Dipartimento di Ingegneria e Scienza dell'Informazione

– KnowDive Group –

Codebook Template

Document Data:	Reference Persons:
17/12/2020	Pena Roncero, Javier (javier.penaroncero@studenti.unitn.it)
	Micheli, Enrico (enrico.micheli-1@studenti.unitn.it)
	Dal Moro, Devis (devis.dalmoro@studenti.unitn.it)

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

Revision History:

Revision	Date	Author	Description of Changes
0.1.1	11.10.2020	Javier, Devis, Enrico	Scope and Personas definitions
0.1.2	18.10.2020	Javier, Devis, Enrico	Competency queries definition
0.1.3	20.10.2020	Javier, Devis, Enrico	Inception Datasets description
0.2.1	25.10.2020	Javier, Devis, Enrico	Informal Modeling: EER definition
0.2.2	27.10.2020	Javier, Devis, Enrico	Informal Modeling: EER description
0.2.3	30.10.2020	Javier, Devis, Enrico	Informal Modeling: data level operations
0.3.1	15.11.2020	Javier, Devis, Enrico	Formal Modeling: ontology definition
0.3.2	15.11.2020	Javier, Devis, Enrico	Formal Modeling: data level operations
0.4.1	13.12.2020	Javier, Devis, Enrico	Data Integration: linking operations

1 Knowledge Graph Codebook

The first of the two sections, in the current document, contains the codebook of the whole KG (Knowledge Graph), including the description of all the data and information that it contains.

1.1 Knowledge Graph general description

This sub section aims to give a general description of the KG, reporting:

- the context/domain in which the KG lives and works;
- *The Problem* the KG aims to solve;
- How the KG can solve *The Problem*

1.2 Data level

The data level section aims to describe in details the (final version of) datasets collected and managed by the KG, with a description of each variable involved.

1.2.1 Datasets general details

In this section are reported the metadata at datasets level, so the metadata regarding the sources, the authors, the collection methods, and so on.

1.2.2 Datasets metadata documentation

In this section are reported the metadata at dataset attribute level, through a description of each variable involved in the datasets collected, specifying the variable types, meanings, value-set (possible values), and every other meaningful variable information.

1.3 Ontology level

The ontology level section aims to describe the underlying KG ontology, through the description of its elements at each level, reporting so the language, conceptual and schema resources used within it.

1.3.1 Ontology general details

This first sub section of the ontology level description, report the general details such as authors, sources and the description of external ontology eventually adopted to generate the final one.

1.3.2 Ontology metadata documentation

In this section instead, are reported the more specific metadata describing the single elements of the ontology (terms, concepts, ETypes and relations).

1.4 Knowledge Graph Evaluation

In the final section of this first chapter, the KG Evaluation is reported. It aims to describe, through specific metrics, the quality of the overall KG on different aspect, like domain coverage, usability, domain representation, and other meaningful aspects.

2 Knowledge Graph Development Process

The second chapter of this document aims to describe, in a detailed way, the KG development process. The sections below describe each phase of the KG building project, reporting for each phase, the description of the datasets and their evolution respect the previous phases, the schema construction which will generate the KG ontology in the end, as well as the description of the procedures adopted to manage the data and finally achieve those results. Moreover for each phase is reported an evaluation section, which aims to evaluate the quality of the results achieved at the end of each phase.

2.1 Scope Definition

While most smart services flourish around urban environments, hence the term *Smart Cities*, the facilities and in general rural aspects are left behind. While there might be some merits to this choice this hinders a quick and easy planning of activities that are carried out in such places.
We propose a system to obviate this lack, in the form of an assistant tool that can help plan out mountain hikes or bike trips, as well as lake visits and ski excursions. As such the data we will utilize will concern the currently existing mountain paths, operating facilities such as restaurants and alms, and existing cycle paths. We will restrict out geographical scope to the Trentino province, but the system will be set up in a way that it could be amplied provided an adequate dataset. As mentioned this context is not usually associated with the *Smart* world, it follows that while we require updated data this will not always be available, and we will have to rely on a few monolithic collections of data on aspects such as paths location and status.

In our current world, people like to invest their free time in travelling around the world, visiting interesting places and living enriching experiences. Unfortunately, the time needed in order to plan such activities is missing more and more and the risk is to waste our little free time and money in places that do not fulfill our goals and expectations, since information is sparse and confusing, making it difficult to have effective comparisons for a more aware planning. Let us explore some Personas:

Sara is an italian accountant living in Trento for the last three years. Her full time job requires her to invest most of her day sit on her office chair, in a close space. However, she and her partner love spending time diving themselves in the open spaces, within the marvellous environments nature has to offer. Fortunately, they are lucky enough to be always free from work during the weekends. Living in Trentino they have tons of possibilities and options, but since they didn't grow up there, deciding what is the next mountain to climb and which new places they can explore could be a demanding task during the week: information is sparse and hard to find. Sara spends whole nights in order to look up for them, browsing multiple sites. Moreover, on saturday morning they would like to have everything already planned, so to invest to the fullest every moment of their free time. This means that they do not only want to know the next lake to visit and/or a fulfilling hiking trail to go on, but also an accomodation for the night.

Sergio is a professional skier. He lives in Spain with his Italian wife, however he typically spends a lot of time in Italy during the winter for his training and competitions, so that his wife can be closer with the rest of her family. Moreover, he's always loved the mountains in Trentino and he considers their ski tracks among the european best for adjusting his fundamentals and improving his skills. The preparation requires him to properly diversify his work, so he integrates the ski sessions with hiking, bike and snowshoe. Unfortunately, he's still not very familiar with the so many options and he usually ends up doing the same training sessions in the same places, which is not good for an optimal training that requires diversification.

