



UNIVERSITY
OF TRENTO - Italy

Know
dive



DIPARTIMENTO DI INGEGNERIA E SCIENZA DELL'INFORMAZIONE

– KNOWDIVE GROUP –

Geospatial Knowledge Graph

Document Data:

November 27, 2020

Reference Persons:

Bortolon Matteo
Mattedi Daniele
Povoli Sergio
Tessari Michele

© 2020 University of Trento
Trento, Italy

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.

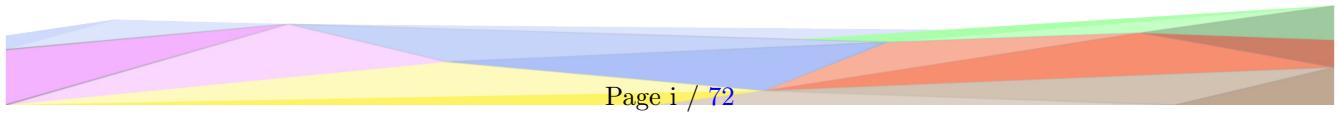


Contents

1 Knowledge Graph Development Process	1
1.1 Scope Definition	1
1.1.1 Problem contest definition	1
1.1.2 Personas	2
1.2 Inception	4
1.2.1 CQs definition	4
1.2.2 Initial Datasets description	8
1.2.3 Datasets metadata documentation	9
1.2.4 Datasets collection process	19
1.2.5 Inception level evaluation	19
1.3 Informal Modeling	20
1.3.1 Schema level	20
1.3.2 Data level	28
1.3.3 Informal Modeling Evaluation	40
1.4 Formal Modeling	40
1.4.1 Schema level	40
1.4.2 Classes	44
1.4.3 Data Property	44
1.4.4 Object Property	44
1.4.5 Enumeration Classes	45
1.4.6 Focus on Classes	45
1.4.7 Data level	60
1.4.8 Formal Modeling Evaluation	72

Revision History:

Revision	Date	Author	Description of Changes
0.1	12.10.2020	Sergio Povoli	Initial personas description
0.2	13.10.2020	Matteo Bortolon	Scope description
0.3	15.10.2020	Sergio Povoli	CQs persons 1,2
0.4	15.10.2020	Matteo Bortolon	CQs persons 3,4
0.5	18.10.2020	Michele Tessari	Initial dataset description
0.6	20.10.2020	Matteo Bortolon, Sergio Povoli	Data Objects
0.7	20.10.2020	Daniele Mattedi	Datasets metadata documentation
0.8	21.10.2020	Matteo Bortolon	Inception level evaluation
0.9	01.11.2020	Daniele Mattedi, Michele Tessari	Dataset management process
1.0	03.11.2020	Daniele Mattedi	Variance respect Inception dataset
1.1	04.11.2020	Daniel Mattedi, Michele Tessari	Dataset metadata documentation
1.2	04.11.2020	Matteo Bortolon	ER description added
1.3	05.11.2020	Sergio Povoli	Formalized the variation to CQs



1 Knowledge Graph Development Process

The aim of this part of the document is describing the development process of the knowledge graph. We decided to use iTelos methodology for developing the knowledge graph because it will guide us step by step during the development process. At the beginning of the document we describe the contest, the problem we need to solve and the personas who can take benefit from our knowledge graph. In the inception phase we describe the Competence Query and an initial list and description of the datasets we used.

1.1 Scope Definition

In this section we describe the contest of the knowledge graph and some stereotypes of person that represent some potential user of our knowledge graph.

1.1.1 Problem contest definition

As GeoSpatial data we intend all the data which are explicitly or implicitly connect to a location. And without thinking about it a lot of our everyday life is connected to some location information: for instance, if my wife has gone and we want to know something about her destination, or when we are deciding where to go out dinner and we want to try a new restaurant. The public administration is one of the entities which mostly uses geospatial data; a business estimate that 90% of the PA data are bounded with a location.¹

As Geospatial field is very large, we decide to focus on the real estate market. In year 2019 in Italy , according to the report from the Italian Tax Bureau, 603'541 residential real estate units were sold and bought². These units are equal to the surface of the city of Bolzano. The real estate trading sector is valued 97.5 billions of euros per year. These numbers do not consider the rents market. In 2019 the new rent agreements reached the staggering amount of 1 million and 7 hundred units. In 2019 the 10% of residential rental units in Italy was rented; this flox generates a market of around 10 billion per year.⁴

In Italy, the biggest part of residential real estate units rented and traded are owned by physical persons. The Italian real estate market status is very different from other nations in Europe, where rented and traded residential buildings are mostly owned by enterprises. In particular, these include banks or investment funds, which are usually protected by shields of ad-hoc front company. This difference has caused the proliferation of the real estate agents in Italy: in Italy there are 43'698 real estate agents for 59 million people, whereas in Germany, where most of the building is rented or bought by businesses, agents are 23'780 for 80 million people⁵. Searching between a large number of real estate operators can represent a considerable expenditure of energy for who wants to rent or buy a unit. To resolve this problem various intermediation portal was born. In these portals, real estate agents (or also the owner directly) list the house. These portals are mainly owned by national companies, instead of being owned by foreign companies (usually this happen in other sectors, such us search engines).

Listings on this portals normally describe very well features of real estate unit, but they normally fail to describe the connected services. These include, for example, the internet connectivity available, the quality of schools in the nearby of the real estate unit or the presence of parking where people could leave their car. These data are

¹<https://philly.curbed.com/2017/11/7/16617296/philadelphia-properties-land-vacant-lot-atlas-map-tool>

²<https://www.agenziaentrante.gov.it/portale/documents/20143/262485/StatisticheOMIRES1202020200605.pdf/3af07746-906c-45af-5853-da521605de03>

³The given data do not include the Italian territories which use the Grundbuch (at the moment they are the provinces of Trento and Bolzano)

⁴<https://www.idealista.it/news/immobiliare/residenziale/2020/05/07/140165-affitti-italia-focus-su-trend-del-mercato-proprietari-e-riduzione-del-canone-di>

⁵<https://www.libropensieroimmobiliare.com/evoluzione-di-un-agente-immobiliare-italiano-dati-e-considerazioni/>

sometimes sparse and difficult to retrieve, as a consequence the real estate agents sometimes do not know or have limited knowledge about these connected services. Sometimes agents know information but avoid to give them to the customer because it could lower property value. These considerations can sometimes be crucial since if the client know them it could choose a unit over instead of another.

Our idea for the project is to build a knowledge graph which can help people looking for a real estate unit to buy or rent. In this way they could easily consult listing and quality of these connected services. Even if the knowledge graph can be useful in almost every location in the world (excluding Antarctica), we decide on a system which work in the province of Trento (Italy), in the first phase, for the amounts of publicly available datasets. For some type of data we decide to focus only on the city of Trento, because they are not available at province level (e.g. civic numbers). The hypothetical user of our project is someone who want to rent or buy a real estate unit. Collecting money from multiple clients can be difficult, therefore also intermediation portals are a potential customer for our knowledge base. The intermediation portals can use our knowledge base to help customers compare the various listings available on the portal. This could potentially be provided through AI-based tools, which suggest the best unit for the customer needs.

As a future development, the knowledge base can be extended to be used by other figures in the buy/rent of a real estate unit market.

1.1.2 Personas

In this subsection is presented the scenario of competence of the project and in particular the different types of Personas that could be the actors of this scenario. All the 4 different types of user presented are characterized by the need for looking for an apartment or house to buy or rent in the geographical area of competence of the project, These people, however, have different attitudes and consequently different needs.

Giuseppe is 24 years old, he is a student of industrial engineering at the University of Trento in Povo. Despite attending university in Povo, it would be very important for Giuseppe to find accommodation in the city in order to be close to essential services and university social life, without neglecting the need to move with public transport every day to reach the university outside the city. Giuseppe is not able to have a constant source of income and his accommodation expenses are paid entirely by his family, for this reason he is looking for a very economical situation. As well as most of his fellow students he is easily satisfied with a place in a room shared by other students. Since he does not have a private transport, he is not looking for an apartment with a private or public parking nearby. His university life and his hobbies take priority choosing the apartment. He would prefer to find accommodation in a home away from elementary schools and kindergartens and possibly away from the railway to have a quiet place to study. To be effective in the studio Giuseppe would need an house which is equipped with a broadband internet connection allowing him to increase his productivity, as well as having a quick access to many online multimedia content he watches in his spare time. Giuseppe also has a passion for fitness and is used to train 3 times a week. For this reason he would like to live near a gym.

Paola and Mario, 43 and 45 years old, respectively, are the parents of Luigi and Angela, twins of 8 years old. They need to find a new housing near the city to get closer to their workplace. Paola is lawyer in a firm based in the city center and prefers to move with her private vehicle because she has a covered parking near her work. His spouse Mario, a computer consultant, is used to go to work by car; however during summer he is inclined to use bike-sharing services offered in the city due to its strong sporting nature. For this reason the couple is looking for accommodation with a large garage but at the same time not too far from the bike-sharing services possibly. The couple's children are lively twins and for this reason Paola would like to find a house with a private garden

and possibly in a quiet and safe area near the essential services for children (playground, elementary and middle school). Economically, the family is wealthy and prefers to spend an important amount of money for their new home as long as it meets with their needs about safety, size and proximity of services for children. Mario would also like the house to be located in a geographically interesting location for evening excursions, in fact he is a fan of running and trekking and after work would have the pleasure of leaving home a couple of hours to keep trained and run in a natural and unpolluted environment.

Giovanna is 35 years old and works remotely for a large software house in Milan. Giovanna, having the possibility to stay at home and work, is used to change house periodically and would like to find an apartment in the surroundings of Trento, which is a very dear place for her family origins. The accommodation she is looking for is an apartment to rent where she can work remotely and invite many guests. Giovanna loves to meet people and have many social interactions, because she thinks that social exchange enrich her. For these reasons Giovanna is looking for an apartment in an area full of people and with many possibilities to create communities and social exchanges of the city. She has no needs for private or public parking as she is a lover of a green lifestyle and would prefer to reach essential services by walking or cycling. For this reason she is looking for a home which is near to essential services such as supermarkets and pharmacies, but would also like to easily reach places of culture such as libraries, cinemas and theaters, because of her innate passion for art and culture. Given her working position, it is essential for Giovanna that her home be equipped with good network coverage and Internet connection. Last but not least, Giovanna loves outdoor sports and in particular she has been practicing climbing for a few years. For this reason, an useful information would be an overview of the rock cliffs in Trentino Alto Adige and their distances from the house.

Luciana and Corrado are a loving elderly couple, she is 73 years old and he is 75 years old, retired. The couple is looking for a setting for their winter and not searching for a classic vacation home; rather, a place where they can feel at home even when they go on vacation, as they plan to spend almost a third of the year in this residence. The couple, being very well off, is looking for a spacious and comfortable house in a quiet neighborhood and as panoramic as possible. Although they have their own means of transport in Trentino, they want to relax and enjoy the peace of a small town. In the search for a house some features are very important for them. The house should be in a quiet area and with a private or public parking nearby and well equipped with essential services and with some ad hoc services to make them feel at home. In fact, Luciana and Corrado, because of their strong faith, think the house should be close to a church, in order to participate in Christian community life. In addition, they would like to have a few old people's homes nearby to spend a few days in company and participate in the activities organized by these centers. Luciana and Corrado have been passionate about winter sports since their youth and this was one of the reasons why they want to look for a house in Trentino-Alto Adige. In particular, in order to choose a house, they are interested in knowing where they are and what ski facilities are available within a few dozen kilometers from their future vacation home.

In the following a table that summarizes the personas is presented.

Persona	Description	Usage
---------	-------------	-------

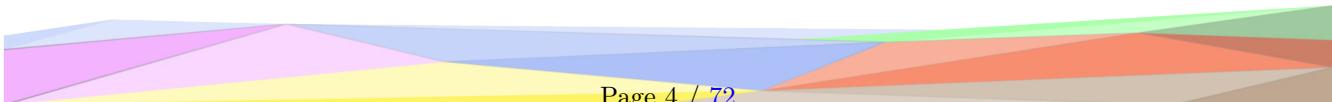
<i>Giuseppe</i>	24 y.o. engineering student who loves university social life and going to the gym but he has not a private car.	Giuseppe is looking for a student room in a quiet environment in the city (away from the railway and main roads) where he can study, equipped with fast internet connection and with the following facilities nearby: essential services (supermarket, bank, pharmacy), the bus stop 5/, a gym, library and nightclubs
<i>Paola and Mario</i>	Paola and Mario, 43 and 45 years old, respectively, are the parents of Luigi and Angela (twins of 8 years old).	Paola and Mario are, together with their children, looking for an house in the city in an uncrowded place. When choosing a house, the couple carefully assesses whether the services nearby meet their needs as parents and their hobbies. In particular, they would like to have them nearby: playground, elementary and middle schools, public parking and bike-sharing service to facilitate their mobility, routes and excursions to do after work
<i>Giovanna</i>	Giovanna is 35 years old and works remotely for a large software house in Milan. She loves culture.	Giovanna works at home, that's why she doesn't need a car and prefers to reach interesting places on foot or by bicycle. Besides having a good Internet connection, she wants to be located in an area of the city where there is a chance to meet many people. She loves social and cultural exchanges. She would like to look for a house close to essential services (supermarkets, drugs, banks) and close to the places she prefers to attend in her free time: cinemas and theaters. To keep fit Giovanna practises climbing and therefore it would be very useful for her to know which cliffs are available in Trentino in order to plan her weekend adventures.
<i>Luciana and Corrado</i>	Luciana and Corrado are a loving elderly couple, she is 73 years old and he is 75 years old, retired. They search for a vacation home.	Luciana and Corrado are looking for a vacation home which should be in an uncrowded place. They are looking for a house which should be close to a parking lot, churches and old people's centers. In addition, the couple has a passion for skiing and would like to know which skiing areas are near their home. In addition, because of their age, the couple would like to live in a well-served area of emergency health services.

1.2 Inception

This section first describe the Competency Queries in a informal format, then the datasets collected and the relative metadata will be described.

1.2.1 CQs definition

In this section we try to think what our example users can ask to the knowledge graph. Based on the datasets selected, we formalized a complete list of competency queries. This queries describe the need of the people as write in the storytelling.



Person	Number	Question	Action
Giuseppe	1.1	Where is the closest library ?	The system given the unit position, returns the position of the nearest library
Giuseppe	1.2	How many library are there within a 5km radius ?	The system given the unit position, returns the number of library within the 5 km
Giuseppe	1.3	What are the nearest supermarket in the area ? What size is it ?	The system given the unit position, returns the distance of the nearest supermarket and its size
Giuseppe	1.4	What kind of internet connection is available in the apartment?	The system given the unit position, returns the level of available internet connection
Giuseppe	1.5	Is there a pharmacy within a radius of 1Km?	The system given the unit position, returns the presence or absence of a pharmacy within a radius of 1Km
Giuseppe	1.6	Is the house in a quiet enough place ?	The system given the unit position, returns the major roads and railways closest to the house indicating the distance from the house.
Giuseppe	1.7	How far is it to the nearest bus stop where bus number 5 runs?	The system, given the location of the unit, returns the distance to the nearest bus stop where bus number 5 goes by. For this bus stop and line the system returns also the timetable.
Giuseppe	1.8	Where is the closest gym?	The system given the unit position, returns the position of the nearest gym
Giuseppe	1.9	What bank branches with an ATM are available in the nearby?	The system given the unit position, returns the distance of the nearest branch for each bank
Giuseppe	1.10	Where is the nearest bar ?	The system given the unit position, returns the position of the nearest bar
Giuseppe	1.11	How far is the train station ?	The system given the unit position, returns the distance of the train station
Giuseppe	1.12	Is there a bus station nearby where in the morning the line 5 passes ?	The system given the unit position, returns the position of the nearest bus stop where in the morning the line 5 passes and for this bus stop and line returns also the timetable
Paola and Mario	2.1	Is the apartment in a quiet place?	The system given the unit position, returns the major roads and railways closest to the house indicating the distance from the house.
Paola and Mario	2.2	Where is the closest elementary school to the house?	The system given the unit position, returns the position of the nearest elementary school
Paola and Mario	2.3	How far is the nearest middle school?	The system given the unit position, returns the distance of the nearest middle school
Paola and Mario	2.4	How many playgrounds are there within 2 km from the house?	The system, given the location of the unit, returns the number of playing fields within 2 km and for each playground it lists the present games
Paola and Mario	2.5	Is the house in a crowded area?	The system given the unit position, counts the number of houses in 3 km
Paola and Mario	2.6	Is there a bike-sharing service nearby?	The system given the unit position, returns the presence or absence of a bike-sharing service within a radius of 1Km and returns also the number of available bike in those racks
Paola and Mario	2.7	What are the bike paths within 1 km?	The system given the unit position, returns the bike paths within 1km
Paola and Mario	2.8	How many excursion or bike paths are there in the vicinity of the house (maximum 3 km)?	The system given the unit position, returns the number of different excursion and bike paths within 3 km.
Paola and Mario	2.9	What are the excursion paths within a radius of 3 km with a estimated duration of less than two hours?	The system given the unit position, select all the paths with a estimated duration of less than 2 hours and returns from this list the ones within 3km
Paola and Mario	2.10	Where is the closest parking lot to the house with a hourly fee of less than a 1.00 €?	The system given the unit position, check for all the parking with a fee less than 1.00 €, then return the nearest parking lot.
Paola and Mario	2.11	How far is the center of the city?	The system given the unit position, returns the distance of the city center.

Paola and Mario	2.12	Where is the closest middle school to the house?	The system given the unit position, returns the position of the nearest middle school
Paola and Mario	2.13	Which high schools are within a radius of 2 km ?	The system given the unit position, returns the name of the high schools within a radius of two kilometers
Paola and Mario	2.9	What are the excursion paths in a radius of 3 km from the unit with a positive height difference of less than 200m ?	The system given the unit position, select all the paths with a height difference of less than 200m and returns from this list the ones within 3km
Giovanna	3.1	Know what type of Internet is available in the apartment	The system given the unit address, return the level of available internet connection
Giovanna	3.2	I want to live in place full of people to develop social exchanges. How many people live near the house ?	The system given the unit position, returns the number of people within 1 km.
Giovanna	3.3	What banks have a branch in the neighbourhood?	The system given the unit position and address, search the nearest bank branches, maximum one for each bank
Giovanna	3.4	What are the post offices and their timetables within a radius of 3 km ?	The system given the unit position, search for all the post offices within 3 km and their opening hours
Giovanna	3.5	Where is the nearest ambulatory?	The system given the unit position and address, returns the position of the nearest ambulatory
Giovanna	3.6	Which are the supermarkets within a radius of 2 km ? How big they are?	The system given the unit position, search the supermarkets within the 2km and returns a list of supermarket and for each one the area and the name
Giovanna	3.7	How far is the nearest pharmacy?	The system given the unit position, returns the distance of the nearest pharmacies
Giovanna	3.8	Which libraries are the closest?	The system given the unit position, returns the library within 1 km
Giovanna	3.9	Where is the nearest cinema?	The system given the unit position, search for the nearest cinemas
Giovanna	3.10	Where is the nearest theaters?	The system given the unit position, search for the nearest theaters
Giovanna	3.11	What are the cliffs within a radius of 20 km ?	The system given the unit position, search for the nearest cliffs and return the position of the cliffs within of 20 km
Luciana and Corrado	4.1	I want to live in a quiet place. Where is the nearest railway line ?	The system given the unit position, search for the nearest piece of rail and return the distance between the unit and it
Luciana and Corrado	4.2	I want to live in a quiet place. Where are the nearest main roads ?	The system given the unit position, search for the nearest pieces of major road and return the distance between the unit and it
Luciana and Corrado	4.3	i want live in a small neighborhood	The system given the unit position, count the number of house in 5 km
Luciana and Corrado	4.4	How far is the nearest public parkings?	The system given the unit position, search for the nearest parkings with no fee or fee equal to zero and return the distance between the unit and the parking. It also return the capacity of the parking.
Luciana and Corrado	4.5	Where is the closest church?	The system given the unit position, search for the nearest church and return the position of the church
Luciana and Corrado	4.6	Ho far is the nearest center for elderly?	The system given the unit position, search for the center for elderly and return the distance between the unit and the center for elderly
Luciana and Corrado	4.7	Which ski areas are available within a radius of 35 km ?	The system given the unit position, search for ski area within 35 km and return the position and the distance between the unit and the ski area. For each ski area in order to provide more information the system returns also the website of the ski area. Moreover the system returns also the price of a daily ticket

Luciana and Corrado	4.8	Which are the hospitals in Trentino ?	The system returns all the hospitals in Trentino and for each of them return the name, the position and the distance to the unit position.
Luciana and Corrado	4.9	Are there ambulatory clinics within 3km from the house ?	The system given the unit position, check if there are some ambulatory clinics within 3km. If there are any ambulatory clinics return the name and position for each of them.
Luciana and Corrado	4.10	How far is the nearest hospital ?	The system given the unit position, search for the nearest hospital and return the distance between the hospital and the unit position.
Luciana and Corrado	4.11	Which are the 3 ski areas with the greatest number of kilometers of red slopes ? For each of these ski areas I would need to know the location, the website to know more about the location and the distance from the house.	The system given the unit position, search for the 3 ski areas with the highest number of black ski slopes and return a list with the name of the area, the location, the website and the distance from the location.
Luciana and Corrado	4.12	What kind of slopes are there in the ski area closest to the house ?	The system given the unit position, search for the nearest skairea and return the and for that returns the various types of difficulty of slopes present and their respective kilometers.
Luciana and Corrado	4.13	What are the nordic skiing trails within 35 km ?	The system given the unit position, returns a list with all the nordic skiing trails within 35 km. For each of them returns also the difficulty.

Each query consult one or more type of data. The types of data needed to resolve the query and the connection between each type and the query that use it are given in the following table:

Types:	Properties:	Notes:	CQ using the type:
<i>Library</i>	Position [coordinate], Name [string]		1:1,2 3:9
<i>Pharmacy</i>	Position [coordinate], Name [string], Opening hours [string]		1:5 3:8
<i>School</i>	Position [coordinate], Name [string], Type [enum]		2:2,12,13
<i>Sports Facility</i>	Position [coordinate], Name [string]		1:8
<i>Bar</i>	Position [coordinate], Name [string]		1:10
<i>Building</i>	Position [coordinate]		2:5 4:3
<i>Park</i>	Position [coordinate], types of games [list of string]		2:4
<i>Bike path</i>	Initial Point [coordinate], End point [coordinate]	It combine dataset of province of Trento with that of the city of Trento	2:7,8
<i>Excursion path</i>	Initial Point [coordinate], End Point [coordinate], Height difference in m [int], Estimated duration [duration], Length in m [int]		2:8,9
<i>City Center</i>	Position [coordinate], name [string]		2:11
<i>Major Road</i>	Name [string], Initial Point [coordinate], End Point [coordinate], Type [enum]		1:6 2:3 3:3 4:2
<i>Post Office</i>	Position [coordinate], Opening hours [string]		3:5
<i>Ambulatory</i>	Position [coordinate]		3:6
<i>Theater</i>	Position [coordinate], Name [string]		3:11
<i>Church</i>	Position [coordinate]		4:5
<i>Elderly center</i>	Position [coordinate], Name [string]		4:6
<i>Ski area</i>	Position [coordinate], Name [string], Total lenght [float], Km Blue slopes [float], Km Red slopes [float], Km Black slopes [float], Price [decimal]		4:7,11,12
<i>Cinema</i>	Position [coordinate], Name [string]		3:10
<i>Bank Branch</i>	Position [coordinate], Name of the bank [string], ATM [bool]		1:9 3:4
<i>Internet</i>	Address [composed field], Level_at_2019 [enum])		1:4 3:1
<i>Supermarket</i>	Position [coordinate], Name [string], Area [double]		1:3 3:7
<i>Train line</i>	Initial Point [coordinate], End Point [coordinate]		1:6 2:1 3:2 4:1
<i>Train Station</i>	Position [coordinate], Name [string], Tratta [Train line]		1:11
<i>Bus Stop</i>	Position [coordinate], Name [string], Lines [list of string]	Lines are the bus lines that go through this stop	1:7,12

<i>Bike-sharing</i>	Position [coordinate], rack slots [int]		2:6
<i>Cliff</i>	Position [coordinate]		3:12
<i>Car parking</i>	Position [coordinate], capacity [int], fee [decimal]		2:10 4:4
<i>Ski slopes</i>	Pointchains [coordinate], name [string], type [string], difficulty [string]		4:13
<i>Stop Times (GTFS)</i>	arrivalTime [timestamp], departureTime [timestamp], stopSequence [int], calendar [Calendar GTFS]		2:7,12

1.2.2 Initial Datasets description

The first source of datasets considered, *OPENdata Trentino*⁶, is a big web portal with a lot of different public data of the province of Trento. From there we took these datasets:

- 257 datasets (for 149 municipalities) of the **locations and points of interest**⁷: for every town in the province of Trento is provided a list in json of the relevant public locations like schools, parks, libraries, parking lots etc. with some additional information attached (name, coordinates, address, descriptions, photos, telephone number) and a GEO version (not present in all towns) with a lot more locations but with less information (only name, coordinates, address). Because of the high number of datasets to download, it has been written a script in python to automate the operation, and another to manage the files.
- 7 datasets of **Bikesharing stations**⁸: the list in json of all the bikesharing stations in 7 towns of Trentino provided by the public transport service containing the name, position, address and the number of bike slots for every station.
- 3 datasets with **nursery**,⁹ **elementary**¹⁰ and **middle school**¹¹ information: the list of Trento's school with name, position, address, number of subscribers, number of staff for every school.
- piste_cyclabili.csv for the **bike trails**¹² in Trento with the position, type, address and the length of the paths.
- civici.web.json for **civic numbers**¹³ of buildings in the Municipality of Trento.

