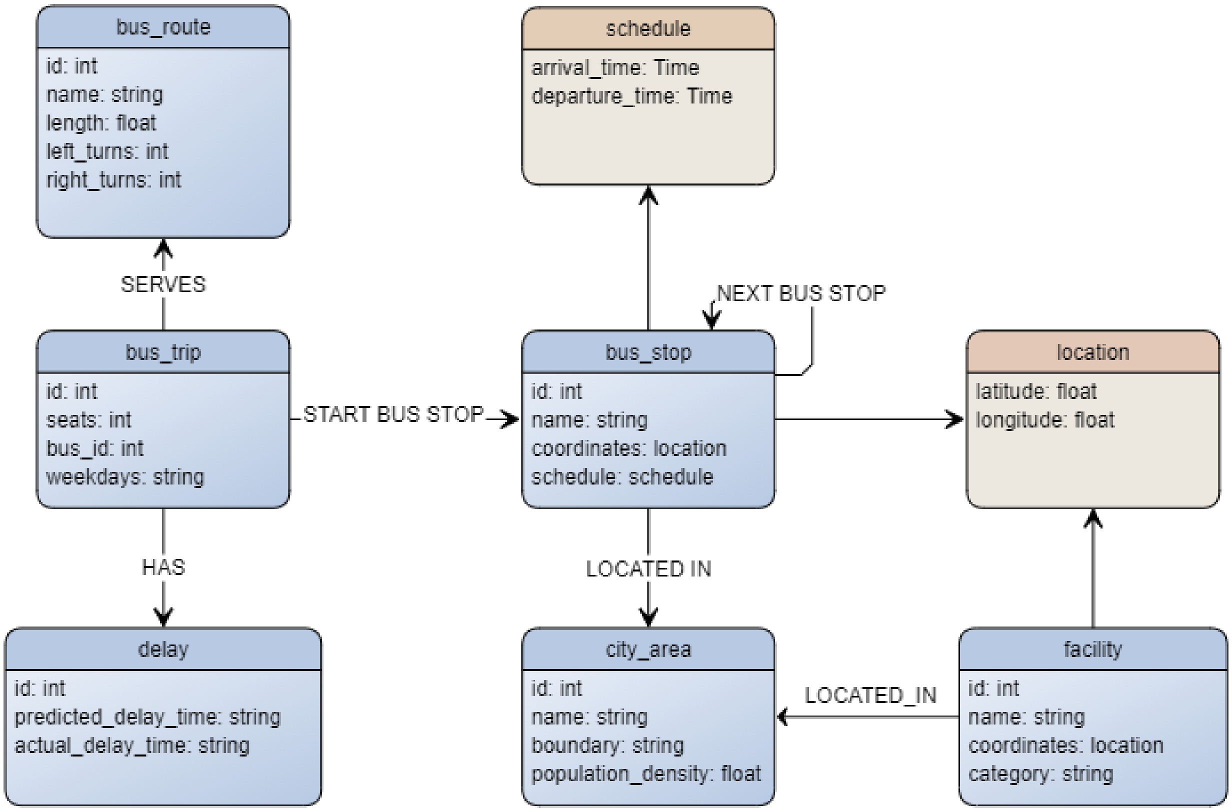


Figure 1: ER model

In our ER model we have 6 ETypes.

1. delay: as the main importance for the project, the delay will have the following proprieties:
    - id: this propriety is the identification of all the entities, as it was used in all the other entities it will not be shown again.
    - predicted_delay_time: chosen to store possible tries to predict the delay in contrast with the actual_delay_time. Can also be used in future predictions models.
    - actual_delay_time: chosen as it will help the future predictions of models in contrast with the predicted_delay_time. It has the problem that we currently have no free database available with it.
2. facility: it is used to represent a location with a purpose that could be used to predict delays. The facility will have the following properties:
    - name: Naming used for the facility, as a matter of identification for people.
    - coordinates: A location based propriety that has its own proprieties that will be explained below. Is used to locate a facility on a map that uses longitude and latitude.
    - category: A string representing the category of this facility.
3. city_area: as previously said in the report, we consider the twelve official division of the urban area of the municipality of Trento, this allow us to model population density for the Trento urban areas. It will be used as way to find how many people live in the area, as many people use the transport, it will bring delays. Each city area is modeled by this entity with the following properties:
    - name: Name of the *circoscrizione*, as a matter of identification for people.
    - boundaries: the geographical boundaries of the area, used to know where the greater population densities are.
    - population_density: the density of the population living in that area of the city of Trento, used to try to infer delays by people using the public transportation.
4. bus_route: this entity represents a bus route with some of the information needed for our purpose that aims at predicting delays, as some of it's proprieties are used in this endeavor, these are the properties:
    - name: the name of the bus route, as a matter of identification for people.
    - length: the length of the bus route path, as delays times can cause ripple effect on it's path.
    - left_turns: the number of left turns in the path, as this maneuver can sometimes conflict with the ongoing traffic, making it slower to follow the predicted path.
    - right_turns: the number of right turns in the path, as the left_turns but in right side driving countries.
5. bus_stop: it represent a bus stop. It has a self-relation to itself, allowing us to represent the sequence of bus stops that a particular bus trip needs to perform. The bus_stop will have the following properties:

- name: the name of the bus_stop.
- coordinates: A location based propriety, allowing us to locate the bus stop.
- arrival_time: type schedule (the schedule type is defined below), contains information about the time of arrival and expected departure of the bus.

6. bus_trip: this entity represent a specific bus trip for a particular bus route, it has a relation to his first bus stop and contains the following properties:

    - seats: the number of seats in the bus doing this trip, also this can be considered as a factor that could possibly contribute to delays, as more people can enter and exit it.
    - bus_id: the identifier of the bus ongoing trip, as delays can be chained to other stations delays, we identify the ones that are in delay.
    - weekdays: is a string of 7 bits containing information about the weekdays in which this trip is available, each bit represent a day of the week, if it's 1, the trip is present in that weekday, if it's 0 no.

We also defined two datatypes to help us modelling our model:

- location: it represent a position, it is composed by the latitude and the longitude used to know where facilities are located.
- schedule: it has property, the arrival and departure time of the bus.

5 Information Gathering

The step of information gathering in the producer side is to collect the data available in the internet, which can, in many times, be a very time consuming step. The data can only be available in websites, not available for download or not be available at all, requiring the producers to request the data to it's owners. The step on the consumer side requires less effort as it's data has clear and high quality metadata and are store in readly available repositories or the requests forms are clear.