James is 21, a dedicated songwriter, singer and musician born in the North of Wales. He's been traveling around Europe since he finished high school, living up on the money he is able to earn doing gigs in pubs, hotels and bars during the nights. Moreover, he frequently goes outdoors to get his creative side inspired, channeling his admiration for nature into his music. While he admires and enjoys a snowy landscape he prefers the more lively aspect of summer mountain woods. Not afraid of long hikes he regularly embarks in long trips to reach the most isolated zones of the countries he visits. He is also willing to perform a selection of his repertoire, possibly his new additions, for the locals.

Selma is a retired woman living in Switzerland. She lives with her husband and they are a really active couple that like to do adventurous trips before they become old. they also want to go skiing with their grandchildren a few days, but it is march, and they don't know if they would have the possibility to skiing. As an alternative, they were also thinking of visiting some lakes and spending the night in a hut. Their grandchildren do not have much experience skiing, for this reason they would like to find a ski resort with a decent number of easy slopes. They would like to come to Trentino, check the possibility of skiing and potentially spending the night in a hut on the mountains, possibly close to a lake.

Name	Age	Interest	Usage	Description
(1) Sara	27	Mountain hiking and accommodations to spend the night with her boyfriend	Plan her sporty and romantic weekends on the mountains without too much struggle	Sara is an accountant working a full time job, all day sit in front of her computer. In the evening she gets pretty tired and surely she doesn't want to spend the night again in front of a screen to find out hiking trails and accommodation for the following weekend. Using the system, she hopes to find an optimal solution wrt position, time and difficulty of hike trails, hut in which to spend the night out with resources already present in it (e.g. food, toilet, blankets, heater,..), so she knows in advance what she needs to bring with her.
(2) Sergio	32	Ski tracks, Mountain hiking, Bike riding, snowshoe	Find all the tracks and places that can give him the possibility to have proper training sessions, while spending time in Italy with his wife	Sergio and his italian wife come to Italy during the winter sessions, so that she can spend some time of the year closer to her family. He looks up to this period of the year, because it's the one in which he has to focus more on training and competition, being a professional skier. That requires him being able to always find proper natural and/or artificial facilities in which he could properly dive into for an average of 6 hours a day to ski, walk and cycle. When he's in Spain, he has no problem in finding even new places to do that, but in Italy he always struggles and ends up in the same places every year.
(3) James	21	Mountain hiking, Huts	He needs to find long mountains itineraries where he can get inspiration from, as well as an audience	James has launched into a life path that is supposed to help him to achieve his dream goal: become a famous singer and songwriter. In order to do that, he feels like he needs to reach with his music a lot of heterogeneous people. That's why he set himself on a journey around Europe, doing gigs in the most disparate bars and pubs. The next country to visit is Italy and, as he did many times before, he wants to find a place in nature where he can get inspiration for one of his next pieces, with preference for non-snow environments. Any huts on the area are a chance to get feedback on his work in progress pieces.
4 Selma	64	Huts, Lakes	Spend the night in huts in the mountains	Selma is retired, but still enjoys having an active life. She feels like the north of Italy could be an interesting option to spend some days in nature. She has a strong preference for having picnics in lakes and reachable by foot from huts in the mountains.

2.2 Inception

2.2.1 CQs definition

Having imagined possible users which are going to be interested in our system and put their needs on paper, we are now able to identify what they could ask to the system when it's going to be up and running. In order to get along with this process, we need to consider also the possible datasets and sources from which we're going to be able to get the sufficient data to answer them. In combination this allows the formal definition of the competency questions you see listed below. By manually processing them, it's possible to extract the single lexical items, grouping them, so to identify the first approximate sets of entity types and relative attributes which will constitute the core, common and contextual data of our knowledge graph.

Pers	#	Question	Action
Sara	1.1	Give the list of all hiking trails that takes up to Cima d'Asta	System will search all the hiking trails that has at one end point Cima d'Asta and returns info such as geo-spatial points, length, difficulty, equipment needed, possible dangers
Sara	1.2	Give me the list of huts which can be found along the trails on the way to Cima D'Asta with toilet, beds and stove	System will search all the huts within the requested area, filtering wrt the needed facilities (e.g. food, toilet,...), if provided
Sara	1.3	Give me the list of B&Bs which can be found in Riva del Garda with the info about their provided offers	System will search all the B&Bs within the requested area, filtering wrt the needs (e.g. bio-food, not-shared toilet,...), if provided
Sara	1.4	Find a hiking trail of moderate difficulty which gives the possibility of doing it during winter, with great views and takes no more than 5 hours	The system will search all the hiking trails that do not require special equipment and/or measure of average length/time to complete, also suitable for the desired season.
Sara	1.5	Find a hiking trail of moderate difficulty, which no high stamina required, which takes no more than 5 hours to reach the side of a lake/river and possibly an accommodation has to be near the lake/river	The system will search all the hiking trails that do not require special equipment and/or measure of average length/time to complete, where there is the possibility to come across to a lake / river and near them a hut or a B&B has to be present