However the **high school** data is not present in the openData Tretino website, and to complete the missed data we did scraping in *comunicittà.it*¹⁴ containing less but still relevant informations (private or public school information, study paths) and saving the content in csv format.

The second big source of datasets used to get a large amount of geo-spatial informations is *OpenStreetMap*¹⁵, the biggest free license world map collaborative project. All the data were extracted using *Overpass Turbo*¹⁶: a tool to make query for specific data extraction from OpenStreetMap. The datasets extracted (all exported in geojson):

- **areaski.geojson**: all the areas where is possible to make ski activities in Trentino.
- **bank.geojson**: all the banks located in Trentino.
- **building.geojson**: the buildings polygons divided in municipality
- **busstops.geojson**: all the bus stops with coordinates, the operator and (sometimes) the bus line.
- **cinema.geojson**: the list of the Trentino's cinema with coordinates and some relevant information where present (3D available, contact informations, number of rooms and other)
- **citycenter.geojson**: provide informations about cities or villages boundary in Trentino and where each city centre is located.
- **climb.geojson**: the cliffs in Trentino where is possible to do climbing activities.
- **parking.geojson**: car parks with the capacity.
- **park.geojson**: all different types of public parks (also dog parks).

⁶dati.trentino.it

⁷dati.trentino.it/dataset?tags=luoghi+e+punti+di+interesse

⁸dati.trentino.it/dataset/stazioni-bike-sharing-emotion-trentino

⁹dati.trentino.it/dataset/scuole-dinfanzia

¹⁰dati.trentino.it/dataset/scuole-elementari

¹¹dati.trentino.it/dataset/scuole-media-inferiori

¹²dati.trentino.it/dataset/piste-cyclabili-open-data

¹³dati.trentino.it/dataset/comune-di-trento-numeri-civici

¹⁴www.comunicitta.it/scuole-secondarie-di-secondo-grado/comune-di-trento-22205

¹⁵www.openstreetmap.org

¹⁶overpass-turbo.eu

- **pharmacy.geojson:** list pf pharmacies with position and (where present) the timetables.
- **post.office.geojson:** post offices with position and (where present) the timetables
- **railway.geojson:** train stations and railways of Trentino.
- **roads.geojson:** all the roads of Trentino.
- **skislopes.geojson:** the ski slopes location in the Trentino.
- **supermarket.geojson:** supermarkets of Trentino.
- **trails.geojson:** list of path (open to all non-motorized vehicles) or trails of Trentino

In order to satisfy the request of a good internet connection of some personas, we took a dataset from *Infratel Italia s.p.a.*¹⁷ with all the information about the availability and the **quality of internet**¹⁸ per home in Trentino.

1.2.3 Datasets metadata documentation

Here the metadata documentation of the fields of the datasets. However, the number of fields in some dataset is very high (up to 193), then the tables shows a max of the 25 most important fields per dataset:

- **locations and point of interest:**

Fields	Type	Description
<i>Titolo</i>	string	name of the location
<i>address</i>	string	street address
<i>description</i>	string	short description of the location
<i>image</i>	image	image of the location
<i>email</i>	string	official email of the location
<i>phone number</i>	string	official phone contact of the location
<i>url</i>	string	website of the location
<i>info</i>	string	short infos
<i>gps</i>	string	coordinates and accurate address

- **locations and point of interest (GEO version):**

Fields	Type	Description
<i>id</i>	int	identification code
<i>coordinates</i>	float[2]	latitude and longitude of the location
<i>properties</i>	object	main properties of the location (name, address)

- **bikesharing-METADATA.json:**

Fields	Type	Description
<i>name</i>	string	name of the bike station
<i>address</i>	string	geographical address of the bike station
<i>id</i>	string	identifier
<i>bikes</i>	int	realtime data with the number of the present bikes
<i>slots</i>	int	realtime data with the number of the free slots
<i>totalSlots</i>	int	total number of slot bikes
<i>position</i>	int[2]	coordinates of the station

- **piste_cliclabili-METADATA.json:**

¹⁷www.infratelitalia.it

¹⁸www.infratelitalia.it/archivio-documenti/documenti/esiti-consultazione-2020-conclusa-una-prima-analisi-dei-dati-forniti-dagli-operatori

Fields	Type	Description
<i>WKT</i>	list int	coordinates of the paths
<i>tipo</i>	string	name of the typology of path (not useful)
<i>fumetto</i>	string	name of the path
<i>descrizione</i>	string	name of the path (equivalent to 'fumetto')
<i>tipologia</i>	string	path dedicated only to cycle or also pedestrian
<i>tratto isolato ciclabile</i>	int	meters without path dedicated to cycle
<i>tratto isolato ciclopedonale</i>	int	meters without path dedicated to cycle or pedestrian
<i>anno</i>	int	building year of cycle path

- **nursery_elementary_and_middle_school_METADATA.json:**

Fields	Type	Description
<i>WKT</i>	string	position in WKT system
<i>civico_alf</i>	int	civic number
<i>destra</i>	string	address
<i>sobborgo</i>	string	suburb
<i>scuola</i>	string	name of the school
<i>fumetto</i>	string	name of the school
<i>iscritti</i>	int	number of subscribers
<i>personale</i>	int	number of staff
<i>mail</i>	string	e-mail information
<i>telefono</i>	string	telephone number

- **high_school_METADATA.json:**

Fields	Type	Description
<i>name</i>	string	name of the school
<i>address</i>	string	address
<i>school type</i>	string	private or public school
<i>study paths</i>	string	study path offers

- **internet_quality_trento_METADATA.json:**

Fields	Type	Description
<i>cod_egonciv</i>	int	EGON civic code
<i>regione</i>	string	Only the TRENTO ALTO ADIGE region
<i>provincia</i>	string	only Trento province
<i>comune</i>	string	town
<i>frazione</i>	string	fraction of the town
<i>via</i>	string	street address
<i>civico</i>	int	civic code
<i>barrato</i>	char	sub-civic code
<i>class_19</i>	string	2019 internet coverage
<i>class_22</i>	string	2022 internet coverage estimation

- **areaski_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	name that identifies a unique object
<i>landuse</i>	string	describes the primary use of areas of land
<i>description</i>	string	describes the type of object
<i>area</i>	string	date rarely existing (value - yes/no)
<i>name</i>	string	name of data
<i>alt_name</i>	string	Another name or names by which the feature is known
<i>name:de</i>	string	german name of data
<i>name:it</i>	string	italian name of data
<i>name:lld</i>	string	Ladin name of data

<i>type</i>	string	type of data
<i>website</i>	string	website
<i>wikidata</i>	string	The ID of the Wikidata item about the feature
<i>ref:pat:cat</i>	string	identification
<i>ref:pat:id</i>	string	identification
<i>@relations</i>	map	describes the commercial relations with other ski areas

- **bank_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	identification code
<i>building</i>	string	type fo building
<i>name</i>	string	name of the facility
<i>type</i>	string	type of element
<i>atm</i>	boolean	indicate the presence of an atm
<i>automated teller machine</i>	boolean	see atm
<i>operator</i>	string	indicate the operator of the facility
<i>brand</i>	string	indicate the brand of the facility
<i>addr:houseNumber</i>	string	the house number component of the address
<i>addr:street</i>	string	the street component of the address
<i>source</i>	string	the source of the data
<i>opening_hours</i>	string	the opening hours of the facility
<i>contact:website</i>	string	the website where contact the facility
<i>building:levels</i>	integer	the level above ground of the facility
<i>brand wikidata</i>	string	wikidata identifier code
<i>brand wikipedia</i>	string	the wikipedia page of the bank branch
<i>wheelchair</i>	boolean	describe if it is accesible for wheelchair users
<i>addr:city</i>	string	the city component of the address
<i>addr:postcode</i>	string	the postcode component of the address
<i>roof:levels</i>	integer	the level of the roof
<i>roof:shape</i>	string	the shape of the roof
<i>drive_through</i>	boolean	describe if the facility offer drive through
<i>addr:full</i>	string	the full address of the facility
<i>addr:country</i>	string	the country component of the address
<i>website</i>	string	the website of the facility
<i>phone</i>	string	the phone of the facility

- **busstop_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	name that identifies a unique object
<i>highway</i>	string	name that identifies objective type
<i>public_transport</i>	string	name that denotes stop positions and platforms of public transport
<i>name</i>	string	name that identifies the name of the bus-stop
<i>shelter</i>	boolean	identifies the presence or absence of the shelter
<i>bus</i>	boolean	Access values describe restrictions or not of the bus in the bus-stop
<i>bench</i>	boolean	identifies the presence or absence of the shelter
<i>alt_name:lld</i>	string	another unofficial but locals fequently name in local dialects
<i>name:it</i>	string	Italin name of the bus-stop
<i>name:lld</i>	string	Ladin name of the bus-stop
<i>departures_board</i>	string	name that identifies the type of time table or departures present in the bus stop
<i>operator:old_name</i>	string	name that identifies company who is in charge of the service at the bus stop
<i>route_ref</i>	liststring	name that identifies a list of all bus lines that serve the stop
<i>bus_routes</i>	liststring	name that identifies a list of Flixbus that serve the stop

<i>created_by</i>	string	name that identifies the computer program (editor or script) which made the changes
<i>wheelchair</i>	string	identifies if the bus stop is suitable to be used with a wheelchair and a person with a disability who uses another mobility device (like a walker)
<i>note:tactile-paving</i>	boolean	identifies if there is Tactile paving
<i>covered:source</i>	string	identifies how the information was collected
<i>ref</i>	liststring	name that identifies a list of all bus lines that serve the stop
<i>name:de</i>	string	German name of the bus stop
<i>addr:city</i>	string	name identifying the name of the city where the bus stop is located
<i>addr:postcode</i>	int	number identifying the postcode of the city where the bus stop is located
<i>addr:street</i>	string	name identifying the name of the street where the bus stop is located
<i>alt_name</i>	string	alternative name of the bus stop
<i>amenity</i>	string	Describes useful and important facilities for visitors and residents
<i>bin</i>	boolean	identifies presence or absence of a waste basket

- **cinema_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	Identifies univocally the object
<i>amenity</i>	string	Describes useful and important facilities for visitors and residents
<i>building</i>	boolean	Is used to mark a given object as a building
<i>name</i>	string	Identifies the name of the cinema
<i>email</i>	string	Identifies the e-mail related to the cinema
<i>phone</i>	string	Identifies the e-mail related to the cinema
<i>source</i>	string	Identifies the information source
<i>website</i>	string	Identifies the website related to the cinema
<i>addr:city</i>	string	Identifies in term of cities the position of the cinema
<i>addr:country</i>	string	Identifies in term of state the position of the cinema
<i>addr:housenumber</i>	int	Identifies the house number of the cinema
<i>addr:postcode</i>	int	Identifies the postcode of the cities where the cinema is located
<i>addr:street</i>	string	Identifies the street name where the cinema is located
<i>cinema:3D</i>	boolean	Identifies the possibility or not to watch 3D projection
<i>contact:phone</i>	string	Identifies the house number of the cinema
<i>contact:website</i>	string	Identifies the website related to the cinema
<i>drive_in</i>	boolean	Indicates if a cinema is a drive-in cinema
<i>operator</i>	string	Indicates the main operator of the cinema
<i>ref:vatin</i>	string	VAT identification number of an object
<i>screen</i>	int	Identifies the number of screens
<i>wheelchair</i>	string	Identifies the possibility or not to access with a wheelchair

- **city_center_METADATA.json:**

Fields	Type	Description
<i>admin_level</i>	integer	the level of the administration roles tree
<i>boundary</i>	string	the type of boundary
<i>name</i>	string	the name of the area
<i>ref:ISTAT</i>	string	the reference to the ISTAT code
<i>ref:catastro</i>	string	the reference to the codice catastale
<i>type</i>	string	the type of the object
<i>wikidata</i>	string	link to the related wikidata object
<i>wikipedia</i>	string	the wikipedia page of the facility
<i>@id</i>	string	name that identifies a unique object

<i>name:cim</i>	string	the name in Cimbro
<i>name:it</i>	string	the name in Italian
<i>population</i>	string	the population of this administrative area
<i>old_name:de</i>	string	the old name in german
<i>natural</i>	string	describe the type of nature inside the boundary
<i>name:de</i>	string	the new name in german
<i>name:mhn</i>	string	the name in mócheno
<i>alt_name:de</i>	string	an alternative name in german
<i>reg_name</i>	string	the name of the region
<i>name:lld</i>	string	the name in ladin
<i>postal_code</i>	string	the postal code of the administrative area
<i>old_name</i>	string	the old name of the administrative area
<i>name:left</i>	string	the name from the left side
<i>name:right</i>	string	the name from the right side
<i>source</i>	string	the source of the information
<i>created_by</i>	string	who created this set

- **climb_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	Used to univocally identifies the object
<i>climbing:rock</i>	string	Identifies the typology of rock
<i>name</i>	string	Identifies the name of the climb
<i>natural</i>	string	Is used to describe wide variety of physical geography, geological and landcover features
<i>sport</i>	string	Is used to identify one or more sports which can be played
<i>type</i>	string	Is used to set a relation's type
<i>building</i>	string	Indicates the presence or absence and the type of a building.
<i>leisure</i>	string	is for places people go in their spare time.
<i>addr:country</i>	string	Indicates the city in the address information
<i>addr:housenumber</i>	string	Indicates the housenumber in the address information
<i>addr:postcode</i>	int	Indicates the postcode in the address information
<i>addr:street</i>	string	Indicates the postcode in the address information
<i>email</i>	string	Indicates the email related to the climb
<i>fax</i>	string	Indicates the fax related to the climb
<i>opening_hours</i>	string	Indicates the opening hours of the climb spot
<i>phone</i>	string	Indicates the phone related to the climb
<i>tourism</i>	string	Identifies the specific interest to tourists including places to see, places to stay, things and places providing information and support to tourists.
<i>website</i>	string	Indicates the website related to the climb
<i>climbing:bolted</i>	string	Provides a first indication how well the routes are secured in terms of anchors
<i>climbing:multipitch</i>	string	Indicates if there are multi-pitch routes and how many multi-pitch routes there is
<i>climbing:orientation</i>	string	Identifies orientation of the rock/wall face.
<i>climbing:quality</i>	string	Identifies the quality of the rock/ice.
<i>climbing:sport</i>	string	Indicates the practicable sports
<i>fee</i>	boolean	Indicates the presence or absence of fee
<i>outdoor</i>	boolean	Indicate if the climb is outside or not
<i>climbing:boulder</i>	boolean	Indicates the presence or absence of boulder

- **park_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	name that identifies a unique object
<i>leisure</i>	string	type of facility inside
<i>type</i>	string	type of data
<i>name</i>	string	the name of the facility

<i>wikidata</i>	string	link to the related wikidata object
<i>landuse</i>	string	describe the usage of the land
<i>access</i>	string	describe the access description
<i>operator</i>	string	the operator of the facility
<i>area</i>	boolean	describe if the area do not have any physical boundary
<i>source</i>	string	the source of the data
<i>alt_name</i>	string	an alternative name for the park
<i>short_name</i>	string	a short name for the park
<i>start_date</i>	date	the opening date of the facility
<i>amenity</i>	string	this variable describe the facility present at the park
<i>created_by</i>	string	the person that created the facility in openstreetmap
<i>old_name</i>	string	the old name of the facility
<i>barrier</i>	string	indicate the type of barrier that limit the facility
<i>tourism</i>	string	indicate why this facility can be of particular interest for the tourist
<i>wikipedia</i>	string	the reference to the wikipedia page
<i>ele</i>	integer	the elevation on the sea level
<i>natural</i>	string	describe the type of nature inside the boundary
<i>sport</i>	string	describe one or more sport that can be played in the facility
<i>surface</i>	string	type of surface in the park
<i>leaf_type</i>	string	describe the type of leave in the facility
<i>wheelchair</i>	boolean	wheelchair service is present
<i>baby</i>	string	specify if the item is tag friendly

- **parking_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	identification code
<i>amenity</i>	string	For describing useful and important facilities for visitors and residents. Facilities include for example toilets, telephones, banks, pharmacies, prisons and schools.
<i>fee</i>	boolean	The fee tag is for specifying whether a fee is usually charged for a service, or for access.
<i>parking</i>	string	specify the type of parking facility.
<i>access</i>	string	describe restrictions on the use of highways and other transportation routes
<i>park_ride</i>	boolean	Park and ride facilities are parking lots with public transport connections that allow commuters and other people heading to city centres to leave their vehicles and transfer to a bus, rail system (rapid transit, light rail, or commuter rail), or carpool for the remainder of the journey.
<i>surface</i>	string	provide additional information about the physical surface
<i>capacity</i>	int	max capacity of the parking lot
<i>name</i>	string	name of the location
<i>barrier</i>	string	type of barrier of the parking lot
<i>source</i>	string	source of the data
<i>capacity:disabled</i>	boolean	Defines whether or not dedicated disabled parking spaces are available
<i>highway</i>	string	main key used for identifying any kind of road, street or path
<i>operator</i>	string	company of the parkig lot
<i>supervised</i>	string	indicates that there is a person who supervises the place
<i>wheelchair</i>	boolean	is wheelchair accessible?
<i>service</i>	string	describe details about types of 'service' roads, railways or waterways; or the service or services offered by a business.
<i>layer</i>	int	describe vertical relationships between crossing or overlapping features
<i>landuse</i>	string	describe the primary use of land by humans
<i>addr:housenumber</i>	int	civic code
<i>addr:postcode</i>	int	postal code of the location

<i>addr:street</i>	string	street address of the location
<i>opening_hours</i>	string	day and hours of opening
<i>smoothness</i>	string	a classification scheme regarding the physical usability of a way for wheeled vehicles, particularly regarding surface regularity/flatness.
<i>website</i>	url	wbesite of the location
<i>motorcycle</i>	string	Legal access restriction for motorcycles.

- **pharmacy_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	identification code
<i>amenity</i>	string	For describing useful and important facilities for visitors and residents. Facilities include for example toilets, telephones, banks, pharmacies, prisons and schools.
<i>building</i>	string	type of building
<i>name</i>	string	name of the location
<i>addr:city</i>	string	city of the location
<i>addr:housenumber</i>	int	civic code
<i>addr:postcode</i>	int	postcode of the location
<i>addr:street</i>	string	street address of the location
<i>drive_through</i>	boolean	do it has the drive through?
<i>healthcare</i>	string	type of healtcare service
<i>dispensing</i>	boolean	specify whether a pharmacy dispenses prescription drugs or not
<i>opening_hours</i>	string	day and hours of opening
<i>operator</i>	string	name of the company
<i>ref:vatin</i>	string	VAT identification code
<i>wheelchair</i>	boolean	is wheelchair accessible?
<i>addr:country</i>	string	country of the location
<i>email</i>	string	email of contact
<i>phone</i>	string	phone number contact
<i>contact:website</i>	url	website of the location
<i>source</i>	string	source of the data
<i>website</i>	url	website of the location

- **post_office_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	name that identifies a unique object
<i>addr:city</i>	string	city component of the address
<i>addr:country</i>	string	the country component of the address
<i>addr:housenumber</i>	string	the house number component of the address
<i>addr:postcode</i>	string	the postal code component of the address
<i>addr:street</i>	string	the street component of the address
<i>amenity</i>	string	this variable describe the usage of the building
<i>building</i>	boolean	describe if the facility is a building
<i>contact:website</i>	string	the website where contact the facility
<i>name</i>	string	the name of the facility
<i>operator</i>	string	the name of the facility operator
<i>atm</i>	boolean	if the facility have an atm
<i>building:colour</i>	string	the colour of the building
<i>building:levels</i>	integer	the number of above-ground levels of the buildlings
<i>building:part</i>	string	describe a part of the building where the attribute are different
<i>height</i>	double	the height of the building
<i>opening_hours</i>	string	the opening hours of the facility
<i>roof:colour</i>	string	the color of the facility roof
<i>roof:shape</i>	string	the shape of the facility roof
<i>brand</i>	string	brand of the post office

<i>brand:wikidata</i>	string	wikidata identifier code
<i>brand:wikipedia</i>	string	brand associated with country
<i>addr:suburb</i>	string	the suburb component of the address
<i>contact:fax</i>	string	the fax to contact the facility
<i>contact:phone</i>	string	the phone to contact the facility
<i>description</i>	string	a description of the building

- **railway_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	identification code
<i>addr:city</i>	string	city of the location
<i>addr:postcode</i>	string	postcode of the location
<i>addr:street</i>	string	street address of the location
<i>building</i>	string	type of building
<i>name</i>	string	name of the location
<i>network</i>	string	network administration name
<i>operator</i>	string	operator administration name
<i>railway</i>	string	railway or station classification
<i>public_transport</i>	string	type of public transport of the location
<i>train</i>	boolean	it's a trainstation
<i>electrified</i>	string	description of the object electrified
<i>maxspeed</i>	int	speed limit of the railway
<i>name:de</i>	string	name in German
<i>passenger_lines</i>	int	number of lines
<i>railway:track_class</i>	string	The Track class of the railway line
<i>railway:traffic_mode</i>	string	Lines dedicated to passenger trains (might have some exceptions, especially temporally i.e. freight only during 0200-0330).
<i>ref</i>	int	The reference number of the railway line the track belongs to.
<i>usage</i>	string	usage of the rail line
<i>voltage</i>	int	Voltage of the railway electrification system
<i>bridge</i>	string	What kind of bridge carrying the track is
<i>layer</i>	int	To describe the vertical relationship to other bridges and features.
<i>start_date</i>	string	Date the Interlocking Tower was put into service
<i>embankment</i>	boolean	Is the track elevated using earth or concrete dam
<i>service</i>	string	type of the track
<i>tunnel</i>	boolean	The track is in a tunnel

- **roads_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	identification code
<i>highway</i>	string	type of road (primary, secondary, tertiary)
<i>name</i>	string	name of the road
<i>oneway</i>	boolean	is oneway
<i>ref</i>	string	road code
<i>junction</i>	string	type of junction (roundabout)
<i>surface</i>	string	type of road surface (asphalt, paved)
<i>lanes</i>	int	number of lanes
<i>maxspeed</i>	int	speed limit
<i>nat_ref</i>	string	road code
<i>toll</i>	boolean	toll present
<i>cycleway:both</i>	boolean	have 2 way cycleway
<i>old_ref</i>	string	road code
<i>bridge</i>	string	it has a bridge
<i>source:maxspeed</i>	string	source of the speed limit data
<i>maxweight</i>	float	max weight in tons

<i>tunnel</i>	boolean	it has a tunnel
<i>foot</i>	boolean	can go by foot
<i>tunnel:name</i>	string	name of the tunnel
<i>access</i>	string	type of access
<i>horse</i>	boolean	horse can go
<i>length</i>	int	length of the road in mt
<i>incline</i>	string	road inclination information
<i>tunnel:length</i>	int	tunnel length in mt
<i>disabled</i>	boolean	the road is disable

- **skislopes_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	identification code
<i>colour</i>	color	graphic color
<i>name</i>	string	name of the ski slopes
<i>network</i>	string	network administration of the ski slopes
<i>piste:type</i>	string	type of the skiing available (nordic, downhill)
<i>website</i>	url	link of the website of the skislope
<i>name:de</i>	string	name in German
<i>operator</i>	string	operator of the skislope
<i>distance</i>	float	distance between skislopes
<i>piste:difficulty</i>	string	difficulty level of the skislope (easy, intermediate, advanced)
<i>piste:grooming</i>	string	type of skiing available (classing, skating)
<i>source</i>	string	source of the data
<i>piste:name</i>	string	name of the piste
<i>sport</i>	string	sport available in the location
<i>bridge</i>	boolean	it's present a bridge