5.1 Data and Knowledge Sources Identification

We identified three resources:

- Trentino OSM Places and his relative ontology, this is a formal resource providing a cleaned and classified OSM dataset with boundary of Trentino. It is organised in a folder tree representing categories and subcategories. The dataset is based on the Trentino OSM Lightweight Ontology. Both the data and the ontology can be accessed in the following catalog: DataScientia Trentino OSM Places
- Since the KGE 2022 - Urban Transportation was missing some data for achieving our purpose, we decided to use Trentino Trasporti Open data - Trentino Trasporti makes some of it's data available for everyone to use, this data only refers to transportation related to the Trentino region. From this dataset we extracted data relating the bus stops, routes and trips entities. The resources can be found in Trentino Transporti Open Data, they provide both urban and extra-urban data, for our purpose we used the urban one.
- The last one is an informal resource for data about the city areas in Trento and their population, taken from the City of Trento:
  - City areas: City of Trento Districts it contains the official subdivision of Trento with the name and the boundaries of each area, we used the KML(Keyhole Markup Language) file available in the link provided.
  - Population: City of Trento statistical data it contains PDF(Portable Document Format) files about the most recent survey on the population in Trento divided by the areas.
  - Services: City of Trento was used to identify where the services are located.

5.2 Resource Collection, Processing and Scraping

5.2.1 Data Cleaning and Formatting

This subsection we will explain how we extracted the data or produced the data on our dataset. All the code used in this step is available in the project Github repository github.com/R-R-Onzi/TTT_KGE.

1. facility: The data was generated using the services entities from the *Esercizi Commerciali* dataset within the City of Trento Districts dataset. This EType changed as initially we wanted to give some facilities more weight on predicting delays, but we did not found data in the Trentino OSM Places dataset that was in the city of Trento, so we couldn't use it.
   - name: It was extracted getting the WKT(Well-known text)[3] propriety from the dataset, that is the coordinates propriety of the data, with a generated polygon with the City of Trento dataset WKT propriety, and checking with opencv point polygon test function.
   - Product category: Will be extracted from the propriety *tipo* in the dataset. It's a new propriety created to identify the purpose of the facility.
   - Store name: Will be extracted from the propriety *nome* in the dataset.
   - Coordinates: It was extracted getting the WKT propriety from the dataset. As the coordinates data was in UTM(Universal Transverse Mercator coordinate), a converter was used based on an old code of university of Wisconsin website.
2. city_area: given the CSV(Comma-separated values) file provided by the city of Trento [4], we extracted the values we needed, combined them with the population data and saved them in a CSV file, the values extracted are:
   - id: Represent the official id attributed to a city area, field "numero_circ" in the CSV.
   - name: Represent the name of the city area, field "nome" in the CSV
   - boundaries: Has been extracted by the polygon object in the CSV.
   - population_density: Has been manually extracted by the PDFs identified [5].
3. routes, bus_stop and trips data will be collected by exploring the Trentino Transporti Open Data and combining it with the data produced. We tried to integrate with the data from KGE 2022, but as they didn't have the shape of the routes, we couldn't estimate the distance of the routes. The Trentino Trasporti data follows the GTFS(General Transit Feed Specification) format [6], here are more details about how we extracted data from it:
   - bus_route:
     - id: we used the route_id provided in the Trentino Trasporti data.
     - name: we used the short_name provided in the Trentino Trasporti data.
     - length: the length of the bus line was done getting the shape_pt_lon and shape_pt_lat proprieties from shapes entity from Trentino Trasporti open data related to the routes of each entity in the dataset and using Haversine formula of distance that considers the earth as a sphere creating the data in kilometres.
     - left_turns: we will need to annotate by hand, as its easier than making a program just for it. For time constraints as there were much variety on the routes, we decided to leave it for further work.

[3] Markup Language for vector representation

[4] https://gis.comune.trento.it/dbexport?db=base&sc=confini&ly=circoscrizioni&fr=csv

[5] https://www.comune.trento.it/Aree-tematiche/Statistiche-e-dati-elettorali/Statistiche/Studi-e-analisi/Dati-statistici-nelle-Circoscrizioni-di-Trento

[6] https://developers.google.com/transit/gtfs

  - right_turns: we will need to annotate by hand, as its easier than making a program just for it. For time constraints as there were much variety on the routes, we decided to leave it for further work.

- bus_trip:
  - id: we used the trip_id provided in the Trentino Trasporti data.
  - weekdays: created from the GTFS calendar.txt data of Trentino Trasporti by merging the weekdays enumerators from monday to sunday in a single string.
  - the first_bus_stop relation has been extracted by exploiting the stop_sequence field of GTFS stop_times.txt provided by Trentino Trasporti.
- bus_stop:
  - id, arrival and departure time, name and coordinates: have been extracted from Trentino Trasporti data using their respective fields.
  - the sequences next_busstop relations have been constructed by using the trip property in Trentino Transporti data.
  - the city_area relation: given latitude and longitude we identified the associated by checking in which area the coordinates belongs.

Delay and ”seats” and ”bus_id” properties of bus_ride we don’t have how to extract the data as Trentino Transporti doesn’t make it available for everyone.

5.3 Knowledge Modeling

Given the resources identified and the data collected, cleaned and formatted we need to associate a schema to each of them. All the schemas generated in this step can be found in the ”Phase 2 - Information Gathering/schemas” folder of the repository.

5.4 Integrate Data with Schemas Using Karma

Given the schemas created in the previous step and the data cleaned and formatted before, we linked them using Karma. All the linked data and schemas can be found in the ”Phase 2 - Information Gathering/schemas+data” folder of the repository. Since as explained before, we don’t have any data for the delay entity, we could not link any data to his schema. Also for the two datatypes location and schedule no data has been linked since they are not proper ETypes but just datatypes.

6 Language Definition

In this step both consumer and producer have the same objective: to identify the concepts of the Etypes and proprieties. What differ them is the output, the consumer side will generate only one KG and the producer will create more KGs. From the extracted ETypes in the last section, we formalized their concepts to GIDs(General Identifiers) from the UKC(Universal Knowledge Core), with them, the standardized the data can be identified by their concepts, rather than language, what makes the more data reusable. We were assigned, in case we don't find any concept with the same meaning, the GID numbers from 10000 to 11000 to add new concepts, if necessary.

6.1 Etypes

6.1.1 Facility

This entity was extracted by the city of Trento open data dataset where it was called *esercizi-commerciali* that would translate to commercial establishments, it was transformed in: a building or place that provides a particular service or is used for a particular industry, as our intention was to add more locations, not only commercial ones. So, for time constraints we couldn't add more and make concepts to focus exactly on our desired scope. Then we clarify the proprieties on 2 and their concepts on 3.