Sara	1.6	Find all the hiking trails to reach the top of Cima della Nara that has a circular path (i.e. coming back to the start) in less than 10 hours and do not require special equipment.	The system will search all the hiking trails that are properly marked, which do not require special equipment and can offer the possibility of a “circular closure”, i.e. following “one going up”, another “going down”, while don’t sharing more than 20% of the same track and in total they do not require more than 10 hours
Sara	1.7	Find all the hiking trails taking to the Tre Cime di Lavaredo which are opened in the autumnal season	The system will search all the hiking trails having as an end point the provided area, which are opened to be covered during the selected season
Sara	1.8	Find all the snowshoe trails taking to the Tre Cime di Lavaredo which are opened in the winter season and can be traversed by using proper snowshoes	The system will search all the hiking trails having as an end point the provided area, which are opened to be covered during the selected season by using the selected equipment
Sergio	2.1	Give the list of ski resorts in Trentino ordered by number of kilometer of black slopes currently open.	The system will output the list of options of ski resorts in the area ordered by the number of black slopes
Sergio	2.2	Give me the list options of bike trails in Trentino that are longer than 10 Km, with a high elevation and requires expertise.	The System will search the lists of bike trails and filter them according to the length requirement
Sergio	2.3	Retrieve the list of hiking trails with a high difference bigger than 300m, great landscape and open in winter.	The System will search the lists of hiking trails and filter them according to the slope restriction
Sergio	2.4	Give the longest biking trail and the accommodation/huts by the route	The System will search the lists of bike trails and filter them according to the length requirement
Sergio	2.5	Find all the challenging off-road biking trails which are primarily thought for mountain bike and are clean to be traversed during March	The system will search for all the mountain bike cycling tracks, which have a difficult at least equal or greater than average and/or takes more than 3 hours to be completed, which are accessible in the selected period of the year

Sergio	2.6	Find all the challenging snowshoe paths which requires good equipment, require a good technique, and are open during February	The system will search for all the mountain bike cycling tracks, which have a difficult at least equal or greater than average and/or takes more than 3 hours to be completed, which are accessible in the selected period of the year
Sergio	2.7	Find all the ski resorts, with more than 30km open of red or black slopes and open during the night	the system will retrieve all the ski resort with the desired rating, and the constrains about time and length.
Sergio	2.8	Find the contact information of all the 4 stars hotels close to the ski resort Madonna di Campiglio	the system will retrieve all the hotels contact information close to the ski resort.
James	3.1	Give the starting point of the route ”Giro della Madonnina di Besta”	The system will provide a list of hiking trails ordered by length normalized wrt. its declared difficulty (if present)
James	3.2	Give the list of hiking trails below 800m	The system will provide a list of hiking trails having as an end-point a geospatial locality which has an altitude of less than 800m
James	3.3	Give the list of hiking trails with huts in their path	The system will provide a list of hiking trails which foresee huts in the nearby area (i.e. not far more than 250m from the trail path)
James	3.4	Give me the alpine dairies which are on the way to Cima d’Asta	The system will provide a list of alpine dairies which are nearby some trails which go up to Cima d’Asta
Selma	4.1	Give the adress of Hotel Ai Spiazzi	The system will output the list of huts available close to te preference
Selma	4.2	Give the list of lakes in the province with all the necessary information	The system will search for all the lakes in the prompted area, providing all the related info that could come useful
Selma	4.3	Find the hiking trails that start from lago di lamar, having a low difficulty and the possibility of being assisted by a proper guide	The system will search for all the relatively easy (i.e. less than two hours, no special equipment needed) hiking trails nearby the selected lake and that can be assisted by a guide
Selma	4.4	Find all the B&Bs near lago di lamar offering a breakfast with typical drinks and foods under 75€/night	The system will search for all the B&Bs near the prompted area, claiming to offer local food and beverages and providing the service for a total under 75€ per night

Selma	4.5	Give the ski resorts open today ordered by the number of blue slopes	The system will output the list of available resorts ordered according to the desired parameter
Selma	4.6	Find all the B&Bs and huts closer than 2 km from a lake	The system will search for all the B&Bs near the prompted area.
Selma	4.7	Find all the hiking routes with moderate or easy difficulty, with no equipment needed, no experience, open in March and ideal for kids	The system will search for all the hiking paths restricted with the information provided.

NUM	**TYPES**	**Attributes**
1.1-2; 1.4-7; 2.3; 3.1-3; 4.3; 4.7	Hiking Trail (Trail)	distance, difficulty, route, elevation profile, stats, equipment, opening season
2.2; 2.4-5; 4.7	Biking Trail (Trail)	distance, difficulty, route, elevation profile, stats, grounds, opening season
1.8; 2.6;	Snowshoe Trail (Trail)	distance, difficulty, route, stats, equipment, grounds, opening season
1.2; 1.5; 2.4; 3.4; 4.6;	Hut (Accommodation)	name, location, room options, commodities, contact information, opening months
3.4;	Alpine Dairy (Accommodation)	name, location, address, contact information, opening months, services
1.3; 4.4; 4.6;	B&B (Accommodation)	name, location, address, room options, commodities, contact information, stars, prices
4.1;	Hotel (Accommodation)	name, location, address, room option, commodities, contact information, stars, prices
2.1; 2.7-8; 4.5;	Ski Resort	name, location, stars, prices, open/closed, slope information (blue, red and black slope), opening hours, opening months
1.5; 4.2-4;	Lake	name, location, surface, max depth