- **supermarket_METADATA.json:**

Fields	Type	Description
<i>@id</i>	string	identification code
<i>building</i>	string	type of supermarket
<i>name</i>	string	name of the supermarket
<i>opening_hours</i>	string	days and hours of opening
<i>payment:cash</i>	boolean	accept cash
<i>payment:debit_cards</i>	boolean	accept debit card
<i>shop</i>	string	type of shop
<i>source</i>	string	source of data
<i>addr:city</i>	string	city of the location
<i>addr:country</i>	string	country if the location
<i>addr:housenumber</i>	int	civic code
<i>addr:postcode</i>	int	postcode
<i>addr:street</i>	string	street of the location
<i>operator</i>	string	operator of the supermarket (coop, conad, desper ...)
<i>ref:vatin</i>	string	value added tax identification number
<i>brand</i>	string	brand of the supermarket (coop, conad, desper ...)
<i>brand:wikidata</i>	string	wikidata identifier code
<i>brand:wikipedia</i>	string	brand associated with country
<i>toilets:wheelchair</i>	boolean	wheelchair toilets is present
<i>wheelchair</i>	boolean	wheelchair service is present
<i>website</i>	url	official website of the brand
<i>addr:province</i>	string	province of the location
<i>addr:state</i>	string	state code of the location
<i>building:levels</i>	int	number of levels of the building
<i>parking</i>	string	type of parking (underground)
<i>toilets</i>	boolean	it has toilets

- **trails_METADATA.json:**

Fields	Type	Description
<i>id</i>	string	name that identifies a unique object
<i>cai_scale</i>	string	Identifies the cai scale difficulty classification
<i>distance</i>	int	Identifies the distance of the trials
<i>duration:backward</i>	string	Identifies the duration of the backward
<i>duration:forward</i>	string	Identifies the duration of the backward
<i>foot</i>	boolean	identifies the possibility to complete the path by foot
<i>from</i>	string	Indicates the start position
<i>highway</i>	string	path which has been look for these dataset
<i>mtb</i>	bool	Indicates if the path is practicable of mtb
<i>mtb:scale</i>	int	Indicates the difficult scale of the mtb path
<i>network</i>	string	specifies the hiking route as a local route (=lwn)
<i>note:project_page</i>	string	Indicates the project page url
<i>operator</i>	string	The name of the organization that maintains the path
<i>osmc:symbol</i>	string	describe route symbol that is used as waymarker or on guideposts
<i>sac_scale</i>	int	A difficulty rating scheme for hiking trails.
<i>source</i>	string	where these data came from
<i>to</i>	string	where the path ends
<i>trail.visibility</i>	string	describes attributes regarding trail visibility (not route visibility) and orientation
<i>type</i>	string	type of the path
<i>website</i>	string	link to the website of the organization
<i>@id</i>	string	identification number
<i>ascent</i>	int	meters of ascent
<i>descent</i>	int	meters of descent
<i>name</i>	string	name of the path
<i>surface</i>	string	Surface values provide additional information about the physical surface of roads/footpaths.
<i>bicycle</i>	bool	affordable for bycicle

- **building_METADATA.json:**

Fields	Type	Description
<i>id</i>	string	identification code
<i>amenity</i>	string	For describing useful and important facilities for visitors and residents. Facilities include for example toilets, telephones, banks, pharmacies, prisons and schools.
<i>building</i>	string	The building tag is used to mark a given object as a building
<i>name</i>	string	name of the location
<i>type</i>	string	The key type is commonly used to set a relation's type
<i>wikidata</i>	string	The ID of the Wikidata item about the feature
<i>@id</i>	string	identification code
<i>historic</i>	string	used to identify features that are of historic interest
<i>addr:city</i>	string	city of the location
<i>addr:country</i>	string	country of the location
<i>addr:housenumber</i>	string	civic code
<i>addr:postcode</i>	string	postcode of the location
<i>addr:street</i>	string	street address
<i>contact:email</i>	string	email contact
<i>contact:fax</i>	string	fax contact
<i>contact:phone</i>	string	phone number contact
<i>contact:website</i>	url	website of the location
<i>email</i>	string	email contact
<i>operator</i>	string	used to name a company, corporation, person or any other entity who is directly in charge of the current operation of a map object.
<i>ref:vatin</i>	string	VAT code information

<i>castle_type</i>	string	used to distinguish between the various type of historic castles - stately vs defensive etc
<i>building:levels</i>	string	used for marking the number of above-ground levels of a building
<i>building:use</i>	string	describes what kind of function a building=* serves
<i>roof:levels</i>	string	For stating the number of levels in the roof of a building
<i>access</i>	string	describe restrictions on the use of highways and other transportation routes (railways, waterways), as well as facilities such as buildings, building entrances, amenities and leisure entities.
<i>layer</i>	string	one of several methods used to describe vertical relationships between crossing or overlapping features

- **civici_web_METADATA.json:**

Fields	Type	Description
<i>civico_num</i>	string	civic number (without slash)
<i>civico_let</i>	string	slash, if exist
<i>civico_alf</i>	string	numero civico con barra (se presente)
<i>desvia</i>	string	street description
<i>strada</i>	string	street code
<i>cap</i>	int	postal code
<i>tipo_num</i>	string	type of civic: 'principale' or 'secondaria' (language: italy)
<i>tipo_en</i>	string	type of civic: 'principal' or 'secondary' (language: english)
<i>ingresso</i>	string	entrance type: 'abitazione', 'cancello' or 'altro' (language: italy)
<i>ingr_en</i>	string	entrance type: 'abitation', 'gate' or 'other' (language: english)
<i>fumetto</i>	string	complete address
<i>url</i>	string	not useful
<i>sobborgo</i>	string	Cadastral community

1.2.4 Datasets collection process

In order to download the 257 locations and point of interests datasets from open data Trentino we made a script in python to automate the process: the script "points_of_interests_download.py"¹⁹ looks in the search page of the website with the tag "luoghi e punti di interesse"; starting from page 1 and going to 8, it looks all the datasets containing "luoghi" and "interesse" in the title, then for every dataset found it download in the dataset page the json, (half of the times) the GEO version of the dataset (which it has more records) and the metadata. After the download process it's been used "points_of_interests_reorganize.py"²⁰ to move the metadata files in the right folder.

Because of the missing high school data in open data Trentino, we made scraping in [comuniecittá.it](#): the script "comuniecitta_download.py"²¹ it take all the information in the main page (name, address, if it's private or public) and then for every school it take the study path information. Finally the script saves the data in csv format.

The information has been exported from **OpenStreetMap** thanks to **Overpass turbo**, a web based data mining tool for OpenStreetMap.

1.2.5 Inception level evaluation

Considering the dimension of the group on the work on this project (4 persons), we had maintained a near-constant communication between the knowledge group and the data scientist group. This result that before developing the CQs both the components of the knowledge team have a very good knowledge of the datasets available. This has permitted to model the query based on it, with only a couple of queries out of 43 developed CQs (around 4%) that we had to rewrite to fit the metadata available inside the dataset.

Our main concern is instead in the quality of the data we collected, as an example while the metadata inside the data objects is present for near all the collected data, other possible useful metadata are at end available only for one or two data row.

¹⁹https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/points_of_interests_download.py

²⁰https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/points_of_interests_reorganize.py

²¹https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/comuniecitta_download.py

1.3 Informal Modeling

This section describe the steps of the informal modelling phase of the iTelos methodology. This include a first section that describe the schema of the data (including the ER and variance respect to CQs) and a second section describing the data management and describe in detail the metadata.

1.3.1 Schema level

The schema level in this phase report the first informal definition of the ETYPES and of the EER model constructed using them.

1.3.1.1 ETYPES and EER Model definition

We have divided the ETYPES in three categories, common that contain all the ETYPES shared by the elements inside the core category. The core category contains all the ETYPES key to answer to the competency questions describe above. The last category is the contextual ones that contain all the elements that are not strictly necessary to answer the CQs and are related to element into the core category. These ETYPES are useful to help the CQs to be more complex and / or complete. In the ER schema we identify the common ETYPES in blue, the core ones in green and the contextual in red. You can find the complete ER schema in fig. 1.

The presence of a line with a empty arrow at the end of a line inside the diagram means that the connected entities have a hierarchical dependency and the one with the arrow is the father. We use the empty rhombus form to identify a element where the maximum cardinality is 1, instead a full rhombus is used when the maximum cardinality is n . The connection reference is always stored inside the side without the rhombus. If the filled rhombus is present in both side of the line this mean that the relation have a cardinality n-n.

The central element of the common category is the etype point each point is a real world 3D space coordinate. Then a series of this point contribute to forming a chain. The point chain can be divided in paths (that group all the paths travelled for pleasure or more generally using human locomotion) and in transportation ways (that normally contains all the chain travelled for necessity and with the help of machines). Each point can be the position of an address, while we try to maintain a generic approach to the problems when possible, the address format is specific to each country, considering our context is the Province of Trento, we adopt the Italian format. Each point then can be a point of interest, exist various types of point of interest each one with different attributes, that are necessary to resolve the CQs, for this reason they are inserted inside the core category.

The core category contain all the various derivations of the points of interest. This include as an example the pharmacies, the libraries, the bars, ecc. The fields depend mostly on the type of the point of interest and are better describe in detail in the paragraph 1.3.2.2. A point can also represent the position of a building and we use this information aggregated together to estimate the density of a particular area. The last group of etypes inside the core category are the ones that are identifiable with a line, like for example the bike path. This one are divided into two group the first that inheritance from the path etype and the second that inheritance from the transportation way. The first group include the ski slopes, bike paths and excursion path. Transportation way include the roads and the train lines. Ski resort and excursion paths with the related entities are taken from the tourist facility group. The bus stop, stop time, calendar and calendar dates are taken from the transportation group. The metadata are described inside the ER.

The last category is represented by the contextual etypes. This category include a series of enumerator that describe the various values that the connected field can assume. For a detail description of the values please see the paragraph 1.3.2.2. Considering they are not critical for the resolution of the main problem also the stop times and route tables are inside the contextual category.

In the following there is a list of all the E-types presented in the EER model with a small semantic description and with the relation with other e-types and attribute. This types is defined trying to aligned our informal schema and our ontology, presented in the next section, with the etypes present in the schema.org ontology and the ontologies made by the transportation and tourist facilities group for the overlapping part:

Common Etypes

- **GeoCoordinate:** a *common type* entity describing a single point in the world.

Relations:

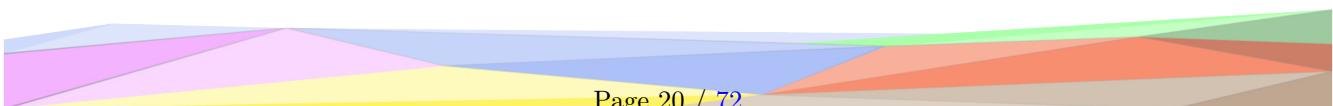
- it may have some Postal Address (has PostalAddress realtion that is the inverse relation of isInPosition)
- it may be part of some GeoShape (isContained relation that is the inverse relation of hasPoints)
- it is the position of one Place (coordinateHasPlace relation that is the inverse relation of hasGeo)

Attributes:

- **latitude:** latitude coordinate in the world
- **longitude:** longitude coordinate in the world
- **altitude:** height of the point from the sea level

- **Place:** a *common type* entity describing a point of interest entities that have a somewhat fixed, physical extension.

Relations:



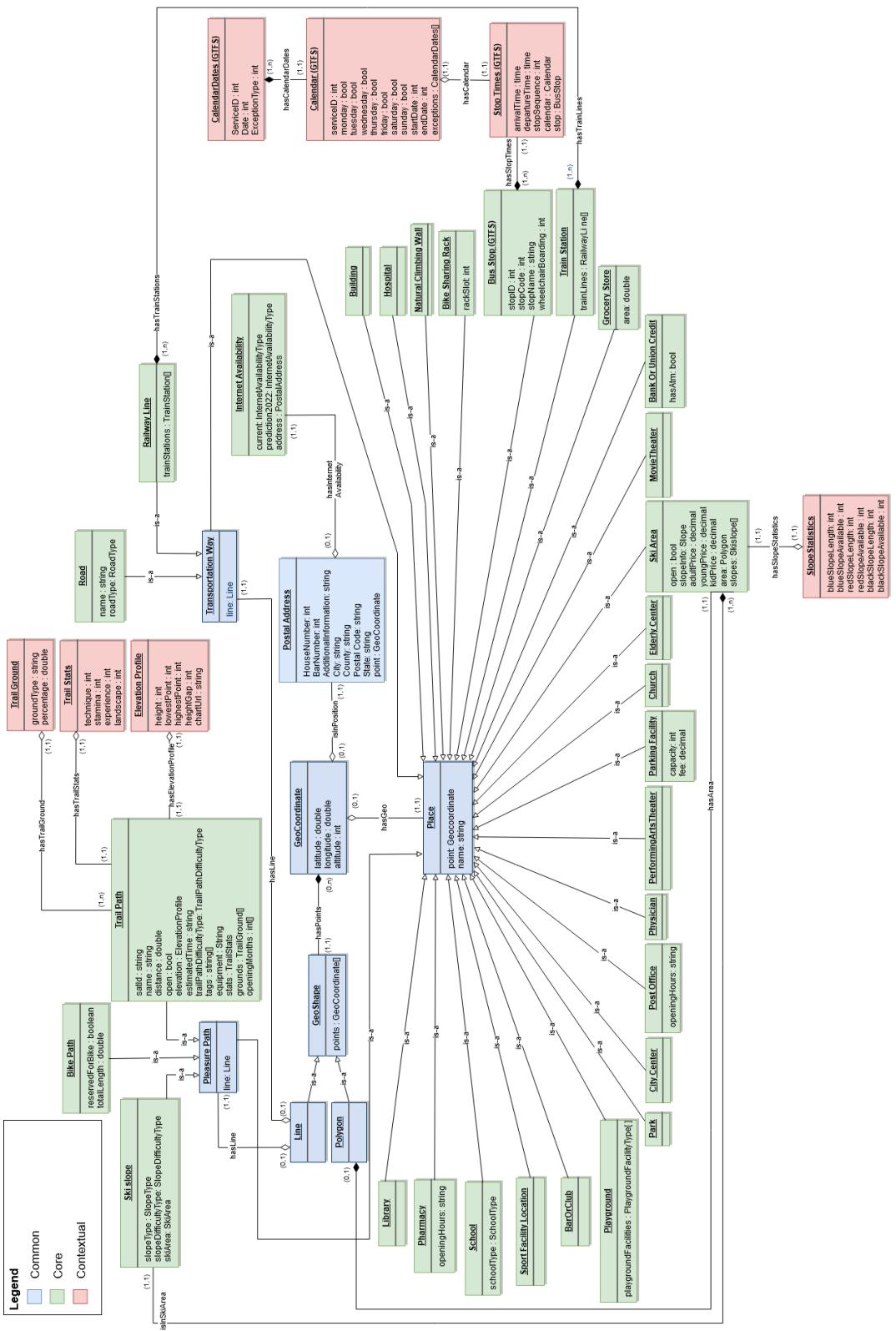


Figure 1: EER Diagram of the data used inside the project

- Is the super-class of the following classes: Library, Pharmacy, School, Sport Facility Location, Bar or Club, Playground, Park, City Center, Post Office, Physician, Performing Arts Theater, Parking Facility, Church, Elderly Center, Ski Area, Movie Theater, Bank or Credit Union, Grocery Store, Train Station, Bus Stop, Bike Sharing Rack, Natural Climbing Wall, Hospital and Building
- it has min 1 GeoCoordinate (hasGeo)

Attributes:

- **name:** it has max 1 string that refers to the name of the place.

- **Postal Address:** a *common type* entity describing the street address of a point.

Relations:

- it is in only one position defined by the GeoCoordinate (isInPosition)
- it has zero or one Internet Availability (hasInternetAvailability)

attributes:

- **HouseNumber:** integer that defines the civic number of the building
- **BarNumber:** integer that defines the number of the door building
- **AdditionalInformation:** string that defines the additional information about the address
- **City:** string that defines the city of the address
- **County:** string that defines the county of the address
- **PostalCode:** string that defines the postal code of the address
- **State:** string that defines the state of the address

- **GeoShape:** a *common type* entity describing a geographical shape such as line or polygon.

Relations:

- it has min 2 Geocordinate
- it is the superclass of the classes Line and Polygon

It hasn't other attributes

- **Line:** a *common type* entity describing a chain of points.

Relations:

- it may be contained in the Pleasure Path or Transportation way path.
- it is a Geo Shape class and for this reason it implies all the Geo Shape relations and attributes

It hasn't other attributes

- **Polygon:** a *common type* entity describing closed figure represented by a chain of points.

Relations:

- it may be contained in the Ski Area.
- it is a Geo Shape class and for this reason it implies all the Geo Shape relations and attributes

It hasn't other attributes

- **Pleasure Path:** a *common type* entity describing a path normally travelled for pleasure and using human locomotion.

Relations:

- it has in the Ski Area.
- it is a Place class and for this reason it implies all the Place relations and attributes

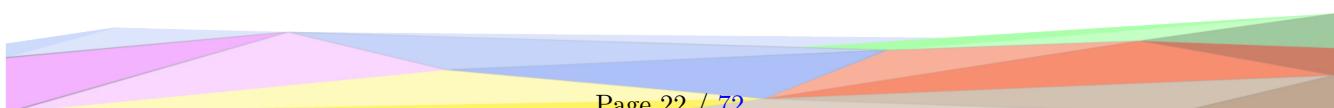
It hasn't other attributes

- **Transportation Way:** a *common type* entity describing a path traveled for duty using motorized transportation mean.

Relations:

- it has in the Ski Area.
- it is a Place class and for this reason it implies all the Place relations and attributes

It hasn't other attributes



Core Etypes

- **Road:** a *core type* entity describing a general road or route.

Relations:

- It is a Transportation Way and for this reason it implies all the Transportation Way relations and attributes.

Attributes:

- **name:** it is a string that describe the name of the road
- **type:** it is a enumeration class that describe the type of road (highway, primary, secondary tertiary)

- **Railway Line:** a *core type* entity describing a pair of rails on which a railway train runs or a group of railway tracks running parallel, allowing one track to be used for each direction (a double-track railway line), or allowing segregation of fast trains from stopping trains (a four-track railway line).

Relations:

- It is a Transportation Way and for this reason it implies all the Transportation Way relations and attributes.
- it has at least one Train Station or more (has TrainStations relation)

It hasn't other attributes

- **Internet Availability:** a *core type* entity describing the availability of internet in a particular address.

Relations: it has minimum one Postal Address (hasInternetAvailability relation)

Attributes:

- **current:** current availability of internet and the type of service describe by an enumeration class called *InternetAvailabilityType* (`black_novhc`, `black_novhc_fo`, `black_novhc_fwa`, `grey_novhc_copper`, `grey_novhc_radio`, `grey_novhc_fo`, `grey_novhc_fwa`, `no_covarage`)
- **prediction2022:** prediction of the availability of internet and the type of service describe by an enumeration class called *InternetAvailabilityType* (`black_novhc`, `black_novhc_fo`, `black_novhc_fwa`, `grey_novhc_copper`, `grey_novhc_radio`, `grey_novhc_fo`, `grey_novhc_fwa`, `no_covarage`)

- **Bike Path:** a *core type* entity indicating the cycle/bike paths

Relations:

- It is a Pleasure Path and for this reason it implies all the Pleasure Path relations and attributes.

Attributes:

- **reservedforBike:** it is a boolean that describes if the path is dedicated only to cycle or not.
- **totalLength:** it is a double that describe the length of the path

- **Trail Path:** a *core type* entity indicating a path for excursion purpose

Relations:

- It is a Pleasure Path and for this reason it implies all the Pleasure Path relations and attributes.
- It has one and only one trail stats class that describe the statistics of the trail path. (hasTrailStats relation)
- It has one and only one elevation profile class that describe the elevation profile of the trail. (hasElevationProfile relation)
- It has one and only one trail ground class that describe the typology of the ground. (hasTrailGround relation)

Attributes:

- **satId:** it has a string that describes trail id in the SAT registry
- **name:** it has a string that describes name of the path
- **distance:** it has a double that describes length of the path in meters
- **open:** it is a boolean that indicates whether the trail is currently open
- **estimatedTime:** it has a string that describes time estimated to run across the trail
- **tags:** it has a list of string that describes the tags associated to the trail
- **equipment:** it has a string that describes equipments needed for the trail
- **openingMonths:** it has a list of integer that describes in month when the trail is usable.

- **Ski slope:** a *core type* entity describing ski.

Relations:

- It is a Pleasure Path and for this reason it implies all the Pleasure Path relations and attributes.
- It is collocated in specific entity of the Ski Area etype (isInSkiArea relation)

Attributes:

- **SlopeType:** it has an enumeration type, called SlopeType, that describes the type ski track (nordic or downhill)
- **SlopeDifficultyType:** It has an enumeration type, called SlopeDifficultyType, that describes the difficulty level of the skislope (easy, intermediate, advanced).

- **Library:** a *core type* entity describing libraries

Relations:

- It is Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Pharmacy:** a *core type* entity describing pharmacies or drug stores

Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **openingHours:** it is a string that describes timetable of the pharmacy

- **School:** a *core type* entity describing schools

Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **SchoolType:** It has an enumeration class, called SchoolType, that describes type of the school (nursery, elementary, middle, high)

- **Sport Facility Location:** a *core type* entity describing sports location, such as playing fields.

Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Bar or Club:** a *core type* entity describing bar, pub and cafes

Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Playground:** a *core type* entity describing playgrounds.

Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **playgroundfacilities:** it has an enumeration class, called PlaygroundFacilityType, that gives info about the facility that exist in the leisure

- **Park:** a *core type* entity describing parks.

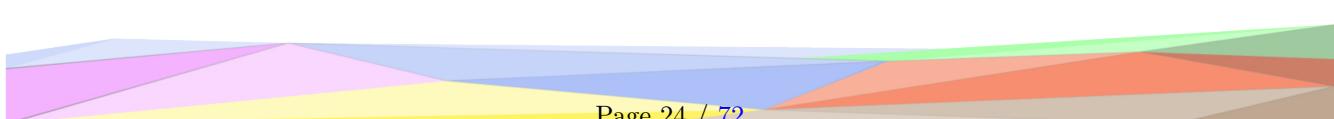
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **City Center:** a *core type* entity describing the city centers of towns.

Relations:



-
- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Post Office:** a *core type* entity describing post offices
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **openingHours:** it has a string that describes opening hours of the post office

- **Physician:** a *core type* entity describing physician and ambulatory
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **PerformingArtsTheater:** a *core type* entity describing performing arts theaters. *Relations:*

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Church:** a *core type* entity describing churches
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Elderly Center:** a *core type* entity describing elderly centers or a healthcare residence for Elderly
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Movie Theater:** a *core type* entity describing cinemas and movie theaters
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Hospital:** a *core type* entity describing hospitals.
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Building:** a *core type* entity contains the position of a generic building.
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

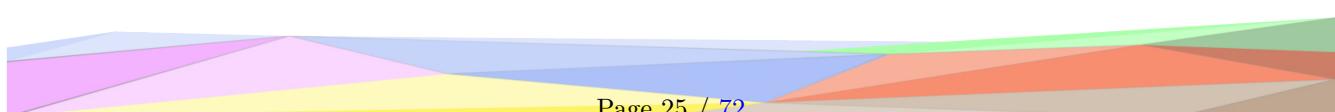
It hasn't other attributes

- **Natural Climbing Wall:** a *core type* entity describing a climbing wall of natural origin
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

It hasn't other attributes

- **Bike Sharing Rack:** a *core type* entity describing the racks of a bike sharing service
Relations:



- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **rackSlot:** it has an integer that describes the number of rack slot.

- **Grocery Store:** a *core type* entity describing a grocery store or supermarket
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **area:** it has a double value that describes the size of the area that the supermarket covers.

- **Bank or Credit Union:** a *core type* entity describing bank or a credit union
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **hasAtm:** it has a boolean value that gives the information about the presence or absence of an ATM.

- **Parking Facilities:** a *core type* entity describing a parking lot or other parking facility
Relations:

- It is a Place and for this reason it implies all the Place relations and attributes.

Attributes:

- **capacity:** it has one integer that describes the maximum capacity of the parking lot.