Table 2: Facility Declaration.

Proprieties	Propriety Type	Etype
id, name, coordinates, category	Data property	facilities
located	Object property	facilities

Table 3: Facility Concepts.

Concept Labels	Description
facility_GID-17982	a building or place that provides a particular service or is used for a particular industry
id_GID-10003	unique identifier, being it any entity, for it's collection
name_GID-2	a language unit by which a person or thing is known
coordinate_GID-32628	a number that identifies a position relative to an axis
category_GID-31828	a general concept that marks divisions or coordinations in a conceptual scheme
located_GID-93733	situated in a particular spot or position

6.1.2 City Area

It was produced using both data from the city of Trento demography data within the open data repository. Here we used the a Italian based definition of *circoscrizione*, as the data we are using is based on it. We clarify it's proprieties on 4 and their concepts on 5.

Table 4: City Area Declaration.

Proprieties	Propriety Type	Etype
id, name, boundary, population_density	Data property	city_area

Table 5: City Concepts.

Concept Labels	Description
city areaGID-10004	Territorial division over which the exercise of the functions of an office or local or decentralized, civil or ecclesiastical authority extends
id_GID-10003	unique identifier, being it any entity, for it's collection
name_GID-2	a language unit by which a person or thing is known
boundary_GID-73920	a line determining the limits of an area
population density_GID-118000	the concentration of people in the GPE area (measured in number of people per square km)

6.1.3 Bus Stop

Based on the KGE 2022 and data from city of Trento one main difference was the addition of the next bus stop on the same entity for the trip. For this reason, in particular, were created the datatypes schedule and location, location we used with facility as in the beginning both would be derived from a place entity, but facility was created to be more generic. We clarify it's proprieties on 6 and their concepts on 7.

Table 6: Stops Declaration.

Proprieties	Propriety Type	Etype
id, name, coordinates, schedule	Data property	bus_stop
located, next_bus_stop	Object property	bus_stop

Table 7: Stops Concepts.

Concept Labels	Description
bus stop_GID-45937	a place on a bus route where buses stop to discharge and take on passengers
id_GID-10003	unique identifier, being it any entity, for it's collection
name_GID-2	a language unit by which a person or thing is known
coordinate_GID-32628	a number that identifies a position relative to an axis
schedule_GID-103679	plan for an activity or event
next_bus_stop_GID-10010	the upcoming bus stop on a bus route where the vehicle is scheduled to make a stop
located_GID-93733	situated in a particular spot or position

6.1.4 Bus Trip

It's used to represent a single trip from a bus from the start bus stop to the last one, for it we also had to add week day as day of the week as it fits more the definition. We also have not found a suitable seats definition. We clarify it's proprieties on 8 and their concepts on 9.

Table 8: Trip Declaration.

Proprieties	Propriety Type	Etype
id, seats, bus_id, weekdays	Data property	bus_trip
serves, start_bus_stop	Object property	bus_trip

Table 9: Trip Concepts.

Concept Labels	Description
bus_trip_GID-10005	a single trip from a passenger bus from the start bus stop to the last one
id_GID-10003	unique identifier, being it any entity, for it's collection
seats_GID-10006	quantity of places for people inside a vehicle
day of the week_GID-80754	any one of the seven days in a week
bus id_GID-10007	unique identifier, for a passenger bus
serves_GID-10008	to fulfill or operate in accordance with a specific purpose or function
start_bus_stop_GID-10009	the initial bus stop on a bus route where a journey commences or originates

6.1.5 Bus Route

Bus Route represent the path followed by the bus, we clarify it's proprieties on 10 and their concepts on 11.

Table 10: Route Declaration

Proprieties	Propriety Type	Etype
id, name, length, left_turn, right_turn	Data property	bus_route

Table 11: Route Concepts

Concept Labels	Description
bus route_GID-45936	the route regularly followed by a passenger bus
id_GID-10003	unique identifier, being it any entity, for it's collection
name_GID-2	a language unit by which a person or thing is known
left_GID-1800	a turn toward the side of the body that is on the north when the person is facing east
right_GID-1799	a turn toward the side of the body that is on the south when the person is facing east
length_GID-28281	the property of being the extent of something from beginning to end

6.1.6 Delay

As the main concept of our project, we had to create definitions for both predicted delay and actual delay. We clarify it's proprieties on 10 and their concepts on 11.

Table 12: Delay Declaration.

Proprieties	Propriety Type	Etype
id, predicted delay, actual delay,	Data property	delay

Table 13: Delay Concepts

Concept Labels	Description
delay_GID-102604	cause to be slowed down or delayed
id_GID-10003	unique identifier, being it any entity, for it's collection
predicted delay_GID-10001	delay predicted by any mean of any event until it's end
actual delay_GID-10002	actual delay of any event until it's end

6.1.7 Location

Location in our case is just use as point in space to hold the latitude and longitude locations of places used in this project. We clarify it's proprieties on 14 and it's concepts on 15.

Table 14: Location Declaration.

Proprieties	Propriety Type	Etype
id, latitude, longitude	Data property	location

Table 15: Location Concepts

Concept Labels	Description
location_GID-132	a point or extent in space
id_GID-10003	unique identifier, being it any entity, for it's collection
latitude_GID-46264	an imaginary line around the Earth parallel to the equator
longitude_GID-46270	the angular distance between a point on any meridian and the prime meridian at Greenwich

6.1.8 Schedule

Schedule is used in the bus stop entity to hold the time the city of Trento estimates that the busses will arrive and departure from it. We clarify it's proprieties on **??** and it's concepts on 17.

Table 16: Schedule Declaration.

Proprieties	Propriety Type	Etype
id, arrival time, departure time	Data property	location

Table 17: Schedule Concepts

Concept Labels	Description
schedule_GID-103679	plan for an activity or event
id_GID-10003	unique identifier, being it any entity, for it's collection
arrival time_GID-80845	the time at which a public conveyance is scheduled to arrive at a given destination
departure time_GID-80846	the time at which a public conveyance is scheduled to depart from a given point of origin

7 Knowledge Definition

Here the consumer and producer steps are similar only diverging as last section 6 in the result being one instead of the many in the producer side, but this time, what is created are/is ontologies. Their objective is to create ontology/ies interoperable by reusability. In this section we aim at defining the knowledge structure for our project, to do so we will follow the kTelos process, consisting of these steps:

- Top-Down: reuse of a Lightweight Ontology (aligned to the UKC)
- Bottom-Up: modelling of a Teleology (aligned to the requirements modelled as CQs)
- Middle-Out: aligning of a Teleology grounded into the Lightweight Ontology to generate a Teleontology.
- Knowledge annotation.