2.2.2 Initial Datasets description

The initial collection of datasets gathered in this phase are thought to answer to the necessities of the personas explained in section [2.1]. Naturally this includes an heterogeneous and wide spread set of information resources. Additionally, to narrow down, i.e. be more specific in the the dataset exploration and collection, we took into account the competency queries (CQ) elicited in section 2.2.1.
First, to fit the queries related to accommodation we decided to get hotels, bed and breakfast and apartments information from the Trentino open data platform, respectively: hotels Trentino and BB and apartment Trentino. These are two datasets in XML that we will have to merge in the following steps.
To get the data related to ski resorts and their characteristics, we decided to scrape the ski resort website. This webpage is updated live.
Additionally, we decided to get the information concerning huts by scraping the webpage outdooractive.com. From the same website, updated live with the contribution of a live digital community, keeping metadata with respect to every instance of interest, we were able to fetch the data used to answer the queries related to trails as well: hiking trails, biking trails and snowhoe trails. As for the biking trails we can consider to use also the data provided by the Trento website which is based on OpenData datasets bike trails. Unfortunately, it just contains the biking trails infrastructures within the city of Trento. Thus, we don't know if it is really worth it, considering also our targeted users which we modelled starting from the Personas. To have more solid information which we can use and integrate with the data regarding trails fetched from outdooractive.com, we've decided to consider also the official ”Società Alpinisti Tridentini” (SAT) registry, which in addition it'll be able to give us also the specific identifiers for more legit and officially recognized by the corresponding authorities and national organization (e.g. CAI) trail paths.
For the information regarding lakes and rivers we decided to go again with Open Data Trentino retrieving names and measurements from this RDF resource.

2.2.3 Datasets metadata documentation

Regarding the accommodation information, the hotel's dataset contains the validity date of the data, the name of the hotel, address, type of service, number of starts, phone number, webpage, fax, number of rooms, number of beds, the price per room, maximum capacity, and some additional information of each hotel such as if it has TV or elevator. In the other dataset with apartments and BB, we can obtain the city, type of accommodation, name, address, phone number, open season, altitude, number of rooms, number of beds, rating, number of rooms, and price. The former dataset was created in 2014 and last updated in 2017, and the latter was created and last modified in 2014. These datasets are concerning since they might be outdated. As another option, we've thought to take into consideration using and/or integrating the data regarding this resources which can be fetched from Booking.com APIs.

Concerning the ski resort information, as we said before, we are going to scrape the data from the skiresort website, from this website we will obtain the name of the ski station, the number of slopes, specifying how many kilometers there are of each type of slopes (red, black and blue). Additionally, we are going to recall the fee and how many ski lifts there are and the number of kilometers/lifts available, and if the resort is open or not.

Regarding the data about lakes we'll be able to have the name, dimension (perimeter and area) of the lakes. Hopefully, we'll be able to get additional information regarding it: both structured, like average temperature of the water and geographic point which delimits it and unstructured (e.g. an information description of the lake itself). The specific RDF resource for lakes and rivers we decided to focus on foresees its author and maintainer in the person of Zanolini Roberto. The resource was originally created on the 12th of September of 2014 and, up to now, this represents also the last date it was modified. The format of this dataset is RDF.

By scraping the website outdooractive.com and merging it with the data provided by the SAT registry,we plan to extract useful information about hiking trails and huts in Trentino. With respect to the trails, it'd be consider almost vital to collect from the web pages at least the following: name, end points, altitudes, length, duration and declared level of difficulty. Supplied reviews seem to be a bit out of our scope of work, i.e. could be conditioned by the time the trail was traversed by the reviewer and could violate the privacy of the author, still the grade in the typical star-like fashion (e.g. *"x out of 5"*) could be integrated, being a more general data. Important information regarding the current state of the trail (e.g. "Closed") needs also to be gathered from the website. Concerning the huts, name, position, elevation, sleeping berth and provided supplies and/or commodities seem the main target. In the case of trails and huts scraped from the website, we'll be able to attach to each single instance of them the corresponding metadata regarding the creation and last modified dates wrt. the provided resource we're getting the value from.