- **fee:** it has a decimal value that gives the information about fee (is it charged for a service, or for access)

- **Bus Stop:** a *core type* entity describing the bus stops *Relations:*

- is a **Point of Interest**
- it has at least one **Stop Times** or more (hasStopTimes)

Attributes:

- **stopID:** an integer indicating uniquely the bus stop

- **stopCode:**

- **stopName:** name of the bus stop

- **wheelchairBoarding:** number of wheelchair boarding

- **Ski Area:** a *core type* entity describing ski resorts
relations:

- it is a **point of interest**
- it has one **slope statistics** (hasSlopeStatistics)
- it has one or more **skiSlope**
- its area is described by one **Polygon**

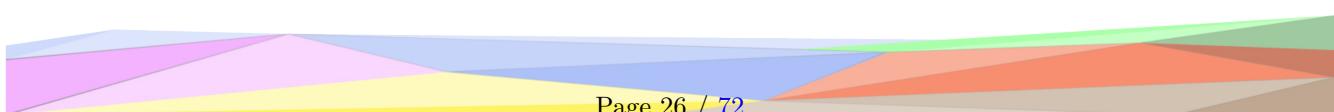
attributes:

- **name:** name of the ski resort
- **open:** current state of the resort (open or note)
- **slopeInfo:** general info of the resort divided for path difficulty
- **adultPrice:** price for adult
- **kidPrice:** price for kid

- **Train station:** a *core type* entity describing train stations *Relations:*

- it has at least one or more **TrainLines** (hasTrainLines)

it hasn't other attributes



Contextual Etypes

- **Trail Ground:** a *contextual type* entity describing the ground of the excursion path

Relations:

- it has exactly one **Excursion Path**

Attributes:

- **groundType:** type of the terrain of the trail
- **percentage:** percentage of the type of terrain the trail is composed

- **Trail Stats:** a *contextual type* entity describing scores of the difficulty of the excursion path

Relations:

- it has exactly one **Excursion Path**

Attributes:

- **technique:** technique score
- **stamina:** stamina score
- **experience:** experience score
- **landscape:** landscape score

- **Elevation Profile:** a *contextual type* entity describing the elevation of the excursion path

Relations:

- it has exactly one **Excursion Path**

Attributes:

- **lowestPoint:** altitude of the lowest point of the trail
- **highestPoint:** altitude of the highest point of the trail
- **height:** height of the trail
- **heightGap:** gap between the higher and lower points of the trail
- **chartURL:** url to the trail-elevation graph

- **Stop Times:** a *contextual type* entity describing the stop times of bus in Trentino

Relations:

- it has exactly one **Calendar** (hasCalendar)
- it has at exactly one **Stop** (hasStopTimes)

Attributes:

- **arrivalTime:** arrival time of the bus in that stop
- **departureTime:** departure time of the bus in that stop
- **stopSequence:**

- **Calendar:** a *contextual type* entity describing the calendar (days of the week) for the bus in Trentino

Relations:

- it has one **CalendarDates** (hasCalendarDates), which describes exceptions
- it is contained in **Stop Times**

Attributes:

- **serviceID:** ID of the service
- **monday:** boolean which advise if something exist monday
- **tuesday:** boolean which advise if something exist tuesday
- **wednesday:** boolean which advise if something exist wednesday
- **thursday:** boolean which advise if something exist thursday
- **friday:** boolean which advise if something exist friday

- **saturday**: boolean which advise if something exist saturday
- **sunday**: boolean which advise if something exist sunday
- **startDate**: starting date of the time line
- **endDate**: end date of the time line
- **Calendar Dates**: a *contextual type* entity describing ...
 - Relations*:
 - it is contained in **Calendar**
- **Slope Statistics**: a *core type* entity describing generically ski slopes in a resort
 - Relations*:
 - it is contained in **SkiArea**

Attributes:

- **ServiceID**:
- **Date**:
- **ExceptionType**:

- **Slope Statistics**: a *core type* entity describing generically ski slopes in a resort
 - Relations*:
 - it is contained in **SkiArea**

Attributes:

- **blueSlopeLength**: number of kilometers of blue slopes in the resort
- **blueSlopeAvailable**:
- **redSlopeLength**: number of kilometers of red slopes in the resort intermediate, advanced)
- **redSlopeAvailable**:
- **blackSlopeLength**: number of kilometers of black slopes in the resort
- **blackSlopeAvailable**:

1.3.1.2 Variance respect CQs definition

This section aims to define the variance between the schema elements produced in this phase, and the definition of the CQs reported in the previous phase. This is a way to define the quality of the outcomes for the current phase as well as the alignment of the overall project development process.

During the development of this phase we have revised and integrated some CQs from the last phase. This process was done to add complexity to the proposed CQs to make the project more complete and interesting for our purpose. Specifically, this enrichment of the questions has happened thanks to two different specific actions:

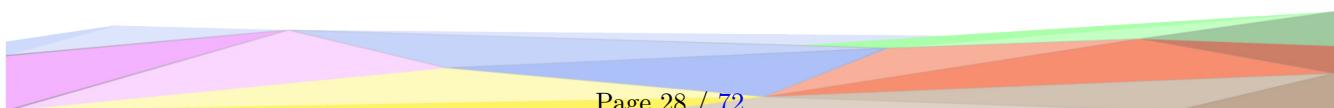
- More in-depth exploration of the previously recovered datasets allowing us to add to some interested etypes in the CQs made more complex others attributed initially discarded.
- Integration of our project with some parts of two projects developed by our colleagues. In particular, we have created a collaboration and exchange of data in a standardized and fully integrated way with the group "Transportation" and the group "Tourist facilities".

Specifically, thanks to the actions listed above, the CQs relating to the ski areas, ski slopes, hiking trails to playgrounds, Bank Branch and bus stops have been deepened. Thanks to the deeper exploration of the datasets now in fact the system is designed to answer much more detailed questions about these categories, for example:

- Which games are present in the nearest playground?
- Where is the bank with an available ATM?
- What are the Nordic skiing trails within 35 k?
- Which are the 3 ski areas with the greatest number of kilometers of red slopes?
- What are the excursion paths in a radius of 3 km from the unit with a positive height difference of less than 200m?

1.3.2 Data level

The data level section in this phase reports the evolution of the datasets collected previously, reporting the metadata information for each new data, or new version of data, obtained.



1.3.2.1 Datasets management process

All the datasets have been filtered using a script in python: `data_filtering.py`²² read a configuration file (`data_filtering_config.json`²³) to understand for every dataset what fields to keep in the new datasets. We decides which attributes were useful by looking at the output provided by `count_json.m`²⁴: it shows for each attributes how many objects contain them. Therefore, only attributes which were not sparse in the datasets has been selected. Then, the script read every dataset in the configuration file, it compares the files and it select the ones appearing in the configuration file and finally it writes the selected data in a new file in the Informal Modeling folder. The "luoghi e punti di interesse per comune" and "building" datasets, because of the number of files, are directly filtered in the script without using the configuration file. Since there are two types of datasets in "luoghi e punti di interesse per comune" (the GEO version and the "rich" version), the files are been merged with only the filtered attributes. All the files are converted in json and each record is listed in the same way in every file.

Using the same configuration file, the script `metadata_filtering.py`²⁵ applies the changes of the fields of the data in the metadata. All the metadata of "luoghi e punti di interesse per comune" are been merged in a single file `luoghi_e_punti_di_interesse_per_comune_METADATA.json`²⁶.

1.3.2.2 Datasets metadata documentation

These are the new metadata of the filtered data with some enrichment (like source, format or timestamp):

`areaski_Metadata.json`:

Dataset Properties	Description	Type	Data Definition
<code>type</code>	FeatureCollection		
<code>generator</code>	overpass-ide		
<code>copyright</code>	The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.		
<code>timestamp</code>	2020-10-19T19:05:03Z		
Attributes			
<code>@id</code>	name that identifies a unique object	string	
<code>name</code>	name of data	string	
<code>type</code>	type of geometry	string	
<code>coordinates</code>	array the points which compose the polygon	array	

`luoghi_e_punti_di_interesse_per_comune_Metadata.json`:

Since this includes 147 datasets and metadatas, here are reported the common properties and fields. The different values are set with *. The full metadata are in the repository ²⁷

Dataset Properties	Description	Type	Data Definition
<code>source</code>	https://dati.trentino.it/dataset/ *		
<code>format</code>	json		
<code>Identificativo del dataset</code>	*		
<code>Data di rilascio</code>	*		
<code>Data di modifica</code>	*		
<code>Lingue del dataset</code>	italiano		
<code>Titolare</code>	*		
<code>Frequenza di aggiornamento</code>	in continuo aggiornamento		
<code>Autore</code>	*		
Attributes			
<code>@id</code>	name that identifies a unique object	string	Common
<code>name</code>	name of data	string	Core
<code>type</code>	type of geometry	string	Common
<code>coordinates</code>	array the points which compose the polygon	array	Common

`bank_Metadata.json`:

²²https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/data_filtering.py

²³https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/data_filtering_config.json

²⁴https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/count_json.m

²⁵https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/metadata_filtering.py

²⁶https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/dataset/Informal%20Modeling/metadata/luoghi_e_punti_di_interesse_per_comune_METADATA.json

²⁷https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/dataset/Informal%20Modeling/metadata/luoghi_e_punti_di_interesse_per_comune_METADATA.json

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T13:50:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	common
<i>name</i>	name of the facility	string	core
<i>atm</i>	indicate the presence of an atm	boolean	core
<i>addr:housenumber</i>	the house number component of the address	string	common
<i>addr:street</i>	the street component of the address	string	common
<i>addr:city</i>	the city component of the address	string	common
<i>addr:postcode</i>	the postcode component of the address	string	common
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

busstop_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>copyright</i>	The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.		
<i>timestamp</i>	2020-10-19T10:54:03Z		
Attributes			
<i>@id</i>	name that identifies a unique object	string	Common
<i>name</i>	name that identifies the name of the bus-stop	string	Core
<i>addr:city</i>	name identifying the name of the city where the bus stop is located	string	Common
<i>addr:postcode</i>	number identifying the postcode of the city where the bus stop is located	int	Common
<i>addr:street</i>	name identifying the name of the street where the bus stop is located	string	Common
<i>type</i>	type of geometry	string	Common
<i>coordinates</i>	array the points which compose the polygon	array	Common

cinema_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>copyright</i>	The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.		
<i>timestamp</i>	2020-10-19T15:03:03Z		
Attributes			
<i>name</i>	Identifies the name of the cinema	string	Core
<i>addr:city</i>	Identifies in term of cities the position of the cinema	string	Common
<i>addr:housenumber</i>	Identifies the house number of the cinema	int	Common
<i>addr:postcode</i>	Identifies the postcode of the cities where the cinema is located	int	Common
<i>addr:street</i>	Identifies the street name where the cinema is located	string	Common
<i>type</i>	type of geometry	string	Common
<i>coordinates</i>	array the points which compose the polygon	array	Common

city_center_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T10:54:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	the name of the area	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

climb_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T15:21:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	Used to univocally identifies the object	string	core
<i>name</i>	Identifies the name of the climb	string	core
<i>addr:housenumber</i>	Indicates the housenumber in the address information	string	common
<i>addr:postcode</i>	Indicates the postcode in the address information	int	common
<i>addr:street</i>	Indicates the postcode in the address information	string	common
<i>opening_hours</i>	Indicates the opening hours of the climb spot	string	core
<i>height</i>	Indicates the height of the climbing	int	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

park_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T19:30:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	name that identifies a unique object	string	core
<i>leisure</i>	type of facility inside	string	core
<i>name</i>	the name of the facility	string	core
<i>playground</i>	describe the type of playground available	string	contextual
<i>playground:basketswing</i>	describe if the facility have a basketswing	boolean	contextual
<i>playground:chain_ladder</i>	describe if the facility have a chain ladder	boolean	contextual
<i>playground:slide</i>	describe if the facility have a slide	boolean	contextual
<i>playground:swing</i>	describe if the facility have a swing	boolean	contextual
<i>opening_hours</i>	the opening hours of the park	string	contextual
<i>addr:city</i>	the city component of the address	string	common
<i>addr:housenumber</i>	the house number component of the address	string	common

<i>addr:postcode</i>	the postal code component of the address	string	common
<i>addr:street</i>	the street component of the address	string	common
<i>playground:aerialrotator</i>	describe if the facility have an aerialrotator	boolean	contextual
<i>playground:basketball_backboard</i>	describe if a zone where play basketball is available	boolean	contextual
<i>playground:exercise</i>	describe if an equipment are for the exercise is present	boolean	contextual
<i>playground:horizontal_bar</i>	describe if an horizontal bar is present	boolean	contextual
<i>playground:seesaw</i>	describe if an seesaw is present	boolean	contextual
<i>playground:tunnel_tube</i>	describe if an tunnel tube is present	boolean	contextual
<i>playground:climbingframe</i>	describe if a climbing frame is present	boolean	contextual
<i>playground:multi_play</i>	describe if a multiple usage surface is available	boolean	contextual
<i>playground:sandpit</i>	describe if a sandpit is present	boolean	contextual
<i>playground:theme</i>	describe the theme of the park	string	contextual
<i>playground:skate_equipment</i>	describe if the park is equipment to play with the skateboard	boolean	contextual
<i>playground:teenshelter</i>	describe if a teen shelter is available	boolean	contextual
<i>playground:water</i>	describe if a water playground is available	boolean	contextual
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

parking_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T09:57:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	core
<i>fee</i>	The fee tag is for specifying whether a fee is usually charged for a service, or for access.	boolean	core
<i>capacity</i>	max capacity of the parking lot	int	core
<i>name</i>	name of the location	string	core
<i>addr:city</i>	name of the city	string	common
<i>addr:housenumber</i>	civic code	int	common
<i>addr:postcode</i>	postal code of the location	int	common
<i>addr:street</i>	street address of the location	string	common
<i>opening_hours</i>	day and hours of opening	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

pharmacy_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T15:28:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	core
<i>name</i>	name of the location	string	core
<i>addr:city</i>	city of the location	string	common
<i>addr:housenumber</i>	civic code	int	common

<i>addr:postcode</i>	postcode of the location	int	common
<i>addr:street</i>	street address of the location	string	common
<i>opening_hours</i>	day and hours of opening	string	common
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

post_office_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T15:26:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	name that identifies a unique object	string	core
<i>addr:city</i>	city component of the address	string	common
<i>addr:housenumber</i>	the house number component of the address	string	common
<i>addr:postcode</i>	the postal code component of the address	string	common
<i>addr:street</i>	the street component of the address	string	common
<i>name</i>	the name of the facility	string	core
<i>opening_hours</i>	the opening hours of the facility	string	core
<i>addr:province</i>	the province component of the address	string	common
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

railway_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T18:07:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	core
<i>name</i>	name of te location	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

roads_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-20T07:44:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	core
<i>highway</i>	type of road (trunk,primary, secondary, tertiary)	string	contextual

<i>name</i>	name of the road	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

skislopes_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T19:07:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	core
<i>name</i>	name of the ski slopes	string	core
<i>piste:type</i>	type of the skiing available (nordic, downhill)	string	core
<i>piste:difficulty</i>	difficulty level of the skislope (easy, intermediate, advanced)	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

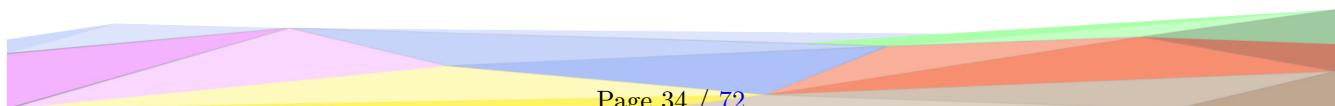
supermarket_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T13:28:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	core
<i>name</i>	name of the supermarket	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

trails_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T20:02:05Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification number	string	core
<i>name</i>	name of the path	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

building_METADATA.json:



Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T13:50:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>@id</i>	identification code	string	core
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

civici_web_METADATA.json:

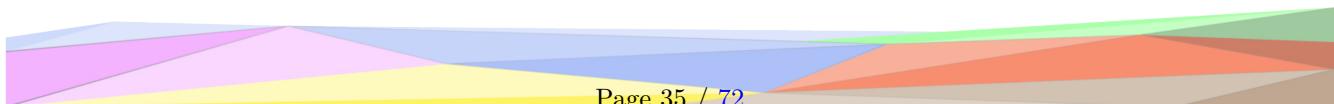
Dataset Properties	Description	Type	Data Definition
<i>Dataset ID</i>	4f474fc8-181d-4b15-9ee4-60b3f54f4068		
<i>Source</i>	https://dati.trentino.it/dataset/comune-di-trento-numeri-civici		
<i>Release date</i>	22-02-2013		
<i>Modification date</i>	08-05-2019		
<i>Geographical Coverage</i>	comune di Trento		
<i>Temporal extension</i>	'From': '22-02-2013'		
<i>Holder</i>	'Name': 'Comune di Trento', 'Code IPA/IVA': 'c_l378'		
<i>Update frequency</i>	daily		
<i>Format</i>	json		
Attributes			
<i>civico_num</i>	civic number (without slash)	string	common
<i>desvia</i>	street description	string	common
<i>fumetto</i>	complete address	string	common
<i>sobborgo</i>	Cadastral community	string	common
<i>type</i>	type of geometry	string	common
<i>coordinates</i>	array the points which compose the polygon	array	common

piste_ciclabili_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Dataset ID</i>	c_l378-1129110		
<i>Holder</i>	'Nome': 'Comune di Trento', 'Code IPA/IVA': 'c_l378'		
<i>Release date</i>	09-11-2017		
<i>Modification date</i>	08-05-2019		
<i>Geographical Coverage</i>	Comune di Trento		
<i>Source</i>	https://dati.trentino.it/dataset/piste-ciclabili-open-data		
<i>GeoNames URI</i>	http://www.geonames.org/6541469		
<i>Dataset language</i>	italiano		
<i>Update frequency</i>	continuous updating		
<i>Format</i>	json		
Attributes			
<i>WKT</i>	coordinates of the paths	list int	common
<i>tipo</i>	name of the typology of path (not useful)	string	
<i>fumetto</i>	name of the path	string	common
<i>descrizione</i>	name of the path (equivalent to 'fumetto')	string	common
<i>tipologia</i>	path dedicated only to cycle or also pedestrian	string	core
<i>tratto isolato ciclabile</i>	meters without path dedicated to cycle	int	core
<i>tratto isolato ciclopedonale</i>	meters without path dedicated to cycle or pedestrian	int	core

elementari_METADATA.json:

Dataset Properties	Description	Type	Data Definition
--------------------	-------------	------	-----------------



<i>Dataset ID</i>	9368a92a-dbe9-4a43-9fc0-66a5f4695e16		
<i>Release date</i>	28-04-2020		
<i>Modification date</i>	31-03-2016		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>Source</i>	https://dati.trentino.it/		
<i>GeoNames URI</i>	http://www.geonames.org/3165243		
<i>Holder</i>	'Nome': 'Provincia autonoma di Trento', 'Code IPA/IVA': 'p-TN'		
<i>Update frequency</i>	unknown		
<i>Format</i>	json		
Attributes			
<i>WKT</i>	Coordinates in Well-Known Text format	string	Common
<i>civico_alf</i>	civic number	string	Common
<i>destra</i>	street address	string	Common
<i>sobborgo</i>	suburb of the school	string	Common
<i>scuola</i>	name of the school	string	Core

materne_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Dataset ID</i>	3dd956d1-a376-4e9a-8182-d839de4163dd		
<i>Release date</i>	28-04-2020		
<i>Modification date</i>	31-03-2016		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>Source</i>	https://dati.trentino.it/		
<i>GeoNames URI</i>	http://www.geonames.org/3165243		
<i>Holder</i>	'Nome': 'Provincia autonoma di Trento', 'Code IPA/IVA': 'p-TN'		
<i>Update frequency</i>	unknown		
<i>Format</i>	json		
Attributes			
<i>WKT</i>	Coordinates in Well-Known Text format	string	Common
<i>civico_alf</i>	civic number	string	Common
<i>destra</i>	street address	string	Common
<i>sobborgo</i>	suburb of the school	string	Common
<i>scuola</i>	name of the school	string	Core

medie_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Dataset ID</i>	ade37eb8-13af-4584-a965-6765e26f1870		
<i>Release date</i>	28-04-2020		
<i>Modification date</i>	31-03-2016		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>URI di GeoNames</i>	http://www.geonames.org/3165243		
<i>Holder</i>	'Nome': 'Provincia autonoma di Trento', 'Code IPA/IVA': 'p-TN'		
<i>Update frequency</i>	unknown		
<i>Format</i>	json		
Attributes			
<i>WKT</i>	Coordinates in Well-Known Text format	string	Common
<i>civico_alf</i>	civic number	string	Common
<i>destra</i>	street address	string	Common
<i>sobborgo</i>	suburb of the school	string	Common
<i>scuola</i>	name of the school	string	Core

superiori_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Release date</i>	19-09-2020		

<i>source</i>	https://www.comuniecitta.it/scuole-secondarie-di-secondo-grado/comune-di-trento-22205		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>Update frequency</i>	unknown		
<i>Format</i>	json		
Attributes			
<i>name</i>	name of the school	string	Core
<i>address</i>	address of the school	string	Common
<i>school type</i>	type of school (public or private)	string	Core
<i>study paths</i>	the study paths offerted from the school (comma separated)	string	Core

internet_quality_trento_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>source</i>	www.infratelitalia.it		
<i>timestamp</i>	2020-09-09		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>comune</i>	town	string	Common
<i>via</i>	street address	string	Common
<i>civico</i>	civic code	int	Common
<i>barrato</i>	sub-civic code	char	Common
<i>class_19</i>	2019 internet coverage	string	Core
<i>class_22</i>	2022 internet coverage estimation	string	Core

bikesharing_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>source</i>	https://dati.trentino.it/dataset/stazioni-bike-sharing-emotion-trentino		
<i>format</i>	json		
<i>timestamp</i>	2020-10-18		
<i>Dataset ID</i>	p_TN:9b9c14d6-ee20-4802-a274-4c17ac96cdd5		
<i>Temi del dataset</i>	'Trasporti': ['4816 trasporti terrestri', '4806 politica dei trasporti'], 'Energia': ['6626 energia dolce', '6606 politica energetica']		
<i>Dataset publisher</i>	'Nome': 'Servizio Trasporti pubblici', 'Codice IPA/IVA': '0OK0PZ'		
<i>Release date</i>	18-11-2014		
<i>Modification date</i>	03-07-2017		
<i>Geographical Coverage</i>	Comune di Trento		
<i>GeoNames URI</i>	http://www.geonames.org/3165241		
<i>Dataset language</i>	italiano		
<i>Holder</i>	'Nome': 'Provincia Autonoma di Trento', 'Codice IPA/IVA': 'p-TN'		
<i>Upload frequency</i>	continuous		
Attributes			
<i>name</i>	name of the bike station	string	Core
<i>address</i>	geographical address of the bike station	string	Common
<i>id</i>	identifier	string	Core
<i>totalSlots</i>	total number of slot bikes	int	Core
<i>position</i>	coordinates of the station	int[2]	Common

stops_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Dataset Identifier</i>	p_TN: d3c9f167-3271-4a43-b5c1-e0879aa5ad3f		