In our case we used as Lightweight Ontology the Trentino OSM LWOntology. We will utilize the `Trentino_OSM_LW_` prefix to denote the concepts defined by this ontology in our ultimate teleology.

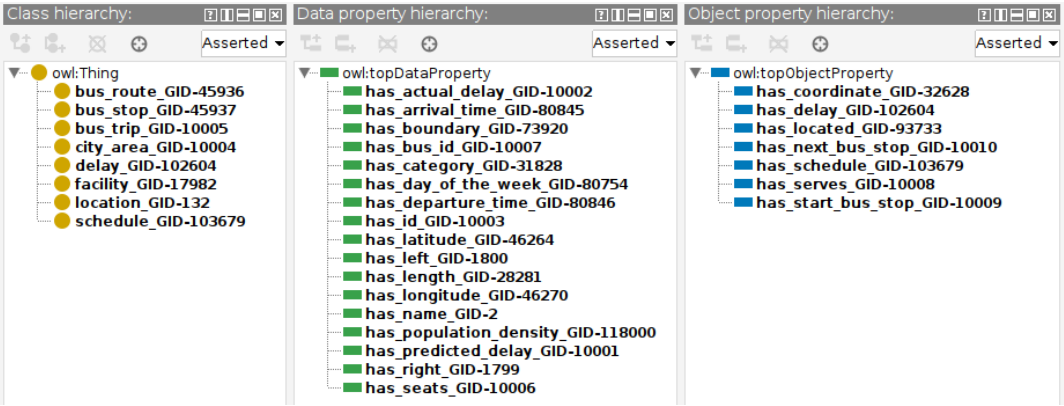
In the following subsections we will describe more in details the Teleology and Teleontology steps we performed. All the outputs of this phase is available in the project Github repository github.com/R-R-Onzi/TTT_KGE, in the *"Phase 4 - Knowledge Definition"* folder.

7.1 Teleology definition

Starting from the ER model in Figure 2, we formalized that model into our final teleology using Protégé. Here is the result in Protégé:

7.2 Teleontology definition

Given the teleology created and the lightweight ontology identified, we derived a teleontology in Protégé, in Figure 3 we report a diagram representing the result of this phase:



- owl:Thing
  - bus_route_GID-45936
  - bus_stop_GID-45937
  - bus_trip_GID-10005
  - city_area_GID-10004
  - delay_GID-102604
  - facility_GID-17982
  - location_GID-132
  - schedule_GID-103679

- owl:topDataProperty
  - has_actual_delay_GID-10002
  - has_arrival_time_GID-80845
  - has_boundary_GID-73920
  - has_bus_id_GID-10007
  - has_category_GID-31828
  - has_day_of_the_week_GID-80754
  - has_departure_time_GID-80846
  - has_id_GID-10003
  - has_latitude_GID-46264
  - has_left_GID-1800
  - has_length_GID-28281
  - has_longitude_GID-46270
  - has_name_GID-2
  - has_population_density_GID-118000
  - has_predicted_delay_GID-10001
  - has_right_GID-1799
  - has_seats_GID-10006

- owl:topObjectProperty
  - has_coordinate_GID-32628
  - has_delay_GID-102604
  - has_located_GID-93733
  - has_next_bus_stop_GID-10010
  - has_schedule_GID-103679
  - has_serves_GID-10008
  - has_start_bus_stop_GID-10009

Figure 2: Teleology created in Protégé

8 Data Definition

The Data Definition phase of the iTelos methodology takes in input the cleaned data resources and the teleontology we created in the previous phases and it combines them in our final KG. This phase is composed of three different activities:

- Entity Matching
- Entity Identification
- Entity Mapping

At producer side, the phase aims at producing the KG-based version for each dataset collected and handled during the previous phases, while at consumer side, we aim at producing the final KG, that needs to satisfy the purpose (i.e. Competency Questions). To do this in Karma, at producer side, the KGs files will remain separate, in order to be exploited for other purposes, while at consumer side, the files are composed together (copy-pasted) into a single file to define the single purpose-specific KG.

All the outputs are available in the project Github repository github.com/R-R-Onzi/TTT_KGE, in the *”Phase 5 - Data Definition”* folder. As Trentino Transporti does not publicly share delay data, we have developed a mock version for reference. We uploaded both the knowledge graphs with and without the delay mock in our GitHub repository. These are the resources available in Github for this phase:

- `R2RML-karma` is a folder containing all the R2RML Models of the transformations, modelling and mapping done in Karma.
- folder `without_delay_mock`:
  - `KGs.zip` contains the KG files for each dataset handled in the previous phases.
  - `ALL_KG.zip` contains the final KG for our project as the composition of all the KGs in `KGs.zip` together into a single file representing our Knowledge Graph.