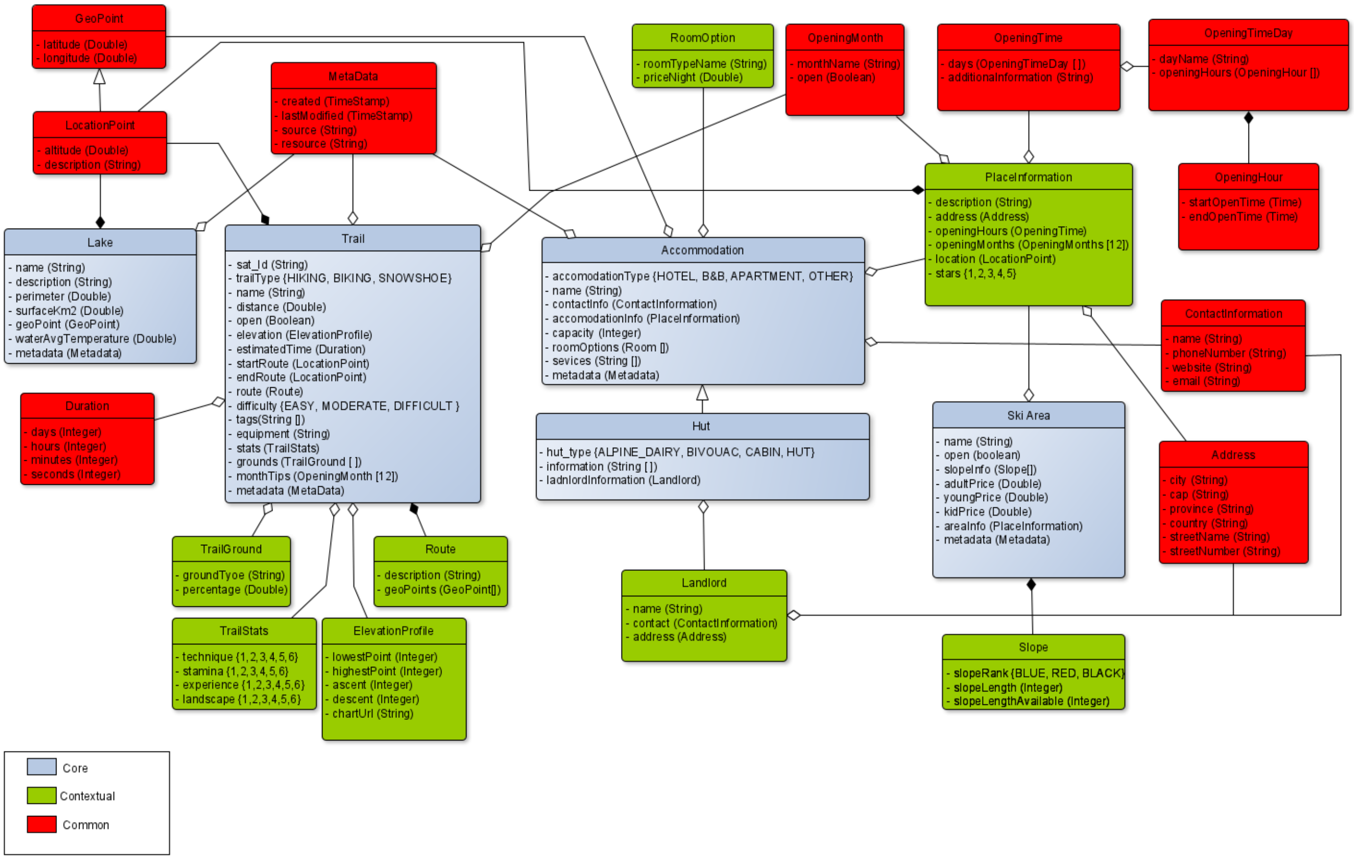
2.3 Informal Modeling

We've started this phase by taking as main input the generalized queries in the table above along with the other one resuming what were sure to be the core entities in our domain. Using a modelling tool (yED), we were able to discuss and analyse more deeply the structure of the objects which will compose our knowledge base. Note that the process has been taken forward thanks also to the collected data that gave us a more clear idea of what we actually are able to get in response to the listed queries. Hence, it has came almost naturally to revisit the queries as well, after the completion of the first proposal for the EER model. The EER model is very similar to the ER model, but it allows for more flexibility. For instance, within its schema we were able to distinguish the three main categories of possible entities:

- Core entities: the most important object classes regarding our scope of work, which appears as the direct instances as a response to the defined queries. In the schema they appear in blue.
- Contextual entities: object classes which are not essential, but come in support to the other ones, adding some auxiliary features to them. In the schema they appear in green.
- Common entities: object classes mostly related to time and space, which are possible to be found also in other domains. In the schema they appear in red.

2.3.1 Schema level

We put below the first proprosal for the EER schema.



2.3.1.1 ETypes and EER Model definition

The main core entities that were explored are the following:

- ***Lake***: we think that the most important attributes wrt. this type of entity are for sure the *name* and the *location* of the lake, which we hope to describe with its geographical points establishing its *perimeter*. We plan to have also some small *description* and maybe a few technical data as well, but that can be interesting to tourists too (e.g. the *surface* extension of the lake, the *average temperature* of its water)

- ***Trail***: for the trail it's important from the beginning to state the fact that we distinguish three main kinds of trails: ***Hiking Trails***, ***Bike Trails*** and ***Snowshoe Trails***. We were able to merge them in a uniform and standard structure, so it didn't seem worth it to have three subclasses in this informal modeling phase. Having said that, one of the most important attribute is for sure the one able to distinguish the *type* of trail. Note, that in a more formal definition, this could be removed to leave the space to three different entities attached to their specific concepts sharing the same base common schema.

  Another relevant attribute is the *SAT* (Società Alpinisti Tridentini) *ID*, which is put on "more or less" regular checkpoints along the natural path to help out the climbers and not get lost. Notice that not all the trails in our datasets are "official", i.e. they are not all put in the SAT registry, thus a part of them won't have this ID. *Distance*, *elevation profile*,*route* to be followed with its geographical points, level of declared *difficulty* and types of *grounds* with their relative coverage on the trail are important elements as well that will be part of this type of core entity. Don't forget that in any case it'll be vital to state also the two *end points* of the trail (i.e. where it start and where it ends). Auxiliary information will be more detailed within the related contextual entities or appear as fashionable to have, but are clearly additional. Still, it seems worth it to mention here the most relevant to the subject which are the one encapsulating the *suggested months* in which to go on the specific trail and others suggesting *equipment*, giving more *technical stats* (e.g. levels of *experience*,*stamina*,*landscape*,... all graded between 1 and 6 by a community[1]) or provided string *tags* that eventually will come helpful in order to perform some associations and answer to more peculiar queries (e.g. "trail holding elements of cultural/historic value" marked as *cultural/historic*)

- ***Accommodation***: as happened for the case above, also for this entity we came up with a uniform structure encapsulating elements relevant to the tourists regarding different types of accommodations which are ***Hotels***, ***Bed Breakfast***, ***Apartments*** and ***Huts***. Except for the ***Huts*** in which there could be specific information relevant to hikers, in the other cases it didn't seem worth it to have three subclasses. Hence, as done for the other entity, we'll need an attribute *type*[2].

  With respect to what could be the possible tourists' concerns regarding the accommodations and the data we were able to collect, we found *name*, *location*, *address* and *contact information* to be the most relevant characteristics for the accommodation. Of course it'll be nice for the tourists to have info related to the *capacity* (e.g. how many bedrooms there are) and the *room options* (e.g. the type of available bedroom options and the prices related to them) as well.