<i>Dataset Publisher</i>	'Name': 'Public Transport Service', 'IPA/VAT Code': '0OK0PZ'		
<i>Date of modification</i>	2017-10-24		
<i>Geographic coverage</i>	Trento		
<i>URI of GeoNames</i>	https://www.geonames.org/3165241		
<i>Languages of the dataset</i>	Italian		
<i>Holder</i>	Autonomous Province of Trento		
<i>Refresh Rate</i>	Half yearly		
<i>Author</i>	'Name': 'Public Transport Service', 'IPA/VAT': '0OK0PZ'		
<i>Url</i>	https://www.trentinotrasporti.it/opendata/google_transit_urbano_tte.zip		
<i>License</i>	Creative Commons Attribution 4.0 International (CC BY 4.0)		
<i>License_Type</i>	https://w3id.org/italia/controlled-vocabulary/licences/A21_CCBY40		
<i>Format</i>	txt		
Attributes			
<i>stop_id</i>	identification code of the bus stop	int	Core
<i>stop_code</i>		string	Core
<i>stop_name</i>	name of the bus stop	string	Core
<i>stop_desc</i>		string	Contextual
<i>stop_lat</i>	latitude coordinate of the stop	double	Common
<i>stop_lon</i>	longitude coordinate of the stop	double	Common
<i>zone_id</i>	identification code of the zone	int	Contextual
<i>wheelchair_boarding</i>	number of wheelchair boarding	int	Core

stop_times_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Dataset Identifier</i>	p_TN: d3c9f167-3271-4a43-b5c1-e0879aa5ad3f		
<i>Dataset Publisher</i>	'Name': 'Public Transport Service', 'IPA/VAT Code': '0OK0PZ'		
<i>Date of modification</i>	2017-10-24		
<i>Geographic coverage</i>	Trento		
<i>URI of GeoNames</i>	https://www.geonames.org/3165241		
<i>Languages of the dataset</i>	Italian		
<i>Holder</i>	Autonomous Province of Trento		
<i>Refresh Rate</i>	Half yearly		
<i>Author</i>	'Name': 'Public Transport Service', 'IPA/VAT': '0OK0PZ'		
<i>Url</i>	https://www.trentinotrasporti.it/opendata/google_transit_urbano_tte.zip		
<i>License</i>	Creative Commons Attribution 4.0 International (CC BY 4.0)		
<i>License_Type</i>	https://w3id.org/italia/controlled-vocabulary/licences/A21_CCBY40		
<i>Format</i>	txt		
Attributes			
<i>trip_id</i>	identification code of the trip line	int	Contextual
<i>arrival_time</i>	arrival time of the bus in that stop	string	Contextual
<i>departure_time</i>	departure time of the bus in that stop	string	Contextual
<i>stop_id</i>	identification code of the bus stop	int	Core
<i>stop_sequence</i>		int	Core

trails_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>lastModified</i>	2020-07-02T08:14:00.000Z		
<i>source</i>	www.outdooractive.com		
<i>update_frequency</i>	no update		
<i>geographical Coverage</i>	Provincia di Trento		

<i>format</i>	json		
Attributes			
<i>sat_Id</i>	trail id in the SAT registry	string	core
<i>trainType</i>	specifies whether the trail is a hiking, biking or snowshoe trail	string	core
<i>name</i>	name of the trail	string	core
<i>distance</i>	length of the trail	double	core
<i>open</i>	whether the trail is currently open	boolean	core
<i>elevation</i>	contains other 5 attributes (<i>lowestPoint</i> , <i>highestPoint</i> , <i>ascent</i> , <i>descent</i> , <i>chartUrl</i>)	elevationProfile	contextual
<i>estimatedTime</i>	contains other 4 attributes (<i>days</i> , <i>hours</i> , <i>minutes</i> , <i>seconds</i>)	duration	core
<i>startRoute</i>	contains other 2 attributes (<i>altitude</i> , <i>description</i>)	locationPoint	core
<i>endRoute</i>	contains other 2 attributes (<i>altitude</i> , <i>description</i>)	locationPoint	core
<i>route</i>	contains other 2 attributes (<i>description</i> , <i>geoPoints</i>)	route	core
<i>difficulty</i>	description of the trail difficulty	string	contextual
<i>tags</i>	array of tags	array	core
<i>equipment</i>	equipments needed for the trail	String	core
<i>stats</i>	contains other 4 attributes (<i>technique</i> , <i>stamina</i> , <i>experience</i> , <i>landscape</i>)	TrailStats	contextual
<i>grounds</i>	contains other 2 attributes (<i>groundType</i> , <i>percentage</i>)	array	contextual
<i>monthTips</i>	contains other 2 attributes (<i>month</i> , <i>open</i>)	array	core
<i>metadata</i>	contains other 4 attributes (<i>created</i> , <i>lastModified</i> , <i>source</i> , <i>resource</i>)	MetaData	core

skiResorts_currentState_METADATA:

Dataset Properties	Description	Type	Data Definition
<i>lastModified</i>	2020-10-24		
<i>source</i>	https://www.skiresort.info/ski-resorts/trentino/		
<i>update_frequency</i>	unknown		
<i>geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	contains the name of the ski resort	string	core
<i>open/closed</i>	has the information if the resort is open or noty	string	contextual

skiResorts_static_METADATA:

Dataset Properties	Description	Type	Data Definition
<i>lastModified</i>	2020-10-24		
<i>source</i>	https://www.skiresort.info/ski-resorts/trentino/		
<i>update_frequency</i>	unknown		
<i>geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	contains the name of the ski resort	string	core
<i>Total lenght</i>	total number of kilometers available in the resort	double	contextual
<i>km blue slope</i>	number of kilometers of blue slopes in the resort	double	contextual
<i>km red slope</i>	number of kilometers of red slopes in the resort	double	contextual
<i>km black slope</i>	number of kilometers of black slopes in the resort	double	contextual
<i>price</i>	Price per person	string	contextual

1.3.2.3 Variance respect Inception datasets

This section aims to define the variance between the data elements (datasets and attributes within them) produced in this phase, and the initial datasets collected in the previous phase. This is a way to define the quality of the outcomes for the current phase as well as the alignment of the overall project development process.

Datasets have been filter in order to reduce sparse attributes: in fact, datasets downloaded from Openstreetmap are often messy and many attributes exist only for few objects. Thus, the filtering has been applied mainly to Openstreetmap. Also, other information has been taken off in order to remove useless attributes.

A great difference can be also found in park dataset, since it initially included dog parks. It has been chosen to take it off.

Finally, area-ski and bus stop datasets have been integrated with dataset of other groups of work ("Transportation Tomorrow" and "Turist facilities"). Datasets involved in this collaboration and exchange of data are transportation, ski area and hike path. Particularly, the data provided by transportation group has been integrated with our dataset, which was previously filtered. Actually, datasets that provide information about bus stops are tree: `stops.json`, `stop.time.json` and `busstop.json`. On the other hand, ski area and trails datasets have been replaced by the ones provided by turist facilities group.

The following table shows the difference between dataset in the Scope Definition Inception phase and Informal modeling one.

Dataset	Initial nr. of attributes	Final nr. of attributes
areaski	17	5
bank	71	9
bikesharing	7	5
building	162	3
busstop	43	7
stop time	/	5
stops	/	8
cinema	23	7
city center	28	3
civici web	14	5
climb	73	9
internet quality	15	15
luoghi e punti d'interesse	21	5
school: nursery	12	5
school: elementary	12	5
school: middle	12	5
school: high	4	4
park	82	28
parking	116	11
pharmacy	51	9
piste ciclabili	8	8
post office	49	10
railway	50	4
roads	153	5
skislopes	69	6
supermarket	66	4
trails	193	29

1.3.3 Informal Modeling Evaluation

The last section of the Informal Modeling phase report the evaluation of the outcomes obtained in this phase, through specif evaluation metrics.

1.4 Formal Modeling

This section describe the Formal Modeling phase. The first part of the section describe the schema while the second part describe the data segment of the phase.

1.4.1 Schema level

Inside this subsection a first paragraph describe how we defined the ontology, the second one list the classes created. The last paragraph describe the change made to the schema compared to the informal phase.

1.4.1.1 Ontology definition

The first step of the process for the development of the ontology schema consist on searching for other reference ontologies. We found two valid ontologies the schema.org ones ²⁸ and the ones develop for the city of Florence by the DISIT group of the University of Florence, km4city ²⁹. While km4city is complete and already formalized a large number of the class we use it also use some things very specific to the city of Florence. For this reason we decide to inspire our work to the schema.org ontology, that instead is more general and try to take in consideration more general needs.

Then we compare the names on our ER with the ones already present on KOS. Based on this we developed a first list of concepts to add to KOS. Then we download from KOS using the API, the already present ontology as a RDF/XML file. Then we start to add the classes necessary with the corresponding concept and adding the concept to KOS if necessary. We use Protégé to modify the RDF file. Then we add the data

²⁸<https://schema.org/>

²⁹<http://wlode.disit.org/WLODE/extract?url=http://www.disit.org/km4city/schema>

properties and the object properties using the same procedure. To add the enumerations we adopt a procedure differently from the ones published inside the Manchester tutorial³⁰, because it is not currently supported by the KOS system where we need to import the ontology. You can see the complete schema under the form of a graph in the Fig. 3, the EER updated with the changes made in the formal phase can be seen in the Fig. 2.

³⁰<http://mowl-power.cs.man.ac.uk/protegeowltutorial/resources/ProtegeOWLTutorialP4v13.pdf>

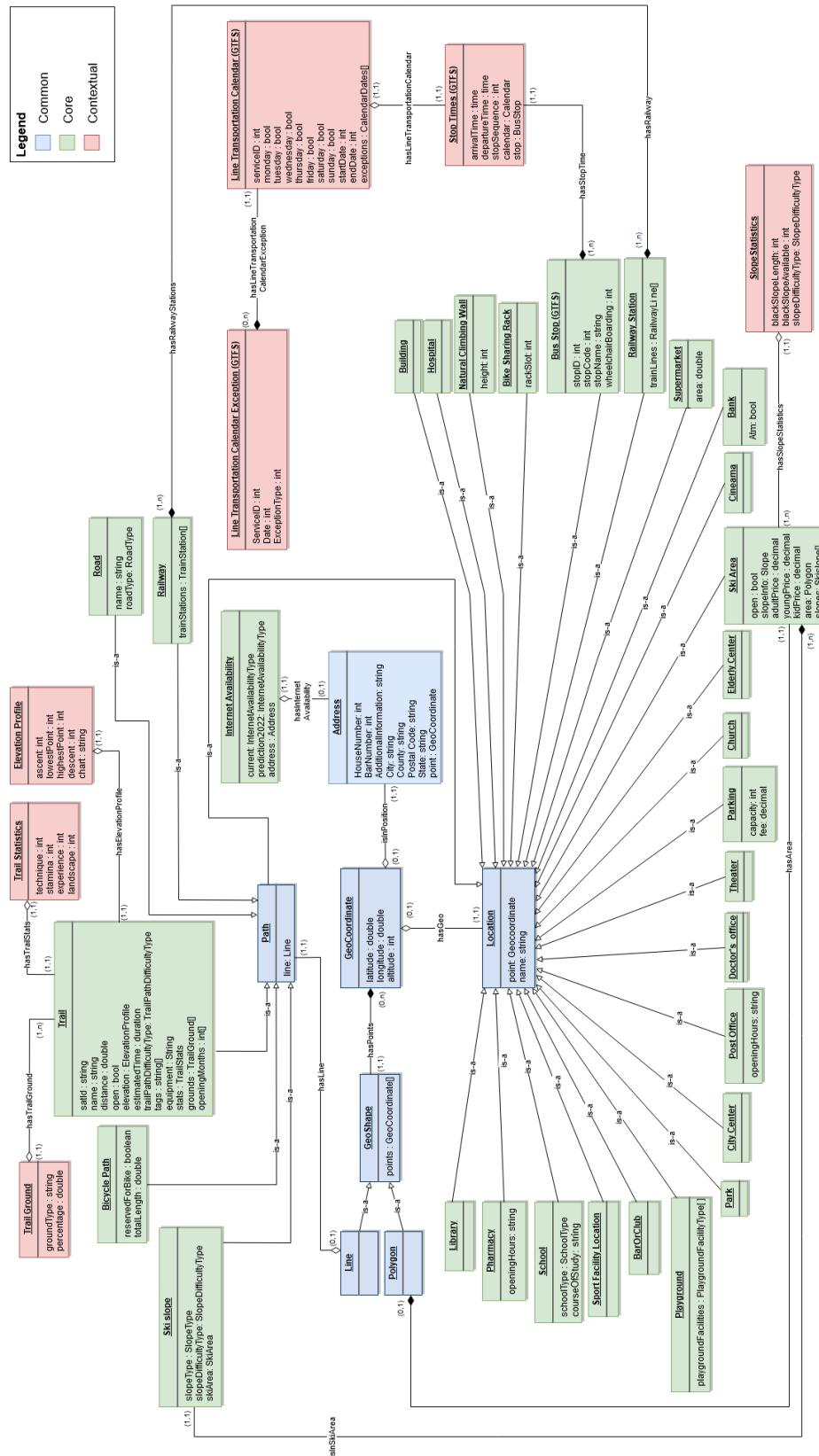


Figure 2: EER with the modification made during the execution of the formal phase

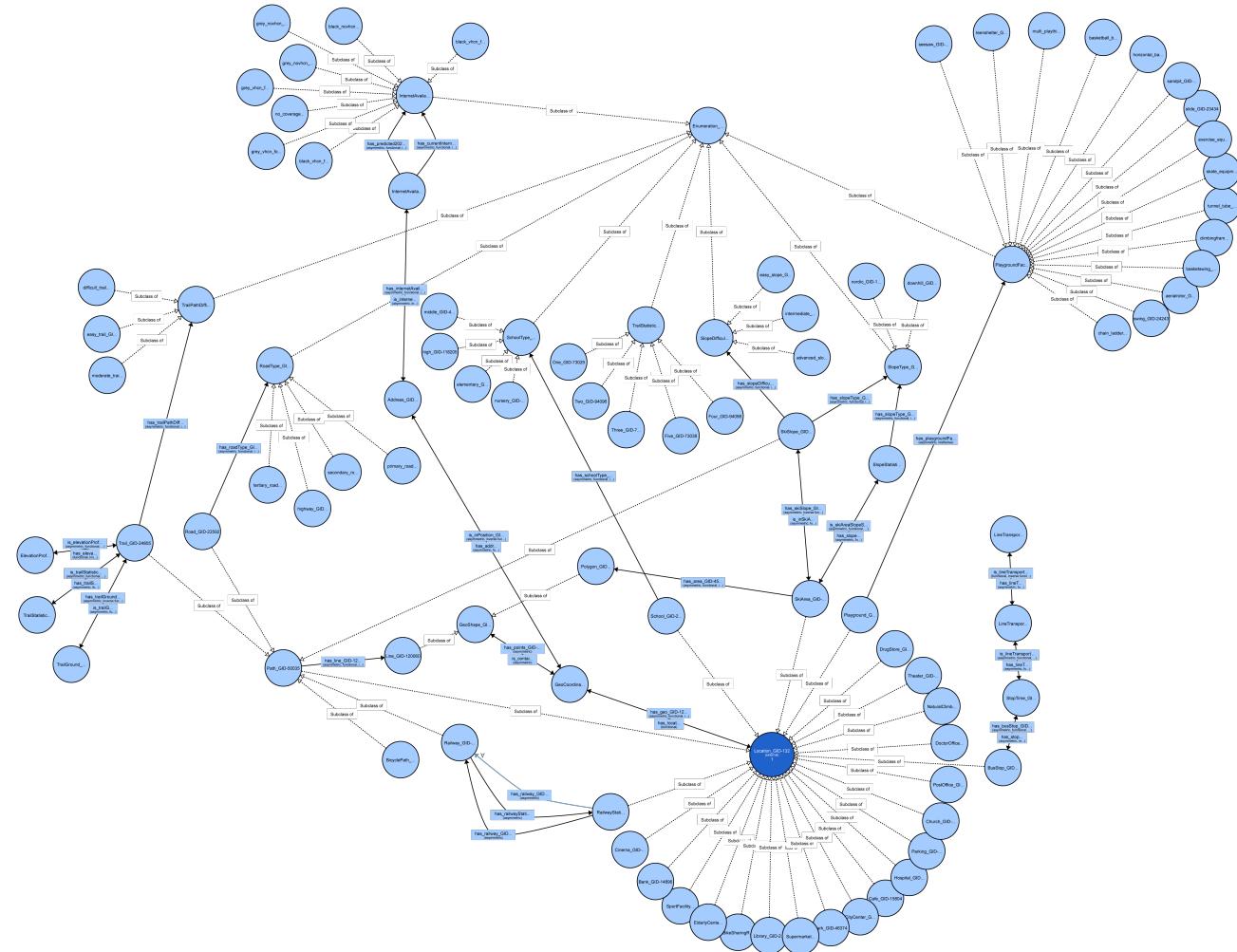


Figure 3: Image of the created schema under the form of a graph

In the following there is reported some ontology metrics in order to have an overview of the ontology.

Ontology Metrics			
Description	Total:	Added by us:	Already present in KOS .owl
Class count	122	98	24
Object property count	69	37	32
Data property count	92	63	29
Individuals count	8	0	8
Class axioms			
Description	Total:	Added by us:	Already present in KOS .owl
SubClass axioms count	218	197	21
Disjoint classes axioms count	15	15	0
Object property axioms			
Description	Total:	Added by us:	Already present in KOS .owl
Sub object property axioms count	65	37	28
Inverse object properties axioms count	12	12	0
Functional object property axioms count	27	27	0
Inverse functional object property axioms count	14	14	0
Anti-symmetric object property axioms count	34	34	0
Irreflexive object property axioms count	27	27	0
Object property domain axioms count	66	37	29
Object property range axioms count	66	37	29
Data property axioms			
Description	Total:	Added by us:	Already present in KOS .owl
Sub data property axioms count	86	63	23
Data property domain axioms count	89	63	26
Data property range axioms count	89	62	27
Annotation axioms			
Description	Total:	Added by us:	Already present in KOS .owl
Entity annotation axioms count	625	538	87

Moreover a list of metadata is reported in this section, in order to describe all the elements of the ontology defined.

1.4.2 Classes

In this table ³¹ there is the list of the classes of our ontology and for each of this classes there are the related concept, GID and comment. In the last column of the table there is a Boolean value that indicate if the concept related to that class is imported by us in the ontology or not.

1.4.3 Data Property

In this table ³² is instead reported the table that shows all the data properties of the ontology. For each data property in addition to the name is associated the name and the GID of the concept, the comment the domain and the range. In the last columns is reported the cardinality.

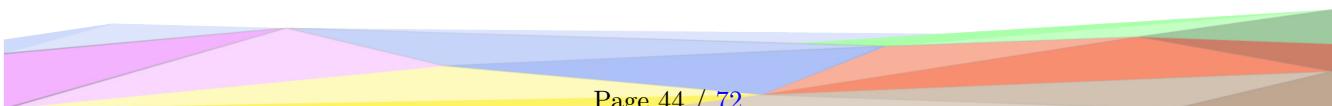
1.4.4 Object Property

In this table ³³ the object properties of our ontology are listed. For each line is reported the formal name of the object property, followed by the name and GID of the concept, the Domain and range information and a series of Booleans that define the characteristics of the property. In the last columns are finally reported the cardinality and the inverse object property.

³¹<https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/documentation/project%20report/Classes%20List.pdf>

³²<https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/documentation/project%20report/DataProperties%20List.pdf>

³³<https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/documentation/project%20report/Object%20Properties%20List.pdf>



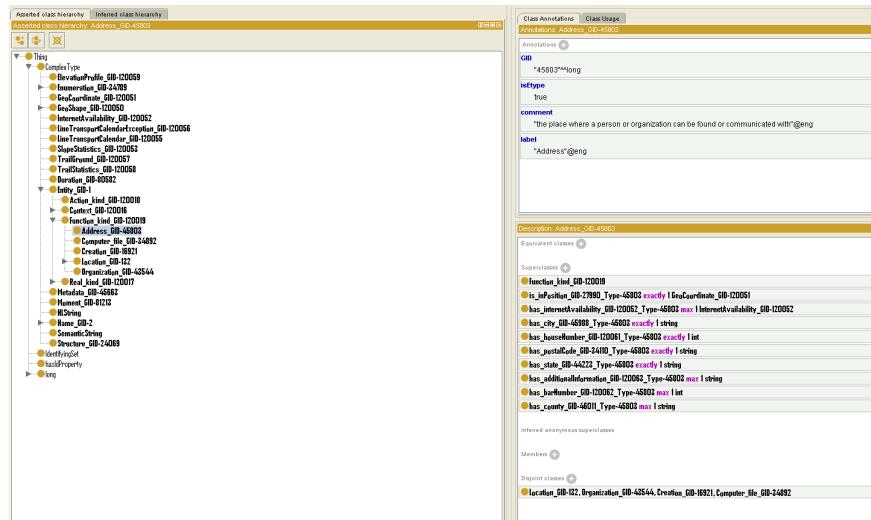
1.4.5 Enumeration Classes

In this table ³⁴ the enumeration classes are reported instead. To create these classes, given the constraints of the system that did not allow us to search for individuals, we adopted and designed a design pattern different from that commonly found in literature. In our ontology the enumeration classes are defined as classes, therefore there is an object property that connects them to the classes that need these enumerations, that present as their sub-classes the various values that enumeration can assume. With this pattern we can afford to go, in the future, to insert more information and detail to these enumeration classes too.

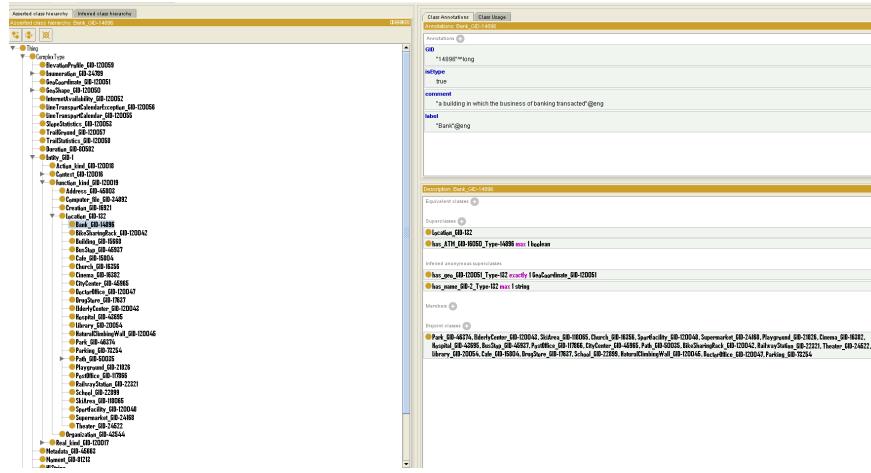
1.4.6 Focus on Classes

³⁴<https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/documentation/project%20report/Enumerations%20List.pdf>