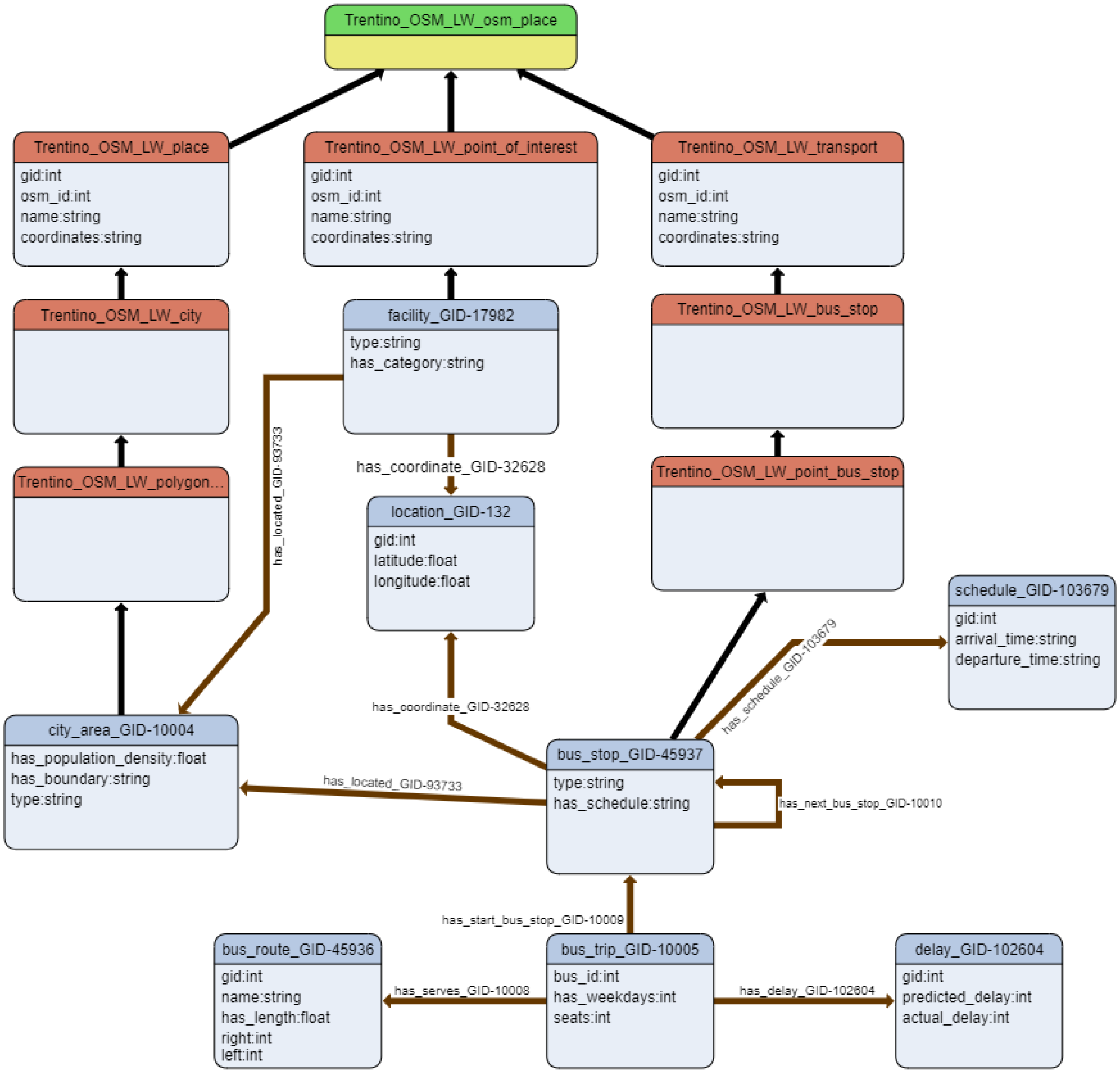


Figure 3: Teleontology diagram

- folder `with_delay_mock`, contains two zips, `KGs_with_delay_mock.zip` and `ALL_KG_with_delay_mock.zip`, same as the previous folder, but using the mock for the delays.

8.1 Entity Matching and Identification

8.1.1 Entity Matching

The first task involves Entity Matching: real world entities can be represented through different properties and properties values across different datasets.The goal is to synchronize these representations. In our case, we didn’t had this problem because we didn’t had entities coming from multiple datasets.

8.1.2 Entity Identification

Next we have Entity Identification, we need to identify the different entities, within a single dataset and within different datasets. In our case each entity already had an unique id assigned during the previous phases of the project.

8.2 Data Mapping

In the last step, we merged together the teleontology and the data created in the previous phases. To perform the Entity Mapping we useed the Karma tool, in Figure 4 there is an example of this operation for the facility data. Karma produced a KG file for each dataset, that are then composed together into a single file with a simple content copy and paste into our final KG.

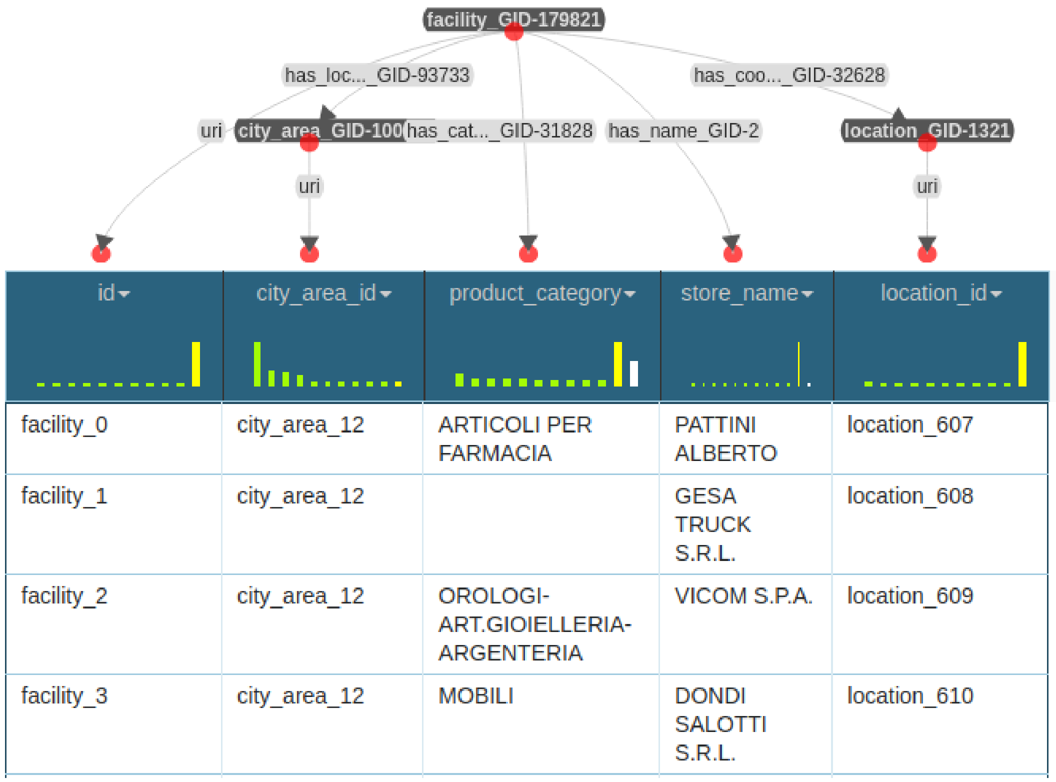
9 Evaluation

For the evaluation of our results, we are using the coverage methods. The teleology can be evaluated separating into ontology and competency questions queries. We also will evaluate it’s entities and proprieties for each query.

9.1 Coverage metric

In $Cov_E = \frac{|Cq_E \cap T_E|}{Cq_E}$ and in $Cov_p = \frac{|Cq_p \cap T_p|}{Cq_p}$ we evaluate the competency questions, where ”Cov” is the coverage and ”Cq” and ”T” are competency questions and teleology respectively.