  Specifically for ***Huts***, we are going to distinguish sub-types of them within the attribute *hut type*. They are going to be alpine dairies, bivouacs, cabins, or simply generic huts (where we don't know how to classify them any further). This kind of distinction will help the user to understand the provided services and features

[1] this suggests why it's important to have metadata attached to each instance as well, explicitly saying where and when the values where created and last modified

[2] Notice that in the EER we'll define this as enum attribute with 4 possible values: HOTEL, BB, APARTMENT and OTHER. Huts can fall into the OTHER category

of the facility. Furthermore, since the hut represents a more private form of accommodation, having the *landlord's contact information*, so ask him/her specific questions regarding availability, supplied commodities, difficulties, etc. can be important as well.

- ***SkiArea***: even if having some similarities with the Accomodation, we decided to have a separated entity for this object class. The shared attributes are more related to generic *place information* (e.g. *address*, *contacts*, *location*) for which in any case we created a contextual entity. However, the SkiArea is characterized by the kind of *slopes* that it offers, their total and currently available length, distinguishing them with respect to their declared difficulty level[3]. *Prices* for using them are also relevant to the case and typically they're different between children and adult, so it's okay to distinguish them in two different attributes.

2.3.1.2 Variance respect CQs definition

After having modelled our *etypes* with their related properties, it's been easier and almost natural to perform a revision of the previous phases, with a major focus on the competency queries definition. In fact, the model has required us to go more into deep in what knowledge is actually at our disposal and what kind of inference tasks are going to be feasible. Realising that, we were invited to modify accordingly the queries: removing the ones which didn't appear to be either ”appealing” or unfeasible anymore, but also by adding and/or making more technical some pre-existing queries with their corresponding filters which are encapsulated within them. Undoubtedly they are indeed linked to what kind of attributes an entity exposes and, along this phase, this has become clearer.

2.3.2 Data level

During this phase, a cleaning and study of the datasets have been carried out. The main task was to select which information was going to be used to answer the queries showed in section 2.2.1. Moreover, This assignment was done in parallel with the development of the EER schema presented in section 2.3.1. This means that the number of datasets has not changed much in this phase, in fact most of the effort has seen us busy designing and writing code to automatize as much as possible the data processing. Our target with respect to that was to clean and filter the collected data, so that in the following phases we'll have them already as much compliant as possible to our schema. Note that some tasks needed to be performed manually. All details of this work are going to be better explained in the following sections.

2.3.2.1 Datasets management process

During the Informal Modeling phase the datasets collected in the previous phase have been filtered and managed in order to obtain more suitable sets of data. In this section are described the procedures adopted to obtain that result. Along the L4 informal model definition, we have taken care of the trail datasets, so to have them compliant with the data structure of the *etype* ***Trail*** which has been formally defined in this very phase. Snowshoe, Biking and Hiking Trails data scraped and fetched from outdooractive.com were all cleaned up, filtered and re-mapped into this new standard form. The same process has been performed for the data fetched from the SAT registry. Both processes were run as automated Node scripts which have been uploaded in the appropriate subfolder of the github repository. Note that having a uniform representation comes at a price: unfortunately this has lead to have some blank/unknown value for a few properties in some instances where the specific data was not available. However,

[3] In the technical language, blue slopes are the ”easy” one, red slopes are for intermediate skiers, while the black ones are for the most experienced skier

this is not a ”downgrade” with respect to its original representation, meaning that the value was not present there either if we weren't able to re-map it its new structure. Hence, this filtering and re-mapping process seem to be an upgrade in terms of what and how it can help within reasoning, modelling and inferring scenarios.
A similar process has took place in order to re-structure and clean the data regarding huts still fetched from outdooractive.com. Unfortunately here, a few steps have required manual intervention (e.g. data written in pure natural language concerning opening times that needed to be read, analyzed and re-mapped into our more useful structure), but a high burden of the ”cleanup” has been dealt with automated scripts also in this scenario.
Additionally, the accommodation dataset and the ski resort one have also been updated. The former has required an intensive cleaning process in order to be compliant with the EER schema.

2.3.2.2 Datasets metadata documentation

In this phase we have started to put more in concrete the concept of the metadata, meaning that it has become clearer to us the need of attaching a set of metadata to our datasets. They aim at reporting the provenance of the data themselves, as well as the relevant *DateTime* values attached to them, like for instance the last time the dataset was modified or when it was created and by whom. Another level of the metadata targets the structural description of the instances within them, which at the end of this phase will look very close to the structures depicted in the schema at 2.3.1.

2.3.2.3 Variance respect Inception datasets

As described in the previous section 2.3.2.1, the datasets were hugely processed in this phase, so to take leverage of it in the next steps. Basically, we've tried to make it as much compliant as possible with the EER schema definition presented above at 2.3.1, regardless of the fact that the data fetched from some sources will miss some of the cited properties. Hopefully, in the next phase, we'll be able to make some assumptions and. based on them, start the process to integrate our data fetched from different sources, so to have as less missing information as possible.

2.3.3 Informal Modeling Evaluation

For this phase evaluation we resorted to sourcing adequate entity schemas to match at the least our core entities, plus some common and context entities. This was deemed necessary as the provided reference schema did not correspond in an adequate manner to our produced SKG. This is most likely the result of our decision to specialize the application to a subset of the Tourist-Facilities context, leading us to expand on a smaller subset of entities and at the same time to ignore other entities used in the more general context.
The sourced schemas come from schema.org and disit.org and relate to lakes, trails, accommodations, ski-resorts, place, location, duration.
The evaluation was thus executed on the properties only, since the part about commonality of entities was not applicable in our case.
Another unexpected step was finding how to properly match these properties. For example, the accommodationInfo property in our Accommodation entity is a property that refers to an entity that has itself other properties such as address and location, the last one of which has again properties of its own such as elevation, and further down the inheritance line latitude and longitude. The Accommodation entity in schema.org has both address, latitude and longitude as text or numerical data properties. It thus means that the property matches, partially, but not perfectly.
Assuming the property matching when one of its parts matches the property of the respective entity, these are the results of the evaluation calculations.