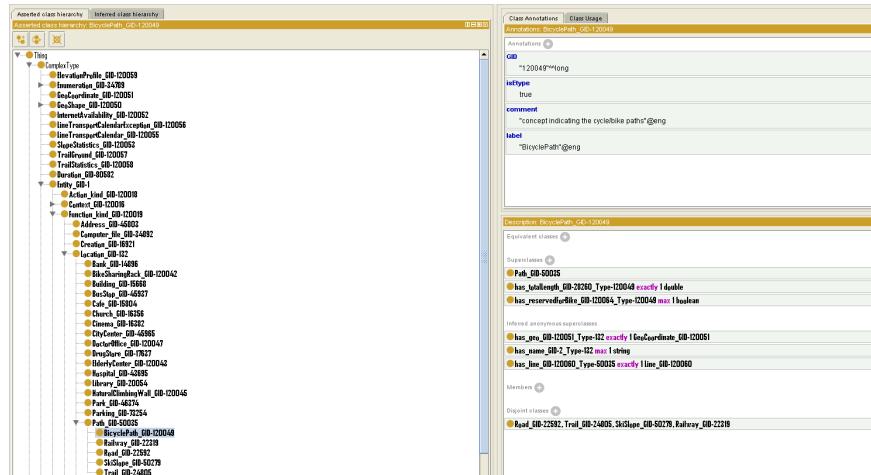
- Address_GID-45803



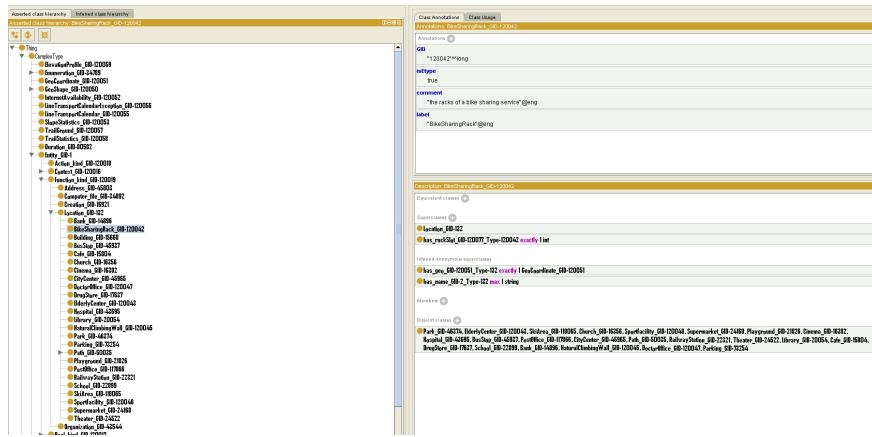
- Bank_GID-14896



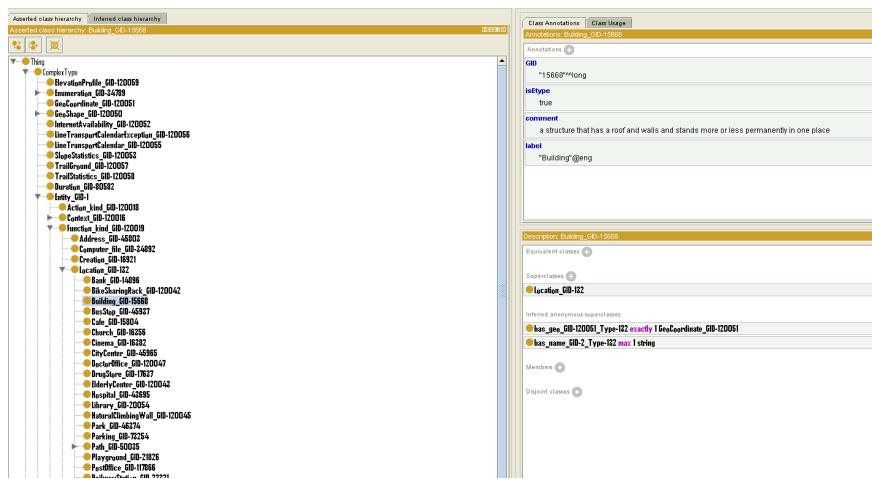
- Bicycle Path_GID-120049



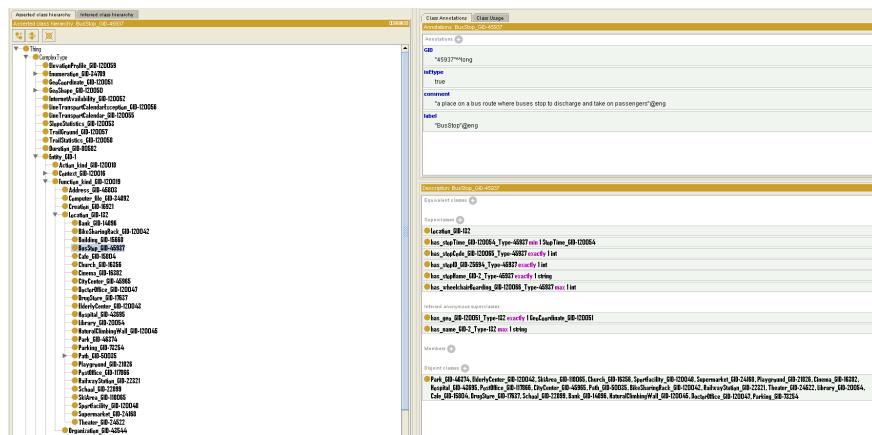
- Bike Sharing Rack_GID-120042



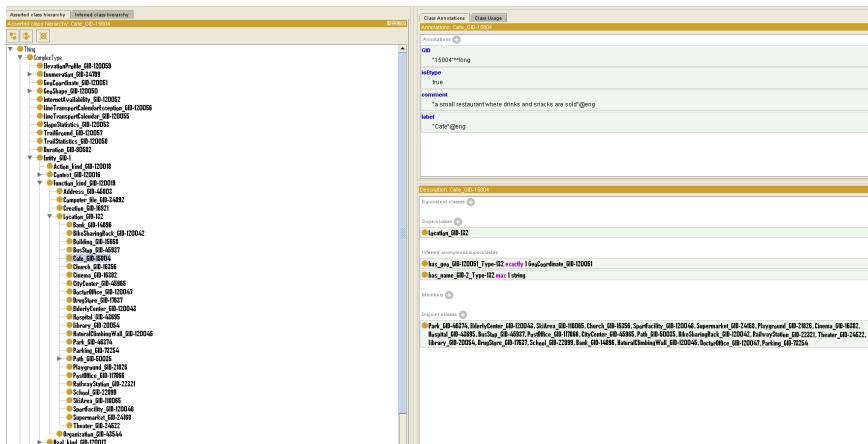
- Building_GID-15668



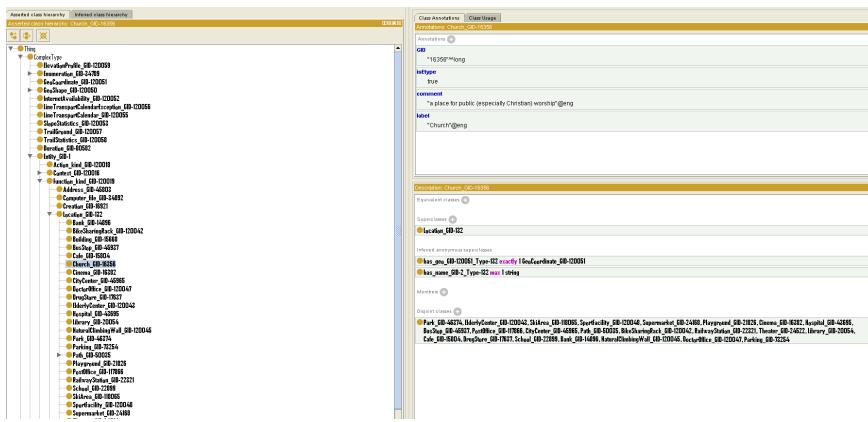
- Bus Stop_GID-45937



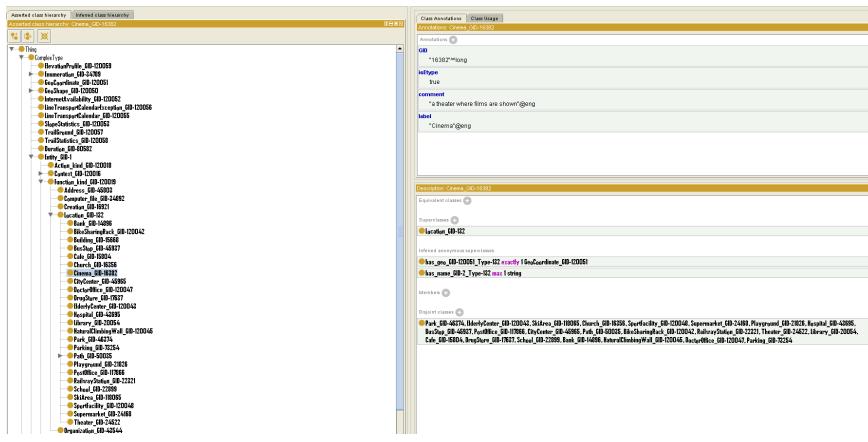
- Cafe_GID-15804



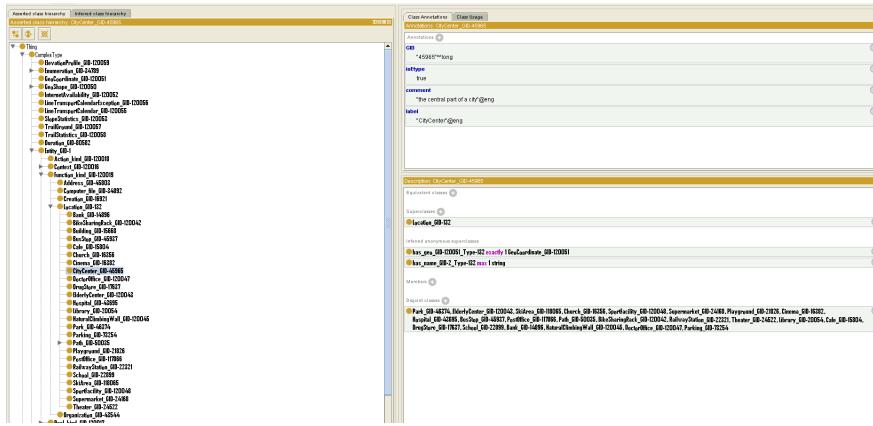
- Church_GID-16356



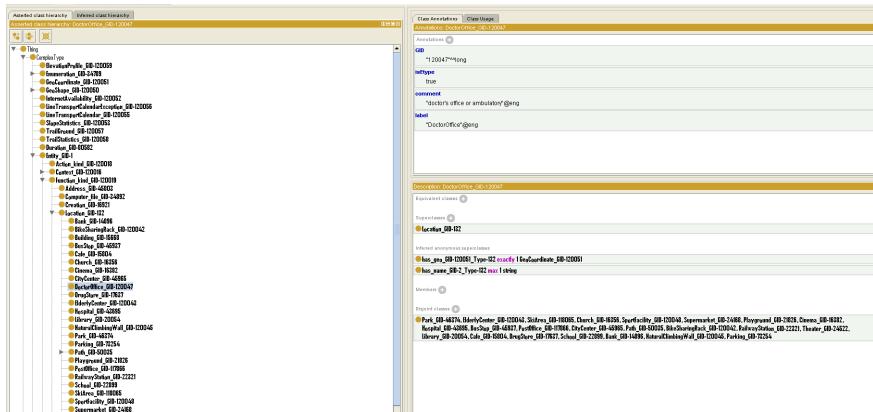
- Cinema_GID-16382



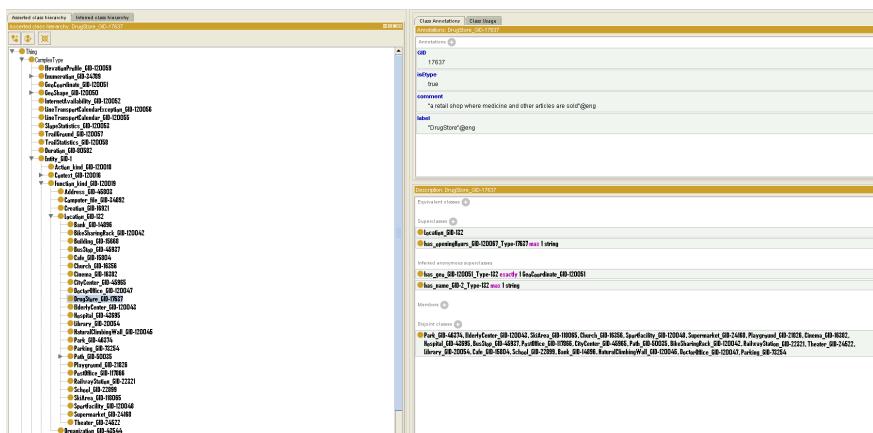
- City Center_GID-45965



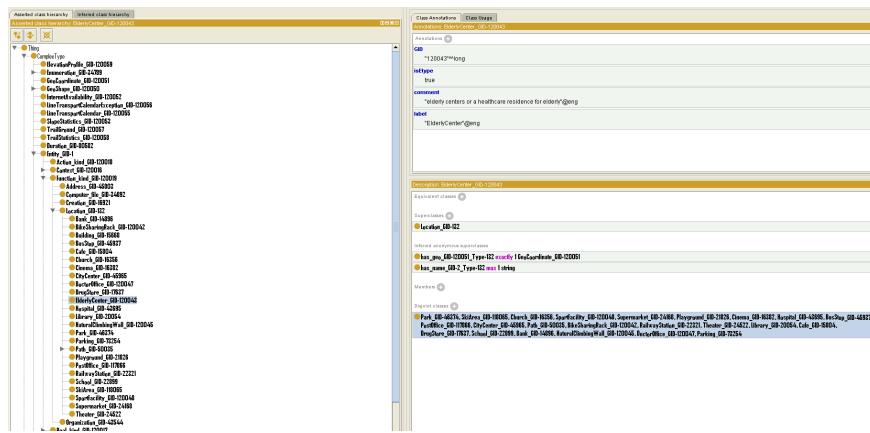
- Doctor's Office_GID-120047



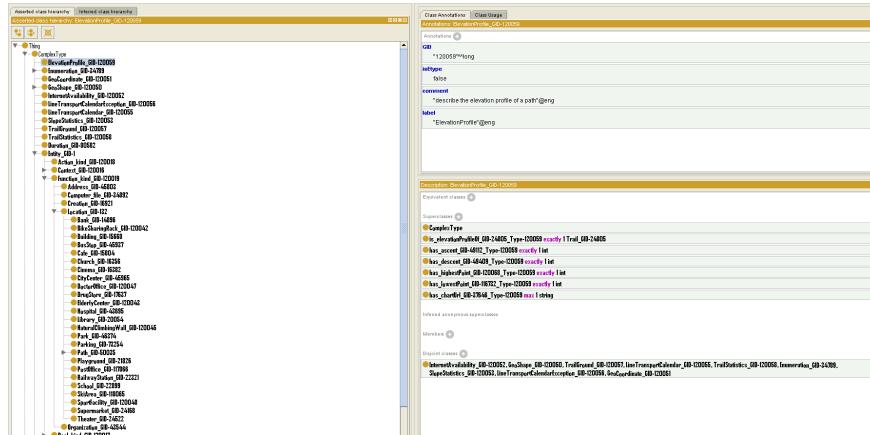
- Drug Store_GID-17637



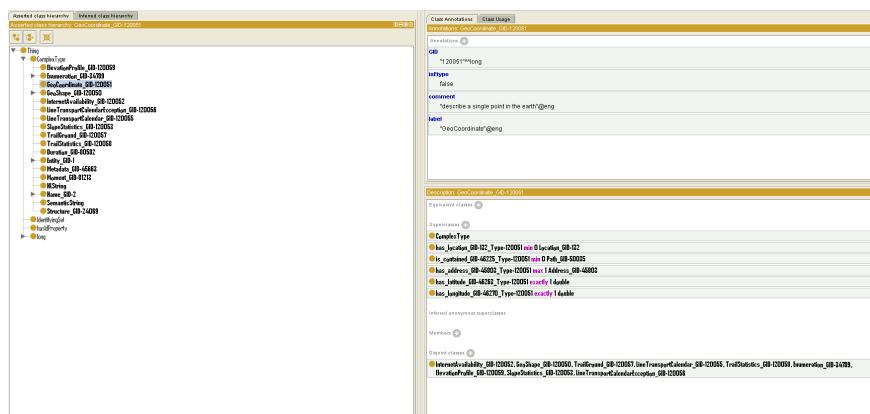
- Eldery Center_GID-120043



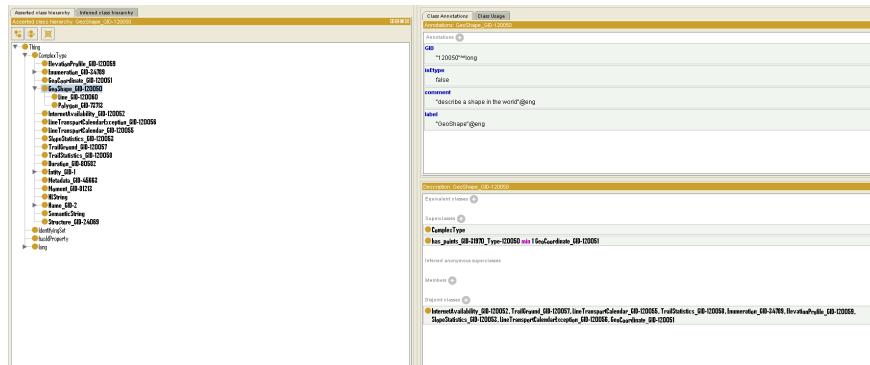
- Elevation Profile_GID-120059



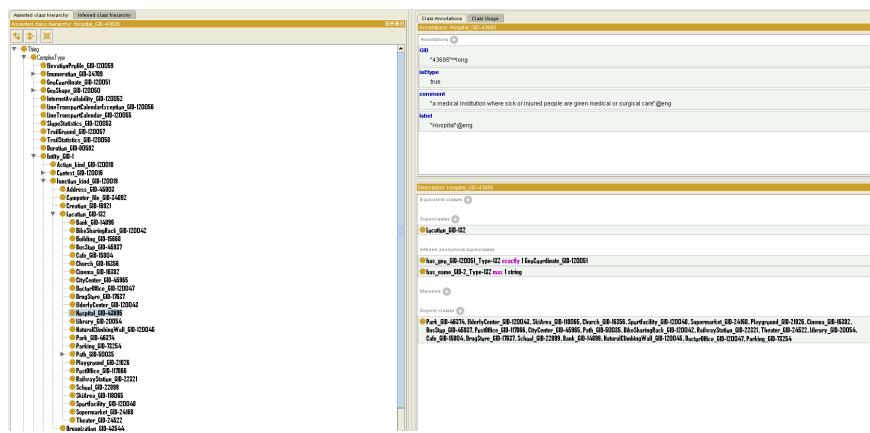
- GeoCoordinate_GID-120051



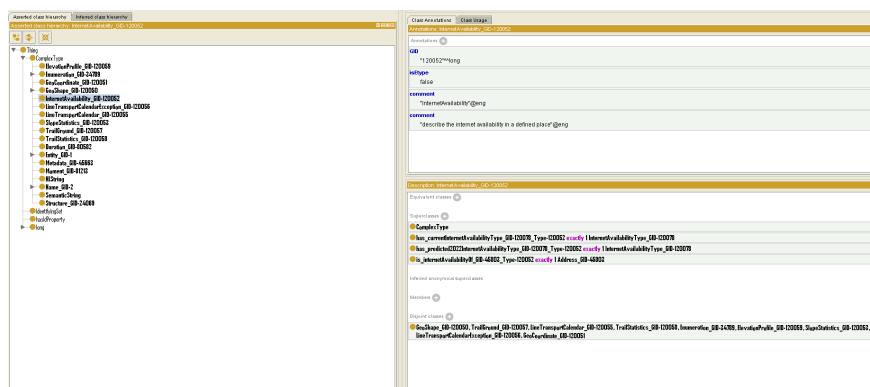
- GeoShape_GID-120050



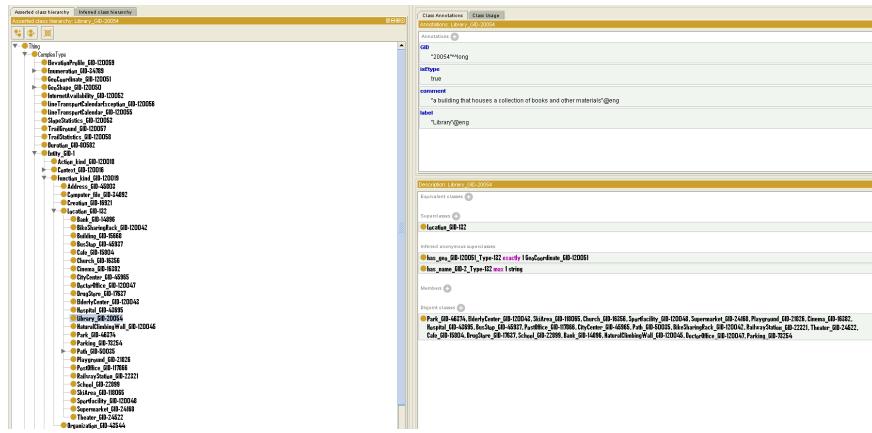
- Hospital_GID-43695



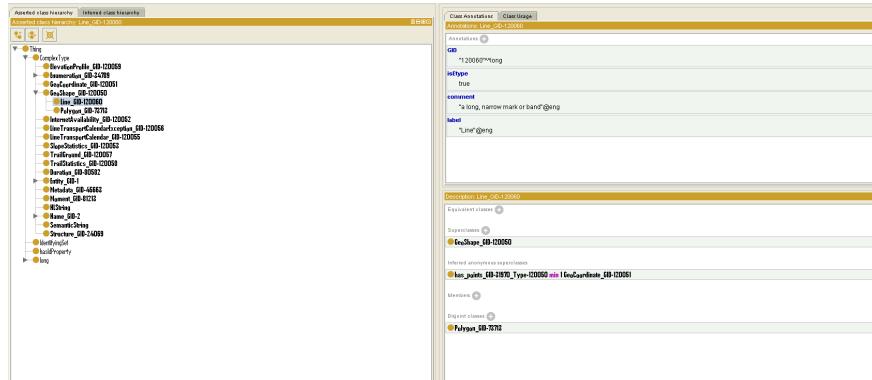
- Internet Availability_GID-120052



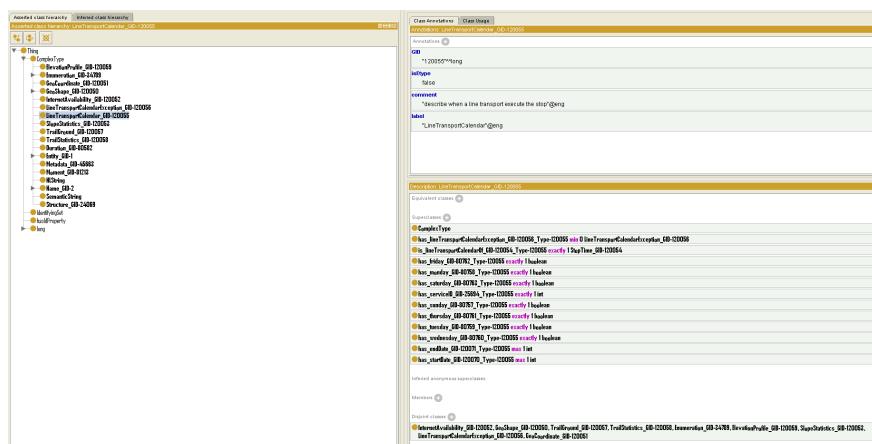
- Library_GID-20054



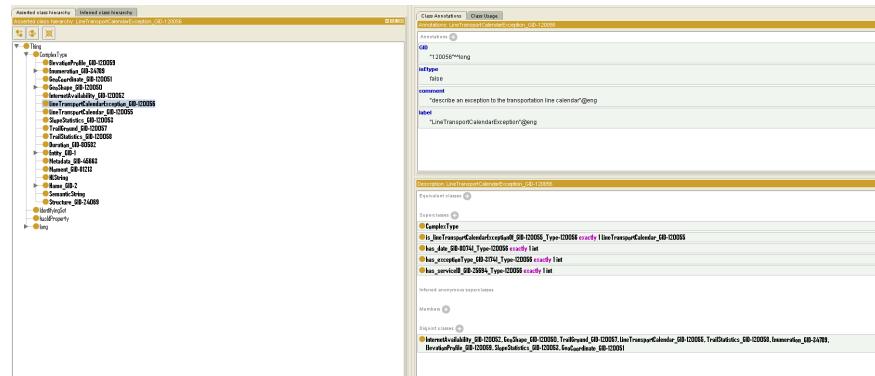
- Line_GID-120060



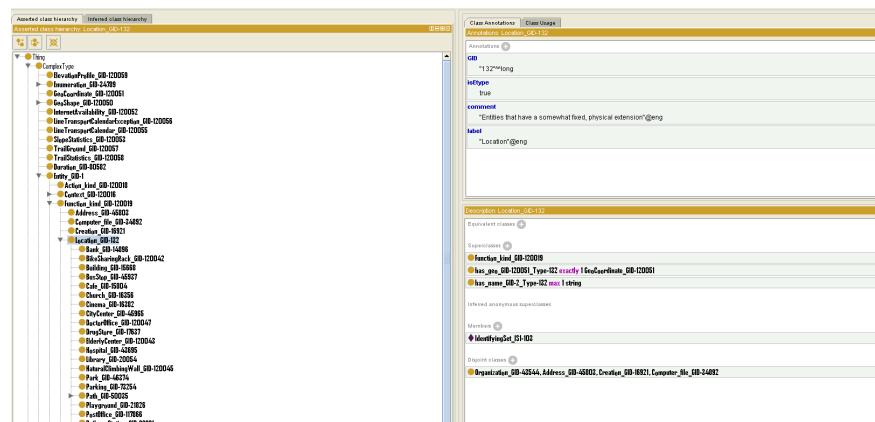
- Line Transport Calendar.GID-120055



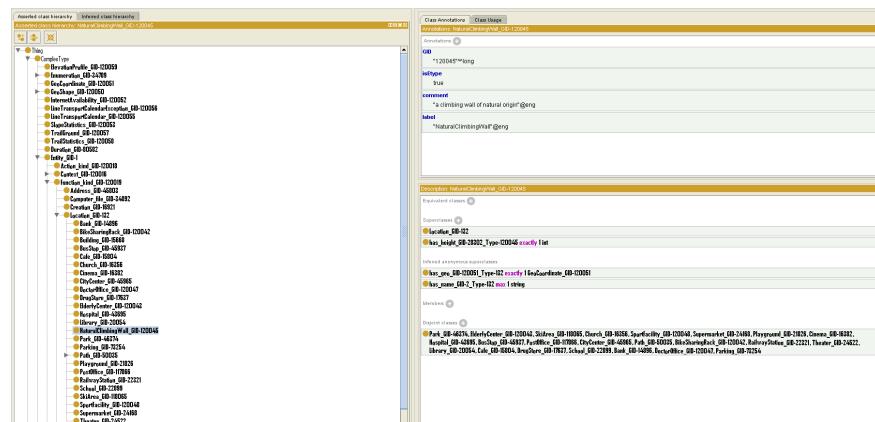
- Line Transport Calendar Exception_GID-120056