In $Cov_E = \frac{|RO_E \cap T_E|}{RO_E}$ and in $Cov_p = \frac{|RO_p \cap T_p|}{RO_p}$ we evaluate the competency questions, where ”Cov” is the coverage and ”RO”(reference ontology) and ”T” are reference ontology and teleology respectively.



id	city_area_id	product_category	store_name	location_id
facility_0	city_area_12	ARTICOLI PER FARMACIA	PATTINI ALBERTO	location_607
facility_1	city_area_12		GESA TRUCK S.R.L.	location_608
facility_2	city_area_12	OROLOGI-ART.GIOIELLERIA-ARGENTERIA	VICOM S.P.A.	location_609
facility_3	city_area_12	MOBILI	DONDI SALOTTI S.R.L.	location_610

Figure 4: Example of mapping the facility entities to the teleontology using Karma

9.1.1 Results

- For the item ER evaluation we compare our created teleology and the ER created in the beginning of the project. In the ER we have 8 entities and in the competency questions we have 8 entities as well. The quantity size of the competency questions was modified from 6 to 8 to include schedule and location. $Cov_ER = \frac{|Cq_E \cap T_E|}{Cq_E} = \frac{|8|}{8} = 1$
- In the evaluation of the properties of the Teleology is made the same way as in 9.1. This results reflects on the lack of data in the Delay propriety and not having the seats number propriety for the buses in their route. $Cov_ER = \frac{|Cq_p \cap T_p|}{Cq_p} = \frac{|16|}{21} = .76$
- Now, with the evaluation of the teleontology and the ontologies we use the Trentino *Trasporti* Open data reference ontology, that contain 9 entities. This RO has 4 Entities in common with our Teleontology. $Cov_E = \frac{|RO_E \cap T_E|}{RO_E} = \frac{|4|}{9} = .44$
- At last, for the evaluation of the teleontology and the ontologies, now on the propriety side, we use the Trentino *Trasporti* Open data reference ontology, that contain 43 properties and object properties. This RO has 8 Proprieties in common with our Teleontology. $Cov_P = \frac{|RO_P \cap T_P|}{RO_P} = \frac{|8|}{46} = .17$

The coverage metric evaluates the final results, it does it using the intersection of the competencies questions and

- the final Knowledge Graph information statistics (like, number of etypes and properties, number of entities for each etype, and so on).
- Knowledge layer evaluation: the results of the application of the evaluation metrics applied over the knowledge layer of the final KG.
- Data layer evaluation: the results of the application of the evaluation metrics applied over the data layer of the final KG.
- Query execution: the description of the competency queries executed over the final KG in order to test the suitability of the KG to satisfy the project purpose.

9.2 KG Exploitation

The objective of this Knowledge Graph (KG) is to provide valuable information for forecasting bus delays. As described in the initial stages of this project, our aim was to merge data from Trento's urban bus transportation with factors relevant to delay prediction. Presented below are two query examples illustrating the integration of the two primary datasets integrated: facilities and population density information for the various city areas, as well as informations about the length of the bus trip.

In Section 9.2.1, CQ 1 combines bus data with details regarding the number of facilities in an area. On the other hand, in Section 9.2.2, we incorporate population density data along with route information, such as the length of the bus path.

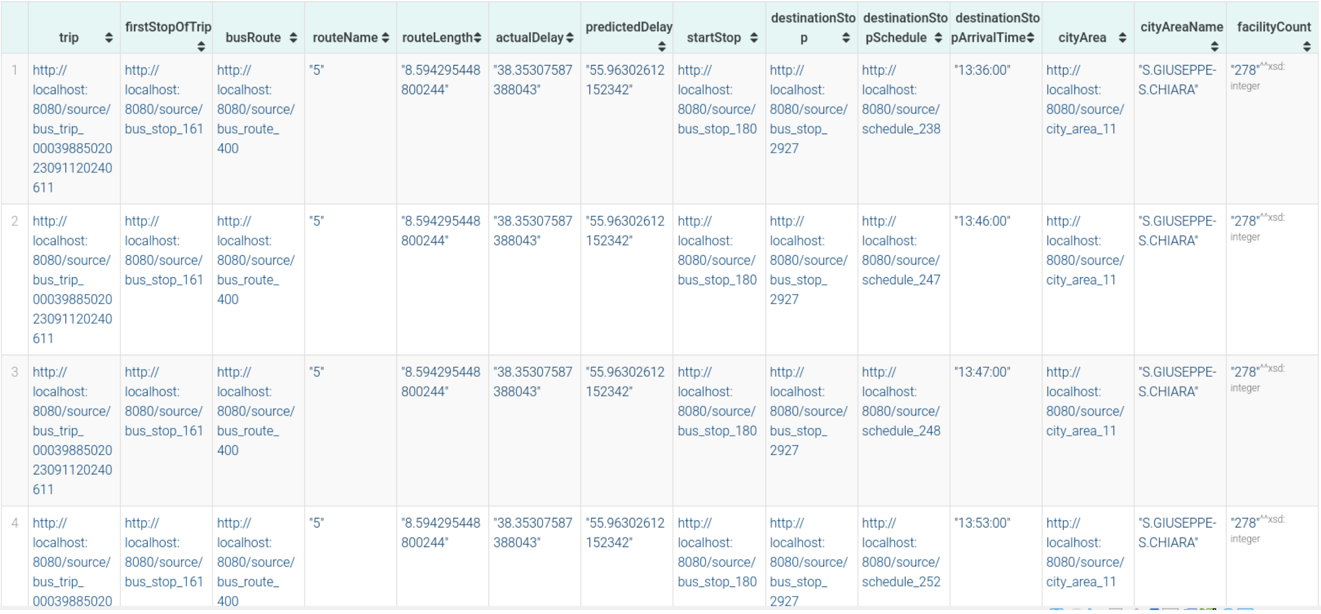
9.2.1 Query Competency Question 1

CQ 1 *Isabella, after lunch, wants to reach the city center, where she can find lot of shops to buy groceries to prepare sweets to her daughter. She wants to know how much time it is gonna take to reach the center, and arrive home for dinner.*

In Listing 1 we have the SPARQL query, while in Figure 5 we have the results of the query.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX etype: <http://knowdive.disi.unitn.it/etype#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT DISTINCT ?trip ?firstStopOfTrip ?busRoute ?routeName ?routeLength ?actualDelay
    ?predictedDelay ?startStop ?destinationStop ?destinationStopSchedule
    ?destinationStopArrivalTime ?cityArea ?cityAreaName (COUNT(?facility) as ?facilityCount)
WHERE{
    # find starting bus stop
    ?startStop rdf:type etype:bus_stop_GID-45937;
            etype:has_name_GID-2 ”Spré Pinara” .