	Lake	Trail	Accommodation	Hut	SkiArea	PlaceInformation	Location	Duration
Coverage	0.06	0.35	0.09	0.2	0.04	0.08	0.22	0
Flexibility	0.08	0.65	0.03	0.09	0.03	0.04	0	0.33
Extensiveness	0.07	0.32	0.03	0.07	0.02	0.04	0	0.25
Sparsity	0.87	0.65	0.79	0.69	0.92	0.84	0.63	1

We can see that overall the first three metrics have relatively low values while the last has values closer to its upper limit 1. We think this means that overall the reference schema is still not completely adequate, but not necessarily that our produced schema is invalid. For example considering the Extensiveness we get very low values which might seem to indicate our contribution to the knowledge graph is limited, but it depends heavily on the amount of properties the entities in schema.org have, as considering the trail entity which comes from disit.org instead has relatively higher values for all the first three metrics.

Another consideration can be made on the Duration entity, here used as example for a common entity, where the implementation in the reference schema was drastically different than our implementation at its core. Where the reference schema simply uses a numeric or text value to express a ”duration” we use four properties describing days, hours, minutes and seconds, thus while the entity exists in both schemas there are no common properties leading to values that seem to indicate a lower commonality than it actually is.

2.4 Formal Modeling

After completing the Informal Modeling phase and clarifying what will be our entities, attributes, and relationships, we started the fourth phase of the i-Telos methodology: the formal modeling phase.

2.4.1 Schema level

At the schema level, we've focused on the translation of the EER presented at 2.3.1 into a formal SKG wrt. OWL definition taking leverage of tools like Protégé, but also using the platform provided by the professors to explore the UKC and the concepts defined in them. These lasts were used in order to add L1, L2 annotations in attachment to the formal definition of the Etypes. Whenever a concept was not defined or not very compliant to our needed semantics, a specific importer tool still provided by the professor has come to its use, so that we could define our own concepts and also relate them to pre-existing ones through IS_A relations to map them within the whole concept tree.

2.4.1.1 Ontology definition

Starting from the EER model we used the KOS UKC explorer to find suitable concepts matching both the entities and data properties defined in the previous phase. This meant going through different meanings of the same word at times, or considering synonyms of the term to find the proper concept in case it was already inserted in the UKC under a different identifier. Once we established which concepts were missing we defined them in a spreadsheet to be fed to the KOS API. This included both the definition of the new terms and the definition of the relationships between these and the already existing concepts, as well as relationships between new concepts.
As you may imagine, we've defined new concepts specific to our domain of work, e.g. *Blue slope*, *Red slope*, *Black slope* which was clearly unimaginable to found them already there, otherwise we would have need to settle for something too much generic,i.e. which on the concept tree would be at least one level above (e.g. Slope). The same could be said about *Hiking/Biking/Snowshoe Trails* with respect to the more generic concept of "simply" *Trails.* But in other cases, we have been surprised to find that some definitions were instead too much specific, like for instance the case of the *url* concept, which was reported as "*the address of a web page on the world wide web*". It is an appropriate definition, but maybe too much specific considering the case that it wouldn't fit for an url giving the web location of another kind of resource which is not a web-page (e.g. a png image, an xml document and so on..). A more generic definition for *url* which comprehends also these semantics would be "*string schema which allows to uniform locate a resource*". That's one reason why we decided to overload some pre-existing concept word with a new (and in this case more generic) meaning. The other reason is the one that sometimes all the possible concepts attached to a word are missing the one you're actually looking for (e.g. *email* with respect to the email-address, or enumeration as in the "*data type in which it's explicitly stated the range of accepted values one by one*"). With the full list of concepts needed which were found but also defined and then imported, we've proceeded to define the ETypes in Protégé starting from the template SKG obtained from the platform API. This consisted in creating Entities following the given naming format which references the concept they are instance of, then the data properties for each entity again following the given naming format. At the end of the definition using Protégé, not only the name appears linked to the UKC concepts, but also the attached L1-L2 annotations are mainly connected to them, i.e. reporting in the *label* the word for the related concept, in *GID* field its Global Id, while the "gloss" (i.e. semantic) was put as the *comment* annotation. Another relevant annotation was a flag used to state if the definition regards an EType or not, simply stated as "*isEType*".

2.4.1.2 Variance respect to the EER Model

During this phase, some small readjustments have been made to the EER model presented during the previous phase. **NOTE**: the model presented in the section 2.3.1 with its attached description takes already in consideration also the modifications performed in subsequent phases. Hence, you'll find the end result of the process described below, but we hope it's easy to get an idea of how and why we needed to perform this changes from our explanation.

The first modification was made after comparing all the information available in the datasets and the old EER, after that, a few information was found missing. For instance, we realize that typically we have also altitude information in attachment to an accommodation or a ski-resort "*PlaceInformation*", thus we switch the type of the attribute "location" from a raw and basic *Geographic Point* to a more specific and tailored *Location Point*, containing also a description string for the location. This leads us to remove description from *PlaceInformation* to avoid useless redundancy. For the same reason, we avoid to put name in *PlaceInformation* too, while we found that in the entity *ContactInformation* it was surely needed.
Moreover, we decided to convert *Slope* and *Route* entities, which were both core entities during the previous phase, to contextual entities, since the information provided by them, even if essential to know, is not always there and logically can be put at the same level of *RoomOption* with respect to *Accomodation*.
As a way to simplify and clean the EER we modified the content of the *Slope* entity. In the previous phase it consisted of a enumeration of the different difficulties of slopes,with their available length. This design was long and chunky. During this phase we decided to simplify it by generalizing, having multiple instances of the same entity type where each one of them can focus on a particular kind of slope (e.g. blue, red or black) highlighted by an enum attribute.
Minor restructurations which are not worth to be mentioned here and were mainly due to partial knowledge of the domain, have been taken care of following a similar guideline with respect to the one that can be extracted from the elicited modifications above (e.g. just to mention one: *Duration* was not originally defined in the EER, even if we thought that to each *Trail* would have been good to attach an estimated time to complete it).
To conclude, in some situations, the concept searching in the UKC has brought us to decide for a renaming of some Etypes and/or data properties, which could have been wrongly mistaken for something else. For instance *Ski Area* was originally defined as *Ski Resort*, but as you may realize the concept of a Ski Resort is typically slightly different from the one of the EType we're trying to map which mainly revolves around its ski-facilities.