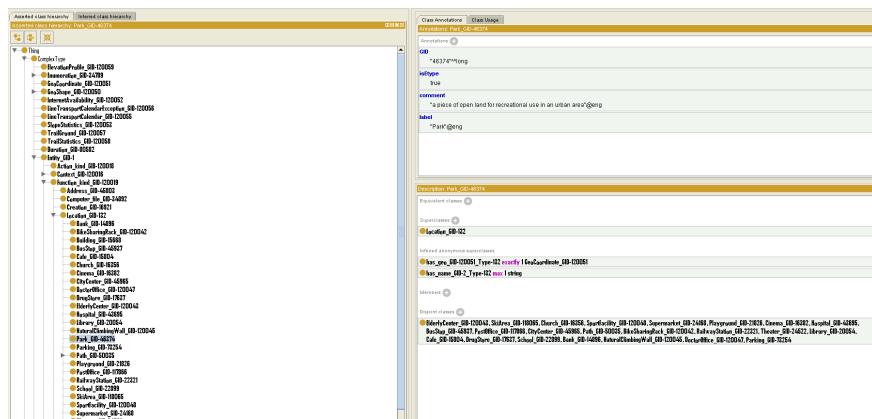
- Location_GID-132



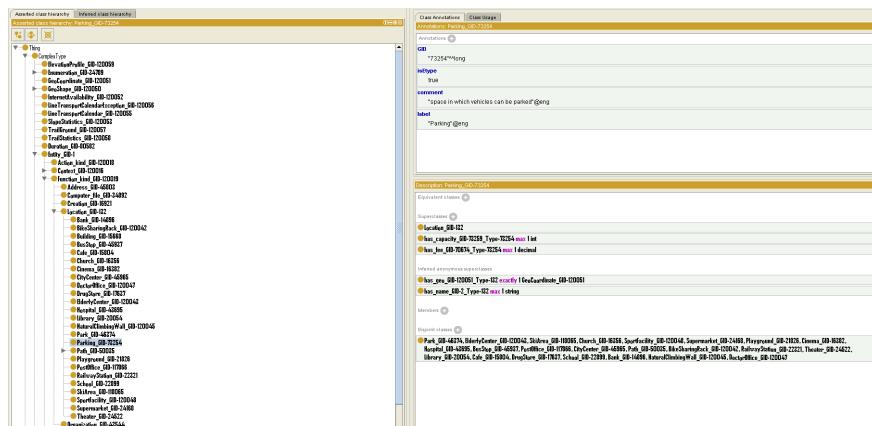
- Natural Climbing Wall_GID-120045



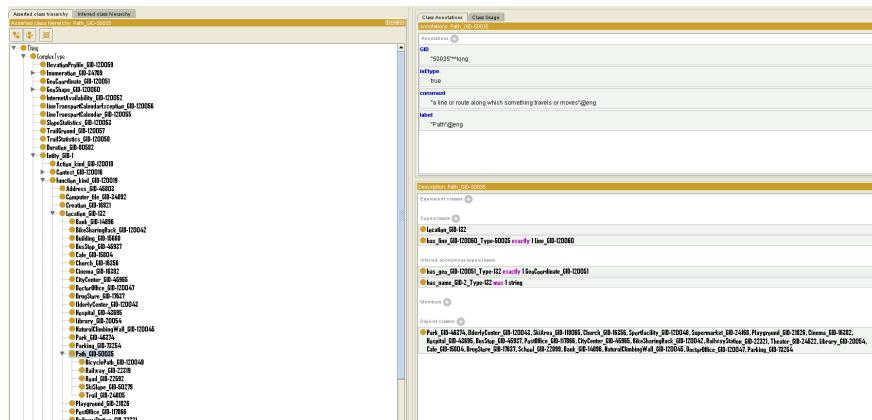
- Park_GID-46374



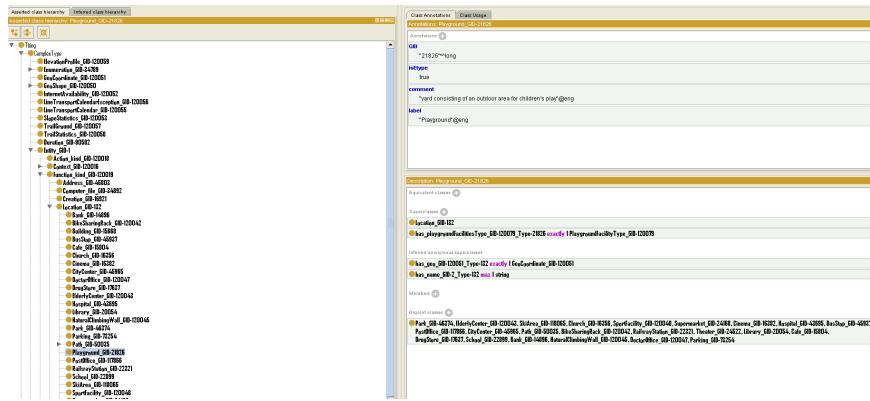
- Parking_GID-73254



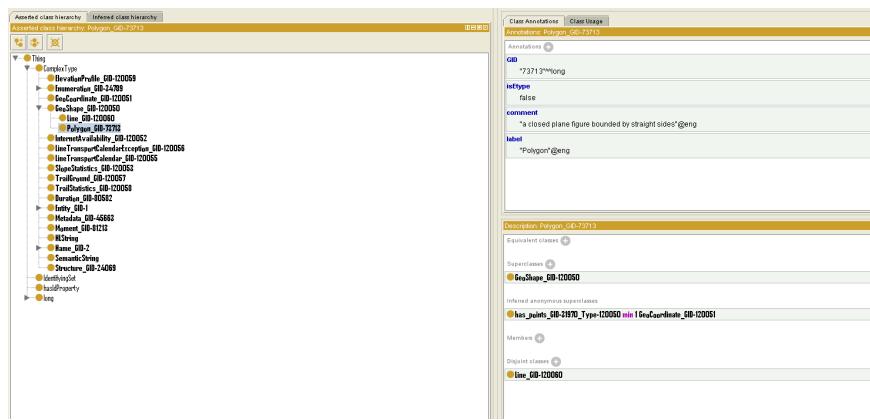
- Path_GID-50035



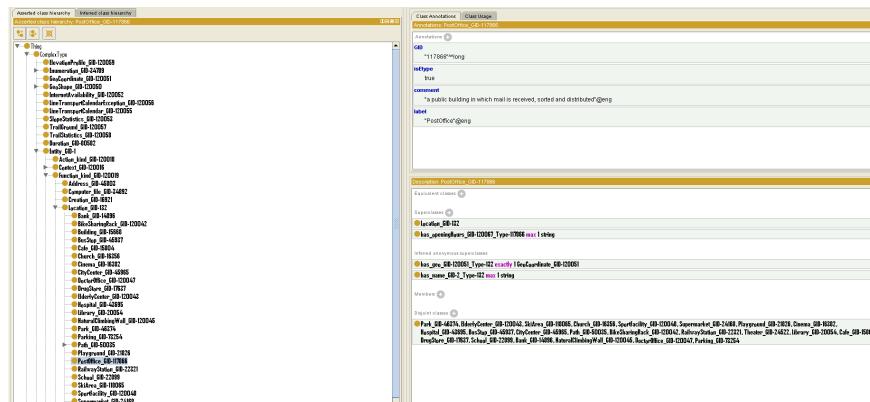
- Playground_GID-21826



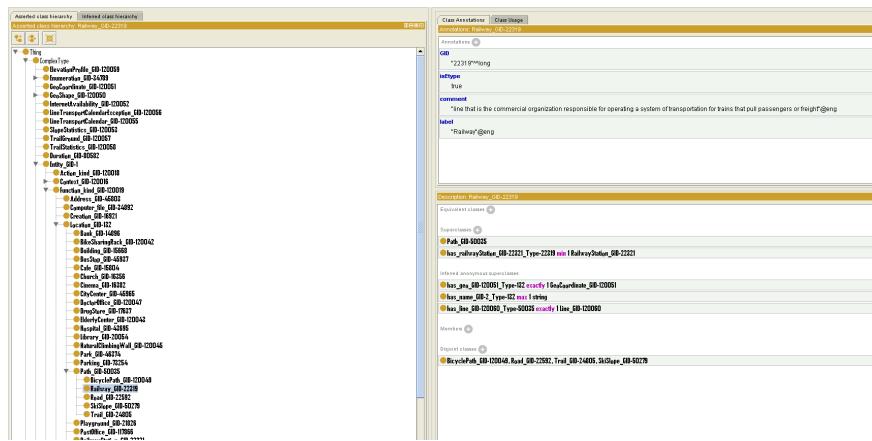
- Polygon_GID-73713



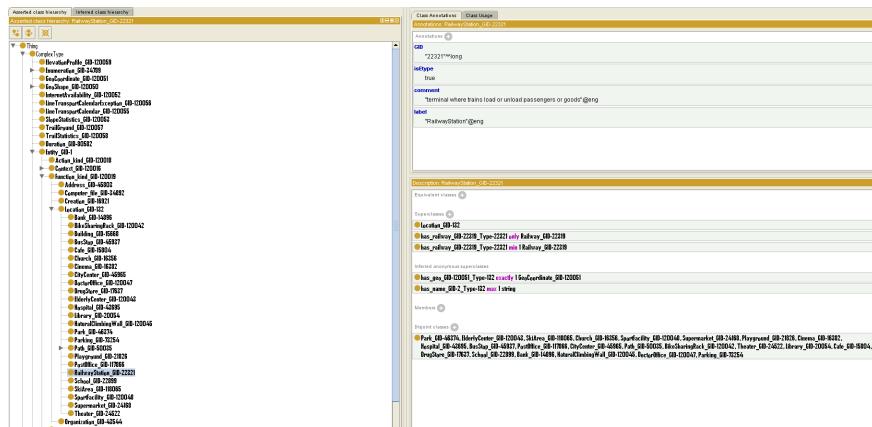
- Post Office_GID-117866



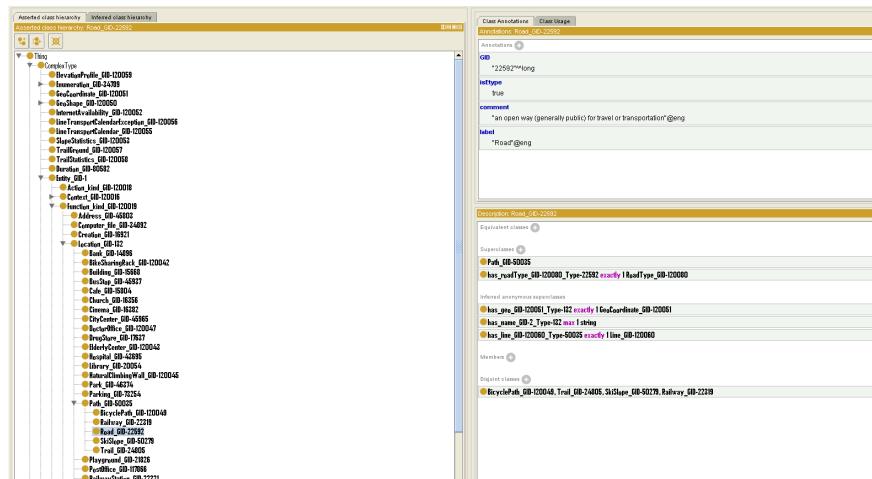
- Railway_GID-22319



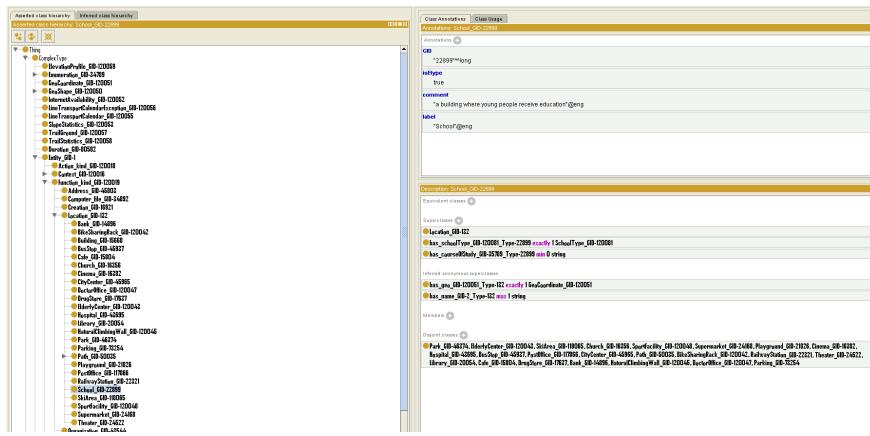
- Railway Station_GID-22321



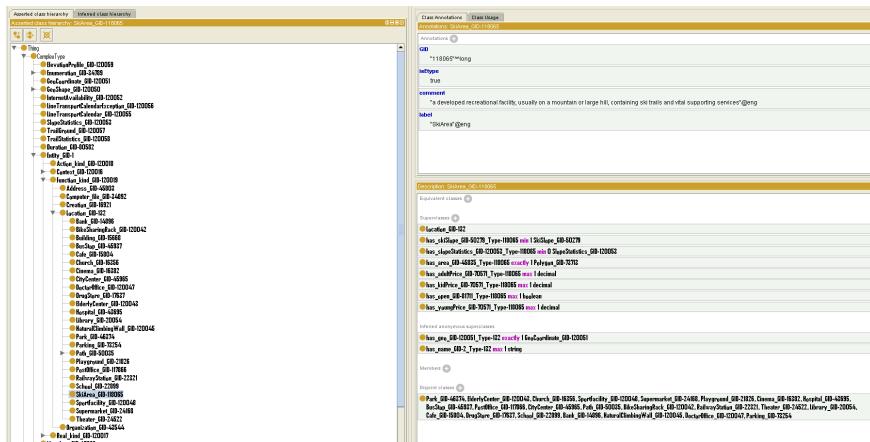
- Road_GID-22592



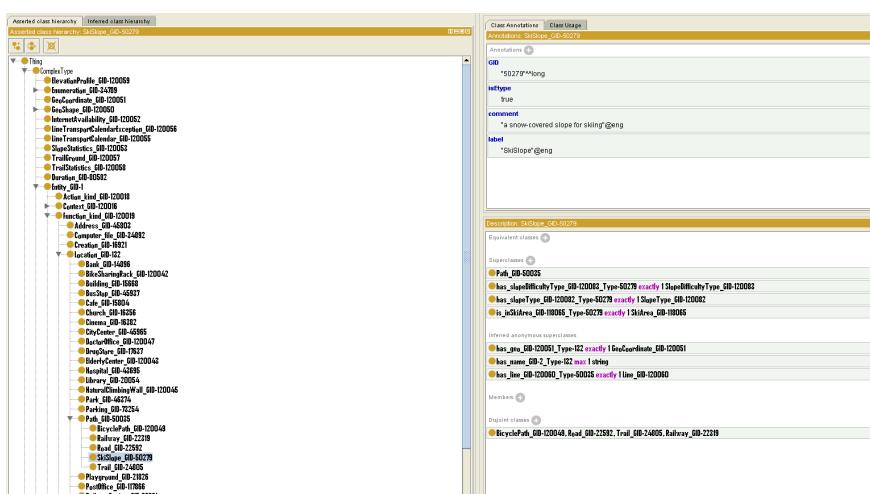
- School_GID-22899



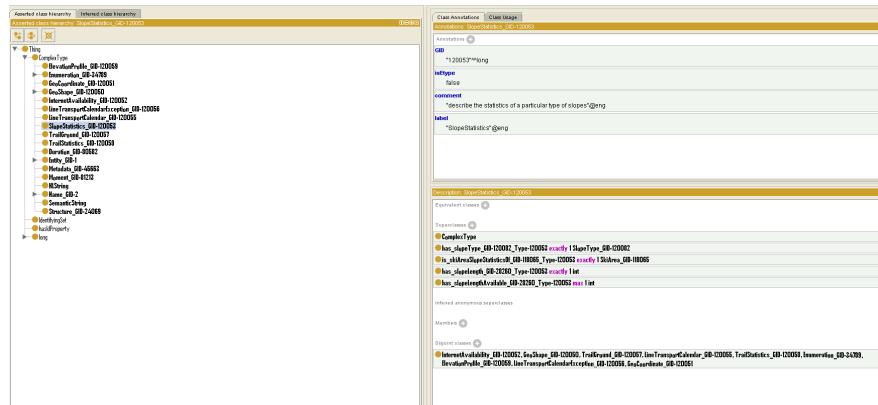
- Ski Area_GID-118065



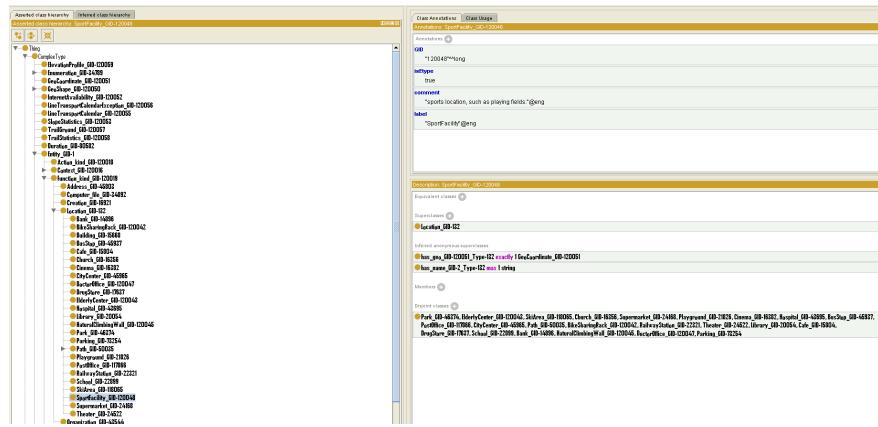
- Ski Slope_GID-50279



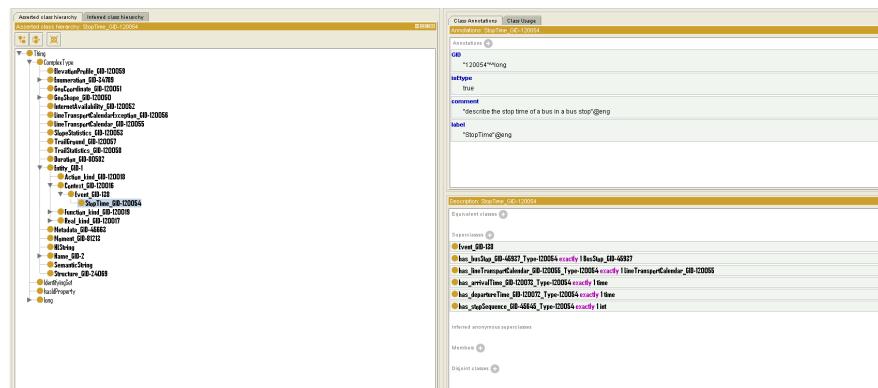
- Slope Statistics_GID-120053



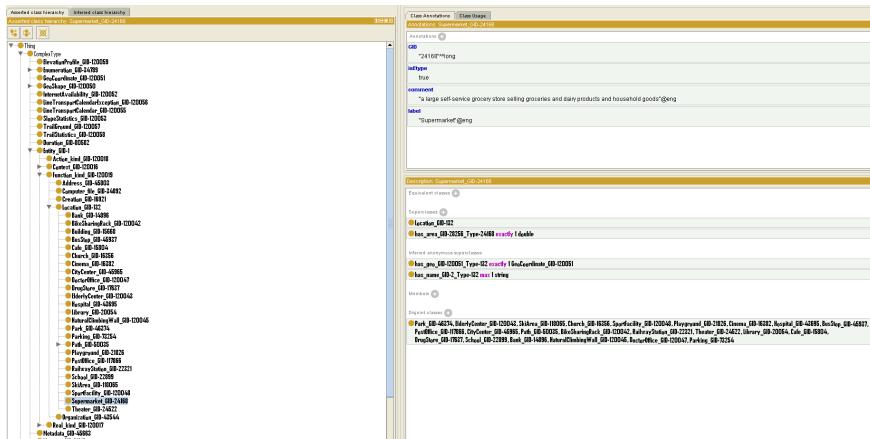
- Sport Facility_GID-120048



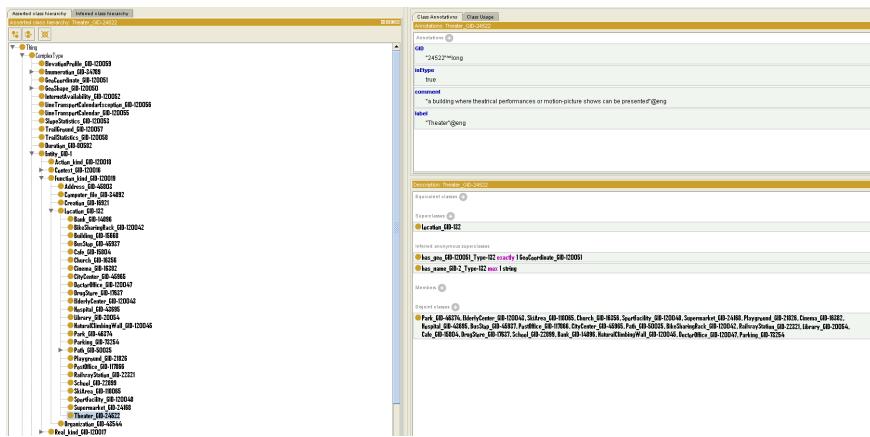
- Stop Time_GID-120054



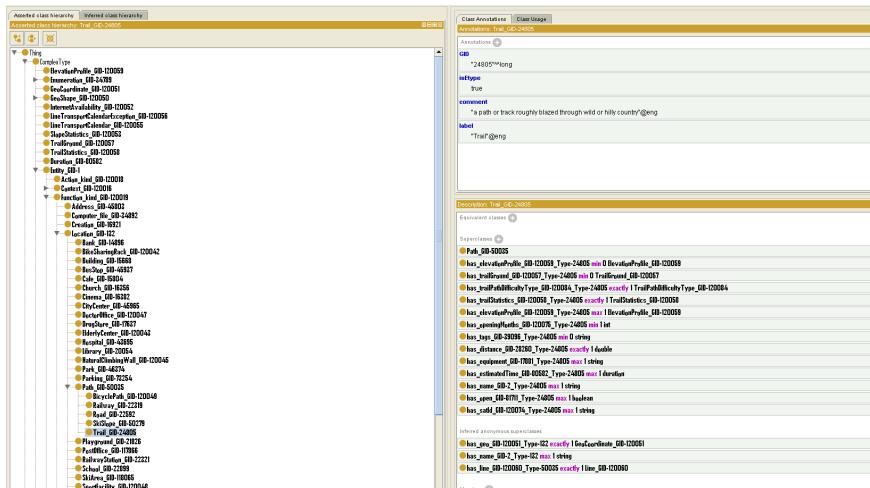
- Supermarket_GID-24168



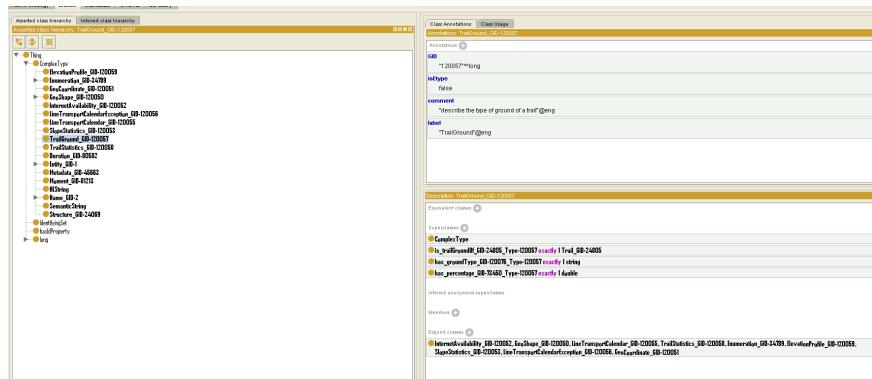
- Theater_GID-24522



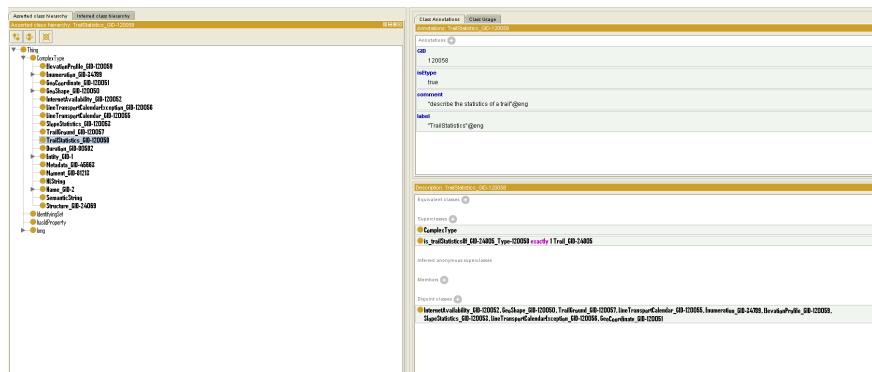
- Trail_GID-24805



- Trail_Ground_GID-120057



- Trail_Statistics_GID-120058



1.4.6.1 Variance respect to the EER Model

Once the ontology has been built, this section report the differences, and so the variance, respect the EER model defined in the previous phase. This a way to define the quality of the outcomes for the current phase as well as the alignment of the overall project development process. Compared to the EER model created in the previous phase, during this formalization phase the names of both classes and some attributes were changed, two new attributes were added and a small logical change of a class was made. These changes are mainly due to the link between class and concept found in KOS that allowed us to formalize classes. In addition, attributes have been added in the School class and the Natural Climbing Wall class. This addition is derived from a deeper analysis of the datasets we had available. Other small changes in attribute nomenclature, related to Trail and Trail related classes have been applied for a matter of alignment with the Tourist Facilities group. The most important conceptual change has been made with the elimination of the classes Pleasure Path and Transportation way that have been made to collapse in a single class named Path. This modification was guided by the meaning of Path found in ontology.