    # find destination bus stop
    ?destinationStop rdf:type etype:bus_stop_GID-45937;
            etype:has_name_GID-2 ”Venezia \”Port’aquila\”” .

    # get schedule for arrival time
    ?destinationStop etype:has_schedule_GID-103679 ?destinationStopSchedule .
    ?destinationStopSchedule etype:has_arrival_time_GID-80845 ?destinationStopArrivalTime .

    # filter after lunch
    FILTER (STRDT(STR(?destinationStopArrivalTime), xsd:time) > ”13:30:00”^^xsd:time &&
        STRDT(STR(?destinationStopArrivalTime), xsd:time) < ”14:30:00”^^xsd:time)

    # get the trip
    ?firstStopOfTrip rdf:type etype:bus_stop_GID-45937 .
    ?trip rdf:type etype:bus_trip_GID-10005;
          etype:has_serves_GID-10008 ?busRoute;
          etype:has_delay_GID-102604 ?delay;
          etype:has_start_bus_stop_GID-10009 ?firstStopOfTrip.

    # the trip should pass through the start bus stop and then through the destination
    ?firstStopOfTrip etype:has_next_bus_stop_GID-10010* ?startStop .
    ?startStop etype:has_next_bus_stop_GID-10010* ?destinationStop .

    # get route information
    ?busRoute etype:has_name_GID-2 ?routeName;
              etype:has_length_GID-28281 ?routeLength .

    # get delay of the trip
    ?delay etype:has_actual_delay_GID-10002 ?actualDelay;
           etype:has_predicted_delay_GID-10001 ?predictedDelay .

    # get destination area and facilities in that area
    ?destinationStop etype:has_located_GID-93733 ?cityArea .
    ?cityArea etype:has_name_GID-2 ?cityAreaName .

    ?facility rdf:type etype:facility_GID-17982;
              etype:has_located_GID-93733 ?cityArea .
}
GROUP BY ?trip ?firstStopOfTrip ?busRoute ?routeName ?routeLength ?actualDelay
?predictedDelay ?startStop ?destinationStop ?destinationStopSchedule
?destinationStopArrivalTime  ?cityArea ?cityAreaName
LIMIT 5
```

Listing 1: SPARQL query for CQ 1



	trip	firstStopOfTrip	busRoute	routeName	routeLength	actualDelay	predictedDelay	startStop	destinationStop	destinationStopSchedule	destinationStopArrivalTime	cityArea	cityAreaName	facilityCount
1	http://localhost:8080/source/bus_trip_0003988502023091120240611	http://localhost:8080/source/bus_stop_161	http://localhost:8080/source/bus_route_400	"5"	"8.594295448800244"	"38.35307587388043"	"55.96302612152342"	http://localhost:8080/source/bus_stop_180	http://localhost:8080/source/bus_stop_2927	http://localhost:8080/source/schedule_238	"13:36:00"	http://localhost:8080/source/city_area_11	"S.GIUSEPPE-S.CHIARA"	"278"^^xsd:integer
2	http://localhost:8080/source/bus_trip_0003988502023091120240611	http://localhost:8080/source/bus_stop_161	http://localhost:8080/source/bus_route_400	"5"	"8.594295448800244"	"38.35307587388043"	"55.96302612152342"	http://localhost:8080/source/bus_stop_180	http://localhost:8080/source/bus_stop_2927	http://localhost:8080/source/schedule_247	"13:46:00"	http://localhost:8080/source/city_area_11	"S.GIUSEPPE-S.CHIARA"	"278"^^xsd:integer
3	http://localhost:8080/source/bus_trip_0003988502023091120240611	http://localhost:8080/source/bus_stop_161	http://localhost:8080/source/bus_route_400	"5"	"8.594295448800244"	"38.35307587388043"	"55.96302612152342"	http://localhost:8080/source/bus_stop_180	http://localhost:8080/source/bus_stop_2927	http://localhost:8080/source/schedule_248	"13:47:00"	http://localhost:8080/source/city_area_11	"S.GIUSEPPE-S.CHIARA"	"278"^^xsd:integer
4	http://localhost:8080/source/bus_trip_00039885020	http://localhost:8080/source/bus_stop_161	http://localhost:8080/source/bus_route_400	"5"	"8.594295448800244"	"38.35307587388043"	"55.96302612152342"	http://localhost:8080/source/bus_stop_180	http://localhost:8080/source/bus_stop_2927	http://localhost:8080/source/schedule_252	"13:53:00"	http://localhost:8080/source/city_area_11	"S.GIUSEPPE-S.CHIARA"	"278"^^xsd:integer

Figure 5: Query results for CQ 1

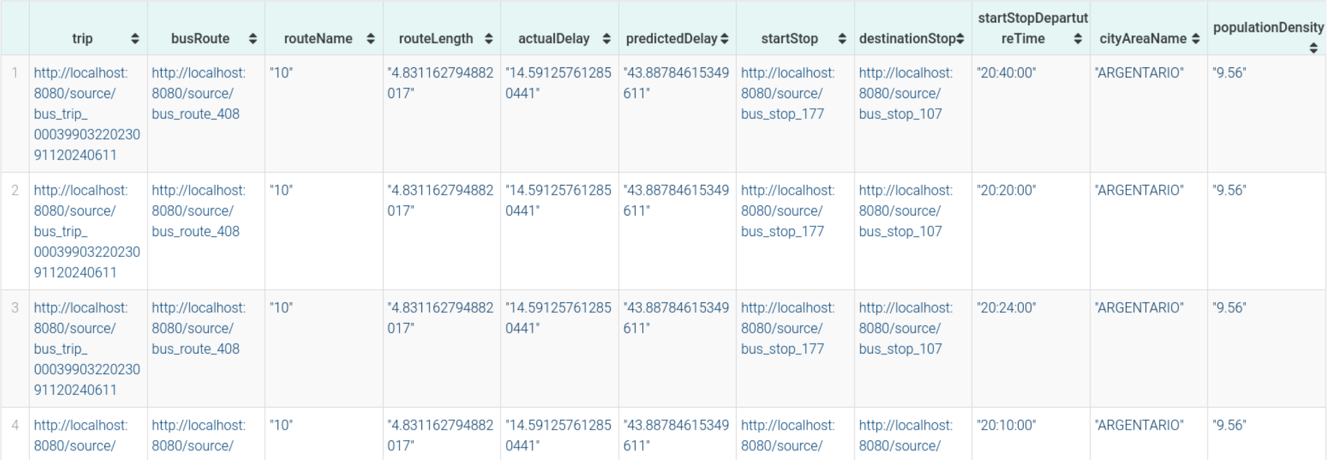
9.2.2 Query Competency Question 5

CQ 5 *After enjoying a dinner with his friends, Giovanni decide to head to one of his friend's houses in Martignano. They are fortunate to be right on schedule for the last bus. The buses to Martignano are usually punctual, as the area had fewer residents compared to the city center, resulting in less traffic. They want to confirm if the bus will run on time so that they can arrive at their destination on time.*

In Listing 2 we have the SPARQL query, while in Figure 6 we have the results of the query.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX etype: <http://knowdive.disi.unitn.it/etype#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT DISTINCT ?trip ?busRoute ?routeName ?routeLength ?actualDelay
?predictedDelay ?startStop ?destinationStop ?startStopDepartutreTime
?cityAreaName ?populationDensity
WHERE{
    ?startStop rdf:type etype:bus_stop_GID-45937;
        etype:has_name_GID-2 ”Cervara \”Port’aquila\”” .

    ?destinationStop rdf:type etype:bus_stop_GID-45937;
                etype:has_name_GID-2 ”Martignano P.Zza Menghin” .

        # get the departure time from the start stop
    ?startStop etype:has_schedule_GID-103679 ?schedule .
    ?schedule etype:has_departure_time_GID-80846 ?startStopDepartutreTime .

    # departure after dinner
    FILTER (STRDT(STR(?startStopDepartutreTime), xsd:time) > ”20:00:00”^^xsd:time)

    ?firstStopOfTrip rdf:type etype:bus_stop_GID-45937 .
    ?trip rdf:type etype:bus_trip_GID-10005;
          etype:has_serves_GID-10008 ?busRoute;
          etype:has_delay_GID-102604 ?delay;
          etype:has_start_bus_stop_GID-10009 ?firstStopOfTrip.

    ?firstStopOfTrip etype:has_next_bus_stop_GID-10010* ?startStop .
    ?startStop etype:has_next_bus_stop_GID-10010* ?destinationStop .

    ?busRoute etype:has_name_GID-2 ?routeName;
              etype:has_length_GID-28281 ?routeLength .
    FILTER (?routeName = ”10”) # to martignano

    ?delay etype:has_actual_delay_GID-10002 ?actualDelay;
           etype:has_predicted_delay_GID-10001 ?predictedDelay .

    ?destinationStop etype:has_located_GID-93733 ?cityArea .
    ?cityArea etype:has_name_GID-2 ?cityAreaName;
              etype:has_population_density_GID-118000 ?populationDensity.
}
LIMIT 5
```