2.4.2 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.

2.4.2.1 Formal Modeling datasets management

In this phase, at the data level we were able to took leverage of the cleaning, filtering and remapping process performed via scripting (using python and/or JS) and manually in the previous phase. One of the changes to the datasets came with respect to the lake dataset. This dataset was originally in RDF format, the first reprocessing step was to transform it into CSV to improve the readability. Additionally, this file contained the location of the lakes in shapefile format, which was not relevant for the project since it was needed to have coordinates. For this reason, with the help of the professor Fausto Giunchiglia we got in contact with the creator of the dataset, Pavel

Shvaiko, who showed us how to obtain the coordinates from the shapefile. These shape files were later transform to actual coordinates in a csv file.

2.4.2.2 Datasets metadata documentation

In this phase we've started to actually write down in a more formal way the metadata in JSON files which are compliant to the specification the professors gave to us. They all reference each other and the idea is to have a "master" file, where all the others regarding the SKG, LKG, DKG, but also with respect to the imported dataset files using KarmaLinker in the next and final phase can be reached and explored from it. As previously discussed, it'll be our duty to transcribe information regarding both the provenance of our data and the structure of it. We plan to insert the creation and last modified time of our datasets, the author(s), links to get the raw resources, description of them as well as description of the entities, data and object properties considering their values not only in the input files, but also in the SKG formal definition. Furthermore, with respect to the latter, it's important to highlight the mapping between the two.

2.4.3 Formal Modeling Evaluation

Again like in the previous evaluation section of the informal modeling phase we had to resort to using single entities as reference material for the process, finding it unsuitable to use a pre-existing complete schema.

Our translation from informal to formal model lead to a not too dissimilar pair of models, but the differences we introduced following adaptations to best fit the development process meant that there are some slight variations in the values obtained in this step.

	Lake	Trail	Accommodation	Hut	Ski_Area	Place_Information	Location	Duration
Coverage	0.08	0.35	0.18	0.18	0.09	0.14	0.33	0
Flexibility	0.22	1	0.09	0.11	0.07	0.02	0.33	0.16
Extensiveness	0.17	0.42	0.07	0.08	0.06	0.01	0.19	0.14
Sparsity	0.89	0.7	0.78	0.76	0.89	0.69	0.58	1

Similar to the analysis of the results in the informal phase the values in this phase are generally low for the first three metrics although less so, probably due to this phase model using more inherited properties from generic entities, leading to a higher base commonality with the reference schema.

The Trail entity has again the higher values in those metrics, a result of the simpler modeling with less properties in the reference schema used.

2.5 Data integration

2.5.1 Data integration operations and tool

The operations in this last phase of the iTelos methodology has seen us involved in exploiting an API supplied by the provided platform (specifically *GET /data/exportTypesRDF*) to download the complete and full ontology uploaded as last step of the formal modelling phase. Of course, it was not any different from the one uploaded (except some small and not relevant changes in the format which doesn't affect the content in any way), but within this step we were able to double check again that all the etypes, object relations and dataproperties were correctly there. Having done that, it was time to go to karmalinker and importing the ontology file. Dataset by dataset we tried to perform the actual linking and of course some minor changes were still need due to be done. Nothing too drastic, but either because the tool wasn't interpreting correctly some array of values which were supposed to be data properties or because we forgot to adjust and restructure appropriately some fields in the previous phase (and now it's obvious to realise that), we had to. The details of these changes are briefly described in the apposite paragraph of this section.
All the linked dataset were exported as EML, model and RDF files, the first format was then needed to be imported on KOS. The RDF comes instead as a good representation to explore then the final ontology composed of all linked datasets in GraphDB.

2.5.2 Variance respect Formal Modeling datasets

The most important change made to the datasets has involved the proper splitting of some of them in order to not have huge files to import (the platform is still under development and, not knowing the amount of dedicated computing resources, we would avoid stressing it too much) and to not have etypes which are "siblings" in the same dataset (e.g. cabins and alpine dairies were all under the same dataset, because at the beginning we thought to have a type property to distinguish them, but then along the way we realize it would have been better to separate them in different and more specific types). Minor changes were due to difficulties of the platform to deal with specific array of strings where ideally we would have like that each of those values represented a data property attribute for an instance of one of our etype instances. This seemed not feasible and the platform wasn't responding in a desirable way by looking at the output file (e.g. instead of finding the array of string values related to the property of an instance in the .eml file, we found multiple instances which had the different properties). Scenario by scenario, we tackled the problem either by splitting the single column containing the array of raw values in multiple columns (e.g. *services* became *serv_1*, *serv_2*,.. and so on) or by concatenating all the strings in one (e.g. *services* array of strings can become a single string with some delimiters separating them[4]).

[4] Of course this can represent a careful choice to be made when you need to perform some specific queries, i.e. implying more difficulties on that side