1.4.7 Data level

As in the previous phase the data level section here, reports the description of the new version of the datasets, after formatting operations.

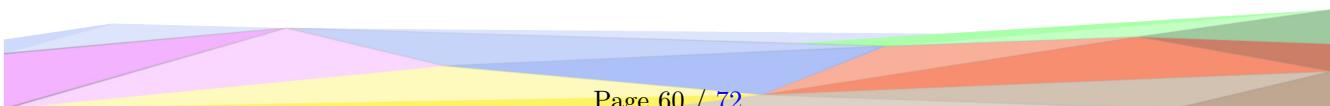
1.4.7.1 Formal Modeling datasets management

In the formal phase we organized better the data in order to align them to the ER schema:

- In the point of interests datasets we extracted some of the entities in the ER: PerformingArtsTheater, eldery center, Physician, BarOrClub, Library, SportFacilityLocation, church. We also changed the name of some attribute in order to align them with the attributes of the entities. We used the script `point_of_interests_allineation.py`³⁵ to make this work.
- The bikesharing datasets were merged in order to have a single file with all the bikesharing stations (renaming some attributes) using the script `bikesharing_allineation.py`³⁶.

³⁵https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/point_of_interests_allineation.py

³⁶https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/bikesharing_allineation.py



- The building datasets were merged in order to have all the buildings in a single file (renaming some attributes) using the script `building_allineation.py`³⁷.
- The school datasets were merged in order to have all the schools in a single file renaming some attribute. We also reprojected the coordinates in the EPSG 4326 standard to align with the rest of the datasets. We used the script `school_allineation.py`³⁸ to do this work.
- The rest of the datasets were aligned using the script `everything_allineation.py`³⁹ which it has renamed some of the attributes in order to align with the ER and it has reprojected the coordinates present in `piste.ciclabilis.json` in the EPSG 4326 standard. It rename also the attributes in the metadatas.

1.4.7.2 Datasets metadata documentation

In this section eventually new metadata information are added in order to describe the evolution of the datasets.

areaski_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<code>type</code>	FeatureCollection		
<code>generator</code>	overpass-ide		
<code>copyright</code>	The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.		
<code>timestamp</code>	2020-10-19T19:05:03Z		
Attributes			
<code>name</code>	name of data	string	
<code>GeoShape</code>	object containing location information	GeoShape	Common
<code>id</code>	name that identifies a unique object	string	
<code>type</code>	type of GeoShape (Line, Polygon, Point)	string	Common
<code>GeoCoordinates</code>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

bank_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<code>type</code>	FeatureCollection		
<code>generator</code>	overpass-ide		
<code>source</code>	www.openstreetmap.org		
<code>copyright</code>	ODbL		
<code>timestamp</code>	2020-10-19T13:50:02Z		
<code>update_frequency</code>	no update		
<code>Geographical Coverage</code>	Provincia di Trento		
<code>format</code>	json		
Attributes			
<code>name</code>	name of the facility	string	core
<code>atm</code>	indicate the presence of an atm	boolean	core
<code>addr:housenumber</code>	the house number component of the address	string	common
<code>addr:street</code>	the street component of the address	string	common
<code>addr:city</code>	the city component of the address	string	common
<code>addr:postcode</code>	the postcode component of the address	string	common
<code>GeoShape</code>	object containing location information	GeoShape	Common
<code>id</code>	identification code	string	common
<code>type</code>	type of GeoShape (Line, Polygon, Point)	string	Common
<code>GeoCoordinates</code>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

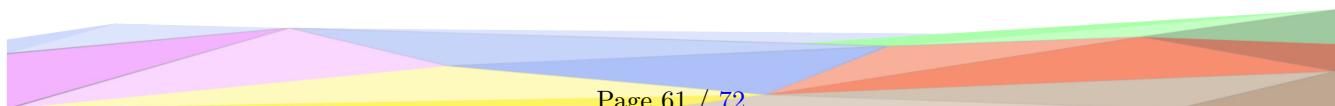
busstop_METADATA.json:

Dataset Properties	Description	Type	Data Definition

³⁷https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/building_allineation.py

³⁸https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/school_allineation.py

³⁹https://github.com/UNITN-KDI-2020/GeoSpatial-Domain-project/blob/master/code/everything_allineation.py



<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>copyright</i>	The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.		
<i>timestamp</i>	2020-10-19T10:54:03Z		
Attributes			
<i>name</i>	name that identifies the name of the bus-stop	string	
<i>addr:city</i>	name identifying the name of the city where the bus stop is located	string	
<i>addr:postcode</i>	number identifying the postcode of the city where the bus stop is located	int	
<i>addr:street</i>	name identifying the name of the street where the bus stop is located	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	name that identifies a unique object	string	
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

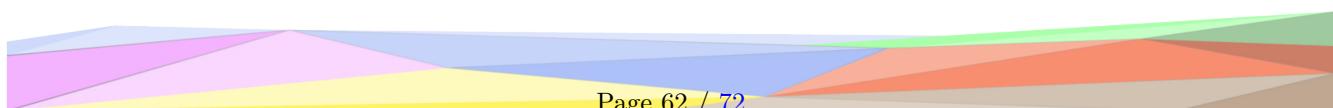
cinema_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>copyright</i>	The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.		
<i>timestamp</i>	2020-10-19T15:03:03Z		
Attributes			
<i>name</i>	Identifies the name of the cinema	string	
<i>addr:city</i>	Identifies in term of cities the position of the cinema	string	
<i>addr:housenumber</i>	Identifies the house number of the cinema	int	
<i>addr:postcode</i>	Identifies the postcode of the cities where the cinema is located	int	
<i>addr:street</i>	Identifies the street name where the cinema is located	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

city_center_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T10:54:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	the name of the area	string	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

climb_METADATA.json:



Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T15:21:03Z		
<i>update-frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	Identifies the name of the climb	string	core
<i>addr:housenumber</i>	Indicates the housenumber in the address information	string	common
<i>addr:postcode</i>	Indicates the postcode in the address information	int	common
<i>addr:street</i>	Indicates the postcode in the address information	string	common
<i>opening_hours</i>	Indicates the opening hours of the climb spot	string	core
<i>height</i>	Indicates the height of the climbing	int	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	Used to univocally identifies the object	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

park_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T19:30:02Z		
<i>update-frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>leisure</i>	type of facility inside	string	core
<i>name</i>	the name of the facility	string	core
<i>playground</i>	describe the type of playground available	string	contextual
<i>playground:basketswing</i>	describe if the facility have a basketswing	boolean	contextual
<i>playground:chain_ladder</i>	describe if the facility have a chain ladder	boolean	contextual
<i>playground:slide</i>	describe if the facility have a slide	boolean	contextual
<i>playground:swing</i>	describe if the facility have a swing	boolean	contextual
<i>opening_hours</i>	the opening hours of the park	string	contextual
<i>addr:city</i>	the city component of the address	string	common
<i>addr:housenumber</i>	the house number component of the address	string	common
<i>addr:postcode</i>	the postal code component of the address	string	common
<i>addr:street</i>	the street component of the address	string	common
<i>playground:aerialrotator</i>	describe if the facility have an aerialrotator	boolean	contextual
<i>playground:basketball_backboard</i>	describe if a zone where play basketball is available	boolean	contextual
<i>playground:exercise</i>	describe if an equipment are for the exercise is present	boolean	contextual
<i>playground:horizontal_bar</i>	describe if an horizontal bar is present	boolean	contextual
<i>playground:seesaw</i>	describe if an seesaw is present	boolean	contextual
<i>playground:tunnel_tube</i>	describe if an tunnel tube is present	boolean	contextual
<i>playground:climbingframe</i>	describe if a climbing frame is present	boolean	contextual
<i>playground:multi_play</i>	describe if a multiple usage surface is available	boolean	contextual
<i>playground:sandpit</i>	describe if a sandpit is present	boolean	contextual
<i>playground:theme</i>	describe the theme of the park	string	contextual
<i>playground:skate_equipment</i>	describe if the park is equipment to play with the skateboard	boolean	contextual
<i>playground:teenshelter</i>	describe if a teen shelter is available	boolean	contextual
<i>playground:water</i>	describe if a water playground is available	boolean	contextual
<i>GeoShape</i>	object containing location information	GeoShape	Common

<i>id</i>	name that identifies a unique object	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

parking_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T09:57:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>fee</i>	The fee tag is for specifying whether a fee is usually charged for a service, or for access.	boolean	core
<i>capacity</i>	max capacity of the parking lot	int	core
<i>name</i>	name of the location	string	core
<i>addr:city</i>	name of the city	string	common
<i>addr:housenumber</i>	civic code	int	common
<i>addr:postcode</i>	postal code of the location	int	common
<i>addr:street</i>	street address of the location	string	common
<i>opening_hours</i>	day and hours of opening	string	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	identification code	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

pharmacy_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T15:28:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	name of the location	string	core
<i>addr:city</i>	city of the location	string	common
<i>addr:housenumber</i>	civic code	int	common
<i>addr:postcode</i>	postcode of the location	int	common
<i>addr:street</i>	street address of the location	string	common
<i>opening_hours</i>	day and hours of opening	string	common
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	identification code	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

post_office_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		

<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T15:26:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>addr:city</i>	city component of the address	string	common
<i>addr:housenumber</i>	the house number component of the address	string	common
<i>addr:postcode</i>	the postal code component of the address	string	common
<i>addr:street</i>	the street component of the address	string	common
<i>name</i>	the name of the facility	string	core
<i>opening_hours</i>	the opening hours of the facility	string	core
<i>addr:province</i>	the province component of the address	string	common
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	name that identifies a unique object	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

railway_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T18:07:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	name of te location	string	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	identification code	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

roads_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-20T07:44:03Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>highway</i>	type of road (trunk,primary, secondary, tertiary)	string	contextual
<i>name</i>	name of the road	string	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	identification code	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

skislopes_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T19:07:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	name of the ski slopes	string	core
<i>piste:type</i>	type of the skiing available (nordic, downhill)	string	core
<i>piste:difficulty</i>	difficulty level of the skislope (easy, intermediate, advanced)	string	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	identification code	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

supermarket_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T13:28:02Z		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	name of the supermarket	string	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	identification code	string	core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

trails_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>lastModifiedAt</i>	2020-07-02T08:14:00.000Z		
<i>source</i>	www.outdooractive.com		
<i>update_frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>sat_Id</i>	trail id in the SAT registry	String	
<i>trainType</i>	specifies whether the trail is a hiking, biking or snowshoe trail	String	
<i>name</i>	name of the trail	String	
<i>distance</i>	length of the trail	Double	
<i>open</i>	whether the trail is currently open	Boolean	
<i>elevation</i>	elevation details	ElevationProfile	
<i>estimatedTime</i>	detail on the estimated time required	Duration	
<i>startRoute</i>	starting point for the trail	LocationPoint	
<i>endRoute</i>	ending point of the trail	LocationPoint	
<i>route</i>	collection of points marking the trail path	Route	

<i>difficulty</i>	description of the trail difficulty	String	
<i>tags</i>	array of tags	array	
<i>equipment</i>	equipments needed for the trail	String	
<i>stats</i>	scores for the trail	TrailStats	
<i>grounds</i>	array of types of grounds in the trail	array	
<i>monthTips</i>	optional monthly variance details	array	
<i>metadata</i>	metadata	MetaData	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

building_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-10-19T13:50:02Z		
<i>update-frequency</i>	no update		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	identification of the record	string	Core
<i>city</i>	city of the building	string	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

civici_web_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Dataset ID</i>	4f474fc8-181d-4b15-9ee4-60b3f54f4068		
<i>Source</i>	https://dati.trentino.it/dataset/comune-di-trento-numeri-civici		
<i>Release date</i>	22-02-2013		
<i>Modification date</i>	08-05-2019		
<i>Geographical Coverage</i>	comune di Trento		
<i>Temporal extension</i>	'From': '22-02-2013'		
<i>Holder</i>	'Name': 'Comune di Trento', 'Code IPA/IVA': 'c_l378'		
<i>Update frequency</i>	daily		
<i>Format</i>	json		
Attributes			
<i>civico_num</i>	civic number (without slash)	string	common
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>address</i>	street description	string	common
<i>city</i>	Cadastral community	string	common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

piste_ciclabili_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>Dataset ID</i>	c.l378-1129110		
<i>Holder</i>	'Name': 'Comune di Trento', 'Code IPA/IVA': 'c_l378'		
<i>Release date</i>	09-11-2017		
<i>Modification date</i>	08-05-2019		

<i>Geographical Coverage</i>	Comune di Trento		
<i>Source</i>	https://dati.trentino.it/dataset/piste-ciclabili-open-data		
<i>GeoNames URI</i>	http://www.geonames.org/6541469		
<i>Dataset language</i>	italiano		
<i>Update frequency</i>	continuous updating		
<i>Format</i>	json		
Attributes			
<i>tipo</i>	name of the typology of path (not useful)	string	
<i>descrizione</i>	name of the path (equivalent to 'fumetto')	string	common
<i>tipologia</i>	path dedicated only to cycle or also pedestrian	string	core
<i>tratto isolato ciclabile</i>	meters without path dedicated to cycle	int	core
<i>tratto isolato ciclopedonale</i>	meters without path dedicated to cycle or pedestrian	int	core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>address</i>	name of the path	string	common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

schools_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>high_school</i>	'Release date': '19-09-2020', 'source': ' https://www.comuniecitta.it/scuole-secondarie-di-secondo-grado/comune-di-trento-22205 ', 'Geographical Coverage': 'Provincia di Trento', 'Update frequency': 'unknown', 'Format': 'json'		
<i>middle_school</i>	'Dataset ID': 'c.l378-1129125', 'Release date': '09-11-2017', 'Modification date': '08-05-2019', 'Geographical Coverage': 'Provincia di Trento', 'URI di GeoNames': ' http://www.geonames.org/6541469 ', 'source': ' https://dati.trentino.it/dataset/localizzazione-delle-scuole-medie-open-data ', 'Holder': 'Nome': 'Comune di Trento', 'Code IPA/IVA': 'c.l378', 'Update frequency': 'continuously updated', 'Format': 'json'		
<i>nursery_school</i>	'Dataset ID': 'c.l378-1129127', 'Release date': '09-11-2017', 'Modification date': '08-05-2019', 'Geographical Coverage': 'Provincia di Trento', 'source': ' https://dati.trentino.it/dataset/localizzazione-scuole-dell-infanzia-open-data ', 'GeoNames URI': ' http://www.geonames.org/6541469 ', 'Holder': 'Nome': 'Comune di Trento', 'Code IPA/IVA': 'c.l378', 'Update frequency': 'continuously updated', 'Format': 'json'		
<i>elementary_school</i>	'Dataset ID': 'c.l378-1129126', 'Release date': '09-11-2017', 'Modification date': '08-05-2019', 'Geographical Coverage': 'Provincia di Trento', 'Source': ' https://dati.trentino.it/dataset/localizzazione-delle-scuole-elementari-open-data ', 'GeoNames URI': ' http://www.geonames.org/6541469 ', 'Holder': 'Nome': 'Comune di Trento', 'Code IPA/IVA': 'c.l378', 'Update frequency': 'continuously updated', 'Format': 'json'		
Attributes			
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>HouseNumber</i>	civic number	string	Common
<i>address</i>	street address	string	Common
<i>city</i>	suburb of the school	string	Common
<i>name</i>	name of the school	string	Core
<i>SchoolType</i>	type of school (nursery, elementary, middle, high)	string	Core
<i>study paths</i>	the study paths offered from the school (comma separated)	string	Core
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

bikesharing_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>source</i>	https://dati.trentino.it/dataset/stazioni-bike-sharing-emotion-trentino		
<i>format</i>	json		
<i>timestamp</i>	2020-10-18		
<i>Dataset ID</i>	p_TN:9b9c14d6-ee20-4802-a274-4c17ac96cd5		
<i>Temi del dataset</i>	'Trasporti': ['4816 trasporti terrestri', '4806 politica dei trasporti'], 'Energia': ['6626 energia dolce', '6606 politica energetica']		
<i>Dataset publisher</i>	'Nome': 'Servizio Trasporti pubblici', 'Codice IPA/IVA': '0OK0PZ'		
<i>Release date</i>	18-11-2014		
<i>Modification date</i>	03-07-2017		
<i>Geographical Coverage</i>	Comune di Trento		
<i>GeoNames URI</i>	http://www.geonames.org/3165241		
<i>Dataset language</i>	italiano		
<i>Holder</i>	'Nome': 'Provincia Autonoma di Trento', 'Codice IPA/IVA': 'p_TN'		
<i>Upload frequency</i>	continuous		
Attributes			
<i>name</i>	name of the bike station	string	Core
<i>address</i>	geographical address of the bike station	string	Common
<i>city</i>	city of the bike station	string	Common
<i>id</i>	identifier	string	Core
<i>totalSlots</i>	total number of slot bikes	int	Core
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

skiResorts_static_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>lastModifiedAt</i>	2020-10-24		
<i>source</i>	https://www.skiresort.info/ski-resorts/trentino/		
<i>update_frequency</i>	unknown		
<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>	Name of the ski resort	String	
<i>Total lenght</i>	total number of kilometers available in the resort	double	
<i>km blue slope</i>	number of kilometers of blue slopes in the resort	int	
<i>km red slope</i>	number of kilometers of red slopes in the resort	double	
<i>km black slope</i>	number of kilometers of black slopes in the resort	int	
<i>price</i>	Price per person	double	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

hospital_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>type</i>	FeatureCollection		
<i>generator</i>	overpass-ide		
<i>source</i>	www.openstreetmap.org		
<i>copyright</i>	ODbL		
<i>timestamp</i>	2020-11-15T17:12:02Z		
<i>update_frequency</i>	no update		

<i>Geographical Coverage</i>	Provincia di Trento		
<i>format</i>	json		
Attributes			
<i>name</i>			
<i>addr:city</i>	Identifies in term of cities the position of the cinema	string	
<i>addr:housenumber</i>	Identifies the house number of the cinema	int	
<i>addr:postcode</i>	Identifies the postcode of the cities where the cinema is located	int	
<i>addr:street</i>	Identifies the street name where the cinema is located	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>id</i>	name that identifies a unique object	string	
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

bars_METADATA.json:

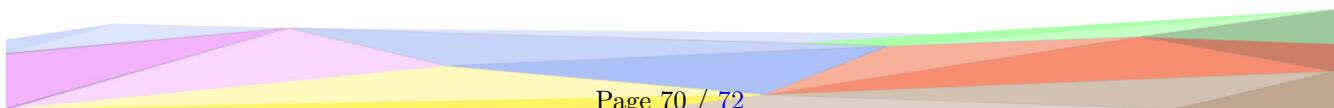
Dataset Properties	Description	Type	Data Definition
<i>source</i>	https://dati.trentino.it/		
<i>format</i>	json		
<i>relese date</i>	15-11-2020		
<i>Lingue del dataset</i>	italiano		
<i>Upload frequency</i>	unknow		
<i>Conforme a</i>	Standard: conforme a REST/JSON		
<i>description</i>	dataset merged from 'luoghi_e_punti_di_interesse_per_comune'		
Attributes			
<i>id</i>	identification code of the location	string	
<i>name</i>	name of the location	string	
<i>address</i>	address of the location	string	
<i>city</i>	city of the location	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

churches_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>source</i>	https://dati.trentino.it/		
<i>format</i>	json		
<i>relese date</i>	15-11-2020		
<i>Lingue del dataset</i>	italiano		
<i>Upload frequency</i>	unknow		
<i>Conforme a</i>	Standard: conforme a REST/JSON		
<i>description</i>	dataset merged from 'luoghi_e_punti_di_interesse_per_comune'		
Attributes			
<i>id</i>	identification code of the location	string	
<i>name</i>	name of the location	string	
<i>address</i>	address of the location	string	
<i>city</i>	city of the location	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

eldery_centers_METADATA.json:

Dataset Properties	Description	Type	Data Definition
--------------------	-------------	------	-----------------



<i>source</i>	https://dati.trentino.it/		
<i>format</i>	json		
<i>relese date</i>	15-11-2020		
<i>Lingue del dataset</i>	italiano		
<i>Upload frequency</i>	unknow		
<i>Conforme a</i>	Standard: conforme a REST/JSON		
<i>description</i>	dataset merged from 'lu-oghi_e_punti_di_interesse_per_comune'		
Attributes			
<i>id</i>	identification code of the location	string	
<i>name</i>	name of the location	string	
<i>address</i>	address of the location	string	
<i>city</i>	city of the location	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

libraries_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>source</i>	https://dati.trentino.it/		
<i>format</i>	json		
<i>relese date</i>	15-11-2020		
<i>Lingue del dataset</i>	italiano		
<i>Upload frequency</i>	unknow		
<i>Conforme a</i>	Standard: conforme a REST/JSON		
<i>description</i>	dataset merged from 'lu-oghi_e_punti_di_interesse_per_comune'		
Attributes			
<i>id</i>	identification code of the location	string	
<i>name</i>	name of the location	string	
<i>address</i>	address of the location	string	
<i>city</i>	city of the location	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

theaters_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>source</i>	https://dati.trentino.it/		
<i>format</i>	json		
<i>relese date</i>	15-11-2020		
<i>Lingue del dataset</i>	italiano		
<i>Upload frequency</i>	unknow		
<i>Conforme a</i>	Standard: conforme a REST/JSON		
<i>description</i>	dataset merged from 'lu-oghi_e_punti_di_interesse_per_comune'		
Attributes			
<i>id</i>	identification code of the location	string	
<i>name</i>	name of the location	string	
<i>address</i>	address of the location	string	
<i>city</i>	city of the location	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

physician_METADATA.json:

Dataset Properties	Description	Type	Data Definition
<i>source</i>	https://dati.trentino.it/		
<i>format</i>	json		
<i>relese date</i>	15-11-2020		
<i>Lingue del dataset</i>	italiano		
<i>Upload frequency</i>	unknow		
<i>Conforme a</i>	Standard: conforme a REST/JSON		
<i>description</i>	dataset merged from 'luoghi_e_punti_di_interesse_per_comune'		
Attributes			
<i>id</i>	identification code of the location	string	
<i>name</i>	name of the location	string	
<i>address</i>	address of the location	string	
<i>city</i>	city of the location	string	
<i>GeoShape</i>	object containing location information	GeoShape	Common
<i>type</i>	type of GeoShape (Line, Polygon, Point)	string	Common
<i>GeoCoordinates</i>	Coordinates of the location in EPSG 4326 projection standard	int[2]	Common

1.4.7.3 Variance respect Informal Modeling datasets

The main variance respect the informal modeling dataset is caused by the *Data preparation*, which is composed by two main activities: *Data alignment* and *Data formatting*. Initially, in *informal phase*, school datasets were composed of 4 differents files: `materne.json`, `medie.json` and `superiori.json`. These have merged in order to obtain the unique dataset `schools.json`. An important work has been done in order to guarantee that data are in agreement with ETypes. Thus, many attribute names changed name. Then, as a problem in data formatct coordinates has been found, all the coordinates are aligned in the EPSG 4326 standard.

1.4.8 Formal Modeling Evaluation

The last section of the Formal Modeling phase report the evaluation of the outcomes obtained in this phase, through specif evaluation metrics.