Listing 2: SPARQL query for CQ 1



	trip	busRoute	routeName	routeLength	actualDelay	predictedDelay	startStop	destinationStop	startStopDepartutreTime	cityAreaName	populationDensity
1	http://localhost:8080/source/bus_trip_0003990322023091120240611	http://localhost:8080/source/bus_route_408	"10"	"4.831162794882017"	"14.591257612850441"	"43.88784615349611"	http://localhost:8080/source/bus_stop_177	http://localhost:8080/source/bus_stop_107	"20:40:00"	"ARGENTARIO"	"9.56"
2	http://localhost:8080/source/bus_trip_0003990322023091120240611	http://localhost:8080/source/bus_route_408	"10"	"4.831162794882017"	"14.591257612850441"	"43.88784615349611"	http://localhost:8080/source/bus_stop_177	http://localhost:8080/source/bus_stop_107	"20:20:00"	"ARGENTARIO"	"9.56"
3	http://localhost:8080/source/bus_trip_0003990322023091120240611	http://localhost:8080/source/bus_route_408	"10"	"4.831162794882017"	"14.591257612850441"	"43.88784615349611"	http://localhost:8080/source/bus_stop_177	http://localhost:8080/source/bus_stop_107	"20:24:00"	"ARGENTARIO"	"9.56"
4	http://localhost:8080/source/	http://localhost:8080/source/	"10"	"4.831162794882017"	"14.591257612850441"	"43.88784615349611"	http://localhost:8080/source/	http://localhost:8080/source/	"20:10:00"	"ARGENTARIO"	"9.56"

Figure 6: Query results for CQ 5

10 Metadata Definition

The metadata is the data about the data, so having extense and concise amount, improves it‘s reusability by making the possible future users having an opportunity to understand it a priori in some amount. For This purpose the metadata was separated in three sections: people, project and dataset

10.1 People resources metadata description

The people resources section aim to identify the people involved in the project. link

Identifier	firstName	lastName	email	nationality	gender	affiliation	personalWebpage
To be added	Eugenio	Ferrari	eugenio.ferrari-1@studenti.unitn.it	Italian	Male	Unitn	None
To be added	Rubens	Rissi Onzi	rubens.rissionzi@unitn.studenti.it	Brazilian, Italian	Male	Unitn	None

10.2 Project resources metadata description

The Project resources metadata can be found in the link

Title	Trentino Territory & Transportation 2024
URL	https://r-r-onzi.github.io/TTT_KGE/
Keywords	Transportation, Trento, Trentino, Teritory
Type	To be added
Description	This project focused in store and predict possible and actual delays in the trentino region
StartDate	04/10/2023
EndDate	09/01/2024
FundingAgency	None
Input	To be added
Output	To be added
Coodinator	Simone Bocca
Observations	To be added

10.3 Dataset resources metadata description

The Dataset resources metadate can be found in the link, since this part of the metadata is too big to be reported in a table we just leave the link to directly access it.

11 Open Issues

In this section we will present the main issues of the project.

11.1 No data available for Delay

There was no delay data available for the project to use about delays of buses in the city of Trento, for this reason we used mocks for the queries. One solution for the problem would be asking the trentino authorities to provide or start to making the data as it would be very interesting for the population of the city.

11.2 No good data on services on the city of Trento

We used a very generic dataset for the facilities on the city of Trento, as we have not found data more specific on the city of Trento. We had the Trentino OSM Places dataset to use as a base, but all of the specific facilities were from outside of Trento city proper, so we used a generic dataset provided by the city of Trento.

The Trentino OSM Places just had a part of the data of the whole trentino region lacking the data of it‘s biggest city on it, as a solution we could wait for the data to be added in to it or even do it ourselves, but due to the time contraint of the project, it was not done.

11.3 Right and left turns

We had intention to use right and left turns on the dataset, as a paper before mentioned raised our awareness that this type of action made by the bus was an direct factor on bus delays.

This task was not only made because of the time constraint of the project, as we had the points of the project, but not only, we also had the description on google of each route, what could mean that this data could be created manually in the worse case scenario.